

ONLINE SHOPPING SYSTEM

Project submitted to the Department of Computer Science

ST.MARY'S COLLEGE(AUTONOMOUS),THOOTHUKUDI.

Affiliated to

ManonmaniamSundaranarUniversity,Tirunelveli

In partial fulfilment of the requirement for the award of the degree in

BACHELOR OF COMPUTER SCIENCE

Submitted by

A.Alagu Shiny - 18AUCS02

E.Elvinasa - 18AUCS08

M.Jemimal - 18AUCS10

A.Mathuri Shenbagam - 18AUCS27

A.Siluvai Merywin Delsiya - 18AUCS43

Under the Supervision and Guidance of

Mrs. S .Gnanathangam M.C.A., M.Phil.,(HOD)



DEPARTMENTCOMPUTER SCIENCE

ST.MARY'S COLLEGE(Autonomous)

Reaccredited with "A+" Grade by NAAC

THOOTHUKUDI.

(2020 -2021)

CERTIFICATE
ST. MARY'S COLLEGE (Autonomous),
THOOTHUKUDI
Re-Accredited with 'A+' Grade by NAAC (4th cycle)
Affiliated to
MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE
This is to certify that project entitled
ONLINE SHOPPING SYSTEM

Submitted by

A. Alagu Shiny	18AUCS02
E. Elvinsa	18AUCS08
M. Jemimal	18AUCS10
A. Mathuri Shenbagam	18AUCS27
A. Siluvai Merywin Delisiya	18AUCS43

In partial fulfillment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.

S. Gnaly
Guide

S. Gnaly
Head of the Department

Submitted for the examination held on 15/04/2021

Jeetha
Examiner

Lucia Rose
Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ONLINE SHOPPING SYSTEM



ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First and Foremost. We would like to express our gratitude from the bottom of our heart to the Lord for his countless blessings which gave us strength and confidence in successful completion of our project.

Our sincere thanks to our principal Dr .Sr. J . .Lucia Rose M.Sc., PGDCA,M.Phil.,Ph .D. for providing us necessary infrastructure to undertake our project work successfully.

We are ever so grateful to Mrs. S .Gnanathangam M.C.A., M.Phil., (H.O.D) of computer science department for her prompt notification and guidance which helped us in successful progress of our project.

Our Heartful thanks to our project guide Mrs. S .Gnanathangam M.C.A., M.Phil., (H.O.D) for her encouragement, guidance and beneficial criticisms which significantly helped us in Scheming the project in a constructive manner.

We are highly indebted to our lectures Mrs. J.AnilaMaily M.Sc., M. Phil., Mrs M.Jothimani B.E., M.Sc., M.Phil and Mrs. S.Andrews Felicita M. Sc., M.B.A., for their kind co-operation and support in successful completion of our project.

We would like to thank our lab assistant Mrs.Cladina for her kind and patient assistance in lab during our project hours.

We thank our family members especially our parents for their support both morally and financially which motivated us to finish the project successfully.

Last but no least, I owe our friends for their encouragement, support and creative ideas which led our project in a successful path.

S.NO	CONTENT	PAGENO
1.	INTRODUCTION	7
2.	SYSTEM ANALYSIS	15
	2.1 Existing System	16
	2.2 Proposed System	16
3.	SYSTEM REQUIREMENTS AND SPECIFICATIONS	17
	3.1 General Description	18
	3.2 System Objectives	18
	3.3 System Requirements	18
4.	SYSTEM DESIGN	21
	4.1 Input and Output Design	22
	4.2 Database Design	23
	4.3 Table Design	26
	4.4 Dataflow Diagram	29
	4.5 Form Design	32
	4.6 Source Code	37
5.	FUTURE ENHANCEMENT	60
6.	CONCLUSION	62
7.	BIBLIOGRAPHY	64

INTRODUCTION

1.INTRODUCTION

1.1 ABOUT THE PROJECT:

This project is a web based shopping system for an existing shop.

Online shopping is the process whereby consumers directly buy goods or services from a seller in real-time, without an intermediary service, over the Internet. It is a form of electronic commerce. This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps buying the products in the shop anywhere through internet by using an android device. Thus the customer will get the service of online shopping and home delivery from his favorite shop

PROJECT OBJECTIVE:

The objective of the project is to make an application in android platform to purchase items in an existing shop. In order to build such an application complete web support need to be provided. A complete and efficient web application which can provide the online shopping experience is the basic objective of the project. The web application can be implemented in the form of an android application with web view.

PROJECT OVERVIEW:

The central concept of the application is to allow the customer to shop virtually using the Internet and allow customers to buy the items and articles of their desire from the store. The information pertaining to the products are stores on an RDBMS at the server side (store).

The Server process the customers and the items are shipped to the address submitted by them. The application was designed into two modules first is for the customers who wish to buy the articles. Second is for the storekeepers who maintains and updates the information pertaining to the articles and those of the customers. The end user of this product is a departmental store where the application is hosted on the web and the administrator maintains the database. The application which is deployed at the customer database, the details of the items are brought forward from the database for the customer view based on the selection through the menu and the database of all the products are updated at the end of each transaction. Data entry into the

application can be done through various screens designed for various levels of users. Once the authorized personnel feed the relevant data into the system, several reports could be generated as per thesecurity.

PROJECT SCOPE:

This system can be implemented to any shop in the locality or to multinational branded shops having retail outlet chains. The system recommends a facility to accept the orders 24*7 and a home delivery system which can make customers happy.

If shops are providing an online portal where their customers can enjoy easy shopping from anywhere, the shops won't be losing any more customers to the trending online shops such as flipkart or epay. Since the application is available in the Smartphone it is easily accessible and always available.

STUDY OF THE SYSTEM

MODULES:

The system after careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

- Administrator
- Moderators
- Users

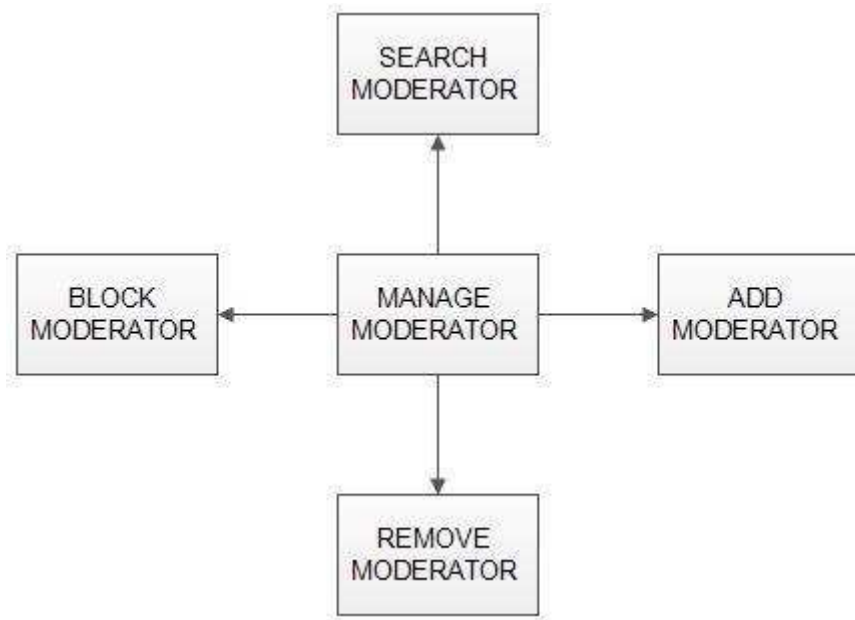
ADMINISTRATOR:

The administrator is the super user of this application. Only admin have access into this admin page. Admin may be the owner of the shop. The administrator has all the information about all the users and about all products.

This module is divided into different sub-modules.

1. Manage Moderators
2. Manage Products
3. Manage Users
4. Manage Orders

MANAGE MODERATORS:



Add Moderator:

Only admin is having the privilege to add a moderator. A moderator can be considered as a staff who manages the orders or owner of a group of products.

Block Moderator

Admin can restrict a moderator from managing the orders by blocking them. Admin can unblock a blocked user if needed.

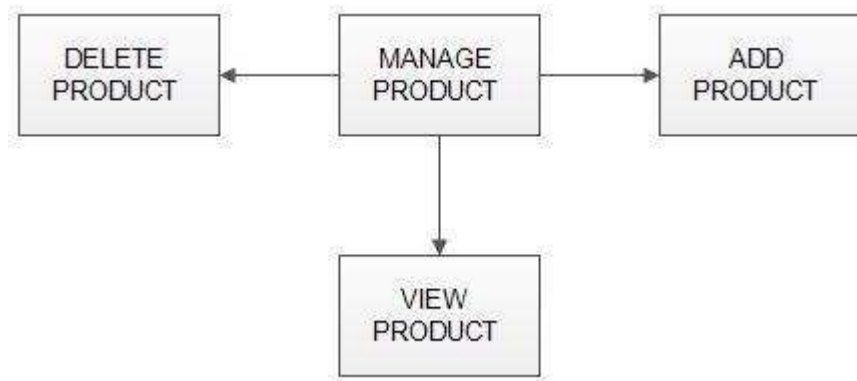
Remove Moderator

Admin has privilege to delete a moderator who was added.

Search Moderator:

All existing moderators can be viewed by the administrator as a list. If there is number of moderators and admin need to find one of them, the admin can search for a moderator by name.

MANAGE PRODUCTS:



Add Products

The shopping cart project contains different kind of products. The products can be classified into different categories by name. Admin can add new products into the existing system with all its details including an image.

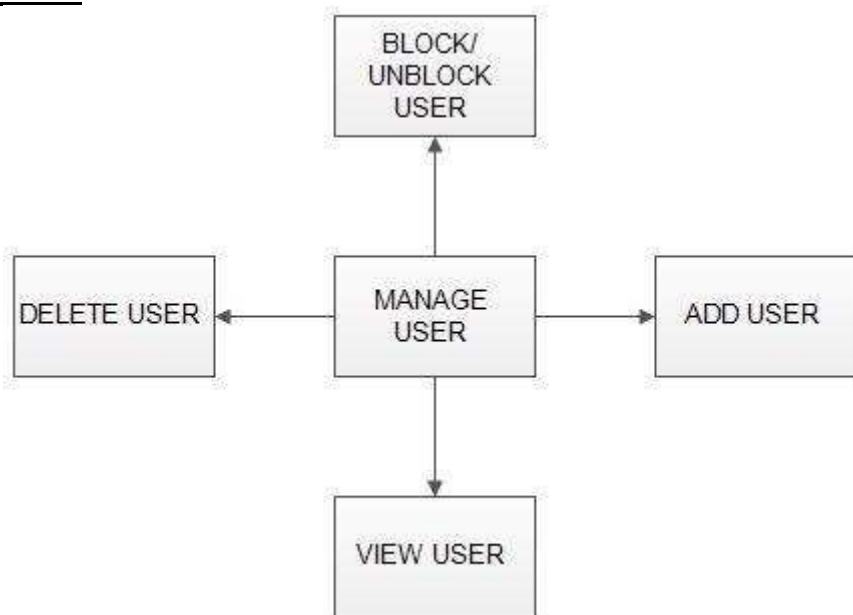
Delete Products

Administrator can delete the products based on the stock of that particular product.

Search Products

Admin will have a list view of all the existing products. He can also search for a particular product by name.

MANAGE USER:



View Users

The admin will have a list view of all the users registered in the system. Admin can view all the details of each user in the list except password.

Add Users

Admin has privileges to add a user directly by providing the details.

Delete & Block Users

Administrator has a right to delete or block a user. The default status of a new user registered is set as blocked. The admin must accept the new user by unblocking him.

MANAGE ORDERS:

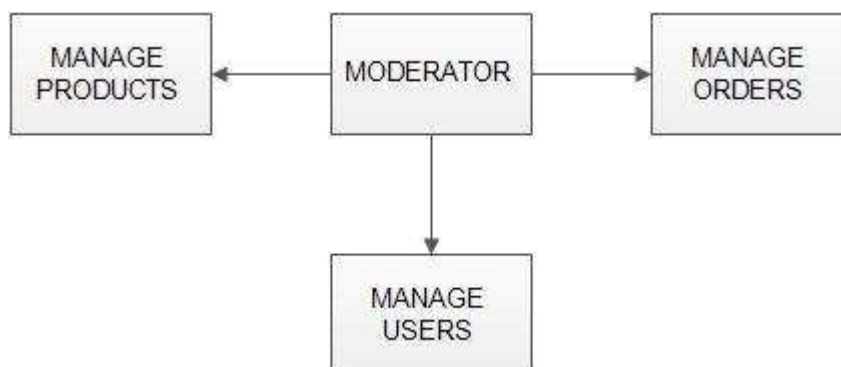
View Order

Administrator can view the Orders which is generated by the users. He can verify the details of the purchase.

Delete Order

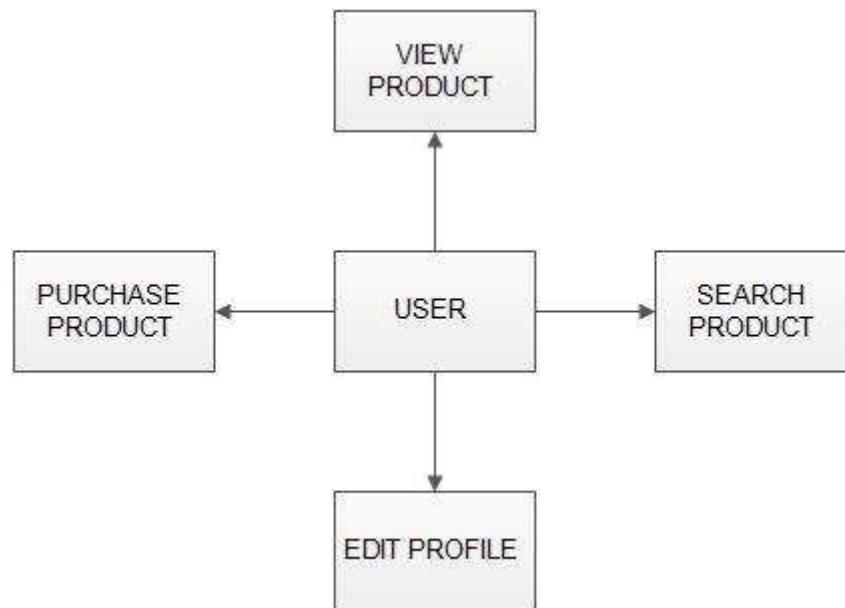
Admin can delete order from the orders list when the product is taken for delivery.

MODERATORS:



A moderator is considered as a staff who can manage orders for the time being. As a future update moderator may give facility to add and manage his own products. Moderators can reduce the work load of admin. Now moderator has all the privilege an admin having except managing other.

USERS:



Registration:

A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept a new user by unblockinghim.

Login

A user must login with his user name and password to the system after registration.

View Products

User can view the list of products based on their names after successful login. A detailed description of a particular product with product name, products details, product image, price can be viewed by users.

Search Product:

Users can search for a particular product in the list by name.

Add to Cart:

The user can add the desired product into his cart by clicking add to cart option on the product.

He can view his cart by clicking on the cart button. All products added by cart can be viewed in the cart. User can remove an item from the cart by clicking remove.

Submit Cart:

After confirming the items in the cart the user can submit the cart by providing a delivery address. On successful submitting the cart will become empty.

Edit Profile:

The user can view and edit the profile.

SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers.

System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified and the system is subjected to close study to identify the problem areas. The solutions are given as a proposal. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

2.1 EXISTING SYSTEM

The current system for shopping is to visit the shop manually and from the available product choose the item customer want and buying the item by payment of the price of the item .

- It is less user-friendly.
- User must go to shop and select products.
- It is difficult to identify the required product.
- Description of the product limited.
- It is a time consuming process.
- Not in reach of distant users.

2.2 PROPOSED SYSTEM:

In the proposed system customer need not go to the shop for buying the products. He can order the product he wish to buy through the application in his Smartphone. The shop owner will be admin of the system. Shop owner can appoint moderators who will help owner in managing the customers and product orders. The system also recommends a home delivery system for the purchased products.

SYSTEM REQUIREMENTS AND SPECIFICATIONS

3.SYSTEM REQUIREMENTS & SPECIFICATIONS

3.1 GENERAL DESCRIPTION

Product Description:

The system consists of two parts .A web application which can provide the online shopping service and an android application for the customer to access the web service from his Smartphone. Web application should be able to help the customer for selecting his item and to help the owner in managing the orders from the customers.

Problem Statement:

As online shopping became a trend nowadays the regular shops are losing their customers to online brands. Customers have effortless shopping experience and saving time through shopping online. For competing with those online brands , If shops are providing an online portal where their customers can shop through internet and get the products at their doors it will increase the number of customers.

3.2 SYSTEM OBJECTIVES:

- To provide an android application for online shopping of products in an existing shop.
- To provide a online shopping web site for the same shop.

3.3 SYSTEM REQUIREMENTS:

NON FUNCTIONAL REQUIREMENTS

i. EFFICIENCY REQUIREMENT

When an online shopping cart android application implemented customer can purchase product in an efficient manner.

ii. RELIABILITY REQUIREMENT

The system should provide a reliable environment to both customers and owner. All orders should be reaching at the admin without any errors.

iii. USABILITY REQUIREMENT

The android application is designed for user friendly environment and ease of use.

iv. IMPLEMENTATION REQUIREMENT

Implementation of the system using css and html in front end with jsp as back end and it will be used for database connectivity. And the database part is developed by my sql. Responsive web designing is used for making the website compatible for any type of screen.

v. DELIVERY REQUIREMENT

The whole system is expected to be delivered in four months of time with a weekly evaluation by the project guide.

FUNCTIONAL REQUIREMENTS:

USER

Description of feature

This feature used by the user to login into system. A user must login with his user name and password to the system after registration. If they are invalid, the user not allowed to enter the system.

Functional requirement

- Username and password will be provided after user registration is confirmed.
- Password should be hidden from others while typing it in the field

REGISTER NEW USER

Description of feature

A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept a new user by unblocking him.

Functional requirement

- System must be able to verify and validate information.
- The system must encrypt the password of the customer to provide security.

PURCHASING AN ITEM

Description of feature

The user can add the desired product into his cart by clicking add to cart option on the product. He can view his cart by clicking on the cart button. All products added by cart can be viewed in the cart. User can remove an item from the cart by clicking remove. After confirming the items in the cart the user can submit the cart by providing a delivery address. On successful submitting the cart will become empty.

Functional requirement

- System must ensure that, only a registered customer can purchase item.

SYSTEM DESIGN

4.SYSTEM DESIGN

System design is the solution for the creation of a new system. This phase focuses on the detailed implementation of the feasible system. It emphasis on translating design.Specifications to performance specification. System design has two phases of development

- Logical design
- Physical design

During logical design phase the analyst describes inputs (sources), output s(destinations), databases (data sores) and procedures (data flows) all in a format that meets the user requirements. The analyst also specifies the needs of the user at a level that virtually determines the information flow in and out of the system and the data resources. Here the logical design is done through data flow diagrams and database design. The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications, which specify exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user, perform necessary processing on accepted data and produce the required report on a hard copy or display it on the screen.

4.1 INPUT AND OUTPUT DESIGN

INPUT DESIGN:

Input design is the link that ties the information system into the world of its users. The input design involves determining the inputs, validating the data, minimizing the data entry and provides a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design. The user-originated inputs are converted to a computer based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the student details are to be entered at the time of registration. A page is designed for this purpose

which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

OUTPUT DESIGN:

Computer output is the most important and direct source of information to the user. Output design is a very important phase since the output needs to be in an efficient manner. Efficient and intelligible output design improves the system relationship with the user and helps in decision making. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. The output module of this system is the selected notifications.

4.2 DATABASE DESIGN

Databases are the storehouses of data used in the software systems. The data is stored in tables inside the database. Several tables are created for the manipulation of the data for the system. Two essential settings for a database are

- ☐ **Primary Key** - the field that is unique for all the record occurrences.
- ☐ **Foreign Key** - the field used to set relation between tables.

Normalization is a technique to avoid redundancy in the tables.

SYSTEM TOOLS

The various system tools that have been used in developing both the front end and the back end of the project are being discussed in this c.

FRONT END

PHP, HTML, CSS, JAVA SCRIPT are utilized to implement the front end.

PHP:

PHP is a general-purpose scripting language especially suited to web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994. Php is an acronym for “**PHP: Hypertext Preprocessor**”.

COMMON USES OF PHP:

- It is used for create dynamic website.
- To Interacting with web server (Apache etc)
- To interacting with any back-end / database server e.g. MySQL.
- To interaction with the native file system of the OS.
- To implement the business logical layers (one or more)
- It can Encrypt Data.
- Access Cookies variable and set cookies.

CHARACTERISTICS OF PHP:

- Familiarity.
- Simplicity.
- Efficiency.
- Security.
- Flexibility.

ADVANTAGES OF USING PHP:

- **Speed** up custom web application development
- Simplify web application **maintenance**
- No need to write additional code
- Work with databases more efficiently

HTML (Hyper Text Markup Language):

HTML is a syntax used to format a text document on the web.

CSS (Cascading Style Sheets):

CSS is a style sheet language used for describing the look and formatting of a document written in a markup language.

JAVA SCRIPT:

JS is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.

Java Script is used to create pop up windows displaying different alerts in the system like “User registered successfully ”,” Product added to cart” etc.

BACK END

The back end is implemented using MySQL which is used to design the databases.

MYSQL:

MySQL is the world's second most widely used open-source relational database system (RDBMS). The SQL phrase stands for Structured Query Language. An application software called Navicat was used to design the tables in MySQL.

FEATURES OF MYSQL:

- **Relational Database Management System (RDBMS)**

MySQL is a relational database management system.

- **Easy to use**

MySQL is easy to use. We have to get only the basic knowledge of SQL. We can build and interact with MySQL by using only a few simple SQL statements.

- **It is secure**

MySQL consists of a solid data security layer that protects sensitive data from intruders. Also, passwords are encrypted in MySQL.

- **Client/ Server Architecture**

MySQL follows the working of a client/server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server.

- **Free to download**

MySQL is free to use so that we can download it from MySQL official website without any cost.

- **It is scalable**

MySQL supports multi-threading that makes it easily scalable. It can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, we can increase this number to a theoretical limit of 8 TB of data.

- **Speed**

MySQL is considered one of the very fast database languages, backed by a large number of the benchmark test.

- **High Flexibility**

MySQL supports a large number of embedded applications, which makes MySQL very flexible.

4.3 TABLE DESIGN

PRODUCT DETAILS:

product_details					
Field Name	Field Type	Default	AllowNull	PriKey	Extra
id	int(11) FIRST		NO	YES	auto_increment
name	varchar(100) AFTER `id`		NO	NO	
description	varchar(500) AFTER `name`		YES	NO	
price	varchar(50) AFTER `description`		YES	NO	
photo	varchar(100) AFTER `price`		YES	NO	

PRODUCT ORDERS:

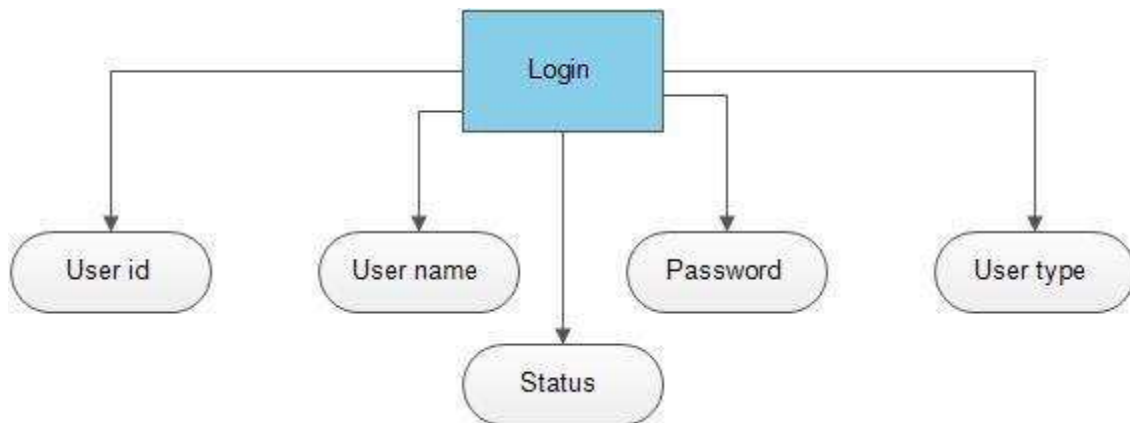
product_orders					
Field Name	Field Type	Default	AllowNull	PriKey	Extra
id	int(11) FIRST		NO	YES	auto_increment
product_id	int(11) AFTER `id`		NO	NO	
user_id	int(11) AFTER `product_id`		NO	NO	
deliver_address	varchar(500) AFTER `user_id`		YES	NO	

LOGIN:

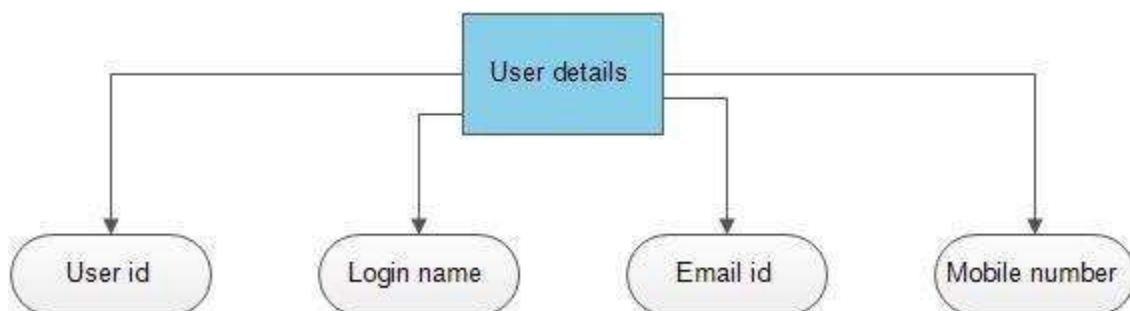
login					
Field Name	Field Type	Default	AllowNull	PriKey	Extra
uid	int(11) FIRST		NO	YES	auto_increment
uname	varchar(50) AFTER `uid`		NO	NO	
upass	varchar(50) AFTER `uname`		NO	NO	
utype	varchar(20) AFTER `upass`		NO	NO	
enabled	varchar(10) AFTER `utype`		NO	NO	

E-R DIAGRAMS

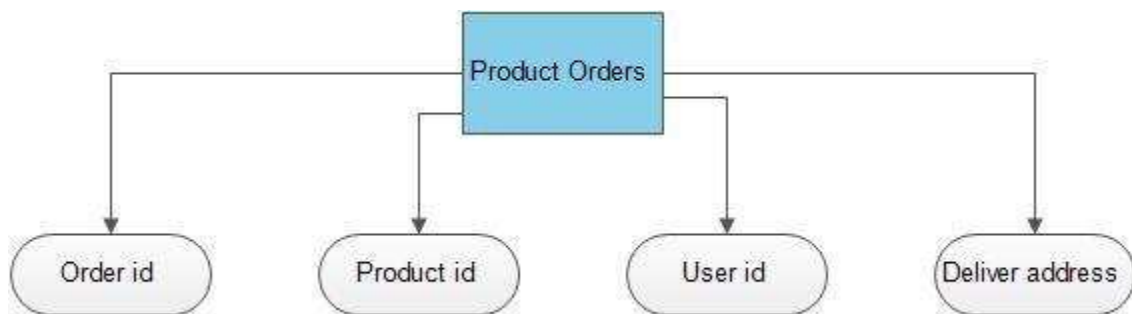
➤ LOGIN



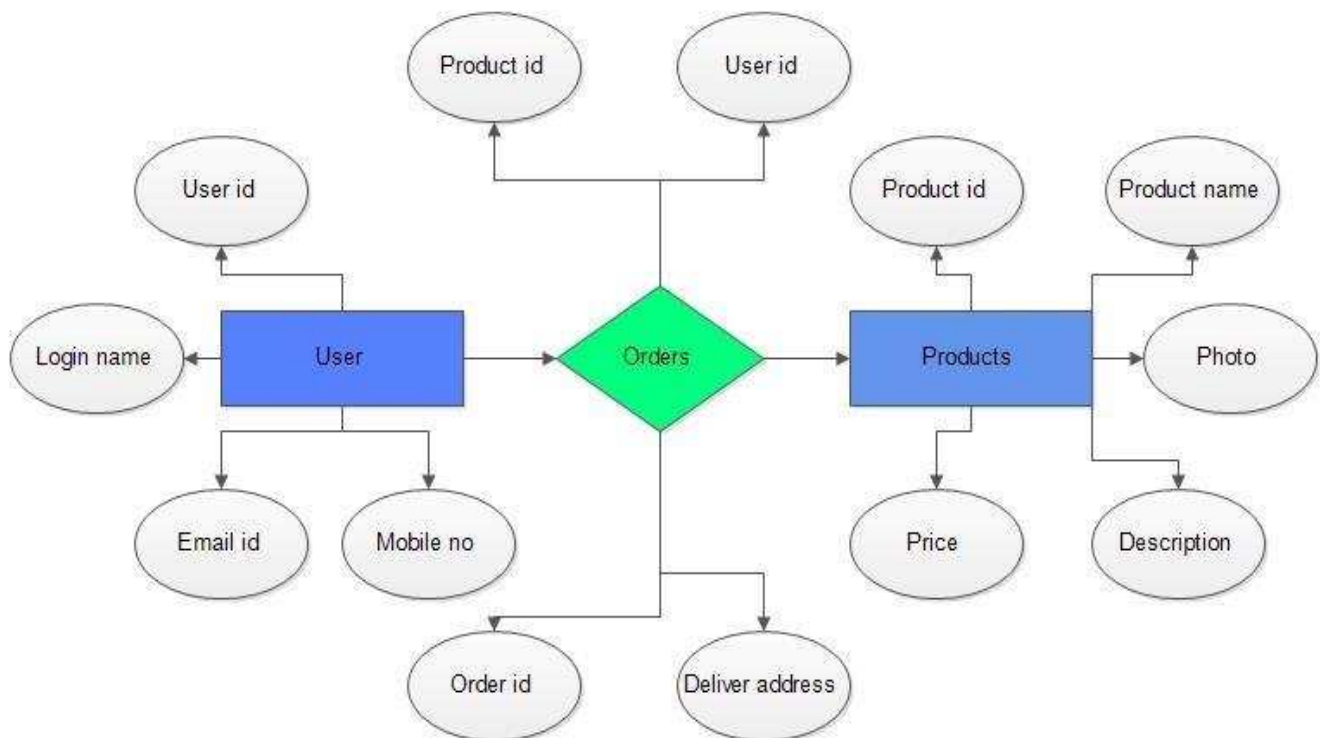
➤ USERDETAILS



➤ PRODUCT ORDERS



➤ COMPLETE DIAGRAM



4.4 DATA FLOW DIAGRAM:

A Data Flow Diagram (DFD) is a structured analysis and design tool that can be used for flowcharting. A DFD is a network that describes the flow of data and the processes that change or transform the data throughout a system. This network is constructed by using a set of symbols that do not imply any physical implementation. It has the purpose of clarifying system requirements and identifying major transformations. So it is the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. DFD can be considered to an abstraction of the logic of an information-oriented or a process-oriented system flow-chart. For these reasons DFD's are often referred to as logical data flow diagrams.

EXTERNAL ENTITY

An external entity is a source or destination of a data flow. Only those entities which originate or receive data are represented on a data flow diagram. The symbol used is a rectangular box.

PROCESS

A process shows a transformation or manipulation of data flow within the system. The symbol used is an oval shape.

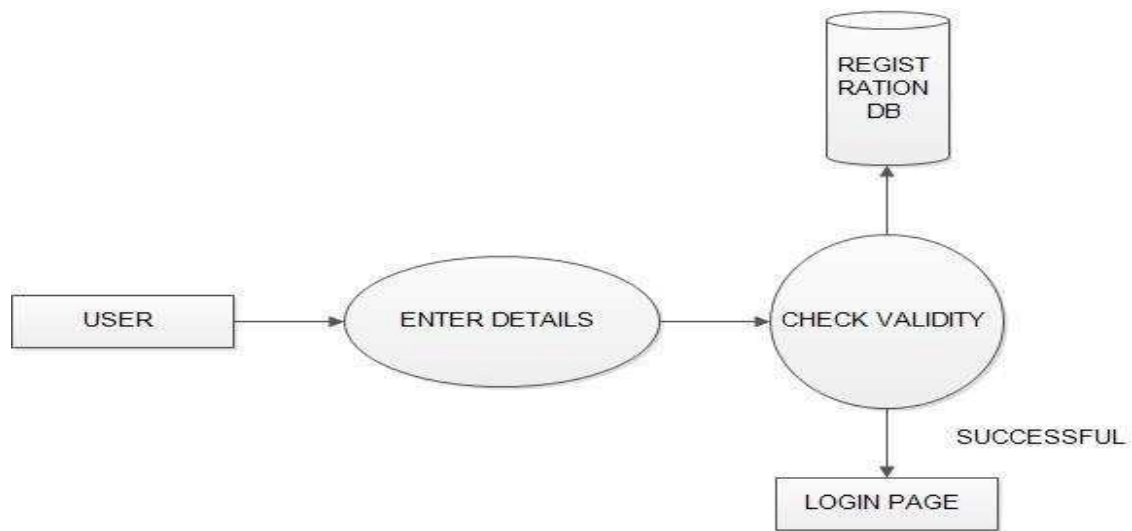
DATAFLOW

The data flow shows the flow of information from a source to its destination. Data flow is represented by a line, with arrow heads showing the direction of flow. Information always flows to or from a process and may be written, verbal or electronic. Each data flow may be referenced by the processes or data stores at its head and tail, or by a description of its contents

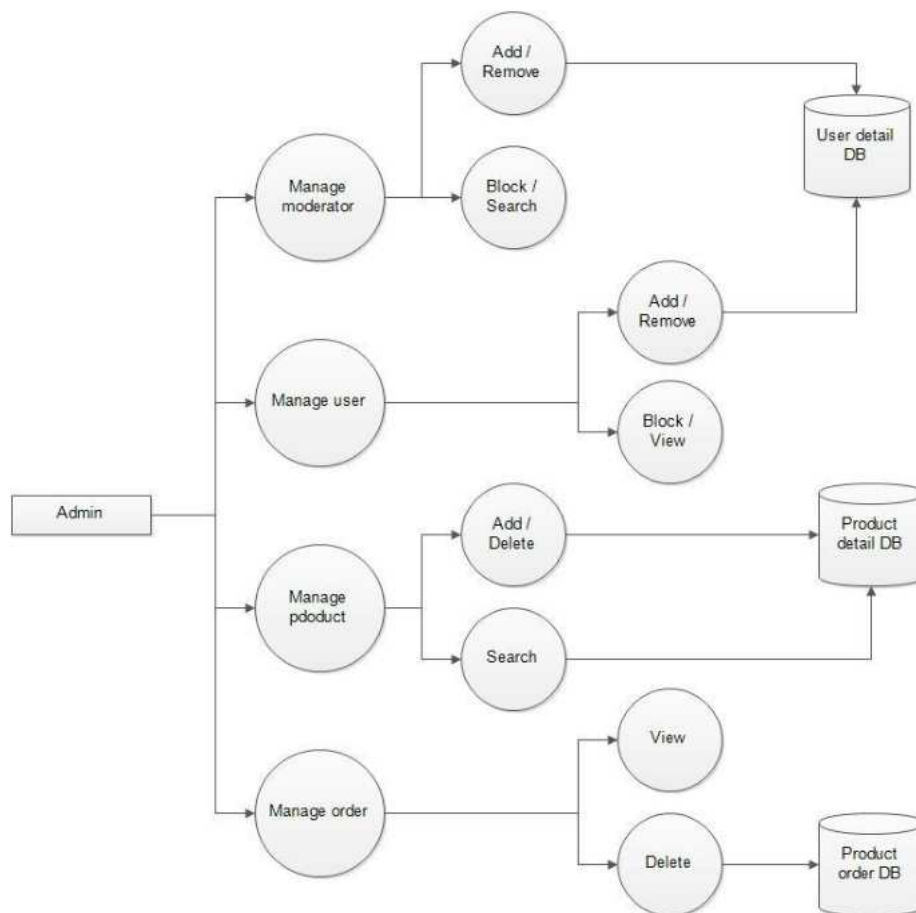
DATA STORE

A data store is a holding place for information within the system: It is represented by an open ended narrow rectangle. Data stores may be long-term files such as sales ledgers, or may be short-term accumulations: for example batches of documents that are waiting to be processed. Each data store should be given a reference followed by an arbitrary number.

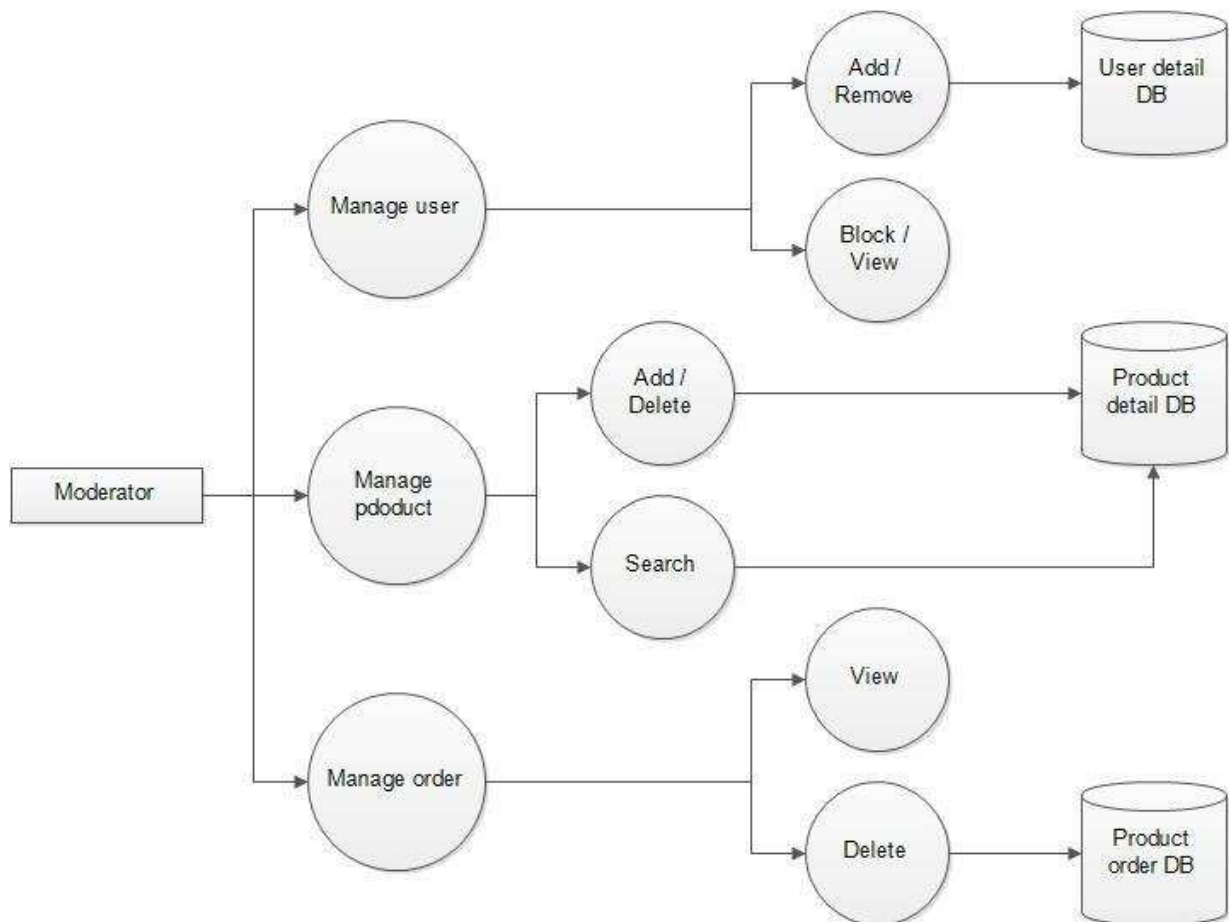
LOGIN DFD



ADMIN DFD

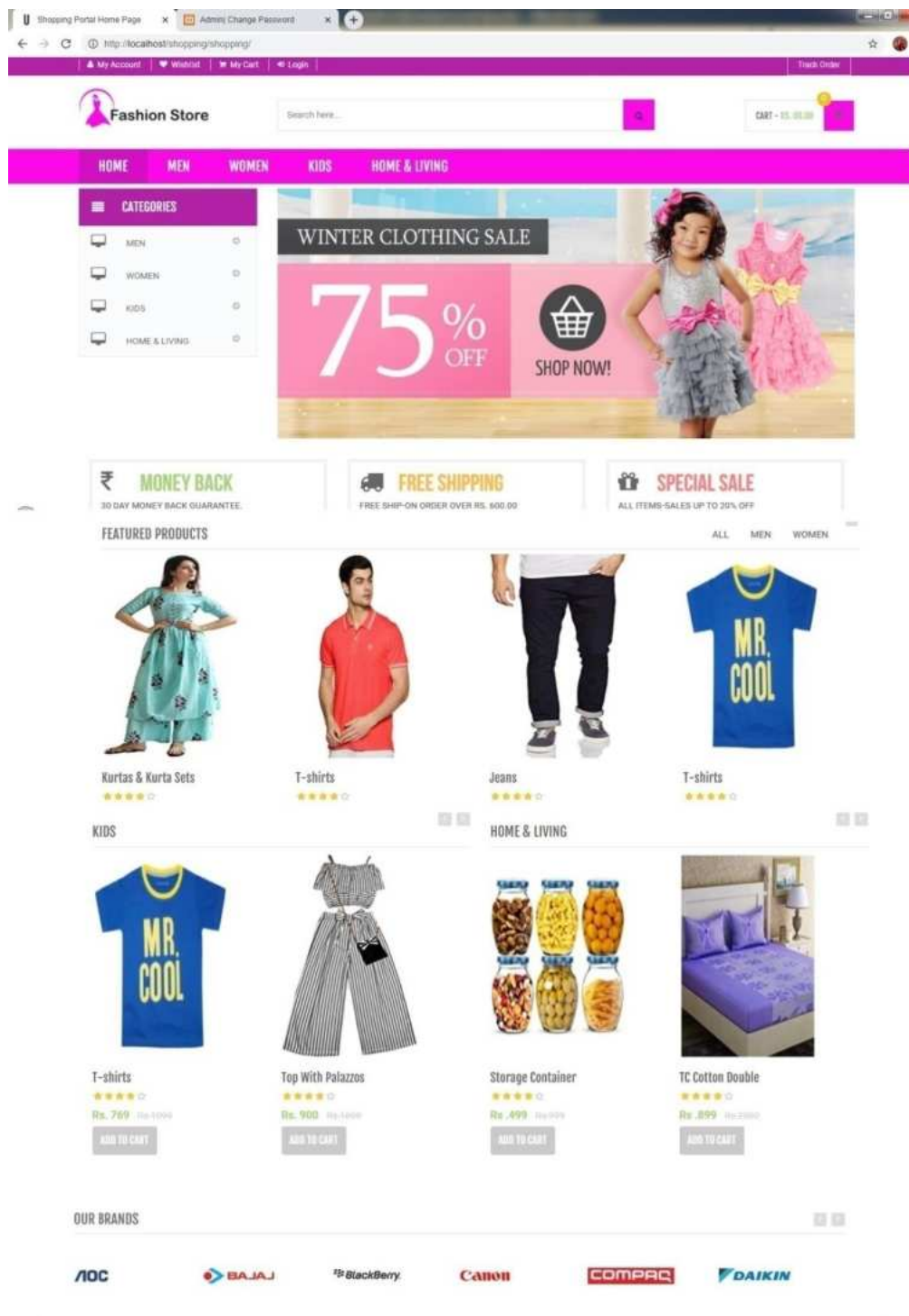


MODERATOR DFD

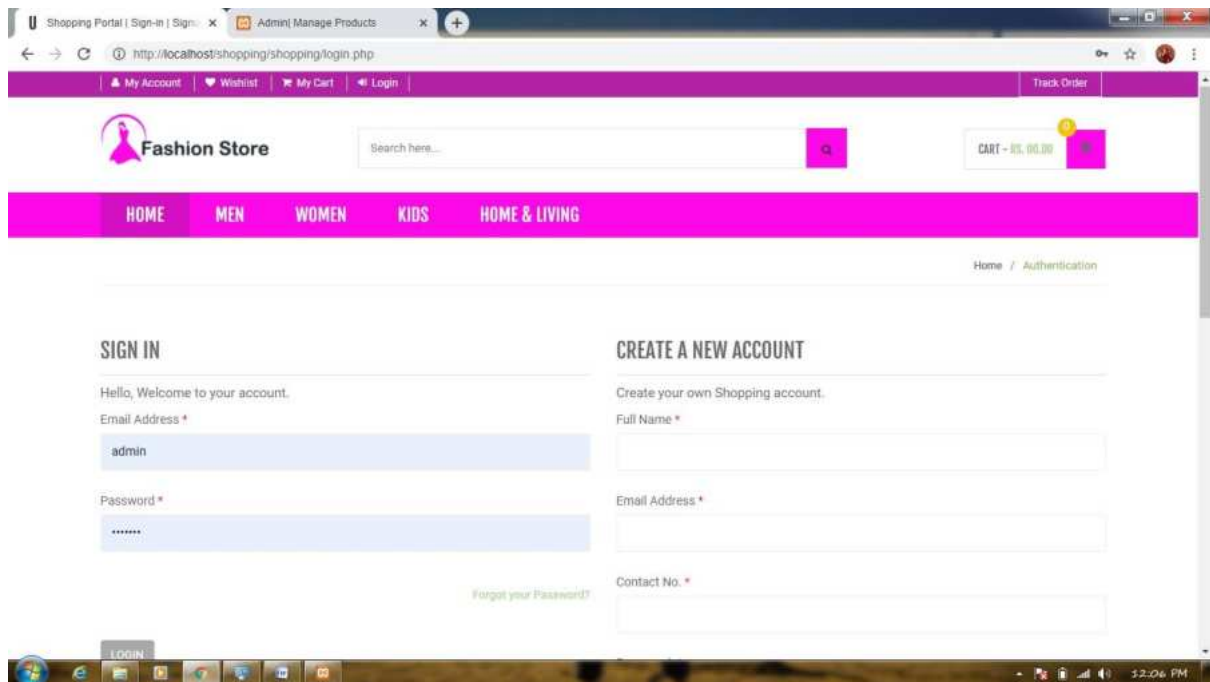


4.5 FORM DESIGN

HOME PAGE



LOGIN



The screenshot shows the login page of a web application titled "Fashion Store". The browser's address bar displays "http://localhost/shopping/shopping/login.php". The page features a purple header with navigation links: "My Account", "Wishlist", "My Cart", "Login", and "Track Order". Below the header is a search bar and a cart icon showing "CART - \$11, 00.00". The main navigation bar includes "HOME", "MEN", "WOMEN", "KIDS", and "HOME & LIVING". The page content is divided into two sections: "SIGN IN" and "CREATE A NEW ACCOUNT".

SIGN IN

Hello, Welcome to your account.

Email Address *

admin

Password *

[Forgot your Password?](#)

CREATE A NEW ACCOUNT

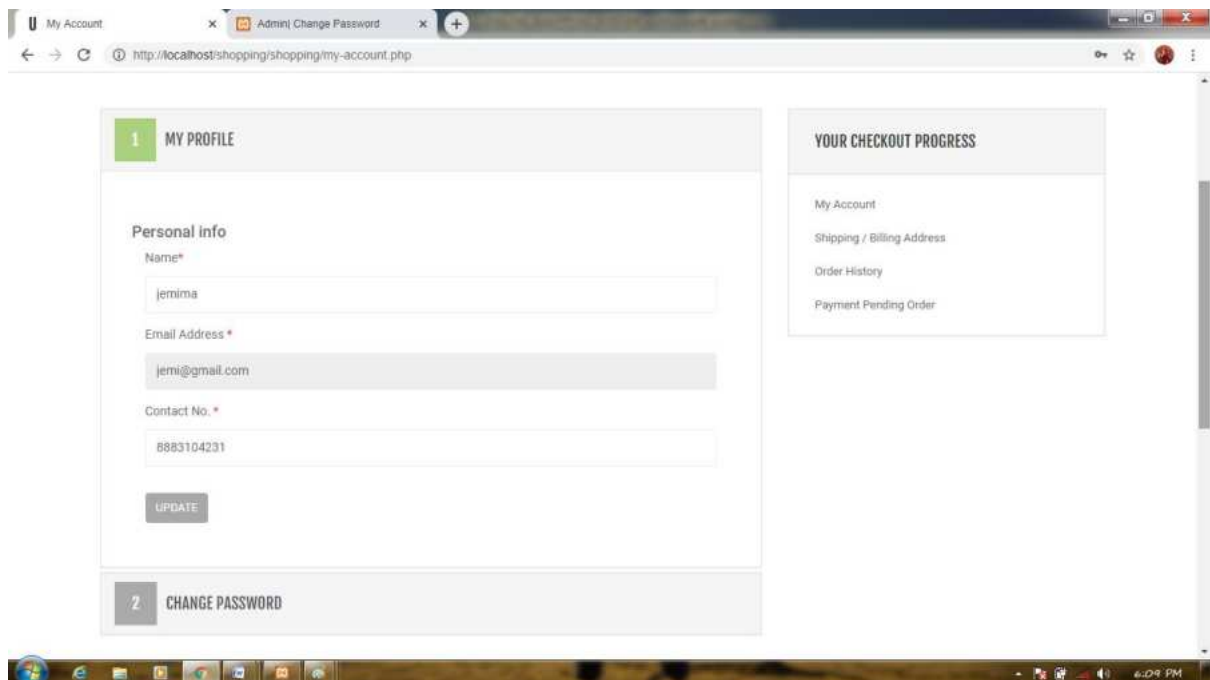
Create your own Shopping account.

Full Name *

Email Address *

Contact No. *

[LOGIN](#)



The screenshot shows the "My Account" page of the web application. The browser's address bar displays "http://localhost/shopping/shopping/my-account.php". The page is divided into two main sections: "MY PROFILE" and "CHANGE PASSWORD".

1 MY PROFILE

Personal info

Name*

jemima

Email Address *

jemi@gmail.com

Contact No. *

8883104231

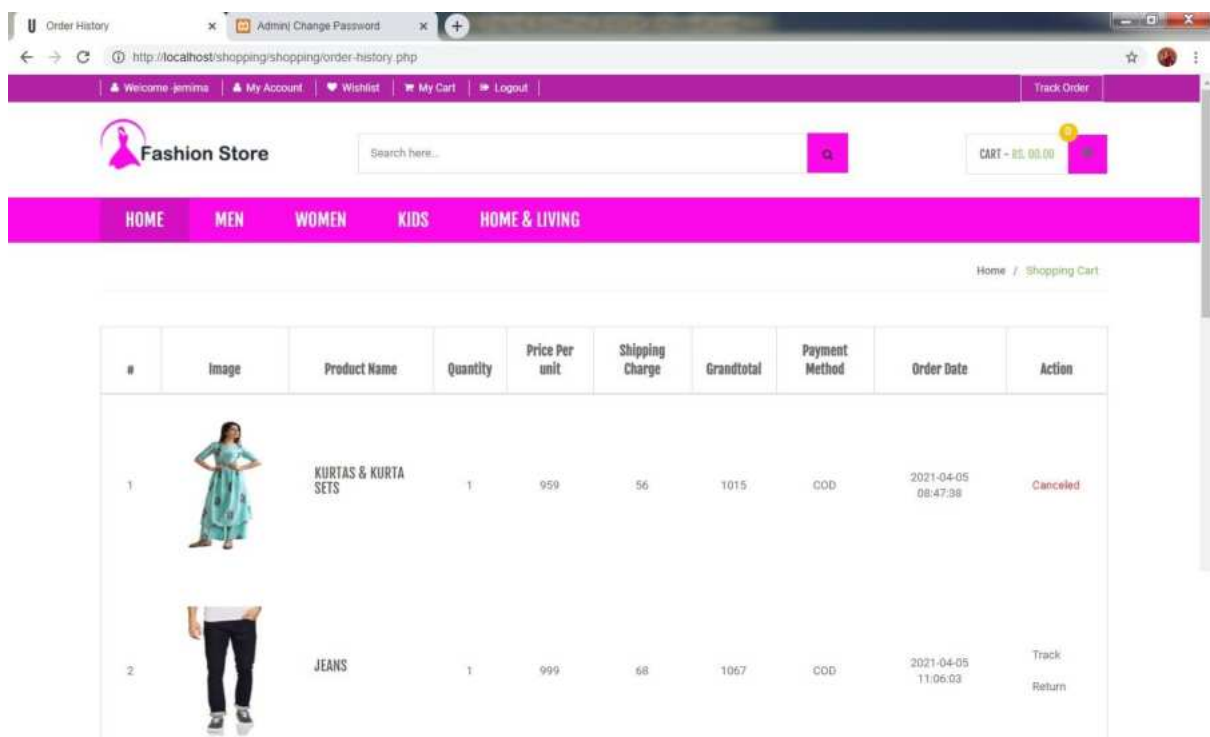
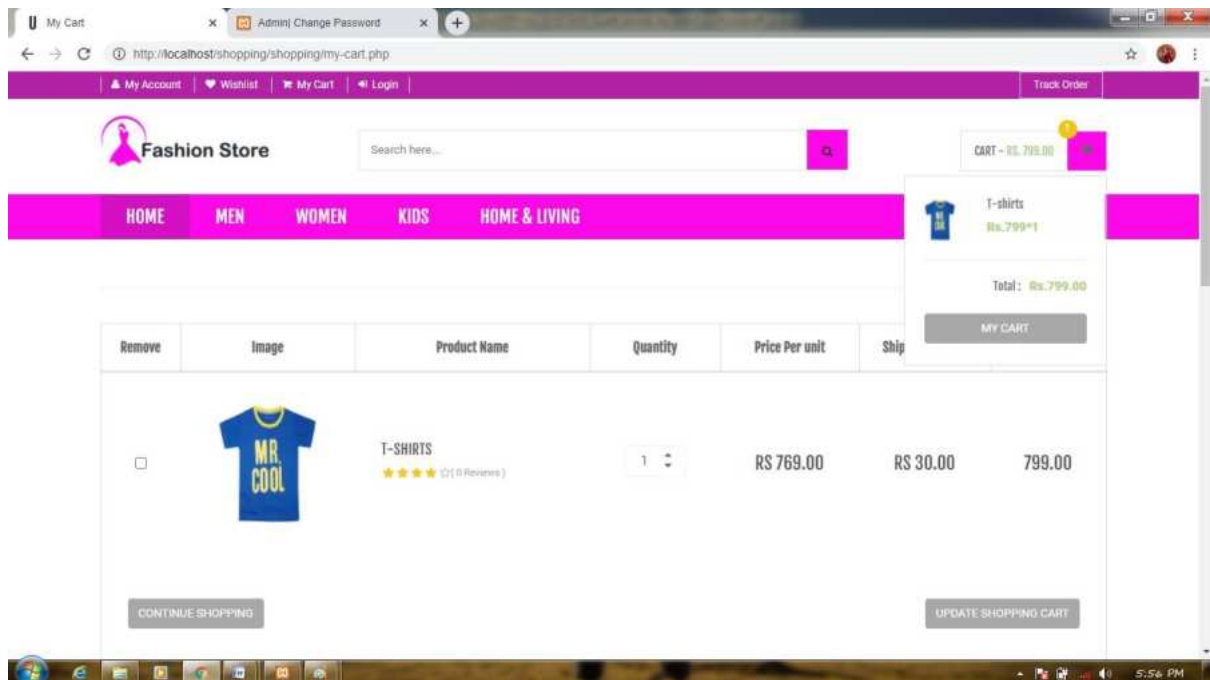
[UPDATE](#)

2 CHANGE PASSWORD

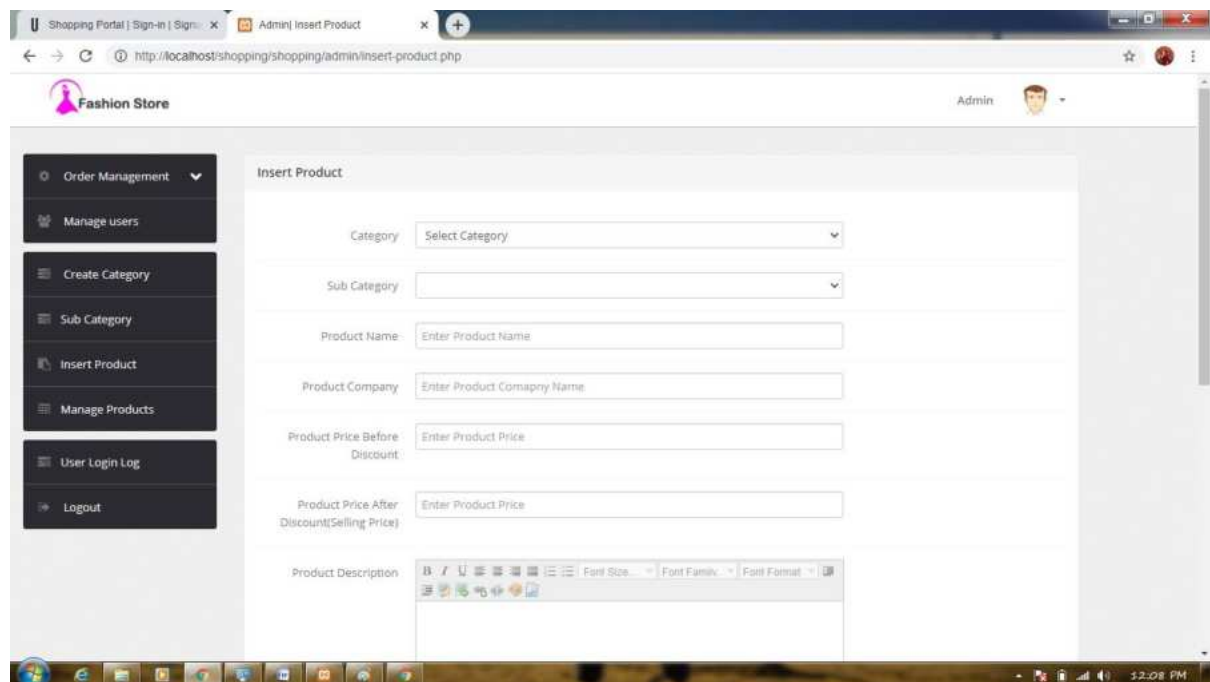
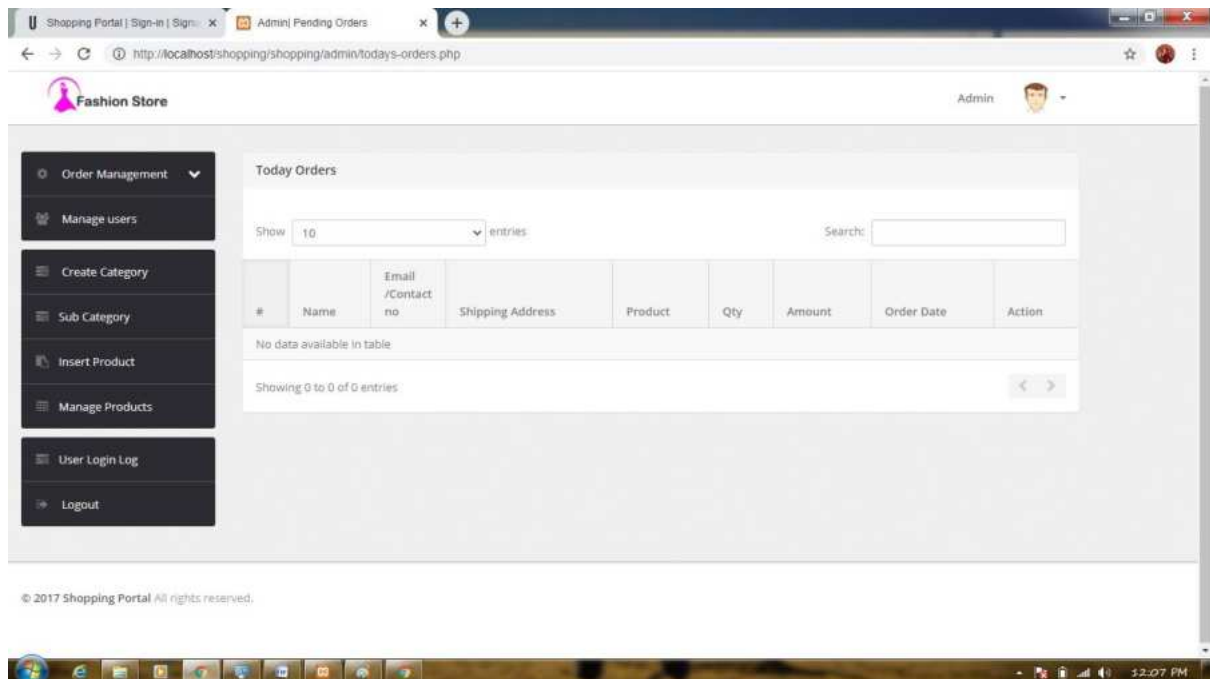
YOUR CHECKOUT PROGRESS

- My Account
- Shipping / Billing Address
- Order History
- Payment Pending Order

USER



ADMIN



Shopping Portal | Sign-in | Sign-out x Admin| Manage Products x

http://localhost/shopping/shopping/admin/manage-products.php

Fashion Store Admin

Order Management v

Manage users

Create Category

Sub Category

Insert Product

Manage Products

User Login Log

Logout

Manage Products

Show 10 entries Search:

#	Product Name	Category	Subcategory	Company Name	Product Creation Date	Action
1	T-shirts	Men	Topwear	Fashion Store	2021-02-24 16:01:52	Edit Delete
2	Jeans	Men	Bottomwear	Fashion Store	2021-02-24 16:10:14	Edit Delete
3	Kurtas & Kurta Sets	Women	Western Wear	Fashion Store	2021-02-24 15:55:18	Edit Delete
4	A-line Kurta	Women	Indian & fusion wear	Fashion Store	2021-02-24 17:01:22	Edit Delete
5	Womens Rayon Floral Print Western Top (Navy Blue)	Women	Western Wear	STYLELIME	2021-04-05 12:49:08	Edit Delete

My Account x Admin| Pending Orders x Update Complaint x

http://localhost/shopping/shopping/admin/updateorder.php?oid=9

Update Order !

order Id: 9

At Date: 2021-04-06 18:14:21

Status: Delivered

Remark: Your product is Delivered successfully..

Product Delivered

4.6 SOURCE CODE

LOGIN PAGE

```
<?php
session_start();
error_reporting(0);
include('includes/config.php');
// Code user Registration
if(isset($_POST['submit']))
{
$name=$_POST['fullname'];
$email=$_POST['emailid'];
$contactno=$_POST['contactno'];
$password=md5($_POST['password']);
$query=mysqli_query($con,"insert into users(name,email,contactno,password)
values('$name','$email','$contactno','$password')");
if($query)
{
    echo "<script>alert('You are successfully register');</script>";
}
else{
echo "<script>alert('Not register something went worng');</script>";
}
}
// Code for User login
if(isset($_POST['login']))
{
    $email=$_POST['email'];
    $password=md5($_POST['password']);
    $query=mysqli_query($con,"SELECT * FROM users WHERE email='$email'
and password='$password'");
    $num=mysqli_fetch_array($query);
    if($num>0)
    {
        $extra="my-cart.php";
        $_SESSION['login']=$_POST['email'];
        $_SESSION['id']=$num['id'];
        $_SESSION['username']=$num['name'];
        $uip=$_SERVER['REMOTE_ADDR'];
        $status=1;
        $log=mysqli_query($con,"insert into userlog(userEmail,userip,status)
values('".$_SESSION['login']."','$uip','$status')");
```

```

$host=$_SERVER['HTTP_HOST'];
$uri=rtrim(dirname($_SERVER['PHP_SELF']),'/\');
header("location:http://$host$uri/$extra");
exit();
}
else
{
$extra="login.php";
$email=$_POST['email'];
$uip=$_SERVER['REMOTE_ADDR'];
$status=0;
$log=mysqli_query($con,"insert into userlog(userEmail,userip,status)
values('$email','$uip','$status')");
$host = $_SERVER['HTTP_HOST'];
$uri = rtrim(dirname($_SERVER['PHP_SELF']),'/\');
header("location:http://$host$uri/$extra");
$_SESSION['errmsg']="Invalid email id or Password";
exit();
}
}

```

?>

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <!-- Meta -->
        <meta charset="utf-8">
        <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0, user-scalable=no">
        <meta name="description" content="">
        <meta name="author" content="">
        <meta name="keywords" content="MediaCenter, Template,
eCommerce">
        <meta name="robots" content="all">

        <title>Shopping Portal | Sign-in | Signup</title>

        <!-- Bootstrap Core CSS -->

```

```

<link rel="stylesheet" href="assets/css/bootstrap.min.css">

<!-- Customizable CSS -->
<link rel="stylesheet" href="assets/css/main.css">
<link rel="stylesheet" href="assets/css/green.css">
<link rel="stylesheet" href="assets/css/owl.carousel.css">
    <link rel="stylesheet" href="assets/css/owl.transitions.css">
    <!--<link rel="stylesheet" href="assets/css/owl.theme.css">-->
    <link href="assets/css/lightbox.css" rel="stylesheet">
    <link rel="stylesheet" href="assets/css/animate.min.css">
    <link rel="stylesheet" href="assets/css/rateit.css">
    <link rel="stylesheet" href="assets/css/bootstrap-select.min.css">

    <!-- Demo Purpose Only. Should be removed in production -->
    <link rel="stylesheet" href="assets/css/config.css">

    <link href="assets/css/green.css" rel="alternate stylesheet"
title="Green color">
    <link href="assets/css/blue.css" rel="alternate stylesheet"
title="Blue color">
    <link href="assets/css/red.css" rel="alternate stylesheet" title="Red
color">
    <link href="assets/css/orange.css" rel="alternate stylesheet"
title="Orange color">
    <link href="assets/css/dark-green.css" rel="alternate stylesheet"
title="Darkgreen color">
    <!-- Demo Purpose Only. Should be removed in production : END
-->

<!-- Icons/Glyphs -->
<link rel="stylesheet" href="assets/css/font-awesome.min.css">

<!-- Fonts -->
<link
href='http://fonts.googleapis.com/css?family=Roboto:300,400,500,700'
rel='stylesheet' type='text/css'>

<!-- Favicon -->
<link rel="shortcut icon" href="assets/images/favicon.ico">
<script type="text/javascript">
function valid()
{

```

```

if(document.register.password.value!=
document.register.confirmpassword.value)
{
alert("Password and Confirm Password Field do not match !!");
document.register.confirmpassword.focus();
return false;
}
return true;
}
</script>
<script>
function userAvailability() {
$("#loaderIcon").show();
jQuery.ajax({
url: "check_availability.php",
data:'email='+$("#email").val(),
type: "POST",
success:function(data){
$("#user-availability-status1").html(data);
$("#loaderIcon").hide();
},
error:function (){}
});
}
</script>

```

```

</head>
<body class="cnt-home">

```

```

<!--
===== HEADER
===== -->
<header class="header-style-1">

```

```

<!-- =====
TOP MENU ===== --
>
<?php include('includes/top-header.php');?>
<!-- ===== TOP

```

```

MENU : END =====
-->
<?php include('includes/main-header.php');?>
    <!-- =====
NAVBAR ===== -->
<?php include('includes/menu-bar.php');?>
<!-- ===== NAVBAR
: END ===== -->

</header>

<!-- ===== HEADER
: END ===== -->
<div class="breadcrumb">
    <div class="container">
        <div class="breadcrumb-inner">
            <ul class="list-inline list-unstyled">
                <li><a href="home.html">Home</a></li>
                <li class='active'>Authentication</li>
            </ul>
        </div><!-- /.breadcrumb-inner -->
    </div><!-- /.container -->
</div><!-- /.breadcrumb -->

<div class="body-content outer-top-bd">
    <div class="container">
        <div class="sign-in-page inner-bottom-sm">
            <div class="row">
                <!-- Sign-in -->
<div class="col-md-6 col-sm-6 sign-in">
    <h4 class="">sign in</h4>
    <p class="">Hello, Welcome to your account.</p>
    <form class="register-form outer-top-xs" method="post">
    <span style="color:red;" >
<?php
echo htmlentities($_SESSION['errmsg']);
?>
<?php
echo htmlentities($_SESSION['errmsg']="");
?>
        </span>
        <div class="form-group">
            <label class="info-title" for="exampleInputEmail1">Email

```

```

Address <span>*</span></label>
    <input type="email" name="email" class="form-control unicas-
form-control text-input" id="exampleInputEmail1" >
    </div>
    <div class="form-group">
        <label class="info-title"
for="exampleInputPassword1">Password <span>*</span></label>
        <input type="password" name="password" class="form-control
unicas-form-control text-input" id="exampleInputPassword1" >
        </div>
        <div class="radio outer-xs">
            <a href="forgot-password.php" class="forgot-password pull-
right">Forgot your Password?</a>
        </div>
        <button type="submit" class="btn-upper btn btn-primary checkout-
page-button" name="login">Login</button>
    </form>
</div>
<!-- Sign-in -->

<!-- create a new account -->
<div class="col-md-6 col-sm-6 create-new-account">
    <h4 class="checkout-subtitle">create a new account</h4>
    <p class="text title-tag-line">Create your own Shopping account.</p>
    <form class="register-form outer-top-xs" role="form" method="post"
name="register" onSubmit="return valid();">
    <div class="form-group">
        <label class="info-title" for="fullname">Full Name
    <span>*</span></label>
        <input type="text" class="form-control unicas-form-control text-
input" id="fullname" name="fullname" required="required">
    </div>

        <div class="form-group">
            <label class="info-title" for="exampleInputEmail2">Email
Address <span>*</span></label>
            <input type="email" class="form-control unicas-form-control
text-input" id="email" onBlur="userAvailability()" name="emailid" required >
            <span id="user-availability-status1" style="font-
size:12px;"></span>
        </div>

```

```

<div class="form-group">
    <label class="info-title" for="contactno">Contact No.
<span>*</span></label>
    <input type="text" class="form-control unicast-form-control text-
input" id="contactno" name="contactno" maxlength="10" required >
    </div>

<div class="form-group">
    <label class="info-title" for="password">Password.
<span>*</span></label>
    <input type="password" class="form-control unicast-form-control
text-input" id="password" name="password" required >
    </div>

<div class="form-group">
    <label class="info-title" for="confirmpassword">Confirm
Password. <span>*</span></label>
    <input type="password" class="form-control unicast-form-control
text-input" id="confirmpassword" name="confirmpassword" required >
    </div>

    <button type="submit" name="submit" class="btn-upper btn btn-
primary checkout-page-button" id="submit">Sign Up</button>
</form>
    <span class="checkout-subtitle outer-top-xs">Sign Up Today And You'll
Be Able To : </span>
    <div class="checkbox">
        <label class="checkbox">
            Speed your way through the checkout.
        </label>
        <label class="checkbox">
            Track your orders easily.
        </label>
        <label class="checkbox">
            Keep a record of all your purchases.
        </label>
    </div>
</div>
<!-- create a new account -->                                </div><!-- /.row -->
    </div>
<?php include('includes/brands-slider.php');?>
</div>

```

```

</div>
<?php include('includes/footer.php');?>
    <script src="assets/js/jquery-1.11.1.min.js"></script>

    <script src="assets/js/bootstrap.min.js"></script>

    <script src="assets/js/bootstrap-hover-dropdown.min.js"></script>
    <script src="assets/js/owl.carousel.min.js"></script>

    <script src="assets/js/echo.min.js"></script>
    <script src="assets/js/jquery.easing-1.3.min.js"></script>
    <script src="assets/js/bootstrap-slider.min.js"></script>
    <script src="assets/js/jquery.rateit.min.js"></script>
    <script type="text/javascript" src="assets/js/lightbox.min.js"></script>
    <script src="assets/js/bootstrap-select.min.js"></script>
    <script src="assets/js/wow.min.js"></script>
    <script src="assets/js/scripts.js"></script>

    <!-- For demo purposes – can be removed on production -->

    <script src="switchstylesheet/switchstylesheet.js"></script>

    <script>
        $(document).ready(function(){
            $(".changecolor").switchstylesheet( { seperator:"color" } );
            $('.show-theme-options').click(function(){
                $(this).parent().toggleClass('open');
                return false;
            });
        });

        $(window).bind("load", function() {
            $('.show-theme-options').delay(2000).trigger('click');
        });
    </script>
    <!-- For demo purposes – can be removed on production : End -->

</body>
</html>

```


ADMIN PAGE

```
<?php
session_start();
error_reporting(0);
include("include/config.php");
if(isset($_POST['submit']))
{
    $username=$_POST['username'];
    $password=md5($_POST['password']);
    $ret=mysqli_query($con,"SELECT * FROM admin WHERE
    username='$username' and password='$password'");
    $num=mysqli_fetch_array($ret);
    if($num>0)
    {
        $extra="change-password.php";//
        $_SESSION['alogin']=$_POST['username'];
        $_SESSION['id']=$num['id'];
        $host=$_SERVER['HTTP_HOST'];
        $uri=rtrim(dirname($_SERVER['PHP_SELF']),'/\');
        header("location:http://$host$uri/$extra");
        exit();
    }
    else
    {
        $_SESSION['errmsg']="Invalid username or password";
        $extra="index.php";
        $host = $_SERVER['HTTP_HOST'];
        $uri = rtrim(dirname($_SERVER['PHP_SELF']),'/\');
        header("location:http://$host$uri/$extra");
        exit();
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Shopping Portal | Admin login</title>
    <link type="text/css" href="bootstrap/css/bootstrap.min.css"
```

```

rel="stylesheet">
    <link type="text/css" href="bootstrap/css/bootstrap-responsive.min.css"
rel="stylesheet">
    <link type="text/css" href="css/theme.css" rel="stylesheet">
    <link type="text/css" href="images/icons/css/font-awesome.css"
rel="stylesheet">
    <link type="text/css"
href='http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,40
0,600' rel='stylesheet'>
</head>
<body>

    <div class="navbar navbar-fixed-top">
        <div class="navbar-inner">
            <div class="container">
                <a class="btn btn-navbar" data-toggle="collapse"
data-target=".navbar-inverse-collapse">
                    <i class="icon-reorder shaded"></i>
                </a>

                <a class="brand" href="index.html">
                    Shopping Portal | Admin
                </a>

                <div class="nav-collapse collapse navbar-inverse-
collapse">

                    <ul class="nav pull-right">

                        <li><a href="http://localhost/shopping/">
                            Back to Portal

                        </a></li>

                    </ul>
                </div><!-- /.nav-collapse -->
            </div>
        </div><!-- /navbar-inner -->
    </div><!-- /navbar -->

```

```

<div class="wrapper">
  <div class="container">
    <div class="row">
      <div class="module module-login span4 offset4">
        <form class="form-vertical" method="post">
          <div class="module-head">
            <h3>Sign In</h3>
          </div>
          <span style="color:red;" ><?php echo
htmlentities($_SESSION['errmsg']); ?><?php echo
htmlentities($_SESSION['errmsg']="");?></span>
          <div class="module-body">
            <div class="control-group">
              <div class="controls row-
fluid">
                <input class="span12"
type="text" id="inputEmail" name="username" placeholder="Username">
              </div>
            </div>
            <div class="control-group">
              <div class="controls row-
fluid">
                <input class="span12" type="password"
id="inputPassword" name="password" placeholder="Password">
              </div>
            </div>
            <div class="module-foot">
              <div class="control-group">
                <div class="controls
clearfix">
                  <button
type="submit" class="btn btn-primary pull-right"
name="submit">Login</button>
                </div>
              </div>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>

```

```

        </div>
    </div><!--/.wrapper-->

    <div class="footer">
        <div class="container">

            <b class="copyright">&copy; 2017 Shopping Portal </b>
All rights reserved.
        </div>
    </div>
    <script src="scripts/jquery-1.9.1.min.js" type="text/javascript"></script>
    <script src="scripts/jquery-ui-1.10.1.custom.min.js"
type="text/javascript"></script>
    <script src="bootstrap/js/bootstrap.min.js"
type="text/javascript"></script>
</body>

```

USER PAGE

```

<?php
session_start();
include('include/config.php');
if(strlen($_SESSION['alogin'])==0)
    {
header('location:index.php');
    }
else{

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Admin| Users log</title>
    <link type="text/css" href="bootstrap/css/bootstrap.min.css"
rel="stylesheet">
    <link type="text/css" href="bootstrap/css/bootstrap-responsive.min.css"
rel="stylesheet">
    <link type="text/css" href="css/theme.css" rel="stylesheet">

```

```

    <link type="text/css" href="images/icons/css/font-awesome.css"
rel="stylesheet">
    <link type="text/css"
href='http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,40
0,600' rel='stylesheet'>
</head>
<body>
<?php include('include/header.php');?>

```

```

    <div class="wrapper">
        <div class="container">
            <div class="row">
<?php include('include/sidebar.php');?>
                <div class="span9">
                    <div class="content">

```

```

    <div class="module">

```

```

        <div class="module-head">
            <h3>Manage Users</h3>
        </div>
        <div class="module-body table">

```

```

                                <table cellpadding="0"
cellspacing="0" border="0" class="datatable-1 table table-bordered table-
striped          display" width="100%">
=====

```

```

        <thead>
            <tr>

```

```

        <th>#</th>

```

```

        <th>

```

```

        User Email</th>

```

```

        <th>User

```

```

        IP </th>

```

```

        <th>Login Time</th>

```

```

        <th>Logout Time </th>

```

```

        <th>Status </th>

```

```

                                </tr>
                                </thead>

```

```

<tbody>

<?php $query=mysqli_query($con,"select * from userlog");
$cnt=1;
while($row=mysqli_fetch_array($query))
{
?>

<tr>

<td><?php echo htmlentities($cnt);?></td>

<td><?php echo htmlentities($row['userEmail']);?></td>

<td><?php echo htmlentities($row['userip']);?></td>
<td>
<?php echo htmlentities($row['loginTime']);?></td>

<td><?php echo htmlentities($row['logout']); ?></td>
<td><?php
$st=$row['status'];

if($st==1)
{
    echo "Successfull";
}
else
{
    echo "Failed";
}

?></td>

<?php
$cnt=$cnt+1; } ?>

</table>
</div>
</div>
<!--/.content-->

```

```

        </div><!--/.span9-->
    </div>
</div><!--/.container-->
</div><!--/.wrapper-->

<?php include('include/footer.php');?>

    <script src="scripts/jquery-1.9.1.min.js" type="text/javascript"></script>
    <script src="scripts/jquery-ui-1.10.1.custom.min.js"
type="text/javascript"></script>
    <script src="bootstrap/js/bootstrap.min.js"
type="text/javascript"></script>
    <script src="scripts/flot/jquery.flot.js" type="text/javascript"></script>
    <script src="scripts/datatables/jquery.dataTables.js"></script>
    <script>
        $(document).ready(function() {
            $('.datatable-1').dataTable();
            $('.dataTables_paginate').addClass("btn-group datatable-
pagination");
            $('.dataTables_paginate > a').wrapInner('<span />');
            $('.dataTables_paginate > a:first-child').append('<i
class="icon-chevron-left shaded"></i>');
            $('.dataTables_paginate > a:last-child').append('<i
class="icon-chevron-right shaded"></i>');
        } );
    </script>
</body>
<?php } ?>

//account

<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['login'])==0)
    {
header('location:login.php');
    }
else{
    if(isset($_POST['update']))
    {
        $name=$_POST['name'];

```

```

        $contactno=$_POST['contactno'];
        $query=mysqli_query($con,"update users set
name='$name',contactno='$contactno' where id='".$_SESSION['id']."'");
        if($query)
        {
echo "<script>alert('Your info has been updated');</script>";
        }
    }
}

```

```

date_default_timezone_set('Asia/Kolkata');// change according timezone
$currentTime = date( 'd-m-Y h:i:s A', time () );

```

```

if(isset($_POST['submit']))
{
$sql=mysqli_query($con,"SELECT password FROM users where
password='".$_md5($_POST['cpass'])."' && id='".$_SESSION['id']."'");
$num=mysqli_fetch_array($sql);
if($num>0)
{
    $con=mysqli_query($con,"update students set
password='".$_md5($_POST['newpass'])."', updationDate='$currentTime' where
id='".$_SESSION['id']."'");
echo "<script>alert('Password Changed Successfully !!');</script>";
}
else
{
    echo "<script>alert('Current Password not match !!');</script>";
}
}

```

```

?>
<!DOCTYPE html>
<html lang="en">
    <head>
        <!-- Meta -->
        <meta charset="utf-8">
        <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0, user-scalable=no">
        <meta name="description" content="">

```



```

        <meta name="author" content="">
        <meta name="keywords" content="MediaCenter, Template,
eCommerce">
        <meta name="robots" content="all">

<title>My Account</title>

<!-- Bootstrap Core CSS -->
<link rel="stylesheet" href="assets/css/bootstrap.min.css">

<!-- Customizable CSS -->
<link rel="stylesheet" href="assets/css/main.css">
<link rel="stylesheet" href="assets/css/green.css">
<link rel="stylesheet" href="assets/css/owl.carousel.css">
    <link rel="stylesheet" href="assets/css/owl.transitions.css">
    <!--<link rel="stylesheet" href="assets/css/owl.theme.css">-->
    <link href="assets/css/lightbox.css" rel="stylesheet">
    <link rel="stylesheet" href="assets/css/animate.min.css">
    <link rel="stylesheet" href="assets/css/rateit.css">
    <link rel="stylesheet" href="assets/css/bootstrap-select.min.css">

    <!-- Demo Purpose Only. Should be removed in production -->
    <link rel="stylesheet" href="assets/css/config.css">

    <link href="assets/css/green.css" rel="alternate stylesheet"
title="Green color">
    <link href="assets/css/blue.css" rel="alternate stylesheet"
title="Blue color">
    <link href="assets/css/red.css" rel="alternate stylesheet" title="Red
color">
    <link href="assets/css/orange.css" rel="alternate stylesheet"
title="Orange color">
    <link href="assets/css/dark-green.css" rel="alternate stylesheet"
title="Darkgreen color">
    <link rel="stylesheet" href="assets/css/font-awesome.min.css">
    <link
href='http://fonts.googleapis.com/css?family=Roboto:300,400,500,700'
rel='stylesheet' type='text/css'>
    <link rel="shortcut icon" href="assets/images/favicon.ico">
<script type="text/javascript">
function valid()
{
if(document.chngpwd.cpass.value=="")

```

```

{
alert("Current Password Filed is Empty !!");
document.chngpwd.cpass.focus();
return false;
}
else if(document.chngpwd.newpass.value=="")
{
alert("New Password Filed is Empty !!");
document.chngpwd.newpass.focus();
return false;
}
else if(document.chngpwd.cnfpass.value=="")
{
alert("Confirm Password Filed is Empty !!");
document.chngpwd.cnfpass.focus();
return false;
}
else if(document.chngpwd.newpass.value!= document.chngpwd.cnfpass.value)
{
alert("Password and Confirm Password Field do not match !!");
document.chngpwd.cnfpass.focus();
return false;
}
return true;
}
</script>

</head>
<body class="cnt-home">
<header class="header-style-1">

<!-- =====
TOP MENU ===== --
>
<?php include('includes/top-header.php');?>
<!-- ===== TOP
MENU : END =====
-->
<?php include('includes/main-header.php');?>
<!-- =====
NAVBAR ===== -->
<?php include('includes/menu-bar.php');?>
<!-- ===== NAVBAR

```

```

: END ===== -->

</header>
<!-- ===== HEADER
: END ===== -->
<div class="breadcrumb">
    <div class="container">
        <div class="breadcrumb-inner">
            <ul class="list-inline list-unstyled">
                <li><a href="#">Home</a></li>
                <li class='active'>Checkout</li>
            </ul>
        </div><!-- /.breadcrumb-inner -->
    </div><!-- /.container -->
</div><!-- /.breadcrumb -->

<div class="body-content outer-top-bd">
    <div class="container">
        <div class="checkout-box inner-bottom-sm">
            <div class="row">
                <div class="col-md-8">
                    <div class="panel-group checkout-steps"
id="accordion">
                        <!-- checkout-step-01 -->
                        <div class="panel panel-default checkout-step-01">

                            <!-- panel-heading -->
                            <div class="panel-heading">
                                <h4 class="unicase-checkout-title">
                                    <a data-toggle="collapse" class="" data-parent="#accordion"
href="#collapseOne">
                                        <span>1</span>My Profile
                                    </a>
                                </h4>
                            </div>
                            <!-- panel-heading -->

                                <div id="collapseOne" class="panel-collapse collapse in">

                                    <!-- panel-body -->
                                    <div class="panel-body">
                                        <div class="row">
                                            <h4>Personal info</h4>

```

```

        <div class="col-md-12 col-sm-12 already-registered-
login">

    <?php
    $query=mysqli_query($con,"select * from users where
    id='".$_SESSION['id']."'");
    while($row=mysqli_fetch_array($query))
    {
    ?>

        <form class="register-form" role="form"
method="post">
    <div class="form-group">
        <label class="info-title"
for="name">Name<span>*</span></label>
        <input type="text" class="form-control
unicase-form-control text-input" value="<?php echo $row['name'];?>"
id="name" name="name" required="required">
        </div>

        <div class="form-group">
            <label class="info-title"
for="exampleInputEmail1">Email Address <span>*</span></label>
            <input type="email" class="form-control unicase-form-
control text-input" id="exampleInputEmail1" value="<?php echo
$row['email'];?>" readonly>
            </div>
            <div class="form-group">
                <label class="info-title" for="Contact
No.">Contact No. <span>*</span></label>
                <input type="text" class="form-control
unicase-form-control text-input" id="contactno" name="contactno"
required="required" value="<?php echo $row['contactno'];?>"
maxlength="10">
                </div>
                <button type="submit" name="update"
class="btn-upper btn btn-primary checkout-page-button">Update</button>
            </form>
            <?php } ?>
        </div>
    <!-- already-registered-login -->

```

```

        </div>
    </div>
    <!-- panel-body -->

</div><!-- row -->
</div>
<!-- checkout-step-01 -->

        <!-- checkout-step-02 -->
        <div class="panel panel-default
checkout-step-02">

            <div class="panel-heading">
                <h4 class="unicase-checkout-title">
                    <a data-toggle="collapse"
class="collapsed" data-parent="#accordion" href="#collapseTwo">
                        <span>2</span>Change Password
                    </a>
                </h4>
            </div>
            <div id="collapseTwo" class="panel-
collapse collapse">

                <div class="panel-body">

                    <form class="register-form" role="form"
method="post" name="chngpwd" onSubmit="return valid();">
                    <div class="form-group">

                        <label class="info-title" for="Current
Password">Current Password<span>*</span></label>
                        <input type="password" class="form-control
unicase-form-control text-input" id="cpass" name="cpass"
required="required">

                    </div>

                    <div class="form-group">
                        <label class="info-title" for="New
Password">New Password <span>*</span></label>
                        <input type="password" class="form-control unicase-form-
control text-input" id="newpass" name="newpass">
                    </div>
                    <div class="form-group">
                        <label class="info-title" for="Confirm

```

```

Password">Confirm Password <span>*</span></label>
      <input type="password" class="form-control
unicase-form-control text-input" id="cnfpass" name="cnfpass"
required="required" >
      </div>
      <button type="submit" name="submit"
class="btn-upper btn btn-primary checkout-page-button">Change </button>
    </form>

```

```

      </div>
    </div>
  </div>
<!-- checkout-step-02 -->

```

```

    </div><!-- /.checkout-steps -->
  </div>
  <?php include('includes/myaccount-sidebar.php');?>
</div><!-- /.row -->
</div><!-- /.checkout-box -->
<?php include('includes/brands-slider.php');?>

```

```

</div>
</div>
<?php include('includes/footer.php');?>
  <script src="assets/js/jquery-1.11.1.min.js"></script>

  <script src="assets/js/bootstrap.min.js"></script>

  <script src="assets/js/bootstrap-hover-dropdown.min.js"></script>
  <script src="assets/js/owl.carousel.min.js"></script>

  <script src="assets/js/echo.min.js"></script>
  <script src="assets/js/jquery.easing-1.3.min.js"></script>
  <script src="assets/js/bootstrap-slider.min.js"></script>
  <script src="assets/js/jquery.rateit.min.js"></script>
  <script type="text/javascript" src="assets/js/lightbox.min.js"></script>
  <script src="assets/js/bootstrap-select.min.js"></script>
  <script src="assets/js/wow.min.js"></script>
  <script src="assets/js/scripts.js"></script>

```

```

<!-- For demo purposes – can be removed on production -->

<script src="switchstylesheet/switchstylesheet.js"></script>

<script>
    $(document).ready(function(){
        $(".change-color").switchstylesheet( { separator:"color" } );
        $('#show-theme-options').click(function(){
            $(this).parent().toggleClass('open');
            return false;
        });
    });

    $(window).bind("load", function() {
        $('#show-theme-options').delay(2000).trigger('click');
    });
</script>
</body>
</html>
<?php } ?>

```

FUTURE ENHANCEMENT

5.FUTURE ENHANCEMENT

Although the project has been completed there are several plans for enhancement of the project in future.

- Providing online placement record through our site.
- Providing personalized inbox to the user.
- Providing video conferencing with the Artists.

CONCLUSION

6.CONCLUSION

The project entitled **Online shopping system** was completed successfully.

The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming. The purpose of this project was to develop a web application and an android application for purchasing items from a shop.

This project helped us in gaining valuable information and practical knowledge on several topics like designing web pages using HTML & CSS, usage of responsive templates, and of database using MySQL . The entire system is secured. Also the project helped us understanding about the development phases of a project and software development life cycle. We learned how to test different features of a project.

This project has given us great satisfaction in having designed an application which can be implemented to any nearby shops or branded shops selling various kinds of products by simple modifications.

BIBLIOGRAPHY

7..BIBLIOGRAPHY

REFERENCES:

- JavaScript Enlightenment,Cody Lindley-First Edition, basedon JavaScript 1.5, ECMA-262,Edition
- McGrawHill's , Java : The complete reference 7thEdition,Herbert Schildit
- Complete CSS Guide ,MaxineSherrin and John Allsopp-O'Reilly Media; September2012

WEBSITES:

- <http://www.w3schools.com/html/default.asp>,
- <http://www.w3schools.com/css/default.asp>,
- <http://www.w3schools.com/js/default.asp>,

SPARKCART

Project Submitted to the Department Of Computer Science

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



In partial fulfilment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE done by

Aarthi Angeline .P	-	18AUCS01
Asikka .L	-	18AUCS04
Marcina .A	-	18AUCS22
Muthu Iswarya .V	-	18AUCS29
Roshini .R	-	18AUCS41
Santhiya .R	-	18AUCS42

Under the Supervision and Guidance of

Mrs. S. Andrews Felicita M.Sc., M.B.A.

DEPARTMENT OF COMPUTER SCIENCE

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

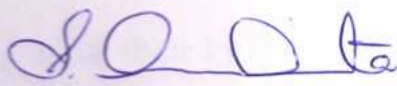
This is certify that the project entitled

SPARKCART

Submitted by

Aarthi Angeline .P	-	18AUCS01
Asikka .L	-	18AUCS04
Marcina .A	-	18AUCS22
Muthu Iswarya .V	-	18AUCS29
Roshini .R	-	18AUCS41
Santhiya .R	-	18AUCS42

In partial fulfilment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.


Guide


Head of the Department

Submitted for the examination held on 15.04.2021


Examiner


Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and blessings to finish our project successfully.

We are infinitely indebted to our college principal Dr. Sr. A. S. J. Lucia Rose M.SC, M.Phil., PGDCA, Ph.D., for providing us necessary infrastructure to undertake this project and work successfully.

We are so grateful to Mrs. S. Gnanathangam M.C.A, M.Phil (Associate professor & HOD), Department of Computer Science, for her gracious guidance. Our heartfelt thanks to our project guide Mrs.S. Andrews Felicita M.Sc.,M.B.A, (Assistant Professor), Department of Computer Science, for her valuable assistance and kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- Mrs.M. Jothimani B.E., M.Sc., M.Phil., (Associate professor)
- Sr.K.Reetha M.Sc.,M.Phil.(Assistant professor)
- Mrs.J. Anila Maily M.Sc , M.Phil. (Associate Professor),

for their kind cooperation in successful completion of the project.

Index

S.NO.	CONTENT	PAGE.NO
1	Introduction <ul style="list-style-type: none">○ Objective	6
2	System Analysis <ul style="list-style-type: none">○ System Analysis○ Existing System○ Proposed System	8
3	Project Description	12
4	System Specifications <ul style="list-style-type: none">● Hardware Requirement● Software Requirements● Network Requirements● Software Description	14

5	System Configuration <ul style="list-style-type: none">• Modules	19
6	Data Flow Diagram	22
7	TestingImplementation	27
8	Coding	30
9	Screenshots	62
10	Future Enhancements	67
11	Conclusion	69
12	Bibliography	71

INTRODUCTION

OBJECTIVE:

The emergence and application of information and communication, technologies have changed the tools that people use in their daily life. However not all the collectives use the technology in the same way. One case to take especially into account is older people. For them technology should be an inclusive factor but it can be also exclusive. With flexible apps for the common people, with adaptations that would make older peoples' lives easier this app is created. A 2018 survey of adults found that 76% of those aged 50 and older wanted to continue to live in their current residence as long as possible. Seniors who want to age at home require a variety of services including transportation, food delivery, medicines, groceries, housekeeping, etc. When things settle down, the app's primary aim is to provide a piece of mind "for senior citizens as well as their children" living away or abroad. Their children can look at the app, and be in the loop. They no longer have to ask favours from neighbours, and can live their daily life independently.

As the official Integrated Development Environment for all Flutter applications, Android Studio seems to be at the top list of preferred tools for developers. Our idea is to develop an E-Services App by using Android Development Toolkit.

SYSTEM ANALYSIS

System analysis:

System analysis is the detailed study of various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optional or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to the system.

- **Technical Feasibility**
- **Economic Feasibility**
- **Physical Feasibility**

Technical Feasibility:

The System Developer's task is to view the needed capabilities in light of currently available technologies. Our app works hand in hand with high technologies. Security and usability are considered as the most important factors of the system design that increases the user friendliness of the system. A database has to be maintained in order to update and backup whenever necessary. To create a database, we used Firebase.

Economic Feasibility:

In making recommendations, a study of the economics of the proposed system should be made. Even though finding out the cost of the proposed project is difficult we assume and estimate the costs and benefits as follows:

It is a cost effectively designed app and it can be categorized in two ways.

- Cost associated with the services.
- Cost associated with the Android OS.

Physical Feasibility:

A system resembling a mini authentication activity which is physically a less component space to make the system with high potential security. So the desired E-Service App is physically feasible indeed.

Existing System:

E-Service App is only for ordering foods and booking several services. Online payments such as credit or debit cards are not possible.

Drawbacks of Existing System:

- Online Payments are not possible.
- Limited stocks available.
- Through online, once an order is placed it cannot be cancelled.

PROPOSED SYSTEM:

SPARKCART is a mobile-based e-service app. E-Services are services which use information and communication technologies. The three main components of e-services are service provider, service receiver and the channels of service delivery. It connects users and admin in an online community allowing users to select the services and book them. Admin can view the bookings and can hire people suitable for those services.

Our project has used Flutter and Firebase to back the interface with strong database functionality. For bookings, SPARKMART has many categories like checkboxes for selection, voice notes, prescription images and this will integrate the Firebase as well as front-end booking interface.

Some of the major cases include user login, booking services, ordering foods using our Firebase database.

ADVANTAGES:

- Can be used by all mobile users
- Less memory needed to run this app
- Inventory management
- Effective Marketing.

PROJECTDESC RIPTION

PROJECT DESCRIPTION:

The emergence and application of the information and communication technologies have changed the tools that people use in their daily life. However not all the collectives use the technology in the same way. One case to take especially into account is older people. For them technology should be an inclusive factor but it can be also exclusive. This app has some special abilities and features for older people; moreover they do not always meet their needs. In order to satisfy the daily needs this app has been created for their easy use.

Our project has used Flutter and Firebase to back the interface with strong database functionality. For bookings, SPARKMART has many categories like checkboxes for selection, voice notes, prescription images and this will integrate the Firebase as well as front-end booking interface. It connects users and admin in an online community allowing users to select the services and book them. Admin can view the bookings and can hire people suitable for those services.

SYSTEM SPECIFICATIONS

SOFTWARE REQUIREMENT:

- FRONT END: Android Studio/Flutter
- CODING: Dart
- BACK END: Firebase
- OPERATING SYSTEM: Windows XP/10

HARDWARE REQUIREMENT:

- INTEL DUAL CORE 2.0Ghz CPU
- 2GB RAM
- 320GB HDD
- Mobile/Tablet

NETWORK REQUIREMENT:

- Internet Connection

SOFTWARE DESCRIPTION:

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance and commercially sponsored by Google. It was unveiled in November 2007, with the first commercial Android device launched in September 2008. It is free and open source software; its source code is known as Android Open Source Project (AOSP), which is primarily licensed under the Apache License. However most Android devices ship

with additional proprietary software pre-installed, most notably Google Mobile Services (GMS) which includes core apps such as Google Chrome, the digital distribution platform Google Play and associated Google Play

Services development platform. Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of January 2021, the Google Play Store features over 3 million apps. The current stable version is Android 11, released on September 8, 2020.

SOFTWARE DEVELOPMENT KIT:

Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.

The first version of Flutter was known as codename "Sky" and ran on the Android operating system. It was unveiled at the 2015 Dart developer summit, with the stated intent of being able to render consistently at 120 frames per second. During the keynote of Google Developer Days in Shanghai, Google announced Flutter Release Preview 2, which is the last big release before Flutter 1.0. On December 4, 2018, Flutter 1.0 was released at the Flutter Live event, denoting the first "stable" version of the Framework. On December 11, 2019, Flutter 1.12 was released at the Flutter Interactive event.

On May 6, 2020, the Dart SDK in version 2.8 and the Flutter in version 1.17.0 were released, where support was added to the Metal API, improving

performance on iOS devices (approximately 50%), new Material widgets, and new network tracking.

On March 3, 2021, Google released Flutter 2 during an online Flutter Engage event. This major update brought official support for web-based applications as well as early-access desktop application support for Windows, MacOS, and Linux.

Framework architecture:

- Dart platform
- Flutter engine
- Foundation library
- Design-specific widgets

Dart platform

Flutter apps are written in the Dart language and make use of many of the language's more advanced features.

On Windows, macOS, and Linux Flutter runs in the Dart virtual machine, which features a just-in-time execution engine. While writing and debugging an app, Flutter uses Just In Time compilation, allowing for "hot reload", with which modifications to source files can be injected into a running application. Flutter extends this with support for stateful hot reload, where in most cases changes to source code are reflected immediately in the running app without requiring a restart or any loss of state.

Release versions of Flutter apps are compiled with ahead-of-time (AOT) compilation on both Android and iOS, making Flutter's high performance on mobile devices possible.

Flutter engine

Flutter's engine, written primarily in C++, provides low-level rendering support using Google's Skia graphics library. Additionally, it interfaces with platform-specific SDKs such as those provided by Android and iOS. The Flutter Engine is a portable runtime for hosting Flutter applications. It implements Flutter's core libraries, including animation and graphics, file and network I/O, accessibility support, plugin architecture, and a Dart runtime and compile toolchain. Most developers interact with Flutter via the Flutter Framework, which provides a reactive framework and a set of platform, layout, and foundation widgets.

Foundation library

The Foundation library, written in Dart, provides basic classes and functions that are used to construct applications using Flutter, such as APIs to communicate with the engine.

Design-specific widgets

The Flutter framework contains two sets of widgets that conform to specific design languages: Material Design widgets implement Google's design language of the same name, and *Cupertino* widgets implement Apple's iOS Human interface guidelines.

SYSTEM CONFIGURATION

SYSTEMCONFIGURATION:

MODULES:

- User Login
- Dashboard
- About us
- Announcements
- Contact us
- Rate us on PlayStore
- Logout

USER LOGIN:

User Login is for the user who visits our app. The details should be entered and the user will receive an OTP number. Using the given OTP number the login is completed. This is the entrance module where the user must sign in for further knowledge about our app.

Dashboard:

Dashboardpage gives you an overview of services available in the app.

There are 4 categories available in our app:

- Transport
 - Car
 - Bike
- Delivery
 - Food and Snacks
 - Grocery
 - Vegetables
 - Medicine
 - Fruits

- Household Services
 - Carpenter
 - Painter
 - Home Tuition
 - Cook
 - Electrician
 - Security
 - Plumber
 - Mechanic
 - Welder
- Services
 - Land sale
 - Catering
 - Housekeeping
 - Construction
 - Tuition
 - Event management

About Us:

User can know about the quality of service provided by the admin.

Announcements:

This gives information about the available offers or any advertisements based on the app for the users.

CONTACT US:

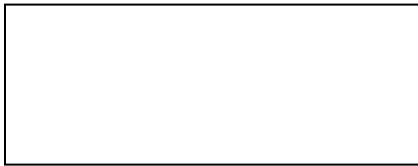
Users can contact us through the phone number mentioned in our contact info.

DATA-FLOW DIAGRAM

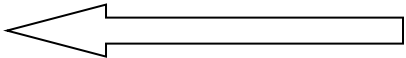
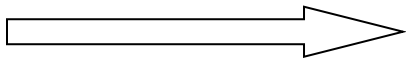
DATA-FLOW DIAGRAM:

Data flow is the movement of data from a specific origin to a specific destination indicated by line and arrows. Data flow diagram is the graphical representation of a data movement, processes and files used in the support of an information system.

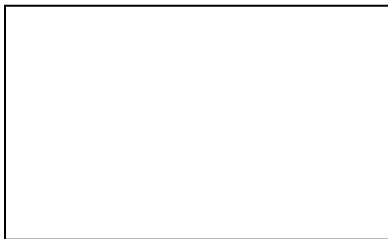
DATA-FLOW DIAGRAM SYMBOLS:



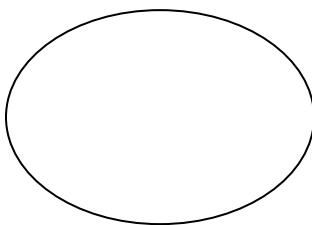
Source and Destination



Data Flow

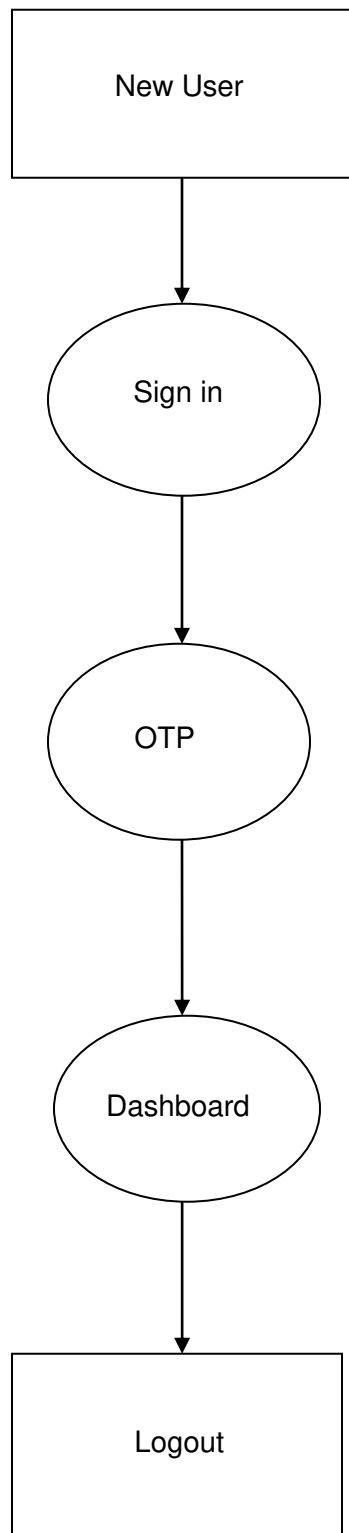


Database

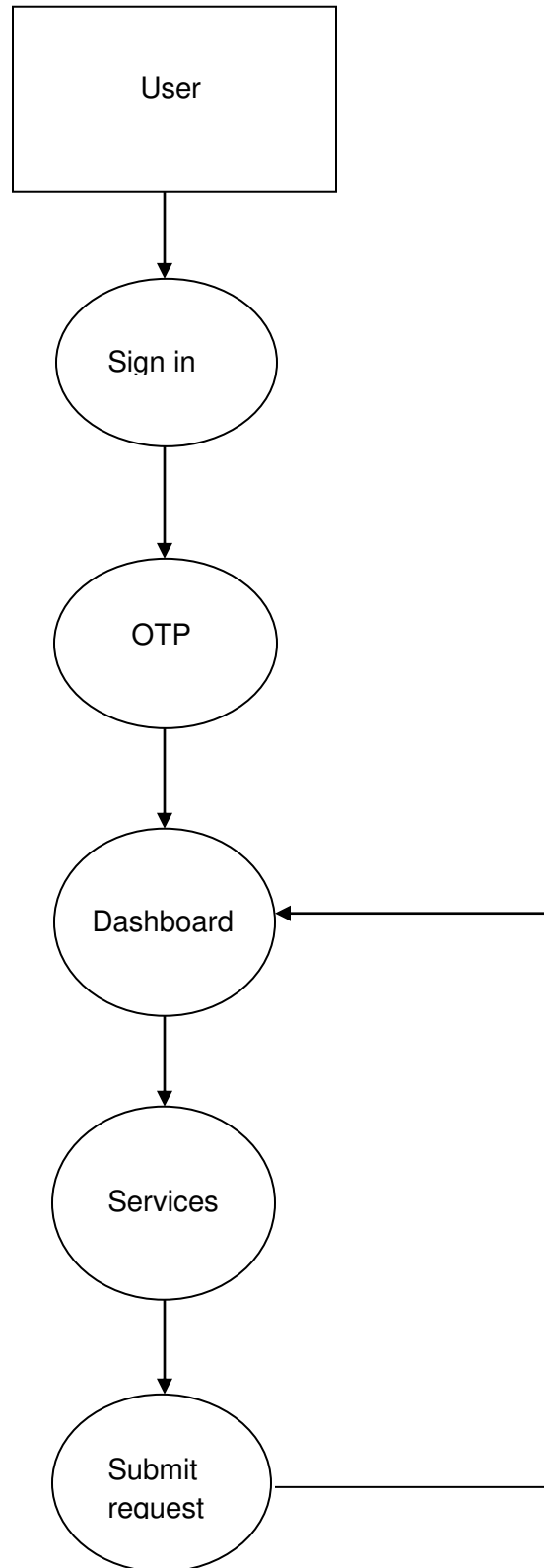


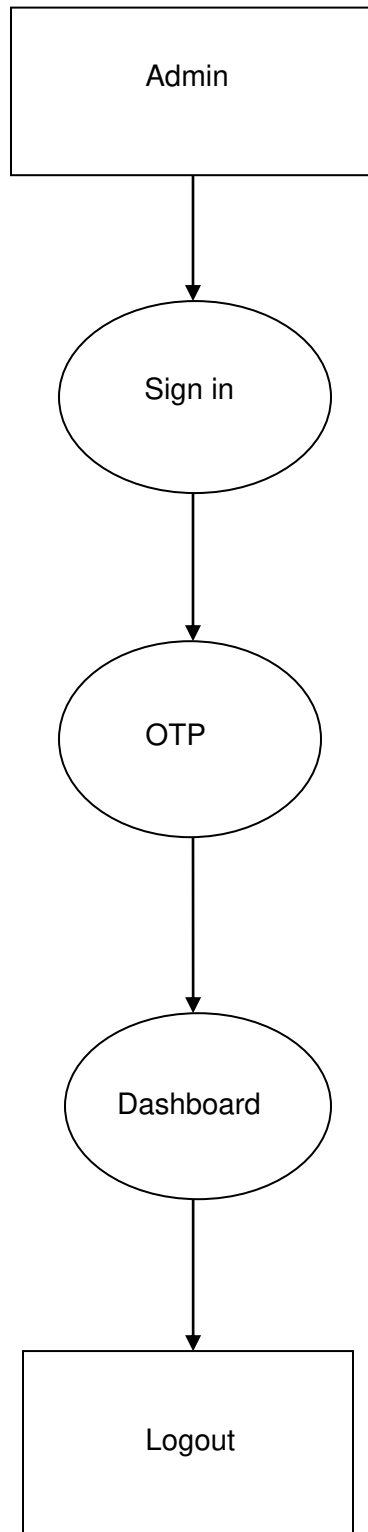
Process

New User:



User:





TESTING IMPLEMENTATION

27

SYSTEM IMPLEMENTATION:

Implementation is the most crucial stage in achieving a successful system and giving the user's confidence that the new system is workable and effective. Implementation of a modified application is to replace an existing one. This type of conversion is relatively easy to handle, provided there are no major changes in the system.

Each program is tested individually at the time of development using the data and has verified that this program linked together in the way specified in the programs specification, the computer system and its environment is tested to the satisfaction of the user. The app that has been developed is accepted and proved to be satisfactory for the user. A simple operating procedure is included so that the user can understand the different functions clearly and quickly.

TESTING:

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of conditions known as test cases and the output is evaluated to determine whether the program is performing as expected.

Software testing is the process of testing the functionality and correctness of software by running it. Process of executing a program with the intent of finding an error.

A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an

undiscovered error. Software testing is usually performed for two reasons.

- ~~Defect detection~~
- **Reliability estimation**

TESTING OBJECTIVES:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probability of finding an as yet undiscovered.
- A successful test is one that uncovers an undiscovered error.

TESTING PRINCIPLES:

- All tests should be traceable to customer requirements.
- Tests should be planned large before testing begins.
- Testing should begin “In the Small” and progress towards “In the Large”.

VALIDATION:

The system has been tested and implemented successfully and thus ensured that all the requirements as listed in the software requirements specification are completely fulfilled. In case of erroneous input, corresponding error messages are displayed.

Source Code

30

//login.dart

```

import 'package:eservices/home/homepage.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:eservices/login/phoneauth.dart';

class LoginPage extends StatefulWidget {
  final bool hasdata;
  LoginPage({this.hasdata=false});
  @override
  _LoginPageState createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  final GlobalKey<ScaffoldState> _scaffoldKey = new GlobalKey<ScaffoldState>();
  final formKey = new GlobalKey<FormState>();
  bool _isLoading = false;
  String phoneNo, verificationId, smsCode,name;
  bool codeSent = false;
  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: Scaffold(
        key: _scaffoldKey,
        body: ListView(
          children: <Widget>[
            Stack(
              children: <Widget>[
                Container(
                  width: MediaQuery.of(context).size.width-20,
                  child: Image.asset('assets/images/logo.jpg',
                    fit: BoxFit.cover,),
                ),
              ],
            ),
          ],
        ),
      ),
    );

    widget.hasdata?
    Positioned(
      right: 0.5,

```

```

      child:
        Column(
          children: <Widget>[
            IconButton(
              onPressed: () {
                Navigator.pushReplacement(context, MaterialPageRoute(builder:
(context)=>Homepage()),);

                },
              icon: Icon(Icons.home),
            ),
            Text('HOME')
          ],
        ),):
        Positioned(
          right: 0.5,
          child: FlatButton(onPressed: (){
            Navigator.pushReplacement(context, MaterialPageRoute(builder:
(context)=>Homepage()),);
          }, child: Text('Skip',
            style: TextStyle(
              fontSize:17.0,
            )),
          )
        ],
      ),
      Container(
        alignment: Alignment.center,
        padding: EdgeInsets.all(41),
        child: Text(
          'Login with your credentials',

          style: TextStyle(
            fontSize: 25),
        )),

```

```
Container(
```

```
  alignment: Alignment.center,
```

```
  padding: EdgeInsets.all(10),
```

```
  child: Text(
```

```
    'Sign in',
```

```
    style: TextStyle(
```

```
      fontSize: 30),
```

```
  )),
```

```
Container(
```

```
  padding: EdgeInsets.all(10),
```

```
  child: Form(
```

```
    key: formKey,
```

```
    child: Column(
```

```
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
```

```
      children: <Widget>[
```

```
        TextFormField(
```

```
          autofocus: true,
```

```
          decoration: InputDecoration(
```

```
            border: OutlineInputBorder(
```

```
              borderRadius: BorderRadius.circular(35.0),
```

```
            ),
```

```
            labelText: 'Name',
```

```
            prefixIcon: Icon(Icons.person, color: Colors.black),
```

```
          ),
```

```
          validator: (val) => val.isEmpty ? 'Please enter your Name' : null,
```

```
          onSave: (val) => name = val
```

```
        ),
```

```
        SizedBox(height:20),
```

```
        TextFormField(
```

```
          initialValue: '+91',
```

```
          decoration: InputDecoration(
```

```
            border: OutlineInputBorder(
```

```
              borderRadius: BorderRadius.circular(35.0),
```

```

        ),
        labelText: 'Phone Number',
        prefixIcon: Icon(Icons.phone, color: Colors.black,),
    ),
    keyboardType: TextInputType.phone,
    validator: (val) => val.isEmpty ? 'Please enter your Phone Number' : null,
    onSaved: (val) => phoneNo = val,
  ), SizedBox(height: 20.0),
  codeSent ? TextFormField(
    keyboardType: TextInputType.phone,
    decoration: InputDecoration(
      suffix: FlatButton(onPressed: ()=>verifyPhone(phoneNo), child:
Text('Resend OTP')),
      hintText: 'Enter OTP'),
    onChanged: (val) {
      setState(() {
        this.smsCode = val;
      });
    },
  ) : Container(),
  SizedBox(height: 20.0),
  Container(
    width: MediaQuery.of(context).size.width,
    child: FloatingActionButton.extended(
      label: codeSent ? Text('Verify'):Text('Send OTP'),
      onPressed: () async{
        final form = formKey.currentState;
        form.save();
        if(form.validate()){
          setState(() {
            _isLoading = true;
          });
          if(codeSent == true){
            AuthService().signInWithOTP(smsCode,
verificationId,name,phoneNo,context);

```

```

        widget.hasdata==true??Navigator.pushReplacement(context,
MaterialPageRoute(builder: (context)=>Homepage()));
        setState(() {
            _isLoading = false;
        });

    }
    else{

        if(_isLoading==true){
            verifyPhone(phoneNo);
        }
        _scaffoldKey.currentState.showSnackBar(new SnackBar(
            duration: Duration(seconds: 8),
            content: new Text('Please Wait...'),),);
    }
    })),
    ),
    SizedBox(height: 20,),
    ],
    )),
    ),
    ],
    ),
    ),
    );
}

```

```

Future _buildErrorDialog(BuildContext context, _message) {
    return showDialog(
        builder: (context) {
            return AlertDialog(
                title: Text('Error Message'),
                content: Text(_message),

```

35

```

actions: <Widget>[
    FlatButton(
        child: Text('Cancel'),

```



```

        onPressed: () {
          Navigator.of(context).pop();
        })
      ],
    );
  },
  context: context,
);
}

```

```

Future<void> verifyPhone(phoneNo) async {
  final PhoneVerificationCompleted verified = (AuthCredential authResult) {
    AuthService().signIn(authCreds: authResult, phone: phoneNo, name: name);
  };

```

```

  final PhoneVerificationFailed verificationfailed =
    (AuthException authException) {
      return _buildErrorDialog(context, authException.message);
    };

```

```

  final PhoneCodeSent smsSent = (String verId, [int forceResend]) {
    this.verificationId = verId;
    setState(() {
      this.codeSent = true;
    });
  };

```

```

  final PhoneCodeAutoRetrievalTimeout autoTimeout = (String verId) {
    this.verificationId = verId;
  };

```

```

  await FirebaseAuth.instance.verifyPhoneNumber(
    phoneNumber: phoneNo,
    timeout: const Duration(seconds: 45),

```

36

```

verificationCompleted: verified,
verificationFailed: verificationfailed,

```

```

        codeSent: smsSent,
        codeAutoRetrievalTimeout: autoTimeout);
    }
}

```

//homepage.dart

```

import 'dart:async';
import 'package:eservices/home/about.dart';
import 'package:eservices/home/deals.dart';
import 'package:eservices/login/login.dart';
import 'package:eservices/login/phoneauth.dart';
import 'package:permission_handler/permission_handler.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:eservices/home/service.dart';
import 'package:eservices/home/announcement.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:flutter/rendering.dart';
import 'package:geolocator/geolocator.dart';
import 'package:android_intent/android_intent.dart';

```

```

class Homepage extends StatefulWidget {
  final FirebaseUser currentUser;
  Homepage({this.currentUser});
  @override
  _HomepageState createState() => _HomepageState();
}

```

37

```

class _HomepageState extends State<Homepage> {
  Future _checkGps() async {
    if (!(await Geolocator().isLocationServiceEnabled())) {

```

```

if (Theme.of(context).platform == TargetPlatform.android) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text("Can't get current location"),
        content:
          const Text('Please make sure you enable GPS and try again'),
        actions: <Widget>[
          FlatButton(
            child: Text('Enable GPS'),
            onPressed: () {
              final AndroidIntent intent = AndroidIntent(
                action: 'android.settings.LOCATION_SOURCE_SETTINGS');
              intent.launch();
              Navigator.of(context, rootNavigator: true).pop();
            },
          ),
        ],
      );
    },
  );
}

@override
void initState(){
  // TODO: implement initState
  super.initState();
  [Permission.microphone, Permission.storage, Permission.location].request();
  _checkGps();
}

38

@override
Widget build(BuildContext context) {
  return SafeArea(

```

```
child: Scaffold(
```

```
  drawer: Drawer(
```

```
    child: ListView(
```

```
      children: <Widget>[
```

```
        UserAccountsDrawerHeader(
```

```
          currentAccountPicture: Image.asset('assets/images/profile.png'),
```

```
          accountName: Center(
```

```
            child: widget.currentUser == null? Text('Hi User',
```

```
            style: TextStyle(
```

```
              fontSize: 20.0,
```

```
            ),): Text(
```

```
              'Hi ${widget.currentUser.displayName}',
```

```
            ),
```

```
          ),
```

```
          accountEmail: widget.currentUser == null? null: Text(
```

```
            widget.currentUser.phoneNumber
```

```
          ),
```

```
        ),
```

```
        ListTile(
```

```
          leading: Icon(Icons.dashboard),
```

```
          title: Text('Dashboard'),
```

```
          onTap: () {
```

```
            Navigator.of(context).pop();
```

```
          },
```

```
        ),
```

```
        Divider(),
```

```
        ListTile(
```

```
          leading: Icon(Icons.info_outline),
```

```
          title: Text('About us'),
```

```
          onTap: () {
```

```
            Navigator.push(context,
```

```
              MaterialPageRoute(builder: (context) => AboutUs()));
```

```
          },
```

```
39
```

```
),
```

```
  Divider(),
```

```
  ListTile(
```

```

        leading: Icon(Icons.volume_up),
        title: Text('Announcements'),
        onTap: () {
          Navigator.push(context,
            MaterialPageRoute(builder: (context) => AnnounceMents()));
        },
      ),
      Divider(),
      ListTile(
        leading: Icon(Icons.mail_outline),
        title: Text('Contact Us'),
      ),
      Divider(),
      ListTile(
        leading: Icon(Icons.star_half),
        title: Text('Share our App'),
        onTap: () {},
      ),
      Divider(),
      ListTile(
        leading: widget.currentUser == null
          ? Icon(Icons.person):
            Icon(Icons.exit_to_app),
        title:
          widget.currentUser == null ? Text('Login/signup') :
            Text('LogOut'),
        onTap: (){
          widget.currentUser == null
            ? Navigator.push(context,
              MaterialPageRoute(builder: (context) => LoginPage())):
              AuthService().signOut();
        },
      ),
    ),

```

40

```

    Divider(),
  ],

```

```

    ),
    appBar: AppBar(
      elevation: 0.0,
      title: Text('Dhanamuthra People Service'),
    ),
    body: StreamBuilder(
      stream: Firestore.instance
.collection('mainservices')
      .snapshots(),
      builder: (context, snapshot) {
        if (!snapshot.hasData) {
          return Center(
            child: CircularProgressIndicator(),
          );
        } else {
          return ListView(
            children: <Widget>[
              Container(
                height: 75,
                child: InkWell(
                  onTap: () {
                    Navigator.push(context, MaterialPageRoute(builder:
(context)=>Deals(user: widget.currentUser,)));
                  },
                  child: Card(
                    elevation: 2.0,
                    child: Center(
                      child: Row(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: <Widget>[
                          Icon(Icons.local_offer,
                            size: 20.0,),

                          Text("Todays Deals",
                            style: TextStyle(
                              color: Colors.blue[700],

```

```

        fontSize: 20.0,
        fontWeight: FontWeight.bold,
      ),),
    ],
  ),),
),
),
),
Container(
  height: 280,
  child: Image.asset('assets/images/banner.jpeg',
fit: BoxFit.cover)
),
GridView.builder(
  physics: NeverScrollableScrollPhysics(),
  shrinkWrap: true,
  itemCount: snapshot.data.documents.length,
  gridDelegate:
    SliverGridDelegateWithFixedCrossAxisCount(
      crossAxisCount: 2),
  itemBuilder: (context, index) {
    DocumentSnapshot data =
      snapshot.data.documents[index];
    return Column(
      children: <Widget>[
        Container(
          height: 150,
          width: 250,
          child: InkWell(
            onTap: (){
              Navigator.push(
                context,

```

42

```

MaterialPageRoute(
  builder: (context) =>
    ServicePage(name: data,service:

```

```

data['name'],currentUser: widget.currentUser));
    },
    child: Card(
      shape: RoundedRectangleBorder(
        borderRadius:
          BorderRadius.circular(3.0),
      ),
      child: Image.network(
        data['image'],
        fit: BoxFit.cover,
      )),
    ),
  ),
  Container(child: Text(data['name']))
],
);
}),
],
);
}
}),
),
);
}
}

```

//AdminHome.dart

```

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:eservices/Admin/Viewimage.dart';
import 'package:eservices/Admin/announcement.dart';

```

43

```

import 'package:eservices/Admin/audio.dart';
import 'package:eservices/Admin/user.dart';

```



```

import 'package:eservices/admin/adminservice.dart';
import 'package:eservices/adminservices/newrequest.dart';
import 'package:eservices/adminservices/ongoing.dart';
import 'package:eservices/home/deals.dart';
import 'package:eservices/login/phoneauth.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:firebase_messaging/firebase_messaging.dart';
import 'package:flutter/material.dart';
import 'package:flutter_slidable/flutter_slidable.dart';
import 'package:eservices/Admin/reports.dart';
import 'package:permission_handler/permission_handler.dart';
import 'history.dart';

```

```

class AdminHome extends StatefulWidget {
  final FirebaseUser currentUser;
  AdminHome({this.currentUser});
  @override
  _AdminHomeState createState() => _AdminHomeState();
}

```

```

class _AdminHomeState extends State<AdminHome> {
  final AuthService _auth = AuthService();
  final FirebaseMessaging _firebaseMessaging = FirebaseMessaging();
  @override
  void initState() {
    _firebaseMessaging.subscribeToTopic('admin');
    super.initState();
    _firebaseMessaging.configure(
      onMessage: ((message) {
        print(message['notification']['title']);
        return;
      }),
    ),
  },

```

44

```

onLaunch: ((message) {
  print(message['notification']['title']);

```

```

        return;
    }},
    onResume: ((message) {
        print(message['notification']['title']);
        return;
    }},
);
}

```

```

@override
Widget build(BuildContext context) {
    return SafeArea(
        child: Scaffold(
            drawer: Drawer(
                child: ListView(
                    children: <Widget>[
                        UserAccountsDrawerHeader(
                            accountName: Center(
                                child: Text(
                                    'Admin',
                                    style: TextStyle(
                                        fontSize: 22.0,
                                    ),
                                ),
                            ),
                            accountEmail: null),
                        ListTile(
                            leading: Icon(Icons.dashboard),
                            title: Text('Dashboard'),
                            onTap: () {
                                Navigator.of(context).pop();
                            },
                        ),

```

45

```

        Divider(),
        ListTile(

```

```

leading: Icon(Icons.settings),
title: Text('Services'),
onTap: () {
  Navigator.push(context,
    MaterialPageRoute(builder: (context) => AdminService()));
},
),
Divider(),
ListTile(
  leading: Icon(Icons.announcement),
  title: Text('Announcements'),
  onTap: () {
    Navigator.push(context,
      MaterialPageRoute(builder: (context) => AnnounceMents()));
  },
),
Divider(),
ListTile(
  leading: Icon(Icons.local_offer),
  title: Text('Hot Deals'),
  onTap: () {
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => Deals(
          user: widget.currentUser,
        )),
    );
  },
),
Divider(),
ListTile(
  leading: Icon(Icons.list),
  title: Text('Reports'),

  onTap: () {
    Navigator.push(
      context,

```

```

        MaterialPageRoute(
            builder: (context) => ReportsPage(),
        ));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.people),
    title: Text('Users'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => UserData()));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.history),
    title: Text('History'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => HistoryPage()));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.mic),
    title: Text('Delete Audios'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => Audiopage()));
    },
),

```

```

Divider(),
ListTile(
    leading: Icon(Icons.exit_to_app),

```

```

        title: Text('logOut'),
        onTap: () async {
          await _auth.signOut();
        },
      ),
      Divider(),
    ],
  ),
),
appBar: AppBar(
  title: Text('Hello Admin'),
),
body: SingleChildScrollView(
  child: Column(
    children: <Widget>[
      Column(
        children: <Widget>[
          StreamBuilder(
            stream: Firestore.instance
.collection('requests')
              .orderBy('time')
              .snapshots(),
            builder: (context, snapshot) {
              if (!snapshot.hasData) {
                return Center(
                  child: CircularProgressIndicator(),
                );
              } else {
                return Column(
                  children: <Widget>[
                    Padding(
                      padding: const EdgeInsets.all(8.0),

child: Row(
  children: <Widget>[
    Text(

```

```

'New Requests',
style: TextStyle(
  fontSize: 20.0,
  fontWeight: FontWeight.bold,
),
),
 SizedBox(width: 10.0),
 CircleAvatar(
   radius: 12,
   child: Text(snapshot.data.documents.length
     .toString()))
 ],
 ),
 ),
 Container(
   height: 300.0,
   child: Card(
     child: ListView.builder(
       shrinkWrap: true,
       physics: ClampingScrollPhysics(),
       itemCount: snapshot.data.documents.length ==
         0
         ? 0
         : snapshot.data.documents.length == 1
         ? 1
         : 2,
       itemBuilder: (context, index) {
         DocumentSnapshot req =
           snapshot.data.documents[index];
         String url = req['services'];
         String prescription = req['prescription'];
         return Slidable(
           actionPane: SlidableBehindActionPane(),

           actionExtentRatio: 0.30,
           child: Card(
             child: Padding(

```

```
padding: const EdgeInsets.all(8.0),
child: Column(
  crossAxisAlignment:
    CrossAxisAlignment.start,
  mainAxisAlignment:
    MainAxisAlignment.spaceEvenly,
  children: <Widget>[
    Text(
      'Name : ${req['name']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Phone No : ${req['phone']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Category : ${req['service_name']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Time : ${req['time']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
  ],
),
```

```
req['service'] != null
? Text(
  'Service Name : ${req['service']}',
```

```

        style: TextStyle(
          fontSize: 18.0,
        ),
      )
    : req['prescription'] !=
      null
    ? Row(
      children: <Widget>[
        Text(
          'Prescription : '),
        Container(
          height: 70,
          width: 70,
          child: InkWell(
            onTap: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) => ViewPrescription(
                    url: prescription,
                  )),
              );
            },
            child: Card(
              child: Image.network(
                prescription,
                fit: BoxFit
                  .cover),
            ),
          ),
        ),
      ],
    )

```

```

      : PlayAudio(url: url)
    ],
  ),

```



```

    ),
    secondaryActions: <Widget>[
      IconSlideAction(
        caption: 'Move to Ongoing',
        color: Colors.green,
        icon: Icons.exit_to_app,
        onTap: () async {
          Firestore.instance
            .collection('requests')
              .document(req.documentID)
                .delete();
          req['services'] == null
            ? await Firestore.instance
              .collection('ongoing')
                .add({
                  "name": req['name'],
                  "phone": req['phone'],
                  "service_name":
                    req['service_name'],
                  "service":
                    req['service'],
                  'time': req['time'],
                  'date': req['date'],
                  'month': req['month'],
                  'year': req['year']
                })
            : await Firestore.instance
              .collection('ongoing')
                .add({
                  "name": req['name'],
                  "phone": req['phone'],
                  "service_name":
                    req['service_name'],
                  "services":

```

```

        req['services'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    });
    })
    ],
    );
    }},
    ),
    ),
    ],
    );
    }
    },
    ),
    SizedBox(height: 15.0),
    FloatingActionButton.extended(
      heroTag: Text('btn'),
      onPressed: () {
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) => NewRequest()));
      },
      label: Text('View All'),
    ),
    ],
    ),
    Column(
      children: <Widget>[

```

```

    StreamBuilder(
      stream: Firestore.instance
        .collection('ongoing')

```

```

        orderBy('time')
      ).snapshots(),
    builder: (context, snapshot) {
      if (!snapshot.hasData) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else {
        return Column(
          children: <Widget>[
            Padding(
              padding: const EdgeInsets.all(8.0),
              child: Row(
                children: <Widget>[
                  Text(
                    'Ongoing Requests',
                    style: TextStyle(
                      fontSize: 20.0,
                      fontWeight: FontWeight.bold,
                    ),
                  ),
                  SizedBox(width: 10.0),
                  CircleAvatar(
                    radius: 12,
                    child: Text(snapshot.data.documents.length
                      .toString()))
                ],
              ),
            ),
          ],
          Container(
            height: 300.0,
            child: Card(
              child: ListView.builder(
                shrinkWrap: true,
                physics: ClampingScrollPhysics(),

```

```
itemCount: snapshot.data.documents.length ==
```

```
0
```

```
? 0
```

```
: snapshot.data.documents.length == 1
```

```
? 1
```

```
: 2,
```

```
itemBuilder: (context, index) {
```

```
  DocumentSnapshot req =
```

```
    snapshot.data.documents[index];
```

```
  String url = req['services'];
```

```
  return Slidable(
```

```
    actionPane: SlidableBehindActionPane(),
```

```
    actionExtentRatio: 0.30,
```

```
    child: Card(
```

```
      child: Padding(
```

```
        padding: const EdgeInsets.all(8.0),
```

```
        child: Column(
```

```
          crossAxisAlignment:
```

```
            CrossAxisAlignment.start,
```

```
          mainAxisAlignment:
```

```
            MainAxisAlignment.spaceEvenly,
```

```
          children: <Widget>[
```

```
            Text(
```

```
              'Name : ${req['name']}',
```

```
              style: TextStyle(
```

```
                fontSize: 18.0,
```

```
            ),
```

```
          ),
```

```
            Text(
```

```
              'Phone No : ${req['phone']}',
```

```
              style: TextStyle(
```

```
                fontSize: 18.0,
```

```
            ),
```

```
          ),
```

```
            Text(
```

```
              'Time : ${req['time']}',
```

```

        style: TextStyle(
          fontSize: 18.0,
        ),
      ),
      req['service'] != null
        ? Text(
          'Service Name : ${req['service']}',
          style: TextStyle(
            fontSize: 18.0,
          ),
        )
        : PlayAudio(
          url: url,
        )
      ],
    ),
  ),
  secondaryActions: <Widget>[
    IconSlideAction(
      caption: 'Move to Completed',
      color: Colors.green,
      icon: Icons.exit_to_app,
      onTap: () async {
        Firestore.instance
          .collection('ongoing')
            .document(req.documentID)
              .delete();
        req['services'] == null
          ? await Firestore.instance
            .collection('completed')
              .add({
                "name": req['name'],
                "phone": req['phone'],
                "service_name":

```

```

        req['service_name'],
        "service":
            req['service'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    })
    : await Firestore.instance
.collection('completed')
    .add({
        "name": req['name'],
        "phone": req['phone'],
        "service_name":
            req['service_name'],
        "services":
            req['services'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    });
    })
    ],
    );
    },
    ),
    ],
    );
    }
    },
    ),

```

57

```

    SizedBox(height: 15.0),
    FloatingActionButton.extended(

```

```

heroTag: Text('btn'),
onPressed: () {
  Navigator.push(
    context,
    MaterialPageRoute(
      builder: (context) => OnGoingRequest()));
},
label: Text('View All'),
),
],
),
Column(
  children: <Widget>[
    StreamBuilder(
      stream: Firestore.instance
.collection('completed')
      .orderBy('time')
      .limit(2)
      .snapshots(),
    builder: (context, snapshot) {
      if (!snapshot.hasData) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else {
        return Column(
          children: <Widget>[
            Padding(
              padding: const EdgeInsets.all(8.0),
              child: Text(
                'Completed',
                style: TextStyle(
                  fontSize: 20.0,

                  fontWeight: FontWeight.bold,
                ),
              ),
            ),
          ],
        ),
      }
    ),
  ],
),

```

```

    ),
    Container(
      height: 300.0,
      child: Card(
        child: ListView.builder(
          shrinkWrap: true,
          physics: ClampingScrollPhysics(),
          itemCount: snapshot.data.documents.length,
          itemBuilder: (context, index) {
            DocumentSnapshot req =
              snapshot.data.documents[index];
            String url = req['services'];
            return Card(
              child: Padding(
                padding: const EdgeInsets.all(8.0),
                child: Column(
                  crossAxisAlignment:
                    CrossAxisAlignment.start,
                  mainAxisAlignment:
                    MainAxisAlignment.spaceEvenly,
                  children: <Widget>[
                    Text(
                      'Name : ${req['name']}',
                      style: TextStyle(
                        fontSize: 18.0,
                      ),
                    ),
                    Text(
                      'Phone No : ${req['phone']}',
                      style: TextStyle(
                        fontSize: 18.0,
                      ),
                    ),
                    Text(
                      'Time : ${req['time']}',
                      style: TextStyle(

```


MaterialPageRoute(

builder: (context) => HistoryPage()));

},

label: Text('View All'),

),

],

),

],

),

),

),

);

}

}

Screenshots





1:15

Von 4G+ LTE1 83%

←

car

🏠

test

☐

Other districts

☐

with in the city

☐

Other

Service Name

Submit Request

OR

Didn't Find the Service you need ?

You can also Record Audio of the Service you need

Record Audio

🎤

▶

Start

■

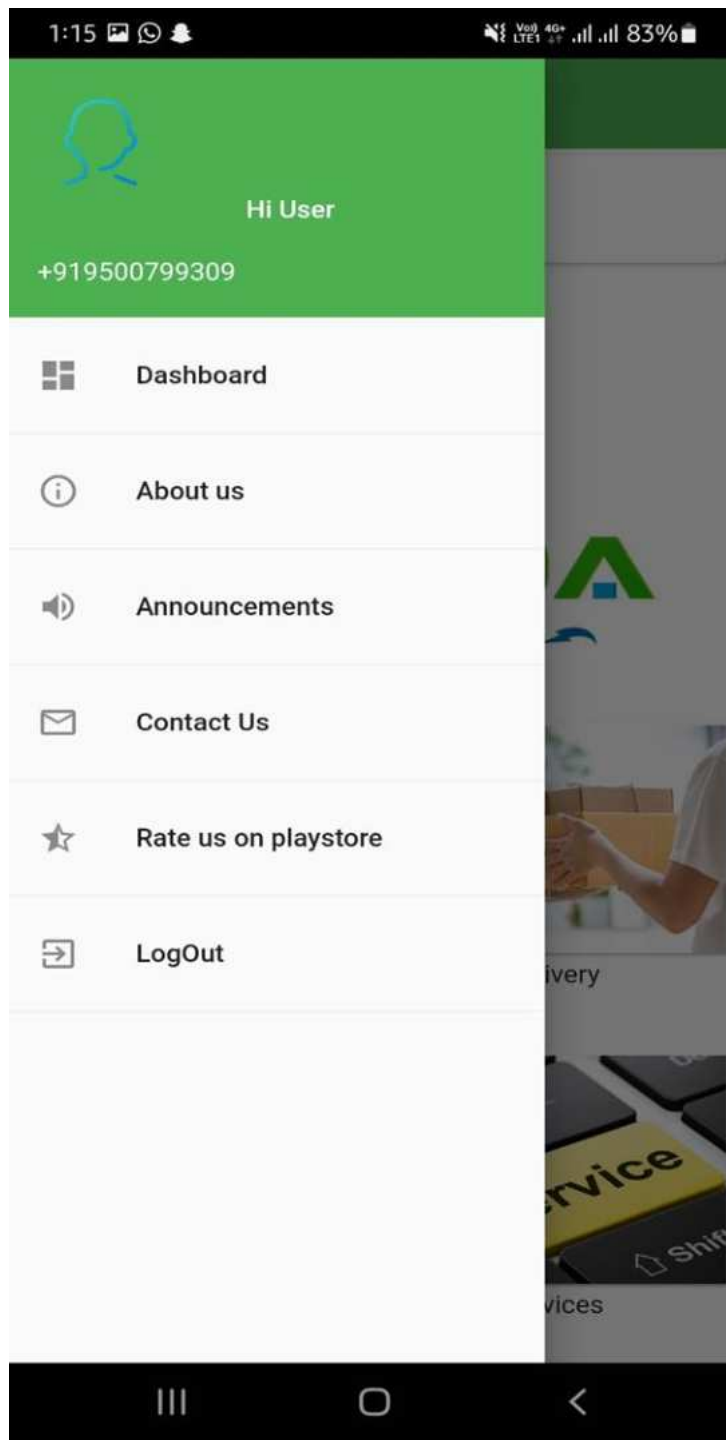
Stop

00:00

|||

○

<



Future Enhancement

No single project should be considered as complete project forever because our mind is always thinking something new and the requirement of users should be change over the time. We always want something more than what we have.

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this app:

- Make online chat facility for client and developer interaction.
- Get notifications.
- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
- Implementation of online payment mode.
- Order tracking can also be added.
- Based on future security issues, security can be improved using emerging technologies.

Conclusion

Advances in modern technology and electronic devices make the lives of people in all age groups easier and more convenient.

While we can now see adults and even kids being hooked to their mobile phone apps, utilization among senior citizens is rising too. Due to this rise in demand, we developed this app targeting the senior citizens and help them to live a better life.

This Project will thus pave the way for an efficient means to carry out the booking of the products. The main advantage of this project is to provide the basic facilities for the elderly people and full fill their needs. The User only needs basic products like a Phone and an internet connection.

Our app for the senior citizens cover all your needs like transportation, food delivery, medicines, groceries, housekeeping, etc. This is useful for senior citizens in India which will enrich their life or your loved once. Embrace technology instead of fearing or avoiding it as we live in a digital era where technology should be accessible for our ageing parents and grandparents.

Bibliography

FLUTTER REFERENCE:

<https://www.javatpoint.com/flutter>

<https://www.tutorialspoint.com/flutter/index.htm>

FIREBASE REFERENCE:

<https://www.tutorialspoint.com/firebase/index.htm>

<https://www.javatpoint.com/firebase>

ONLINE HOTEL MANAGEMENT

Project submitted to the Department of Computer Science

ST.MARY'S COLLEGE(AUTONOMOUS),THOOTHUKUDI.

Affiliated to

Manonmaniam Sundaranar University,Tirunelveli

In partial fulfilment of the requirement for the award of the degree in

BACHELOR OF COMPUTER SCIENCE

Submitted by

S.Antony Mary Tresa	-18AUCS03
B.Mahalakshmi	-18AUCS20
N.Maria Routh Selsia	-18AUCS26
S.Remma Grace	-18AUCS39
N.Syed Afia	-18AUCS47

Under the Supervision and Guidance of

Sr. K. Reetha (M.Sc., M. Phil.,)



DEPARTMENT COMPUTER SCIENCE

ST.MARY'S COLLEGE(Autonomous)

Reaccredited with "A+" Grade by NAAC

THOOTHUKUDI.

(2020 -2021)

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is certify that the project entitled

ONLINE HOTEL MANAGEMENT

Submitted by

S.Antony Mary Tresa	-	18AUCS03
B.Mahalakshmi	-	18AUCS20
N.Maria Routh Selsia	-	18AUCS26
S.Rema Grace	-	18AUCS39
N.Syed Afia	-	18AUCS47

In partial fulfilment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.


Guide


Head of the Department

Submitted for the examination held on 15.4.2021.


Examiner


Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ONLINE HOTEL MANAGEMENT



ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and blessings to finish our project successfully.

We are infinitely indebted to our college principal **Dr. Sr. A. S. J. Lucia Rose M.SC, M.Phil., PGDCA, Ph.D.**, for providing us necessary infrastructure to undertake this project and work successfully.

We are so grateful to **Mrs. S. Gnanathangam M.C.A, M.Phil (Associate professor & HOD)**, Department of Computer Science, for her gracious guidance. Our heartfelt thanks to our project guide **Mrs.S. Andrews Felcita M.Sc.,M.B.A (Assistant Professor)**, Department of Computer Science, for her valuable assistance and kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- **Mrs.M. Jothimani B.E., M.Sc., M.Phil., (Associate professor)**
 - **Sr.K.Reetha M.Sc.,M.Phil.(Assistant professor)**
 -
 - **Mrs.J. Anila Maily M.Sc , M.Phil (Associate Professor)**
- for their kind co-operation in successful completion of the project.

We would like to thank our lab assistant, **Mrs.J.Cladina**, for her kind assistance in lab during the project hours.

We thank our family members for their encouragement and support both morally and financially which helped us to finish the project successfully.

Finally we express our special thanks to our friends who gave us wonderful encouragement and ideas to complete the project.

**TABLE
OF
CONTENTS**

CONTENTS

S.NO	CONTENT	PAGE NO
1.	SYNOPSIS	1
2.	SYSTEM ANALYSIS 2.1 EXISTING SYSTEM 2.2 PROPOSED SYSTEM 2.3 ANALYSING THE SYSTEM	3
3.	SYSTEM REQUIREMENTS AND SPECIFICATION 3.1 WORKING ENVIRONMENT 3.2 ABOUT SOFTWARE	7
4.	SYSTEM DESIGN 4.1 TABLE DESIGN 4.2 DATA FLOW DIAGRAM 4.3 FLOW CHART 4.4 FORM DESIGN 4.5 CODE DESIGN	14
5.	FUTURE ENHANCEMENT	62
6.	CONCLUSION	64
7.	BIBLOGRAPHY	66

SYNOPSIS

SYNOPSIS

“HOTEL MANAGEMENT “ is a software developed for maintaining an online reservation system for rooms that will replace the manual method for booking rooms. By using this, time managing will be good. Payments will be faster and easier. The purpose of this Project is to develop and implement an online hotel reservation system for hotels that will replace the manual method of booking for hotel rooms. By using this, time managing will be good. Payments will be faster and easier. Smart works will be inside in booking .The mission is to facilitate easy management and administration of a hotel with capabilities to do Booking or reservations of the rooms, cancellation of the rooms ,cash billing ,room service, restaurant service restaurant billing, total billing, travels arrangement etc. Using the automated hotel management software. One can keep detailed records or information on an unlimited amount of customers .The system lets the user know which all rooms are available for occupancy at any point of time .This makes the booking considerably faster .And thus helps the hotel in better management and reduce a lot of paper work as well as manpower

The main objective of the entire activity is to automate the process day to day activities of hotel like:

- 1.Admission of a new customer.
- 2.Check out for booking.
- 3.Assign a room according to customer's needs.
- 4.Finaaly compute the bill.
- 5.Online cancellation.

SYSTEM ANALYSIS

SYSTEM ANALYSIS

2.1 Existing System:

Hotel Management involves maintaining various operations of the hotel like Booking or reservations of the rooms, cancellation of the rooms, cash billing, room service etc. The existing system is a manual one and there is lot of issues like erroneous data, slow process, lack of security etc. Finding out final payment amount completely relies on the hotel manager and if he is absent, it irritates a long time to find out the details during check out and is prone to errors.

- Too difficult in booking the rooms.
- Only star hotel rooms can be booked online.
- Payment details and room rents are difficult in understanding.
- Admin can't view all user details about their payment.
- No Security assurance for user information.
- Manual process are too difficult

Drawbacks of existing system:

- Manual entry consumes more time.
- It is difficult to maintain bulk of record in manual.
- Restrictions in the users.
- Not easy to prepare the daily reports.
- Lack of accuracy and error prone.
- Overall efficiency is less.
- Lot of paperwork.
- Non secure.

2.2 Proposed System:

- Simple way of online booking is produced for all types of users.
- All types of hotel rooms can be booked through online.
- Detail views of payments are providing to admin.
- Admin can view each and every user's details individually.
- User registration form and their details will be secured.
- Manual Processes are reduced.

Advantages of proposed system:

- Secure data
- Faster process
- Error Free
- Better management
- Save a lot of manpower
- Can easily make the daily reports
- Elimination of Paper work.
- High reliability and security.
- Fast and economical.

2.3 ANALYSING THE SYSTEM:

The seven types of modules are

- Login
- Booking rooms
- Client payment
- Client booking cancellation
- Rooms available
- Client details
- Payment details.

Login:

Login the page to verify the user id and admin page.

Booking Rooms:

Booking the needed rooms with available facilities called A/C and the Extra Bed.

Client Payment:

Transact the amount as per the room booked.

Client Booking Cancellation:

To cancel the room booked.

Rooms Available:

This is for admin to view the available number of rooms and types of Rooms available.

SYSTEM REQUIREMENTS AND SPECIFICATIONS

SYSTEM SPECIFICATION

3.1 WORKING ENVIRONMENT

HARDWARE SPECIFICATION

Processor	: Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz
RAM	: 4GB
Speed	: 2.40 GHz
Hard Disk Driver	: 1 TB
Display	: 15.6 1366 *786

SOFTWARE SPECIFICATION

Operating System	: Windows 10
Environment	: Visual Studio 2012
.NET FRAMEWORK	: Version 4.0
Front-End	: Active Server Page(ASP.NET)
Programming Language	: VB
Back-End	: MicroSoft Access 2007

3.2 ABOUT SOFTWARE

ASP.NET

In developing the online website, we have used ASP.NET for as the database. In which .Net framework 4 is included. Here are some features of Asp.Net 4....

ASP.NET is the server-side web application framework designed for web development to produce dynamic web pages.

Front End : ASP.NET 2010

Back End : Access server 2007

ASP.NET stands for Action Pages.NET and is developed by Microsoft. ASP.NET is used to create web pages and web technologies and is an integral part of Microsoft's .NET Framework vision. As a member of the .NET Framework, ASP.NET is a very valuable tool programmer and development it allies them to build dynamic, rich web sites and web application using compiled language like VB and C#.

ASP.NET is not limited to script language, it allows you to make use of .NET language like C#, VB etc....

It allows developers to build very compelling applications by making use of Visual Studio; the development tool provided by Microsoft ASP.NET is purely server-side technology. It is built on a common language runtime that can be used on many windows server to host powerful ASP.NET website and technologies.

Advantages ASP.NET Over web development models:

- ASP.NET drastically reduces the amount of code required to build applications.
- With built-in window authentication and per-application configuration, your application is safe and secured.
- It provides better performance by taking advantage of early binding, just-in-time.
- The ASP.NET framework is complemented by a rich toolbox and designer in the Visual Studio integrated development environment..WYSIWYG editing, drag and drop server control and

automatic deployment sure just a few of the features this powerful tool provides.

- It provides simplicity as ASP.NET makes it easy to perform common tasks, from simple form submission and client authentication to development and site configuration.
- The source code and HTML are together therefore ASP.NET pages are easy to maintain and write. Also the source code is executed on the server. This provides a lot of power and flexibility to the web pages.
- All the processor closely monitored and managed by the ASP.NET runtime, so that if process is dead, a new process can be created in its place, which helps keep your application constantly available to handle request.
- It is purely server-side technologies so ASP.NET code executes on the server before it is sent to the browser.
- Being language independent it allows you to choose the language that best applies to your application or partition your application across many languages.
- ASP.NET makes for easy deployment. There is no need to register components because the configuration information is built-in.
- The web server continuously monitors the pages, components and applications running on it. If it notices any memory leaks, infinite loops, other illegal activities.
- Easily works with ADO.NET using data binding and page formatting features. It is an application which runs faster and counters large volumes of users without having performance problems.

STANDARD CONTROLS USED:

- **BUTTON:**

Very widely used control. Writing some lines about this control is very tinny thing. It is used everywhere like have to register, admin, user, book, update, cancel, exit etc.....

- **TEXTBOX:**

This control is used to enter information for processes. Like if we are wishing to create a web page where user can enter his username and password, surely take the advantage of textbox here. We can change the Textbox, TEXTMODE PROPERTY by single line, multiline and password.

- **IMAGE:**

This control is used to link any other page. Like if you open any community website and have look your image means they have used image control. Using image URL property we can change the image.

- **RADIO BUTTON:**

This controls is used to accept data in true/false or say in yes/no Boolean format. Like if we select it then it says true or yes. BY DEFAULT ITS CHECKED PROPERTY IS FALSE but we can set it true. Radio button cannot be bind with database.

- **LABEL:**

This control is mostly used to display texts conform. Programmatically we can hide/unhide or set the text or label, and this feature helps the developer to display some message when any task completed.

- **DROP DOWN LIST:**

This control is used to select any one item from list. We cannot select multiple item from drop down list. It has also database property.

MS ACCESS

Microsoft Access is a Database Management System (DBMS) from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools. It is a member of the Microsoft Office suite of applications, included in the professional and higher editions.

- ❖ Microsoft Access is just one part of Microsoft's overall data management product strategy.
- ❖ It stores data in its own format based on the Access Jet Database Engine.
- ❖ Like relational databases, Microsoft Access also allows you to link related information easily. For example, customer and order data. However, Access 2013 also complements other database products because it has several powerful connectivity features.
- ❖ It can also import or link directly to data stored in other applications and databases.
- ❖ As its name implies, Access can work directly with data from other sources, including many popular PC database programs, with many SQL (Structured Query Language) databases on the desktop, on servers, on minicomputers, or on mainframes, and with data stored on Internet or intranet web servers.
- ❖ Access can also understand and use a wide variety of other data formats, including many other database file structures.
- ❖ You can export data to and import data from word processing files, spreadsheets, or database files directly.
- ❖ Access can work with most popular databases that support the Open Database Connectivity (ODBC) standard, including SQL Server, Oracle, and DB2.
- ❖ Software developers can use Microsoft Access to develop application software.

SYSTEM DESIGN

TABLE DESIGN

4.1 TABLE DESIGN:

A database is a collection of information. A primary objective of a database management system is to provide a convenient environment to retrieve and store database information. The most popular and flexible organize data in the form of tables.

TABLE NAME: register

USERNAME	DATA TYPE
Admin	Short Text
Password	Short Text

TABLE NAME : register

COLUMN NAME	DATA TYPE
ClientID	Number
First Name	Long Text
Last Name	Long Text
Email-ID	Long Text
Password	Short Text

TABLE NAME: clientpayment

COLUMN NAME	DATA TYPE
ClientID	Number
Amount	Number

TABLE NAME : cancelform

COLUMN NAME	DATA TYPE
ClientID	Number

DATA FLOW DIAGRAM

4.2 DATA FLOW DIAGRAM

Data flow is a movement of origin to a specific destination indicated by lines and arrows. Data Flow Diagram is a graphical representation of a data movements, processes and files used in the support of an information system..

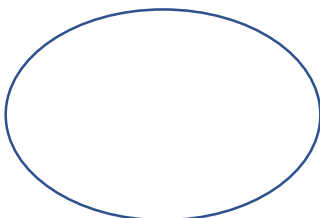
DESIGN NOTATIONS



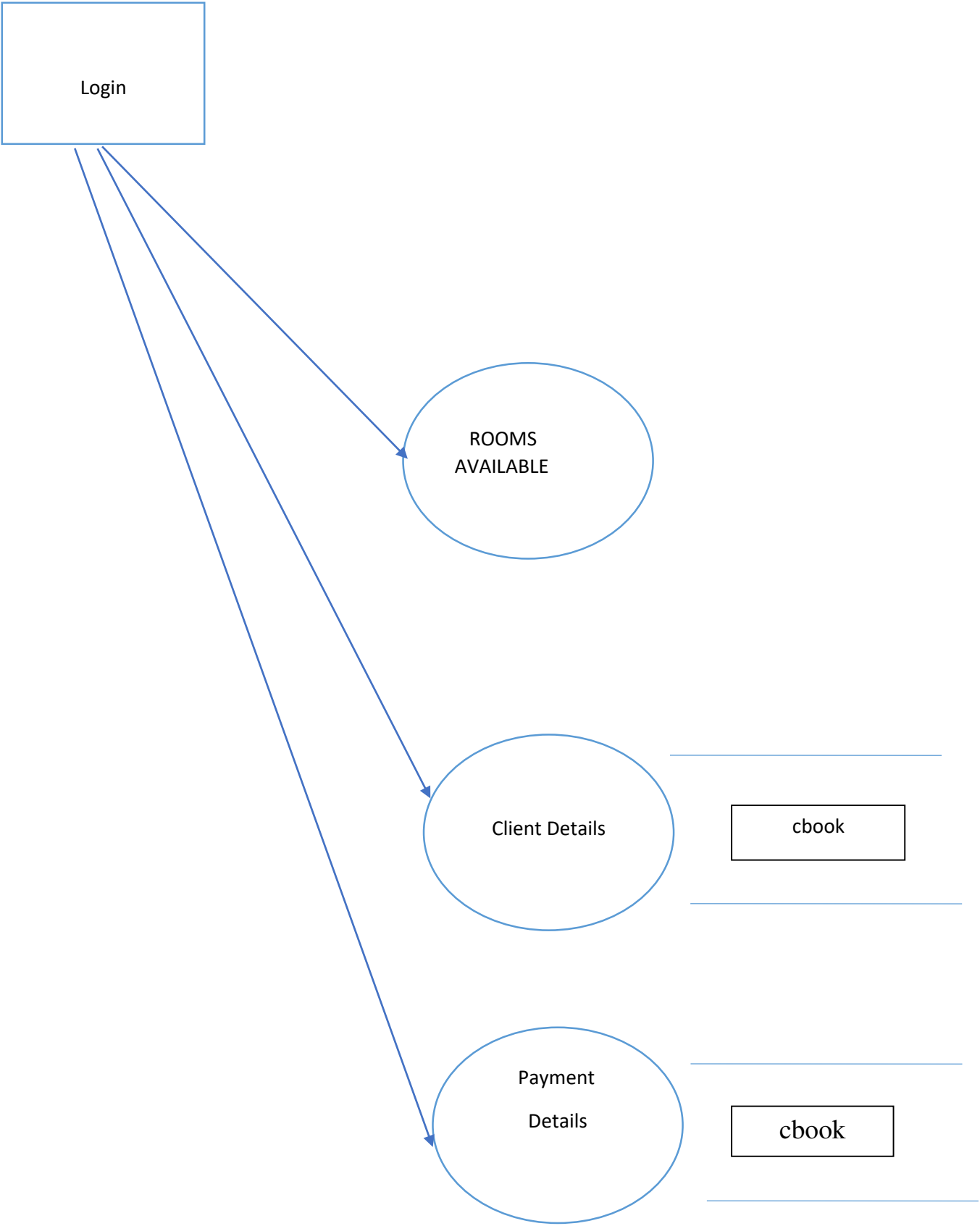
Entity

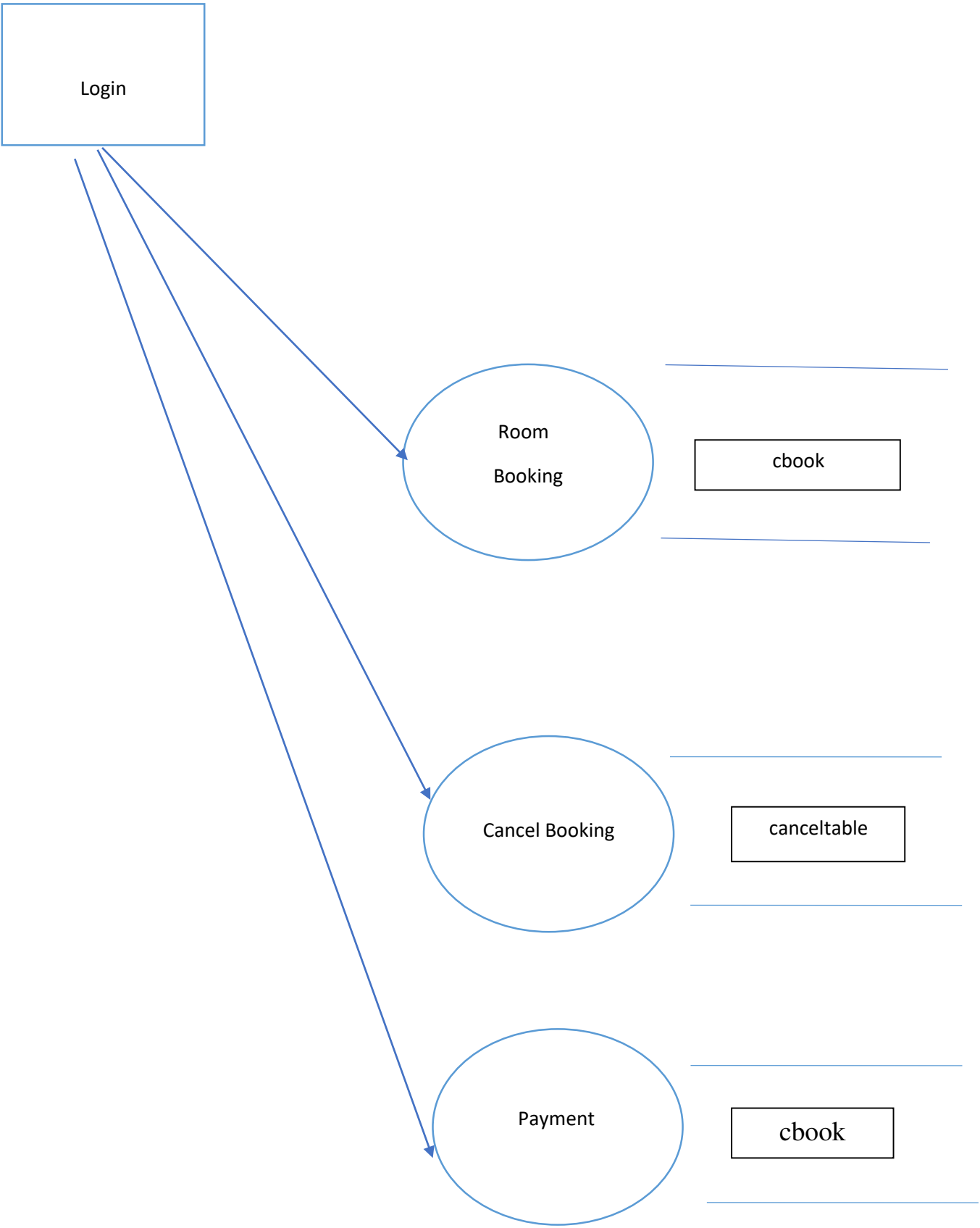


Dataflow



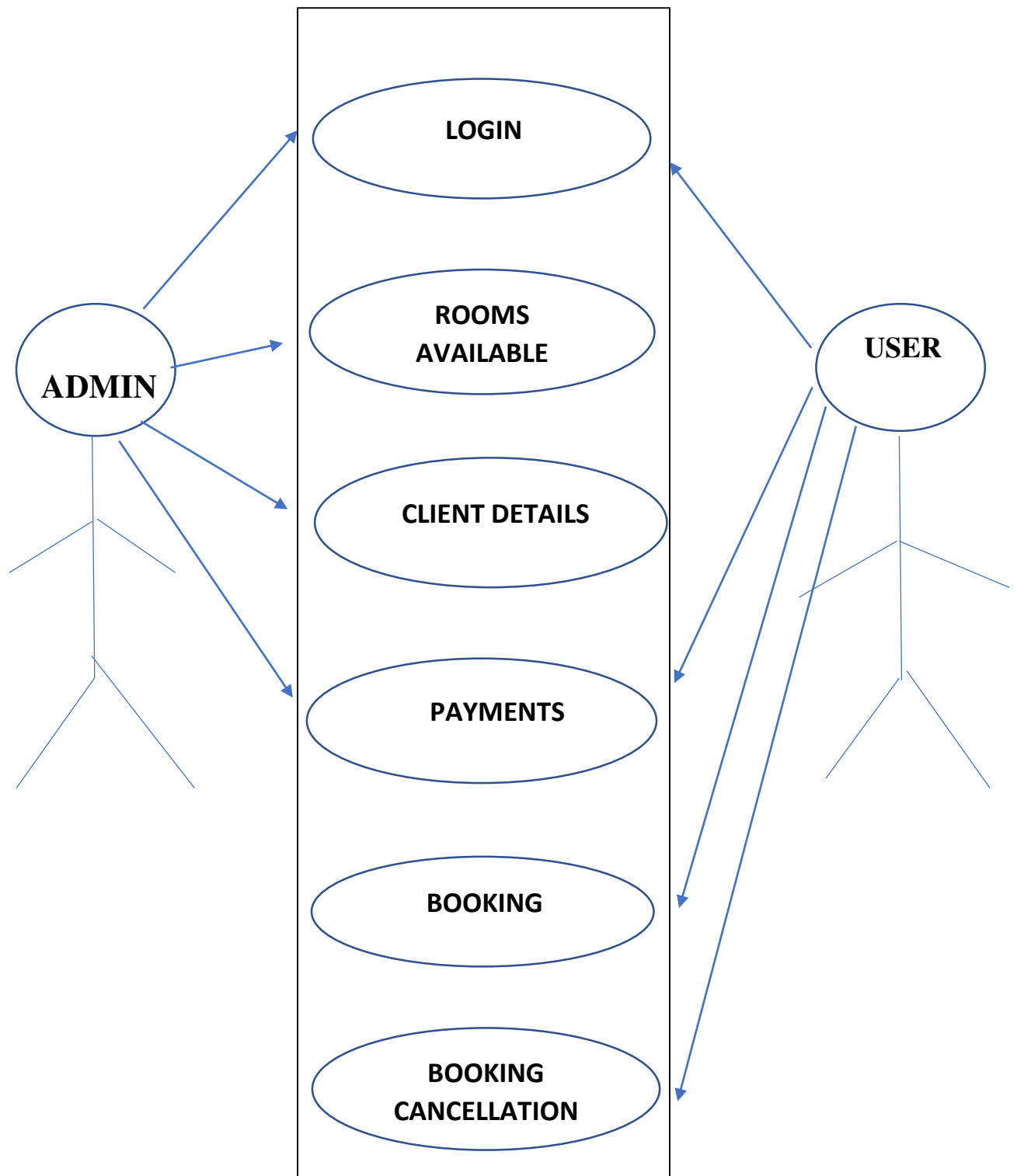
Process





FLOW CHART

4.3 FLOW CHART



FORM DESIGN

Introduction Page



Registration Page

Register

Hotel Fontana

Hotel Fontana Registration Form

Client ID	8221
First Name	Kayal
Last Name	vizhi
Phone Number	7708787809
Address	142/76,mani Street,Chennai
E-Mail ID	kayal@gmail.com
Password	kv00

Register **Reset** **Exit**

Basuri Glass Hotel

Your Registration is Completed

OK

Login Page



Admin Login Page



The image shows a web browser window titled "Hotel Fontana" with a close button (X) in the top right corner. The background of the login page features a cartoon yellow character with large eyes and a red mouth, holding a silver tablet with a black silhouette of a person. The character is positioned on the left side of the page. The background also shows a cityscape at dusk with a building that has "FONTANA" written on it.

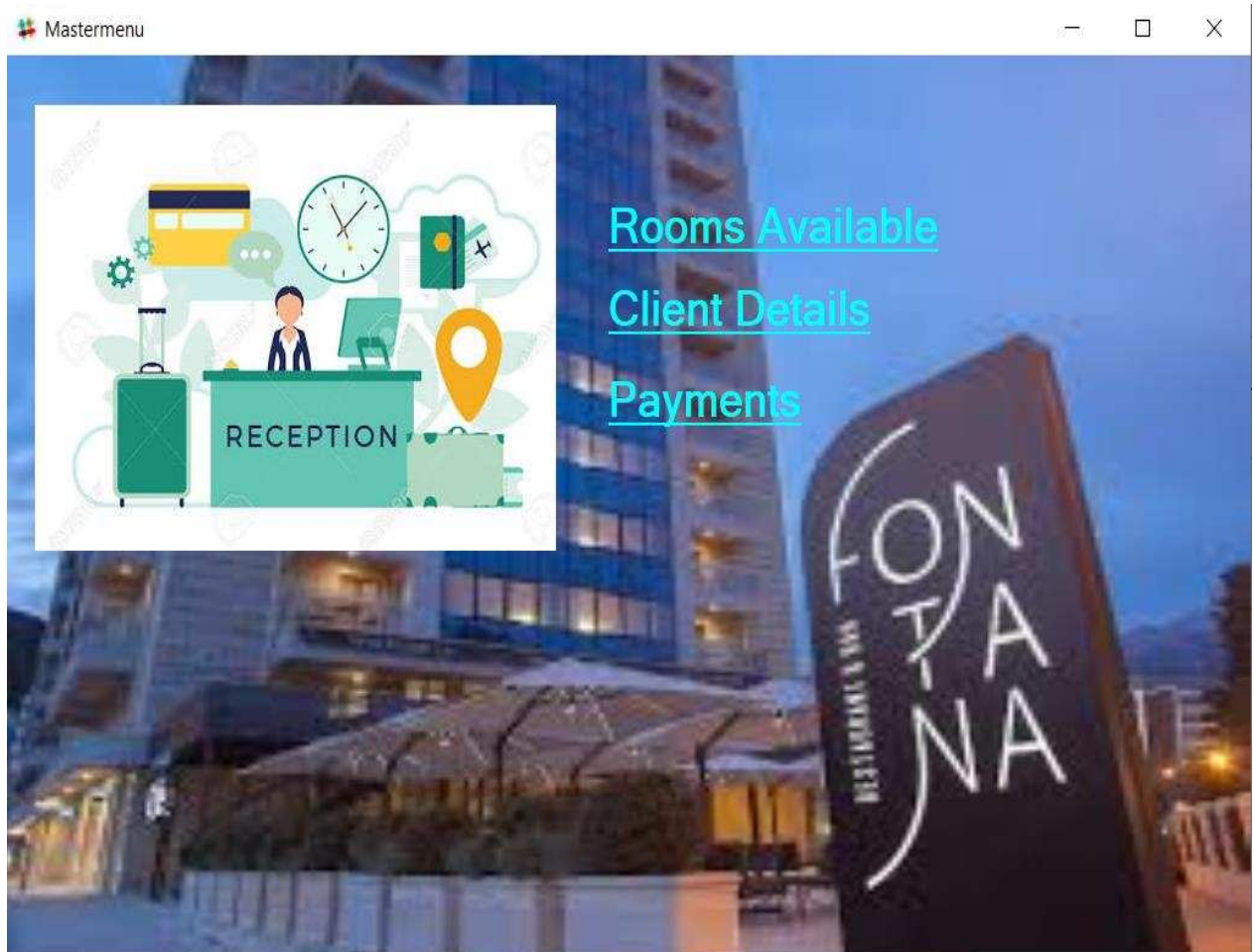
User name

admin

Password

Login **Cancel**


Master Menu



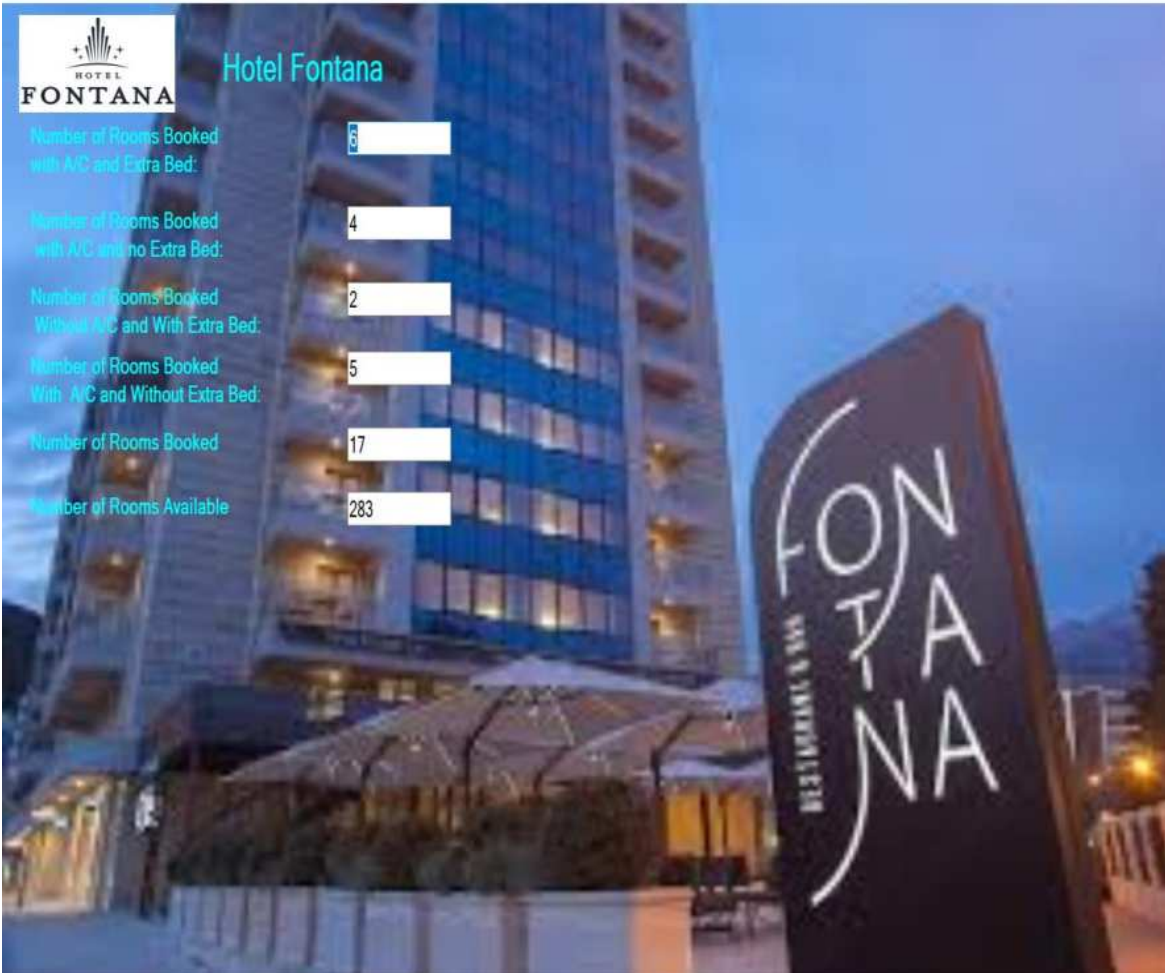
Master Rooms

Masterrooms

Hotel Fontana

 HOTEL FONTANA

Number of Rooms Booked with A/C and Extra Bed:	6
Number of Rooms Booked with A/C and no Extra Bed:	4
Number of Rooms Booked Without A/C and With Extra Bed:	2
Number of Rooms Booked With A/C and Without Extra Bed:	5
Number of Rooms Booked	17
Number of Rooms Available	283



Master Model



Hotel Fontana

	CLIENTID	ROOMR...	PHONE...	TOTAL...	F^
5454	1		9878543211	1	At
5454	1		9878543211	13	At
9787	3		9878543211	3	At
1001	4		8610235029	1	At
111	5		9750597193	1	Nc
1897	6		90989197271	1	Nc
12345	7		909898988	1	Nc
345	8		7708726308	36	At
345	8		7708726308	36	At
8989	10		9822334455	8	Nc
8989	10		9822334455	8	Nc
909	12		990990099	2	At
909	12		990990099	2	At
955	14		7708726308	12	Nc
955	14		7708726308	12	Nc



Master Payment

Masterpayment

Hotel Fontana

Complete Payment Done By:

Client ID:

Name:


Room Number:

Amount:

Payment:

CLIENTID	CNAME	ROOMR...	PHONE...	TOTAL...	AMOUNT	ROOMT...
5454	subha	1	9876543211	1	4000	A/C
5454	subha	1	9876543211	13	52000	A/C
9787	angel	3	9876543211	3	10500	A/C
1001	dekh	4	9610235029	1	3500	A/C
111	meera	5	9750967193	1	3500	Non A/C
1987	elva	6	90989197271	1	3000	Non A/C
12345	sathish	7	9898989898	1	3500	Non A/C
345	vishnu	8	7708726308	36	144000	A/C
345	vishnu	8	7708726308	36	144000	A/C
8989	durka	10	9822334455	8	24000	Non A/C
8989	durka	10	9822334455	8	24000	Non A/C
999	ashu	19	9999999999	9	30000	A/C

Client Login Page



The image shows a client login window titled "Hotel Fontana" with a close button (X) in the top right corner. The background features a cartoon yellow character holding a smartphone and a cityscape at night. The login form includes a "User name" field with the text "ashi", a "Password" field with four asterisks, and two buttons labeled "Login" and "Cancel".

Hotel Fontana

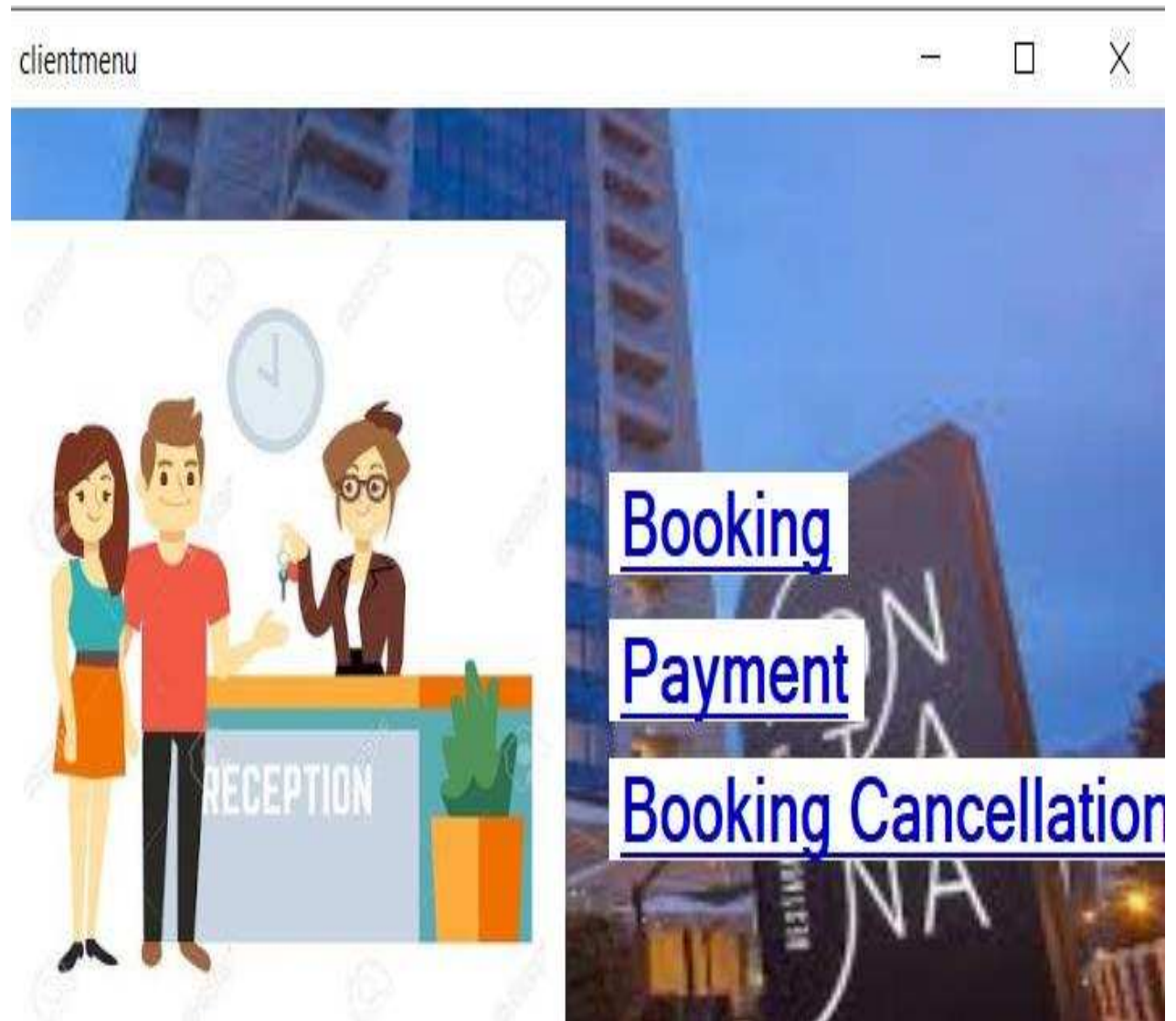
User name

ashi

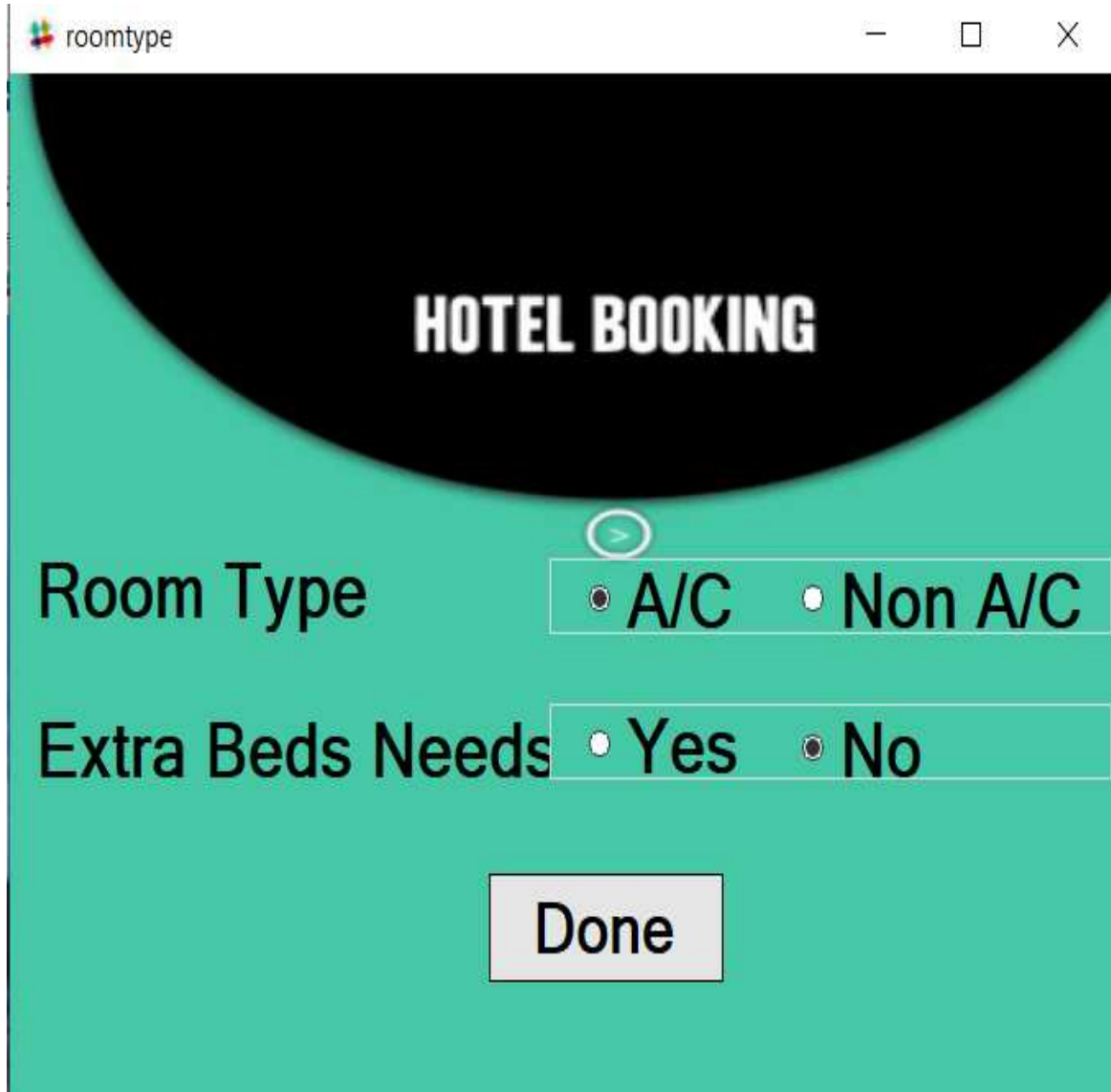
Password

Login Cancel

Client Menu



Room Type



A screenshot of a web application window titled "roomtype" with standard window controls (minimize, maximize, close). The main content area has a teal background with a black curved header at the top containing the text "HOTEL BOOKING" in white. Below the header, there are two radio button groups. The first group is labeled "Room Type" and contains two options: "A/C" (selected) and "Non A/C". The second group is labeled "Extra Beds Needs" and contains two options: "Yes" (selected) and "No". At the bottom center, there is a light gray button with the text "Done".

roomtype

HOTEL BOOKING

Room Type


☒ A/C ☐ Non A/C

Extra Beds Needs

☐ Yes ☒ No

Done

Client Booking



Basuri Glass Hotel

Client ID

Total Rooms

Rooms Booked

Rooms Available

Room Number Reseved for You

Phone Number

Day IN

Day OUT

Total Days Calculate

Amount

Reserve

Reset

Exit

1. Without A/C and Without Extra Bed Rs.3000 per day

2. With A/C and With Extra Bed Rs.4000 per day


3. Without A/C and With Extra Bed Rs.3500 per day

4. With A/C and Without Extra Bed Rs.3500 per day

ROOM ...	EXTRA ...	CLIENTID	CNAME	ROOM ...	PHONE. ^
A/C	No	1001	daksh	4	8610235028
Non A/C	Yes	111	meera	5	9750597198
Non A/C	No	1997	elsa	6	9098919727
Non A/C	Yes	12345	sathiesh	7	9898989898
A/C	Yes	345	vishnu	8	7708726308
A/C	Yes	345	vishnu	8	7708726308
Non A/C	No	8989	durka	10	9922334458
Non A/C	No	8989	durka	10	9922334458
A/C	No	909	ashi	12	9900990098
A/C	No	909	ashi	12	9900990098
Non A/C	No	955	maria	14	7708726308
Non A/C	No	955	maria	14	7708726308
A/C	Yes	5566	nesmi	16	9940898557
A/C	Yes	5566	nesmi	16	9940898557

Client Payment

clientpayment

 Hotel Fontana

Client ID

Amount

Cancel Form

cancelfrom

Hotel Fontana

Client ID 9999 [Delete](#)



	CLIENTID	CLIENT...	ROOMT...	EXTRA...	TOTAL...	TOTAL...	PAYME...
	6666	anu	A/C	No	6	21000	Not Paid
	6666	anu	A/C	No	6	21000	Not Paid
▶	9999	shree	Non A/C	No	1	3000	Not Paid
	1000	hari	Non A/C	No	6	18000	Not Paid
	1000	hari	Non A/C	No	6	18000	Not Paid
	945	vidhya	A/C	Yes	26	104000	Not Paid
	945	vidhya	A/C	Yes	26	104000	Not Paid
*							

CODE DESIGN

4.5 CODE DESIGN

HOME PAGE

Public Class intro

Private Sub Panel1_Paint(sender As Object, e As PaintEventArgs) Handles Panel1.Paint
End Sub

Private Sub LinkLabel1_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked

Form1.Show()

End Sub

End Class

FORM DESIGN

Public Class Form1

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
Dim l As New LoginForm1
l.Show()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
Dim ff As New Regis
ff.Show()

End Sub

Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
Application.Exit()

End Sub

Private Sub Label1_Click(sender As Object, e As EventArgs) Handles Label1.Click
End Sub

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
End Sub

End Class

LOGIN FORM

Imports System.Data.OleDb

Public Class LoginForm1

Dim read As String

Dim connstring As String

Dim datafile As String

Dim myconnection As OleDbConnection = New OleDbConnection

' TODO: Insert code to perform custom authentication using the provided username and
password

' (See <http://go.microsoft.com/fwlink/?LinkId=35339>).
 ' The custom principal can then be attached to the current thread's principal as follows:
 ' My.User.CurrentPrincipal = CustomPrincipal
 ' where CustomPrincipal is the IPrincipal implementation used to perform authentication.
 ' Subsequently, My.User will return identity information encapsulated in the CustomPrincipal object

' such as the username, display name, etc.
 Private Sub OK_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles OK.Click
 Me.Close()
 con.Open()
 cmd.Connection = con
 cmd.CommandText = "Select*from registerdb"
 Dim dr As OleDbDataReader = cmd.ExecuteReader
 dr.Read()
 If UsernameTextBox.Text = dr("FIRSTNAME") And PasswordTextBox.Text = dr("PASSWORD") Then
 ' MessageBox.Show("WELCOME TO OUR HOTEL FONTANA")
 Dim ff As New clientmenu
 ff.Show()
 read = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" &
 datafile = "C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-001\hotelmanagement\hotelmanagement\registerdb.accdb"
 connstring = read & datafile
 myconnection.ConnectionString = connstring
 Dim cmd As OleDbCommand = New OleDbCommand("select *from[registerdb] where[FIRSTNAME]='" & UsernameTextBox.Text & "' and [PASSWORD]='" & PasswordTextBox.Text & "'", myconnection)
 myconnection.Open()
 Dim dr As OleDbDataReader = cmd.ExecuteReader
 Dim userfound As Boolean = False
 Dim firstname As String = ""
 Dim lastname As String = ""
 Dim cid As String = " "
 While dr.Read
 userfound = True
 firstname = dr("FIRSTNAME").ToString
 lastname = dr("LASTNAME").ToString
 cid = dr("CLIENTID").ToString
 End While
 myconnection.Close()
 If userfound = True Then
 Dim ff As New clientmenu
 ff.Show()
 roomtype.Label3.Text = UsernameTextBox.Text
 roomtype.Label4.Text = PasswordTextBox.Text
 ff.Label1.Text = UsernameTextBox.Text
 ff.Label2.Text = PasswordTextBox.Text
 ff.TextBox1.Text = cid
 ElseIf UsernameTextBox.Text = "admin" And PasswordTextBox.Text = "admin" Then

```

        MessageBox.Show("Welcome Admin")
        mastermenu.Show()
    ElseIf UsernameTextBox.Text = "" Then
        MessageBox.Show("Please enter the username")
    ElseIf PasswordTextBox.Text = "" Then
        MessageBox.Show("Please enter the password")
    Else
        MessageBox.Show("Incorrect User Name or Password")
    End If
End While
'con.Close()
'cmd.ExecuteNonQuery()
End Sub
Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Cancel.Click
    Me.Close()
End Sub
Private Sub LoginForm1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
End Sub
End Class

```

REGISTER FORM

```

Imports System.Text.RegularExpressions
Imports System.Data.OleDb
Public Class Regis
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    Private Sub TextBox4_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox4.KeyPress
        Dim cc As Integer
        cc = Len(TextBox3.Text)
        If cc > 10 Or cc < 10 Then
            Label9.Visible = True
            TextBox3.Text = ""
        Else
            Label9.Visible = False
        End If
    End Sub
    Private Sub TextBox4_TextChanged(sender As Object, e As EventArgs) Handles
TextBox4.TextChanged
End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        con.Open()
        'read = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source="
        'datafile = "D:\cars\hotelmanagement\hotelmanagement\registerdb.accdb"
    End Sub

```

```

'connstring = read & datafile
'con.ConnectionString = connstring
Dim cmd As OleDbCommand = New OleDbCommand("select *from[registerdb]
where[CLIENTID]='" & TextBox7.Text & "'", con)
Dim dr As OleDbDataReader = cmd.ExecuteReader
Dim userfound As Boolean = False
Dim firstname As Integer
While dr.Read
    userfound = True
    firstname = dr("CLIENTID").ToString
End While
con.Close()
If userfound = True Then
    MessageBox.Show("This client id is already exist please try another.....")
    Me.Close()
ElseIf TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or
TextBox4.Text = "" Or EmailTextBox.Text = "" Or TextBox6.Text = "" Then
    MessageBox.Show("Please Enter the Details Completely..", "Hotel Fontana",
MessageBoxButtons.OK)
Else
    con.Open()
    cmd.Connection = con
    cmd.CommandText = "insert into registerdb values(" & TextBox7.Text & "," &
TextBox1.Text & "," & TextBox2.Text & "," & TextBox3.Text & "," & TextBox4.Text & "," &
+ EmailTextBox.Text & "," & TextBox6.Text & ")"
    cmd.ExecuteNonQuery()
    MessageBox.Show("Your Registration is Completed", "Hotel Fontana",
MessageBoxButtons.OK)
    con.Close()
End If
End Sub
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    EmailTextBox.Text = ""
    TextBox6.Text = ""
    TextBox7.Text = ""
End Sub
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    Me.Close()
End Sub
Private Sub Regis_Load(sender As Object, e As EventArgs) Handles MyBase.Load
End Sub
Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles
TextBox1.TextChanged
End Sub
Private Sub TextBox6_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox6.KeyPress

```

```

    Dim tt As String
    tt = "^[a-zA-Z0-9_\.]+\)((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.)(\([a-zA-Z0-9\-\]\.)+))([a-zA-Z]{2,4}|\[0-9]{1,3})\(\)?\$"
    Dim eam As Match = Regex.Match(EmailTextBox.Text, tt)
    If eam.Success Then
        Label10.Visible = False
    Else
        Label10.Visible = True
        EmailTextBox.Text = ""
    End If
End Sub
Private Sub TextBox6_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox6.TextChanged
End Sub
Private Sub Label9_Click(sender As Object, e As EventArgs) Handles Label9.Click
End Sub
Private Sub TextBox3_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox3.TextChanged
End Sub
End Class

```

ROOM TYPE

```

Imports System.Data.OleDb
Public Class roomtype
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
    001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    Private Sub Label1_Click(sender As Object, e As EventArgs) Handles Label1.Click
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim a, b As String
        If RadioButton1.Checked = True Then
            'RadioButton2.Enabled = False
            a = "A/C"
        Else
            'RadioButton1.Enabled = False
            a = "Non A/C"
        End If
        If RadioButton3.Checked = True Then
            b = "Yes"
        Else
            b = "No"
        End If
        Dim obj As New clientbooking
        obj.str1 = a
        obj.str11 = b
        obj.TextBox1.Text = Label5.Text
        obj.TextBox5.Text = Label6.Text
    End Sub
End Class

```

```

        obj.Show()
        Me.Hide()
    End Sub
    Private Sub roomtype_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        con.Open()
        cmd.Connection = con
        cmd.CommandText = "select *from registerdb where FIRSTNAME='" & Label3.Text &
        "'and PASSWORD='" & Label4.Text & "'"
        Dim dr As OleDbDataReader = cmd.ExecuteReader
        While dr.Read
            Label5.Text = dr("CLIENTID")
            Label6.Text = dr("PHONENO")
        End While
        'cmd.ExecuteNonQuery()
        con.Close()
    End Sub
End Class

```

CLIENT BOOKING

```

Imports System.Data.OleDb
Public Class clientbooking
    Public Property str1 As String
    Public Property str11 As String
    Public Property str2 As String
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
    001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    'Dim tb As New DataTable("cbook")
    'Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=D:\cars\hotelmanagement\hotelmanagement\registerdb.accdb
    Dim datevalue1 As Date
    Dim datevalue2 As Date
    Dim datetime As TimeSpan
    Dim difference As Double
    Dim ds As New DataSet
    Private Sub TextBox3_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox3.TextChanged
    End Sub
    Private Sub clientbooking_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Dim str As String = "select *from cbook"
    Dim da As New OleDbDataAdapter(str, con)
        da.Fill(ds, "cbook")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "cbook"
        DataGridView1.Refresh()
        With DataGridView1.ColumnHeadersDefaultCellStyle
            .BackColor = Color.Navy
            ForeColor = Color.Black
        End With
    End Sub
End Class

```



```

        .Font = New Font("Lucida Calligraphy", 10, FontStyle.Bold)
        .WrapMode = DataGridViewTriState.False
    End With
    DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect
    DataGridView1.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    DataGridView1.MultiSelect = False
    With Me.DataGridView1
        .RowsDefaultCellStyle.BackColor = Color.Beige
        .AlternatingRowsDefaultCellStyle.BackColor = Color.Bisque
    End With
    If DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect Then
    End If
    TextBox9.Text = str1
    TextBox10.Text = str11
    TextBox3.Text = DataGridView1.RowCount
    TextBox4.Text = 300 - TextBox3.Text
    TextBox8.Text = TextBox3.Text + 1
End Sub
Private Sub TextBox6_TextChanged(sender As Object, e As EventArgs) Handles
TextBox6.TextChanged
End Sub
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
    datevalue1 = Me.DateTimePicker1.Value.ToShortDateString
    datevalue2 = Me.DateTimePicker2.Value.ToShortDateString
    datetime = (datevalue2 - datevalue1)
    difference = datetime.Days
    'Me.TextBox6.Text = ("" + Str(Int(difference / 365)) + "Year(s)")
    'difference = difference Mod 365
    'Me.TextBox6.Text = Me.TextBox5.Text + (Str(Int(difference / 30)) + "Months(s)")
    'difference = difference Mod 30
    Me.TextBox6.Text = (Str(difference)) + 1
    If TextBox9.Text = "A/C" And TextBox10.Text = "Yes" Then
        TextBox7.Text = 4000 * TextBox6.Text
    ElseIf TextBox9.Text = "Non A/C" And TextBox10.Text = "No" Then
        TextBox7.Text = 3000 * TextBox6.Text
    ElseIf TextBox9.Text = "A/C" And TextBox10.Text = "No" Then
        TextBox7.Text = 3500 * TextBox6.Text
    ElseIf TextBox9.Text = "Non A/C" And TextBox10.Text = "Yes" Then
        TextBox7.Text = 3500 * TextBox6.Text
    End If
    con.Open()
    cmd.Connection = con
    cmd.CommandText = "select FIRSTNAME from registerdb where CLIENTID=" +
TextBox1.Text + ""
    Dim dr As OleDbDataReader = cmd.ExecuteReader
    While dr.Read
        TextBox12.Text = dr("FIRSTNAME")
    End While
    'cmd.ExecuteNonQuery()

```

```

        con.Close()
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        con.Open()
        cmd.Connection = con
        cmd.CommandText = "insert into cbook values(" + TextBox1.Text + "," +
        TextBox12.Text + "," + TextBox8.Text + "," + TextBox5.Text + "," +
        TextBox6.Text + TextBox7.Text + "," + TextBox9.Text + "," + TextBox10.Text + "," +
        TextBox11.Text + ")"
        Dim str As String = "select *from cbook"
        Dim da As New OleDbDataAdapter(str, con)
        da.Fill(ds, "cbook")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "cbook"
        ds.Clear()
        DataGridView1.Refresh()
        MessageBox.Show("Reserved.....", "Hotel Fontana", MessageBoxButtons.OK)
        da.Fill(ds, "cbook")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember= "cbook"
        DataGridView1.Refresh()
        cmd.CommandText = "insert into canceltb values(" + TextBox1.Text + "," +
        TextBox12.Text + "," + TextBox9.Text + "," + TextBox10.Text + "," + TextBox6.Text +
        "," + TextBox7.Text + "," + TextBox11.Text + ")"
        cmd.ExecuteNonQuery()
        con.Close()
    End Sub
    Private Sub DataGridView1_CellContentClick(sender As Object, e As
    DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        TextBox1.Text = ""
        TextBox4.Text = ""
        TextBox5.Text = ""
        TextBox6.Text = ""
        TextBox7.Text = ""
    End Sub
    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        Me.Close()
    End Sub
End Class

```

CLIENT MENU

Imports System.Data.OleDb

Public Class clientmenu

```

Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")

```

```
Dim cmd As New OleDbCommand
```

```
Private Sub LinkLabel1_LinkClicked(sender As Object, e As  
LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
```

```
Dim ff As New roomtype
```

```
ff.Label3.Text = Label1.Text  
ff.Label4.Text = Label2.Text  
ff.Show()
```

```
End Sub
```

```
Private Sub clientmenu_Load(sender As Object, e As EventArgs) Handles MyBase.Load  
End Sub
```

```
Private Sub LinkLabel2_LinkClicked(sender As Object, e As  
LinkLabelLinkClickedEventArgs) Handles LinkLabel2.LinkClicked
```

```
con.Open()  
Dim cmd1 As OleDbCommand = New OleDbCommand("select *from[cbook]  
where[CLIENTID]='" & TextBox1.Text & "'and[cname]='" & Label1.Text & "'", con)
```

```
Dim dr1 As OleDbDataReader = cmd1.ExecuteReader
```

```
Dim userfound1 As Boolean = False
```

```
Dim lastname1 As String = ""
```

```
While dr1.Read
```

```
userfound1 = True
```

```
lastname1 = dr1("PAYMENT").ToString
```

```
End While
```

```
con.Close()
```

```
If lastname1 = "Not Paid" Then
```

```
Dim dd As New clientpayment
```

```
dd.Show()
```

```
Else
```

```
MessageBox.Show("You have already paid.....")
```

```
End If
```

```
End Sub
```

```
Private Sub LinkLabel3_LinkClicked(sender As Object, e As  
LinkLabelLinkClickedEventArgs) Handles LinkLabel3.LinkClicked
```

```
Dim cc As New cancelfrom
```

```
cc.Show()
```

```
End Sub
```

```
End Class
```

CLIENT PAYMENT

```
Imports System.Data.OleDb
```

```
Public Class clientpayment
```

```
Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data  
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-  
001\hotelmanagement\hotelmanagement\registerdb.accdb")
```

```
Dim cmd As New OleDbCommand
```

```
Private Sub clientpayment_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```

End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim a As String
    a = "Paid"
    Dim obj As New clientbooking
    obj.str2 = a
    con.Open()
    cmd.Connection = con
    cmd.CommandText = "select AMOUNT,PAYMENT from cbook where CLIENTID="
+ TextBox1.Text
    Dim dr As OleDbDataReader = cmd.ExecuteReader
    While dr.Read
        TextBox2.Text = dr("AMOUNT")
        TextBox3.Text = dr("PAYMENT")
    End While
    'cmd.ExecuteNonQuery()
    con.Close()
End Sub
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    con.Open()
    ' Dim cmd1 As OleDbCommand
    'cmd1.Connection = con
    cmd.Connection = con
    TextBox3.Text = ""
    TextBox3.Text = "Paid"
    cmd.CommandText = "(update cbook set PAYMENT=" + TextBox4.Text + " where
CLIENTID=" + TextBox1.Text + ")"
    'cmd.CommandText = "(update canceltb set PAYMENT=" + TextBox4.Text + " where
CLIENTID=" + TextBox1.Text + ")"
    MessageBox.Show("Transferred.....", "Hotel Fontana", MessageBoxButtons.OK)
    cmd.ExecuteNonQuery()
    con.Close()
End Sub
Private Sub TextBox3_TextChanged(sender As Object, e As EventArgs) Handles
TextBox3.TextChanged
End Sub
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    Me.Hide()
End Sub
End Class

```

MASTER MENU

```

Public Class mastermenu
Private Sub LinkLabel1_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
    masterrooms.Show()
End Sub
Private Sub LinkLabel2_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel2.LinkClicked

```

```

        mastermodel.Show()
End Sub
Private Sub LinkLabel3_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel3.LinkClicked
    mastrepayment.Show()
End Sub

Private Sub mastermenu_Load(sender As Object, e As EventArgs) Handles MyBase.Load
End Sub
End Class

```

MASTER MODEL

```
Imports System.Data.OleDb
```

```
Public Class mastermodel
```

```

    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    Dim ds As New DataSet
    Private Sub mastermodel_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim str As String = "select *from cbook"
        Dim da As New OleDbDataAdapter(str, con)
        da.Fill(ds, "cbook")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "cbook"
        DataGridView1.Refresh()
        With DataGridView1.ColumnHeadersDefaultCellStyle
            .BackColor = Color.Navy
            ForeColor = Color.Black
            .Font = New Font("Lucida Calligraphy", 10, FontStyle.Bold)
            .WrapMode = DataGridViewTriState.False
        End With
        DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect
        DataGridView1.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        DataGridView1.MultiSelect = False
        With Me.DataGridView1
            .RowsDefaultCellStyle.BackColor = Color.Beige
            .AlternatingRowsDefaultCellStyle.BackColor = Color.Bisque
        End With
        If DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect Then

            End If
        End Sub
    Private Sub Panel1_Paint(sender As Object, e As PaintEventArgs) Handles Panel1.Paint
    End Sub
    Private Sub Label3_Click(sender As Object, e As EventArgs) Handles Label3.Click
    End Sub

```

End Class

MASTER ROOMS

Imports System.Data.OleDb

Public Class masterrooms

 'Public Property s11 As String

 'Public Property ss1 As String

Private Sub RichTextBox31_TextChanged(sender As Object, e As EventArgs)

End Sub

Private Sub RichTextBox19_TextChanged(sender As Object, e As EventArgs)

End Sub

Private Sub RichTextBox18_TextChanged(sender As Object, e As EventArgs)

End Sub

Private Sub masterrooms_Load(sender As Object, e As EventArgs) Handles MyBase.Load

 'TextBox5.Text = s11

 'TextBox6.Text = 300 - s11

 Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")

 Dim cmd As New OleDbCommand

 con.Open()

 cmd.Connection = con

 cmd.CommandText = "select count(*) as countvalue from cbook where
ROOMTYPE='A/C' and EXTRSBED='Yes'"

 Dim dr As OleDbDataReader = cmd.ExecuteReader

 While dr.Read

 TextBox1.Text = dr("countvalue")

 'TextBox2.Text = dr("countvalue1")

 'TextBox3.Text = dr("countvalue2")

 'TextBox4.Text = dr("countvalue3")

 'TextBox2.Text = dr("PAYMENT")

End While

 dr.Close()

 cmd.CommandText = "select count(*) as countvalue1 from cbook where
ROOMTYPE='A/C' and EXTRSBED='NO'"

 Dim dr1 As OleDbDataReader = cmd.ExecuteReader

 While dr1.Read

 'TextBox1.Text = dr("countvalue")

 TextBox2.Text = dr1("countvalue1")

 'TextBox3.Text = dr("countvalue2")

 'TextBox4.Text = dr("countvalue3")

 'TextBox2.Text = dr("PAYMENT")

End While

 dr1.Close()

 cmd.CommandText = "select count(*) as countvalue2 from cbook where
ROOMTYPE='Non A/C' and EXTRSBED='Yes'"

 Dim dr2 As OleDbDataReader = cmd.ExecuteReader

 While dr2.Read

 'TextBox1.Text = dr("countvalue")

```

        'TextBox2.Text = dr("countvalue1")
        TextBox3.Text = dr2("countvalue2")
        'TextBox4.Text = dr("countvalue3")
        'TextBox2.Text = dr("PAYMENT")
    End While
    dr2.Close()
    cmd.CommandText = "select count(*) as countvalue3 from cbook where
ROOMTYPE='Non A/C' and EXTRSBED='No'"
    Dim dr3 As OleDbDataReader = cmd.ExecuteReader
    While dr3.Read
        'TextBox1.Text = dr("countvalue")
        'TextBox2.Text = dr("countvalue1")
        'TextBox3.Text = dr("countvalue2")
        TextBox4.Text = dr3("countvalue3")
        'TextBox2.Text = dr("PAYMENT")
    End While
    dr3.Close()
    cmd.CommandText = "select count(*) as countvalue4 from cbook"
    Dim dr4 As OleDbDataReader = cmd.ExecuteReader
    While dr4.Read
        'TextBox1.Text = dr("countvalue")
        'TextBox2.Text = dr("countvalue1")
        'TextBox3.Text = dr("countvalue2")
        TextBox5.Text = dr4("countvalue4")
        TextBox6.Text = 300 - TextBox5.Text
        'TextBox2.Text = dr("PAYMENT")
    End While
    dr4.Close()
    'cmd.ExecuteNonQuery()
    con.Close()
End Sub
Private Sub Panel1_Paint(sender As Object, e As PaintEventArgs) Handles Panel1.Paint
End Sub
Private Sub Label7_Click(sender As Object, e As EventArgs) Handles Label7.Click
End Sub
End Class

```

MASTER PAYMENYT

```

Imports System.Data.OleDb
Public Class mastrepayment
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    Dim ds As New DataSet
    Private Sub Label1_Click(sender As Object, e As EventArgs)
    End Sub
    Private Sub Label2_Click(sender As Object, e As EventArgs) Handles Label2.Click
    End Sub

```



```

Private Sub mastrepayment_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Dim str As String = "select *from cbook"
    Dim da As New OleDbDataAdapter(str, con)
    da.Fill(ds, "cbook")
    DataGridView1.DataSource = ds
    DataGridView1.DataMember = "cbook"
    DataGridView1.Refresh()
    con.Open()
    cmd.Connection = con
    With DataGridView1.ColumnHeadersDefaultCellStyle
        .BackColor = Color.Navy
        .ForeColor = Color.Black
        .Font = New Font("Lucida Calligraphy", 10, FontStyle.Bold)
        .WrapMode = DataGridViewTriState.False
    End With
    DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect
    DataGridView1.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
    DataGridView1.MultiSelect = False
    With Me.DataGridView1
        .RowsDefaultCellStyle.BackColor = Color.Beige
        .AlternatingRowsDefaultCellStyle.BackColor = Color.Bisque
    End With
    If DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect Then
        TextBox1.DataBindings.Add("text", ds, "cbook.CLIENTID")
        TextBox3.DataBindings.Add("text", ds, "cbook.CNAME")
        TextBox4.DataBindings.Add("text", ds, "cbook.ROOMRESERVED")
        TextBox5.DataBindings.Add("text", ds, "cbook.AMOUNT")
        TextBox6.DataBindings.Add("text", ds, "cbook.PAYMENT")
    End If
    'cmd.CommandText = "select count(*) as countvalue from cbook"
    'Dim dr As OleDbDataReader = cmd.ExecuteReader
    'While dr.Read
    '    TextBox1.Text = dr("countvalue")
    'TextBox2.Text = dr("countvalue1")
    'TextBox3.Text = dr("countvalue2")
    'TextBox4.Text = dr("countvalue3")
    'TextBox2.Text = dr("PAYMENT")
    'End While
    'dr.Close()
    cmd.CommandText = "select count(*) as countvalue1 from cbook where
PAYMENT='Paid'"
    Dim dr1 As OleDbDataReader = cmd.ExecuteReader
    While dr1.Read

        TextBox2.Text = dr1("countvalue1")
    End While
    dr1.Close()
    'cmd.CommandText = "select sum(AMOUNT) as countvalue2 from cbook"
    'Dim dr2 As OleDbDataReader = cmd.ExecuteReader

```



```

        'While dr2.Read
        '   TextBox3.Text = dr2("countvalue2")
        'End While
        'dr2.Close()
    End Sub
Private Sub DataGridView1_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick
    End Sub
End Class

```

CANCEL FORM

```

Imports System.Data.OleDb
Public Class cancelfrom
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\HP\Downloads\hotelmanagement-20210318T152611Z-
001\hotelmanagement\hotelmanagement\registerdb.accdb")
    Dim cmd As New OleDbCommand
    Dim ds As New DataSet
    Private Sub cancelfrom_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim str As String = "select *from canceltb"
        Dim da As New OleDbDataAdapter(str, con)
        da.Fill(ds, "canceltb")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "canceltb"
        DataGridView1.Refresh()
        With DataGridView1.ColumnHeadersDefaultCellStyle
            .BackColor = Color.Navy
            ForeColor = Color.Black
            .Font = New Font("Lucida Calligraphy", 10, FontStyle.Bold)
            .WrapMode = DataGridViewTriState.False
        End With
        DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect
        DataGridView1.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter
        DataGridView1.MultiSelect = False
        With Me.DataGridView1
            .RowsDefaultCellStyle.BackColor = Color.Beige
            .AlternatingRowsDefaultCellStyle.BackColor = Color.Bisque
        End With
        If DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect Then
            TextBox1.DataBindings.Add("text", ds, "canceltb.CLIENTID")
        End If
    End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    con.Open()
    cmd.Connection = con
    cmd.CommandText = "delete *from canceltb where CLIENTID=" + TextBox1.Text
    MessageBox.Show("Your Booking is Cancelled")
    cmd.ExecuteNonQuery()

```

```
ds.Clear()  
con.Close()  
Dim str As String = "select *from canceltb"  
Dim da As New OleDbDataAdapter(str, con)  
da.Fill(ds, "canceltb")  
DataGridView1.DataSource = ds  
DataGridView1.DataMember = "canceltb"  
DataGridView1.Refresh()  
End Sub  
End Class
```

FUTURE ENHANCEMENT

FUTURE ENHANCEMENT

- This will be extended for all other company products also.
- In future data will be stored in SQL Database.
- Users review will be collected.
- If any doubt about the websites or our about our hotel user may ask any questions and also they get the answers.
- Finger print access methods will be provided.

CONCLUSION

- This project can be extended for future updates.
- This project can be easily modified with different backend.
- This project has designed to be user-friendly and allow the user to access the services effectively and also easily.
- Reduce manual handling.

BIBLIOGRAPY

BIBLIOGRAPHY

REFERENCES

BOOKS

- **ELIAS M AWARD** 1996 ‘System analysis and design’ , **Galgotia Publication**, Second Edition
- **Jeff Prosise**
2001 ‘ProgrammingMicrosoft.NET’,Microsoft Press
,Second Edition
- **Thuan L.Thai,Hoang Lam** 2001’.NET Framework
Essentials’ ,O’ Reily Publications, Second Edition
- **Pressman S.Roger** 1990 ‘Software Engineering – A
Practitioners Approach ‘ McGraw -Hill book
Company,Fourth Edition

WEB URLS

1. <http://www.w3schools.com>
2. http://www.c_sharpcorner.com
3. www.dotnetspider.com
4. www.programmersheaven.com

CERTIFICATE
DEPARTMENT OF COMPUTER SCIENCE
ST.MARY'S COLLEGE (AUTONOMOUS), THOOTHUKUDI
Re-Accredited with 'A+' Grade by NAAC (4th Cycle)
Affiliated to
MANONMANIAM SUNDARANAR UNIVERSITY, TIRUNELVELI.

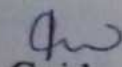


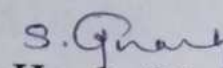
This is to certify that project entitled
COLLEGE ADMISSION MANAGEMENT

Submitted by

Divaincia. K	18AUCS05
Hemavalli. R	18AUCS09
Macrinal. J	18AUCS19
Muthu Mari. A	18AUCS30
Parameswari Bharathi. U	18AUCS34

In partial fulfillment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.


Guide


Head of the Department

Submitted for the Examination held on _____

Examiner



Principal

St. Mary's College (Autonomous)
Thoothukudi - 628 001.

COLLEGE ADMISSION MANAGEMENT

Project Submitted to the Department Of Computer Science

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, TIRUNELVELI.



In partial fulfilment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE done by

Divaincia. K	18AUCS05
Hemavalli. R	18AUCS09
Macrinal. J	18AUCS19
Muthu Mari. A	18AUCS30
Parameswari Bharathi. U	18AUCS34

Under the Supervision and Guidance of

Sr. K. Reetha M.Sc., M.Phil.

DEPARTMENT OF COMPUTER SCIENCE

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and blessings to finish our project successfully.

We are infinitely indebted to our college principal Dr. Sr. A. S. J. Lucia Rose M.SC, M.Phil., PGDCA, Ph.D., for providing us necessary infrastructure to undertake this project and work successfully.

We are so grateful to Mrs. S. Gnanathangam M.C.A, M.Phil (Associate professor & HOD), Department of Computer Science, for her gracious guidance. Our heartfelt thanks to our project guide Sr.K.Reetha M.Sc.,M.Phil.(Assistant professor) Department of Computer Science, for her valuable assistance and kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- Mrs.M. Jothimani B.E., M.Sc., M.Phil., (Associate professor)
 - Mrs.S. Andrews Felicita M.Sc.,M.B.A, (Assistant Professor),
 - Mrs.J. Anila Maily M.Sc , M.Phil. (Associate Professor),
- for their kind cooperation in successful completion of the project.

S.NO	CONTENTS	Page.No
1.	SYNOPSIS	5
2.	SYSTEM ANALYSIS	6
	❖ EXISTING SYSTEM	
	❖ PROPOSED SYSTEM	
	❖ ANALYZING THE SYSTEM	
3.	SYSTEM REQUIREMENTS & SPECIFICATION	9
	❖ WORKING ENVIRONMENT	
	❖ SYSTEM IMPLEMENTATION	
	❖ ABOUT SOFTWARE	
4.	SYSTEM DESIGN	17
	❖ TABLE DESIGN	22
	❖ DATA FLOW DIAGRAM	45
	❖ CODE DESIGN	49
	❖ FORM DESIGN	50
5	FUTURE ENHANCEMENT	51
	CONCLUSION	
	BIBLIOGRAPHY	

1.SYNOPSIS

The Project "COLLEGE ADMISSION MANAGEMENT" is designed with PHP as frontend and MySQL as backend.

- The aim of the project enables the admin to select/reject the users who apply for the seats.
- The admin maintains the details of the users who apply for the seats.
- The main advantage of this online college admission management is users can apply seats from any location
- The Online College Admission Management System can save our time.
- The Joy College of Engineering and Technology Online College Admission Management is designed to be user friendly.

2.SYSTEM ANALYSIS

The goal of the system development is to deliver the system in the line with user's requirements. Analysis is the heart of the process. It is the detailed study of the various operations performed by a system.

Existing System

The existing online college admission management websites are running successfully in the real world applications.

- The existing system doesn't provide users to view the status of their application.
- The System doesn't allow the user to raise queries or give feedback.

Proposed System

The development of the new system contains the following activities.

- This system allows the user to view the status of their application.
- This system is user friendly, easier and more flexible.
- This system provides more information about college.

Analyzing the Software

The system after careful analysis has been identified to be presented with the following systems:

The project is divided into two main systems:

- Admin Section
- User Section

These section are classified into the following subsystem:

- Administrator Module
- Home Module
- Contact Module
- About Module
- View status Module

Administrator Module:

- The Admin uses the user name and the password for login.
- The Admin can select or reject the application of the user.

Home Module:

- The online college management system the user can easily apply for the preferred course by just filling the details in the application. The user details are maintained confidentially only the admin has the access to view their details.

Contact Module:

- The User can (contact) raise their queries or give their feedback.

About Module:

- The Users are provided with more information about the management.

View Status Module:

- The Users are allowed to view the status of their application.

3. SYSTEM REQUIREMENT & ANALYSIS

Working Environment

Hardware Specification:

Hardware support the PC with below mentioned configuration

Processor : i3 processor

RAM : 5GB

Hard Disk : 1TB

Software Specification:

The software specification for the system are detailed as follows:

Front End : Html, CSS

Back End : Php, MySQL

Documentation : MS Word

System Implementation

This project implementation is aimed at ensuring that it works accurately and efficiently before live operation commences.

Implementation is vital to the success of any system. Implementation makes a logical assumption that all parts of the system are correct and the goal will be successfully achieved. The system is subjected to a variety of tests like response, security, recovery and usability before the system is ready for the user acceptance testing.

Validation:

The system has been tested and implemented successfully and thus ensured that all the requirements specification are completely fulfilled. In case of erroneous input corresponding error messages are displayed.

Testing:

This program was successfully loaded and executed. Because of good programming there were no execution errors. The successful output screens are placed in the output screen section.

Security:

The Admin can only view the details provided by the user. Admin can enter only using login form. So the username and password will verify authentication.

About Software

PHP:

PHP started out as a small open source project that evolved as more people found out how useful it was. Ramus Lerdorf unleashed the first version of PHP way back in 1994.

- PHP is a recursive acronym for PHP: “Hypertext Preprocessor”.
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, database, session tracking, and even built entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the UNIX side. The MySQL server, once started, once started, executes even very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP and LDAP. PHP4 added support for Java and object architecture (COM and CORBA), making n-tier development a possibility for the first time.
- PHP Syntax is C-LIKE.

Common uses of PHP:

- Using php you can add, delete, modify elements within your database thru PHP. Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- It can handle forms, i.e. gather data from files, save data to a file.

Characteristics of PHP:

- Familiarity.
- Simplicity.
- Efficiency.
- Security.
- Flexibility.
- Open source.
- Object Oriented.

Advantages of PHP:

- It runs on different platforms such as Windows, Linux, Unix, etc.
- This language is very simple to learn and runs efficiently on the server side.
- It is compatible with almost all servers used today, such as Apache, IIS, etc.
- It supports many databases such as MySQL, Oracle, PostgreSQL etc.
- It is often used together with Apache (web server) on various operating systems. It can be also used with Microsoft's IIS on Windows.
- It is open source and it is free downloadable.

MySQL: MySQL is a relational database management system based on SQL – Structured Query Language. It can run on various platforms UNIX, Linux, Windows, etc.

Features of MySQL

- Relational Database Management System (RDBMS),MySQL is a relational database management system.
- Easy to use. MySQL is easy to use.
- It is secure.
- Client/ Server Architecture.
- Free to download.
- It is scalable.
- Speed.
- High Flexibility.

4.SYSTEM DESIGN

Table Design:

Table:login

Column Name	Data Type
Email	varchar(30)
Password	varchar(30)

Table:response

Column Name	Data Type
Mail	varchar(30)
Subject	varchar(200)

Table:studentdetails

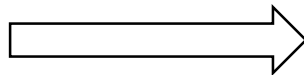
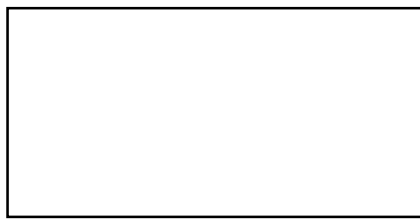
Column Name	Data Type
Name	varchar(30)
Mail	varchar(30)
father's name	varchar(30)

mobno	int(20)
Dob	int(11)
address	varchar(30)
gender	varchar(30)
sname	int(11)
10mark	int(11)
12mark	int(11)
cutoff	int(11)
department	varchar(30)
status	varchar(30)

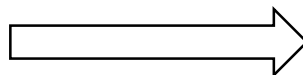
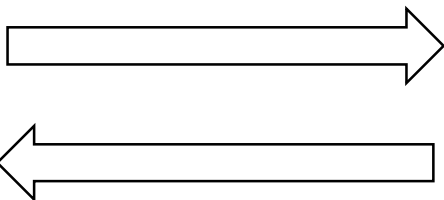
Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

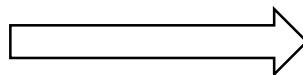
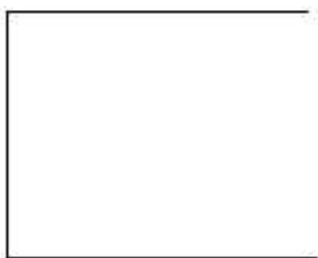
Data Flow Diagram Symbols:



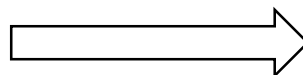
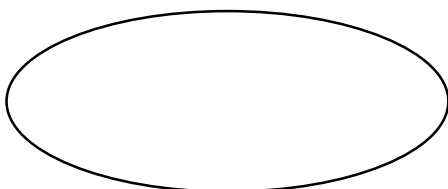
Source/Destination



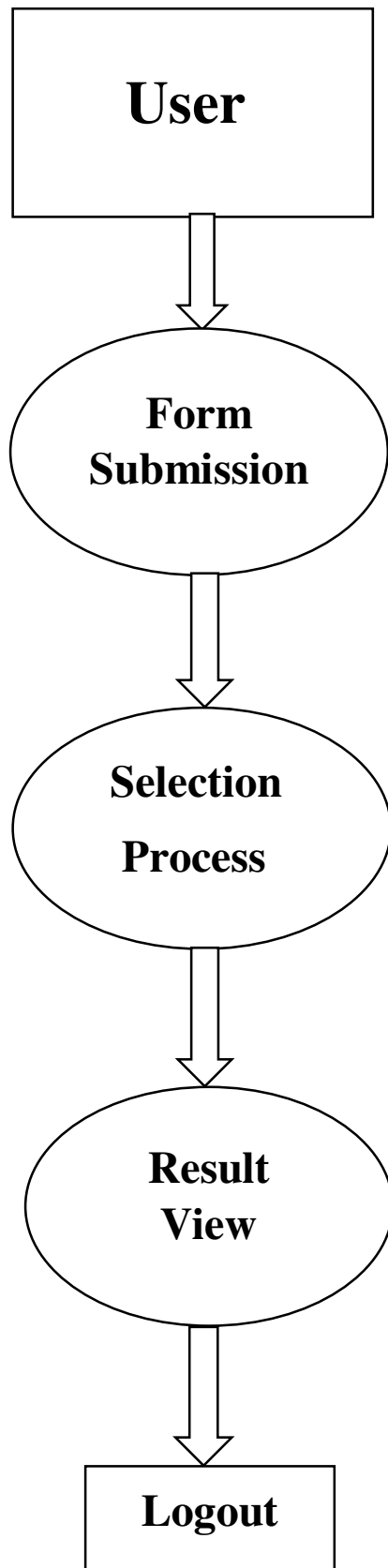
Data Flow



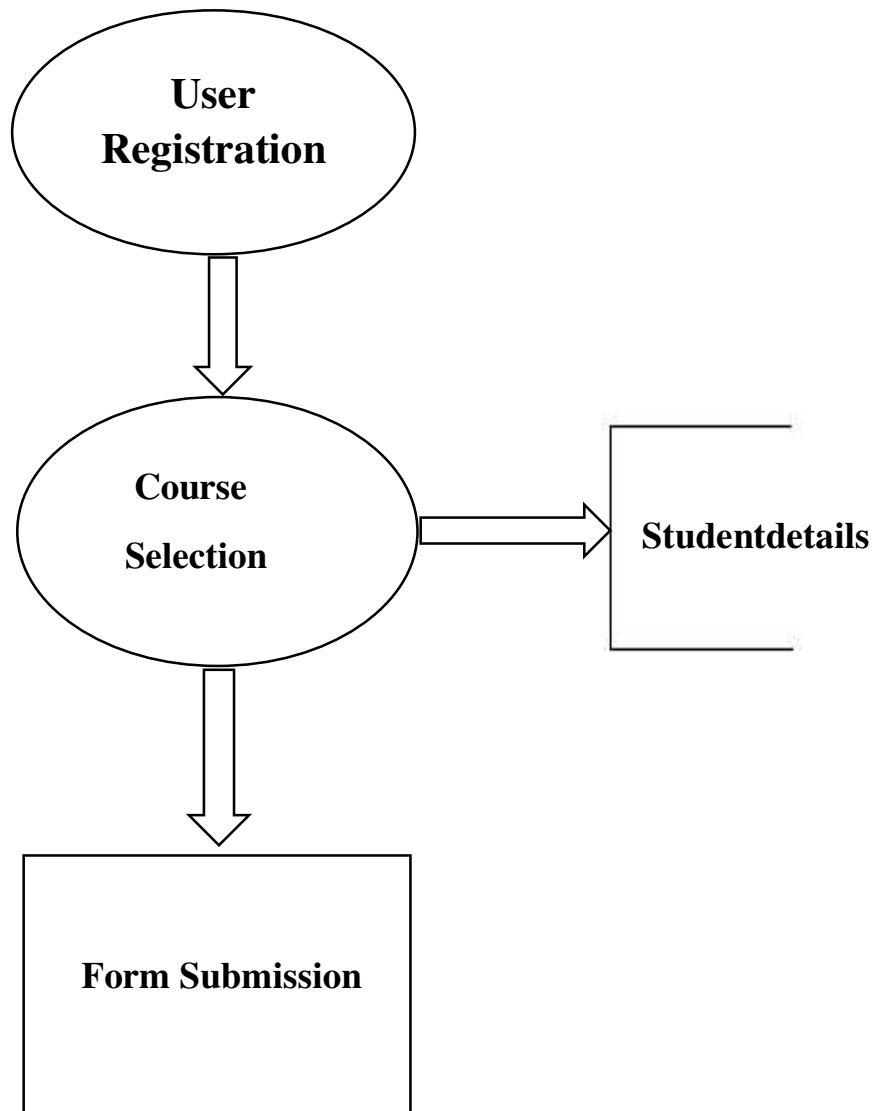
Database



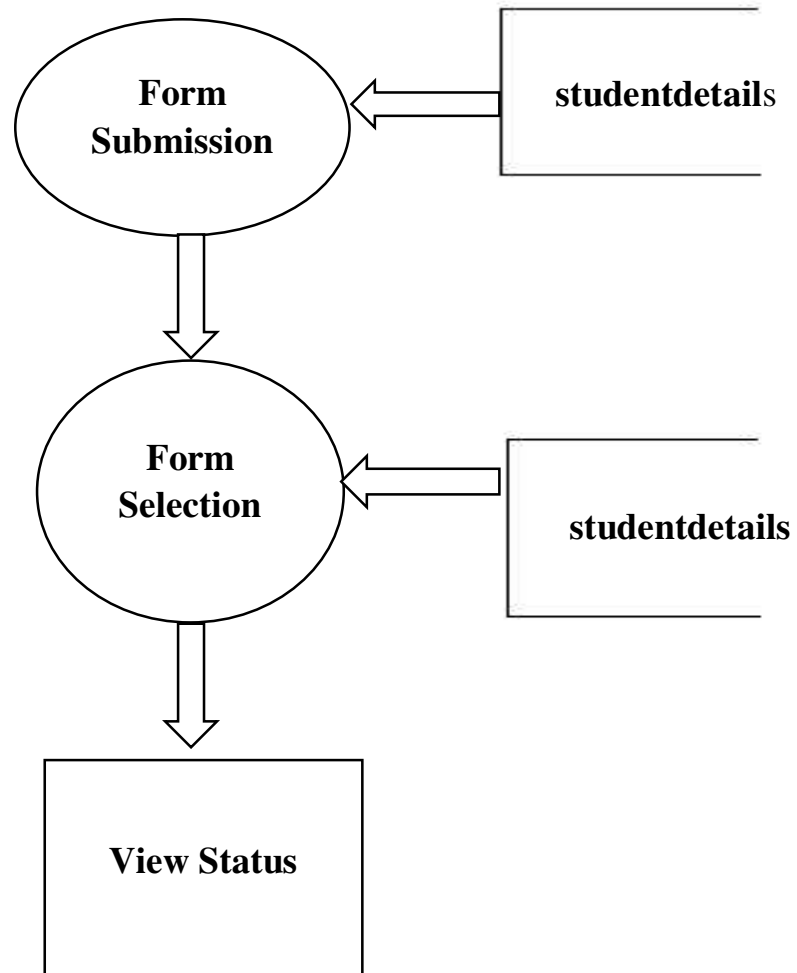
Process



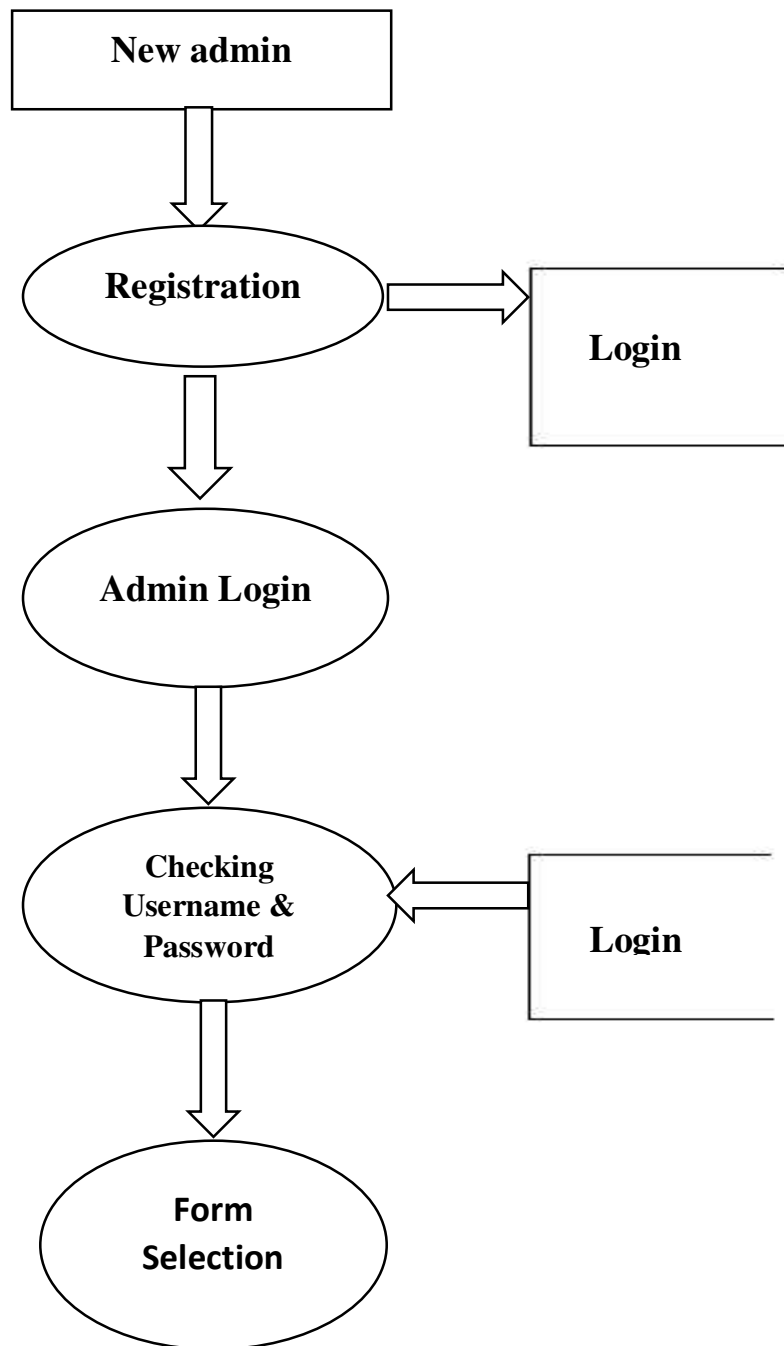
Level 1:



Level 2:



Admin Process:



Code Design

//login.php

```
<?php
$password="";
$pswrepeat="";
if($_SERVER["REQUEST_METHOD"]=="POST"){
    $password = $_POST['psw'];
    $email = $_POST['email'];
    //echo $password;
    //echo $email;
    $con=mysqli_connect("localhost","root","","clgmanagement");
    $query="select * from login";
    $res=mysqli_query($con,$query);
    if(mysqli_num_rows($res)>0){

while($row = mysqli_fetch_assoc($res))
    {
        if($email==$row['email'] && $password==$row['password']){

header("location:acpt.php");
        }
    else{
```

```
}  
    }  
}  
}?>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta name="viewport" content="width=device-width, initial-  
scale=1">
```

```
<style>
```

```
body, html {  
    height: 100%;  
    font-family: Arial, Helvetica, sans-serif;  
}
```

```
* {
```

```
    box-sizing: border-box;  
}
```

```
.bg-img {  
    /* The image used */
```

```
background-image: url("http://stmlportal.net/images/staff.png");
min-height: 400px;
/* Center and scale the image nicely */
background-position: center;
background-repeat: no-repeat;
position: relative;
}

ul {
list-style-type: none;
margin: 0;
padding: 0;
overflow: hidden;
background-color: #333;
}

li {
float: left;
}

li a {
display: block;
color: white;
text-align: center;
padding: 14px 16px;
```



```
    text-decoration: none;
}

li a:hover {
    background-color: #111;
}

/* Add styles to the form container */
.container {
    position: absolute;
    right: 0;
    margin: 20px;
    max-width: 300px;
    padding: 16px;
    background-color: white;
}

/* Full-width input fields */
input[type=text], input[type=password] {

width: 100%;
padding: 15px;
margin: 5px 0 22px 0;
border: none;
background: #f1f1f1;
}
```

```
input[type=text]:focus, input[type=password]:focus {  
    background-color: #ddd;  
    outline: none;  
}
```

```
/* Set a style for the submit button */
```

```
.btn {  
    background-color: #4CAF50;  
    color: white;  
    padding: 16px 20px;  
    border: none;  
    cursor: pointer;  
    width: 100%;  
    opacity: 0.9;  
}
```

```
.btn:hover {
```

```
    opacity: 1;  
}
```

```
</style>
```

```
</form>
```

```
</head>
```

```
<body>
<div class="bg-img">
<form action="<?php echo $_SERVER['PHP_SELF'];?>"
method="POST" class="container">
    <h1>Login</h1>
    <label for="email"><b>Email</b></label>
    <input type="text" placeholder="Enter Email" name="email"
required>
    <label for="psw"><b>Password</b></label>
    <input type="password" placeholder="Enter Password"
name="psw" required>
    <button type="submit" class="btn">Login</button>
    <h3>Do not have an account?</h3>
    <a href="signup.php">signup</a>
    <a href="firstpage.php">Back</a>
</form>
</div>
</body>
</html>
```

//admission.php

<?php

\$name="";

\$fname="";

\$mnumber="";

\$mail="";

\$birthday="";

\$address="";

\$gender="";

\$sname="";

\$ten="";

\$twelve="";

\$cutoff="";

\$dept="";

if(\$_SERVER["REQUEST_METHOD"]=="POST"){

 \$mail=\$_POST['mail'];

 \$name=\$_POST['name'];

 \$fname=\$_POST['Fname'];

 \$mnumber=\$_POST['mnumber'];

 \$birthday=\$_POST['birthday'];

 \$address=\$_POST['address'];

 \$gender=\$_POST['gender'];

```

$name=$_POST['sname'];
$ten=$_POST['ten'];
$twelve=$_POST['twelve'];
$cutoff=$_POST['cutoff'];
$cour=$_POST['course'];
$dept="";
$c="";
foreach ($cour as $c1) {
    $dept.=$c1.", ";
}
$status="waiting";
$con=mysqli_connect("localhost","root","","clgmanagement");
$query="select * from studentdetails";
$res=mysqli_query($con,$query);
if(mysqli_num_rows($res)>=0){
    $query="INSERT INTO
studentdetails(name,mail,fathersname,mobno,dob,address,gender,sname,10mark,12mark,cutoff,department,status)
values('$name','$mail','$fname','$mnumber','$birthday','$address','$gender','$sname','$ten','$twelve','$cutoff','$dept','$status')";
    $res=mysqli_query($con,$query);
} else {
}
}??>

```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta name="viewport" content="width=device-width, initial-  
scale=1">
```

```
<style>
```

```
body, html {
```

```
    height: 100%;
```

```
    font-family: Arial, Helvetica, sans-serif;
```

```
    background-color: coral;
```

```
}
```

```
* {
```

```
    box-sizing: border-box;
```

```
}
```

```
ul {
```

```
    list-style-type: none;
```

```
    margin: 0;
```

```
    padding: 0;
```

```
    overflow: hidden;
```

```
    background-color: #333;
```

```
}
```

```
li {  
    float: left;  
}
```

```
li a {  
    display: block;  
    color: white;  
    text-align: center;  
    padding: 14px 16px;  
    text-decoration: none;  
}
```

```
li a:hover {  
    background-color: #111;  
}
```

```
/* Add styles to the form container */  
.container {  
    position: absolute;  
    right:0;  
    margin: 20px;  
    max-width: 600px;
```

```
padding: 16px;
background-color: white;
}

/* Full-width input fields */
input[type=text], input[type=password] {
width: 100%;
padding: 15px;
margin: 5px 0 22px 0;
border: none;
background: #f1f1f1;
}

input[type=text]:focus, input[type=password]:focus {
background-color: #ddd;
outline: none;
}

/* Set a style for the submit button */
.btn {
background-color: #4CAF50;
color: white;
padding: 16px 20px;
border: none;
```



```

    cursor: pointer;
    width: 20%;
    opacity: 0.9;
}

.btn:hover {
    opacity: 1;
}
</style>
</head>
<body>
<ul>
    <li><a class="active" href="firstpage.php">Home</a></li>
    <li><a href="contact.php">Contact</a></li>
    <li><a href="about.php">About</a></li>
</ul>
<div class="bg-img">

<form action="<?php echo $_SERVER['PHP_SELF'];?>"
method="POST" class="container">
    <h1>Fill details</h1>
    <label for="email"><b>Name</b></label>
    <input type="text" placeholder="Enter Name" name="name"
required><br>

```

<label for="psw">Mail</label>

<input type="text" placeholder="Enter Mail" name="mail" required>

<label for="psw">Father's name</label>

<input type="text" placeholder="Enter father's name" name="Fname" required>

<label for="psw">Mobile number</label>

<input type="text" placeholder="Enter Number" name="mnumber" required>

<label for="psw">DOB:</label>

<input type="date" id="birthday" name="birthday">

<label for="psw">Address</label>

<textarea type="text" placeholder="Enter Address" name="address" required></textarea>

<label for="psw">Gender</label>

<input type="radio" id="male" name="gender" value="male">

<label for="male">Male</label>

<input type="radio" id="female" name="gender" value="female">

<label for="female">Female</label>

<input type="radio" id="other" name="gender" value="other">

<label for="other">Other</label>


```

<label for="name"><b> School Name</b></label>

  <input type="text" placeholder="Enter School Name"
name="sname" required>

  <label for="name"><b> Enter 10th mark</b></label><br>

  <input type="text" placeholder="Enter 10th mark" name="ten"
required>

  <label for="name"><b> Enter 12th mark</b></label><br>

  <input type="text" placeholder="Enter 12th mark" name="twelve"
required>

  <label for="name"><b> 12th cutoff</b></label><br>

  <input type="text" placeholder="Cutoff" name="cutoff" required>
<label for="course" ><b> Department</b></label><br>

  <input type="checkbox" id="course1" name="course[]" value="It">
<label for="vehicle1"> Information Technology</label><br>

  <input type="checkbox" id="course2" name="course[]"
value="mech">
<label for="vehicle2"> Mechanical</label><br>

  <input type="checkbox" id="course3" name="course[]"
value="Cs">
<label for="vehicle3"> Computer Science</label><br><br>
<button type="submit" class="btn">Submit</button>

  <a href="firstpage.php"> Back</a>

</form></div></body></html>

```

//acpt.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<style>

table {

margin: 0 auto;

font-size: large;

border: 1px solid black;

}

body {

background-image:

url('https://wallpaperaccess.com/full/1209458.jpg');

}

h1 {

text-align: center;

color: #006600;

font-size: xx-large;

font-family: 'Gill Sans', 'Gill Sans MT',

' Calibri', 'Trebuchet MS', 'sans-serif';

```
}
```

```
td {  
    background-color: #E4F5D4;  
    border: 1px solid black;  
}
```

```
th,
```

```
td {  
  
    font-weight: bold;  
    border: 1px solid black;  
    padding: 10px;  
    text-align: center;  
}
```

```
td {  
    font-weight: lighter;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Welcome</h1>
```

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>"
method="POST" class="container">
```

```
<br><br>
```

```
</form>
```

```
<section>
```

```
<h1>Admission forms</h1>
```

```
<h3><a href="response.php">Feedback</a></h3>
```

```
</div>
```

```
<!-- TABLE CONSTRUCTION-->
```

```
<table>
```

```
<tr>
```

```
<th>Name</th>
```

```
<th>mail</th>
```

```
<th>father name</th>
```

```
<th>Mobile number</th>
```

```
<th>DOB</th>
```

```
<th>Address</th>
```

```
<th>Gender</th>
```

```
<th>School Name</th>
```

```
<th>Dept</th>
```

```
<th>10th mark</th>
```

```
<th>12th mark</th>
```

```

        <th>cutoff</th>

    </tr>

    <!-- PHP CODE TO FETCH DATA FROM ROWS-->

    <?php // LOOP TILL END OF DATA
    $con=mysqli_connect("localhost","root","","clgmanagement");

    $query="select * from studentdetails WHERE
status='waiting'";

    $res=mysqli_query($con,$query);
    while($rows = mysqli_fetch_assoc($res)){
    $dept=$rows['department'];

    echo "<tr>";
    echo "<td>".$rows['name']."</td>";
    echo "<td>".$rows['mail']."</td>";

    echo "<td>".$rows['fathersname']."</td>";
    echo "<td>".$rows['mobno']."</td>";
    echo "<td>".$rows['dob']."</td>";
    echo "<td>".$rows['address']."</td>";
    echo "<td>".$rows['gender']."</td>";
    echo "<td>".$rows['sname']."</td>";
    echo "<td>".$rows['department']."</td>";
    echo "<td>".$rows['10mark']."</td>";
    echo "<td>".$rows['12mark']."</td>";

```

```

        echo "<td>".$rows['cutoff']."</td>";

        echo '<td><form action="req.php" method="POST">

        <input type="hidden" name="email"
value='.$rows['mail'].'><input type="submit"
value="Accept"></form></td>';

        echo '<td><form action="req1.php" method="POST">

        <input type="hidden" name="email"
value='.$rows['mail'].'><input type="submit"
value="Decline"></form></td>';

        echo "</tr>";

    }

    ?>

</table>

<form action='login.php'><input type='submit'
value='Back'></form>

</section>

</body>

</html>

```


//response.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<style>

table {

margin: 0 auto;

font-size: large;

border: 1px solid black;

}

body {

background-image:

url('https://wallpaperaccess.com/full/1209458.jpg');

}

h1 {

text-align: center;

color: #006600;

font-size: xx-large;

font-family: 'Gill Sans', 'Gill Sans MT',

' Calibri', 'Trebuchet MS', 'sans-serif';

}

td {

```
background-color: #E4F5D4;
border: 1px solid black;
}
```

th,

```
td {
    font-weight: bold;
    border: 1px solid black;
    padding: 10px;
    text-align: center;
}
```

```
td {
    font-weight: lighter;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>"
method="POST" class="container">
```

```
<br><br>
```

```
</form>
```

```
<section>
```

```
<!-- TABLE CONSTRUCTION-->
```

```

<table> <h1>FEEDBACK</h1>

  <tr>

    <th>Mail</th>

    <th>SUbject</th>

  </tr>

  <!-- PHP CODE TO FETCH DATA FROM ROWS-->

  <?php  // LOOP TILL END OF DATA
$con=mysqli_connect("localhost","root","","clgmanagement");
    $query="select * from response";
    $res=mysqli_query($con,$query);

while($rows = mysqli_fetch_assoc($res)){
    echo "<tr>";
    echo "<td>".$rows['Mail']."</td>";
    echo "<td>".$rows['Subject']."</td>";
    echo "</tr>";
}
?>

</table>

  <form action="acpt.php"><input type='submit'
value='Back'></form>

</section>

</body>

</html>

```

Form Design

Home Page:



Admin Login:



Login

Email

Password

Do not have an account?

[signup](#) [Back](#)

Welcome

Admission forms

[Feedback](#)

Name	mail	father name	Mobile number	Birth year	Address	Gender	School Name	Dept	10th mark	12th mark	cutoff		
muthu mari	Kalamuthuraj@gmail.com	ayyappan	2147483647	2021	6,uiui	female	Holy Cross girl hr.sec school	It.mech.	496	1123	89	<input type="button" value="Accept"/>	<input type="button" value="Decline"/>

[Back](#)

Application Form:

[Home](#) [Contact](#) [About](#) [View Status](#)

Fill details

Name
M. Kala

Mail
Kalamuthuraj@gmail.com

Father's name
Muthu Raj

Mobile number
9098755674

DOB: 22-10-2001

Address 7, Martin Colony,
Tirunelveli

Gender

School Name
St Thomas Matriculation school

Enter 10th mark
496

Enter 12th mark
1156

12th cutoff
88

Department
Engineering Programs:-
☐ Civil Engineering
☐ Mechanical Engineering
☒ Computer Science Engineering
☐ Electrical and Electronics Engineering
☒ Electrical and Communication Engineering
B.Tech Programs:-
☒ Aeronautical Engineering
☐ Biotechnology Engineering
☐ Information Technology

[Submit](#) [Back](#)

About Us

Joy College of Engineering and Technology is an institution of higher learning dedicated to global enrichment and meeting the evolving needs of our diverse communities. With an emphasis on innovative programs, dynamic curriculum and general education skills, we are committed to being a pioneer in the field of career-focused education. We empower our students, faculty and staff to exceed the expectations of society through academic excellence, community enrichment and service to the public good.

Our Team



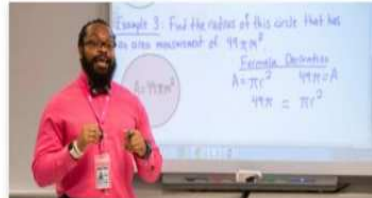
J Janet

Chairman & Founder



John Doe

Principal



Mike Ross

Chief of staff

Dr. Mike Ross is the Chief of Staff in Joy College of Engineering and technology, an enthusiastic person who always maintain the unity among the staffs and the students.

JoyCollege.com

Responsive Contact Section

Joy College Of Engineering And Technology

Phone Number: 7485327865



Email ID

Subject

Write something..

Submit

Back

5.FUTURE ENHANCEMENT

Even though the project has been completed there are several plans for enhancement of the project in future.

SMS and E-mail facility for the registered users to receive notification about their application status.

6. CONCLUSION

This project online college admission management which minimizes the time required for the manual admission process system. Records of the students can be produced easily in a short span of time. Workflow of the process will lead to efficiency and will be user friendly.

7.BIBLIOGRAPHY

Reference:

- <https://www.javatpoint.com/php-tutorial>
- <https://www.nopaperforms.com/>
- <https://www.w3schools.com/php/>
- https://www.tutorialspoint.com/php/php_web_concepts.html
- <https://www.guru99.com/what-is-php-first-php-program.html>
- PHP and MySQL by Mike McGrath
- PHP: The Complete Reference by Steven Holzner

LEAVE MANAGEMENT SYSTEM

Project Submitted to the Department of Computer Science

ST.MARY'S COLLEGE (AUTONOMOUS), THOOTHUKUDI

Affiliated to

MONONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.

In partial fulfilment of the requirement for the award of the degree in

BACHELOR OF COMPUTER SCIENCE

Done by

M.Durga Devi	18AUCS06
B.Muneeswari	18AUCS28
M.Muthumeenachi	18AUCS31
P.Preethi	18AUCS38
E.Revathi	18AUCS40

Under the Supervision and guidance of

Mrs.M.Jothimani B.E.,M.Sc(IT),M.Phil.

(Associate Professor of Computer Science)



DEPARTMENT OF COMPUTER SCIENCE

ST.MARY'S COLLEGE (AUTONOMOUS)

(Re-accredited with 'A+' Grade by NAAC)

THOOTHUKUDI – 628001

BATCH 2018-2021

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is certify that the project entitled

LEAVE MANAGEMENT SYSTEM

Submitted by

M.Durga Devi

18AUCS06

B.Muneeswari

18AUCS28

M.Muthumeenachi

18AUCS31

P.Preethi

18AUCS38

E.Revathi

18AUCS40

In partial fulfilment of the requirement for the award of degree in **BACHELOR OF COMPUTER SCIENCE** during the year 2020-2021.

Guide

Head of the Department

Submitted for the Examination held on 15.04.2021

Examiner

Lucia Rose
Principal

St. Mary's College (Autonomous)
Thoothukudi - 628 001.



Acknowledgement

First and foremost we thank the Almighty God for his gracious help and shower of blessings for having rendered us the strength to finish our project successfully.

We are infinitely indebted to our college Principal Dr. Sr. A. S. J. Lucia Rose M.Sc., M.Phil., PGDCA, Ph.D. for providing us necessary infrastructure to undertake this project and work successfully.

We are ever so grateful to Mrs. S. Gnanathangam M.C.A., M.Phil., (Associate Professor & H.O.D) of the Computer Science Department for her prompt notification and guidance. Our heartfelt thanks to our project guide Mrs. M. Jothimani B.E., M.Sc(IT)., M.Phil.,(Associate Professor) for the valuable assistance and the kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- Sr. K. Reeta M.Sc., M.Phil.,(Assistant Professor)
- Mrs. J. Anila Maily M.Sc., M.Phil.,(Associate Professor)
- Mrs. S. Andrews Felcita M.Sc., M.B.A.,(Assistant Professor)

for their kind cooperation and support in successful completion of the project.

We would like to thank our lab assistant Mrs. J. Cladina, for her kind patient assistance in lab during our project hours.

We thank our family members especially our parents for their encouragement and support both morally and financially which helped us to finish the project successfully.

Finally we express our special thanks to our friends who gave us a wonder encouragement ideas to complete the project in much desired.

S.NO	CONTENT	PAGE NO.
1.	SYNOPSIS	7
2.	SYSTEM ANALYSIS 2.1 EXISTING SYSTEM 2.2 PROPOSED SYSTEM 2.3 ANALYSING THE SYSTEM	9 10 11 11
3.	SYSTEM REQUIREMENTS AND SPECIFICATIONS 3.1 WORKING ENVIRONMENT 3.2 SYSTEM IMPLEMENTATION 3.3ABOUT SOFTWARE	15 16 17 18
4.	SYSTEM DESIGN 4.1 TABLE DESIGN 4.2 DATA FLOW DIAGRAM 4.3 FORM DESIGN 4.4 SOURCE CODE	22 23 26 32 38
5.	FUTURE ENHANCEMENT	98
6.	CONCLUSION	99
7.	BIBILOGRAPHY	101

1

SYNOPSIS

1.SYNOPSIS

The Project “**LEAVE MANAGEMENT SYSTEM**” is designed with HTML and CSS as front end, PHP and MySQL as backend.

- The aim of the project enables the admin to maintain a record of leaves available.
- User friendliness is provided in the application with various controls.
- The system makes the overall project management much easier and flexible.
- The admin can approve or deny the leave permissions.
- The students can place their leaves with proper reasons.
- This Leave Management System is designed to be more user friendly.
- This will save the time of the staff and students.





2. SYSTEM ANALYSIS



2.1 EXISTING SYSTEM



2.2 PROPOSED SYSTEM



2.3 ANALYSING THE SYSTEM



2. SYSTEM ANALYSIS

After analysing the requirements of the task to be performed, the next step is to analyse the problem and understand its context. The first activity in the phase is studying the existing system and other is to understand the requirements and domain of the new system. Both the activities are equally important, but the first activity serves as a basis of giving the functional specifications and then successful design of the proposed system. Understanding the properties and requirements of a new system is more difficult and requires creative thinking and understanding of existing running system is also difficult, improper understanding of present system can lead diversion from solution.

2.1 Existing System

The existing leave management system websites is running successfully in the real world application.

- The existing system doesn't provide access to different languages.
- The system doesn't provide Loyalty points.
- Doesn't give user more flexibility to update their profile.

2.2 Proposed System

The development of the new system contains the following activities.

- User friendliness is provided in the application with various controls.
- The system makes the overall project management much easier and flexible.
- The admin can change the language of the system if needed.
- The system provides high level of security with better performance as compared to existing one.
- User registration form and their details will be secured.
- Admin can view each and every user's details individually.

2.3 Analysing the Software

The system after careful analysis has been identified to be presented with the following systems:

The project is divided into 2 main systems:

- ❖ Admin Section
- ❖ User Section

The 11 types of modules are:

- ✓ Login
- ✓ Home page
- ✓ Students Login

- ✓ Leave form
- ✓ Teacher's Sign up
- ✓ Teacher's Login
- ✓ Accepted List
- ✓ Accept / Decline
- ✓ Report
- ✓ Delete
- ✓ View Status

Administrator:

- The Admin uses the user name and password for log in.
- The Admin can access all other Modules of the system.

Login Module:

The login page to verify the users username and password.

System Module:

The module is the whole description of every particular option available in the leave management system.

Home Page:

The home page contains three modules: the student, the staff incharge and the HOD.

Students Login:

This verify the students username and password. The students who are not registered by the admin cannot login to this page.

Leave Forms Module:

You can apply the leaves by entering from date and to date in the given space or you can select the dates by clicking the calendar. Then the user need to give the valid reasons for their leaves and then click the submit button.

Teacher's Sign Up:

This allows the user to enter their mail id and password to be registered in the admin.

Teacher's Login:

This module verifies the teacher's mail id and the password that has been already signed up.

Accepted List:

In this the user will be able to see only the accepted list of the students leave.

Accept / Decline:

In the admin side that is the HOD page, the admin will receive the list of leaves applied. The Admin either accept or decline the request of the students.

Submit:

It is a button which you have to click after selecting the options from the given list. It will take you to the next page of the application and you can look for the details of your leave on the next page.

Report:

In this module the user will get the complete leave reports of the particular student.

Delete Module:

In this module the admin can delete a particular student from the list that appears.

Teacher Page Module:

If the admin accepted the request of the user then the teachers will receive the list of leaves that are being accepted. If the admin declines the request of the user that messages will not be present in the teacher's module.

Back Module:

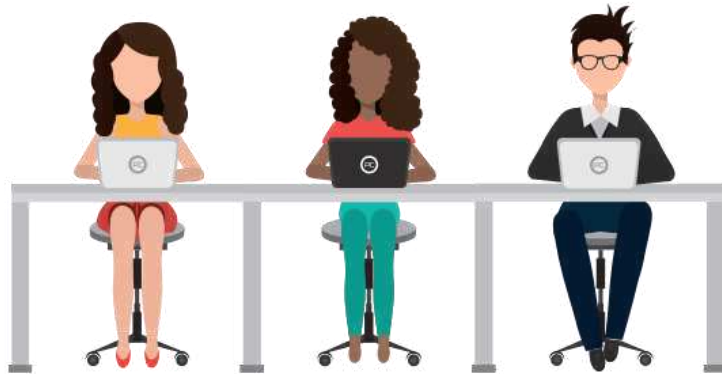
The Back module will be used to go back to the main page of the system.

View Status:

In this module the students can view whether their leave has been approved or denied by the HOD. They will get both the approved and denied list.



3. SYSTEM REQUIREMENTS AND SPECIFICATION



3.1 WORKING ENVIRONMENT (Hardware & software specification, server & client side)



3.2 SYSTEM APPLICATION (Validation & Security)



3.3 ABOUT THE SOFTWARE (PHP, Characteristics, Common users & Advantages + MySQL, Features of MySQL)

3. SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 Working Environment:

Hardware Specification:

Hardware support with below mentioned configurations.

Server side:

- Hard Disk : 1TB
- RAM : 8GB
- Processor : i3

Client side:

- Hard Disk : 500GB
- RAM : 2GB
- Processor : i3
- Protocol : HTTP

Software Specification:

The software specification for the system are detailed as follows

- Front-End Tools : PHP
- Back-End Tools: MYSQL Database
- Documentation: MS Word.

3.2System Implementation:

This Project implementation is aimed at ensuring that it works accurately and efficiently before live operation commences.

Implementation is vital to the success of any system. Implementation makes a logical assumption that all parts of the system are correct and the goal will be successfully achieved .The system is subjects to a variety of test like response, security, recovery and usability before the system is ready for the user acceptance testing.

Validation:

In the validation testing, System is a analyzed by giving various input values. While passing through different values such as zero, large values, negative values etc.,We can determine whether the system works under all circumstances.

Security:

Anyone who wants to use the system can enter only by the login form. Further, navigation is controlled by the password scheme. So the username and password will verify authentication.

3.3 About the Software:

PHP:

PHP started out as a small open source project that evolved as more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

- PHP is a recursive acronym for: PHP: "Hypertext Processor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, database, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular database, including MySQL, PostgreSQL, Oracle, Sybase, Informix and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the UNIX side. The MySQL server, once started, execute even very complex queries with huge result sets in record-setting time.
- PHP support a large number of major protocols such as POP3, IMAP and LDAP. PHP4 added support for java and object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP Syntax is C-LIKE.

Common uses of PHP:

- PHP perform system function, i.e. from files on a system it can create, open read, write and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete and modify elements within your database through PHP.
- Access cookies variable and set cookies.
- Using PHP, you can restrict users to access some pages of your websites.

Characteristics of PHP:

Five important characteristics make PHP's practical nature possible...

- Simplicity
- Efficiency
- Security
- Flexibility
- Familiarity

Advantages of using PHP:

- PHP runs on various platforms(Windows, Linux, Unix, MacOS X,etc).
- PHP is compatible with almost all servers used today(Apache, IIS,etc.).
- PHP support a wide range of database. It is free.
- PHP is easy to learn and runs efficiently on the server side.

MYSQL:

MYSQL is a fast, easy to use relational database. It is currently the most popular open-source database. It is very commonly used in conjunction with PHP script to create powerful and dynamic server-side application. It is developed, marketed and supported by MYSQL AB, a Swedish company. It is written in C and C++.

Features of MYSQL:

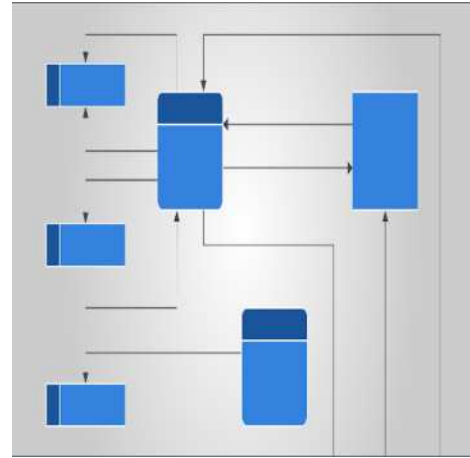
- Relational Database Management System (RDBMS): MYSQL is a relational database management system.
- Easy to Use: MYSQL is easy to use. You have to get only the basic knowledge of SQL. You can build and interact with MYSQL with only a few simple SQL statements.
- It is Secure: MYSQL consists of a solid data security layer that protect sensitive data from intruders. Password are encrypted in MYSQL query data, save change.
- Client Server Architecture: MYSQL follows a client/server architecture. There is a database server (MYSQL) and arbitrarily many client (application program) , which communicate with the server.
- It is Scalable: MYSQL can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4GB. However you can increase this number to a theoretical limit of 8 TB of data.
- Allows roll-back: MYSQL allows transactions to be rolled back, commit and crash recovery.

- High Performance: MYSQL is a faster, more reliable and cheaper because of its unique storage engine architecture.
- High Flexibility: MYSQL support a large number of embedded application which makes MYSQL very flexible.
- Free to download: MYSQL is free to use and you can download it from MYSQL official websites.

4. SYSTEM DESIGN



4.1 TABLE DESIGN



4.2 DATA FLOW DIAGRAM



4.3 FORM DESIGN



4.4 SOURCE CODE

4. SYSTEM DESIGN

4.1 Table Design:

Table: hodlogin

Column Name	Data Type
Email	varchar(30)
Password	varchar(30)

Table: leavedetails

Column Name	Data Type
Name	varchar(30)
Email	varchar(30)
Fdate	date
Tdate	date
Reason	varchar(30)
Status	varchar(30)
RegisterNumber	varchar(30)
batch	varchar(30)

Table: login

Column Name	Data Type
Email	varchar(200)
Password	varchar(30)

Table: teacherlogin

Column Name	Data Type
Email	varchar(30)
Password	varchar(30)

Table: loginform

Column Name	Data Type
ID	Int(11)
User	Varchar(50)
Pass	Varchar(50)

Table:viewstatus

Column Name	Data Type
Name	varchar(30)
Email	varchar(30)
Fdate	date
Tdate	date
Reason	varchar(30)
Status	varchar(30)
RegisterNumber	varchar(30)
batch	varchar(30)

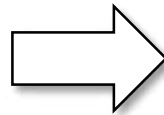
Table:hodacpt

Column Name	Data Type
Name	varchar(30)
Email	varchar(30)
Fdate	date
Tdate	date
Reason	varchar(30)
Accept	button
Decline	button
RegisterNumber	varchar(30)
batch	varchar(30)

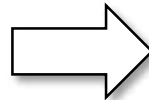
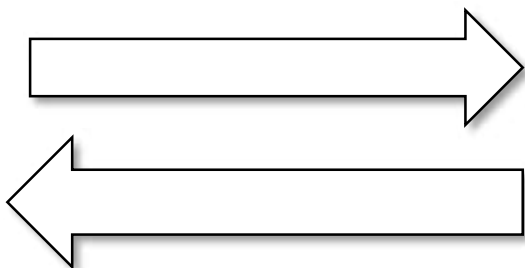
4.2 DATA FLOW DIAGRAM:

Data flow is the movement of origin to a specific destination indicated by lines and arrows. Data flow diagram is the graphical representation of a data movement, processes and files used in the support of an information system.

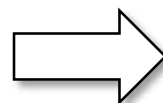
Data Flow Diagram Symbols



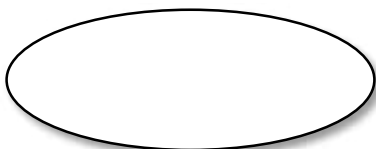
Source/Destination



Data Flow

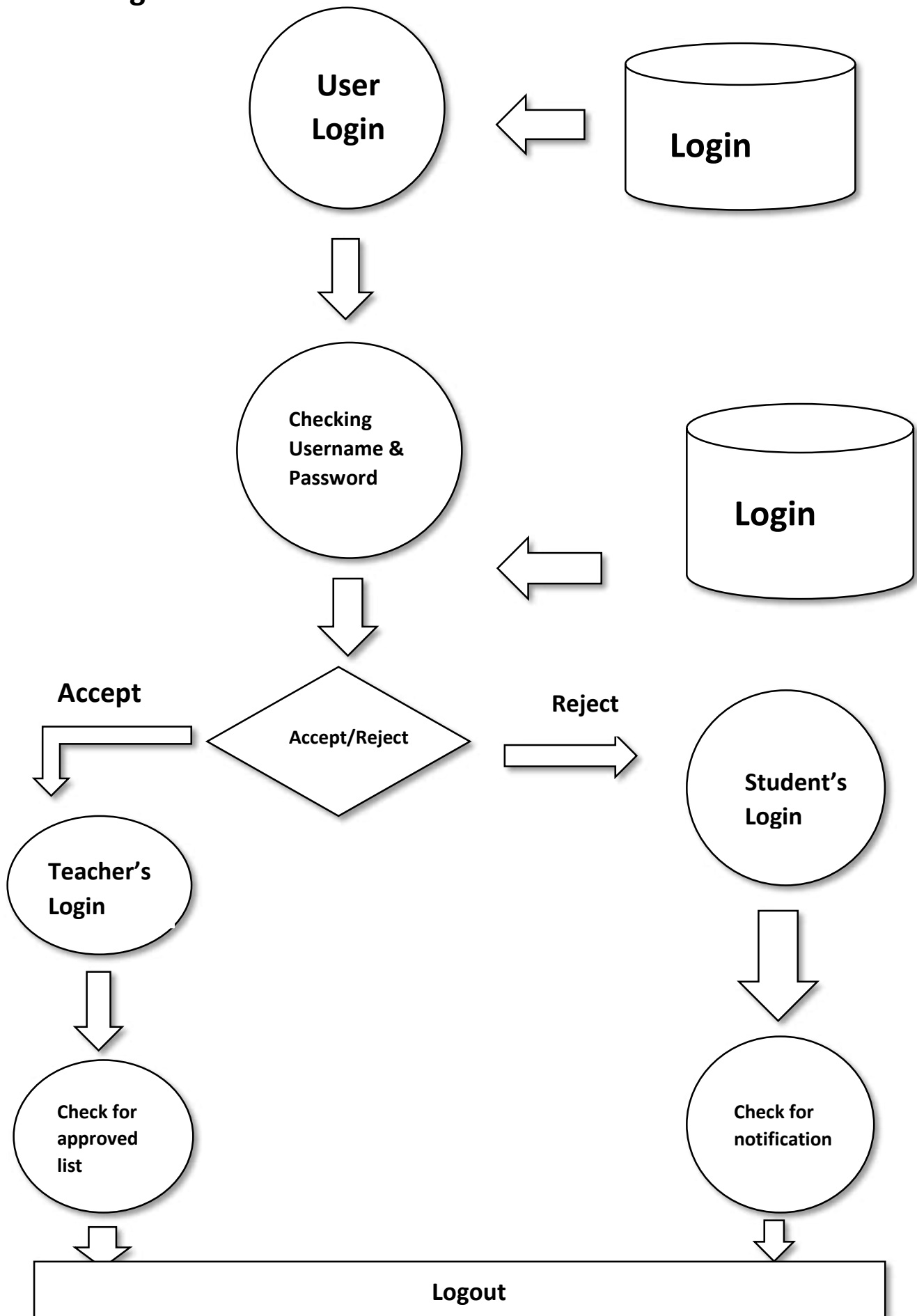


Database

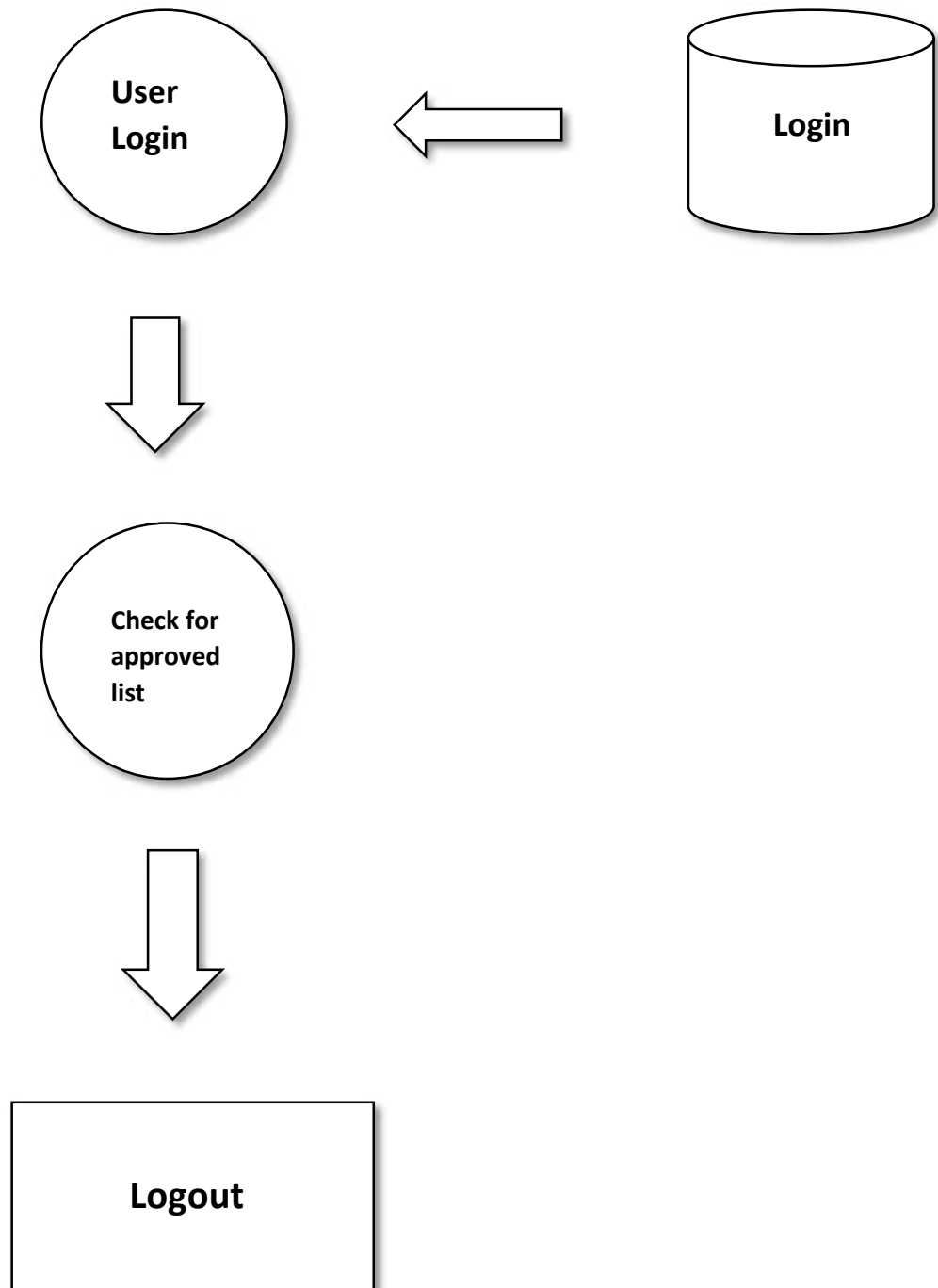


Process

HOD Login:

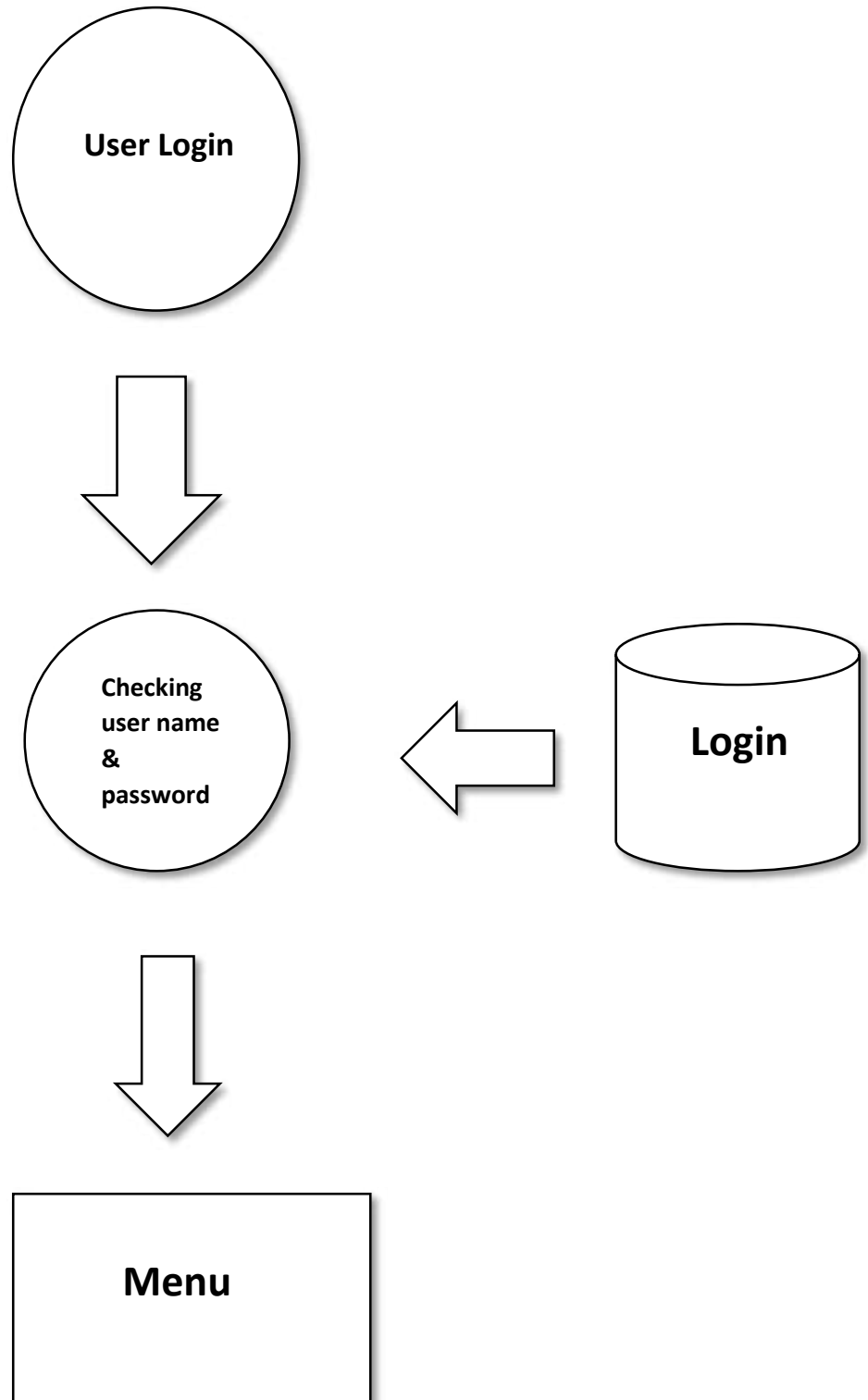


Teacher's Login:

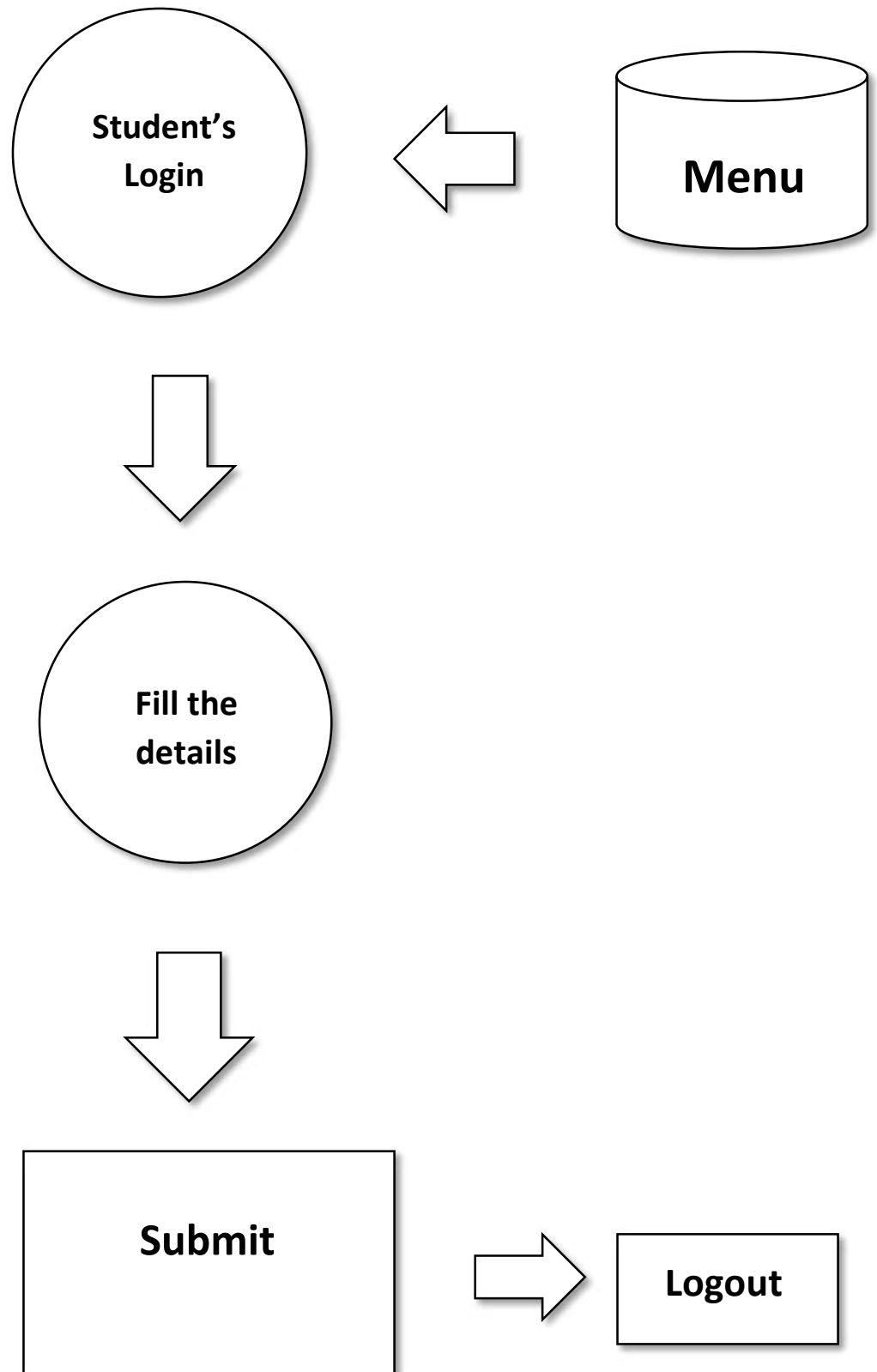


Student's Login:

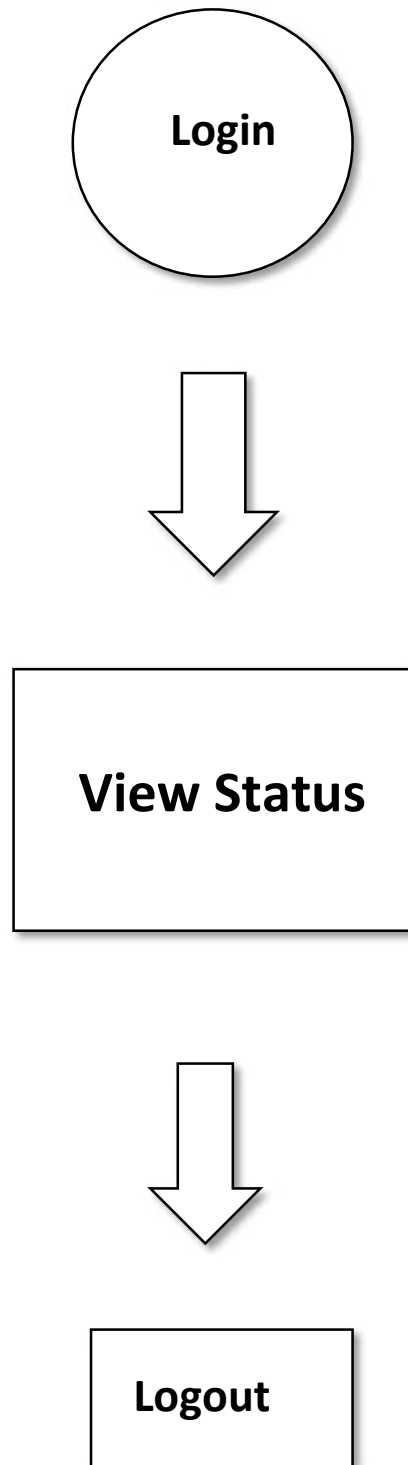
Level 1:



Level 2:

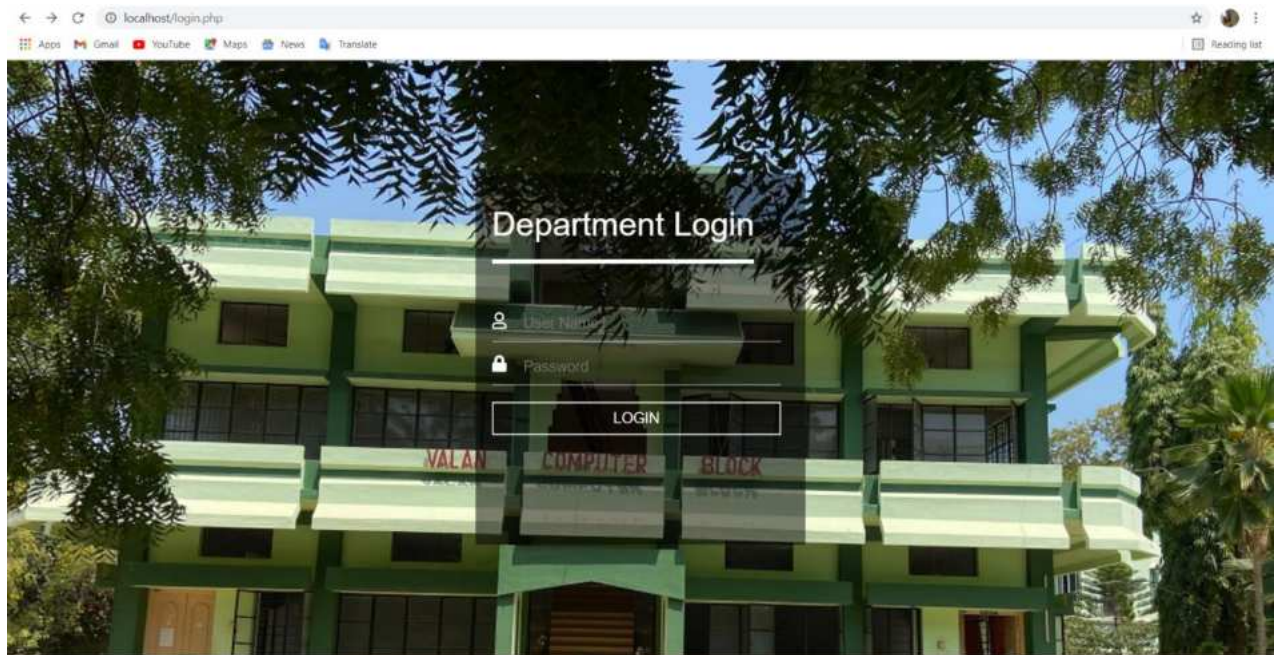


Level 3:

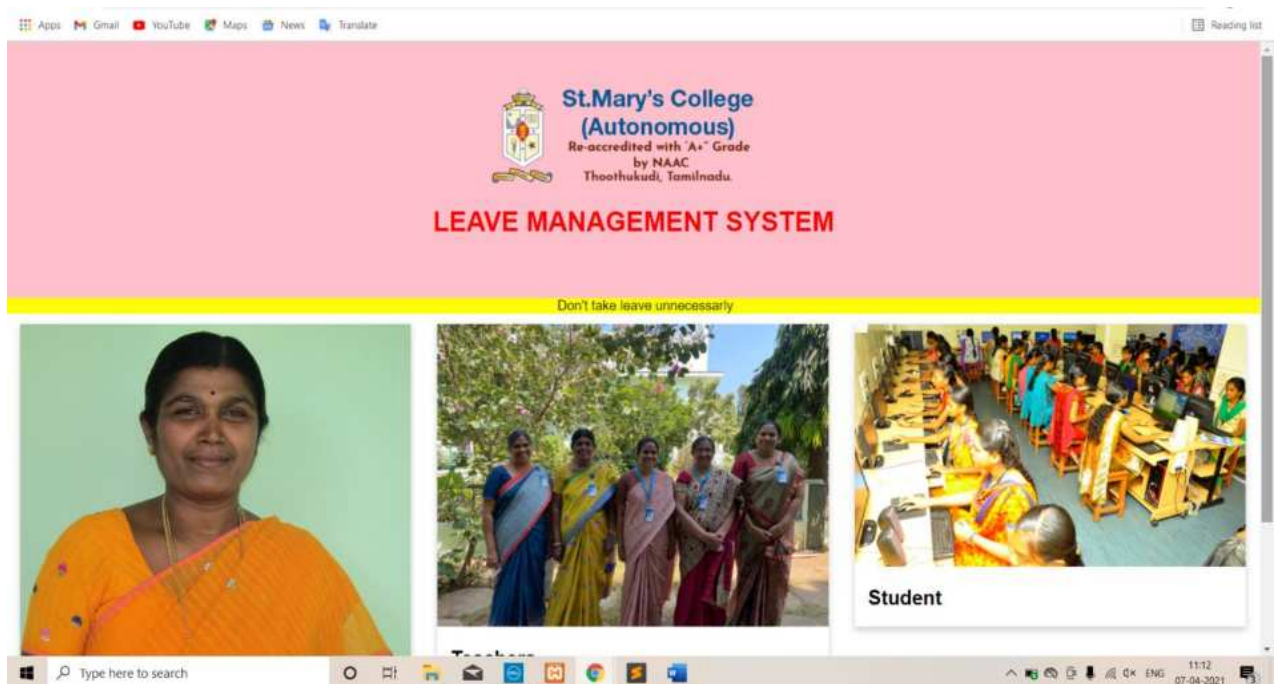


4.3 Screenshots:

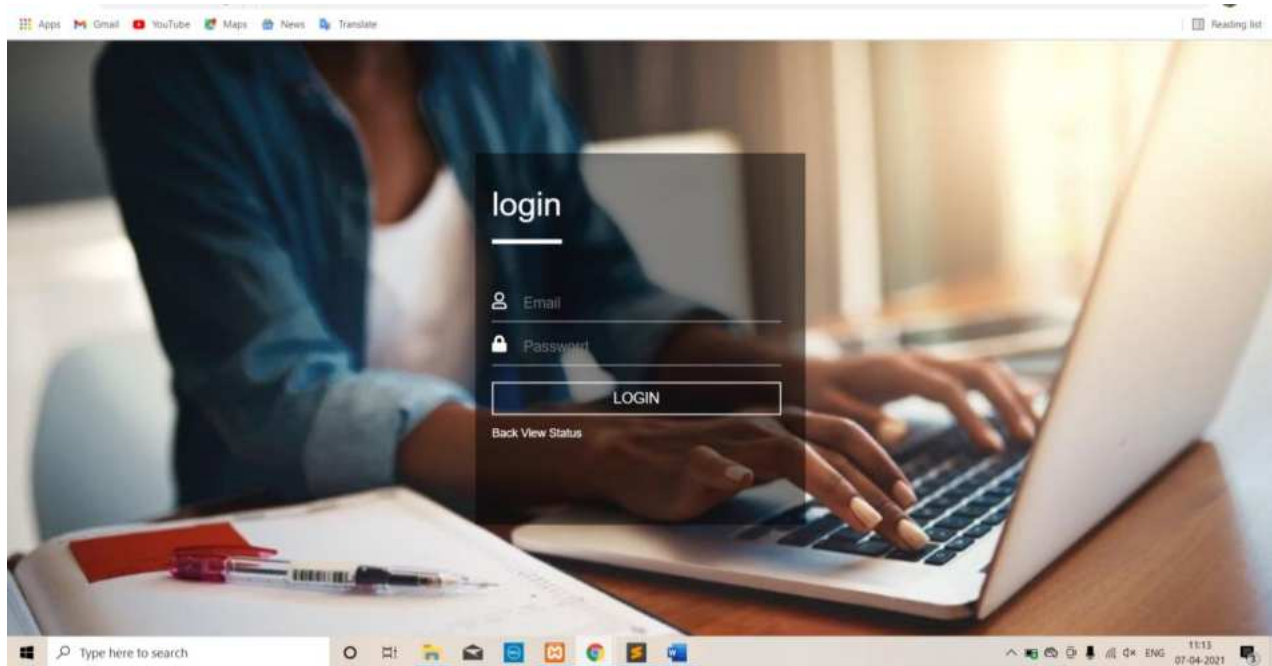
Login Page:



Home Page:



Student's Login:



The screenshot shows a web browser window with a login modal. The modal has a title "login" and a "Back View Status" link. It contains two input fields: "Email" and "Password", followed by a "LOGIN" button. The background image shows a person's hands typing on a laptop keyboard.

login

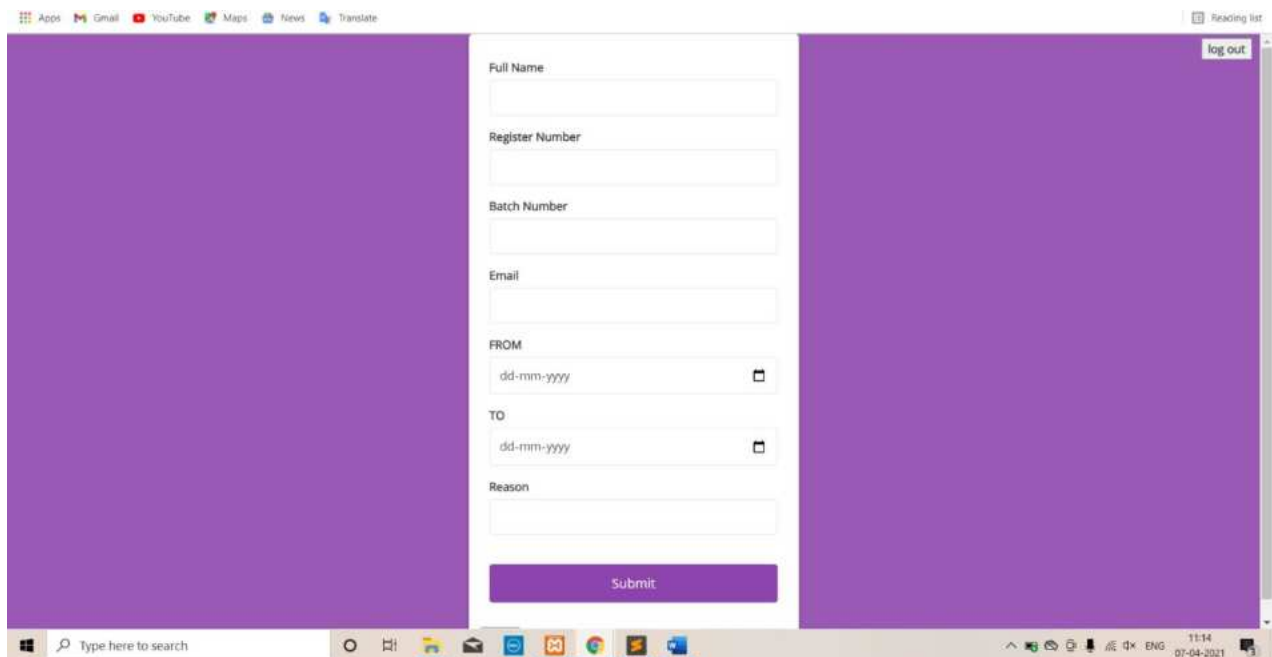
[Back View Status](#)

Email

Password

LOGIN

Leave Form:



The screenshot shows a web browser window with a leave form. The form has a "log out" button in the top right corner. It contains several input fields: "Full Name", "Register Number", "Batch Number", "Email", "FROM" (with a date picker), "TO" (with a date picker), and "Reason". A "Submit" button is at the bottom.

log out

Full Name

Register Number

Batch Number

Email

FROM

dd-mm-yyyy

TO

dd-mm-yyyy

Reason

Submit

HOD Login:

Apps Gmail YouTube Maps News Translate Reading list


Welcome HOD

Username :
Enter email

Password :
Enter Password

Log in

☒ Remember me



Windows taskbar: Type here to search, Taskbar icons, System tray: 11:19, 07-04-2021

Apps Gmail YouTube Maps News Translate Reading list

Welcome

Leave forms

Name	Email	from	To	reason	Register Number	Batch		
Revathi	revathi@gmail.com	2021-04-07	2021-04-08	fever	18AUCS40	2018-2021	<input type="button" value="Accept"/>	<input type="button" value="Decline"/>
Preethi	preethi@gmail.com	2021-04-08	2021-04-09	head ache	18AUCS38	2018-2021	<input type="button" value="Accept"/>	<input type="button" value="Decline"/>
Meenachi	meenachi@gmail.com	2021-04-12	2021-04-13	stomach ache	18AUCS31	2018-2021	<input type="button" value="Accept"/>	<input type="button" value="Decline"/>

Back
Student details
Remove Student

Windows taskbar: Type here to search, Taskbar icons, System tray: 11:19, 07-04-2021

Student's Report:

Apps Gmail YouTube Maps News Translate Reading list

REPORT

Email

SUBMIT

Type here to search

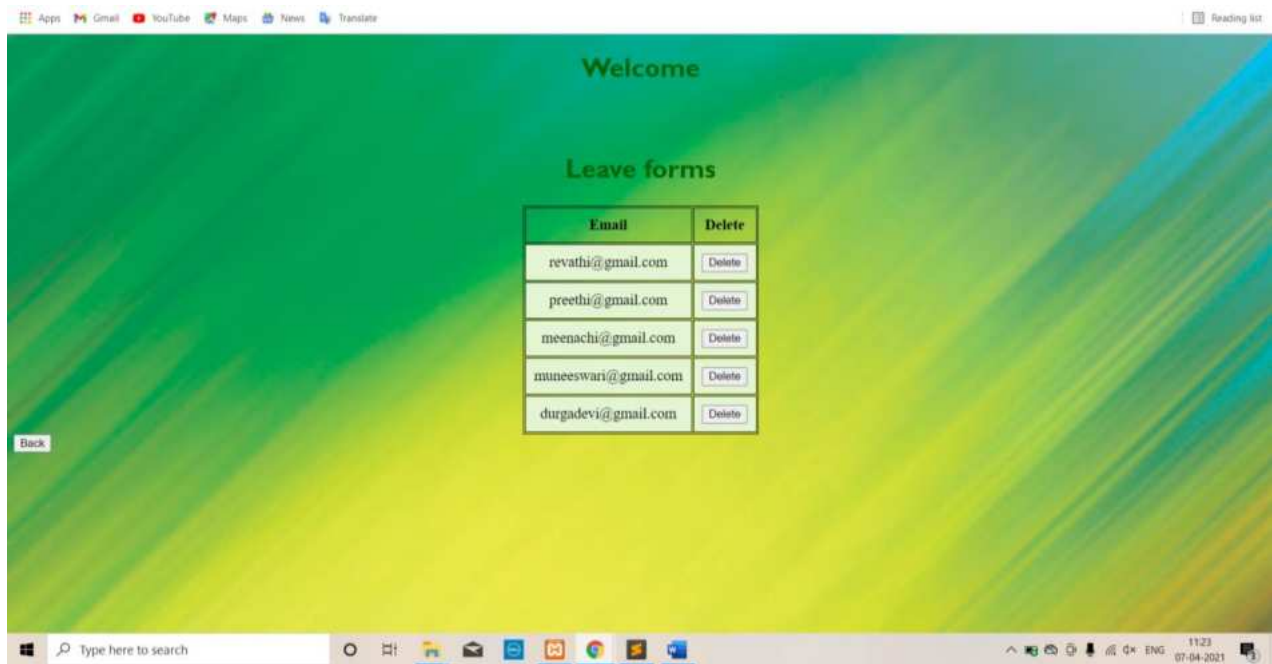
11:21 07-04-2021

Apps Gmail YouTube Maps News Translate Reading list

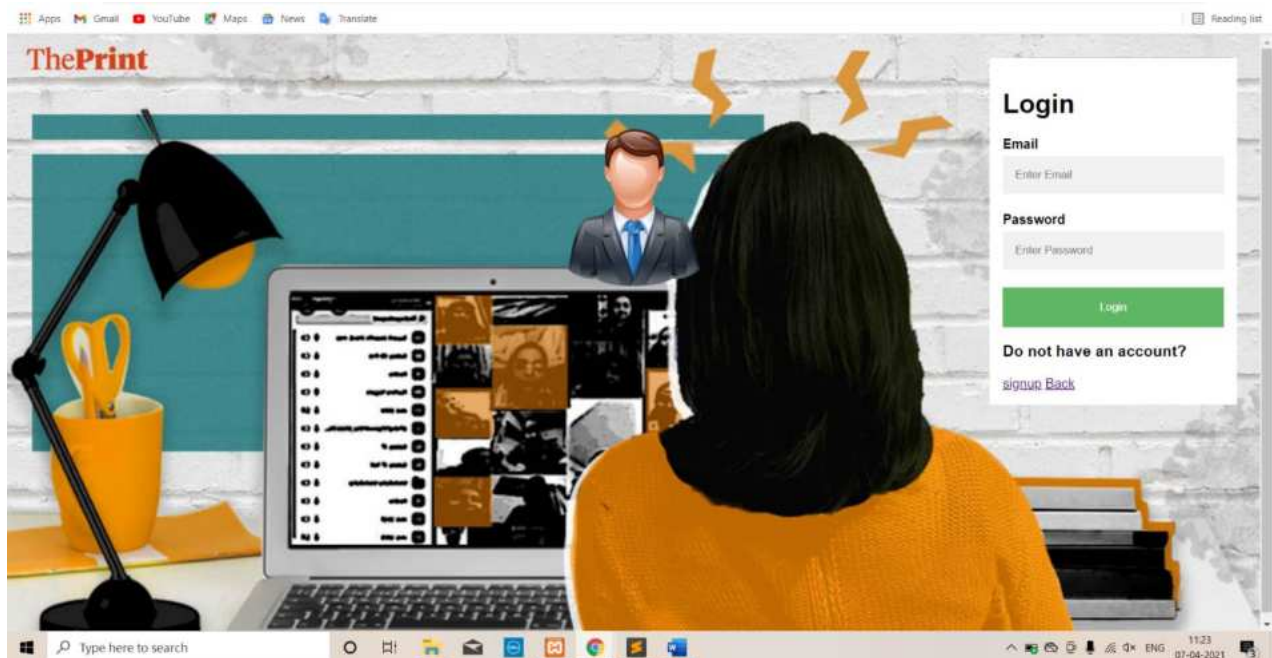
Name	Email	From date	To date	Reason
Preethi	preethi@gmail.com	2021-04-08	2021-04-09	head ache
Preethi	preethi@gmail.com	2021-04-15	2021-04-16	fever

Type here to search

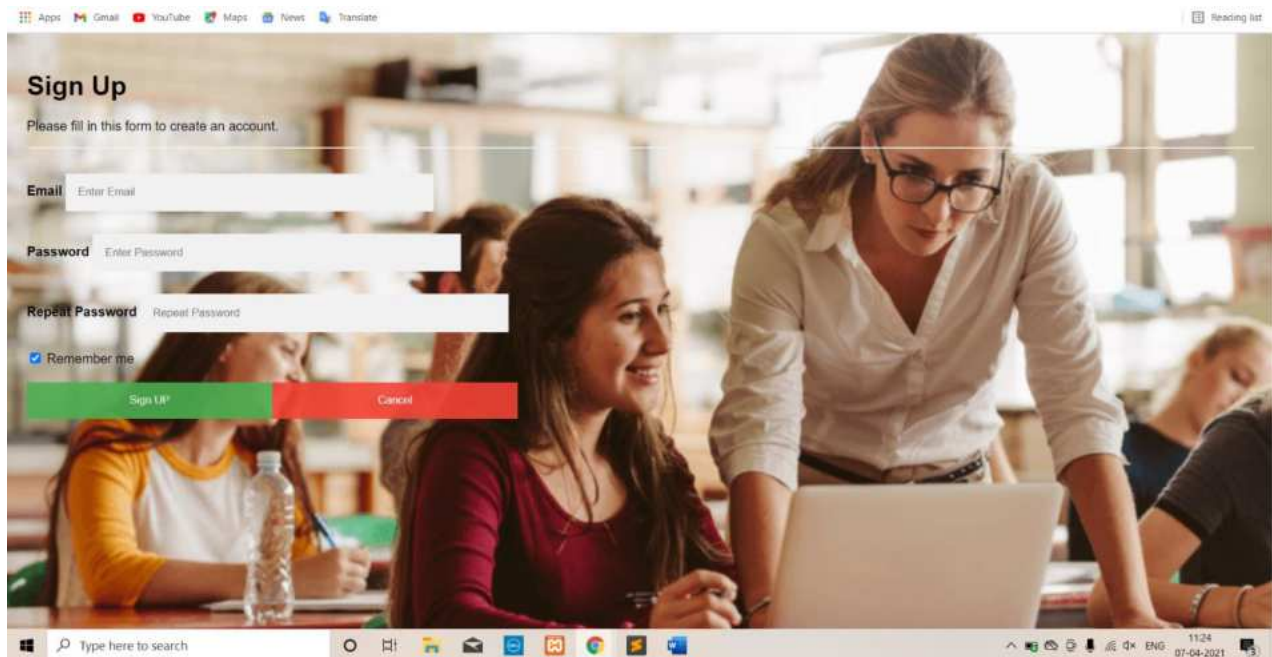
11:28 07-04-2021



Teacher's Login:



Teacher's Sign Up:



Apps Gmail YouTube Maps News Translate Reading list

Sign Up

Please fill in this form to create an account.

Email

Password

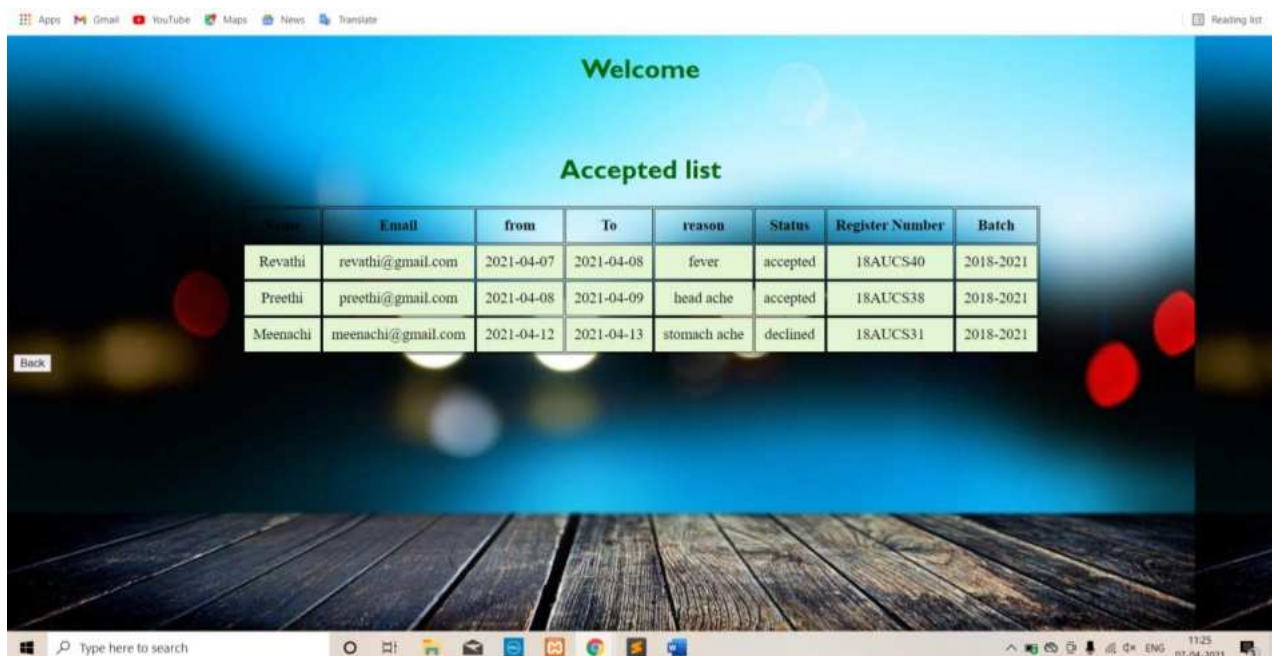
Repeat Password

☒ Remember me

Type here to search

11:24 07-04-2021

Accepted List:



Apps Gmail YouTube Maps News Translate Reading list

Welcome

Accepted list

Name	Email	from	To	reason	Status	Register Number	Batch
Revathi	revathi@gmail.com	2021-04-07	2021-04-08	fever	accepted	18AUCS40	2018-2021
Preethi	preethi@gmail.com	2021-04-08	2021-04-09	head ache	accepted	18AUCS38	2018-2021
Meenachi	meenachi@gmail.com	2021-04-12	2021-04-13	stomach ache	declined	18AUCS31	2018-2021

[Back](#)

Type here to search

11:25 07-04-2021

4.4 Source Code:

acptmail.php

```
<?php
$mail = $_POST['mail'];
//echo $password;
//echo $email;
$con=mysqli_connect("localhost","root","","leavemanagement");
$query="Update leavedetails SET status='accepted' where mail='".$mail."'";
if(mysqli_query($con,$query))
{
    $body="This is your OTP for Covid E-pass Login.\n\r Thank You
";
    if(mail($mail,$body,"From:arunalakshmi609@gmail.com"))
    {
        echo $mail;
        echo '<script>alert("OTP Unsuccessfull")</script>';
    }
    else
    {
        echo '<script>alert("OTP Unsuccessfull")</script>';
    }
    header('refresh:0.1,url=hodacpt.php');
}
else{
    echo '<script>alert("OTP Unsuccessfull")</script>';
}
?>
```

alert.php

```
<!DOCTYPE html>
<html>
<head>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

body {

  font-family: Arial, Helvetica, sans-serif;

  margin: 0;

}

html {

  box-sizing: border-box;

}

*, *:before, *:after {

  box-sizing: inherit;

}

.column {

  float: left;

  width: 33.3%;

  margin-bottom: 16px;

  padding: 0 8px;

}

.card {

  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);

  margin: 8px;

}

.about-section {

  padding: 50px;

  text-align: center;

  background-color: #474e5d;

  color: white;

}

.container {

  padding: 0 16px;
```



```
}  
  
.container::after, .row::after {  
  content: "";  
  clear: both;  
  display: table;  
}  
  
.title {  
  color: grey;  
}  
  
.button {  
  border: none;  
  outline: 0;  
  display: inline-block;  
  padding: 8px;  
  color: white;  
  background-color: #000;  
  text-align: center;  
  cursor: pointer;  
  width: 100%;  
}  
  
.button:hover {  
  background-color: #555;  
}  
  
@media screen and (max-width: 650px) {  
  .column {  
    width: 100%;  
    display: block;  
  }  
}  
  
</style>
```

```
</head>
```

```
<body>
```

```
<div class="about-section">
```

```
  <h1>Ur response is not submitted because ur leave limit has been excluded.Please contact your  
  HOD as a person and ask a permission for leave</h1>
```

```
</div>
```

form1.php

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>FORM</title>
```

```
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-  
    awesome/4.7.0/css/font-awesome.min.css">
```

```
    <style>
```

```
      @import url('https://fonts.googleapis.com/css?family=Roboto');
```

```
    *{
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      box-sizing: border-box;
```

```
      outline: none;
```

```
      font-family: 'Roboto', sans-serif;
```

```
    }
```

```
    body{
```

```
      background: url('baige.jpeg') no-repeat top center;
```

```
      background-size: cover;
```

```
      height: 100vh;
```

```
    }
```

```
    .wrapper{
```

```
      position: absolute;
```

```
      top: 50%;
```

```
      left: 50%;
```

```

transform: translate(-50%, -50%);
width: 100%;
max-width: 550px;
background: rgba(0,0,0,0.8);
padding: 30px;
border-radius: 5px;
box-shadow: 0 0 10px rgba(0,0,0,0.3);
box-shadow: 20px 20px 50px rgba(0,0,0,0.5);
+}

.wrapper .title h1{
  color:#fff;
  text-align: center;
  margin-bottom: 25px;
}

.contact-form{
  display: flex;
}

.input-fields{
  display: flex;
  flex-direction: column;
  margin-right: 4%;
}

.input-fields,
.msg{
  width: 48%;
}

.input-fields .input,
.msg textarea{
  margin: 10px 0;

```

```
background: transparent;
border: 0px;
border-bottom: 2px solid #fff;
padding: 10px;
color: #fff;
width: 100%;
}
.msg textarea{
height: 212px;
}
::-webkit-input-placeholder {
/* Chrome/Opera/Safari */
color: #fff;
}
::-moz-placeholder {
/* Firefox 19+ */
color: #fff;
}
:-ms-input-placeholder {
/* IE 10+ */
color: #fff;
}
.btn {
background: crimson;
text-align: center;
padding: 15px;
border-radius: 5px;
color: #fff;
cursor: pointer;
text-transform: uppercase;
```

```
}

@media screen and (max-width: 600px){
  .contact-form{
    flex-direction: column;
  }
  .msg textarea{
    height: 80px;
  }
  .input-fields,
  .msg{
    width: 100%;
  }
}

.rating{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%,-50%) rotateY(180deg);
  display: flex;
  margin-left: -130px;
  margin-top: -30px;
}

.rating input{
  display: none;
}

.rating label{
  display: block;
  cursor: pointer;
  width: 50px;
}
```

```

    background: #ccc;
}
.rating label:before{
    content: '\f005';
    font-family: fontAwesome;
    position: absolute;
    display: block;
    font-size: 40px;
    color: whitesmoke;

}

.rating label:after{
    content: '\f005';
    font-family: fontAwesome;
    position: absolute;
    display: block;
    font-size: 40px;
    color: gold;
    top: 0;
    opacity: 0;
    transition: .5s;
    text-shadow: 0 2px 5px rgba(0, 0, 0, 0.5);
}

.rating label:hover:after,
.rating label:hover~label:after,
.rating input:checked~label:after{
    opacity: 1;
}

</style>

</head>

```

```

<body>

  <form action="vie.php" method="POST"
style="border:1px solid #ccc">

    <div class="wrapper">

      <div class="title">

        <h1>REPORT</h1>

      </div>

      <div class="contact-form">

        <div class="input-fields">

          <input type="text" class="input" name="email" placeholder="Email">

          <button type="Submit" class="btn">Submit</button>>

          <!-- <input type="text" class="input" placeholder="Job title">-->

        </div>

        <!--<div class="msg">

          ---- <textarea placeholder="Description"></textarea>--

          <div class="btn">Submit</div>

        </div>-->

      </div>

      <!--<div class="rating">

        <input type="radio" name="star" id="star1"><label for="star1"></label>

        <input type="radio" name="star" id="star2"><label for="star2"></label>

        <input type="radio" name="star" id="star3"><label for="star3"></label>

        <input type="radio" name="star" id="star4"><label for="star4"></label>

        <input type="radio" name="star" id="star5"><label for="star5"></label>

      </div>-->

    </div></form>

</body>

</html>

```

hodgelete.php

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <style>

    table {

      margin: 0 auto;

      font-size: large;

      border: 1px solid black;

    }

    body {

      background-image:
url('https://png.pngtree.com/thumb_back/fw800/background/20190307/pngtree-abstract-
background-image_88869.jpg');

      background-repeat: no-repeat;

      background-size: cover;

    }

    h1 {

      text-align: center;

      color: #006600;

      font-size: xx-large;

      font-family: 'Gill Sans', 'Gill Sans MT',
        ' Calibri', 'Trebuchet MS', 'sans-serif';

    }

    td {

      background-color: #E4F5D4;

      border: 1px solid black;

    }

    th,

    td {
```



```

        font-weight: bold;
        border: 1px solid black;
        padding: 10px;
        text-align: center;
    }
    td {
        font-weight: lighter;
    }
</style>
</head>

<body>
    <h1>Welcome</h1>
    <form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" class="container">
    <br><br>
    </form>
    <section>
        <h1>Leave forms</h1>
    </div>
    <!-- TABLE CONSTRUCTION-->
    <table>
        <tr>
            <th>Email</th>
            <th>Delete</th>
        </tr>
        <!-- PHP CODE TO FETCH DATA FROM ROWS-->
        <?php // LOOP TILL END OF
DATA      $con=mysqli_connect("localhost","root","","leavemanagement");
          $query="select * from login ";
          $res=mysqli_query($con,$query);
          while($rows = mysqli_fetch_assoc($res)){

```

```

        echo "<tr>";
        echo "<td>".$rows['email']."</td>";
        echo '<td><form action="requ.php" method="POST">
        <input type="hidden" name="email" value="'.$rows['email'].'"><input type="submit"
value="Delete"></form></td>';
        echo "</tr>";
    }
    ?>
</table>

<form action='hodlogin.php'><input type='submit' value='Back'></form>

</section>

</body>

</html>

```

Leaveform.php

```

<?php
$name="";
$regno="";
$email="";
$from="";
$to="";
$reason="";
$batch="";
if($_SERVER["REQUEST_METHOD"]=="POST"){
    $email=$_POST['email'];
    $name=$_POST['name'];
    $fdate=$_POST['fdate'];
    $tdate=$_POST['tdate'];
    $reason=$_POST['reason'];
    $status="waiting";
    $regno=$_POST['regno'];
    $batch=$_POST['batch'];
}

```

```

$count=0;

$con=mysqli_connect("localhost","root","","leavemanagement");

$query="SELECT * FROM leavedetails ";

$res=mysqli_query($con,$query);

if(mysqli_num_rows($res)>=0){

    while($rows = mysqli_fetch_assoc($res)){

        if($email==$rows['email'] ){

            $count++;}}

    if($count>=16){

        header("location:alert.php");

    }

    else{

        $query="INSERT INTO
leavedetails(name,email,fdate,tdate,reason,status,RegisterNumber,batch)
values('$name','$email','$fdate','$tdate','$reason','$status','$regno','$batch')";

        $res=mysqli_query($con,$query);

        header("location:studentlogin.php");

    }

}??>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <script src="https://kit.fontawesome.com/1c2c2462bf.js" crossorigin="anonymous"></script>

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/3.3.7/css/bootstrap.min.css">

</head>

<style>

```

```
@import url('https://fonts.googleapis.com/css?family=Open+Sans:400,500&display=swap');

* {
  box-sizing: border-box;
}

body {
  background-color: #9b59b6 ;
  font-family: 'Open Sans', sans-serif ;
  display: flex ;
  align-items: center ;
  justify-content: center ;
  min-height: 100vh ;
  margin: 0 ;
}

.container {
  background-color: #fff ;
  border-radius: 5px ;
  box-shadow: 0 2px 5px rgba(0, 0, 0, 0.3) ;
  overflow: hidden ;
  width: 400px ;
  max-width: 100% ;
}

.form {
  padding: 30px 10px ;
}

.form-controls {
  padding-bottom: 15px ;
  position: relative ;
}

.form-controls label {
  display: inline-block ;
```

```
    margin-bottom: 5px ;
}
.form-controls input {
    border: 2px solid #f0f0f0;
    border-radius: 4px;
    display: block;
    font-family: inherit;
    font-size: 14px;
    padding: 10px;
    width: 100%;
}
.form-controls input:focus {
    outline: 0;
    border-color: #777;
}
.form-controls.success input {
    border-color: #2ecc71;
}
.form-controls.error input {
    border-color: #e74c3c;
}
.form button {
    background-color: #8e44ad;
    border: 2px solid #8e44ad;
    border-radius: 4px;
    color: #fff;
    display: block;
    font-family: inherit;
    font-size: 16px;
    padding: 10px;
```

```

margin-top: 20px;

width: 100%;

}

.logoutLblPos{

position:fixed;

right:10px;

top:5px;

}

</style>

<body>

<div class="container">

    <form id="form" class="form" action="<?php echo $_SERVER['PHP_SELF'];?>"
method="POST" >

        <div class="form-controls">

            <label for="username">Full Name</label>

            <input type="text" id="first_name" name="name"
required/>

        </div>

        <div class="form-controls">

            <label for="username">Register Number</label>

            <input type="text" id="username" name="regno"
required/>

        </div>

        <div class="form-controls">

            <label for="username">Batch Number</label>

            <input type="text" id="first_name" name="batch"
required/>

        </div>

        <div class="form-controls">

            <label for="username">Email</label>

```

```

        <input type="email" id="email" name="email"
        pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$"
        autocomplete="off" required/>
    </div>
    <div class="form-controls">
        <label for="username">FROM</label>
        <input type="date" id="username" name="fdate"
        required/>
    </div>
    <div class="form-controls">
        <label for="username">TO</label>
        <input type="date" id="password" name="tdate" required/>
    </div>
    <div class="form-controls">
        <label for="website">Reason</label>
        <input type="text" id="website" name="reason" required/>
    </div>
    <button id="sub ">Submit</button>
</form>
<form action="studentlogin.php"><button id="sub ">Back</button></form>
<form align="right" name="form1" method="post" action="logout.php">
    <label class="logoutLblPos">
        <input name="submit2" type="submit" id="submit2" value="log out">
    </label>
</form>
</div>
</body>
</html>

```

login.php

```
<?php
$password="";
$pswrepeat="";

if($_SERVER["REQUEST_METHOD"]=="POST"){
$password = $_POST['password'];
$email = $_POST['email'];

//echo $password;

//echo $email;

$con=mysqli_connect("localhost","root","","leavemanagement");

$query="select * from loginform";

$res=mysqli_query($con,$query);

if(mysqli_num_rows($res)>0){

    while($row = mysqli_fetch_assoc($res))

    {

        if($email==$row['User'] && $password==$row['Pass']){

            header("location:fp.php");

        }

    else{

}

    }

}

}??>

<html>

<head>

<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" >

<title>LOGIN PAGE</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">
```



```

<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<script src="https://kit.fontawesome.com/86e1dcc20b.js" crossorigin="anonymous"></script>
<style>

body{
    margin:0;
padding:0;
font-family:sans-serif;
background-image: url("https://s4.sathyainfo.com/s/9b95ed6c5d464c5/img.jpeg");
background-repeat: no-repeat ;
background-size: cover;
    }

.login{
width: 400px;
position: absolute;
top: 50%;
left: 50%;
transform:translate(-50%,-50%);
color:white;
background-color:rgba(0,0,0,0.5);
padding: 0px 30px 50px 20px;
height:450px;
}

.login h1{
float: left;
font-size: 40px;
border-bottom: 6px solid white;
margin-bottom:50px;

```

```
padding:20px 0;
}
.text{
width:100%;
overflow:hidden;
font-size:20px;
padding:8px 0 8px;
margin:8px 0;
border-bottom:1px solid white;
}
.text i{
width:26px;
float:left;
text-align:center;
}
.text input{
border: none;
outline: none;
background: none;
color: #ffff;
font-size: 18px;
width:80%;
float:left;
margin:0 10px;
}
.btn-submit{
width:100%;
background: none;
border:2px solid white;
color: #ffff;
```

```
padding:5px;
font-size:18px;
cursor:pointer;
margin:12px 0 12px;
}
.sign{
padding:10px;
font-size: 18px;
}
</style>
</head>
<body>
<div class="login">
<h1>Department Login</h1>
<div class="text">
<i class="far fa-user"></i>
<input type="text" placeholder="User Name" name="email" >
</div>
<div class="text">
<i class="fas fa-lock"></i>
<input type="password" placeholder="Password" name="password" >
</div>
<button type="submit" class="btn-submit">LOGIN</button>
</div>
</body>
</html>
```

teachersignun.php

```
<?php
$password="";
$pswrepeat="";
```

```

if($_SERVER["REQUEST_METHOD"]=="POST"){
$password = $_POST['psw'];
$pswrepeat = $_POST['psw-repeat'];
if ($password != $pswrepeat) {
    echo '<script>alert("Enter the correct Password")</script>';
}else{
    echo '<script>alert("Successful")</script>';
}

$con=mysqli_connect("localhost","root","","leavemanagement");

$query="select * from teacherlogin";
$res=mysqli_query($con,$query);
if(mysqli_num_rows($res)>=0){
    $email=$_POST['email'];
    $query="INSERT INTO teacherlogin(email,password) values('$email','$password')";
    $res=mysqli_query($con,$query);
}else{
    echo"Table created sucessfully";
}
}??>

<!DOCTYPE html>

<html>

<style>

body {font-family: Arial, Helvetica, sans-serif;

background-image: url("https://cdn.elearningindustry.com/wp-content/uploads/2013/08/321-free-tools-for-teachers-free-educational-technology-1024x574.png");

background-repeat: no-repeat;

background-size: cover;}

* {box-sizing: border-box}

/* Full-width input fields */

input[type=text], input[type=password] {

width: 30%;

```

```

padding: 15px;
margin: 5px 0 22px 0;
display: inline-block;
border: none;
background: #f1f1f1;
}
input[type=text]:focus, input[type=password]:focus {
    background-color: #ddd;
    outline: none;
}
hr {
    border: 1px solid #f1f1f1;
    margin-bottom: 25px;
}
/* Set a style for all buttons */
button {
    background-color: #4CAF50;
    color: white;
    padding: 14px 20px;
    margin: 8px 0;
    border: none;
    cursor: pointer;
    width: 50%;
    opacity: 0.9;
}
button:hover {
    opacity: 1;
}
/* Extra styles for the cancel button */
.cancelbtn {

```

```

padding: 14px 20px;
background-color: #f44336;
}
/* Float cancel and signup buttons and add an equal width */
.cancelbtn, .signupbtn {
float: left;
width: 20%;
}
/* Add padding to container elements */
.container {
padding: 16px;

}
/* Clear floats */
.clearfix::after {
content: "";
clear: both;
display: table;
}
/* Change styles for cancel button and signup button on extra small screens */
@media screen and (max-width: 300px) {
.cancelbtn, .signupbtn {
width: 100%;
}
}
</style>
<body>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST">
<form action="/action_page.php" style="border:1px solid #ccc">
<div class="container">

```

```

<h1>Sign Up</h1>
<p>Please fill in this form to create an account.</p>
<hr>
<label for="email"><b>Email</b></label>
<input type="text" placeholder="Enter Email" name="email" required><br>
<label for="psw"><b>Password</b></label>
<input type="password" placeholder="Enter Password" name="psw" required><br>
<label for="psw-repeat"><b>Repeat Password</b></label>
<input type="password" placeholder="Repeat Password" name="psw-repeat" required><br>
<label>
  <input type="checkbox" checked="checked" name="remember" style="margin-
bottom:15px"> Remember me
</label>
<br>
<div class="clearfix">
  <form action="teacherlogin.php">
    <button type="submit" class="signupbtn">Sign UP</button>
  </form>
  <form action="teacherlogin.php">
    <button type="submit" class="cancelbtn">Cancel</button>
  </form>
</div>
</div>
</form>
</body>
</html>

```

vie.php

```

<!DOCTYPE html>
<html lang="en">
<head>

```

```
<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

<style>

    @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800
;900&display=swap');

    body{

        font-family: "Poppins", sans-serif;

        background-color: cornflowerblue;

    }

    table, th, td{

        border: 1px solid black;

    }

    .cont{

        background-color:white;

        box-sizing: border-box;

        border-style: groove;

        border-color: rgb(139, 125, 125);

        position: relative;

        width: 900px;

        height: 800px;

        margin-top: 30px;

        justify-content: center;

        align-items: center;

        margin-left: 200px;

        box-shadow: 0 10px 20px rgba(0,0,0,0.5);

    }

</style>

</head>
```



```

<body>

<div class="cont">

<table style="width: 100%;">

<tr>

<th>Name</th>

<th>Email</th>

<th>From date</th>

<th>

To date

</th>

<th>Reason</th>

</tr>

<?php
session_start();
if($_SERVER["REQUEST_METHOD"]=="POST")
{
$email = $_POST['email'];
$con=mysqli_connect("localhost","root","","leavemanagement");
$query="select * from leavedetails WHERE email='".$email."' ";
$res=mysqli_query($con,$query);
while($rows = mysqli_fetch_assoc($res))
{
echo "<tr>";
echo "<td>".$rows['name']. "</td>";
echo "<td>".$rows['email']. "</td>";
echo "<td>".$rows['fdate']. "</td>";
echo "<td>".$rows['tdate']. "</td>";
echo "<td>".$rows['reason']. "</td>";
}
}
}

```

?>

</table>

</div>

</body>

</body>

</html>

teacherslogin.php

<?php

```
$password="";
```

```
$pswrepeat="";
```

```
if($_SERVER["REQUEST_METHOD"]=="POST"){
```

```
$password = $_POST['psw'];
```

```
$email = $_POST['email'];
```

```
//echo $password;
```

```
//echo $email;
```

```
$con=mysqli_connect("localhost","root","","leavemanagement");
```

```
$query="select * from teacherlogin";
```

```
$res=mysqli_query($con,$query);
```

```
if(mysql_num_rows($res)>0){
```

```
while($row = mysqli_fetch_assoc($res))
```

 $\{$

```
if($email==$row['email'] && $password==$row['password']){
```

```
header("location:view.php");
```

}

```
else{
```

}

}

}

 $\} ? >$

```
<!DOCTYPE html>
```

```

<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body, html {
  height: 100%;
  font-family: Arial, Helvetica, sans-serif;
} body{
  background-image:url("https://theprint.in/wp-content/uploads/2020/07/ONLINE-
CLASSES.jpg");
  background-repeat: no-repeat;
  background-size: cover;
}
* {
  box-sizing: border-box;
}
bg-img {
  /* The image used */
  background-image: url("http://stmlportal.net/images/staff.png");
  min-height: 400px;
  /* Center and scale the image nicely */
  background-position: center;
  background-repeat: no-repeat;
  position: relative;
}
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;

```

```
}  
li {  
    float: left;  
}  
li a {  
    display: block;  
    color: white;  
    text-align: center;  
    padding: 14px 16px;  
    text-decoration: none;  
}  
li a:hover {  
    background-color: #111;  
}  
/* Add styles to the form container */  
.container {  
    position: absolute;  
    right: 0;  
    margin: 20px;  
    max-width: 300px;  
    padding: 16px;  
    background-color: white;  
}  
/* Full-width input fields */  
input[type=text], input[type=password] {  
    width: 100%;  
    padding: 15px;  
    margin: 5px 0 22px 0;  
    border: none;  
    background: #f1f1f1;
```

```

}
input[type=text]:focus, input[type=password]:focus {
    background-color: #ddd;
    outline: none;
}
/* Set a style for the submit button */
.btn {
    background-color: #4CAF50;
    color: white;
    padding: 16px 20px;
    border: none;
    cursor: pointer;
    width: 100%;
    opacity: 0.9;
}
.btn:hover {
    opacity: 1;
}
</style>
</form>
</head>
<body>
<div class="bg-img">
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" class="container">
<h1>Login</h1>
<label for="email"><b>Email</b></label>
<input type="text" placeholder="Enter Email" name="email" required>
<label for="psw"><b>Password</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>
<button type="submit" class="btn">Login</button>

```

```
<h3>Do not have an account?</h3>
<a href="teachersignup.php">signup</a>
<a href="fp.php">Back</a>
</form>
</div>
</body>
</html>
```

viewstatus.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <style>
    table {
      margin: 0 auto;
      font-size: large;
      border: 1px solid black;
    }
    body {
      background-image: url('https://wallpapercave.com/wp/wp2704824.jpg');
    }
    h1 {
      text-align: center;
      color: #006600;
      font-size: xx-large;
      font-family: 'Gill Sans', 'Gill Sans MT',
        'Calibri', 'Trebuchet MS', 'sans-serif';
    }
    td {
      background-color: #E4F5D4;
```

```

        border: 1px solid black;
    }
    th,
    td {
        font-weight: bold;
        border: 1px solid black;
        padding: 10px;
        text-align: center;
    }
    td {
        font-weight: lighter;
    }
</style>
</head>
<body>
    <h1>Welcome</h1>
    <form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" class="container">
    <br><br>
    </form>
    <section>
        <h1>Accepted list</h1>
    </div>
    <!-- TABLE CONSTRUCTION-->
    <table>
        <tr>
            <th>Name</th>
            <th>Email</th>
            <th>from</th>
            <th>To</th>
            <th>reason</th>

```

```

        <th>Status</th>

        <th>Register Number</th>

        <th>Batch</th>

    </tr>

    <!-- PHP CODE TO FETCH DATA FROM ROWS-->

    <?php // LOOP TILL END OF DATA

        $con=mysqli_connect("localhost","root","","leavemanagement");

        $query="select * from leavedetails ";

        $res=mysqli_query($con,$query);

        while($rows = mysqli_fetch_assoc($res)){

            echo "<tr>";

            echo "<td>".$rows['name']. "</td>";

            echo "<td>".$rows['email']. "</td>";

            echo "<td>".$rows['fdate']. "</td>";

            echo "<td>".$rows['tdate']. "</td>";

            echo "<td>".$rows['reason']. "</td>";

            echo "<td>".$rows['status']. "</td>";

            echo "<td>".$rows['RegisterNumber']. "</td>";

            echo "<td>".$rows['batch']. "</td>";

            echo "</tr>";

        }

    ?>

    </table>

    <form action='hodacpt.php'><input type='submit' value='Back'></form>

</section>

</body>

</html>

```

request.php

```

<?php

$mail = $_POST['email'];

```



```

//echo $password;
//echo $email;
$con=mysqli_connect("localhost","root","","leavemanagement");
$query="Update leavedetails SET status='accepted' where email='".$mail."'";
if(mysqli_query($con,$query))
{
    $body="This is your OTP for Covid E-pass Login.\n\r Thank You ";
    if(mail($mail,$body,"From:arunalakshmi609@gmail.com"))
    {
        echo $mail;
        echo '<script>alert("OTP Unsuccessfull")</script>';
    }
    else
    {
        echo '<script>alert("OTP Unsuccessfull")</script>';
        header('refresh:0.1,url=hodacpt.php');
    }
}
?>

```

requ.php

```

<?php
$mail = $_POST['email'];
//echo $password;
//echo $email;
$con=mysqli_connect("localhost","root","","leavemanagement");
$query="DELETE FROM login where email='".$mail."'";
if(mysqli_query($con,$query))
{
    echo '<script>alert("Deleted successfully")</script>';
    header("location:hoddelete.php");
}

```

```
}
```

```
?>
```

mail.php

```
<?php
```

```
$sub="durga";
```

```
$msg="durga";
```

```
$rec="elangorevathi06@gmail.com";
```

```
mail($rec,$sub,$msg);
```

```
?>
```

request1.php

```
<?php
```

```
$email = $_POST['email'];
```

```
//echo $password;
```

```
//echo $email;
```

```
$con=mysqli_connect("localhost","root","","leavemanagement");
```

```
$query="Update leavedetails SET status='declined' where email='".$email."'";
```

```
if(mysqli_query($con,$query)){
```

```
    header("location:hodacpt.php");
```

```
}
```

```
?>
```

studentlogin.php

```
<?php
```

```
$password="";
```

```
$pswrepeat="";
```

```
if($_SERVER["REQUEST_METHOD"]=="POST"){
```

```
$password = $_POST['password'];
```

```
$email = $_POST['email'];
```

```
//echo $password;
```

```
//echo $email; $con=mysqli_connect("localhost","root","","leavemanagement");
```

```
$query="select * from login";

$res=mysqli_query($con,$query);

if(mysqli_num_rows($res)>0){

    while($row = mysqli_fetch_assoc($res))

    {

        if($email==$row['email'] && $password==$row['password']){

            header("location:leaveform.php");

        }

    else{

}

    }

}

}?>

<html>

<head>

<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" >

<title>LOGIN PAGE</title>

<meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"

href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

    <script

src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>

<script src="https://kit.fontawesome.com/86e1dcc20b.js" crossorigin="anonymous"></script>

<style>

body{

margin:0;

padding:0;

font-family:sans-serif;
```

```
background-image: url("https://img.emg-services.net/HtmlPages/HtmlPage12273/untitled-
design-10.jpg");
background-repeat: no-repeat ;
background-size: cover;
}
.login{
width: 400px;
position: absolute;
top: 50%;
left: 50%;
transform:translate(-50%,-50%);
color:white;
background-color:rgba(0,0,0,0.5);
padding: 0px 30px 50px 20px;
height:450px;
}
.login h1{
float: left;
font-size: 40px;
border-bottom: 6px solid white;
margin-bottom:50px;
padding:20px 0;
}
.text{
width:100%;
overflow:hidden;
font-size:20px;
padding:8px 0 8px;
margin:8px 0;
border-bottom:1px solid white;
}
```

```
.text i{
width:26px;
float:left;
text-align:corner;
}

.text input{
border: none;
outline: none;
background: none;
color: #ffff;
font-size: 18px;
width:80%;
float:left;
margin:0 10px;
}

.btn-submit{
width:100%;
background: none;
border:2px solid white;
color: #ffff;
padding:5px;
font-size:18px;
cursor:pointer;
margin:12px 0 12px;
}

.sign{
padding:10px;
font-size: 18px;
}

</style>
```

```

</head>

<body>

<div class="login">

<h1>login</h1>

<div class="text">

<i class="far fa-user"></i>

<input type="text" placeholder="Email" name="email" >

</div>

<div class="text">

<i class="fas fa-lock"></i>

<input type="password" placeholder="Password" name="password" >

</div>

<button type="submit" class="btn-submit">LOGIN</button>

<a href="fp.php" style="color:white"> Back</a>

<a href="viewstatus.php" style="color:white"> View Status</a><br>

</div>

</div>

</body>

</html>

```

studentsignup.php

```

<?php

$password="";

$pswrepeat="";

if($_SERVER["REQUEST_METHOD"]=="POST"){

$password = $_POST['password'];

$name = $_POST['name'];

$con=mysqli_connect("localhost","root","","leavemanagement");

$query="select * from login";

$res=mysqli_query($con,$query);

if(mysqli_num_rows($res)>=0){

```

```

$email=$_POST['email'];
$query="INSERT INTO login(email,password) values('$email','$password')";
$res=mysqli_query($con,$query);
header("location:studentlogin.php");
}
}??>
<html>
<head>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" >
<title>SIGNIN PAGE</title>
<meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<script src="https://kit.fontawesome.com/86e1dcc20b.js" crossorigin="anonymous"></script>
<style>
body{
    margin=0;
padding=0;
font-family:sans-serif;
background-image: url("https://w3layouts.com/wp-
content/uploads/2017/12/student_registration_form_Free13-12-2017_1736833576.jpg") ;
background-repeat:no-repeat ;
background-size: cover;
    }
.login{
width: 400px;
position: absolute;

```

```
top: 50%;
left: 50%;
transform:translate(-50%,-50%);
color:white;
background-color:lightpink;
padding: 0px 30px 50px 20px;
height:450px;
}

.login h1{
float: left;
font-size: 40px;
border-bottom: 6px solid white;
margin-bottom:50px;
padding:20px 0;
}

.text{
width:100%;
overflow:hidden;
font-size:20px;
padding:8px 0 8px;
margin:8px 0;
border-bottom:1px solid white;
}

.text i{
width:26px;
float:left;
text-align:corner;
}

.text input{
border: none;
```



```

outline: none;
background: none;
color: #ffff;
font-size: 18px;
width:80%;
float:left;
margin:0 10px;
}

.btn-submit{
width:100%;
background: none;
border:2px solid white;
color: #ffff;
padding:5px;
font-size:18px;
cursor:pointer;
margin:12px 0 12px;
}

.sign{
padding:10px;
font-size: 18px;
}

</style>

</head>

<body>

    <form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" >

<div class="login">

<h1>SIGNIN</h1>

<div class="text">

<i class="far fa-user"></i>

```

```

<input type="text" placeholder="Name" name="name" required="required">
</div>
<div class="text">
<i class="far fa-user"></i>
<input type="text" placeholder="Email" name="email" required="required">
</div>
<div class="text">
<i class="fas fa-lock"></i>
<input type="password" placeholder="Password" name="password" required="required" >
</div>
<button type="submit" class="btn-submit">Signin</button>
</form>
<form action="studentlogin.php"><button type="submit" class="btn-
submit">Back</button></form>
</div>
</body>
</html>

```

hodacpt.php

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<style>
table {
margin: 0 auto;
font-size: large;
border: 1px solid black;
}
body {

```

```

        background-image:
url('https://png.pngtree.com/thumb_back/fw800/background/20190307/pngtree-abstract-
background-image_88869.jpg');

        background-repeat: no-repeat;

        background-size: cover;

    }

h1 {

    text-align: center;

    color: #006600;

    font-size: xx-large;

    font-family: 'Gill Sans', 'Gill Sans MT',
    ' Calibri', 'Trebuchet MS', 'sans-serif';

}

td {

    background-color: #E4F5D4;

    border: 1px solid black;

}

th,

td {

    font-weight: bold;

    border: 1px solid black;

    padding: 10px;

    text-align: center;

}

td {

    font-weight: lighter;

}

</style>

</head>

<body>

```

```

<h1>Welcome</h1>

<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" class="container">

<br><br>

</form>

<section>

    <h1>Leave forms</h1>

</div>

<!-- TABLE CONSTRUCTION-->

<table>

    <tr>

        <th>Name</th>

        <th>Email</th>

        <th>from</th>

        <th>To</th>

        <th>reason</th>

        <th>Register Number</th>

        <th>Batch</th>

    </tr>

    <!-- PHP CODE TO FETCH DATA FROM ROWS-->

    <?php // LOOP TILL END OF DATA

        $con=mysqli_connect("localhost","root","","leavemanagement");

        $query="select * from leavedetails WHERE status='waiting'";

        $res=mysqli_query($con,$query);

        while($rows = mysqli_fetch_assoc($res)){

            echo "<tr>";

            echo "<td>".$rows['name'].</td>";

            echo "<td>".$rows['email'].</td>";

            echo "<td>".$rows['fdate'].</td>";

            echo "<td>".$rows['tdate'].</td>";

            echo "<td>".$rows['reason'].</td>";

```

```

        echo "<td>".$rows['RegisterNumber']."</td>";
        echo "<td>".$rows['batch']."</td>";
        echo '<td><form action="request.php" method="POST">
            <input type="hidden" name="email" value='.$rows['email'].'><input type="submit"
value="Accept"></form></td>'; echo '<td><form action="request1.php" method="POST">
            <input type="hidden" name="email" value='.$rows['email'].'><input type="submit"
value="Decline"></form></td>';
        echo "</tr>";
    }
    ?>
</table>

<form action='hodlogin.php'><input type='submit' value='Back'></form>
<form action='form1.php'><input type='submit' value='Student details'></form>
<form action='hoddelete.php'><input type='submit' value='Remove Student'></form>
</section>
</body>
</html>

```

hodlogin.php

```

<?php
$password="";
$pswrepeat="";
if($_SERVER["REQUEST_METHOD"]=="POST"){
    $password = $_POST['password'];
    $email = $_POST['email'];
    //echo $password;
    //echo $email;

    $con=mysqli_connect("localhost","root","","leavemanagement");
    $query="select * from hodlogin";
    $res=mysqli_query($con,$query);
    if(mysqli_num_rows($res)>0){

```

```
while($row = mysqli_fetch_assoc($res))
{
    if($email==$row['email'] && $password==$row['password']){
        header("location:hodacpt.php");
    }
else{
        header("location:fp.php");
    }
}

}}?>

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<title> Login Page </title>

<style>

Body {

font-family: Calibri, Helvetica, sans-serif;

background-color: pink;

background-image:url('https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcSJTEHmnbce8DXuHFoQ3Y6F_1cucabUP8FgzW&usqp=CAU');

}

button {

background-color: #4CAF50;

width: 20%;

color: orange;

padding: 15px;

margin: 10px 0px;
```

```

    border: none;
    cursor: pointer;
}
form {
    border: 3px solid #f1f1f1;
}
input[type=text], input[type=password] {
    width: 20%;
    margin: 8px 0;
    padding: 12px 20px;
    display: inline-block;
    border: 2px solid green;
    box-sizing: border-box;
}
button:hover {
    opacity: 0.7;
}
.cancelbtn {
    width: auto;
    padding: 10px 18px;
    margin: 10px 5px;
}
.container {
    padding: 25px;
    background-color: lightblue;
    background-image: url("https://media.giphy.com/media/l2QE73aOjxO1fkd6E/giphy.gif");
min-height: 400px;
/* Center and scale the image nicely */
background-position: center;
background-repeat: no-repeat;

```

```

    position: relative;

    }
</style>
</head>
<body>

    <center> <h1> Welcome HOD </h1> </center>

    <form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" >

        <div class="container">

            <label>Username : </label> <br>

            <input type="text" placeholder="Enter email" name="email" ><br>

            <label>Password : </label> <br>

            <input type="password" placeholder="Enter Password" name="password" > <br>

            <button type="submit">Login</button> <br>

            <input type="checkbox" checked="checked"> Remember me

            <button type="submit" class="cancelbtn"> Cancel</button>

        </div>

    </form>
</body>
</html>

```

view.php

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <style>
        table {
            margin: 0 auto;
            font-size: large;
            border: 1px solid black;

```



```

    }

    body {
        background-image: url('https://wallpapercave.com/wp/wp2704824.jpg');
    }

    h1 {
        text-align: center;
        color: #006600;
        font-size: xx-large;
        font-family: 'Gill Sans', 'Gill Sans MT',
        ' Calibri', 'Trebuchet MS', 'sans-serif';
    }

    td {
        background-color: #E4F5D4;
        border: 1px solid black;
    }

    th,
    td {
        font-weight: bold;
        border: 1px solid black;
        padding: 10px;
        text-align: center;
    }

    td {
        font-weight: lighter;
    }

</style>
</head>

<body>
    <h1>Welcome</h1>

```

```

<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="POST" class="container">
<br><br>
</form>
<section>
    <h1>Accepted list</h1>
</div>
<!-- TABLE CONSTRUCTION-->
<table>
    <tr>
        <th>Name</th>
        <th>Email</th>
        <th>from</th>
        <th>To</th>
        <th>reason</th>
        <th>Status</th>
        <th>Register Number</th>
        <th>Batch</th>
    </tr>
    <!-- PHP CODE TO FETCH DATA FROM ROWS-->
    <?php // LOOP TILL END OF DATA
        $con=mysqli_connect("localhost","root","","leavemanagement");
        $query="select * from leavedetails WHERE status='accepted'";
        $res=mysqli_query($con,$query);
        while($rows = mysqli_fetch_assoc($res)){
            echo "<tr>";
            echo "<td>".$rows['name'].</td>";
            echo "<td>".$rows['email'].</td>";
            echo "<td>".$rows['fdate'].</td>";
            echo "<td>".$rows['tdate'].</td>";
            echo "<td>".$rows['reason'].</td>";

```

```

        echo "<td>".$rows['status']. "</td>";
        echo "<td>".$rows['RegisterNumber']. "</td>";
        echo "<td>".$rows['batch']. "</td>";
        echo "</tr>";
    }
    ?>
</table>

<form action='teacherlogin.php'><input type='submit' value='Back'></form>

</section>

</body>

</html>

```

fp.php

```

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

body {

    font-family: Arial, Helvetica, sans-serif;

    margin: 0;

}

html {

    box-sizing: border-box;

}

*, *:before, *:after {

    box-sizing: inherit;

}

.column {

    float: left;

```

```
width: 33.3%;  
margin-bottom: 16px;  
padding: 0 8px;  
}  
.card {  
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);  
  margin: 8px;  
}  
.about-section {  
  padding: 50px;  
  text-align: center;  
  background-color: pink;  
  color: white;  
}  
.container {  
  padding: 0 16px;  
}  
.container::after, .row::after {  
  content: "";  
  clear: both;  
  display: table;  
}  
.title {  
  color: grey;  
}  
.button {  
  border: none;  
  outline: 0;  
  display: inline-block;  
  padding: 8px;
```

```

color: white;
background-color: #000;
text-align: center;
cursor: pointer;
width: 100%;
}
.button:hover {
background-color: #555;
}
@media screen and (max-width: 650px) {
.column {
width: 100%;
display: block;
}
}
</style>
</head>
<body>
<div class="about-section">

<h1 style="color:red">LEAVE MANAGEMENT SYSTEM</h1>
</div>
<marquee bgcolor="yellow">Don't take leave unnecessarily</marquee>
<div class="row">
<div class="column">
<div class="card"><a href="hodlogin.php">

<div class="container"></a>
<h2>H.O.D</h2>
<p class="title">HOD</p>

```

```

    </div>
</div>
</div>
<div class="column">
    <div class="card">
        <a href="teacherlogin.php">
            </a>
        <div class="container">
            <h2>Teachers</h2>
        </div>
    </div>
</div>
</div>
<div class="column">
    <div class="card">
        <a href="studentlogin.php">
            </a>
        <div class="container">
            <h2>Student</h2>
        </div>
    </div>
</div>
</div>
</div>
</body>
</html>

```

index.php

```

<?php
    if (!empty($_SERVER['HTTPS']) && ('on' == $_SERVER['HTTPS'])) {
        $uri = 'https://';
    } else {

```

```

        $uri = 'http://';

    }

    $uri .= $_SERVER['HTTP_HOST'];

    header('Location: '.$uri.'/dashboard/');

    exit;

?>

```

Something is wrong with the XAMPP installation :-(

leavemanagement.sql

```

-- phpMyAdmin SQL Dump
-- version 5.0.4
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Mar 03, 2021 at 02:51 PM
-- Server version: 10.4.17-MariaDB
-- PHP Version: 8.0.2

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `leavemanagement`
--
--
-- -----

```

```

--
-- Table structure for table `hodlogin`
--
CREATE TABLE `hodlogin` (
  `email` varchar(30) NOT NULL,
  `password` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
--
-- Dumping data for table `hodlogin`
--
INSERT INTO `hodlogin` (`email`, `password`) VALUES
('18euit066@skcet.ac.in', 'gowsik');
-----
--
-- Table structure for table `leavedetails`
--
CREATE TABLE `leavedetails` (
  `name` varchar(30) NOT NULL,
  `email` varchar(30) NOT NULL,
  `fdate` date NOT NULL,
  `tdate` date NOT NULL,
  `reason` varchar(30) NOT NULL,
  `status` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
--
-- Dumping data for table `leavedetails`

```



```
--
INSERT INTO `leavedetails` (`name`, `email`, `fdate`, `tdate`, `reason`, `status`)
VALUES
('kanga', '18euit@g.com', '2021-03-04', '2021-03-13', 'as i am Suffering',
'accepted'),
('kanga', '18euit066@nsfnv.ac.in', '2021-03-18', '2021-03-23', 'Suma', 'declined'),
('kanga', '18jkfff@ac.in', '2021-05-13', '2021-03-22', 'Fever', 'declined'),
('Bharath', 'ubcuik@gmail.com', '2021-03-12', '2021-03-15', 'Fever', 'accepted');
```

```
-- -----
```

```
--
-- Table structure for table `login`
```

```
--
CREATE TABLE `login` (
  `email` varchar(200) NOT NULL,
  `password` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--
-- Dumping data for table `login`
--
INSERT INTO `login` (`email`, `password`) VALUES
('thangam@gmail.com', 'Thangam@2000');
```

```
-- -----
```

```
--
-- Table structure for table `teacherlogin`
```

```
--
CREATE TABLE `teacherlogin` (
  `email` varchar(30) NOT NULL,
  `password` varchar(30) NOT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--

-- Dumping data for table `teacherlogin`

--

INSERT INTO `teacherlogin` (`email`, `password`) VALUES
('18euit066@skcet.ac.in', 'gowsik');

COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```



Future Enhancement

Although the project has been completed there are several plans for enhancement of the project in future. Every students individual leave record can be tabulated in a pie chart format to ascertain their performance during the working session for the teachers. HOD will receive a notification if a student takes leave above the limit. And the students will also get a warning message. Mail will be sent to a student when the admin approved the particular students leave permission. The summarised data generated by this system can be further provided to different departments.

Conclusion

A magnifying glass with a black handle and a silver rim is positioned over the word "Conclusion". The lens of the magnifying glass is centered over the letters "clu", making them appear larger and more prominent than the rest of the word. The background is a light gray gradient.

6. CONCLUSION

Leave management is very useful for the Department of the college to maintain the leave records of the students. The higher authorities may accept or deny the leave applications requested by the students of that department. Hence this is much easier for the students to apply their leaves. The user only needs basic products like a Computer/Mobile and an internet facility.

Bibliography



7. BIBLIOGRAPHY

PHP REFERENCE:

- ❖ “Beginning PHP and MySQL” W. Jason Gilmore-2010
- ❖ “PHP and MySQL” Tim Converse=Joyee Park and Clark Morgan- Wiley India-2008
- ❖ VIKRAM VASWANI=“PHP and MySQL”- Tata McGraw Hill-2005

MySQL SERVER REFERENCE:

- ❖ MySQL CRASH COURSE-BEN FORTA, SAMS-2006
- ❖ O'Reilly.High.Performance.MySQL.3rd.Edition.Mar.2012

WEBSITES:

- ❖ https://www.w3schools.com/php/php_intro.asp
- ❖ <https://www.tutorialspoint.com>
- ❖ <https://www.codeacademy.com>

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is certify that the project entitled

ONLINE CAB BOOKING

Submitted by

K.Durga Lakshmi

18AUCS07

A.Maria Jubliya Pratheeba

18AUCS25

J.Piuela

18AUCS36

S.Vishnu Priya

18AUCS48

In partial fulfilment of the requirement for the award of degree in **BACHELOR OF COMPUTER SCIENCE** during the year 2020-2021.

Guide

Head of the Department

Submitted for the Examination held on 15.4.2021

Examiner

Principal

St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ACKNOWLEDGEMENT

TABLE

OF

CONTENTS

CONTENTS

S.NO	CONTENT	PAGE NO
1.	SYNOPSIS	
2.	SYSTEM ANALYSIS 2.1 Existing System 2.2 Proposed System. 2.3 Analysing The System	
3.	SYSTEM REQUIREMENTS AND SPECIFICATIONS 3.1 Working Environment 3.2 About Software	
4.	SYSTEM DESIGN 4.1 Table Design 4.2 Data Flow Diagram 4.3 Flow Chart 4.4 Form Design 4.5 Code Design	
5.	FUTURE ENHANCEMENT	
6.	CONCLUSION	
7.	BIBILOGRAPHY	

SYNOPSIS

SYNOPSIS

This project entitled “***Go Taxi Cab Booking System***” acts as a centralized database which contains the details about various types of Cab and Locations. A user visiting the website can see a wide range of cab and facilities and the booking rates. The booking rate depends on the type of locations they where booked.

The Front End of the Project is ASP.NET and the Back End is MS Access.

The project “***Go Taxi Cab Booking System***” is a system based on accessing the internet to book for cab and location. The purpose of this study is to develop and implement an online cab booking system for hotels that will replace the manual method of booking for cab. The previous system for booking rooms were faced with so many problems like, delay in processing the customer booking or paying for cab that is below or beyond his standard, cause difficulty for emergency booking.

The use of online view of cab and uploading of available cabs and facilities was used for the new system so that the customer can view and make his choice before booking, and also in the case of emergency travelling. This new system assisted the cab owners in managing their cabs, because they can also regulate the receptionist moves and avoid fraudulent activities. It also increased the efficiency of the cab booking and also their profit margin, once they have better and good facilities.

SYSTEM ANALYSIS

SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

- To difficult in booking the cabs.
- Only in city and town we can booked cab.
- Payment details and drivers charges are difficult in understanding.
- Admin can't enter the driver's details in online.
- No Security assurance for user information.
- Manual process is too difficult.

DRAWBACKS OF EXISTING SYSTEM

- There may be errors occurring while entering the details manually.
- Various activities are difficult to carry out manually.
- Time consumption is very high.
- No security, more expensive.

2.2 PROPOSED SYSTEM

- Simple way of online booking is produced for all types of users.
- All types of cabs and locations can be booked through online.
- Detail views of locations are provided to users.
- Admin can view each and every user's booking individually.
- User registration form and their details will be secured.
- Manually Processes are reduced.

ADVANTAGES OF PROPOSED SYSTEM

2.3 ANALYSING THE SYSTEM

The Nine types of modules are

- Registration
- Admin Login
- Add Cabs
- Drivers Details
- Client Details
- User Login
- Search Cabs
- Booking Cabs
- Booking Cancel

Registration

This Registration Pages is used only for users for new bookings and new customers. If the Email ID is used once and for another registration they need to try new Email ID.

Admin Login

This Admin Login Page is only used for admins and the users cannot try their username and password. It won't works even they try.

Add Cabs

This page is only for users and in this page admin will register the new driver's vehicle name and the locations they need to travel with their contact number.

Drivers Details

This page is only for the admin and this page they can view the complete details about the driver's location, contact numbers and the vehicle name and type.

Client Details

This page is only for admin they can view the individual user booking and the located area they need the cab.

User Login

This page is only for users they can login their page using their username and the password with they had registered their details.

Search Cab

This page is used for users in this page they can search their cab by using from address and to address locations the need to travel.

Booking Cab

This page is for user after finding the cab they need to book their cab in this form.

Cancel Booking

This page is only for users they can delete the booking if they don't need the booking.

SYSTEM REQUIREMENT AND SPECIFICATIONS

SYSTEM REQUIREMENT AND SPECIFICATIONS

SYSTEM SPECIFICATION

3.1 WORKING ENVIRONMENT

HARDWARE SPECIFICATION

Processor	: Intel Pentium
RAM	: 4GB
Speed	: 2.33 GHz
Hard Disk Driver	: 500 GB
Compact Disk Driver	: LG DVD Writer 22X
Monitor	: 18.5" Color LED Monitor
Keyboard	: 108 keys

SOFTWARE SPECIFICATION

Operating System	: windows 7
Environment	: visual studio 2012
.NET FRAMEWORK	: version 4.0
Front-End	: Active Server Page(ASP.NET)
Back-End	: MS Access

3.2 ABOUT SOFTWARE

ASP.NET

VB.NET stands for Visual Basic.NET, and it is a computer programming language developed by Microsoft. It was first released in 2002 to replace Visual Basic 6. VB.NET is an object-oriented programming language. This means that it supports the features of object oriented programming which include encapsulation, polymorphism, abstraction, and inheritance.

Visual Basic ASP.NET runs on the .NET framework, which means that it has full access to the .NET libraries. It is a very productive tool for rapid creation of a wide range of Web, Windows, Office, and Mobile applications that have been built on the .NET framework.

The language was designed in such a way that it is easy to understand to both novice and advanced programmers. Since VB.NET relies on the .NET framework, programs written in the language run with much reliability and scalability. With VB.NET, you can create applications that are fully object-oriented, similar to the ones created in other languages like C++, Java, or C#.

Programs written in VB.NET can also interoperate well with programs written in Visual C++, Visual C#, and Visual J#. VB.NET treats everything as an object. It is true that VB.NET is an evolved version of Visual Basic 6, but it's not compatible with it. If you write your code in Visual Basic 6, you cannot compile it under VB.NET.

HISTORY OF VB.NET

VB.NET is a multi-paradigm programming language developed by Microsoft on the .NET framework. It was launched in 2002 as a successor to the Visual Basic language. This was the first version of VB.NET (VB.NET 7.0) and it relied on .NET version 1.0. In 2003, the second version of VB.NET, VB.NET 7.1, was released. In 2005, VB.NET 8.0 was released. The .NET core portion was dropped from its name so as to distinguish it from the classical Visual Basic language. This version was named Visual Basic 2005. This version came with many features since Microsoft wanted this language to be used for rapid application developers. They also wanted to make it different from C# language. Some of the features introduced by

this version of VB.NET included partial classes, generics, nullable types, operator overloading, and unsigned integer support. This version also saw the introduction of the Is Not operator.

VB.NET FEATURES

VB.NET comes loaded with numerous features that have made it a popular programming language amongst programmers worldwide. These features include the following:

- ❖ VB.NET is not case sensitive like other languages such as C++ and Java.
- ❖ It is an object-oriented programming language. It treats everything as an object.
- ❖ Automatic code formatting, XML designer, improved object browser etc.
- ❖ Garbage collection is automated.
- ❖ Support for Boolean conditions for decision making.
- ❖ Simple multithreading, allowing your apps to deal with multiple tasks simultaneously.
- ❖ Simple generics.
- ❖ A standard library

ADVANTAGES OF VB.NET

The following are the pros/benefits you will enjoy for coding in VB.NET:

- ❖ Your code will be formatted automatically.
- ❖ You will use object-oriented constructs to create an enterprise-class code.
- ❖ You can create web applications with modern features like performance counters, event logs, and file system.
- ❖ You can create your web forms with much ease through the visual forms designer.
- ❖ You will also enjoy drag and drop capability to replace any elements that you may need.
- ❖ You can connect your applications to other applications created.

DISADVANTAGES OF VB.NET

Below are some of the drawbacks/cons associated with VB.NET:

- ❖ VB.NET cannot handle pointers directly. This is a significant disadvantage since pointers are much necessary for programming. Any additional coding will lead to many CPU cycles, requiring more processing time. Your application will become slow.
- ❖ VB.NET is easy to learn. This has led to a large talent pool. Hence, it may be challenging to secure a job as a VB.NET programmer.
- ❖ VB.NET is easy to learn. This has led to a large talent pool. Hence, it may be challenging to secure a job as a VB.NET programmer.

STANDARD CONTROLS USED

POINTER:

It is just a pointer. If we drag any other controls does not create any control on form. In other word we can say, we select it for to ignore any other selected control.

BUTTON:

Very widely used control. Writing some lines about this control is very tinny thing. It is used everywhere like have to register, admin, user, book, update, cancel, exit etc.....

DROPDOWNLIST:

This control is used to select any one item from list of items. We cannot select multiple item from drop down list. It has also database building property.

TEXTBOX:

This control is used to enter information for processes. Like if we are wishing to create a web page where user can enter his username and password, surely take the advantage of textbox here. We can change the Textbox, TEXTMODE PROPERTY by single line, multiline and password.

IMAGE:

This control is used to link any other page. Like if you open any community website and have look your image means they have used image control. Using image URL property we can change the image.

MICROSOFT ACCESS INTRODUCTION

Microsoft Access is a Database Management System (DBMS) from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools. It is a member of the Microsoft Office suite of applications, included in the professional and higher editions.

- ❖ Microsoft Access is just one part of Microsoft's overall data management product strategy.
- ❖ It stores data in its own format based on the Access Jet Database Engine.
- ❖ Like relational databases, Microsoft Access also allows you to link related information easily. For example, customer and order data. However, Access 2013 also complements other database products because it has several powerful connectivity features.
- ❖ It can also import or link directly to data stored in other applications and databases.
- ❖ As its name implies, Access can work directly with data from other sources, including many popular PC database programs, with many SQL (Structured Query Language) databases on the desktop, on servers, on minicomputers, or on mainframes, and with data stored on Internet or intranet web servers.
- ❖ Access can also understand and use a wide variety of other data formats, including many other database file structures.
- ❖ You can export data to and import data from word processing files, spreadsheets, or database files directly.
- ❖ Access can work with most popular databases that support the Open Database Connectivity (ODBC) standard, including SQL Server, Oracle, and DB2.
- ❖ Software developers can use Microsoft Access to develop application software.

Microsoft Access stores information which is called a database. To use MS Access, you will need to follow these four steps:

- ❖ **Database Creation** – Create your Microsoft Access database and specify what kind of data you will be storing.
- ❖ **Data Input** – after your database is created, the data of every business day can be entered into the Access database.
- ❖ **Query** – this is a fancy term to basically describe the process of retrieving information from the database.
- ❖ **Report** (optional) – Information from the database is organized in a nice presentation that can be printed in an Access Report.

SYSTEM DESIGN

TABLE DESIGN

4.1 TABLE DESIGN

Registration:

addcabs register	
Field Name	Data Type
USERNAME	Text
PASSWORD	Text
GENDER	Text
MOBILENUM	Text
MAILID	Text
ADDRESS	Text

Add Cabs:

addcabs	
Field Name	Data Type
DRIVERNAME	Text
VEHICLENAME	Text
VEHICLETYPE	Text
CONTACTNUMBER	Text
FROM	Text
TO	Text
BOOKED	Text

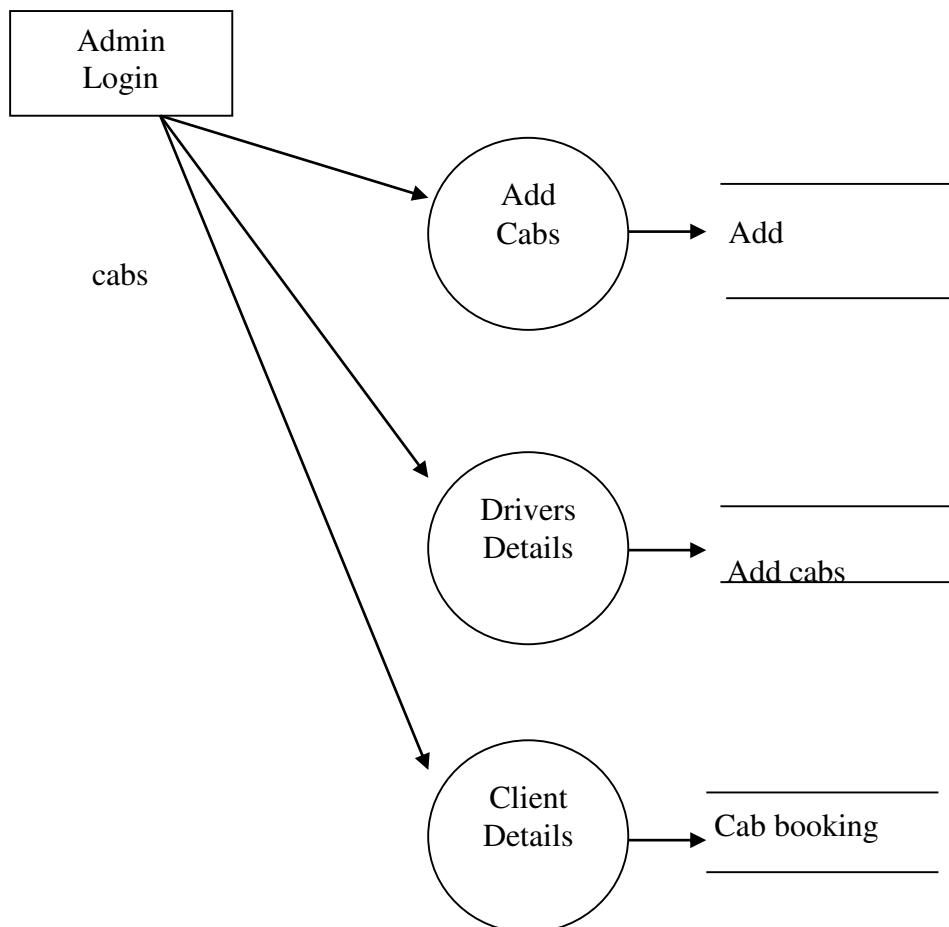
Cab Booking:

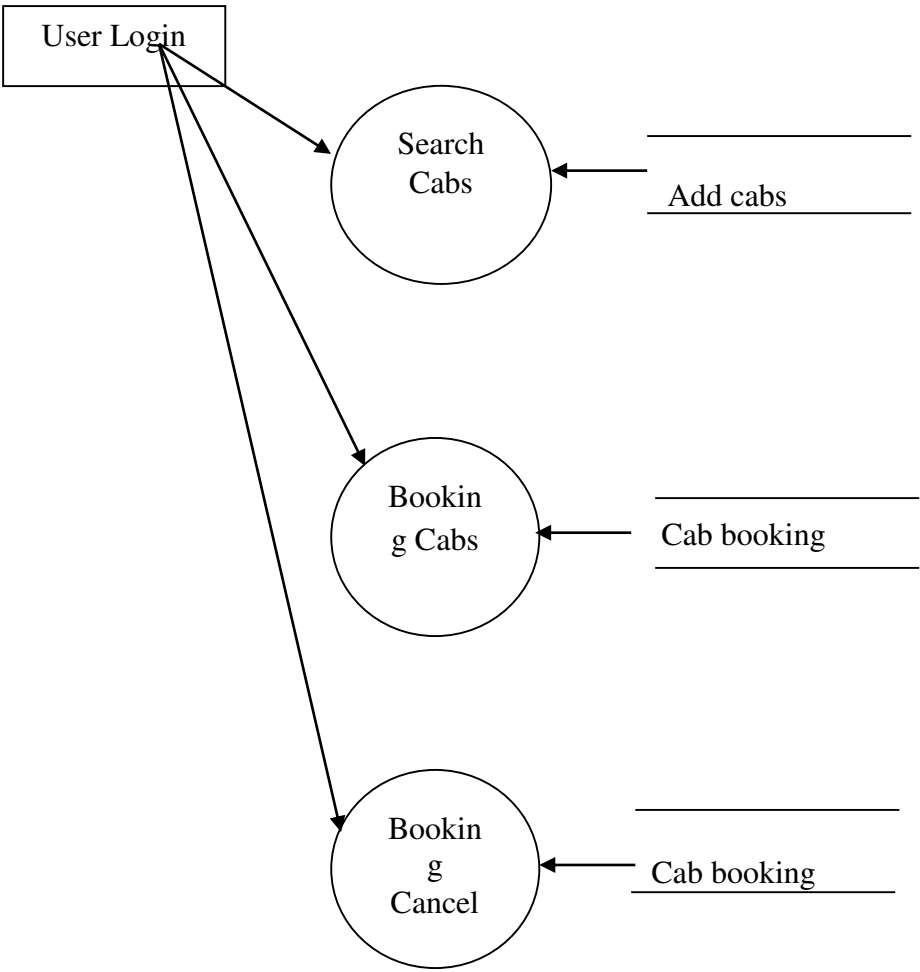
cab_booking	
Field Name	Data Type
DRIVERNAME	Text
VEHICLENAME	Text
VEHICLETYPE	Text
CONTACTNUMBER	Text
FROM_F	Text
TO_T	Text
DATE_ON	Text
TIME_ON	Text
USERNAME	Text

DATA FLOW DIAGRAM

4.2 DATA FLOW DIAGRAM

The data flow diagram (DFD) is a graphical tool used for expressing system requirements in a graphical form. The DFD also known as the “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. Thus DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. The DFD consists of series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system.





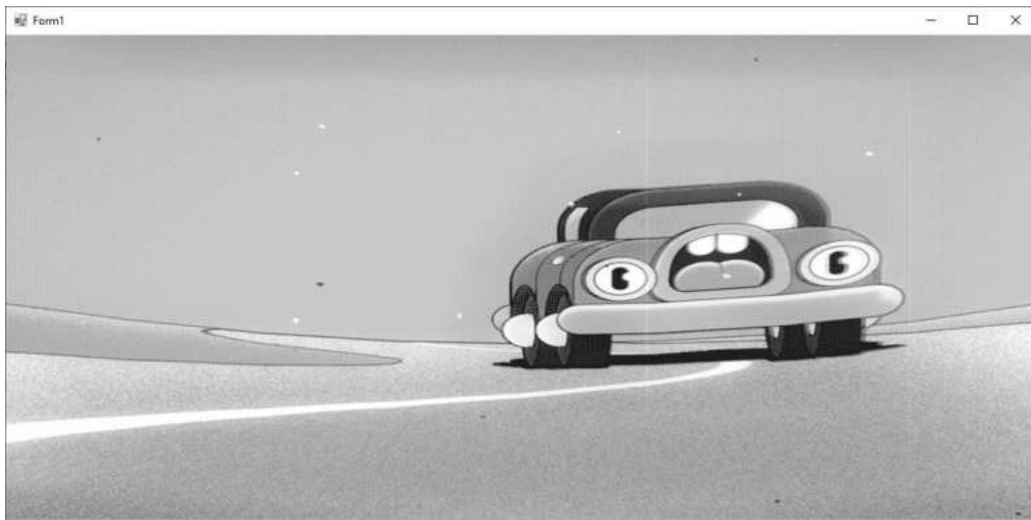
FLOW CHART

4.3 FLOW CHART

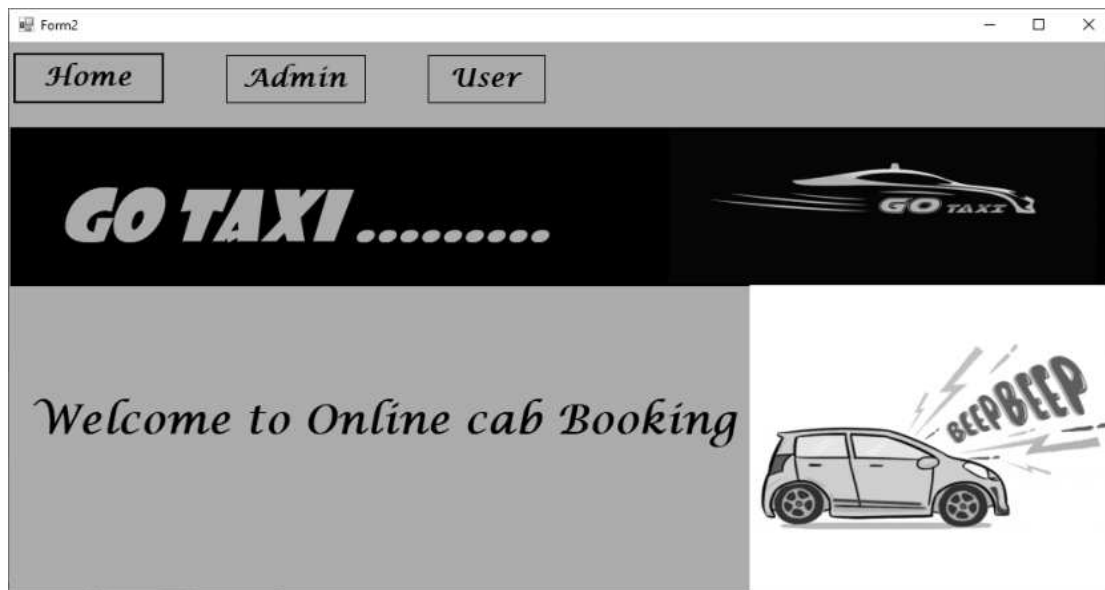
FORM DESIGN

4.4 FORM DESIGN

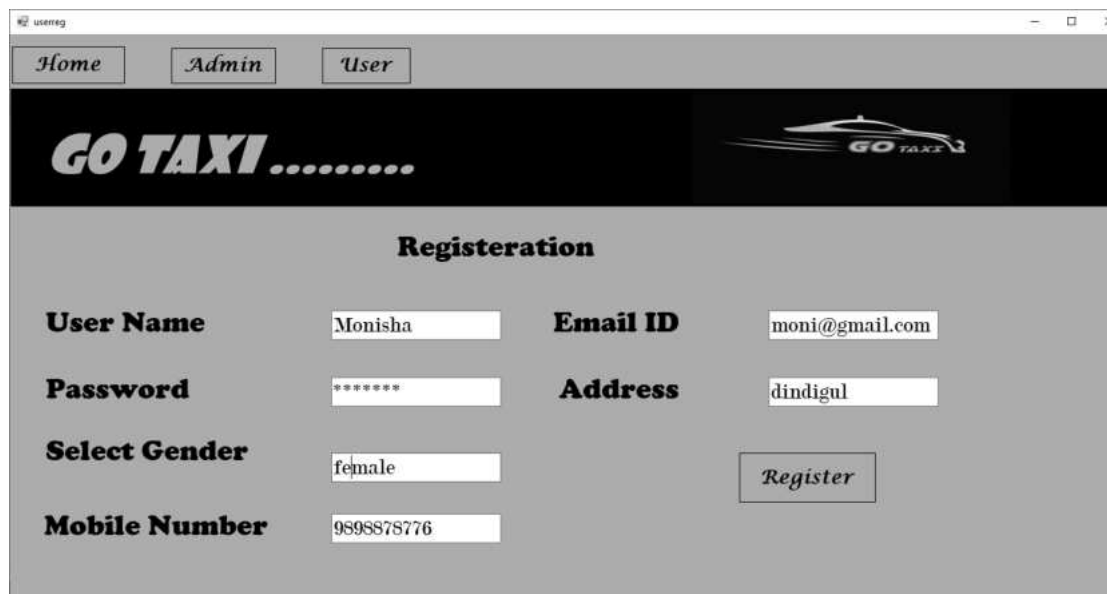
Main From



Home Form



Registration Form



The screenshot shows a web browser window titled 'ucereg'. The navigation bar contains three buttons: 'Home', 'Admin', and 'User'. The main header features the 'GO TAXI.....' logo on the left and a 'GO TAXI 3' logo with a car icon on the right. The central section is titled 'Registration' and contains a form with the following fields and values:

Field	Value
User Name	Monisha
Email ID	moni@gmail.com
Password	*****
Address	dindigul
Select Gender	female
Mobile Number	9898878776

A 'Register' button is located to the right of the form fields.



A dialog box titled 'Go Taxi' with a close button (X) in the top right corner. The message inside the dialog reads: 'Your Registration is Completed'. At the bottom right of the dialog is an 'OK' button.

Client Login Form

userlog

HomeAdminUser

GO TAXI.....



User Login

User Name

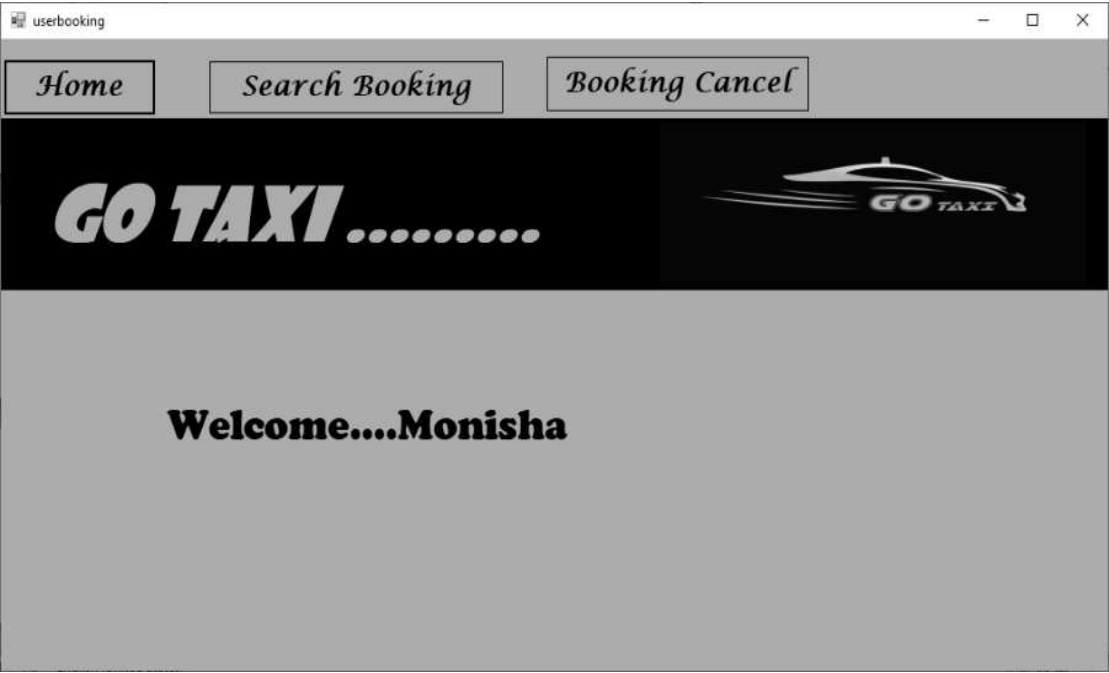
Monisha

Password

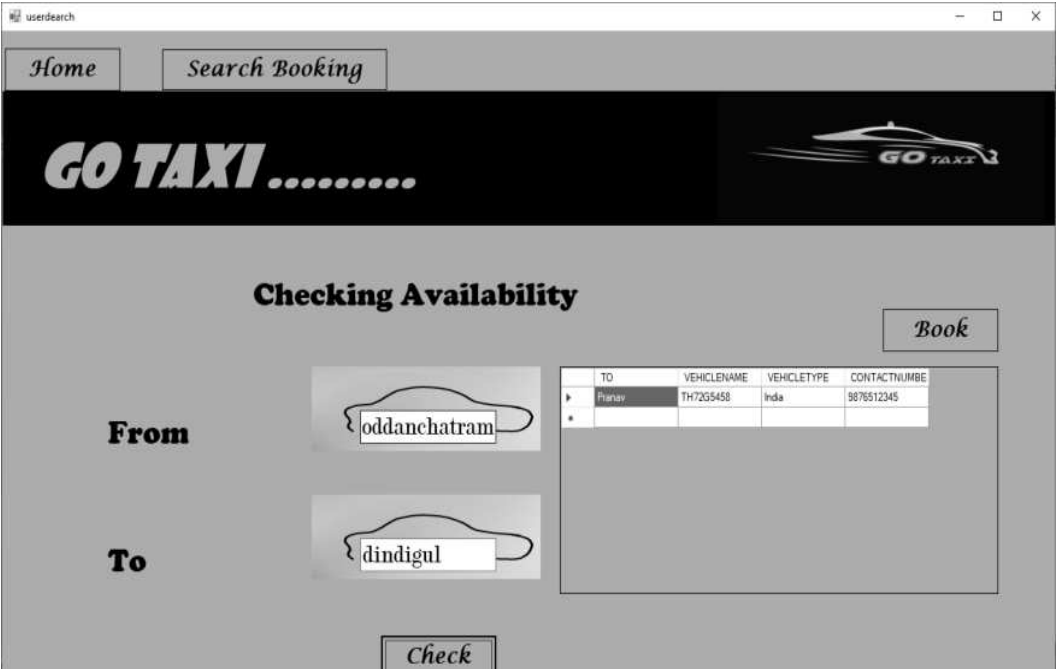
Login

[New User?..... Click Here to Register.....](#)

Client Home



Search Cabs



Booking Cabs

cabbookinguser

Home Search Booking

GO TAXI.....

Driver Name

Vehicle Name

Vehicle Type

Contact Number 9876512345 **To** dindigul

Confirm

Go Taxi X

Your Booking is Completed

OK

3 September 2020

10:00 AM

oddanchatram

Booking Cancel

bookingcancel

Home Search Booking Back

GO TAXI.....

Cancel Booking

User Name Monisha

From oddanchatram

To dindigul

Cancel

Admin Login

adminlog

Home Admin User

Go Taxi

Welcome Admin

OK

Name

Password

Login

Cabs

Add

addcab

Home Admin Drivers Users Booking

GO TAXI.....

Add Cabs

Driver Name	<input type="text" value="Pranav"/>	From	<input type="text" value="oddanchatram"/>
Vehicle Name	<input type="text" value="TH72G5458"/>	To	<input type="text" value="dindigul"/>
Vehicle Type	<input type="text" value="India"/>	<input type="button" value="Add"/>	
Contact Number	<input type="text" value="9876512345"/>		



Drivers Details

newdriver

Home Admin Back

GO TAXI.....

Driver Name **From**

Vehicle Name **To**

Vehicle Type **Contact Number**

	DRIVERNAME	VEHICLENAME	VEHICLETYPE	CONTACTNUMBER	FROM	TO
▶	Pranav	TN52A5946	India	9876543210	tsayanvillai	sathankulam
	karthi	TH21V3256	India	9876598765	sathankulam	tsayanvillai
	subha	TH21Q5454	India	3213213211	udankudi	sathankulam
	shvi	TN6F6547	India	9876983215	truvanthapuram	Nagarkovil

Client Detail

[illegible]

CODE DESIGN

4.5 CODE DESIGN

SOURCE CODE

Main Form

```
Public Class Form1
```

```
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    End Sub
```

```
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click  
        Form2.Show()
```

```
    End Sub
```

```
End Class
```

Home From

```
Public Class Form2
```

```
    Private Sub Form2_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    End Sub
```

```
    Private Sub Panel1_Paint(sender As Object, e As PaintEventArgs) Handles Panel1.Paint
```

```
    End Sub
```

```
    Private Sub Label1_Click(sender As Object, e As EventArgs)
```

```
    End Sub
```

```
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click  
        Me.Show()
```

```
    End Sub
```

```
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click  
        adminlog.Show()  
        Me.Hide()
```

```
    End Sub
```

```
    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
```



```

        userlog.Show()
        Me.Hide()
    End Sub
End Class

```

Register From

```

Imports System.Data.OleDb
Public Class userreg
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=D:\cab_booking\cab_booking\cabdetails.accdb")
    Dim cmd As New OleDbCommand

    Private Sub Label5_Click(sender As Object, e As EventArgs) Handles Label5.Click

    End Sub

    Private Sub Label6_Click(sender As Object, e As EventArgs)

    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        userlog.Show()
        Me.Hide()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        adminlog.Show()
        Me.Hide()
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Form2.Show()
        Me.Hide()
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        con.Open()
        Dim cmd As OleDbCommand = New OleDbCommand("select *from[register]
where[MAILID]='" & TextBox5.Text & "'", con)

        Dim dr As OleDbDataReader = cmd.ExecuteReader
        Dim userfound As Boolean = False
        Dim firstname As String
        While dr.Read
            userfound = True
            firstname = dr("MAILID").ToString
        End While
    End Sub
End Class

```

```

con.Close()
If userfound = True Then
    MessageBox.Show("This email id is already exist please try another.....")
    'Me.Close()
    ElseIf TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or
    TextBox4.Text = "" Or TextBox5.Text = "" Or TextBox6.Text = "" Then
        MessageBox.Show("Please Enter the Details Completely..", "Go Taxi",
        MessageBoxButtons.OK)
    Else
        con.Open()
        cmd.Connection = con
        cmd.CommandText = "insert into register values('" + TextBox1.Text + "','" +
        TextBox2.Text + "','" + TextBox3.Text + "','" + TextBox4.Text + "','" + TextBox5.Text + "','"
        + TextBox6.Text + "')"

        MessageBox.Show("Your Registration is Completed", "Go Taxi",
        MessageBoxButtons.OK)
        cmd.ExecuteNonQuery()
        'con.Close()
        userlog.Show()
        Me.Hide()
    End If

End Sub

Private Sub userreg_Load(sender As Object, e As EventArgs) Handles MyBase.Load

End Sub
End Class

```

User Login Form

```

Imports System.Data.OleDb
Public Class userlog
    Dim myconnection As New
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=D:\cab_booking\cab_booking\cabdetails.accdb")
    Dim cmd As New OleDbCommand
    Private Sub Label2_Click(sender As Object, e As EventArgs) Handles Label2.Click

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Form2.Show()
    Me.Hide()

End Sub

```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    adminlog.Show()
    Me.Hide()
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    Me.Show()

End Sub
```

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
    Dim cmd As OleDbCommand = New OleDbCommand("select *from[register]
where[USERNAME]='" & TextBox1.Text & "' and [PASSWORD]='" & TextBox2.Text &
""", myconnection)
    myconnection.Open()
    Dim dr As OleDbDataReader = cmd.ExecuteReader
    Dim userfound As Boolean = False
    Dim firstname As String = ""
    Dim lastname As String = ""
    While dr.Read
        userfound = True
        firstname = dr("USERNAME").ToString
        lastname = dr("PASSWORD").ToString
        ' Label5.Text = firstname

    End While

myconnection.Close()
    If userfound = True Then
        Dim uh As New userhome
        uh.Label2.Text = TextBox1.Text
        userhome.Show()

        Me.Hide()

    Else
        MessageBox.Show("Incorrect User Name or Password")
    End If

End Sub
```

```
Private Sub LinkLabel1_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
    userreg.Show()
    Me.Hide()
End Sub
```

```
Private Sub userlog_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
End Sub
```

```
End Class
```

Search Cabs Form

```
Imports System.Data.OleDb
```

```
Public Class usersearch
```

```
    Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data  
Source=C:\Users\Mahima\Downloads\cab_booking-20210213T094254Z-  
001\cab_booking\cab_booking\cabdetails.accdb")
```

```
    Dim cmd As New OleDbCommand
```

```
    Dim ds As New DataSet
```

```
Private Sub Button5_Click(sender As Object, e As EventArgs)
```

```
End Sub
```

```
Private Sub Button6_Click(sender As Object, e As EventArgs)
```

```
End Sub
```

```
Private Sub usersearch_Load(sender As Object, e As EventArgs) Handles  
MyBase.Load
```

```
    If ComboBox1.Text = "" Then
```

```
        ComboBox2.Enabled = False
```

```
        ComboBox3.Enabled = False
```

```
    End If
```

```
    'con.Open()
```

```
    'Dim cmd As OleDbCommand = New OleDbCommand("select *from[addcabs]  
where[WOKING_AREA]='" & TextBox1.Text & "'", con)
```

```
    'Dim dr As OleDbDataReader = cmd.ExecuteReader
```

```
    'Dim userfound As Boolean = False
```

```
    'While dr.Read
```

```
    '    userfound = True
```

```
    '    cnum = dr("WOKING_AREA").ToString
```

```
    'End While
```

```
    'con.Close()
```

```
End Sub
```

```

Private Sub Button3_Click(sender As Object, e As EventArgs)

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    userhome.Show()
    Me.Hide()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
    Me.Show()

End Sub

Private Sub Button3_Click_1(sender As Object, e As EventArgs) Handles
Button3.Click
    cmd.Connection = con
    con.Open()
    'cmd.CommandText = " insert into cab_booking values('" +
ComboBox1.Text + "'"")
    cmd.CommandText = "insert into cab_booking values('" +
TextBox8.Text + "','" + TextBox10.Text + "','" + ComboBox1.Text + "','" +
TextBox1.Text + "','" + TextBox2.Text + "','" + TextBox3.Text + "','" +
TextBox6.Text + "','" + ComboBox2.Text + "','" + ComboBox3.Text + "','" +
DateTimePicker1.Text + "','" + TextBox4.Text + "','" + TextBox5.Text +
"')"
    cmd.ExecuteNonQuery()
    cmd.CommandText = "insert into Admin values('" + ComboBox1.Text +
"', '" + TextBox1.Text + "','" + TextBox8.Text + "','" + ComboBox2.Text +
"', '" + ComboBox3.Text + "','" + DateTimePicker1.Text + "','" +
TextBox4.Text + "','" + TextBox9.Text + "'"")
    cmd.ExecuteNonQuery()

    cmd.CommandText = "update addcabs set BOOKED ='" + TextBox7.Text +
"'Where WOKING_AREA='" + ComboBox1.Text + "'and DRIVERNAME='" +
TextBox1.Text + "'and VEHICLENAME='" + TextBox2.Text + "'and
VEHICLETYPE='" + TextBox3.Text + "'and CONTACTNUMBER='" + TextBox6.Text +
"'"
    cmd.ExecuteNonQuery()
    MessageBox.Show("Your Cab has Booked... :)")
    con.Close()
    TextBox8.Text = ""
    TextBox10.Text = ""
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox6.Text = ""

```

```
ComboBox2.Text = ""
ComboBox3.Text = ""
TextBox4.Text = ""
TextBox5.Text = ""
```

End Sub

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs)
```

End Sub

```
Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ComboBox1.SelectedIndexChanged
    If ComboBox1.Text = "Thoothukudi" Then
        ComboBox2.Items.Add("Tharavaikulam")
        ComboBox2.Items.Add("Ambalaseri")
        ComboBox2.Items.Add("Governagiri")
        ComboBox2.Items.Add("Korkai")
        ComboBox2.Items.Add("Manapad")
        ComboBox2.Items.Add("Mullakadu")
        ComboBox2.Items.Add("Punnaikayal")
        ComboBox2.Items.Add("Subramaniyapuram")
        ComboBox2.Items.Add("Thirukkooloor")
        ComboBox2.Items.Add("sathankulam")
        ComboBox3.Items.Add("Tharavaikulam")
        ComboBox3.Items.Add("Ambalaseri")
        ComboBox3.Items.Add("Governagiri")
        ComboBox3.Items.Add("Korkai")
        ComboBox3.Items.Add("Manapad")
        ComboBox3.Items.Add("Mullakadu")
        ComboBox3.Items.Add("Punnaikayal")
        ComboBox3.Items.Add("Subramaniyapuram")
        ComboBox3.Items.Add("Thirukkooloor")
        ComboBox3.Items.Add("sathankulam")
    Else
        ComboBox2.Items.Add("Thenkulam")
        ComboBox2.Items.Add("Abishekapatti")
        ComboBox2.Items.Add("Alangaraperi")
        ComboBox2.Items.Add("Kalakudi")
        ComboBox2.Items.Add("Ettankulam")
        ComboBox2.Items.Add("Karungadu")
        ComboBox2.Items.Add("Karuvanallur")
        ComboBox2.Items.Add("Manur")
        ComboBox2.Items.Add("Palavoor")
        ComboBox2.Items.Add("suttamalli")
        ComboBox3.Items.Add("Thenkulam")
        ComboBox3.Items.Add("Abishekapatti")
        ComboBox3.Items.Add("Alangaraperi")
        ComboBox3.Items.Add("Kalakudi")
        ComboBox3.Items.Add("Ettankulam")
        ComboBox3.Items.Add("Karungadu")
        ComboBox3.Items.Add("Karuvanallur")
        ComboBox3.Items.Add("Manur")
    End If
End Sub
```

```

        ComboBox3.Items.Add("Palavoor")
        ComboBox3.Items.Add("suttamalli")

    End If
    ComboBox2.Enabled = True

End Sub

Private Sub ComboBox1_SelectedValueChanged(sender As Object, e As
EventArgs) Handles ComboBox1.SelectedValueChanged
    ComboBox2.Text = ""
    ComboBox3.Text = ""
    ComboBox2.Items.Clear()
    ComboBox3.Items.Clear()
End Sub

Private Sub ComboBox1_TextChanged(sender As Object, e As EventArgs)
Handles ComboBox1.TextChanged
    con.Open()
    Dim cmd As OleDbCommand = New OleDbCommand("select *from[addcabs]
where[WOKING_AREA]='" & ComboBox1.Text & "'and [BOOKED]='" & TextBox9.Text
& "'", con)

    Dim dr As OleDbDataReader = cmd.ExecuteReader
    Dim userfound As Boolean = False
    'Dim dname As String
    'Dim vname As String
    'Dim vtype As String
    'Dim cnum As String
    While dr.Read
        userfound = True
        TextBox1.Text = dr("DRIVERNAME").ToString
        TextBox2.Text = dr("VEHICLENAME").ToString
        TextBox3.Text = dr("VEHICLETYPE").ToString
        TextBox6.Text = dr("CONTACTNUMBER").ToString
    End While
    con.Close()
    If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = ""
Or TextBox6.Text = "" Then
        MessageBox.Show("Sorry no cab in available...")
    End If
End Sub

Private Sub Label1_Click(sender As Object, e As EventArgs) Handles
Label1.Click

End Sub

Private Sub TextBox5_TextChanged(sender As Object, e As EventArgs)
Handles TextBox5.TextChanged

End Sub

```

```

        Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ComboBox2.SelectedIndexChanged
            ComboBox3.Enabled = True

        End Sub
End Class

```

Booking Cabs Form

```

Imports System.Data.OleDb
Public Class cabbookinguser
    Dim myconnection As OleDbConnection = New
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=D:\cab_booking\cab_booking\cabdetails.accdb")
    Dim cmd As New OleDbCommand
    Private Sub Label12_Click(sender As Object, e As EventArgs) Handles Label12.Click
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Me.Show()
    End Sub

    Private Sub cabbookinguser_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Label11.Text = usersearch.TextBox1.Text
        Label12.Text = usersearch.TextBox2.Text
        TextBox2.Text = userlog.TextBox1.Text

        myconnection.Open()

        Dim cmd As OleDbCommand = New OleDbCommand("select
DRIVERNAME,VEHICLENAME,VEHICLETYPE,CONTACTNUMBER from[addcabs]
where[FROM]=''" & Label11.Text & "'and [TO]=''" & Label12.Text & "'and [BOOKED]=''"
& "Not Booked" & "'", myconnection)
        Dim dr As OleDbDataReader = cmd.ExecuteReader
        Dim userfound As Boolean = False

        While dr.Read
            userfound = True
            Label9.Text = dr("DRIVERNAME").ToString
            Label14.Text = dr("VEHICLENAME").ToString
            Label10.Text = dr("VEHICLETYPE").ToString
            Label16.Text = dr("CONTACTNUMBER").ToString
        End While
        myconnection.Close()

    End Sub

    Private Sub Label16_Click(sender As Object, e As EventArgs) Handles Label16.Click

```


End Sub

```
Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
    myconnection.Open()
    cmd.Connection = myconnection
    cmd.CommandText = "insert into cab_booking values('" + Label9.Text + "','" +
Label14.Text + "','" + Label10.Text + "','" + Label16.Text + "','" + Label11.Text + "','" +
Label12.Text + "','" + DateTimePicker1.Text + "','" + TextBox1.Text + "','" + TextBox2.Text
+ "')"
    cmd.ExecuteNonQuery()
    MessageBox.Show("Your Booking is Completed", "Go Taxi", MessageBoxButtons.OK)
    cmd.CommandText = "(update addcabs set BOOKED='" + "Booked" + "' where
DRIVERNAME='" + Label9.Text + "')"
    cmd.ExecuteNonQuery()
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    usersearch.Show()
    Me.Hide()
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs)
    ' mybooking.Show()
    ' Me.Hide()
End Sub
End Class
```

Booking Cancel Form

```
Imports System.Data.OleDb
Public Class BookingCancel
    Dim con As OleDbConnection = New
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=D:\cab_booking\cab_booking\cabdetails.accdb")
    Dim cmd As New OleDbCommand

    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click

        con.Open()
        cmd.Connection = con
        cmd.CommandText = ("delete
DRIVERNAME,VEHICLENAME,VEHICLETYPE,CONTACTNUMBER,FROM_F,TO_TDATE_
ON,TIME_ON,USERNAME from[cab_booking] where [USERNAME]='" &

TextBox1.Text & "'and [FROM_F]='" & TextBox2.Text & "'and [TO_T]='" & TextBox3.Text & "'")
        MessageBox.Show("Your Booking is Cancelled")
    End Sub
End Class
```

```

        cmd.ExecuteNonQuery()
    End Sub

    Private Sub BookingCancel_Load(sender As Object, e As EventArgs) Handles MyBase.Load

    End Sub
End Class

```

Admin Login Form

```

Public Class adminlog

    Private Sub adminlog_Load(sender As Object, e As EventArgs) Handles MyBase.Load

    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click

        If UsernameTextBox.Text = "admin" And PasswordTextBox.Text = "admin" Then
            MessageBox.Show("Welcome Admin", "Go Taxi", MessageBoxButtons.OK)
            addcab.Show()
            Me.Hide()
        Else
            MessageBox.Show("Incorrect User Name or Password")
        End If

    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        Me.Show()
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

        Form2.Show()
        Me.Hide()
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        userlog.Show()
        Me.Hide()
    End Sub
End Class

```

Add Cabs Form

```
Imports System.Data.OleDb
Public Class addcab
    Dim read As String
    Dim connstring As String
    Dim datafile As String
    Dim myconnection As OleDbConnection = New
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=D:\cab_booking\cab_booking\cabdetails.accdb")
    Dim cmd As New OleDbCommand

    Private Sub Label6_Click(sender As Object, e As EventArgs) Handles Label6.Click

    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Form2.Show()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        adminlog.Show()
        Me.Hide()
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs)
        userlog.Show()

        Me.Hide()
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        cmd.Connection = myconnection
        myconnection.Open()

cmd.CommandText = "insert into addcabs values('" + TextBox1.Text + "','" + TextBox2.Text
+ "','" + TextBox3.Text + "','" + TextBox4.Text + "','" + TextBox5.Text + "','" +
TextBox6.Text + "','" + TextBox7.Text + "')"

        MessageBox.Show("New driver details are added", "Go Taxi",
MessageBoxButtons.OK)

        cmd.ExecuteNonQuery()
        ' myconnection.Close()
        newdriver.Show()
        Me.Hide()
    End Sub
End Class
```

End Sub

Private Sub addcab_Load(sender As Object, e As EventArgs)

End Sub

Private Sub addcab_Load_1(sender As Object, e As EventArgs) Handles MyBase.Load

End Sub

Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click

newdriver.Show()

Me.Hide()

End Sub

Private Sub Button3_Click_1(sender As Object, e As EventArgs) Handles Button3.Click

detailuser.Show()

Me.Hide()

End Sub

End Class

Driver Detail Form

Imports System.Data.OleDb

Public Class newdriver

Dim read As String

Dim connstring As String

Dim datafile As String

Dim myconnection As OleDbConnection = New
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=D:\cab_booking\cab_booking\cabdetails.accdb")

Dim cmd As New OleDbCommand

Private Sub newdriver_Load(sender As Object, e As EventArgs) Handles MyBase.Load

TextBox1.Text = addcab.TextBox1.Text

TextBox2.Text = addcab.TextBox2.Text

TextBox3.Text = addcab.TextBox3.Text

TextBox4.Text = addcab.TextBox4.Text

TextBox5.Text = addcab.TextBox5.Text

TextBox6.Text = addcab.TextBox6.Text

'TODO: This line of code loads data into the 'CabdetailsDataSet.addcabs' table. You can
move, or remove it, as needed.

Me.AddcabsTableAdapter.Fill(Me.CabdetailsDataSet.addcabs)

myconnection.Open()

End Sub

Private Sub datagrideshow()

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

Form2.Show()

Me.Hide()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

adminlog.Show()

Me.Hide()

End Sub

Private Sub Button3_Click(sender As Object, e As EventArgs)

userlog.Show()

Me.Hide()

End Sub

Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click

addcab.Show()

Me.Hide()

End Sub

End Class

FUTURE ENHANCEMENT

FUTURE ENHANCEMENT

- ❖ This will be extended for all other company Cabs also.
- ❖ In future data will be stored in SQL Database.
- ❖ Users review will be collected.
- ❖ If any doubt about the websites or our about our cab user may ask any questions and also they get the answers.
- ❖ Finger print access methods will be provided.
- ❖ Online Transaction will be provided.

CONCLUSION

CONCLUSION

The project entitled “**Go Taxi Cab Booking System**” is developed in ASP.NET. This is the rising age of new technology. In the busy world, today any time for any manual process which is more time consuming. Now a day computer has replaced every field. This project has designed to be user-friendly and allow the user to access the services effectively and also easily.

Thus conclude that the software works for a valid input and finally the result of this project is error free, because there is no manual data handling.

BIBLIOGRAPHY

BIBLIOGRAPHY

Reference Books

1. The Complete Reference ASP.NET by Matthew MacDonald.

-Tata MC GRAW Hill Publication

2. The Complete Reference OFFICE 2007 by Stephen L. Nelson.

- Tata MC GRAW Hill Publication

Website

1. www.aspnippets.com
2. <https://sourceforge.net>
3. <https://kashipara.net>
4. www.enggroom.com

OUR CITY

Project Submitted to the Department Of Computer Science

ST. MARY'S COLLEGE(Autonomous), THOOTHUKUDI

Affiliated to

Manonmaniam Sundaranar University, Tirunelveli

In partial fulfilment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE

Done by

R. Jerina Shiny **18AUCS11**

S. Kaushika **18AUCS16**

S.B. Lavanya **18AUCS17**

L.Luxcima **18AUCS18**

M.Suganthi Mary **18AUCS46**

Under the Supervision and Guidance of

Mrs. S.Andrews Felcita M.Sc.,MBA(HRM)

(Assistant Professor of Computer Science)



DEPARTMENT OF COMPUTER SCIENCE

ST. MARY'S COLLEGE (AUTONOMOUS)

Reaccredited with "A+" Grade by NAAC

THOOTHUKUDI

2018-2021

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

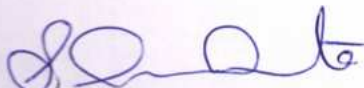
This is certify that the project entitled

OUR CITY

Submitted by

R.Jerina Shiny	-	18AUCS11
S.Kaushika	-	18AUCS16
S.B.Lavanya	-	18AUCS17
L.Luxima	-	18AUCS18
M.Suganthi Mary	-	18AUCS46

In partial fulfilment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.


Guide

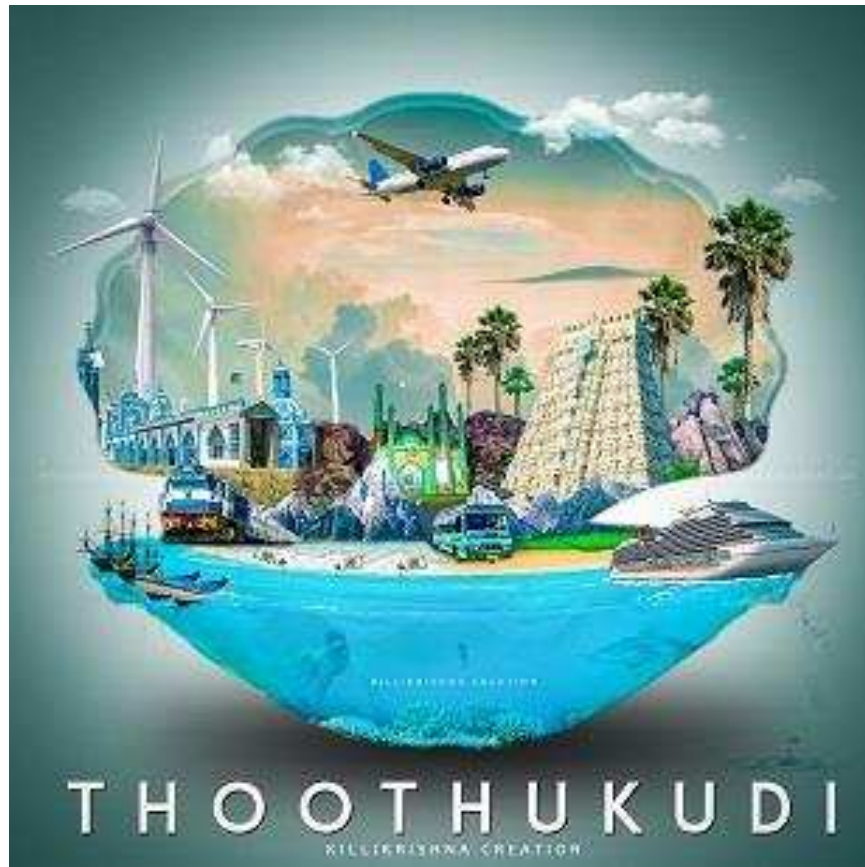

Head of the Department

Submitted for the examination held on 15.04.2021.

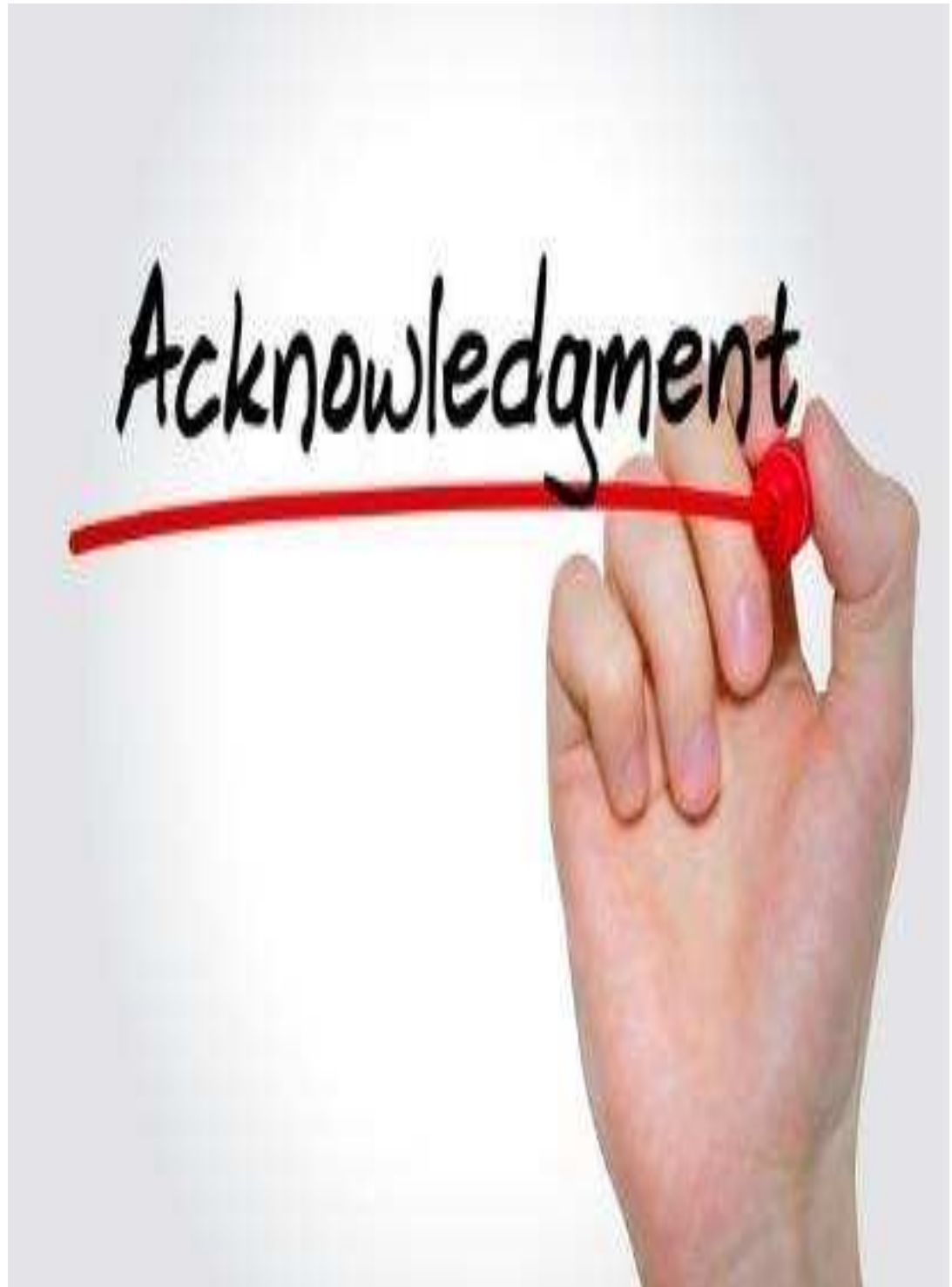

Examiner


Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

OUR CITY



Thoothukudi is known as **“Pearl City”**
&
“Port City”



ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and shower of blessing for having rendered us the strength and support to finish our project successfully.

We are infinitely indebted to our college principal Dr.Sr.A.S.J.Lucia Rose M.Sc.,M.Phil.,PGDCA,Ph.D. for providing us necessary infrastructure to undertake this project and work successfully.

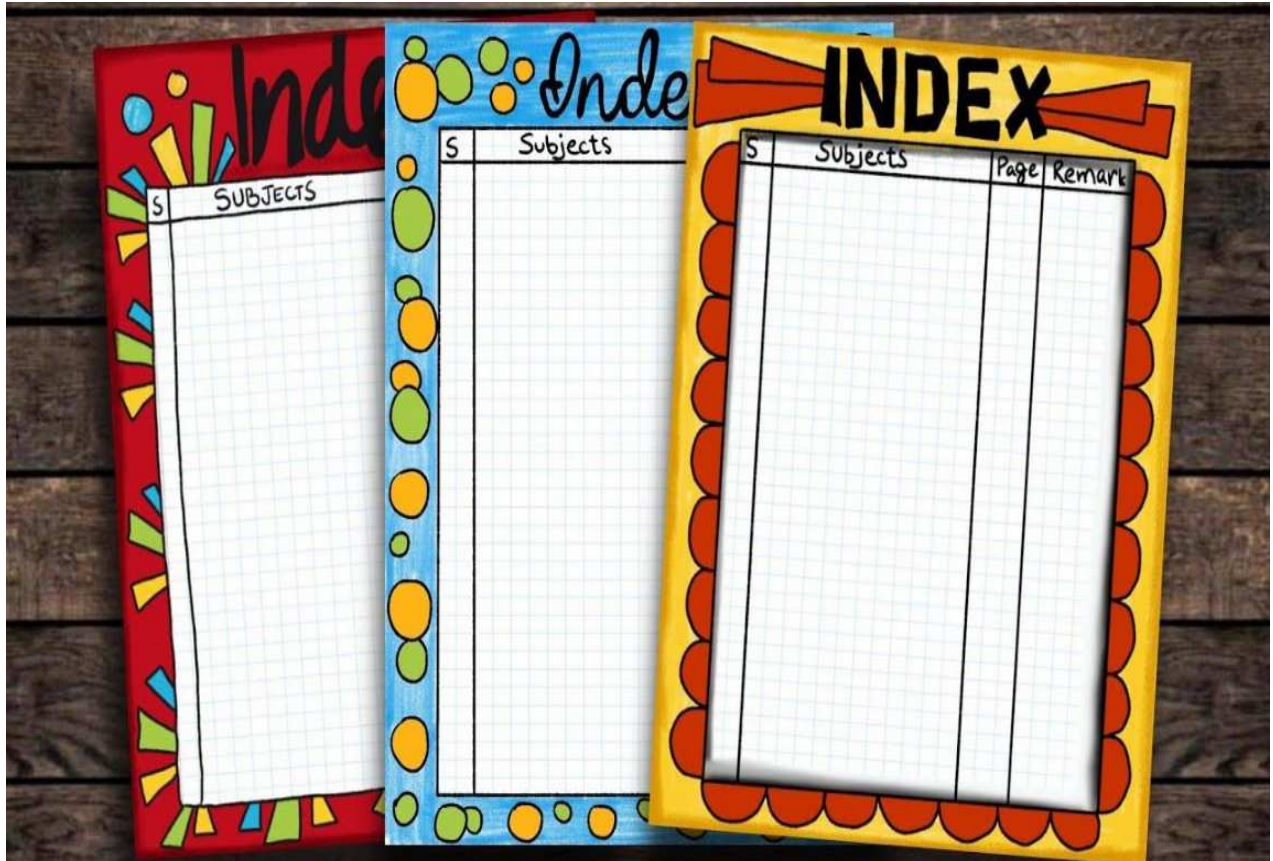
We are so grateful to Mrs.S.Gnanathangam M.C.A.,M.Phil., (Associate Professor & H.O.D) of Computer Science Department for her gracious guidance. Our heartfelt thanks to our project guide Mrs.S.Andrews Felcita MSc.,MBA(HRM) (Assistant Professor) for the valuable assistance and the kindness with which she reviewed our project.

We also express our boundless thanks to our lectures to Sr.K.Reetha M.Sc.,M.Phil.,(Assistant Professor) , Mrs. M. Jothimani B.E, M.Sc(IT).,M.Phil.,(Associate Professor) ,Mrs. J.Anila Maily M.Sc.,M.Phil.,(Associate Professor) for their kind cooperation in successful completion of the project.

We would like to thank our lab assistant, Mrs.J.Cladina, for her kind assistance in lab during project hours.

We thank our family members especially our parents for their encouragement and support both morally and financially which helped us to finish the project successfully.

Finally we express our special thanks to our friends who gave us a wonderful encouragement ideas to complete the project.



S.No	CONTENT	PAGE NO
1.	SYNOPSIS	8
2.	SYSTEM ANALYSIS	10
	2.1 EXISTING SYSTEM	11
	2.2 PROPOSED SYSTEM	11
	2.3 ANALYSING THE SYSTEM	12
3.	SYSTEM REQUIREMENTS AND SPECIFICATIONS	13
	3.1 WORKING ENVIRONMENT	14
	3.2 SYSTEM IMPLEMENTATION	14
	3.3 ABOUT SOFTWARE	16
4.	SYSTEM DESIGN	23
	4.1 TABLE DESIGN	24
	4.2 DATA FLOW DIAGRAM	28
	4.3 FORM DESIGN	31
	4.4 SOURCE CODE	37
5.	FUTURE ENHANCEMENT	82
6.	CONCLUSION	84
7.	BIBILOGRAPHY	86

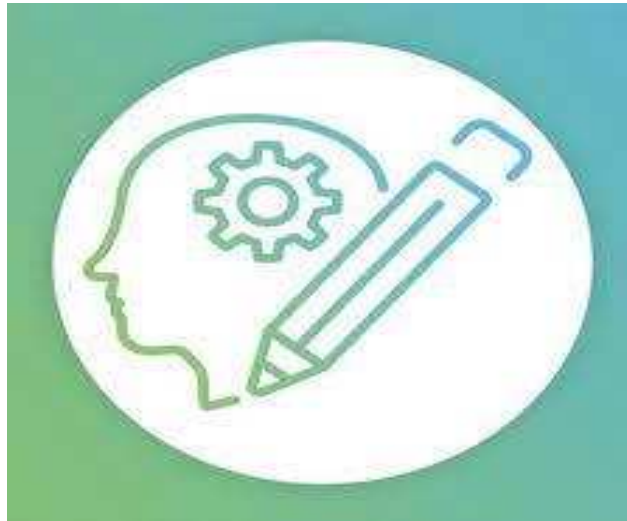
Synopsis

1.SYNOPSIS

The Project “OUR CITY” is designed with PHP as frontend and MySql as backend.

- The aim of the project enables the admin to maintain records of various places in our city (Tuticorin). The admin add, update, delete the information about the places and hotel bookings etc.,
- The admin maintain the records of the users who else booking the hotels.
- In normal world, the user has to spend more time to find hotels to stay and visit various places but with this Our City (for Tuticorin) project, persons from any location can look for best hotels, restaurants and shopping places.
- Online Our City website will save their time and help the user to visit various places within their scheduled time.
- This OUR CITY is designed to be more user friendly

SYSTEM ANALYSIS



2.1 Existing System



2.2 Proposed System



2.3 Analysing the System



2. SYSTEM ANALYSIS

The goal of the system development is to deliver systems in the line with user's requirements. Analysis is the heart of the process. It is the detailed study of the various operations performed by a system and their relationship within and outside the system.

2.1 Existing System

- The Our City website for a particular city is something new to this real world application.
- No other city has the website like this.

2.2 Proposed System

The development of the new system contains the following activities

- The admin can change or update the given places if needed.
- The system give user to give their information for Hotel Booking.
- The system makes the overall project management much easier and flexible.
- The system provides high level of security.
- The system give user more flexibility to delete the hotel bookings if not needed.
- The system give the original information about the places in our city (Tuticorin).

2.3 Analysing the System

The System after careful analysis has been identified to be presented with the following systems:

The project is divided into two main systems:

- Admin Section
- User Section

2.3.1 Admin Module

1. Login

2. Content Management

2.1 Tourist Places

2.2 Hotels

2.3 Malls

2.4 Restaurant

2.5 Cafes

2.6 Shopping

3. Contact Address Management

4. Booking Management

2.3.2 User Module

1. Signup

2. Login

3. Booking Management

4. Finding Places

SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 Working Environment



3.2 System Implementation



3.3 About the Software



3. SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 Working Environment

Hardware Specification

Processor : Intel® Pentium And Above

Hard Disk : 160GB And Above

RAM : 1GB And Above

Key Board : Logitech 118 keys

Software Specification

Operating System : Windows8 And Above

Front End : PHP

Back End : MYSQL

Server : Apache-server-9.0.9.M9

Documentation : MS Word

3.2 System Implementation

3.2.1 The Construction Phase of System Implementation

The construction phase does two things: builds and tests a functional system that fulfills business or organizational design requirements, and implements the interface between the new system and the existing production system. The project team must construct the database, application programs, user and system interfaces, and networks. Some of

these elements may already exist in your project or be subject to enhancement.

3.2.2 Pre-Sales Consulting and Design

Session the early stage, our pre-sales technical staff will have a detailed understanding of the demand of the client, investigate the situation of the client's system, and confirm the construction target for the client's system. Finally, complete the preparation of the system design program based on the target recognized by the client.

3.2.3 Preparatory Stage

Project manager initiates the project, forms implementation team, prepares implementation program and implementation plan.

3.2.4 System Installation and Debugging

Project engineer completes the software and hardware installation, stand alone device debugging, and system joint debugging based on the implementation plan and implementation program

3.2.5 System Testing

According to the system testing program in the preparatory stage, conduct testing for the debugged system. The testing includes device condition inspection, device and system function verification testing and system performance testing. Record the test results of the test program in detail to form the test report, and confirm that the system is in normal condition, all function and performance indicators meet the design requirements after the implementation so as to ensure the achievement of construction target.

3.2.6 Commissioning

The project will enter into commissioning stage after passing all tests. Before the system is put into the production operation, it will be put into small-scale or analog production operation for a period of time, so as to discover potential problem, and provide opportunity for adjustment before the formal operation.

3.2.7 Implementation Plan

The implementation plan includes a description of all the activities that must occur to implement the system and to put it into operation. It identifies the personal responsible for the activities and prepares a time chart for implementation the system. The implementation plan consists of the following steps:

- List all files required for implementation.
- Identify all data required to build new files during the implementation.
- List all new document and procedures that go into the new system.

The usual problems may be missing documents, mixed data formats between current and existing files, errors in data translation, missing data, etc.

3.3 About Software

3.3.1 PHP

PHP is a server side scripting language. That is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages .PHP scripts can only be interpreted on a server that has PHP installed. The client computers accessing the PHP scripts require a web browser only. PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and

executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control

3.3.2 Why to use PHP?

You have obviously heard of a number of programming languages out there; you may be wondering why we would want to use PHP as our poison for the web programming. Below are some of the compelling reasons.

- PHP is **open source and free**.
- Short learning curve compared to other languages such as JSP, ASP etc.
- Large community document
- Most web hosting servers support PHP by default unlike other languages such as ASP that need IIS. This makes PHP a cost effective choice.
- PHP is regularly updated to keep abreast with the latest technology trends.
- Other benefit that you get with PHP is that it's a **server side scripting language**; this means you only need to install it on the server and client computers requesting for resources from the server do not need to have PHP installed; only a web browser would be enough.
- PHP has **in built support for working hand in hand with MySQL**; this doesn't mean you can't use PHP with other database management systems.

You can still use PHP with

- Postgres
- Oracle
- MS SQL Server
- ODBC etc.
- PHP is **cross platform**; this means you can deploy your application on a number of different operating systems such as windows, Linux, Mac OS etc.

3.3.3 Advantages of PHP

The reason behind the popularity of PHP is its several advantages. PHP is most suited for the purpose of web development. The advantages of PHP are discussed briefly below:

1. Cross Platform

All the PHP based applications can run on various types of platforms. PHP is supported by majority of Operating Systems, some of which includes Solaris, UNIX, Windows and Linux. The mentioned platforms can be used to write codes in PHP and also view web pages or run the PHP based applications.

PHP easily interfaces with MySQL and Apache both. An effortless integration of PHP can be done with various other technologies like Java and there is no requirement of re-development. Therefore, saving both time and money, giving it an important advantage.

2. Easy database connection

A programming language like PHP is widely used on the internet and needs to connect to the database very often. Therefore, having a feature that could help PHP to connect to database easily is mandatory. Several websites such as the ecommerce websites, require good database management system.

PHP has a built-in module that helps it in connecting with database easily. Therefore, PHP has a great demand in the field of web development where a data driven website needs to be developed. PHP significantly reduces the time needed in developing the web application that needs an efficient database management system.

3. Easy to use

PHP is widely used because it is easy to use. In contrast with other programming languages that are complex, PHP is simple, fluent, clean and organized, hence it is a boon for the new users. PHP has a well-organized syntax which is logical at the same time.

PHP does not require any intensive studying or manual to use it. Command functions of PHP are easily understood as the user can easily figure out from the name of the commands itself what it does. A person who is new to PHP can still code because the syntax is somewhat similar to C.

A person who is new to PHP can still code because the syntax is somewhat similar to C. Hence, if a person who knows C can easily code in PHP. Hence, it is easier to create and optimize the application using PHP.

Speed is the primary need of web development. There are people who face the challenge of slow internet connection and slow data speed. Furthermore, a fast loading website is always preferred by people across the globe. When compared to other programming languages, PHP is found to be the fastest programming language.

In normal circumstances, it takes a lot of time to connect to the database, when you attempt to fetch certain data from the database. It takes a lot of time in connecting to the database, then executing the statement and finally getting the data. PHP performs these set of tasks faster than other scripting languages. PHP is faster in both connecting to the database and in using other important applications.

The high speed of PHP gives it an advantage over other scripting languages and gives it an application in important administrations such as the server administration and mail functionalities.

4. Open source

One of the important advantages of PHP is that it is Open Source. Therefore, PHP is readily available and is entirely free. In contrast to other scripting languages used for web development which requires the user to pay for the support files, PHP is open to everyone, anytime and anywhere.

A beginner in PHP need not worry about the support as PHP is maintained and developed by a large group of PHP developers which helps in creating support community of PHP that helps people in PHP implementation and manipulation.

3.3.4 MySQL

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

Advantages of Using MySQL

1. Its Easy To Use

MySQL is very easy to install, and thanks to a bevy of third-party tools that can be added to the database, setting up an implementation is a relatively simple task. In addition, it's also an easy database to work with. So long as you understand the language, you shouldn't run into too many problems.

2. Cross Platform Supports

Cross Platform compatibility of MySQL is another advantage. It can be installed in all major Operating Systems as UNIX, Solaris, and LINUX in addition to Windows without a loss of performance. It also runs without compromise with various development interfaces such as JDBC, ODBC, Pearl, Python, PHP, C++ and many others. This is primarily due to the development APIs that are integrated with it.

3. It's Open-Source (Sort Of)

Oracle's purchase of Sun Microsystems (and by association, MySQL) was met with some contention from the development community. The general fear was that Oracle would transform the tool into a closed, proprietary

ecosystem. Thankfully, though Oracle has tightened its grip on MySQL somewhat, it can still be considered an open-source database option, as the code is still available for free online.

4. Its Incredibly Inexpensive

Depending on what you plan to use it for, a MySQL implementation could range in price from free to \$10,000 or more. Either way, it's significantly less expensive than most other database options on the market (save for MySQL's open-source competitors).

5. Its An Industry Standard (And Still Extremely Popular)

Although MySQL's popularity has waned somewhat in recent years, it remains one of the most-used database systems in the world. It's compatible with virtually every operating system, and is more or less an industry standard. This is, of course, inspite of all the folks who say it's on the way out.

Disadvantages of Using MySQL

1. It's Got A Few Stability Issues

According to Digital Ocean, MySQL tends to be somewhat less reliable than its peers. These stability issues are related to the manner in which it handles certain functions (such as references, transactions, and auditing). While the database is certainly still usable in light of these problems, they do tend to make MySQL a poor choice for certain use cases.

2. It Suffers From Relatively Poor Performance Scaling

Although MySQL is equipped to handle a virtually limitless volume of data, it has a troubling tendency to come grinding to a halt if it's forced to deal with too many operations at a given time. This relatively poor performance scaling

means that anyone with high concurrency levels should probably look into an alternative.

3. Development Is Not Community Driven – and Hence Has Lagged

Since Oracle has taken the helm of MySQL's development, progress appears to have ground to a halt, with only one major release in the past several years. The company doesn't accept community-developed patches, nor has it bothered to offer users any sort of roadmap for MySQL development. There's really no way for developers to discuss the database management system with Oracle – and that's a problem.

SYSTEM DESIGN

Tables Slide

Type The Subtitle Of Your Great Here

	TEXT 1	TEXT 2	TEXT 3	TEXT 4
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X
TOTAL				

Tables Slide

Type The Subtitle Of Your Great Here

	TEXT 1	TEXT 2	TEXT 3	TEXT 4	TEXT 5
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X
TOTAL					

Tables Slide

Type The Subtitle Of Your Great Here

	TEXT 1	TEXT 2	TEXT 3	TEXT 4	TEXT 5	TEXT 6
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
This is a text placeholder	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X
TOTAL						

Tables Slide

Type The Subtitle Of Your Great Here

	TEXT 1	TEXT 2
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X
This is a text placeholder	XX.X	XX.X

4. SYSTEM DESIGN

4.1 Table Design

TABLE : Admin Login

Field Name	Data Type
Secret Name	text
Secret Key	text

TABLE : Page Content

Field Name	Data Type
menu	text
p_title	text
address	text
p_content	text
p_map	text
image1	text
image2	text
image3	text

TABLE : Customer

Field Name	Data Type
id	Int (Primary Key)
name	text
email	text
mobile	Int(12)
password	text
city	text
regdate	date

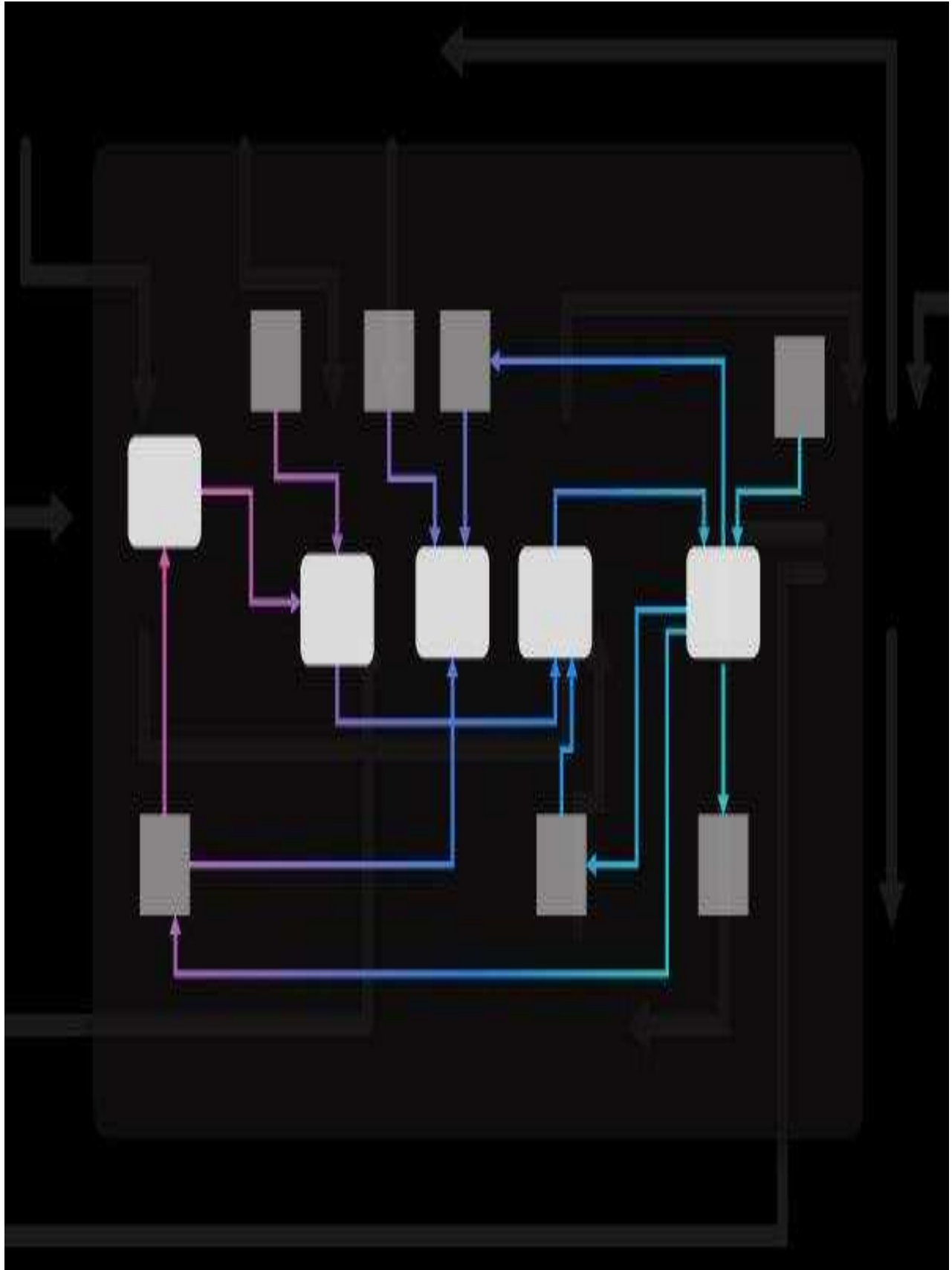
TABLE : Contact

Field Name	Data Type
id	Int (Primary Key)
name	text
mobile	text
email	text
address	text




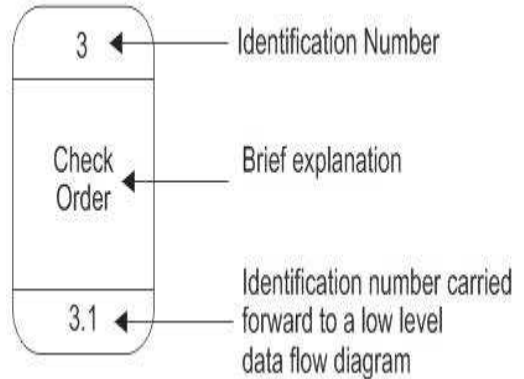



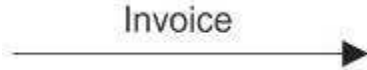
TABLE : Booking

Field Name	Data Type
id	Int (Primary Key)
cid	text
hotelname	text
name	text
mobile	text
fromcity	text
numberofadults	text
numberofchildren	text
checkin	text
checkout	text
bookingdate	date
bookingtime	text
status	Int(10)

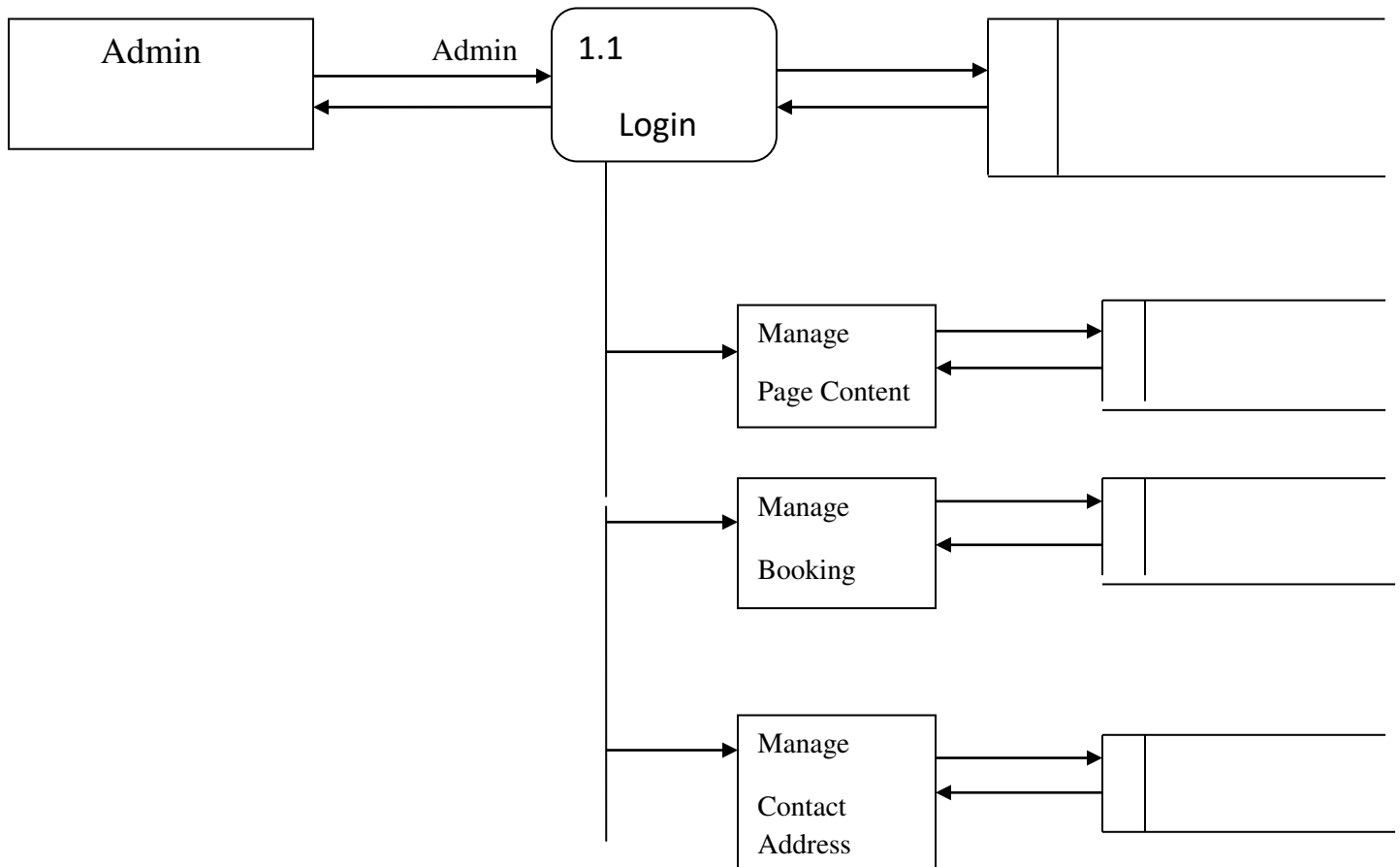
DATA FLOW DIAGRAM



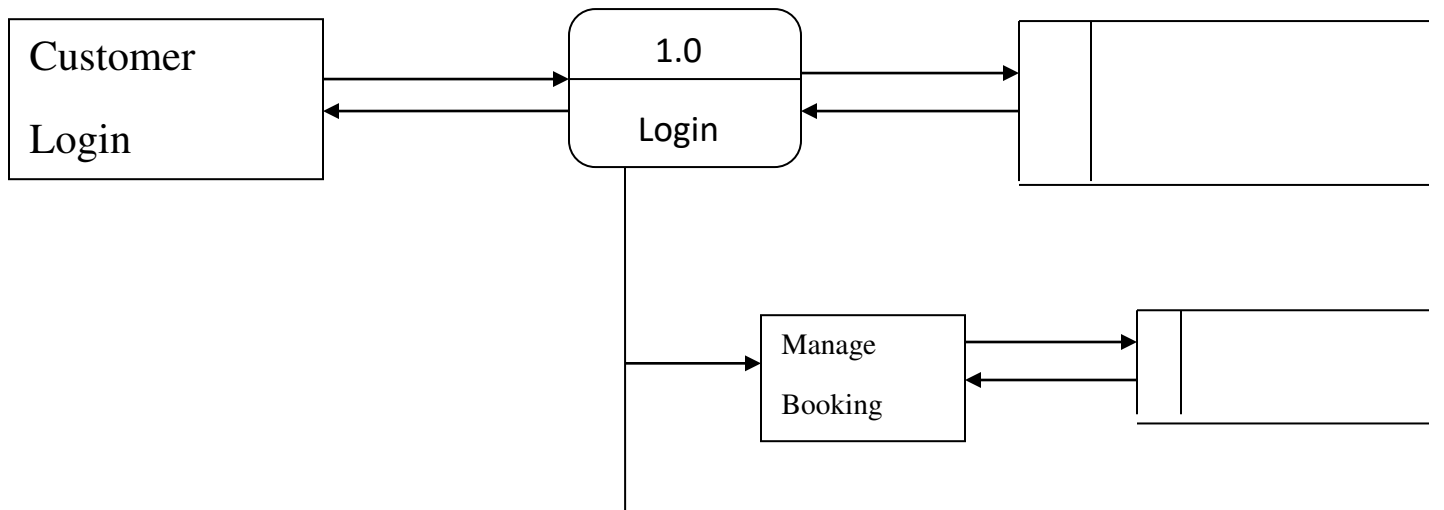
4.2 Data Flow Diagram

Symbol	Meaning	Example
	An entity . A source of data or a destination for data.	
	A process or task that is performed by the system.	
	A data store , a place where data is held between processes.	
	A data flow	

Admin



Customer

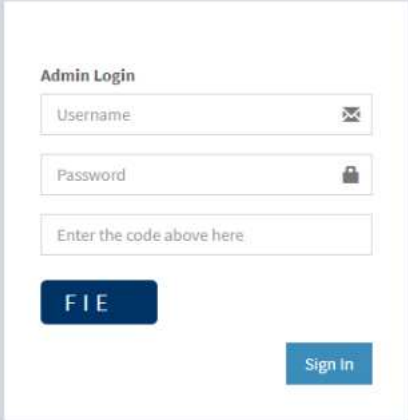


FORM DESIGN



4.3 Form Design

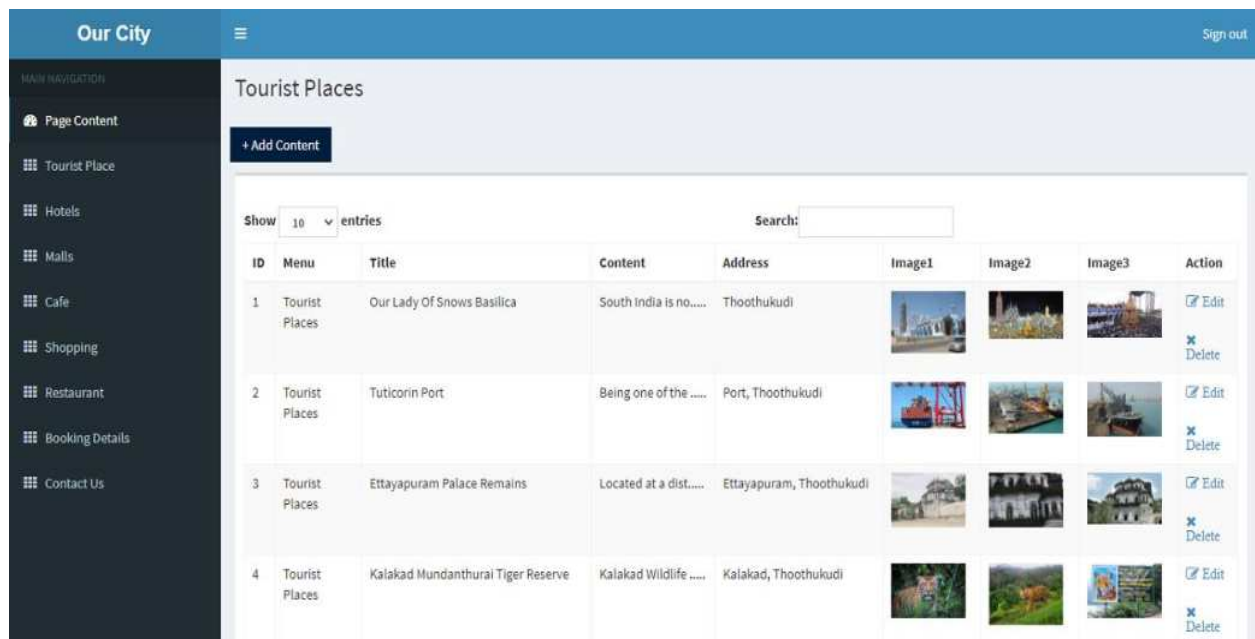
4.3.1 Admin Login















The Admin Login form is a white rectangular box centered on a light blue background. It contains the following elements:

- Admin Login** (Section Header)
- Username** (Text input field with an eye icon)
- Password** (Text input field with a lock icon)
- Enter the code above here** (Text input field)
- F I E** (Dark blue button)
- Sign In** (Dark blue button)

4.3.2 Content Page



The content page for 'Our City' features a dark blue sidebar with navigation links and a main content area for 'Tourist Places'. The sidebar includes links for Page Content, Tourist Place, Hotels, Malls, Cafe, Shopping, Restaurant, Booking Details, and Contact Us. The main content area has a '+ Add Content' button and a table of tourist places.

ID	Menu	Title	Content	Address	Image1	Image2	Image3	Action
1	Tourist Places	Our Lady Of Snows Basilica	South India is no.....	Thoothukudi				Edit Delete
2	Tourist Places	Tuticorin Port	Being one of the	Port, Thoothukudi				Edit Delete
3	Tourist Places	Ettayapuram Palace Remains	Located at a dist.....	Ettayapuram, Thoothukudi				Edit Delete
4	Tourist Places	Kalakad Mundanthurai Tiger Reserve	Kalakad Wildlife	Kalakad, Thoothukudi				Edit Delete

4.3.3 Add Content

Our City

Sign out

Main Navigation

- Page Content
- Tourist Place
- Hotels
- Malls
- Cafe
- Shopping
- Restaurant
- Booking Details
- Contact Us

Content Management

Menu

Select Menu

Title

Page Title

Address

Area,City

X Undo Bold Italic Underline Link Unlink Bulleted List Numbered List Indent Outdent Quote Paste Styles Format ?

Photo

Choose File No file chosen

Photo

Choose File No file chosen

Photo

Choose File No file chosen

Submit

4.3.4 Booking Details

Our City Sign out

Booking Details

Show 10 entries Search:

ID	Customer Name	Hotel Name	Mobile	City	No Of Adults	No Of Children	Checkin	ChecOut	Booking Date/Time	Status	Action
1	raju	SRM Hotels	98437	de	3	2	03/31/2021	03/31/2021	2021-03-27-09:55:02am	Confirmed	Confirm Cancel Delete

Showing 1 to 1 of 1 entries

Previous 1 Next

4.3.5 Customer Login

Our City HOME TOURIST PLACES HOTELS MALLS RESTAURANT SHOPPING CAFES

Login

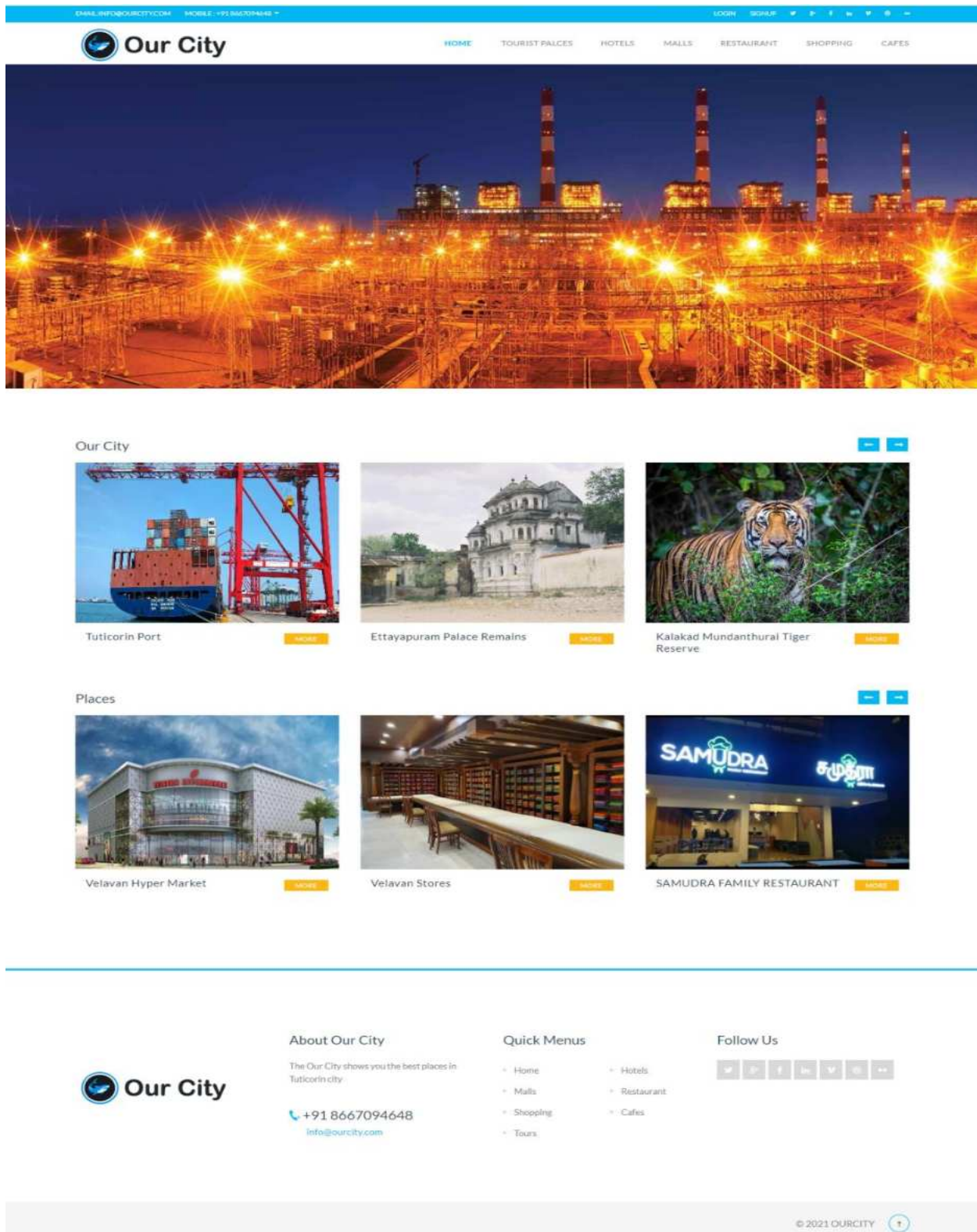
email address

password

LOGIN

Don't have an account? [Sign up](#)

4.3.6 Website Home Page



4.3.7 Customer Signup Page

The screenshot shows the 'Our City' website with a dark header and a city skyline background. A 'Sign Up' modal is centered on the screen. The modal contains the following fields and elements:

- Sign Up** (Title)
- (Name)
- (Mobile)
- (Email address)
- (Password)
- (Confirm password)
- (City)
-
- [Already a Travel member? Login](#)

4.3.8 Booking Details

The screenshot shows the 'Our City' website with a dark header and a city skyline background. The page displays the following content:

- Welcome raju**
- Booking Details**

#	Hotel Name	Number of Adults	Number of Children	CheckIn	CheckOut	Booking Date	Status	Action
1	SRM Hotels	3	2	03/31/2021	03/31/2021	2021-03-27	Confirmed	<input type="button" value="Delete"/>

- Tourist Places** (Sidebar):
 - [Hotels](#)
 - [Restaurants](#)
 - [Malls](#)
 - [Cafes](#)
 - [Shopping](#)

SOURCE CODE



4.4 Source Code

Admin:

//login.php

```
<?php
session_start();
include("db_con.php");
error_reporting(0);

$n=$_GET['param'];
$n1=$_GET['add'];
if($n=="err")
{
    $ans="<font color=red> Please enter correct User name and Password</font>";
}
if($n1=="error")
{
    $ans="You have Logged out Successfully!</CENTER>";
}
if($_GET['ca']=="err")
{
    $ans="<font color=red> Invalid Captcha code.</font>";
}

?>

<!DOCTYPE html>
```



```
<html>

<head>

  <meta charset="utf-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <title>OUR CITY</title>

  <!-- Tell the browser to be responsive to screen width -->

  <meta content="width=device-width, initial-scale=1, maximum-scale=1, user
scalable=no" name="viewport">

  <!-- Bootstrap 3.3.5 -->

  <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">

  <!-- Font Awesome -->

  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-
awesome/4.4.0/css/font-awesome.min.css">

  <!-- Ionicons -->

  <link rel="stylesheet"
href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">

  <!-- Theme style -->

  <link rel="stylesheet" href="dist/css/AdminLTE.min.css">

  <!-- iCheck -->

  <link rel="stylesheet" href="plugins/iCheck/square/blue.css">


  <link rel="stylesheet" href="dist/css/skins/_all-skins.min.css">
</head>

<body class="hold-transition login-page">

  <div class="login-box">

    <div class="col-md-12">
```

```

    <div class="nav-tabs-custom">

        <div class="tab-content">

            <div class="active tab-pane" id="admin">

                <div class="login-box-body">

                    <nav class="codrops-demos" style="color:red;"><?php echo
$ans;?></nav><BR>

                    <form action="checklogin.php" method="post">

                        <div class="form-group has-feedback">

                            <input type="text" class="form-control" name="uname"
placeholder="Username">

                            <span class="glyphicon glyphicon-envelope form-control-
feedback"></span>

                        </div>

                        <div class="form-group has-feedback">

                            <input type="password" class="form-control" name="upass"
placeholder="Password">

                            <span class="glyphicon glyphicon-lock form-control-feedback"></span>

                        </div>

                        <div class="form-group has-feedback">

<?php
function GeraHash($qtd){
$Caracteres = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
$QuantidadeCaracteres = strlen($Caracteres);
$QuantidadeCaracteres--;
$Hash=NULL;

    for($x=1;$x<=$qtd;$x++){

```

```

        $Posicao = rand(0,$QuantidadeCaracteres);

        $Hash .= substr($Caracteres,$Posicao,1);

    }

    return $Hash;

}

$randum_number=GeraHash(3);

$_SESSION['randum_number1']=$randum_number; ?>

        <input type="text" id="captcha_code"
name="captcha_code1" class="form-control" required placeholder="Enter the code
above here">

        <div style="width: 95px;
height: 36px;
margin-top: 20px;
background: #036;
border-radius: 5px;
color: #FFF;
letter-spacing: 5px;
padding: 4px 0px 0px 19px;
font-size: 20px;"><?php echo $randum_number; ?></div>

</div>

        <div class="row">

            <div class="col-xs-4">

                <input type="submit" style="margin-left:200px;" value="Sign In"
name="adminlogin" class="btn btn-primary btn-block btn-flat">

            </div><!-- /.col -->

        </div>

```

```

        </form>

</div><!-- /.login-box-body -->

    </div>

    <div class="tab-pane" id="department">

        <div class="login-box-body">

            <form action="checklogin.php" method="post">

<div class="form-group">

            <label>Department</label>

            <select class="form-control" name="dep" required>

<option>Select</option>

                <?php

                    $select_dep="SELECT * FROM `department` order by `id` ";

                    $result_dep=mysqli_query($conn,$select_dep);

                    while($getdep=mysqli_fetch_array($result_dep,MYSQLI_ASSOC))

                    {

                        ?>

                            <option value="<?php echo $getdep['dept']; ?>"><?php echo

$getdep['dept']; ?></option>

                                <?php } ?>

                            </select>

                        </div>

                        <div class="form-group has-feedback">

                            <input type="text" class="form-control" name="uname"

placeholder="Username">

                                <span class="glyphicon glyphicon-envelope form-control-

feedback"></span>

```

```

</div>

<div class="form-group has-feedback">

    <input type="password" class="form-control" password="upass"
placeholder="Password">

    <span class="glyphicon glyphicon-lock form-control-feedback"></span>

</div>

    <div class="form-group has-feedback">

<?php
function GeraHash2($qtd){
$Caracteres = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
$QuantidadeCaracteres = strlen($Caracteres);
$QuantidadeCaracteres--;
$Hash=NULL;

    for($x=1;$x<=$qtd;$x++){

        $Posicao = rand(0,$QuantidadeCaracteres);

        $Hash .= substr($Caracteres,$Posicao,1);

    }
return $Hash;

}

$randum_number2=GeraHash(4);

$_SESSION['randum_number2']=$randum_number2; ?>

        <input type="text" id="captcha_code2"
name="captcha_code2" class="form-control" required placeholder="Enter the code
above here">

        <div style="width: 95px;

height: 36px;

```

```

margin-top: 20px;
background: #036;
border-radius: 5px;
color: #FFF;
letter-spacing: 5px;
padding: 4px 0px 0px 19px;
font-size: 20px;"><?php echo $randum_number2; ?></div>

</div>

<div class="row">

    <div class="col-xs-4">

        <input type="submit" style="margin-left:200px;" value="Sign In"
name="delogin" class="btn btn-primary btn-block btn-flat">

    </div><!-- /.col -->

</div>

</form>

</div><!-- /.login-box-body -->

    </div>

    </div><!-- /.tab-content -->

    </div><!-- /.nav-tabs-custom -->

    </div><!-- /.col -->

</div><!-- /.login-box -->


<!-- jQuery 2.1.4 -->

<script src="plugins/jQuery/jQuery-2.1.4.min.js"></script>

```

```

<!-- Bootstrap 3.3.5 -->

<script src="bootstrap/js/bootstrap.min.js"></script>

<!-- iCheck -->

<script src="plugins/iCheck/ichack.min.js"></script>

<script>

$(function () {

    $('input').iCheck({

        checkboxClass: 'icheckbox_square-blue',

        radioClass: 'iradio_square-blue',

        increaseArea: '20%' // optional

    });

});

</script>

</body>

</html>

```

//Checklogin.php

```

<?php

session_start();

error_reporting(0);

include "db_con.php";

$username=$_REQUEST['uname'];

$upass=md5($_REQUEST['upass']);

if(isset($_REQUEST['adminlogin']))

```

```

{
$randum_number1=$_SESSION['randum_number1'];
if($_REQUEST['captcha_code1']!=$randum_number1){
    ?><script>window.location="login.php?ca=err";</script><?php
    }else{
        $select_count="select COUNT(*) as num from `webcms` where
        binary `secretname`='$uname' and binary `secretkey`='$upass'";
        $result_count=mysqli_query($conn,$select_count);
        $fetch_count=mysqli_fetch_array($result_count,MYSQLI_ASSOC);
        $total_pages2 = $fetch_count[num];
        if($total_pages2==1)
        {
            $_SESSION['UN']=$uname;
            $_SESSION['USER']="Admin";
            ?><script>window.location="dashboard.php";</script><?php
        }
        else
        {
            ?><script>window.location="login.php?param=err";</script><?php
        }
    }
}
?>

//Viewcontent.php

<?php include("header.php"); ?>

```



```

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">

  <!-- Content Header (Page header) -->
  <section class="content-header">

    <h1>

    <?php echo $_GET['me']; ?>

    </h1>

  </section>

  <!-- Main content -->
  <section class="content">

    <?php if($_SESSION['msg']==="1"){ ?>

      <div class="alert alert-success alert-dismissible col-xs-6">

        <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">×</button>

        <h4><i class="icon fa fa-check"></i> Content deleted
successfully!</h4>

      </div>

      <?php } ?>

    <div class="row">

      <a href="add_page.php">

        <button class="btn bg-navy btn-flat margin">+ Add
Content</button></a>

      <div class="col-xs-12">

        <div class="box">

          <div class="box-header">

            </div><!-- /.box-header -->

```

```

<div class="box-body">

    <table id="example1" class="table table-bordered table-striped">

        <thead>

            <tr>

                <th>ID</th>

                <th>Menu</th>

                <th>Title</th>

                <th>Content</th>

                <th>Address</th>

                <th>Image1 </th>

                <th>Image2</th>

                <th>Image3</th>

                <th>Action</th>

            </tr>

        </thead>

        <tbody>

<?php
    $v=1;
    $me=$_GET['me'];

    $select_user="SELECT * FROM `content` where menu='$me' order
by `id` ";

    $result_user=mysqli_query($conn,$select_user);

    while($getuser=mysqli_fetch_array($result_user,MYSQLI_ASSOC))
    {

?>

```

```

<tr>

    <td><?php echo $v; ?></td>

    <td><?php echo $getuser['menu']; ?></td>

    <td><?php echo $getuser['p_title']; ?></td>

    <td><?php echo substr($getuser['p_content'], 0, 20); ?>.....</td>

    <td><?php echo $getuser['address']; ?></td>

    <td></td>

    <td></td>

    <td></td>


    <td><a href="add_page.php?id=<?php echo $getuser['id']; ?>"><i
class="fa fa-edit"> Edit</i> </a> <br><br> <a href="delete.php?id=<?php echo
$getuser['id']; ?>&m=menu" onclick="return confirm('Are you sure to delete this
record?')"><i class="fa fa-close"> Delete</i></a></td>

</tr>

<?php $v++; } $_SESSION['msg']="";?>

</tbody>

</table>

</div><!-- /.box-body -->

</div><!-- /.box -->

<!-- /.box -->

</div><!-- /.col -->

</div><!-- /.row -->

</section><!-- /.content -->

```

```

</div><!-- /.content-wrapper -->

<!-- Add the sidebar's background. This div must be placed
      immediately after the control sidebar -->
<div class="control-sidebar-bg"></div>
</div><!-- ./wrapper -->

<!-- jQuery 2.1.4 -->
<script src="plugins/jQuery/jQuery-2.1.4.min.js"></script>

<!-- Bootstrap 3.3.5 -->
<script src="bootstrap/js/bootstrap.min.js"></script>

<!-- DataTables -->
<script src="plugins/datatables/jquery.dataTables.min.js"></script>
<script src="plugins/datatables/dataTables.bootstrap.min.js"></script>

<!-- SlimScroll -->
<script src="plugins/slimScroll/jquery.slimscroll.min.js"></script>

<!-- FastClick -->
<script src="plugins/fastclick/fastclick.min.js"></script>

<!-- AdminLTE App -->
<script src="dist/js/app.min.js"></script>

<!-- AdminLTE for demo purposes -->
<script src="dist/js/demo.js"></script>

<!-- page script -->
<script>
$(function () {
    $("#example1").DataTable();

```

```
$('#example2').DataTable({  
    "paging": true,  
    "lengthChange": false,  
    "searching": false,  
    "ordering": true,  
    "info": true,  
    "autoWidth": false  
});  
});  
</script>  
</body>  
</html>
```

//Viewbooking.php

```
<?php include("header.php"); ?>  
  
<!-- Content Wrapper. Contains page content -->  
<div class="content-wrapper">  
    <!-- Content Header (Page header) -->  
    <section class="content-header">  
        <h1>  
        <h3>Booking Details</h3>  
    </h1>  
    </section>  
    <!-- Main content -->
```

```

<section class="content">

<?php if($_SESSION['msg']=="1"){ ?>

<div class="alert alert-success alert-dismissible col-xs-6">

    <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">×</button>

    <h4><i class="icon fa fa-check"></i> Content deleted
successfully!</h4>

</div>

<?php } ?>

    <?php if($_SESSION['msg']=="2"){ ?>

    <div class="alert alert-success alert-dismissible col-xs-6">

    <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">×</button>

    <h4><i class="icon fa fa-check"></i> Booking confirmed
successfully!</h4>

    </div>

    <?php } ?>

    <?php if($_SESSION['msg']=="3"){ ?>

    <div class="alert alert-success alert-dismissible col-xs-6">

    <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">×</button>

    <h4><i class="icon fa fa-check"></i> Booking canceled
successfuslly!</h4>

    </div>

    <?php } ?>

<div class="row">

    <div class="col-xs-12">

```

```

<div class="box">

    <div class="box-header">

</div><!-- /.box-header -->

    <div class="box-body">

        <table id="example1" class="table table-bordered table-striped">

            <thead>

                <tr>

                    <th>ID</th>

                    <th>Customer Name</th>

                    <th>Hotel Name</th>

<th>Mobile</th>

<th>City</th>

<th>No Of Adults</th>

<th>No Of Children</th>

<th>CheckIn</th>

<th>ChecOut</th>

<th>Booking Date/Time</th>

<th>Status</th>

                    <th>Action</th>

                </tr>

            </thead>

            <tbody>

<?php

    $v=1;

    $me=$_GET['me'];

```

```

$select_user="SELECT * FROM `booking` order by `id` ";
$result_user=mysqli_query($conn,$select_user);
while($getuser=mysqli_fetch_array($result_user,MYSQLI_ASSOC))
{
    ?>
<tr>
    <td><?php echo $v; ?></td>
    <td><?php echo $getuser['name']; ?></td>
    <td><?php echo $getuser['hotelname']; ?></td>
    <td><?php echo $getuser['mobile']; ?></td>
    <td><?php echo $getuser['fromcity']; ?></td>

    <td><?php echo $getuser['numberofadults']; ?></td>
    <td><?php echo $getuser['numberofchildren']; ?></td>

    <td><?php echo $getuser['checkin']; ?></td>
    <td><?php echo $getuser['checkout']; ?></td>
    <td><?php echo $getuser['bookingdate'].'-'. $getuser['bookingtime'];
?></td>

    <td><?php if($getuser['status']=="1"){ ?><span
style="color:green;">Confirmed</span> <?php }elseif($getuser['status']=="0"){
?><span style="color:red;">Pending</span> <?php }else{ ?><span
style="color:red;">Cancelled</span> <?php } ?></td>

    <td><a href="delete.php?id=<?php echo $getuser['id']; ?>&c=1"><i
class="fa fa-edit">Confirm</i> </a> <br><br> <a href="delete.php?id=<?php echo
$getuser['id']; ?>&c=3" ><i class="fa fa-close">Cancel</i> </a>

```



```
<br> <a href="delete.php?id=<?php echo $getuser['id']; ?>&c=d"
onclick="return confirm('Are you sure to delete this record?')"><i class="fa fa-
close">Delete</i></a>
```

```
</td>
```

```
</tr>
```

```
<?php $v++; } $_SESSION['msg']="";?>
```

```
</tbody>
```

```
</table>
```

```
</div><!-- /.box-body -->
```

```
</div><!-- /.box -->
```

```
<!-- /.box -->
```

```
</div><!-- /.col -->
```

```
</div><!-- /.row -->
```

```
</section><!-- /.content -->
```

```
</div><!-- /.content-wrapper -->
```

```
<!-- Add the sidebar's background. This div must be placed
immediately after the control sidebar -->
```

```
<div class="control-sidebar-bg"></div>
```

```
</div><!-- /.wrapper -->
```

```
<!-- jQuery 2.1.4 -->
```

```
<script src="plugins/jquery/jquery-2.1.4.min.js"></script>
```

```

<!-- Bootstrap 3.3.5 -->

<script src="bootstrap/js/bootstrap.min.js"></script>

<!-- DataTables -->

<script src="plugins/datatables/jquery.dataTables.min.js"></script>

<script src="plugins/datatables/dataTables.bootstrap.min.js"></script>

<!-- SlimScroll -->

<script src="plugins/slimScroll/jquery.slimscroll.min.js"></script>

<!-- FastClick -->

<script src="plugins/fastclick/fastclick.min.js"></script>

<!-- AdminLTE App -->

<script src="dist/js/app.min.js"></script>

<!-- AdminLTE for demo purposes -->

<script src="dist/js/demo.js"></script>

<!-- page script -->

<script>

$(function () {

    $("#example1").DataTable();

    $('#example2').DataTable({

        "paging": true,

        "lengthChange": false,

        "searching": false,

        "ordering": true,

        "info": true,

        "autoWidth": false

    });

```

```
});  
</script>  
</body>  
</html>
```

User:

//index.php

```
<?php include("include/header.php");?>  
    <section id="content" class="image-bg1">  
        <div class="container" style="margin-top:18px;">  
            </div>  
    </section>  
    <div class="container section">  
        <h2>Our City</h2>  
        <div class="image-carousel style2" data-animation="slide" data-item-  
width="370" data-item-margin="30">  
            <ul class="slides image-box style10">  
  
                <?php  
                    $select_data="SELECT * FROM `content` order by `id` limit 0,4";  
                    $result_data=mysqli_query($conn,$select_data);  
                    while($getdata1=mysqli_fetch_array($result_data,MYSQLI_ASSOC))  
                        {  
  
                            ?>  
  
                <li class="box post">
```

```

        <figure>

            <a href="viewfulldetails.php?id=<?php echo $getdata1['id']; ?>"
class="hover-effect"></a>

        </figure>

        <div class="details">

            <a href="viewfulldetails.php?id=<?php echo $getdata1['id']; ?>"
class="button">MORE</a>

            <h4 class="post-title entry-title"><?php echo $getdata1['p_title'];
?></h4>

        </div>

    </li>

<?php } ?>

</ul>

</div>

<br>

<h2>Places</h2>

<div class="image-carousel style2" data-animation="slide" data-item-
width="370" data-item-margin="30">

    <ul class="slides image-box style10">

<?php

        $select_data="SELECT * FROM `content` order by `id` limit 4,8";
        $result_data=mysqli_query($conn,$select_data);
        while($getdata=mysqli_fetch_array($result_data,MYSQLI_ASSOC))
        {

            ?>

        <li class="box post">

```

```

        <figure>

            <a href="viewfulldetails.php?id=<?php echo $getdata['id']; ?>"
class="hover-effect"></a>

        </figure>

        <div class="details">

            <a href="viewfulldetails.php?id=<?php echo $getdata['id']; ?>"
class="button">MORE</a>

            <h4 class="post-title entry-title"><?php echo $getdata['p_title'];
?></h4>

        </div>

    </li>

<?php } ?>

</ul>

</div>

<br>

</div>

<?php include("include/footer.php");?>

```

//Listview.php

```

<?php include("include/header.php"); ?>

    <script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?sensor=false"></script>

    <script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>

    <script src="http://ajax.aspnetcdn.com/ajax/jquery/ui/1.8.9/jquery-ui.js"
type="text/javascript"></script>

```

```

<link href="css/jquery-ui.css" rel="stylesheet" type="text/css" />
<script type="text/javascript">
    $(function () {
        $("#btnShow").click(function () {
            $("#dialog").dialog({
                modal: true,title: "Kuwait Town Map",
                width: 1100,
                height: 600,
                buttons: {
                    Close: function () {
                        $(this).dialog('close');
                    }
                },
                open: function () {
                    var mapOptions = {
                        center: new google.maps.LatLng(19.0606917,
72.83624970000005),
                        zoom: 18,
                        mapTypeId: google.maps.MapTypeId.ROADMAP
                    }
                    var map = new google.maps.Map($("#dvMap")[0], mapOptions);
                }
            });
        });
    });
};

```

```

</script>

<div class="page-title-container">

    <div class="container">

        <div class="page-title pull-left">

            <h2 class="entry-title"><?php echo $_GET['t']; ?></h2>

        </div>

        <ul class="breadcrumbs pull-right">

            <li><a href="index.php">HOME</a></li>

            <li class="active"> Search Results</li>

        </ul>

    </div>

</div>

<section id="content">

    <div class="container">

        <div id="main">

            <div class="row">

                <div class="col-sm-4 col-md-3">

                    <div class="toggle-container filters-container">

                        <div class="panel style1 arrow-right">

                            <h4 class="panel-title">

                                <a href="listview.php?t=Tourist Places"
class="collapsed">Tourist Places</a>

                            </h4>

                        </div>

                    </div>

                </div>

            </div>

        </div>

    </div>

</section>

```

```
<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Hotels"
class="collapsed">Hotels</a>

    </h4>

</div>

<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Restaurant"
class="collapsed">Restaurants</a>

    </h4>

</div>

<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Malls"
class="collapsed">Malls</a>

    </h4>

</div>

<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Cafe"
class="collapsed">Cafes</a>

    </h4>

</div>

<div class="panel style1 arrow-right">

    <h4 class="panel-title">
```



```

        <a href="listview.php?t=Shopping"
class="collapsed">Shoppings</a>

    </h4>

</div>

</div>

</div>

<div class="col-sm-8 col-md-9">

    <div class="hotel-list listing-style3 hotel">

        <?php
            $me=$_GET['t'];

            $select_data="SELECT * FROM `content` where menu='$me' order
by `id` ";

            $result_data=mysqli_query($conn,$select_data);

            while($getdata=mysqli_fetch_array($result_data,MYSQLI_ASSOC))
            {

                ?>

        <article class="box">

            <figure class="col-sm-5 col-md-4"><a title=""
href="viewfulldetails.php?id=<?php echo $getdata['id']; ?>" class="hover-
effect"></a></figure>

            <div class="details col-sm-7 col-md-8">

                <div>

                    <div>

                        <h4 class="box-title"><?php echo $getdata['p_title'];
?><small><i class="soap-icon-departure yellow-color"></i><?php echo
$getdata['address']; ?></small></h4>

```

```

        </div>

    </div>

    <div>

        <?php echo substr($getdata['p_content'], 0, 200); ?

        <div>

            <a class="button btn-small full-width text-center" title=""
href="viewfulldetails.php?id=<?php echo $getdata['id']; ?>">VIEW</a>
        </div>

        </div>

    </div>

</article>

<?php } ?>

</div>

</div>

</div>

</div>

</section>

<?php include("include/footer.php"); ?>

```

```
<?php include("include/header.php"); ?>

<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?sensor=false"></script>

<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
```

```

<script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.8.9/jquery-ui.js"
type="text/javascript"></script>

<link href="css/jquery-ui.css"
rel="stylesheet" type="text/css" />

<script type="text/javascript">

$(function () {

    $("#btnShow").click(function () {

        $("#dialog").dialog({

            modal: true,title: "Kuwait Town Map",

            width: 1100,

            height: 600,

            buttons: {

                Close: function () {

                    $(this).dialog('close');

                }

            },

            open: function () {

                var mapOptions = {

                    center: new google.maps.LatLng(19.0606917,
72.83624970000005),

                    zoom: 30,

                    mapTypeId: google.maps.MapTypeId.ROADMAP

                }

                var map = new google.maps.Map($("#dvMap")[0], mapOptions);

            }

        });

    });

});

```

```

    });

});

});
</script>

<?php
$id=$_GET['id'];

$select_data="SELECT * FROM `content` where id='$id' order by `id` ";

$result_data=mysqli_query($conn,$select_data);

$getdata=mysqli_fetch_array($result_data);

?>

<div class="page-title-container">

    <div class="container">

        <div class="page-title pull-left">

            <h2 class="entry-title"><?php echo $getdata['p_title']; ?></h2>

        </div>

        <ul class="breadcrumbs pull-right">

            <li><a href="index.php">HOME</a></li>

        </ul>

    </div>

</div>

<section id="content">

    <div class="container tour-detail-page">

        <div class="row">

            <div id="main" class="col-md-9">

                <div class="tab-container style1">

```

```

        <ul class="tabs">

            <h2 class="box-title"><?php echo $getdata['p_title'];
?><small><i class="soap-icon-departure yellow-color"></i><span class="fourty-
space"><?php echo $getdata['address']; ?> </span></small></h2>

        </ul>

        <div id="photos-tab" class="tab-pane fade in active">

            <div class="photo-gallery style1" data-animation="slide" data-
sync="#photos-tab .image-carousel">

                <ul class="slides">

                    <li></li>

                    <li></li>

                    <li></li>

                </ul>

            </div>

            <div class="image-carousel style1" data-animation="slide"
data-item-width="70" data-item-margin="10" data-sync="#photos-tab .photo-
gallery">

                <ul class="slides">

                    <li></li>

                    <li></li>

                    <li></li>

                </ul>

            </div>

```

```

        </div>

</div>

<div id="tour-details" class="travelo-box">

<!-- <div class="tour-google-map block"></div>-->

    <h2><?php echo $getdata['p_title']; ?></h2>

    <p><?php echo $getdata['p_content']; ?></p>

    <div class="intro small-box table-wrapper full-width hidden-table-
sms">

        <div class="col-sm-12 table-cell">

            <iframe src="<?php echo strip_tags($getdata['p_map']); ?>"
style="width:100%; height:300px;" height="" style="border:0;" allowfullscreen=""
loading="lazy"></iframe>

        </div>

    </div>

</div>

</div>

</div>

<script>

function myFunction() {

    var x = document.getElementById("myDIV");

    if (x.style.display === "none") {

        x.style.display = "block";

    } else {

        x.style.display = "none";

    }

}

</script>

```

```

<?PHP if($getdata['menu']=="Hotels" || $getdata['menu']=="Restaurant"){ ?>

    <div class="sidebar col-md-3">

        <div class="travelo-box">

            <div class="row" >

                <button onclick="myFunction()" class="col-xs-12" style="font-size: 22px;">Booking Now</button>

            <?php if($_SESSION['msg']=='ad'){ ?>

                <div class="col-xs-12" style="margin-top:30px;">

                    <label style="color:green; ">Thank you for booking, We will contact you for verification.</label>

                </div>

            <?php $_SESSION['msg']=""; } ?>

            <div class="" id="myDIV" style="display:none; margin-top:30px;">

                <?php if($_SESSION['mobile']!=""){ ?>

                    <form method="post" action="booking.php">

                        <input type="hidden" name="hotelname" value="<?php echo $getdata['p_title']; ?>">

                        <input type="hidden" name="customer" value="<?php echo $_SESSION['USER']; ?>">

                        <input type="hidden" name="cid" value="<?php echo $_SESSION['ID']; ?>">

                        <input type="hidden" name="city" value="<?php echo $_SESSION['city']; ?>">

                        <input type="hidden" name="mobile" value="<?php echo $_SESSION['mobile']; ?>">

                        <input type="hidden" name="pid" value="<?php echo $_GET['id']; ?>">

```

```
<div class="col-xs-12" style="margin-top:30px;">
    <label>Check In</label>
    <div class="datepicker-wrap">
        <input type="text" name="checkin" required
class="input-text full-width" placeholder="mm/dd/yy" />
    </div>
</div>
</br>
```

```
<div class="col-xs-12" style="margin-top:20px;">
    <label>Check Out</label>
    <div class="datepicker-wrap">
        <input type="text" name="checkout"
class="input-text full-width" placeholder="mm/dd/yy" />
    </div>
</div>
```

```
<div class="col-xs-12" style="margin-top:20px;">
    <label>No of Adults</label>
    <div class="selector">
        <select name="noofadults" required>
            <option value="0">00</option>
            <option value="1">01</option>
            <option value="2">02</option>
            <option value="3">03</option>
            <option value="4">04</option>
        </select><span class="custom-select">00</span>
```


</div>

</div>

<div class="col-xs-12" style="margin-top:20px;">

<label>No of Children</label>

<div class="selector">

<select name="noofchildren" required>

<option value="0">00</option>

<option value="1">01</option>

<option value="2">02</option>

<option value="3">03</option>

<option value="4">04</option>

</select>00

</div>

</div>

<div class="col-xs-12" style="margin-top:20px;">

<button type="submit" name="booking" class="full-width uppercase">Proceed
</button>

</div>

</form>

<?php }else{ ?>

<div class="col-xs-12" style="margin-top:30px;">

<label><a href="#travelo-login" class="soap-popupbox"
style="color:#01b7f2;">LOGIN To Continue..</label>

</div>

```

<?php } ?>

</div>

</div>

</div>

</div>

<?php } ?>

<div class="sidebar col-md-3">

    <div class="travelo-box">

        <div class="image-box style14">

            <div class="toggle-container filters-container">

                <div class="panel style1 arrow-right">

                    <h4 class="panel-title">

                        <a href="listview.php?t=Tourist Places"
class="collapsed">Tourist Places</a>

                    </h4>

                </div>

                <div class="panel style1 arrow-right">

                    <h4 class="panel-title">

                        <a href="listview.php?t=Hotels"
class="collapsed">Hotels</a>

                    </h4>

                </div>

                <div class="panel style1 arrow-right">

                    <h4 class="panel-title">

                        <a href="listview.php?t=Restaurant"
class="collapsed">Restaurant</a>

```

```

        </h4>

    </div>

    <div class="panel style1 arrow-right">

        <h4 class="panel-title">

            <a href="listview.php?t=Malls"
class="collapsed">Malls</a>

        </h4>

    </div>

    <div class="panel style1 arrow-right">

        <h4 class="panel-title">

            <a href="listview.php?t=Cafe"
class="collapsed">Cafes</a>

        </h4>

    </div>

    <div class="panel style1 arrow-right">

        <h4 class="panel-title">

            <a href="listview.php?t=Shopping"
class="collapsed">Shoppings</a>

        </h4>

    </div>

</div>
</div>
</div>
</div>
</div>
</div>
</div>

```

</section>

<?php include("include/footer.php"); ?>

//Booking.php

<?php

session_start();

include("ocadmin/db_con.php");

if(isset(\$_REQUEST['booking']))

{

 \$cid=\$_REQUEST['cid'];

 \$hotelname=\$_REQUEST['hotelname'];

 \$customer=\$_REQUEST['customer'];

 \$mobile=\$_REQUEST['mobile'];

 \$checkin=\$_REQUEST['checkin'];

 \$checkout=\$_REQUEST['checkout'];

 \$noofadults=\$_REQUEST['noofadults'];

 \$noofchildren=\$_REQUEST['noofchildren'];

 \$city=\$_REQUEST['city'];

 date_default_timezone_set("Asia/Kolkata");

 \$regdate=date('Y-m-d');

 \$time=date("h:i:sa");

 mysqli_query(\$conn,"insert into
`booking`(cid,hotelname,name,mobile,fromcity,numberofadults,numberofchildren,
 checkin,checkout,bookingdate,bookingtime)values('\$cid','\$hotelname','\$custo
mer','\$mobile','\$city','\$noofadults','\$noofchildren','\$checkin','\$checkout','\$regdate','\$
time')");

```

mysqli_close($conn);

$pid=$_REQUEST['pid'];

$_SESSION['msg']="ad";

echo "<script>window.location='viewfulldetails.php?id=".$pid."'</script>";

}

?>

```

//Dashboard.php

```

<?php include("include/header.php"); ?>

<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?sensor=false"></script>

<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>

<script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.8.9/jquery-ui.js"
type="text/javascript"></script>

<link href="css/jquery-ui.css"

rel="stylesheet" type="text/css" />

<script type="text/javascript">

$(function () {

    $("#btnShow").click(function () {

        $("#dialog").dialog({

            modal: true,title: "Kuwait Town Map",

            width: 1100,

            hright: 600,

            buttons: {

```

```

        Close: function () {
            $(this).dialog('close');
        }
    },
    open: function () {
        var mapOptions = {
            center: new google.maps.LatLng(19.0606917,
72.836249700000005),
            zoom: 30,
            mapTypeId: google.maps.MapTypeId.ROADMAP
        }
        var map = new google.maps.Map($("#dvMap")[0], mapOptions);
    }
});

});

});

</script>

<div class="page-title-container">
    <div class="container">
        <div class="page-title pull-left">
            <h2 class="entry-title">Welcome <?php echo
$_SESSION['USER'];?></h2>
        </div>
    </div>
</div>

```

```

<section id="content">

  <div class="container tour-detail-page">

    <div class="row">

      <div id="main" class="col-md-9">

        <div class="tab-container ">

          <ul class="tabs">

            <h2 class="box-title">Booking Details</h2>

          </ul>

          <br>

          <table class="table" style="background-color:white;">

            <tr>

              <td>#</td>

              <td>Hotel Name</td>

              <td>Number of Adults</td>

              <td>Number of Children</td>

              <td>CheckIn</td>

              <td>CheckOut</td>

              <td>Booking Date</td>

              <td>Status</td>

              <td>Action</td>

            </tr>

            <?php

              $id=$_SESSION['ID'];

              $v=1;

```

```

        $select_data="SELECT * FROM `booking` where cid='$id' order
by `id` ";

        $result_data=mysqli_query($conn,$select_data);

        while($getdata=mysqli_fetch_array($result_data))

            {

                ?>

            <tr>

                <td><?php echo $v; ?></td>

                <td><?php echo $getdata['hotelname']; ?></td>

                <td><?php echo $getdata['numberofadults']; ?></td>

                <td><?php echo $getdata['numberofchildren']; ?></td>

                <td><?php echo $getdata['checkin']; ?></td>

                <td><?php echo $getdata['checkout']; ?></td>

                <td><?php echo $getdata['bookingdate']; ?></td>

                <td><?php if($getdata['status']=="1"){ ?><span
style="color:green;">Confirmed</span> <?php }elseif($getdata['status']=="0"){
?><span style="color:red;">Pending</span> <?php }else{ ?><span
style="color:red;">Cancelled</span> <?php } ?></td>

                <td><a href="delete.php?id=<?php echo $getdata['id'];
?>"><button>Delete</button></a></td>

            </tr>

            <?php $v++; } ?>

        </table>

    </div>

</div>

<div class="sidebar col-md-3">

    <div class="travelo-box">

```



```

<div class="image-box style14">

    <div class="toggle-container filters-container">

        <div class="panel style1 arrow-right">

            <h4 class="panel-title">

                <a href="listview.php?t=Tourist Places"
class="collapsed">Tourist Places</a>

            </h4>

        </div>

        <div class="panel style1 arrow-right">

            <h4 class="panel-title">

                <a href="listview.php?t=Hotels"
class="collapsed">Hotels</a>

            </h4>

        </div>

        <div class="panel style1 arrow-right">

            <h4 class="panel-title">

                <a href="listview.php?t=Restaurant"
class="collapsed">Restaraunt</a>

            </h4>

        </div>

        <div class="panel style1 arrow-right">

            <h4 class="panel-title">

                <a href="listview.php?t=Malls"
class="collapsed">Malls</a>

            </h4>

        </div>

```

```
<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Cafe"
class="collapsed">Cafes</a>

    </h4>

</div>

<div class="panel style1 arrow-right">

    <h4 class="panel-title">

        <a href="listview.php?t=Shopping"
class="collapsed">Shopping</a>

    </h4>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<?php include("include/footer.php"); ?>
```

```
//delete.php
```

```
<?php include("ocadmin/db_con.php");

$_SESSION['msg']='1';
```

```
$id=$_REQUEST['id'];  
  
mysqli_query($conn,"delete from `booking` where id='$id'");  
  
mysqli_close($conn);  
  
><script> window.history.back();</script>  
  
<?php  
  
>
```

//Loginaction.php

```
<?php  
  
session_start();  
  
error_reporting(0);  
  
include "ocadmin/db_con.php";  
  
$uname=$_REQUEST['uname'];  
  
$upass=$_REQUEST['upass'];  
  
if(isset($_REQUEST['uname']) && isset($_REQUEST['upass']))  
{  
  
    $select_count="select COUNT(*) as num from `customer` where  
`email`='$uname' and `password`='$upass';  
  
    $result_count=mysqli_query($conn,$select_count);  
  
    $fetch_count=mysqli_fetch_array($result_count,MYSQLI_ASSOC);  
  
    $total_pages2 = $fetch_count[num];  
  
    if($total_pages2==1)  
    {  
  
        $select_count2="select * from `customer` where `email`='$uname'  
and `password`='$upass';
```

```

$result_count2=mysqli_query($conn,$select_count2);

$fetch_count=mysqli_fetch_array($result_count2,MYSQLI_ASSOC);

    $_SESSION['UN']=$fetch_count['name'];

    $_SESSION['USER']=$fetch_count['name'];

$_SESSION['ID']=$fetch_count['id'];

$_SESSION['city']=$fetch_count['city'];

$_SESSION['mobile']=$fetch_count['mobile'];

        ?><script>window.location="dashboard.php";</script><?php
    }
    else
    {
        ?><script>window.location="index.php?param=err";</script><?php
    }
}
else
{
    ?><script>window.location="index.php?param=mis";</script><?php
}
?>

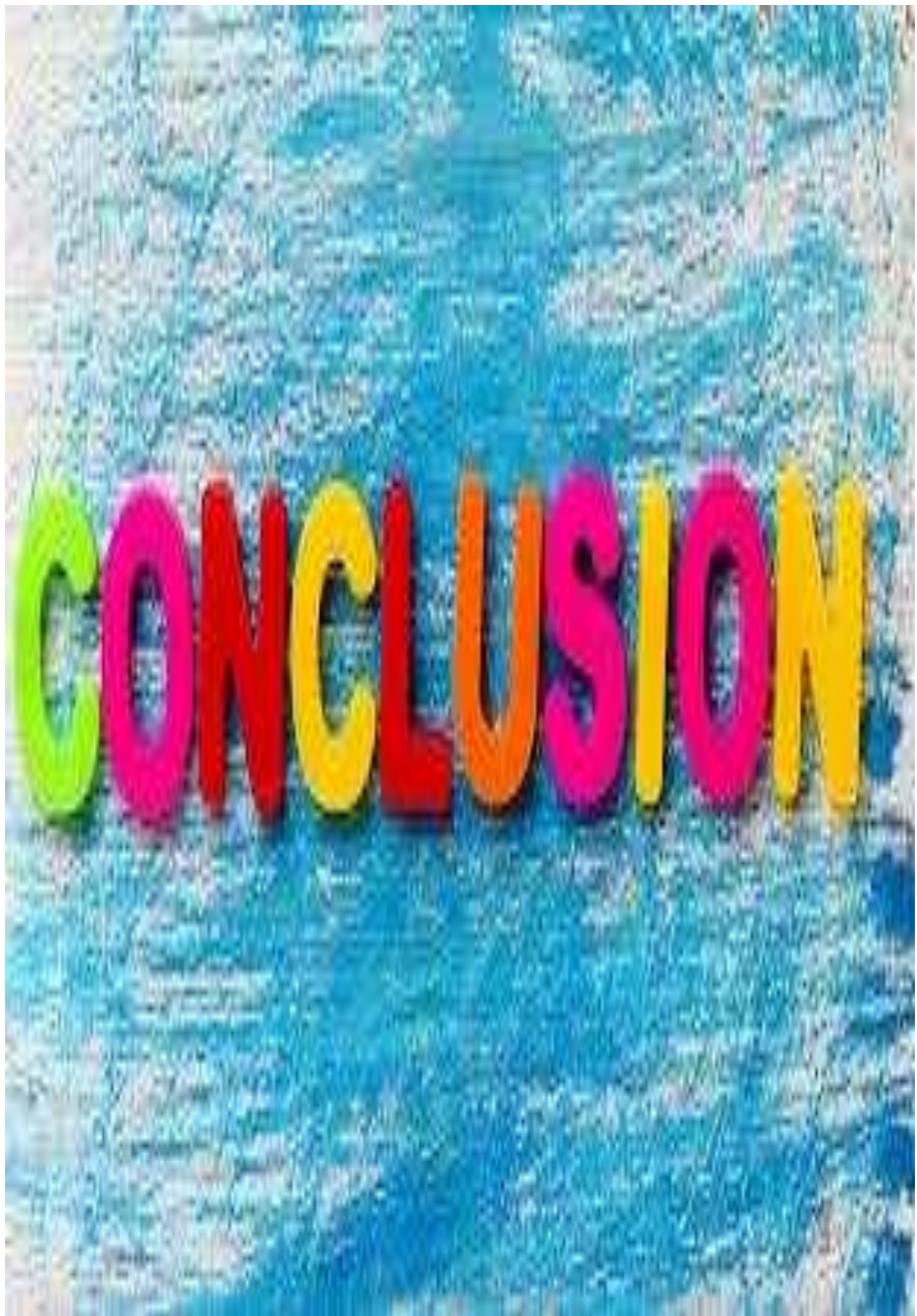
```

FUTURE ENHANCEMENT



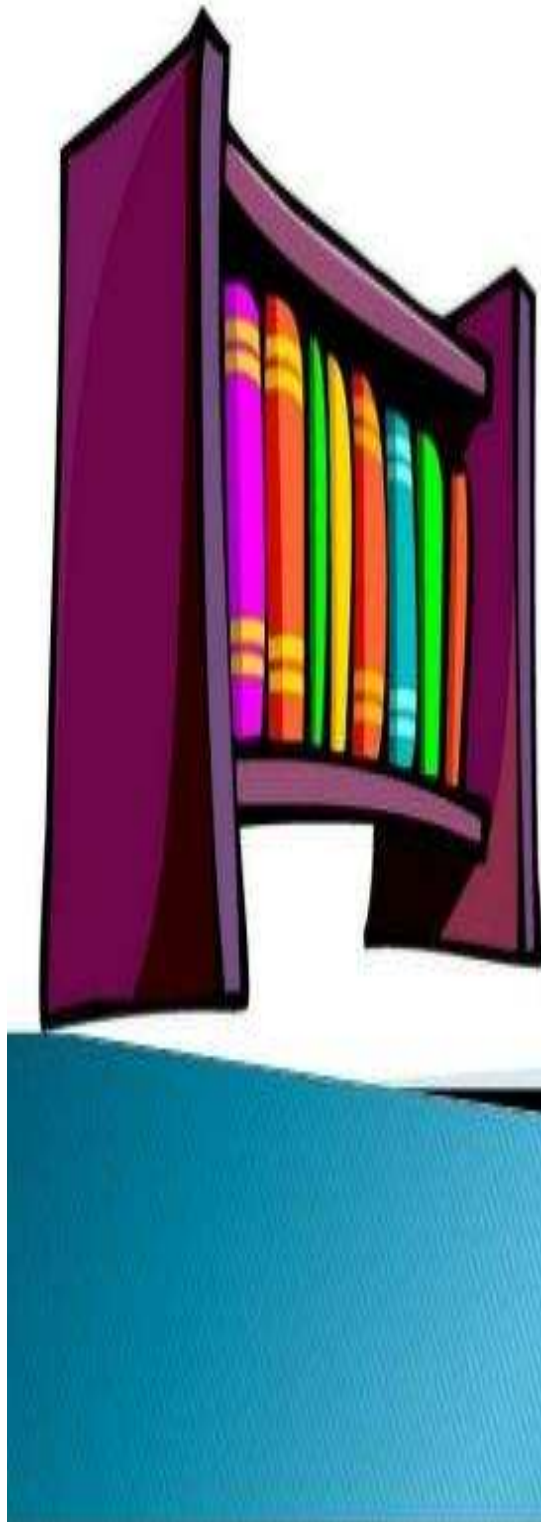
5. FUTURE ENHANCEMENT

Tourism is considered to be an important aspect of economic growth and the development of a city. Tourism management is generally considered a bright and potential employment sector as it offers a wide variety of career opportunities in both the private and government, Our City project will be converted as app the people can use in their mobile. Lot of people know the important places in our city.



6. CONCLUSION

Our City website usefull for finding popular places in our city that are listed in webist. The people can find their need and order or booking through this website and they can find the location where place is located in city.this website will be used as globaly everyone can access this website using their login credentials.



BIBLIOGRAPHY

BIBLIOGRAPHY

- [1]S Selvi, ManasRath, N K Sinha, S P Singh, N N J Hemrom, A Bhattacharya, A K Biswal*HRe-LeaveTourManagementSystematRDCIS,SAILInternationalConferenceonInformation Technology*, June-2014 .
- [2]PandoreYogeshBandopant*E-Notifier: Transport Information Services in -Colleges and ExploringMobileNotificationInternationalJournalofAdvancedResearchinComputerandCom- munication Engineering* Vol. 5, Issue 4, April2016.
- [3]RehmanS.U *Smart agent for automated E-commerce UK IEEE pp.124-128, 7-10 7-10 Nov.2011, School of information System, Computer. Math, Brunel Univ,Uxbridge.*
- [4]Richard Barker, Tasks and Deliverables, AddisonWesley.
- [5]Bell, Charles, Expert MySQL, Apress, 2012.
- [6]<http://www.w3schools.com>
- [7]<http://www.timeoffmanager.com>.
- [8]<http://www.planmyleave.com>.
- [9]<http://adp.in>.

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is certify that the project entitled

BRIDAL MAKEUP BOOKING APP

Submitted by

M. Jeya Bharathi

18AUCS12

J. Joericka

18AUCS13

B. Maria Graycita

18AUCS23

S. Sindhuja

18AUCS45

In partial fulfilment of the requirement for the award of degree in **BACHELOR OF COMPUTER SCIENCE** during the year 2020-2021.

J. Anilindhy
Guide

S. Gnaly
Head of the Department

Submitted for the Examination held on 15.4.2021

M. [Signature]
Examiner

Lucia Rose
Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

BRIDAL MAKEUP BOOKING APP

Project Submitted to the Department Of Computer Science

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI Affiliated to
MANONMANIAM SUNDARANAR UNIVERSITY



In partial fulfillment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE Done by

M. Jeya Bharathi 18AUCS12

J. Joericka 18AUCS13

B. Maria Graycita 18AUCS23

S. Sindhuja 18AUCS45

Under the Supervision and Guidance of

Mrs. J. Anila Maily M.Sc., M.Phil.(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE

ST.MARY'S COLLEGE (AUTONOMOUS) (Reaccredited with "A+" Grade by NAAC)
THOOTHUKUDI-62001

2018-2021

CERTIFICATE

DEPARTMENT OF COMPUTER SCIENCE

ST.MARY'S COLLEGE (Autonomous) THOOTHUKUDI, Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY



This is certify that the project entitled **BRIDAL MAKEUP BOOKING APP**

Submitted by

M. Jeya Bharathi 18AUCS12

J.Joericka 18AUCS13

B.Maria Graycita 18AUCS23

S.Sindhuja 18AUCS45

In partial fulfillment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE during the year 2018-2021

GUIDE

Head of the Department

Submitted for the examination held on _____

Examiner

Principal

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and blessings to finish our project successfully.

We are infinitely indebted to our college principal **Dr. Sr. A. S. J. Lucia Rose M.SC, M.Phil., PGDCA, Ph.D.**, for providing us necessary infrastructure to undertake this project and work successfully.

We are so grateful to **Mrs. S. Gnanathangam M.C.A, M.Phil (Associate professor & HOD)**, Department of Computer Science, for her gracious guidance. Our heartfelt thanks to our project guide **Mrs. J. Anila Maily M.Sc , M.Phil (Associate Professor)**, Department of Computer Science, for her valuable assistance and kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- **Mrs. M. Jothimani B.E., M.Sc., M.Phil., (Associate professor)**
- **Sr. K. Reetha M.Sc.,M.Phil.(Assistant professor)**
- **Mrs. S. Andrews Felcita M.Sc.,M.B.A (Assistant Professor),**

for their kind cooperation in successful completion of the project.

We would like to thank our lab assistant, **Mrs. J. Cladina**, for her kind assistance in the lab during the project hours.

We thank our family members for their encouragement and support both morally and financially which helped us to finish the project successfully.

Finally we express our special thanks to our friends who gave us wonderful encouragement and ideas to complete the project.

| S.NO. | CONTENT | PAGE.NO |
|--------------|--------------------------------------|----------------|
| 1. | <u>Introduction</u> | 8 |
| 2. | <u>System Analysis</u> | 11 |
| 3. | <u>Project Description</u> | 16 |
| 4. | <u>System Specifications</u> | 18 |
| 5. | <u>System Configuration</u> | 25 |
| 6. | <u>Data Flow Diagram</u> | 33 |
| 7. | <u>Testing Implementation</u> | 36 |
| 8. | <u>Future Enhancements</u> | 43 |
| 9. | <u>Conclusion</u> | 45 |

| | | |
|------------|----------------------------|-----------|
| 10. | <u>Codings</u> | 48 |
| 11. | <u>Screenshots</u> | 60 |
| 12. | <u>Bibliography</u> | 67 |

1.INTRODUCTION

1.1 OBJECTIVE:

Every bride dreams to have a wonderful, flawless, gorgeous look at her wedding. She always has a picture framed in her eyes how beautiful and elegant she should look on the special day of her life. Brides usually have everything ready in a frame that has been planned well to execute the best possible look. In that frame, things such as their outfit, hairstyle, footwear and makeup are already decided. To attain the best look at their marriage ceremony, they ought to look perfect and precise. And to look the best it is important to take tender care of the makeup look along with the attire and other stuff because a wrong makeup look can ruin everything you have ever dreamed of for your wedding day.

As the official integrated development environment for all Android applications, Android Studio seems to be at the top list of preferred tools for developers. Our idea is to develop a Bridal Makeup Booking App by using Android Development Toolkit.

2.SYSTEM ANALYSIS

2.1 System analysis:

System analysis is the detailed study of various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optional or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to the system.

- **Technical Feasibility**
- **Economic Feasibility**
- **Physical Feasibility**

2.1.1 Technical Feasibility:

The System Developer's task is to view the needed capabilities in light of currently available technologies. Our app works hand in hand with high technologies. Security and usability are considered as the most important factors of the system design that increases the user friendliness of the system. A database has to be maintained in order to update and backup whenever necessary. To create a database, we used SQLite.

2.1.2 Economic Feasibility:

In making recommendations a study of the economics of the proposed system should be made. Even though finding out the cost of the proposed project is difficult we assume and estimate the costs and benefits as follows:

It is a cost effectively designed app and it can be categorized in two ways.

- Cost associated with the product of cosmetics.
- Cost associated with the Android OS.

2.1.3Physical Feasibility:

A system resembling a mini authentication activity which is physically a less component space to make the system with high potential security. So the desired Bridal Makeup Booking App is physically feasible indeed.

2.2 Existing System:

Makeup Booking App is only for booking and to know whether the particular day for grooming is available or not. Online payments such as credit or debit cards are not possible.

2.2.1 Drawbacks of Existing System:

- Online Payments are not possible.

- Limited stocks available.
- Many orders can't be placed per day.
- Through online, once an order is placed it cannot be cancelled.

2.3 PROPOSED SYSTEM:

Bridal Makeup Booking App is a mobile-based app with appointment scheduling functionality. It connects customers and makeup artists in an online community allowing users to select the types of bridal makeup, book appointments. Beauticians can give appointments for customers and can view and point schedules in a convenient manner.

Our project has used Android and SQLite to back the interface with strong database functionality. For appointments scheduling, Bridal Booking App has calendar and lock design. The categories are designed as drop-down boxes and it will integrate the SQLite database for appointment as well as front-end scheduling interface.

Some of the major use cases include user login, appointment scheduling, schedule viewing to our SQLite database.

2.4 ADVANTAGES:

- Can be used by all mobile users
- Less memory needed to run this app
- Inventory management
- Secure Payment
- Effective Marketing.

3.PROJECT **DESCRIPTION**

3.1 PROJECT DESCRIPTION:

Makeover apps don't just change how women look, but it satisfies you with an image that instantly crops up in your head every time you think of yourself as a bride on your big day. It's major chunk of your nervousness will vanish overnight when you get a rough idea of what you will look like on your big day. All you need is a functional mobile phone with a good internet connection. A quick tap on your mobile, will complete your bridal makeup booking app then download and start experimenting with your looks. From makeup to hairstyles and even hand art, you can experiment with everything under the sky and see what suits you best on your special day.

4.SYSTEM

SPECIFICATIONS

4.1 SOFTWARE REQUIREMENT:

- FRONT END: Android Studio
- CODING: Java
- BACK END: SQLite
- OPERATING SYSTEM: Windows XP/10

4.2 HARDWARE REQUIREMENT:

- INTEL DUAL CORE 2.0Ghz CPU
- 2GB RAM
- 320GB HDD
- KEY BOARD
- MOUSE
- Mobile/Tablet

4.3 NETWORK REQUIREMENT:

- Internet Connection

4.4 SOFTWARE DESCRIPTION:

The early feedback on developing applications for the Android platform was mixed. Issues cited include bugs, lack of documentation, inadequate QA infrastructure, and no public issue-tracking system. (Google announced an issue tracker on 18 January 2008.) In December 2007, MergeLab mobile startup founder Adam MacBeth stated, "Functionality is not there, is poorly documented or just doesn't work... It's clearly not ready for prime time." Despite this, Android-targeted

applications began to appear the week after the platform was announced. The first publicly available application was the Snake game. The Android Dev Phone is a SIM-unlocked and hardware-unlocked device that is designed for advanced developers. While developers can use regular consumer devices purchased at retail to test and use their applications, some developers may choose not to use a retail device, preferring an unlocked or no contact device.

4.5 SOFTWARE DEVELOPMENT KIT:

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator (based on QEMU), documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.4.9 or later, Windows XP or later. The officially supported integrated development environment (IDE) is Eclipse (currently 3.4 or 3.5) using the Android Development Tools (ADT) Plugin, though developers may use any text editor to edit Java and XML file then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android device (e.g., triggering a reboot, installing software package(s) remotely).

A preview release of the Android SDK was released on 12 November 2007. On 15 July 2008, the Android Developer Challenge Team accidentally sent an email to all entrants in the Android Developer Challenge announcing that a new release of the SDK was available in a "private" download area. The email was intended for winners of the

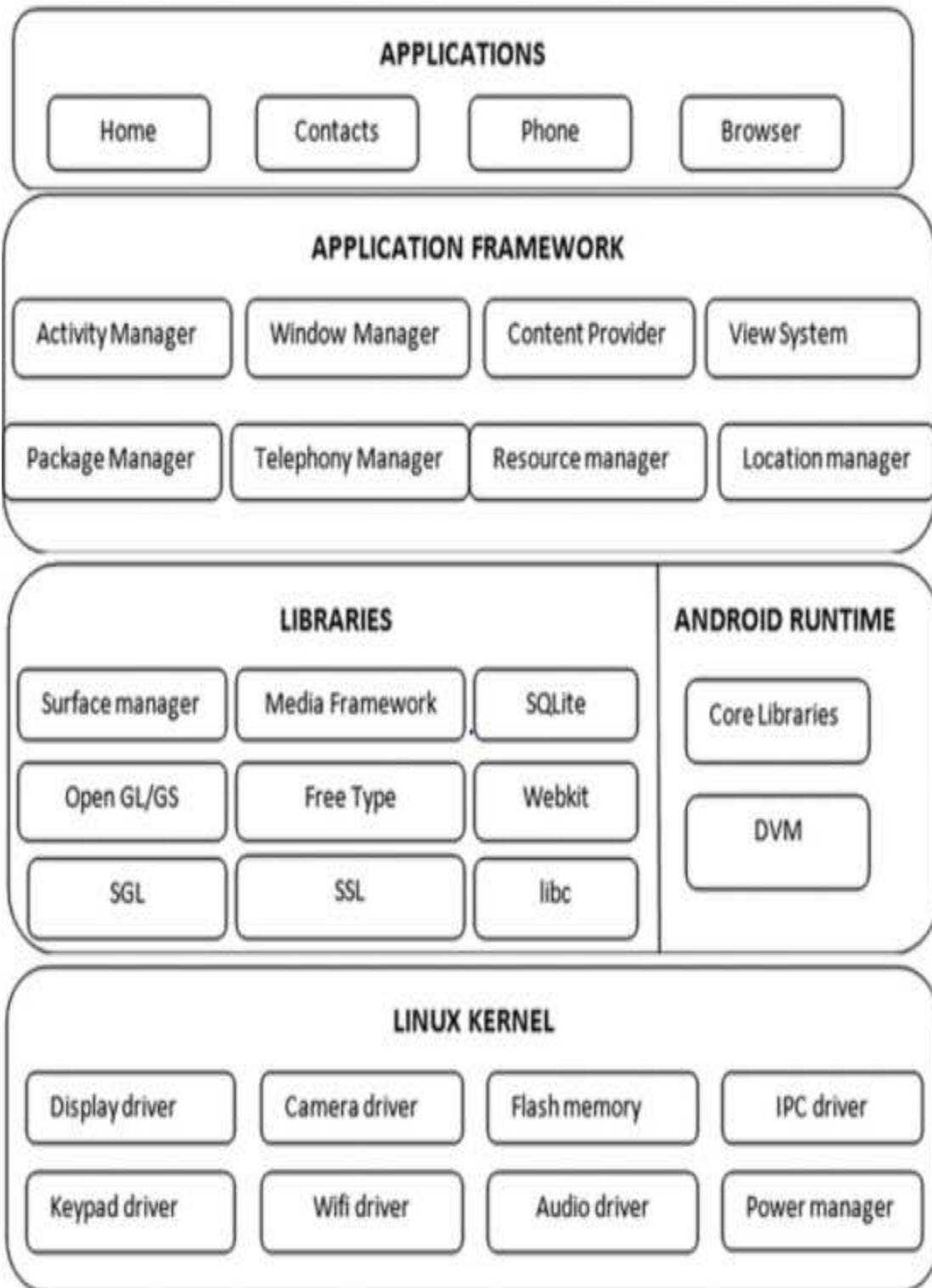
first round of the Android Developer Challenge. The revelation that Google was supplying new SDK releases to some developers and not others (and keeping this arrangement private) led to widely reported frustration within the Android developer community at the time.

On 18 August 2008 the Android 0.9 SDK beta was released. This release provided an updated and extended API, improved development tools and an updated design for the home screen. Detailed instructions for upgrading are available to those already working with an earlier release. ON 23 September 2008 the Android 1.0 SDK (Release 10 WAS RELEASED. According to the release notes, it included “mainly bug fixes, although some smaller features were added.” It also included several API changes from the 0.9 version. Multiple versions have been released since.

Enhancements to Android SDK go hand with the overall Android platform development. The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Development tools are downloadable components, so after one has downloaded the latest version and platform, older platforms and tools can also be downloaded for compatibility testing.

Android applications are packaged in .apk format and stored under/data/app folder on the Android OS (the folder is accessible to root users only for security reasons). The APK package contains .dex files (compiled byte code files called Dalvik executable), resource file, etc.

4.6 ANDROID ARCHITECTURE:



5.SYSTEM

CONFIGURATION

5. SYSTEM CONFIGURATION:

5.1 MODULES:

- ❖ User Login
- ❖ Home
 - Menu
 - Book Now
 - Customer Details
 - Book Now
 - Contact Us
 - About Us
- ❖ Track and Store record of appointment in SQLite Database

5.1.1 USER LOGIN:

User Login is for the user who visits our app and enters the details for further sign in. This is the entrance module where the user must reach and must sign in for further knowledge about our app.

5.1.2 HOME:

Home is used to denote the user that all are the services available in our beauty parlour in the form of images using image sliders. Image sliders are used to slide the images automatically.

5.1.2.1 MENU:

Users can select the services from the list of available in our beauty parlour with their costs. They can select them based on their need.

There are ten categories available in our app having as follows:

- Bridal Makeup: Rs.3000/-
- Facial –Golden: Rs.1200/-
- Facial-Diamond: Rs.1500/-
- Facial-Pearl: Rs.1300/-
- Mehendi: Rs.600/-
- Blouse Stitching: Rs.1000/-
- Jewellery for rent: Rs.1500/-
- Pedicure and Manicure: Rs.500/-
- Waxing: Rs.500/-
- Threading: Rs.40/-

5.1.2.2 CUSTOMER DETAILS:

Users should enter their details such as Name, Phone Number, Address, Select the date and time by using calendar and times respectively for makeup. Users should not select the previous dates of the registration.

5.1.2.2.1 BOOK NOW:

Users can select the above categories by clicking on the drop down menu and selecting the required need and clicking ok. Therefore, the registration gets success.

5.1.2.3 ABOUT US:

Users can know about the quality of products and the hygiene cosmetics which were used in our beauty parlour.

5.1.2.4 CONTACT US:

Users can contact us through phone or by our instagram id or users can reach our parlour by the address mentioned in our contact info.

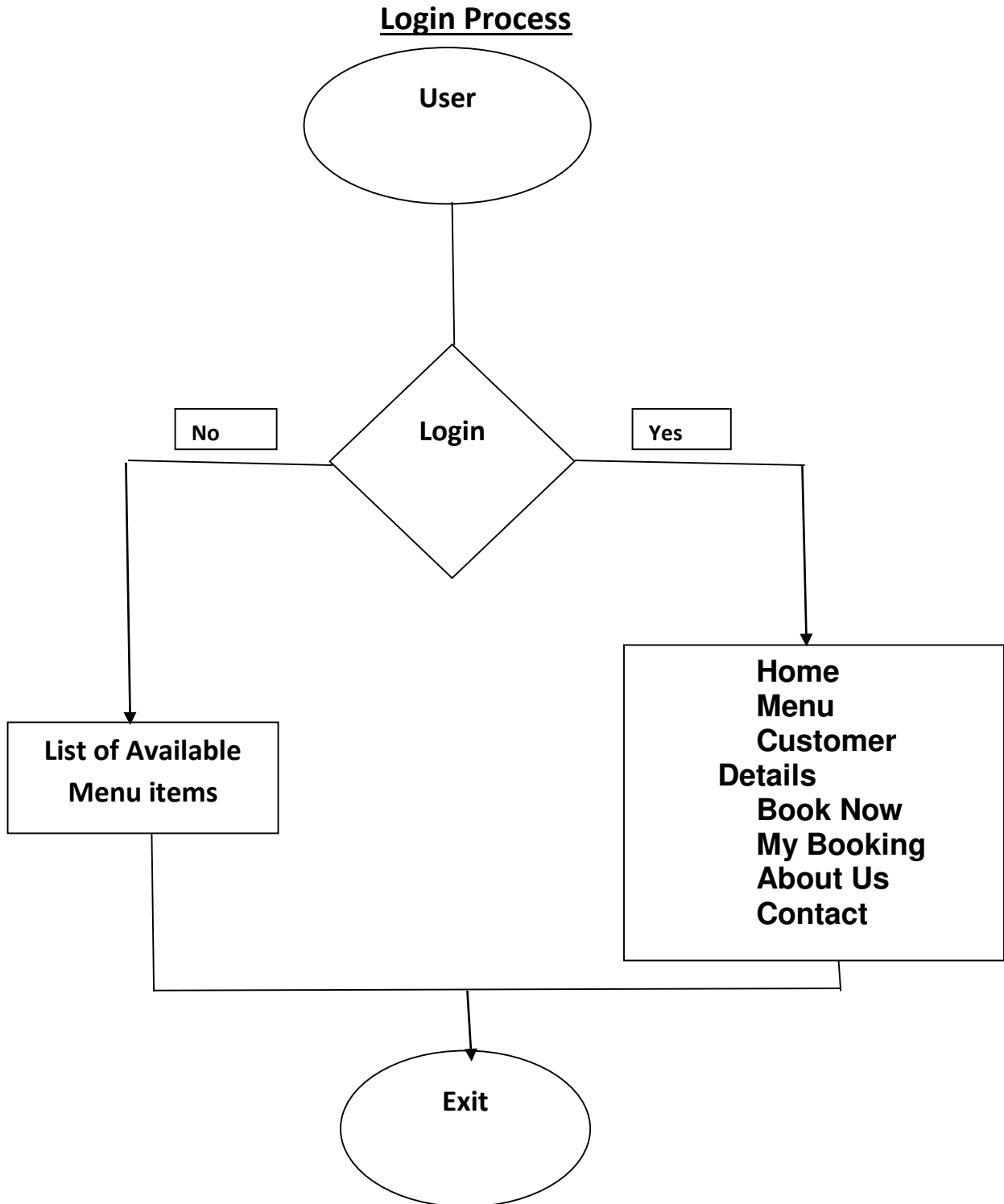
5.1.3 TRACK AND STORE RECORD OF APPOINTMENTS IN SQLITE DATABASE:

User appointments can be retrieved by using My Booking button which is in the Book Now Page. The users can check their details and appointments. These informations are stored and retrieved by SQLite database.

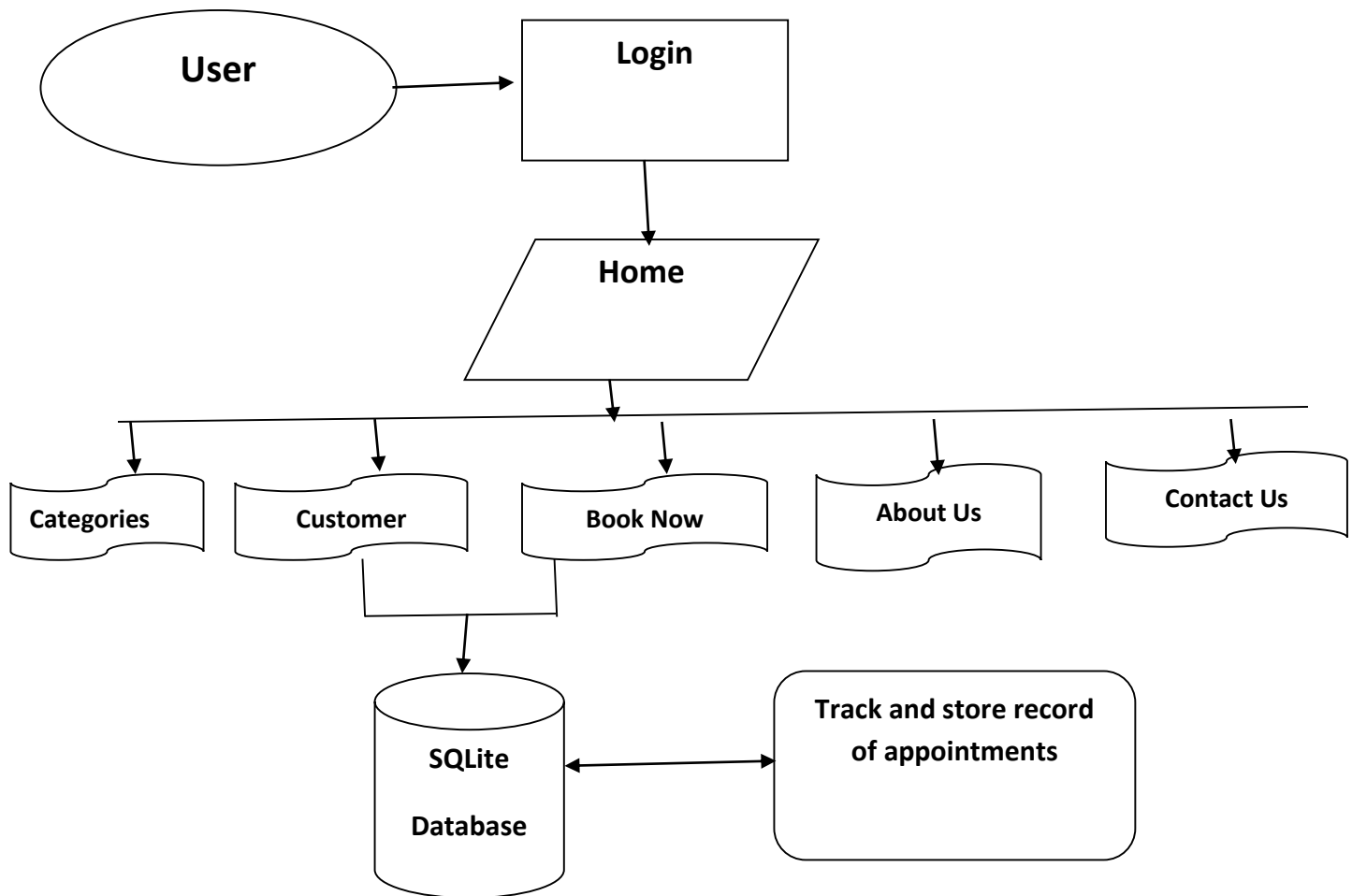
6.DATA FLOW

DIAGRAMS

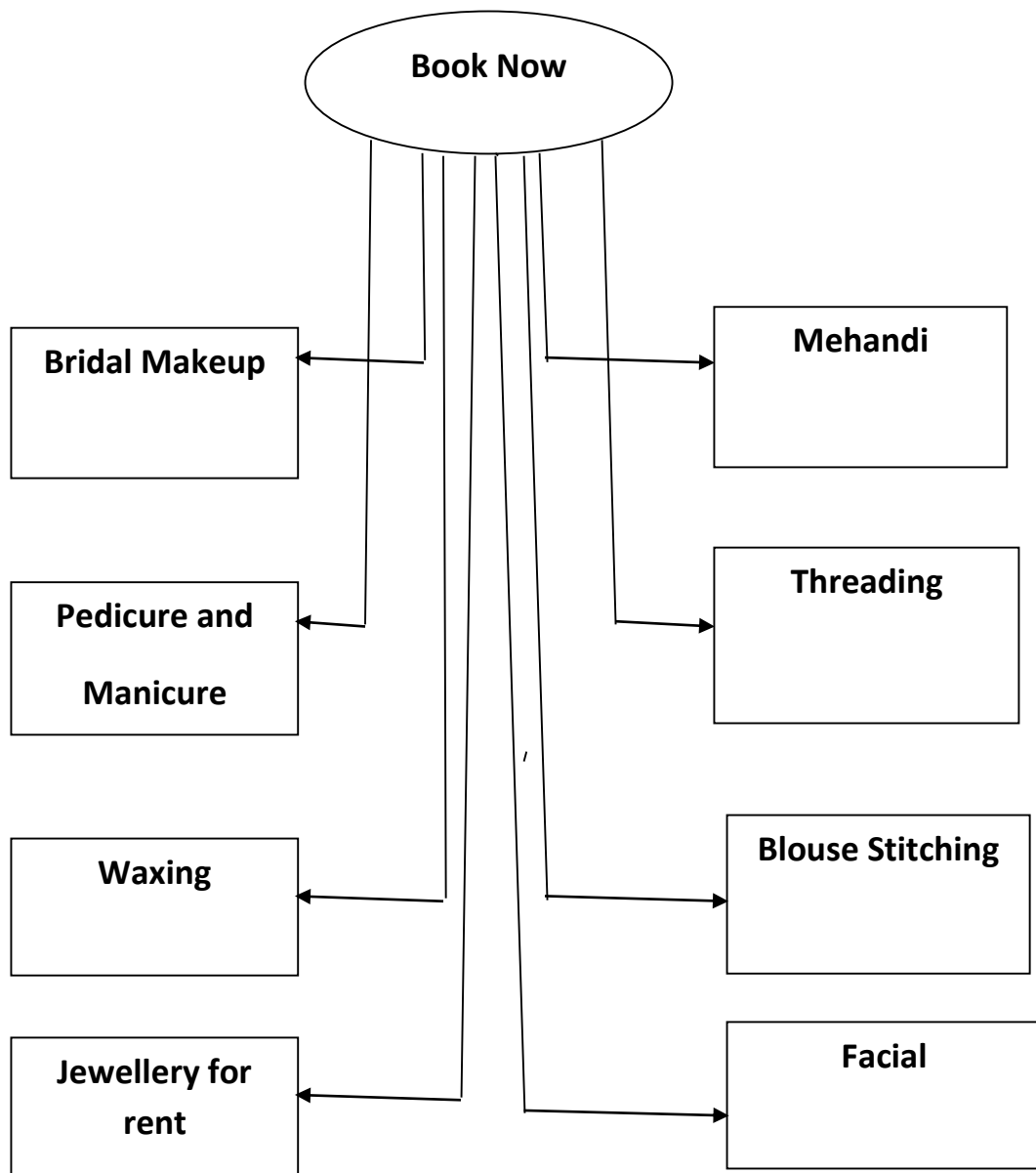
6.1 DFD FLOW CHART-1:



6.2 DFD FLOW CHART-2:



6.3 DFD FLOW CHART-3:



7.TESTING

IMPLEMENTATION

7.1 SYSTEM IMPLEMENTATION:

Implementation is the most crucial stage in achieving a successful system and giving the user's confidence that the new system is workable and effective. Implementation of a modified application to replace an existing one. This type of conversation is relatively easy to handle, provided there are no major changes in the system.

Each program is tested individually at the time of development using the data and has verified that this program linked together in the way specified in the programs specification, the computer system and its environment is tested to the satisfaction of the user. The system that has been developed is accepted and proved to be satisfactory for the user. A simple operating procedure is included so that the user can understand the different functions clearly and quickly.

7.2 TESTING:

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of conditions known as test cases and the output is evaluated to determine whether the program is performing as expected.

Software testing is the process of testing the functionality and correctness of software by running it. Process of executing a program with the intent of finding an error.

A good test case is one that has a high probability of finding an as yet undiscovered error. A successful test is one that uncovers an as yet

undiscovered error. Software testing is usually performed for two reasons.

- **Defect detection**
- **Reliability estimation**

7.3 TESTING OBJECTIVES:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an as yet undiscovered.
- A successful test is one that uncovers an as yet undiscovered error.

7.4 TESTING PRINCIPLES:

- All tests should be traceable to customer requirements.
- Tests should be planned large before testing begins.
- Testing should begin “In the Small” and progress towards “In the Large”.

7.5 TYPES OF TESTING:

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

7.5.1 UNIT TESTING:

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements.

Each module can be tested using the following two strategies:

7.5.1.1 BLACK BOX TESTING:

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been used to find errors in the following categories

- a) Incorrect or missing functions
- b) Interface errors
- c) Errors in data structure or external databases access
- d) Performance errors
- e) Initialization and termination errors.

In this testing only the output is checked for correctness. The logical flow of the data is not checked.

7.5.1.2 WHITE BOX TESTING:

White box testing techniques analyze the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing.

Working process of white box testing:

- **Input:** Requirements, Functional specifications, design documents, source code.

- **Processing:** Performing risk analysis for guiding through the entire process.
- **Proper test planning:** Designing test cases so as to cover entire code. Execute rinse-repeat until error-free software is reached. Also, the results are communicated.
- **Output:** Preparing final report of the entire testing process.

7.5.2 INTEGRATING TESTING:

Integration testing ensures that software and subsystem work together as a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated together.

7.5.3 SYSTEM TESTING:

Involves in-house testing of the entire system before delivery to the user. Its aim is to satisfy the user that the system meets all requirements of the client's specifications.

7.5.4 ACCEPTANCE TESTING:

It is a pre-delivery testing in which the entire system is tested at the client's site on real world data to find errors.

7.5.5 VALIDATION:

The system has been tested and implemented successfully and thus ensured that all the requirements as listed in the software requirements specification are completely fulfilled. In case of erroneous input corresponding error messages are displayed.

7.5.6 COMPILING TEST:

It was a good idea to do our stress testing early on, testing because it gave us time to fix some of the unexpected deadlocks and stability problems that only occurred when components were exposed to very high transaction volumes.

7.5.7 EXECUTION TEST:

This program was successfully loaded and executed. Because of good programming there were no execution errors.

7.5.8 OUTPUT TEST:

The successful output screens are placed in the output screens section above.

8.FUTURE **ENHANCEMENTS**

8.1 SCOPE:

This app is specially designed for the bridal women to look gorgeous on her big day. Our makeup artist does the best makeover and designs your costumes upto your satisfaction with minimum budget. We use good quality products that don't irritate your skin in anyways all the products and cream used are rich and organic. We can also develop our app with your reviews and by extending more pages in our studio upto your likeness and satisfaction.

Although our project has been fully developed there are several plans for the enhancement of this project in future.

- To make our app popular all over the world.
- To have more collections and stocks available.
- To establish a notification message if it is booked or not on the expected days.
- To reach millions by doing our services the best.

9.CONCLUSION

9.1 CONCLUSION:

The Bridal Booking App project that we have developed makes it easy and stress-free for the bridal women. She can place all her orders needed for her wedding under our bridal makeup booking app.

Working on the project as a team was a good experience. We understood the importance of planning and designing as a part of software development but it's very difficult to do the program as a single person. Developing the project has helped us to experience the real-time development procedures.

9.2 BENEFITS OF THE BRIDAL MAKEUP BOOKING APP:

- Growing list of beauticians, stylists and beauty professionals.
- Only vetted beauty professionals with 100% verified user reviews.
- Books appointment in just a few taps as per our schedule.
- Good services offered for the satisfaction of our customers are Facial, Mehendi, Blouse, Pedicure, Manicure, Bridal Makeup etc.

10.CODINGS

SPLASH ACTIVITY

```
package com.example.makeupstudio;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.Window;
import android.view.WindowManager;

public class MainActivity extends AppCompatActivity {
    private static int SPLASH_TIMEOUT = 5000;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        getSupportActionBar().hide();

        this.getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.activity_main);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent homeIntent=new Intent(MainActivity.this,
                    Activity2.class);
                startActivity(homeIntent);
                finish();
            }
        },SPLASH_TIMEOUT);
    }
}
```

LOGIN ACTIVITY

```
package com.example.makeupstudio;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.database.Cursor;
import android.util.Patterns;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Activity2 extends AppCompatActivity {

    EditText email,password;
    Button login;
    boolean isEmailValid,isPasswordValid;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_2);

        email=(EditText)findViewById(R.id.emailId);
        password=(EditText)findViewById(R.id.pwd);
        login=(Button)findViewById(R.id.button);

        login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (email.getText().toString().isEmpty()) {
                    Toast.makeText(getApplicationContext(),"Please
enter email address",Toast.LENGTH_LONG).show();
                    isEmailValid = false;
                } else if
(!Patterns.EMAIL_ADDRESS.matcher(email.getText().toString()).matches())
            } {
```

```

        Toast.makeText(getApplicationContext(), "Invalid
email address", Toast.LENGTH_LONG).show();
        isEmailValid = false;
    } else {
        isEmailValid = true;
    }
    if (password.getText().toString().isEmpty()) {
        Toast.makeText(getApplicationContext(), "Please
enter Password", Toast.LENGTH_LONG).show();
        isPasswordValid = false;
    } else if (password.getText().length() < 8) {
        Toast.makeText(getApplicationContext(), "Password
should have atleast 8 characters", Toast.LENGTH_LONG).show();
        isPasswordValid = false;
    } else {
        isPasswordValid = true;
    }
    if (isEmailValid && isPasswordValid) {
        Toast.makeText(getApplicationContext(), " Login
Successfully", Toast.LENGTH_SHORT).show();
        Intent myIntent=new
Intent(Activity2.this,Activity3.class);
        startActivity(myIntent);

    }

}

});
}

}

```


SLIDER VIEW ACTIVITY

```
package com.example.makeupstudio;
import com.smarteist.autoimageslider.SliderView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.FrameLayout;
import android.widget.TextView;
import java.util.ArrayList;
import java.util.List;

import com.google.android.material.bottomnavigation.BottomNavigationView;

public class Activity3 extends AppCompatActivity {
    FrameLayout frameLayout;
    BottomNavigationView navigationView;
    TextView textView;

    String url1 = "https://www.stayathomemum.com.au/cache/860x380-0/wp-content/uploads/2013/06/bigstock-Young-Woman-Having-Professiona-244205575.jpg";
    String url2 = "https://static-01.daraz.pk/p/ec2f84401d3d2edeea17be71e3a41b06.jpg";
    String
url3 = "https://miro.medium.com/max/900/0\*mLKNw1Z4ks8g\_koo.jpg";
    String
url4 = "https://media.weddingz.in/images/02018a79dc16133ff903c7a10ab72692/latest-mehndi-designs-for-the-2018-bride.jpg";
    String
url5 = "https://i.pining.com/474x/59/50/46/595046a24d6d80b717c1c15a3f18b897.jpg";
    String url6 = "https://www.igracebeauty.com/images/pedicure-manicure-services-in-vizag.jpg";
    String url7 = "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT2NzsHDaoYlCW83UWFY3UdhMWCj6DU4BPcQ&usqp=CAU";
    String url8 = "https://www.radarmagazine.com/wp-content/uploads/2019/06/how-to-use-an-epilator.png";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```

setContentView(R.layout.activity_3);
textView=(TextView)findViewById(R.id.content);
ArrayList<SliderData> sliderDataArrayList = new ArrayList<>();
SliderView sliderView = findViewById(R.id.slider);
sliderDataArrayList.add(new SliderData(url1));
sliderDataArrayList.add(new SliderData(url2));
sliderDataArrayList.add(new SliderData(url3));
sliderDataArrayList.add(new SliderData(url4));
sliderDataArrayList.add(new SliderData(url5));
sliderDataArrayList.add(new SliderData(url6));
sliderDataArrayList.add(new SliderData(url7));
sliderDataArrayList.add(new SliderData(url8));

// passing this array list inside our adapter class.
SliderAdapter adapter = new SliderAdapter(this,
sliderDataArrayList);

// below method is used to set auto cycle direction in left to
// right direction you can change according to requirement.

sliderView.setAutoCycleDirection(SliderView.LAYOUT_DIRECTION_LTR);

// below method is used to
// setadapter to sliderview.
sliderView.setSliderAdapter(adapter);
sliderView.setScrollTimeInSec(3);

// to set it scrollable automatically
// we use below method.
sliderView.setAutoCycle(true);

// to start autocycle below method is used.
sliderView.startAutoCycle();
BottomNavigationView bottomNavigationView =
(BottomNavigationView) findViewById(R.id.nav_view);
bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem
item) {
        switch (item.getItemId()) {
            case R.id.menu:
                Intent myIntent=new
Intent(Activity3.this,ScrollingActivity.class);
                startActivity(myIntent);
                break;

```

```

        case R.id.book:
            Intent cusIntent=new
Intent(Activity3.this,CustomerDetails.class);
            startActivity(cusIntent);
            break;
        case R.id.about:
            Intent abtIntent=new
Intent(Activity3.this,AboutUs.class);
            startActivity(abtIntent);
            break;
        case R.id.contact:
            Intent cntIntent=new
Intent(Activity3.this,Contact.class);
            startActivity(cntIntent);
            break;

    }

    return true;

    }
});
}
}

```

SLIDER VIEW ADAPTER

```
package com.example.makeupstudio;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;
import com.smarteist.autoimageslider.SliderViewAdapter;

import java.util.ArrayList;
import java.util.List;

public class SliderAdapter extends SliderViewAdapter<SliderAdapter
.SliderAdapterViewHolder> {

    // list for storing urls of images.
    private final List<SliderData> mSliderItems;

    // Constructor
    public SliderAdapter(Context context, ArrayList<SliderData>
sliderDataArrayList) {
        this.mSliderItems = sliderDataArrayList;
    }

    // We are inflating the slider_layout
    // inside on Create View Holder method.
    @Override
    public SliderAdapterViewHolder onCreateViewHolder(ViewGroup parent) {
        View inflate = LayoutInflater.from(parent.getContext()).inflate
(R.layout.slider_layout, null);
        return new SliderAdapterViewHolder(inflate);
    }

    // Inside on bind view holder we will
    // set data to item of Slider View.
    @Override
```

```

    public void onBindViewHolder(SliderAdapterViewHolder viewHolder,
final int position) {

        final SliderData sliderItem = mSliderItems.get(position);

        // Glide is use to load image
        // from url in your imageview.
        Glide.with(viewHolder.itemView)
            .load(sliderItem.getImgUrl())
            .fitCenter()
            .into(viewHolder.imageViewBackground);
    }

    // this method will return
    // the count of our list.
    @Override
    public int getCount() {
        return mSliderItems.size();
    }

    static class SliderAdapterViewHolder extends SliderViewAdapter
.ViewHolder {
        // Adapter class for initializing
        // the views of our slider view.
        View itemView;

        ImageView imageViewBackground;

        public SliderAdapterViewHolder(View itemView) {
            super(itemView);
            imageViewBackground = itemView.findViewById(R.id.myimage);
            this.itemView = itemView;
        }
    }
}

```

SLIDER DATA

```
package com.example.makeupstudio;
public class SliderData {

    // image url is used to
    // store the url of image
    private String imgUrl;

    // Constructor method.
    public SliderData(String imgUrl) {
        this.imgUrl = imgUrl;
    }

    // Getter method
    public String getImgUrl(){
        return imgUrl;
    }

    // Setter method
    public void setImgUrl(String imgUrl){
        this.imgUrl = imgUrl;
    }
}
```

DATABASE HELPER

```
package com.example.makeupstudio;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class BackendHelper extends SQLiteOpenHelper {
    public static final String DATABASE_NAME="booking.db";
    public static final String TABLE_NAME="booking_table";
    public static final String COL_1="MAKEUP";
    public static final String COL_2="MEHANDI";
    public static final String COL_3="THREADING";
    public static final String COL_4="PEDIMANI";
    public static final String COL_5="WAXING";
    public static final String COL_6="BLOUSE";
    public static final String COL_7="JEWELLERY";
    public static final String COL_8="FACIAL";
    public static final String COL_9="NAME";
    public static final String COL_10="NUMBER";
    public static final String COL_11="ADDRESS";
    public static final String COL_12="DATE";
    public static final String COL_13="TIME";
    public BackendHelper(Context context){
        super(context,DATABASE_NAME,null,3);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("create table "+TABLE_NAME+"(MAKEUP TEXT,MEHANDI TEXT,THREADING TEXT,PEDIMANI TEXT,WAXING TEXT,BLOUSE TEXT,JEWELLERY TEXT,FACIAL TEXT,NAME TEXT,NUMBER TEXT,ADDRESS TEXT,DATE TEXT,TIME TEXT)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME);
        onCreate(db);
    }
    public boolean insertData(String spinner1,String spinner2,String
```

```

spinner3,String spinner4,String spinner5,String spinner6,String
spinner7,String spinner8,String a,String b,String c,String d,String
e){
    SQLiteDatabase db=this.getWritableDatabase();
    ContentValues contentValues=new ContentValues();
    contentValues.put(COL_1,spinner1);
    contentValues.put(COL_2,spinner2);
    contentValues.put(COL_3,spinner3);
    contentValues.put(COL_4,spinner4);
    contentValues.put(COL_5,spinner5);
    contentValues.put(COL_6,spinner6);
    contentValues.put(COL_7,spinner7);
    contentValues.put(COL_8,spinner8);
    contentValues.put(COL_9,a);
    contentValues.put(COL_10,b);
    contentValues.put(COL_11,c);
    contentValues.put(COL_12,d);
    contentValues.put(COL_13,e);

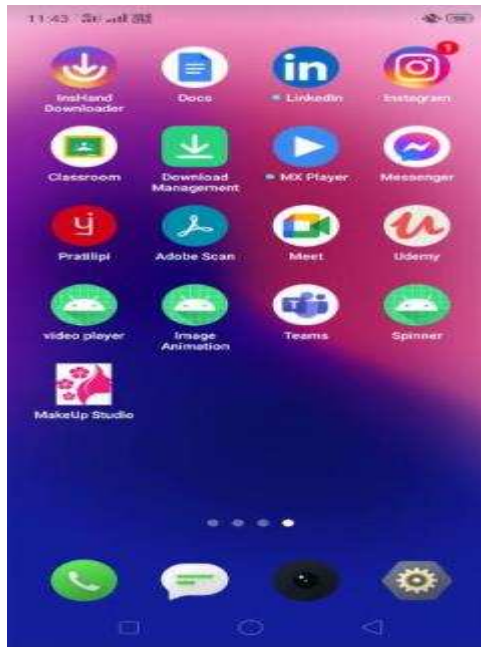
    long result=db.insert(TABLE_NAME,null,contentValues);
    if (result==-1)
        return false;
    else
        return true;
}
public Cursor getAllData(){
    SQLiteDatabase db=this.getWritableDatabase();
    Cursor cursor=db.rawQuery("Select*from "+TABLE_NAME,null);

    return cursor;
}
}

```


11.SCREENSHOTS

11.1 ICON



11.2 SPLASH SCREEN



11.3 LOGIN PAGE



11.4 SLIDER VIEW



11.5 MENU



11.6 CUSTOMER DETAILS



11.7 CALENDAR



11.8 TIME



11.9 ABOUT US



11.10 CONTACT US



11.11 DATABASE



12.BIBLIOGRAPHY

12.1 REFERENCE BOOKS:

- **“Java: A Beginner's Guide”** Book by Herbert Schildt.
- **“Beginning Android programming with Android Studio”** Book by J.F. DiMarzio.

12.2 REFERENCE LINKS:

- www.tutorialspoint.com
- www.stackoverflow.com
- www.geeksforgeeks.org.com

XX

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is certify that the project entitled

FARMINGTON

Submitted by

| | | |
|----------------------|---|----------|
| Joseph Amala Lincy.B | - | 18AUCS14 |
| Kathija Apsana.M | - | 18AUCS15 |
| Nisha Rani.P | - | 18AUCS32 |
| Silvia Anusha.P | - | 18AUCS44 |

In partial fulfilment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.

J. Amala Lincy
Guide

S. Gnaniy
Head of the Department

Submitted for the examination held on 15.04.2021.

Examiner

Lucia Rose
Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty God for his gracious help and blessings to finish our project successfully.

We are infinitely indebted to our college principal Dr. Sr. A. S. J. Lucia Rose M.SC, M.Phil., PGDCA, Ph.D., for providing us necessary infrastructure to undertake this project and work successfully.

We are so grateful to Mrs. S. Gnanathangam M.C.A, M.Phil (Associate professor & HOD), Department of Computer Science, for her gracious guidance. Our heartfelt thanks to our project guide Mrs.S. Andrews Felicita M.Sc.,M.B.A, (Assistant Professor), Department of Computer Science, for her valuable assistance and kindness with which she reviewed our project.

We also express our boundless thanks to our lecturers,

- Mrs.M. Jothimani B.E., M.Sc., M.Phil., (Associate professor)
- Sr.K.Reetha M.Sc.,M.Phil.(Assistant professor)
- Mrs.J. Anila Maily M.Sc , M.Phil. (Associate Professor),

for their kind cooperation in successful completion of the project.

Index

| S.NO. | CONTENT | PAGE.NO |
|-------|--|---------|
| 1 | Introduction <ul style="list-style-type: none">○ Objective | 6 |
| 2 | System Analysis <ul style="list-style-type: none">○ System Analysis○ Existing System○ Proposed System | 8 |
| 3 | Project Description | 12 |

| | | | |
|---|---|---|----|
| | | | |
| | 4 | System Specifications <ul style="list-style-type: none">● Hardware Requirement● Software Requirements● Network Requirements● Software Description | 14 |
| 4 | | | |

| | | |
|-----------|---|-----------|
| 5 | System Configuration <ul style="list-style-type: none">• Modules | 19 |
| 6 | Data Flow Diagram | 22 |
| 7 | TestingImplementation | 27 |
| 8 | Coding | 30 |
| 9 | Screenshots | 62 |
| 10 | Future Enhancements | 67 |
| 11 | Conclusion | 69 |
| 12 | Bibliography | 71 |

INTRODUCTION

OBJECTIVE:

The emergence and application of information and communication, technologies have changed the tools that people use in their daily life. However not all the collectives use the technology in the same way. One case to take especially into account is older people. For them technology should be an inclusive factor but it can be also exclusive. With flexible apps for the common people, with adaptations that would make older peoples' lives easier this app is created. A 2018 survey of adults found that 76% of those aged 50 and older wanted to continue to live in their current residence as long as possible. Seniors who want to age at home require a variety of services including transportation, food delivery, medicines, groceries, housekeeping, etc. When things settle down, the app's primary aim is to provide a piece of mind "for senior citizens as well as their children" living away or abroad. Their children can look at the app, and be in the loop. They no longer have to ask favours from neighbours, and can live their daily life independently.

As the official Integrated Development Environment for all Flutter applications, Android Studio seems to be at the top list of preferred tools for developers. Our idea is to develop an E-Services App by using Android Development Toolkit.

SYSTEM ANALYSIS

System analysis:

System analysis is the detailed study of various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optional or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to the system.

- **Technical Feasibility**
- **Economic Feasibility**
- **Physical Feasibility**

Technical Feasibility:

The System Developer's task is to view the needed capabilities in light of currently available technologies. Our app works hand in hand with high technologies. Security and usability are considered as the most important factors of the system design that increases the user friendliness of the system. A database has to be maintained in order to update and backup whenever necessary. To create a database, we used Firebase.

Economic Feasibility:

In making recommendations, a study of the economics of the proposed system should be made. Even though finding out the cost of the proposed project is difficult we assume and estimate the costs and benefits as follows:

It is a cost effectively designed app and it can be categorized in two ways.

- Cost associated with the services.
- Cost associated with the Android OS.

Physical Feasibility:

A system resembling a mini authentication activity which is physically a less component space to make the system with high potential security. So the desired E-Service App is physically feasible indeed.

Existing System:

E-Service App is only for ordering foods and booking several services. Online payments such as credit or debit cards are not possible.

Drawbacks of Existing System:

- Online Payments are not possible.
- Limited stocks available.
- Through online, once an order is placed it cannot be cancelled.

PROPOSED SYSTEM:

SPARKCART is a mobile-based e-service app. E-Services are services which use information and communication technologies. The three main components of e-services are service provider, service receiver and the channels of service delivery. It connects users and admin in an online community allowing users to select the services and book them. Admin can view the bookings and can hire people suitable for those services.

Our project has used Flutter and Firebase to back the interface with strong database functionality. For bookings, SPARKMART has many categories like checkboxes for selection, voice notes, prescription images and this will integrate the Firebase as well as front-end booking interface.

Some of the major cases include user login, booking services, ordering foods using our Firebase database.

ADVANTAGES:

- Can be used by all mobile users
- Less memory needed to run this app
- Inventory management
- Effective Marketing.

PROJECTDESC RIPTION

PROJECT DESCRIPTION:

The emergence and application of the information and communication technologies have changed the tools that people use in their daily life. However not all the collectives use the technology in the same way. One case to take especially into account is older people. For them technology should be an inclusive factor but it can be also exclusive. This app has some special abilities and features for older people; moreover they do not always meet their needs. In order to satisfy the daily needs this app has been created for their easy use.

Our project has used Flutter and Firebase to back the interface with strong database functionality. For bookings, SPARKMART has many categories like checkboxes for selection, voice notes, prescription images and this will integrate the Firebase as well as front-end booking interface. It connects users and admin in an online community allowing users to select the services and book them. Admin can view the bookings and can hire people suitable for those services.

SYSTEM SPECIFICATIONS

SOFTWARE REQUIREMENT:

- FRONT END: Android Studio/Flutter
- CODING: Dart
- BACK END: Firebase
- OPERATING SYSTEM: Windows XP/10

HARDWARE REQUIREMENT:

- INTEL DUAL CORE 2.0Ghz CPU
- 2GB RAM
- 320GB HDD
- Mobile/Tablet

NETWORK REQUIREMENT:

- Internet Connection

SOFTWARE DESCRIPTION:

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance and commercially sponsored by Google. It was unveiled in November 2007, with the first commercial Android device launched in September 2008. It is free and open source software; its source code is known as Android Open Source Project (AOSP), which is primarily licensed under the Apache License. However most Android devices ship

with additional proprietary software pre-installed, most notably Google Mobile Services (GMS) which includes core apps such as Google Chrome, the digital distribution platform Google Play and associated Google Play

Services development platform. Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of January 2021, the Google Play Store features over 3 million apps. The current stable version is Android 11, released on September 8, 2020.

SOFTWARE DEVELOPMENT KIT:

Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.

The first version of Flutter was known as codename "Sky" and ran on the Android operating system. It was unveiled at the 2015 Dart developer summit, with the stated intent of being able to render consistently at 120 frames per second. During the keynote of Google Developer Days in Shanghai, Google announced Flutter Release Preview 2, which is the last big release before Flutter 1.0. On December 4, 2018, Flutter 1.0 was released at the Flutter Live event, denoting the first "stable" version of the Framework. On December 11, 2019, Flutter 1.12 was released at the Flutter Interactive event.

On May 6, 2020, the Dart SDK in version 2.8 and the Flutter in version 1.17.0 were released, where support was added to the Metal API, improving

performance on iOS devices (approximately 50%), new Material widgets, and new network tracking.

On March 3, 2021, Google released Flutter 2 during an online Flutter Engage event. This major update brought official support for web-based applications as well as early-access desktop application support for Windows, MacOS, and Linux.

Framework architecture:

- Dart platform
- Flutter engine
- Foundation library
- Design-specific widgets

Dart platform

Flutter apps are written in the Dart language and make use of many of the language's more advanced features.

On Windows, macOS, and Linux Flutter runs in the Dart virtual machine, which features a just-in-time execution engine. While writing and debugging an app, Flutter uses Just In Time compilation, allowing for "hot reload", with which modifications to source files can be injected into a running application. Flutter extends this with support for stateful hot reload, where in most cases changes to source code are reflected immediately in the running app without requiring a restart or any loss of state.

Release versions of Flutter apps are compiled with ahead-of-time (AOT) compilation on both Android and iOS, making Flutter's high performance on mobile devices possible.

Flutter engine

Flutter's engine, written primarily in C++, provides low-level rendering support using Google's Skia graphics library. Additionally, it interfaces with platform-specific SDKs such as those provided by Android and iOS. The Flutter Engine is a portable runtime for hosting Flutter applications. It implements Flutter's core libraries, including animation and graphics, file and network I/O, accessibility support, plugin architecture, and a Dart runtime and compile toolchain. Most developers interact with Flutter via the Flutter Framework, which provides a reactive framework and a set of platform, layout, and foundation widgets.

Foundation library

The Foundation library, written in Dart, provides basic classes and functions that are used to construct applications using Flutter, such as APIs to communicate with the engine.

Design-specific widgets

The Flutter framework contains two sets of widgets that conform to specific design languages: Material Design widgets implement Google's design language of the same name, and *Cupertino* widgets implement Apple's iOS Human interface guidelines.

SYSTEM CONFIGURATION

SYSTEMCONFIGURATION:

MODULES:

- User Login
- Dashboard
- About us
- Announcements
- Contact us
- Rate us on PlayStore
- Logout

USER LOGIN:

User Login is for the user who visits our app. The details should be entered and the user will receive an OTP number. Using the given OTP number the login is completed. This is the entrance module where the user must sign in for further knowledge about our app.

Dashboard:

Dashboardpage gives you an overview of services available in the app.

There are 4 categories available in our app:

- Transport
 - Car
 - Bike
- Delivery
 - Food and Snacks
 - Grocery
 - Vegetables
 - Medicine
 - Fruits

- Household Services
 - Carpenter
 - Painter
 - Home Tuition
 - Cook
 - Electrician
 - Security
 - Plumber
 - Mechanic
 - Welder
- Services
 - Land sale
 - Catering
 - Housekeeping
 - Construction
 - Tuition
 - Event management

About Us:

User can know about the quality of service provided by the admin.

Announcements:

This gives information about the available offers or any advertisements based on the app for the users.

CONTACT US:

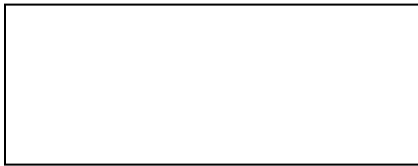
Users can contact us through the phone number mentioned in our contact info.

DATA-FLOW DIAGRAM

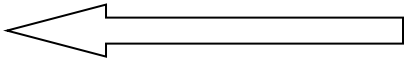
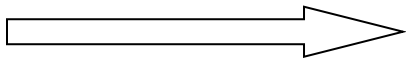
DATA-FLOW DIAGRAM:

Data flow is the movement of data from a specific origin to a specific destination indicated by line and arrows. Data flow diagram is the graphical representation of a data movement, processes and files used in the support of an information system.

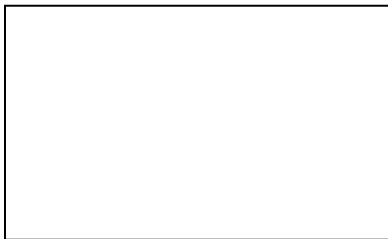
DATA-FLOW DIAGRAM SYMBOLS:



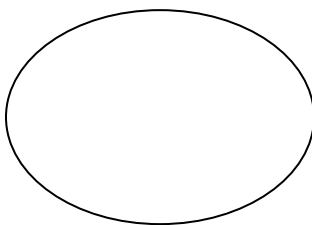
Source and Destination



Data Flow

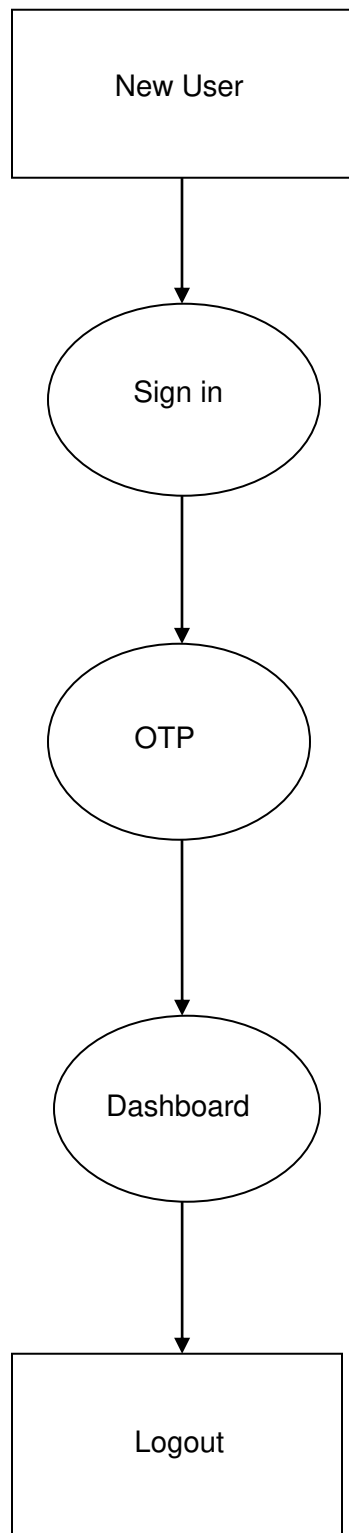


Database

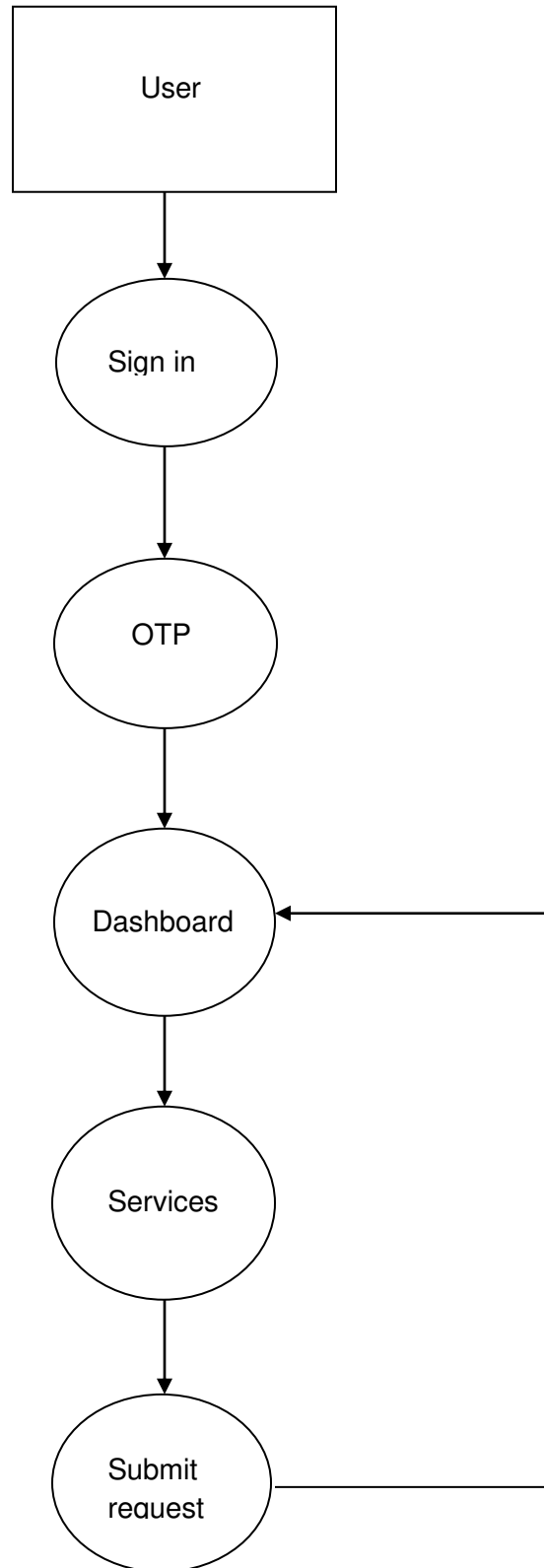


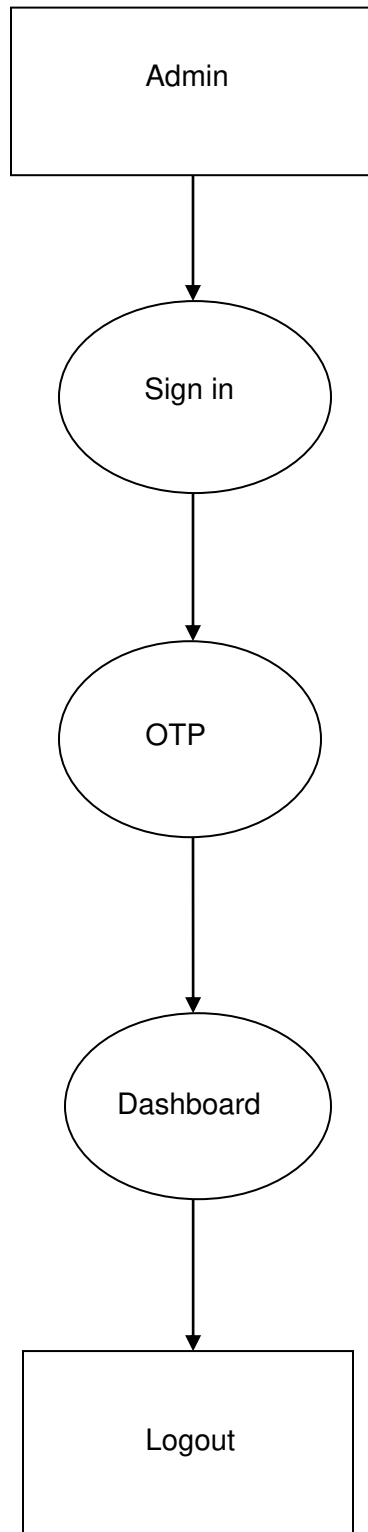
Process

New User:



User:





TESTING IMPLEMENTATION

27

SYSTEM IMPLEMENTATION:

Implementation is the most crucial stage in achieving a successful system and giving the user's confidence that the new system is workable and effective. Implementation of a modified application is to replace an existing one. This type of conversion is relatively easy to handle, provided there are no major changes in the system.

Each program is tested individually at the time of development using the data and has verified that this program linked together in the way specified in the programs specification, the computer system and its environment is tested to the satisfaction of the user. The app that has been developed is accepted and proved to be satisfactory for the user. A simple operating procedure is included so that the user can understand the different functions clearly and quickly.

TESTING:

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of conditions known as test cases and the output is evaluated to determine whether the program is performing as expected.

Software testing is the process of testing the functionality and correctness of software by running it. Process of executing a program with the intent of finding an error.

A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an

undiscovered error. Software testing is usually performed for two reasons.

- ~~Defect detection~~
- **Reliability estimation**

TESTING OBJECTIVES:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probability of finding an as yet undiscovered.
- A successful test is one that uncovers an undiscovered error.

TESTING PRINCIPLES:

- All tests should be traceable to customer requirements.
- Tests should be planned large before testing begins.
- Testing should begin “In the Small” and progress towards “In the Large”.

VALIDATION:

The system has been tested and implemented successfully and thus ensured that all the requirements as listed in the software requirements specification are completely fulfilled. In case of erroneous input, corresponding error messages are displayed.

Source Code

30

//login.dart

```

import 'package:eservices/home/homepage.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:eservices/login/phoneauth.dart';

class LoginPage extends StatefulWidget {
  final bool hasdata;
  LoginPage({this.hasdata=false});
  @override
  _LoginPageState createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  final GlobalKey<ScaffoldState> _scaffoldKey = new GlobalKey<ScaffoldState>();
  final formKey = new GlobalKey<FormState>();
  bool _isLoading = false;
  String phoneNo, verificationId, smsCode,name;
  bool codeSent = false;
  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: Scaffold(
        key: _scaffoldKey,
        body: ListView(
          children: <Widget>[
            Stack(
              children: <Widget>[
                Container(
                  width: MediaQuery.of(context).size.width-20,
                  child: Image.asset('assets/images/logo.jpg',
                    fit: BoxFit.cover,),
                ),
              ],
            ),
          ],
        ),
      ),
    );

    widget.hasdata?
    Positioned(
      right: 0.5,

```



```

      child:
        Column(
          children: <Widget>[
            IconButton(
              onPressed: () {
                Navigator.pushReplacement(context, MaterialPageRoute(builder:
(context)=>Homepage()),);

                },
              icon: Icon(Icons.home),
            ),
            Text('HOME')
          ],
        ),):
        Positioned(
          right: 0.5,
          child: FlatButton(onPressed: (){
            Navigator.pushReplacement(context, MaterialPageRoute(builder:
(context)=>Homepage()),);
          }, child: Text('Skip',
            style: TextStyle(
              fontSize:17.0,
            )),),
        )
      ],
    ),
    Container(
      alignment: Alignment.center,
      padding: EdgeInsets.all(41),
      child: Text(
        'Login with your credentials',

        style: TextStyle(
          fontSize: 25),
      )),

```

```
Container(
```

```
  alignment: Alignment.center,
```

```
  padding: EdgeInsets.all(10),
```

```
  child: Text(
```

```
    'Sign in',
```

```
    style: TextStyle(
```

```
      fontSize: 30),
```

```
  )),
```

```
Container(
```

```
  padding: EdgeInsets.all(10),
```

```
  child: Form(
```

```
    key: formKey,
```

```
    child: Column(
```

```
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
```

```
      children: <Widget>[
```

```
        TextFormField(
```

```
          autofocus: true,
```

```
          decoration: InputDecoration(
```

```
            border: OutlineInputBorder(
```

```
              borderRadius: BorderRadius.circular(35.0),
```

```
            ),
```

```
            labelText: 'Name',
```

```
            prefixIcon: Icon(Icons.person, color: Colors.black),
```

```
          ),
```

```
          validator: (val) => val.isEmpty ? 'Please enter your Name' : null,
```

```
          onSave: (val) => name = val
```

```
        ),
```

```
        SizedBox(height:20),
```

```
        TextFormField(
```

```
          initialValue: '+91',
```

```
          decoration: InputDecoration(
```

```
            border: OutlineInputBorder(
```

```
              borderRadius: BorderRadius.circular(35.0),
```

```

    ),
    labelText: 'Phone Number',
    prefixIcon: Icon(Icons.phone, color: Colors.black,),
  ),
  keyboardType: TextInputType.phone,
  validator: (val) => val.isEmpty ? 'Please enter your Phone Number' : null,
  onSaved: (val) => phoneNo = val,
), SizedBox(height: 20.0),
codeSent ? TextFormField(
  keyboardType: TextInputType.phone,
  decoration: InputDecoration(
    suffix: FlatButton(onPressed: ()=>verifyPhone(phoneNo), child:
Text('Resend OTP')),
    hintText: 'Enter OTP'),
  onChanged: (val) {
    setState(() {
      this.smsCode = val;
    });
  },
) : Container(),
SizedBox(height: 20.0),
Container(
  width: MediaQuery.of(context).size.width,
  child: FloatingActionButton.extended(
    label: codeSent ? Text('Verify'):Text('Send OTP'),
    onPressed: () async{
      final form = formKey.currentState;
      form.save();
      if(form.validate()){
        setState(() {
          _isLoading = true;
        });
      }

      if(codeSent == true){
        AuthService().signInWithOTP(smsCode,
verificationId,name,phoneNo,context);

```

```

        widget.hasdata==true??Navigator.pushReplacement(context,
MaterialPageRoute(builder: (context)=>Homepage()));
        setState(() {
            _isLoading = false;
        });

    }
    else{

        if(_isLoading==true){
            verifyPhone(phoneNo);
        }
        _scaffoldKey.currentState.showSnackBar(new SnackBar(
            duration: Duration(seconds: 8),
            content: new Text('Please Wait...'),),);
    }
    })),
    ),
    SizedBox(height: 20,),
    ],
    )),
    ),
    ],
    ),
    ),
    );
}

```

```

Future _buildErrorDialog(BuildContext context, _message) {
    return showDialog(
        builder: (context) {
            return AlertDialog(
                title: Text('Error Message'),
                content: Text(_message),

```

35

```

actions: <Widget>[
    FlatButton(
        child: Text('Cancel'),

```

```

        onPressed: () {
          Navigator.of(context).pop();
        })
      ],
    );
  },
  context: context,
);
}

```

```

Future<void> verifyPhone(phoneNo) async {
  final PhoneVerificationCompleted verified = (AuthCredential authResult) {
    AuthService().signIn(authCreds: authResult, phone: phoneNo, name: name);
  };

```

```

  final PhoneVerificationFailed verificationfailed =
    (AuthException authException) {
      return _buildErrorDialog(context, authException.message);
    };

```

```

  final PhoneCodeSent smsSent = (String verId, [int forceResend]) {
    this.verificationId = verId;
    setState(() {
      this.codeSent = true;
    });
  };

```

```

  final PhoneCodeAutoRetrievalTimeout autoTimeout = (String verId) {
    this.verificationId = verId;
  };

```

```

  await FirebaseAuth.instance.verifyPhoneNumber(
    phoneNumber: phoneNo,
    timeout: const Duration(seconds: 45),

```

36

```

verificationCompleted: verified,
verificationFailed: verificationfailed,

```

```

        codeSent: smsSent,
        codeAutoRetrievalTimeout: autoTimeout);
    }
}

```

//homepage.dart

```

import 'dart:async';
import 'package:eservices/home/about.dart';
import 'package:eservices/home/deals.dart';
import 'package:eservices/login/login.dart';
import 'package:eservices/login/phoneauth.dart';
import 'package:permission_handler/permission_handler.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:eservices/home/service.dart';
import 'package:eservices/home/announcement.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:flutter/rendering.dart';
import 'package:geolocator/geolocator.dart';
import 'package:android_intent/android_intent.dart';

```

```

class Homepage extends StatefulWidget {
  final FirebaseUser currentUser;
  Homepage({this.currentUser});
  @override
  _HomepageState createState() => _HomepageState();
}

```

37

```

class _HomepageState extends State<Homepage> {
  Future _checkGps() async {
    if (!(await Geolocator().isLocationServiceEnabled())) {

```

```

if (Theme.of(context).platform == TargetPlatform.android) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text("Can't get current location"),
        content:
          const Text('Please make sure you enable GPS and try again'),
        actions: <Widget>[
          FlatButton(
            child: Text('Enable GPS'),
            onPressed: () {
              final AndroidIntent intent = AndroidIntent(
                action: 'android.settings.LOCATION_SOURCE_SETTINGS');
              intent.launch();
              Navigator.of(context, rootNavigator: true).pop();
            },
          ),
        ],
      );
    },
  );
}

@override
void initState(){
  // TODO: implement initState
  super.initState();
  [Permission.microphone, Permission.storage, Permission.location].request();
  _checkGps();
}

38

@override
Widget build(BuildContext context) {
  return SafeArea(

```

```
child: Scaffold(
```

```
  drawer: Drawer(
```

```
    child: ListView(
```

```
      children: <Widget>[
```

```
        UserAccountsDrawerHeader(
```

```
          currentAccountPicture: Image.asset('assets/images/profile.png'),
```

```
          accountName: Center(
```

```
            child: widget.currentUser == null? Text('Hi User',
```

```
            style: TextStyle(
```

```
              fontSize: 20.0,
```

```
            ),): Text(
```

```
              'Hi ${widget.currentUser.displayName}',
```

```
            ),
```

```
          ),
```

```
          accountEmail: widget.currentUser == null? null: Text(
```

```
            widget.currentUser.phoneNumber
```

```
          ),
```

```
        ),
```

```
        ListTile(
```

```
          leading: Icon(Icons.dashboard),
```

```
          title: Text('Dashboard'),
```

```
          onTap: () {
```

```
            Navigator.of(context).pop();
```

```
          },
```

```
        ),
```

```
        Divider(),
```

```
        ListTile(
```

```
          leading: Icon(Icons.info_outline),
```

```
          title: Text('About us'),
```

```
          onTap: () {
```

```
            Navigator.push(context,
```

```
              MaterialPageRoute(builder: (context) => AboutUs()));
```

```
          },
```

```
39
```

```
),
```

```
  Divider(),
```

```
  ListTile(
```



```

        leading: Icon(Icons.volume_up),
        title: Text('Announcements'),
        onTap: () {
          Navigator.push(context,
            MaterialPageRoute(builder: (context) => AnnounceMents()));
        },
      ),
      Divider(),
      ListTile(
        leading: Icon(Icons.mail_outline),
        title: Text('Contact Us'),
      ),
      Divider(),
      ListTile(
        leading: Icon(Icons.star_half),
        title: Text('Share our App'),
        onTap: () {},
      ),
      Divider(),
      ListTile(
        leading: widget.currentUser == null
          ? Icon(Icons.person):
          Icon(Icons.exit_to_app),
        title:
          widget.currentUser == null ? Text('Login/signup') :
          Text('LogOut'),
        onTap: (){
          widget.currentUser == null
            ? Navigator.push(context,
              MaterialPageRoute(builder: (context) => LoginPage())):
            AuthService().signOut();
        },
      ),
    ),
    Divider(),
  ],

```

```

    ),
    appBar: AppBar(
      elevation: 0.0,
      title: Text('Dhanamuthra People Service'),
    ),
    body: StreamBuilder(
      stream: Firestore.instance
.collection('mainservices')
      .snapshots(),
      builder: (context, snapshot) {
        if (!snapshot.hasData) {
          return Center(
            child: CircularProgressIndicator(),
          );
        } else {
          return ListView(
            children: <Widget>[
              Container(
                height: 75,
                child: InkWell(
                  onTap: () {
                    Navigator.push(context, MaterialPageRoute(builder:
(context)=>Deals(user: widget.currentUser,)));
                  },
                  child: Card(
                    elevation: 2.0,
                    child: Center(
                      child: Row(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: <Widget>[
                          Icon(Icons.local_offer,
                            size: 20.0,),

                          Text("Todays Deals",
                            style: TextStyle(
                              color: Colors.blue[700],

```

```

        fontSize: 20.0,
        fontWeight: FontWeight.bold,
      ),),
    ],
  ),),
),
),
),
Container(
  height: 280,
  child: Image.asset('assets/images/banner.jpeg',
fit: BoxFit.cover)
),
GridView.builder(
  physics: NeverScrollableScrollPhysics(),
  shrinkWrap: true,
  itemCount: snapshot.data.documents.length,
  gridDelegate:
    SliverGridDelegateWithFixedCrossAxisCount(
      crossAxisCount: 2),
  itemBuilder: (context, index) {
    DocumentSnapshot data =
      snapshot.data.documents[index];
    return Column(
      children: <Widget>[
        Container(
          height: 150,
          width: 250,
          child: InkWell(
            onTap: (){
              Navigator.push(
                context,

```

42

```

MaterialPageRoute(
  builder: (context) =>
    ServicePage(name: data,service:

```

```

data['name'],currentUser: widget.currentUser));
    },
    child: Card(
      shape: RoundedRectangleBorder(
        borderRadius:
          BorderRadius.circular(3.0),
      ),
      child: Image.network(
        data['image'],
        fit: BoxFit.cover,
      )),
    ),
  ),
  Container(child: Text(data['name']))
],
);
}),
],
);
}
}),
),
);
}
}

```

//AdminHome.dart

```

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:eservices/Admin/Viewimage.dart';
import 'package:eservices/Admin/announcement.dart';

```

43

```

import 'package:eservices/Admin/audio.dart';
import 'package:eservices/Admin/user.dart';

```

```

import 'package:eservices/admin/adminservice.dart';
import 'package:eservices/adminservices/newrequest.dart';
import 'package:eservices/adminservices/ongoing.dart';
import 'package:eservices/home/deals.dart';
import 'package:eservices/login/phoneauth.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:firebase_messaging/firebase_messaging.dart';
import 'package:flutter/material.dart';
import 'package:flutter_slidable/flutter_slidable.dart';
import 'package:eservices/Admin/reports.dart';
import 'package:permission_handler/permission_handler.dart';
import 'history.dart';

```

```

class AdminHome extends StatefulWidget {
  final FirebaseUser currentUser;
  AdminHome({this.currentUser});
  @override
  _AdminHomeState createState() => _AdminHomeState();
}

```

```

class _AdminHomeState extends State<AdminHome> {
  final AuthService _auth = AuthService();
  final FirebaseMessaging _firebaseMessaging = FirebaseMessaging();
  @override
  void initState() {
    _firebaseMessaging.subscribeToTopic('admin');
    super.initState();
    _firebaseMessaging.configure(
      onMessage: ((message) {
        print(message['notification']['title']);
        return;
      }),
    ),
  },

```

44

```

onLaunch: ((message) {
  print(message['notification']['title']);

```

```

    return;
  }},
  onResume: ((message) {
    print(message['notification']['title']);
    return;
  }},
);
}

```

```

@override
Widget build(BuildContext context) {
  return SafeArea(
    child: Scaffold(
      drawer: Drawer(
        child: ListView(
          children: <Widget>[
            UserAccountsDrawerHeader(
              accountName: Center(
                child: Text(
                  'Admin',
                  style: TextStyle(
                    fontSize: 22.0,
                  ),
                ),
              ),
              accountEmail: null),
            ListTile(
              leading: Icon(Icons.dashboard),
              title: Text('Dashboard'),
              onTap: () {
                Navigator.of(context).pop();
              },
            ),

```

45

```

Divider(),
ListTile(

```

```

leading: Icon(Icons.settings),
title: Text('Services'),
onTap: () {
  Navigator.push(context,
    MaterialPageRoute(builder: (context) => AdminService()));
},
),
Divider(),
ListTile(
  leading: Icon(Icons.announcement),
  title: Text('Announcements'),
  onTap: () {
    Navigator.push(context,
      MaterialPageRoute(builder: (context) => AnnounceMents()));
  },
),
Divider(),
ListTile(
  leading: Icon(Icons.local_offer),
  title: Text('Hot Deals'),
  onTap: () {
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => Deals(
          user: widget.currentUser,
        )),
    );
  },
),
Divider(),
ListTile(
  leading: Icon(Icons.list),
  title: Text('Reports'),

  onTap: () {
    Navigator.push(
      context,

```

```

        MaterialPageRoute(
            builder: (context) => ReportsPage(),
        ));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.people),
    title: Text('Users'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => UserData()));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.history),
    title: Text('History'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => HistoryPage()));
    },
),
Divider(),
ListTile(
    leading: Icon(Icons.mic),
    title: Text('Delete Audios'),
    onTap: () {
        Navigator.push(context,
            MaterialPageRoute(builder: (context) => Audiopage()));
    },
),

```

```

Divider(),
ListTile(
    leading: Icon(Icons.exit_to_app),

```



```

        title: Text('logOut'),
        onTap: () async {
          await _auth.signOut();
        },
      ),
      Divider(),
    ],
  ),
),
appBar: AppBar(
  title: Text('Hello Admin'),
),
body: SingleChildScrollView(
  child: Column(
    children: <Widget>[
      Column(
        children: <Widget>[
          StreamBuilder(
            stream: Firestore.instance
.collection('requests')
              .orderBy('time')
              .snapshots(),
            builder: (context, snapshot) {
              if (!snapshot.hasData) {
                return Center(
                  child: CircularProgressIndicator(),
                );
              } else {
                return Column(
                  children: <Widget>[
                    Padding(
                      padding: const EdgeInsets.all(8.0),

                      child: Row(
                        children: <Widget>[
                          Text(

```

```

        'New Requests',
        style: TextStyle(
          fontSize: 20.0,
          fontWeight: FontWeight.bold,
        ),
      ),
      SizedBox(width: 10.0),
      CircleAvatar(
        radius: 12,
        child: Text(snapshot.data.documents.length
          .toString()))
    ],
  ),
),
Container(
  height: 300.0,
  child: Card(
    child: ListView.builder(
      shrinkWrap: true,
      physics: ClampingScrollPhysics(),
      itemCount: snapshot.data.documents.length ==
        0
        ? 0
        : snapshot.data.documents.length == 1
        ? 1
        : 2,
      itemBuilder: (context, index) {
        DocumentSnapshot req =
          snapshot.data.documents[index];
        String url = req['services'];
        String prescription = req['prescription'];
        return Slidable(
          actionPane: SlidableBehindActionPane(),

          actionExtentRatio: 0.30,
          child: Card(
            child: Padding(

```

```
padding: const EdgeInsets.all(8.0),
child: Column(
  crossAxisAlignment:
    CrossAxisAlignment.start,
  mainAxisAlignment:
    MainAxisAlignment.spaceEvenly,
  children: <Widget>[
    Text(
      'Name : ${req['name']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Phone No : ${req['phone']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Category : ${req['service_name']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
    Text(
      'Time : ${req['time']}',
      style: TextStyle(
        fontSize: 18.0,
      ),
    ),
  ],
),
```

```
req['service'] != null
? Text(
  'Service Name : ${req['service']}',
```

```

        style: TextStyle(
          fontSize: 18.0,
        ),
      )
    : req['prescription'] !=
      null
    ? Row(
      children: <Widget>[
        Text(
          'Prescription : '),
        Container(
          height: 70,
          width: 70,
          child: InkWell(
            onTap: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) => ViewPrescription(
                    url: prescription,
                  )),
              );
            },
            child: Card(
              child: Image.network(
                prescription,
                fit: BoxFit
                  .cover),
            ),
          ),
        ),
      ],
    )

```

```

      : PlayAudio(url: url)
    ],
  ),

```

```

    ),
    secondaryActions: <Widget>[
      IconSlideAction(
        caption: 'Move to Ongoing',
        color: Colors.green,
        icon: Icons.exit_to_app,
        onTap: () async {
          Firestore.instance
            .collection('requests')
              .document(req.documentID)
                .delete();
          req['services'] == null
            ? await Firestore.instance
              .collection('ongoing')
                .add({
                  "name": req['name'],
                  "phone": req['phone'],
                  "service_name":
                    req['service_name'],
                  "service":
                    req['service'],
                  'time': req['time'],
                  'date': req['date'],
                  'month': req['month'],
                  'year': req['year']
                })
            : await Firestore.instance
              .collection('ongoing')
                .add({
                  "name": req['name'],
                  "phone": req['phone'],
                  "service_name":
                    req['service_name'],
                  "services":

```

```

        req['services'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    });
    })
    ],
    );
    }},
    ),
    ),
    ],
    );
    }
    },
    ),
    SizedBox(height: 15.0),
    FloatingActionButton.extended(
      heroTag: Text('btn'),
      onPressed: () {
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) => NewRequest()));
      },
      label: Text('View All'),
    ),
    ],
    ),
    Column(
      children: <Widget>[

```

```

    StreamBuilder(
      stream: Firestore.instance
        .collection('ongoing')

```

```

        orderBy('time')
      ).snapshots(),
    builder: (context, snapshot) {
      if (!snapshot.hasData) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else {
        return Column(
          children: <Widget>[
            Padding(
              padding: const EdgeInsets.all(8.0),
              child: Row(
                children: <Widget>[
                  Text(
                    'Ongoing Requests',
                    style: TextStyle(
                      fontSize: 20.0,
                      fontWeight: FontWeight.bold,
                    ),
                  ),
                  SizedBox(width: 10.0),
                  CircleAvatar(
                    radius: 12,
                    child: Text(snapshot.data.documents.length
                      .toString()))
                ],
              ),
            ),
            Container(
              height: 300.0,
              child: Card(
                child: ListView.builder(
                  shrinkWrap: true,
                  physics: ClampingScrollPhysics(),

```

```
itemCount: snapshot.data.documents.length ==
```

```
0
```

```
? 0
```

```
: snapshot.data.documents.length == 1
```

```
? 1
```

```
: 2,
```

```
itemBuilder: (context, index) {
```

```
  DocumentSnapshot req =
```

```
    snapshot.data.documents[index];
```

```
  String url = req['services'];
```

```
  return Slidable(
```

```
    actionPane: SlidableBehindActionPane(),
```

```
    actionExtentRatio: 0.30,
```

```
    child: Card(
```

```
      child: Padding(
```

```
        padding: const EdgeInsets.all(8.0),
```

```
        child: Column(
```

```
          crossAxisAlignment:
```

```
            CrossAxisAlignment.start,
```

```
          mainAxisAlignment:
```

```
            MainAxisAlignment.spaceEvenly,
```

```
          children: <Widget>[
```

```
            Text(
```

```
              'Name : ${req['name']}',
```

```
              style: TextStyle(
```

```
                fontSize: 18.0,
```

```
            ),
```

```
          ),
```

```
            Text(
```

```
              'Phone No : ${req['phone']}',
```

```
              style: TextStyle(
```

```
                fontSize: 18.0,
```

```
            ),
```

```
          ),
```

```
            Text(
```

```
              'Time : ${req['time']}',
```



```

        style: TextStyle(
          fontSize: 18.0,
        ),
      ),
      req['service'] != null
        ? Text(
          'Service Name : ${req['service']}',
          style: TextStyle(
            fontSize: 18.0,
          ),
        )
        : PlayAudio(
          url: url,
        )
      ],
    ),
  ),
  secondaryActions: <Widget>[
    IconSlideAction(
      caption: 'Move to Completed',
      color: Colors.green,
      icon: Icons.exit_to_app,
      onTap: () async {
        Firestore.instance
          .collection('ongoing')
            .document(req.documentID)
              .delete();
        req['services'] == null
          ? await Firestore.instance
            .collection('completed')
              .add({
                "name": req['name'],
                "phone": req['phone'],
                "service_name":

```

```

        req['service_name'],
        "service":
            req['service'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    })
    : await Firestore.instance
.collection('completed')

    .add({
        "name": req['name'],
        "phone": req['phone'],
        "service_name":
            req['service_name'],
        "services":
            req['services'],
        'time': req['time'],
        'date': req['date'],
        'month': req['month'],
        'year': req['year']
    });
    })
    ],
    );
    },
    ),
    ],
    );
    }
    },
    ),

```

57

```

    SizedBox(height: 15.0),
    FloatingActionButton.extended(

```

```

heroTag: Text('btn'),
onPressed: () {
  Navigator.push(
    context,
    MaterialPageRoute(
      builder: (context) => OnGoingRequest()));
},
label: Text('View All'),
),
],
),
Column(
  children: <Widget>[
    StreamBuilder(
      stream: Firestore.instance
.collection('completed')
      .orderBy('time')
      .limit(2)
      .snapshots(),
    builder: (context, snapshot) {
      if (!snapshot.hasData) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else {
        return Column(
          children: <Widget>[
            Padding(
              padding: const EdgeInsets.all(8.0),
              child: Text(
                'Completed',
                style: TextStyle(
                  fontSize: 20.0,

                  fontWeight: FontWeight.bold,
                ),
              ),
            ),
          ],
        ),
      }
    ),
  ],
),

```

```

    ),
    Container(
      height: 300.0,
      child: Card(
        child: ListView.builder(
          shrinkWrap: true,
          physics: ClampingScrollPhysics(),
          itemCount: snapshot.data.documents.length,
          itemBuilder: (context, index) {
            DocumentSnapshot req =
              snapshot.data.documents[index];
            String url = req['services'];
            return Card(
              child: Padding(
                padding: const EdgeInsets.all(8.0),
                child: Column(
                  crossAxisAlignment:
                    CrossAxisAlignment.start,
                  mainAxisAlignment:
                    MainAxisAlignment.spaceEvenly,
                  children: <Widget>[
                    Text(
                      'Name : ${req['name']}',
                      style: TextStyle(
                        fontSize: 18.0,
                      ),
                    ),
                    Text(
                      'Phone No : ${req['phone']}',
                      style: TextStyle(
                        fontSize: 18.0,
                      ),
                    ),
                    Text(
                      'Time : ${req['time']}',
                      style: TextStyle(

```


MaterialPageRoute(

builder: (context) => HistoryPage()));

},

label: Text('View All'),

),

],

),

],

),

),

),

);

}

}

Screenshots





1:15

VoLTE 4G+ 83%

←

car

🏠

test

☐

Other districts

☐

with in the city

☐

Other

Service Name

Submit Request

OR

Didn't Find the Service you need ?

You can also Record Audio of the Service you need

Record Audio

🎤

▶

Start

■

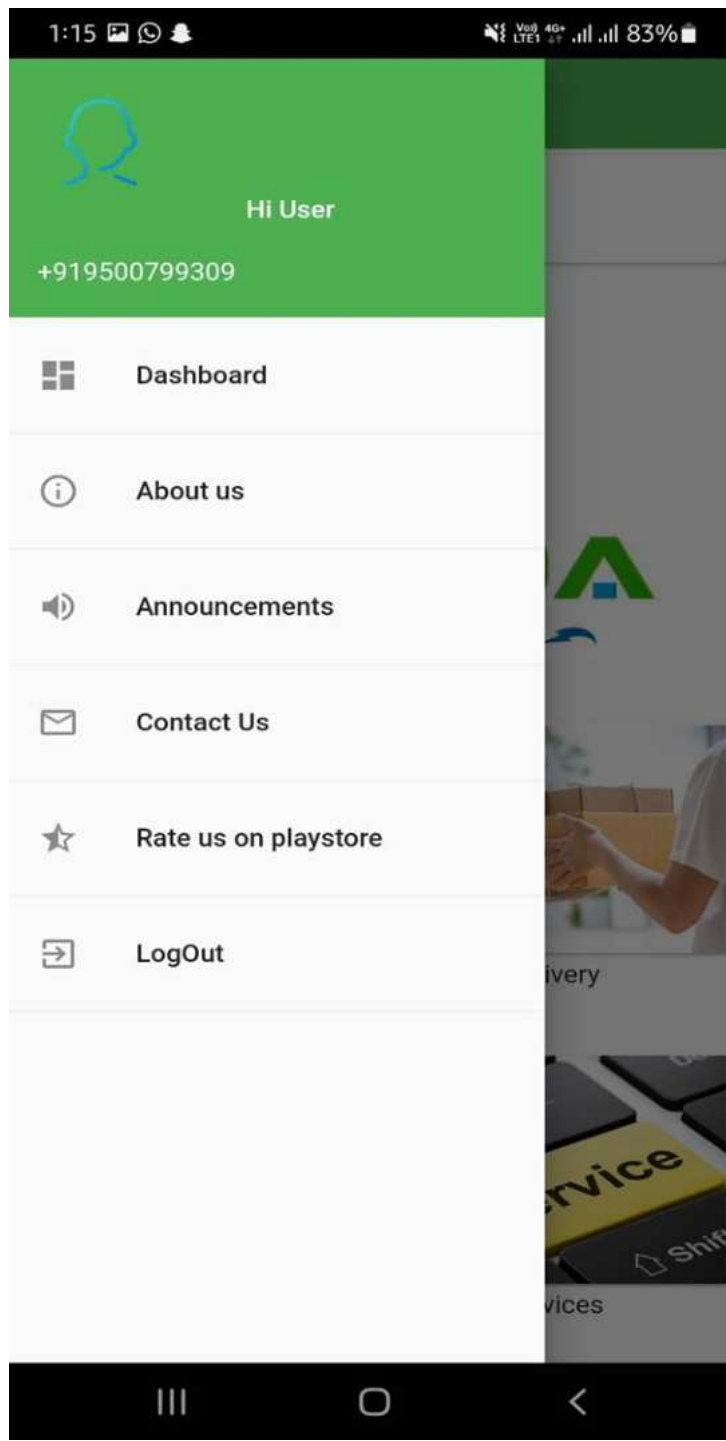
Stop

00:00

|||

○

<



Future Enhancement

No single project should be considered as complete project forever because our mind is always thinking something new and the requirement of users should be change over the time. We always want something more than what we have.

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this app:

- Make online chat facility for client and developer interaction.
- Get notifications.
- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
- Implementation of online payment mode.
- Order tracking can also be added.
- Based on future security issues, security can be improved using emerging technologies.

Conclusion

Advances in modern technology and electronic devices make the lives of people in all age groups easier and more convenient.

While we can now see adults and even kids being hooked to their mobile phone apps, utilization among senior citizens is rising too. Due to this rise in demand, we developed this app targeting the senior citizens and help them to live a better life.

This Project will thus pave the way for an efficient means to carry out the booking of the products. The main advantage of this project is to provide the basic facilities for the elderly people and full fill their needs. The User only needs basic products like a Phone and an internet connection.

Our app for the senior citizens cover all your needs like transportation, food delivery, medicines, groceries, housekeeping, etc. This is useful for senior citizens in India which will enrich their life or your loved once. Embrace technology instead of fearing or avoiding it as we live in a digital era where technology should be accessible for our ageing parents and grandparents.

Bibliography

FLUTTER REFERENCE:

<https://www.javatpoint.com/flutter>

<https://www.tutorialspoint.com/flutter/index.htm>

FIREBASE REFERENCE:

<https://www.tutorialspoint.com/firebase/index.htm>

<https://www.javatpoint.com/firebase>

FARMINGTON

THE FARM MANAGEMENT SYSTEM

Project Submitted to the Department Of Computer Science

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

Affiliated to

MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



In partial fulfilment of the requirement for the award of degree in

BACHELOR OF COMPUTER SCIENCE done by

| | | |
|-----------------------------|---|-----------------|
| Joseph Amala Lincy.B | - | 18AUCS14 |
| Kathija Apsana.M | - | 18AUCS15 |
| Nisha Rani.P | - | 18AUCS32 |
| Silvia Anusha.P | - | 18AUCS44 |

Under the Supervision and Guidance of

Mrs. J. Anila Maily M.Sc., M.Phil

DEPARTMENT OF COMPUTER SCIENCE

ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Re-Accredited with 'A+' Grade by NAAC (4th Cycle)

JOB PORTAL SYSTEM

Project Submitted to the Department of Computer Science
ST. MARY'S COLLEGE (Autonomous), THOOTHUKUDI

Affiliated to
MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



In partial fulfilment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE

Done by

| | |
|-----------------|----------|
| B. Manjula Devi | 18AUCS21 |
| S. Maria Jebina | 18AUCS24 |
| V. Nivethitha | 18AUCS33 |
| K. Pavithra | 18AUCS35 |
| R. Poomari Uma | 18AUCS37 |

Under the Supervision and Guidance of

Mrs. S. Gnanathangam M.C.A., M.Phil.

(H.O.D of Computer Science Department)

DEPARTMENT OF COMPUTER SCIENCE

ST.MARY'S COLLEGE(AUTONOMOUS)

Reaccredited with "A+" Grade by NAAC

THOOTHUKUDI-628 001.

2018-2021

CERTIFICATE

ST.MARY'S COLLEGE (Autonomous), THOOTHUKUDI
Re-Accredited with 'A+' Grade by NAAC(4th cycle)

Affiliated to
MANONMANIAM SUNDARANAR UNIVERSITY, Tirunelveli.



DEPARTMENT OF COMPUTER SCIENCE

This is to certify that project entitled

JOB PORTAL SYSTEM

Submitted by

| | |
|-----------------|----------|
| B. Manjula Devi | 18AUCS21 |
| S. Maria Jebina | 18AUCS24 |
| V. Nivethitha | 18AUCS33 |
| K. Pavithra | 18AUCS35 |
| R. Poomari Uma | 18AUCS37 |

In partial fulfillment of the requirement for the award of degree in
BACHELOR OF COMPUTER SCIENCE during the year 2020-2021.

S. Gnaniy
Guide

S. Gnaniy
Head of the Department

Submitted for the examination held on 15/04/2021

[Signature]
Examiner

Lucia Rose
Principal
St. Mary's College (Autonomous)
Thoothukudi - 628 001.

ACKNOWLEDGEMENT

First and foremost we thank the Almighty God for his gracious help and shower of blessings for having rendered us the strength and support to finish our project successfully.

We are infinitely indebted to our college principal Dr.Sr.A.S.J.Lucia Rose M.Sc.,M.Phil.,PGDCA.,Ph.D., for providing us necessary infrastructure to undertake this project and work successfully.

We are ever so Grateful to Mrs.S.Gnanathangam M.C.A.,M.Phil., (H.O.D) of the Computer Science Department and our project guide for her prompt notification,encouragement,guidance and beneficial criticisms which significantly helped us in scheming the project in a constructive manner.

We are highly indebted to our lecturers, Mrs. M. Jothimani B.E., M.Sc(IT).,M.Phil., Mrs.J.AnilaMaily M.Sc., M.Phil., Mrs. S. AndrewsFelcita M.Sc., M.B.A., and Sr.K. Reetha M.Sc., M.Phil., for the kind cooperation and support in successful completion of the project.

We would like to thank our lab assistant Mrs. Cladina for her kind and patient assistance in the lab during our project hours.

We would thank our family members especially our parents for their support both morally and financially which motivated us to finish the project successfully.

Last but not least, I owe my thanks to our friends for their Encouragement ,support and creative ideas which led our project in a successful path.

INDEX

| S.NO | CONTENTS | PAGENO |
|------|--|--------------------------------|
| 1. | SYNOPSIS
1.1 INTRODUCTION
1.2 MOTIVATION |
7
8 |
| 2. | SYSTEM ANALYSIS
2.1 FEASIBILITY STUDY
2.2 EXISTING SYSTEM
2.3 PROPOSED SYSTEM
2.4 ANALYZING THE SYSTEM |
9
10
11
11 |
| 3. | SYSTEM REQUIREMENTS AND SPECIFICATIONS
3.1 HARDWARE REQUIREMENT
3.2 SOFTWARE REQUIREMENT
3.3 SOFTWARE DESCRIPTION
3.3.1 ANDROID SOFTWARE
3.3.2 FIREBASE CONSOLE |
13
13
13
13
32 |
| 4. | TESTING IMPLEMENTATION
4.1 UNIT TESTING
4.2 INTEGRATION TESTING
4.3 SOFTWARE VERIFICATION AND VALIDATION
4.4 SYSTEM TESTING |
36
37
39
40 |
| 5. | SYSTEM DESIGN
5.1 DATABASE TABLE DESIGN
5.2 DATA FLOW DIAGRAM
5.3 FORM DESIGN |
41
43
49
58 |

| | | |
|-----------|---------------------------|------------|
| | 5.4 SOURCE CODE | |
| 6. | FUTURE ENHANCEMENT | 102 |
| 7. | CONCLUSION | 102 |
| 8. | BIBILOGRAPHY | 103 |

1.SYNOPSIS

In this competitive era, the education among the people is so increasing that the jobs for them are now decreasing. The companies even want the people who are best in their fields. At that time, it becomes difficult to find the people who are intelligent enough to be hired. The work for the companies also increases to find the people who can fulfill their requirements. Thinking about these problems, one can think about the process which can handle this process and make the work less complex. This project is about the recruitment process which is done online. The recruitment process here is handled by the system. This project will allow the person to apply for a job in the company for the interested vacancy which would be available at the company. The person will be having the account after registration and will be then be called by the applied user. If he/she is qualified, he/she would be interacting with the system for the updates. The project is created for fulfilling the requests of the company managers so that the recruitment module can be placed in the company's website and the users who visit the website can view the vacancies in the company and will be able to apply directly from remote place even. The vacancies will be posted by the administrator on the basis of needs of the manpower in the company.

1.1 Introduction:

Job Hunt is an android mobile application, which serves jobseekers to find available job vacancies and Employers to identify eligible job seekers with

the prospect of selecting the most qualified candidates. The only way to select best-qualified candidate is to have a pool of eligible applicants, which is possible by drawing the interest of individuals in the market. Job Hunt best serves this purpose. E-recruitment has become the standard means for employers and job seekers to meet their respective objectives. The traditional methods for recruitment include Job fairs, University career employment services, Employee referrals, advertising in the newspapers, televisions etc. With the advancement in technology and growth of internet usage, the e-recruitment has revolutionized the way organizations hire and candidates search for jobs. With the Online Job search portals, the recruitment process is speeded up at every stage from job postings, to receiving applications from candidates, interviewing process. The cost of searching/posting jobs will be much less compared to the traditional way of advertising. Job Hunt stands as an effective means for Employers to outline the job vacancies, responsibilities and qualifications to attract jobseekers. Using the Job Hunt, jobseekers can extensively search for jobs in companies, organizations and regions they may otherwise have not learnt. In addition, candidates/Employers can write a review about an organization, which might help them to change the way things are done.

1.2 Motivation:

The purpose of developing Job Hunt comes from the idea to make job search efficient and handy, especially during this pandemic time. It helps the recruiters as a primary source of talent search. It also helps the job seekers to search for current vacancies at a single point. Therefore, we can say that Job Hunt act as a bridge of communication between organizations and

applicants. With the evolution of technology and internet being the main source of information for the applicants, these job portals have become an excellent method to reach wide range of audience. Initially, when I am unaware of these portals, I used to do research about companies and their technology stack through their respective websites and apply if the job responsibilities matches my interests. This requires lots of effort and time. However, later when I realized the importance of job search portals, I am able to access jobs in companies, locations that I might not otherwise have learned.

2.SYSTEM ANALYSIS

System analysis is the detailed study of various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optional or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to the system.

2.1 Feasibility Study:

- 1) Technical Feasibility
- 2) Economic Feasibility
- 3) Physical Feasibility

Technical Feasibility:

Security and usability are considered as the most important factors of the system design that increase the user friendliness of the system. The proposed system is technically feasible in such a way that this can't be cracked by the crackers.

Economic Feasibility:

Biometrics authentication is highly expensive and its identification process is slow. But the proposed session password is low cost method in terms of less computational time and low memory requirement.

Physical feasibility:

A grid-based system resembling a mini authenticating activity which is physically a less competent space to make the system with high potentially secures. So the desired session based system physically feasible indeed.

2.2 Existing System:

The existing system for job recruitment includes traditional methods like employment agencies, advertising through newspapers, television and radios, college fairs, etc., which are too slow and stressful. With the advancement of internet, jobseekers rely on the online job portals like Job

Hunt, which makes the job search efficient. Again, most of these are limited to the android applications, which requires jobseekers to have an android mobile connected to internet and is not handy. All processes in existing system are handled manually. All the work that is done in the existing system is done by the human intervention. As all the work is done manually, there is a lot of workload on and it also increases the maximum chances of errors. This is so slow and time consuming. Due to increase in number of user's, the process becomes more difficult.

Disadvantages:

- Time Consuming
- Challenging

2.3 Proposed System:

Job Hunt is a Java-based web application as well as Android application that provides functionalities of e-recruitment on desktop and on portable devices like Android based smart phones/tablets. Daily updates via notifications and other communication media. Job seeker can set privacy level for different companies. Job seeker can save jobs according to their needs. Most recent jobs are displayed on the home page. And also, Counting the number of times the resume of a job seeker is accessed by the company. The application does not require internet to perform the desired functionalities.

Advantages:

- Cost and Time efficient
- Portable

2.4 Analyzing the System:

The objective of the job hunt in android application is to provide flexibility to the jobseekers by providing the functionalities of both job search and job application in a single application. In addition, this application provides an effective means for the employers to post job vacancies and view the job applications by the interested applicants in a single application.

MODULE DESCRIPTION:

- 1) Applicant Registration
- 2) Company Registration
- 3) Job Search
- 4) Vacancy Registration
- 5) Message
- 6) Logout

1) APPLICANT REGISTRATION:

The applicant can register into the site by providing the details and he/she can set his username and password for logging in later.

2) COMPANY REGISTRATION:

The authority of the company can register into the site by providing its details and they can set its username and password for logging in. They can also specify their requirements for each vacancy.

3) JOB SEARCH:

Applicants can search for the job vacancies in the fields of their choice and apply as interested. They can also provide/upload their resume CV and profile image without pay of cost. They can also view the advertisement given by the company and apply it, if interested.

4) VACANCY REGISTRATION:

Authority of the company can provide all the necessary information about the vacancies in each post along with eligibility criteria in the form of advertisement.

5) MESSAGE:

Software selects the apt ones from the registered applicants for each post and informs them by a mobile message/notification alert. The message includes all about the interview details.

6) LOGOUT:

The Applicant/Employer can log out and it will again go back to login page.

3.SYSTEM REQUIREMENTS AND SPECIFICATIONS:

3.1 HARDWARE REQUIREMENTS:

- INTEL DUAL CORE 2.0Ghz CPU
- 2GB RAM
- 320GB HDD
- KEY BOARD
- MOUSE
- MOBILE/TABLET

3.2 SOFTWARE REQUIREMENTS:

- OPERATING SYSTEM: WINDOWS XP/10

- FRONT END: Android Development Kit
- CODING: Java
- BACK END: Firebase
- EMULATOR:

3.3 SOFTWARE DESCRIPTION:

3.3.1 ANDROID SOFTWARE DEVELOPMENT:

The early feedback on developing applications for the Android platform was mixed. Issues cited include bugs, lack of documentation, inadequate QA infrastructure, and no public issue-tracking system. (Google announced an issue tracker on 18 January 2008.) In December 2007, Merge-Lab mobile startup founder Adam Mac-Beth stated, "Functionality is not there, is poorly documented or just doesn't work... It's clearly not ready for prime time." Despite this, Android-targeted applications began to appear the week after the platform was announced. The first publicly available application was the Snake game. The Android Dev Phone is a SIM-unlocked and hardware-unlocked device that is designed for advanced developers. While developers can use regular consumer's devices purchased at retail to test and use their applications, some developers may choose not to use a retail device, preferring an unlocked or no-contract device.

Software development kit:

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator (based on QEMU), documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.4.9 or later, Windows XP

or later. The officially supported integrated development environment (IDE) is Eclipse (currently 3.4 or 3.5) using the Android Development Tools (ADT) Plugin, though developers may use any text editor to edit Java and XML files then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).

A preview release of the Android SDK was released on 12 November 2007. On 15 July 2008, the Android Developer Challenge Team accidentally sent an email to all entrants in the Android Developer Challenge announcing that a new release of the SDK was available in a "private" download area. The email was intended for winners of the first round of the Android Developer Challenge. The revelation that Google was supplying new SDK releases to some developers and not others (and keeping this arrangement private) led to widely reported frustration within the Android developer community at the time.

On 18 August 2008 the Android 0.9 SDK beta was released. This release provided an updated and extended API, improved development tools and an updated design for the home screen. Detailed instructions for upgrading are available to those already working with an earlier release. On 23 September 2008 the Android 1.0 SDK (Release 1) was released. According to the release notes, it included "mainly bug fixes, although some smaller features were added". It also included several API changes from the 0.9 version. Multiple versions have been released since.

Enhancements to Android's SDK go hand in hand with the overall Android platform development. The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Development tools are downloadable components, so after one has

downloaded the latest version and platform, older platforms and tools can also be downloaded for compatibility testing.

Android applications are packaged in. apk format and stored under /data/app folder on the Android OS (the folder is accessible to root user only for security reasons). APK package contains. dex files (compiled byte code file call Dalvik executable) Resource files, etc.



ANDROID APPLICATION FUNDAMENTALS:

Android applications are written in the Java programming language. The compiled Java code — along with any data and resource files required by the application — is bundled by the aapt tool into an Android package, an archive file marked by an .apk suffix. This file is the vehicle for distributing the application and installing it on mobile devices; it's the file users download to their devices. All the code in a single .apk file is considered to be one application.

In many ways, each Android application lives in its own world:

- By default, every application runs in its own Linux process. Android starts the process when any of the application's code needs to be executed, and shuts down the process when it's no longer needed and system resources are required by other applications.
- Each process has its own virtual machine (VM), so application code runs in isolation from the code of all other applications.
- By default, each application is assigned a unique Linux user ID. Permissions are set so that the application's files are visible only to that user and only to the application itself — although there are ways to export them to other applications as well.

It's possible to arrange for two applications to share the same user ID, in which case they will be able to see each other's files. To conserve system resources, applications with the same ID can also arrange to run in the same Linux process, sharing the same VM.

APPLICATION COMPONENTS:

A central feature of Android is that one application can make use of elements of other applications (provided those applications permit it). For example, if your application needs to display a scrolling list of images and another application has developed a suitable scroller and made it available to others, you can call upon that scroller to do the work, rather than develop your own. Your application doesn't incorporate the code of the other application or link to it. Rather, it simply starts up that piece of the other application when the need arises.

For this to work, the system must be able to start an application process when any part of it is needed, and instantiate the Java objects for that part. Therefore, unlike applications on most other systems, Android applications don't have a single entry point for everything in the application (no `main()` function, for example). Rather, they have essential components that the system can instantiate and run as needed. There are four types of components:

Activities:

An activity presents a visual user interface for one focused endeavor the user can undertake. For example, an activity might present a list of menu items users can choose from or it might display photographs along with their captions. A text messaging application might have one activity that shows a list of contacts to send messages to, a second activity to write the message to the chosen contact, and other activities to review old messages or change settings. Though they work together to form a cohesive user interface, each activity is independent of the others. Each one is implemented as a subclass of the Activity base class.

An application might consist of just one activity or, like the text messaging application just mentioned, it may contain several. What the activities are, and how many there are depends, of course, on the application and its design. Typically, one of the activities is marked as the first one that should be presented to the user when the application is launched. Moving from one activity to another is accomplished by having the current activity start the next one.

Each activity is given a default window to draw in. Typically, the window fills the screen, but it might be smaller than the screen and float on top of other windows. An activity can also make use of additional windows — for example, a pop-up dialog that calls for a user response in the midst of the activity, or a window that presents users with vital information when they select a particular item on-screen.

The visual content of the window is provided by a hierarchy of views — objects derived from the base `View` class. Each view controls a particular rectangular space within the window. Parent views contain and organize the layout of their children. Leaf views (those at the bottom of the hierarchy) draw in the rectangles they control and respond to user actions directed at that space. Thus, views are where the activity's interaction with the user takes place. For example, a view might display a small image and initiate an action when the user taps that image. Android has a number of ready-made views that you can use — including buttons, text fields, scroll bars, menu items, check boxes, and more.

A view hierarchy is placed within an activity's window by the `Activity setContentView()` method. The content view is the `View` object at the root of the hierarchy. (See the separate User Interface document for more information on views and the hierarchy.)

Services:

A service doesn't have a visual user interface, but rather runs in the background for an indefinite period of time. For example, a service might play background music as the user attends to other matters, or it might fetch data over the network or calculate something and provide the result to activities that need it. Each service extends the Service base class.

A prime example is a media player playing songs from a play list. The player application would probably have one or more activities that allow the user to choose songs and start playing them. However, the music playback itself would not be handled by an activity because users will expect the music to keep playing even after they leave the player and begin something different. To keep the music going, the media player activity could start a service to run in the background. The system would then keep the music playback service running even after the activity that started it leaves the screen.

It's possible to connect to (bind to) an ongoing service (and start the service if it's not already running). While connected, you can communicate with the service through an interface that the service exposes. For the music service, this interface might allow users to pause, rewind, stop, and restart the playback.

Like activities and the other components, services run in the main thread of the application process. So that they won't block other components or

the user interface, they often spawn another thread for time-consuming tasks (like music playback). See Processes and Threads, later.

Broadcast receivers:

A broadcast receiver is a component that does nothing but receive and react to broadcast announcements. Many broadcasts originate in system code — for example, announcements that the time zone has changed, that the battery is low, that a picture has been taken, or that the user changed a language preference. Applications can also initiate broadcasts — for example, to let other applications know that some data has been downloaded to the device and is available for them to use.

An application can have any number of broadcast receivers to respond to any announcements it considers important. All receivers extend the Broadcast Receiver base class.

Broadcast receivers do not display a user interface. However, they may start an activity in response to the information they receive, or they may use the Notification Manager to alert the user. Notifications can get the user's attention in various ways — flashing the backlight, vibrating the device, playing a sound, and so on. They typically place a persistent icon in the status bar, which users can open to get the message.

Content providers:

A content provider makes a specific set of the application's data available to other applications. The data can be stored in the file system,

in an SQLite database, or in any other manner that makes sense. The content provider extends the Content Provider base class to implement a standard set of methods that enable other applications to retrieve and store data of the type it controls. However, applications do not call these methods directly. Rather they use a Content Resolver object and call its methods instead. A Content-Resolver can talk to any content provider; it cooperates with the provider to manage any inter-process communication that's involved.

See the separate Content Providers document for more information on using content providers.

PROCESSES AND THREADS:

When the first of an application's components needs to be run, Android starts a Linux process for it with a single thread of execution. By default, all components of the application run in that process and thread.

However, you can arrange for components to run in other processes, and you can spawn additional threads for any process.

Processes:

The process where a component runs is controlled by the manifest file. The component elements — `<activity>`, `<service>`, `<receiver>`, and `<provider>` — each have a process attribute that can specify a process where that component should run. These attributes can be set so that each component runs in its own process, or so that some components share a process while others do not. They can also be set so that components of different applications run in the same process — provided that the applications share the same Linux user ID and are signed by the same authorities. The `<application>` element also has a

process attribute, for setting a default value that applies to all components.

All components are instantiated in the main thread of the specified process, and system calls to the component are dispatched from that thread. Separate threads are not created for each instance. Consequently, methods that respond to those calls — methods like `View.onKeyDown()` that report user actions and the lifecycle notifications discussed later in the Component Lifecycles section — always run in the main thread of the process. This means that no component should perform long or blocking operations (such as networking operations or computation loops) when called by the system, since this will block any other components also in the process. You can spawn separate threads for long operations, as discussed under Threads, next.

Android may decide to shut down a process at some point, when memory is low and required by other processes that are more immediately serving the user. Application components running in the process are consequently destroyed. A process is restarted for those components when there's again work for them to do.

When deciding which processes to terminate, Android weighs their relative importance to the user. For example, it more readily shuts down a process with activities that are no longer visible on screen than a process with visible activities. The decision whether to terminate a process, therefore, depends on the state of the components running in that process. Those states are the subject of a later section, Component Lifecycles.

Threads:

Even though you may confine your application to a single process, there will likely be times when you will need to spawn a thread to do some background work. Since the user interface must always be quick to respond to user actions, the thread that hosts an activity should not also host time-consuming operations like network downloads. Anything that may not be completed quickly should be assigned to a different thread.

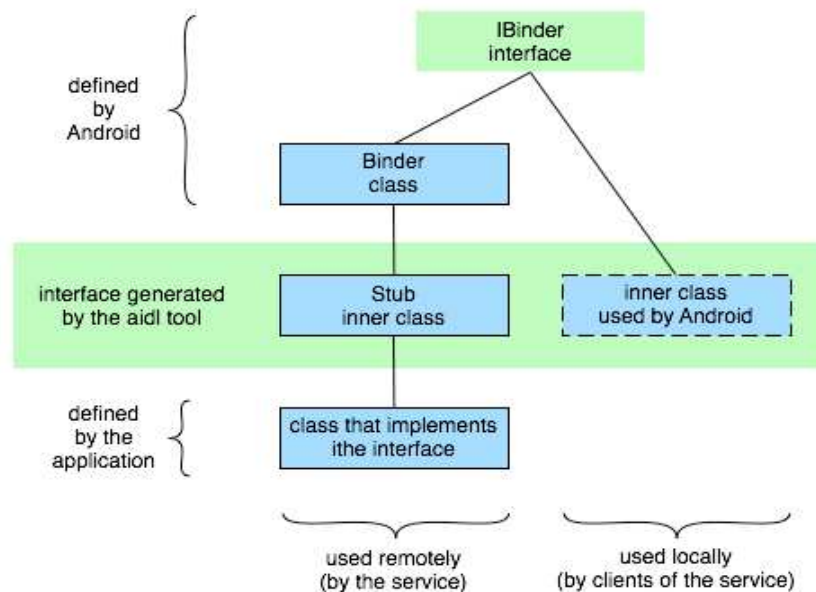
Threads are created in code using standard Java Thread objects. Android provides a number of convenience classes for managing threads — `Looper` for running a message loop within a thread, `Handler` for processing messages, and `Handler Thread` for setting up a thread with a message loop.

Remote procedure calls:

Android has a lightweight mechanism for remote procedure calls (RPCs) — where a method is called locally, but executed remotely (in another process), with any result returned back to the caller. This entails decomposing the method call and all its attendant data to a level the operating system can understand, transmitting it from the local process and address space to the remote process and address space, and reassembling and reenacting the call there. Return values have to be transmitted in the opposite direction. Android provides all the code to do that work, so that you can concentrate on defining and implementing the RPC interface itself.

An RPC interface can include only methods. By default, all methods are executed synchronously (the local method blocks until the remote method finishes), even if there is no return value.

In brief, the mechanism works as follows: You'd begin by declaring the RPC interface you want to implement using a simple IDL (interface definition language). From that declaration, the aidl tool generates a Java interface definition that must be made available to both the local and the remote process. It contains two inner class, as shown in the following **Diagram:**



The inner classes have all the code needed to administer remote procedure calls for the interface you declared with the IDL. Both inner classes implement the I-Binder interface. One of them is used locally and internally by the system; the code you write can ignore it. The other, called Stub, extends the Binder class. In addition to internal code for effectuating the IPC calls, it contains declarations for the methods in the RPC interface you declared. You would subclass Stub to implement those methods, as indicated in the diagram.

Typically, the remote process would be managed by a service (because a service can inform the system about the process and its connections to other processes). It would have both the interface file generated by the aidl tool and the Stub subclass implementing the RPC methods. Clients

of the service would have only the interface file generated by the aidl tool.

Here's how a connection between a service and its clients is set up:

Clients of the service (on the local side) would implement `onServiceConnected()` and `onServiceDisconnected()` methods so they can be notified when a successful connection to the remote service is established, and when it goes away. They would then call `bindService()` to set up the connection.

The service's `onBind()` method would be implemented to either accept or reject the connection, depending on the intent it receives (the intent passed to `bindService()`). If the connection is accepted, it returns an instance of the Stub subclass.

If the service accepts the connection, Android calls the client's `onServiceConnected()` method and passes it an `IBinder` object, a proxy for the Stub subclass managed by the service. Through the proxy, the client can make calls on the remote service.

This brief description omits some details of the RPC mechanism. For more information, see [Designing a Remote Interface Using AIDL](#) and the `IBinder` class description.

COMPONENT LIFECYCLES:

Application components have a lifecycle — a beginning when Android instantiates them to respond to intents through to an end when the instances are destroyed. In between, they may sometimes be active or inactive, or, in the case of activities, visible to the user or invisible. This section discusses the lifecycles of activities, services, and broadcast receivers — including the states that they can be in during their lifetimes, the methods that notify you of transitions between states, and the effect of those states on the possibility that the process hosting them might be terminated and the instances destroyed.

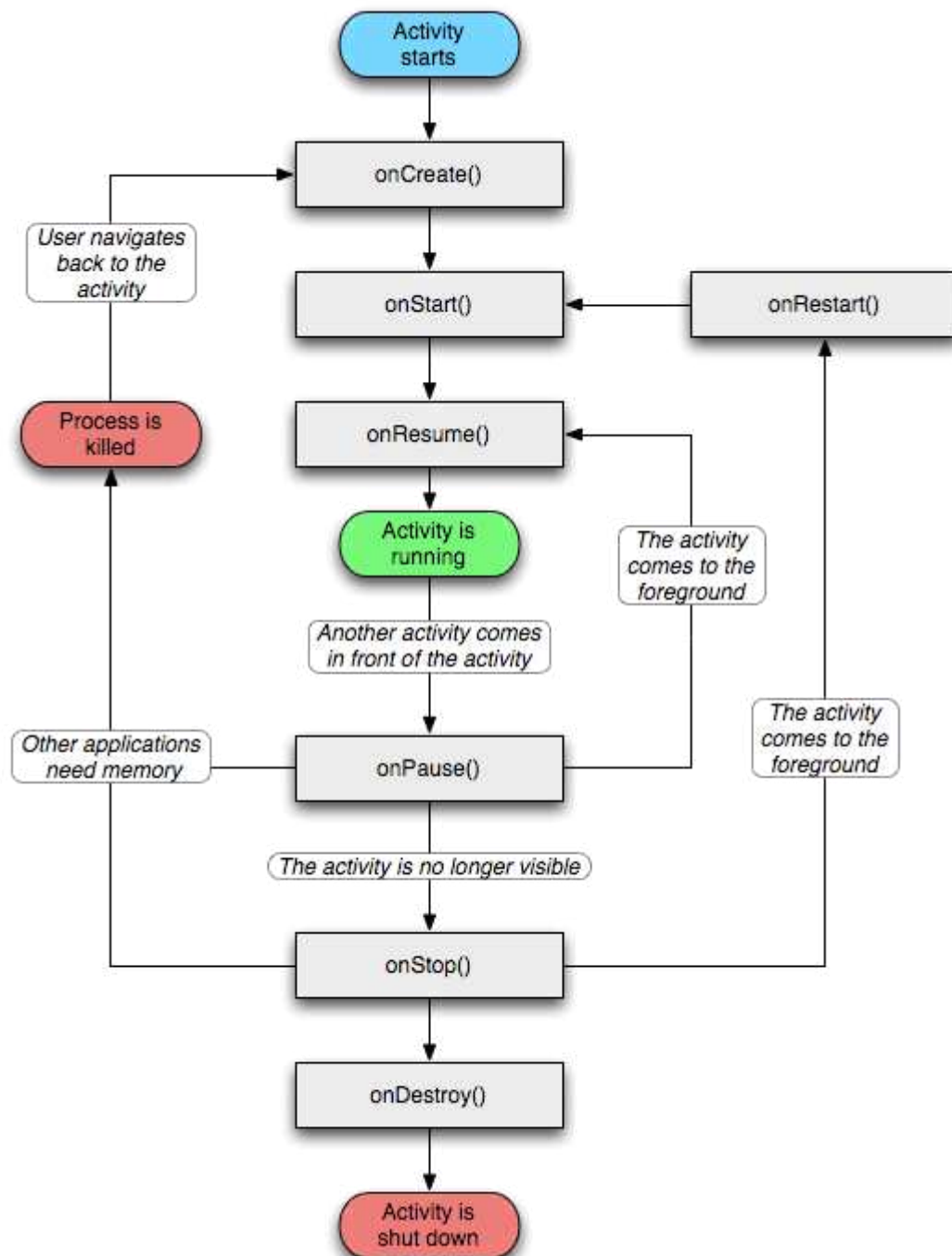
Activity lifecycle:

An activity has essentially three states:

It is active or running when it is in the foreground of the screen (at the top of the activity stack for the current task). This is the activity that is the focus for the user's actions.

It is paused if it has lost focus but is still visible to the user. That is, another activity lies on top of it and that activity either is transparent or doesn't cover the full screen, so some of the paused activity can show through. A paused activity is completely alive (it maintains all state and member information and remains attached to the window manager), but can be killed by the system in extreme low memory situations.

It is stopped if it is completely obscured by another activity. It still retains all state and member information. However, it is no longer visible to the user so its window is hidden and it will often be killed by the system when memory is needed elsewhere.



Whenever there's a request that should be handled by a particular component, Android makes sure that the application process of the component is running, starting it if necessary, and that an appropriate instance of the component is available, creating the instance if necessary.

PROCESSES AND LIFECYCLES:

The Android system tries to maintain an application process for as long as possible, but eventually it will need to remove old processes when memory runs low. To determine which processes to keep and which to kill, Android places each process into an "importance hierarchy" based on the components running in it and the state of those components. Processes with the lowest importance are eliminated first, then those with the next lowest, and so on. There are five levels in the hierarchy. The following list presents them in order of importance:

USER INTERFACE:

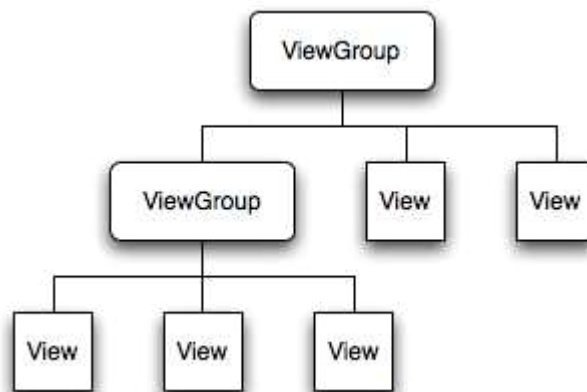
In an Android application, the user interface is built using View and ViewGroup objects. There are many types of views and view groups, each of which is a descendant of the View class.

View objects are the basic units of user interface expression on the Android platform. The View class serves as the base for subclasses called "widgets," which offer fully implemented UI objects, like text fields and buttons. The ViewGroup class serves as the base for subclasses called "layouts," which offer different kinds of layout architecture, like linear, tabular and relative.

A View object is a data structure whose properties store the layout parameters and content for a specific rectangular area of the screen. A View object handles its own measurement, layout, drawing, focus change, scrolling, and key/gesture interactions for the rectangular area of the screen in which it resides. As an object in the user interface, a View is also a point of interaction for the user and the receiver of the interaction events

View Hierarchy:

On the Android platform, you define an Activity's UI using a hierarchy of View and ViewGroup nodes, as shown in the diagram below. This hierarchy tree can be as simple or complex as you need it to be, and you can build it up using Android's set of predefined widgets and layouts, or with custom Views that you create yourself.



In order to attach the view hierarchy tree to the screen for rendering, your Activity must call the `setContentView()` method and pass a reference to the root node object. The Android system receives this reference and uses it to invalidate, measure, and draw the tree. The root node of the hierarchy requests that its child nodes draw themselves — in turn, each view group node is responsible for calling upon each of its own child views to draw themselves. The children may request a size and location within the parent, but the parent object has the final decision on where how big each child can be. Android parses the elements of your layout in-order (from the top of the hierarchy tree), instantiating the Views and adding them to their parent(s). Because these are drawn in-order, if there are elements that overlap positions, the last one to be drawn will lie on top of others previously drawn to that space.

Layout:

The most common way to define your layout and express the view hierarchy is with an XML layout file. XML offers a human-readable structure for the layout, much like HTML. Each element in XML is either a View or ViewGroup object (or descendant thereof). View objects are leaves in the tree, ViewGroup objects are branches in the tree (see the View Hierarchy figure above).

The name of an XML element is respective to the Java class that it represents. So a <TextView> element creates a TextView in your UI, and a <LinearLayout> element creates a LinearLayout view group. When you load a layout resource, the Android system initializes these run-time objects, corresponding to the elements in your layout.

For example, a simple vertical layout with a text view and a button looks like this:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"

    android:orientation="vertical">
    <TextView android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello, I am a TextView" />
    <Button android:id="@+id/button"
        android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"  
    android:text="Hello, I am a Button" />  
</LinearLayout>
```

Notice that the `LinearLayout` element contains both the `TextView` and the `Button`. You can nest another `LinearLayout` (or other type of view group) inside here, to lengthen the view hierarchy and create a more complex layout.

There are a variety of ways in which you can layout your views. Using more and different kinds of view groups, you can structure child views and view groups in an infinite number of ways. Some pre-defined view groups offered by Android (called layouts) include `LinearLayout`, `RelativeLayout`, `TableLayout`, `GridLayout` and others. Each offers a unique set of layout parameters that are used to define the positions of child views and layout structure.

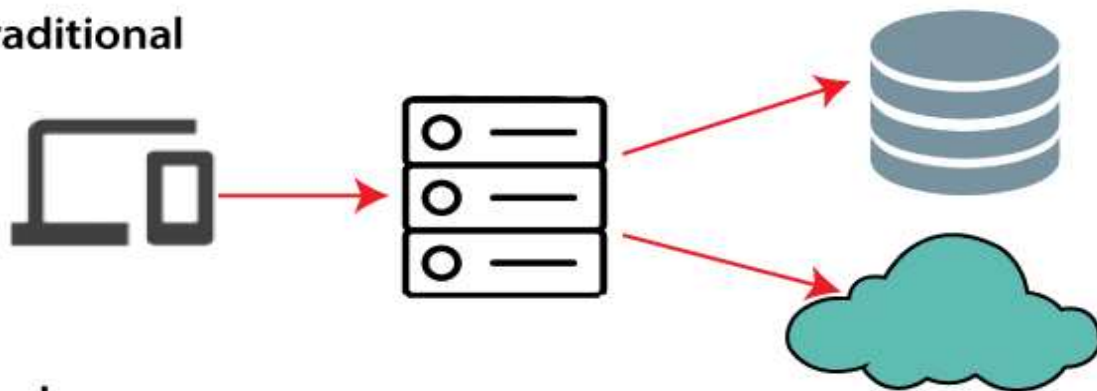
3.3.2 FIREBASE CONSOLE:

Introduction:

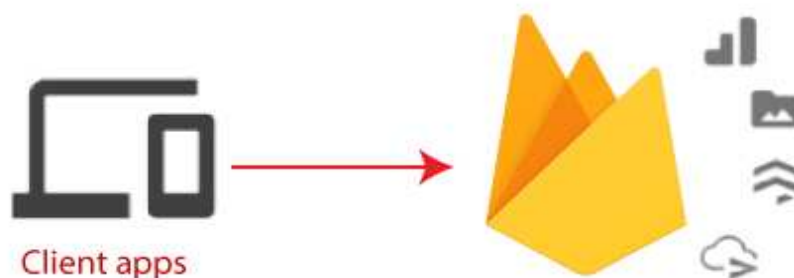
In the era of rapid prototyping, we can get bright ideas, but sometimes they are not applicable if they take too much work. Often, the back end is the limiting factor - many considerations never apply to server-side coding due to lack of knowledge or time.

Firebase is a Backend-as-a-Service (BaaS) which started as a YC11 startup. It grew up into a next-generation app-development platform on Google Cloud Platform. Firebase (a No-SQL-JSON database) is a real-time database that allows storing a list of objects in the form of a tree. We can synchronize data between different devices.

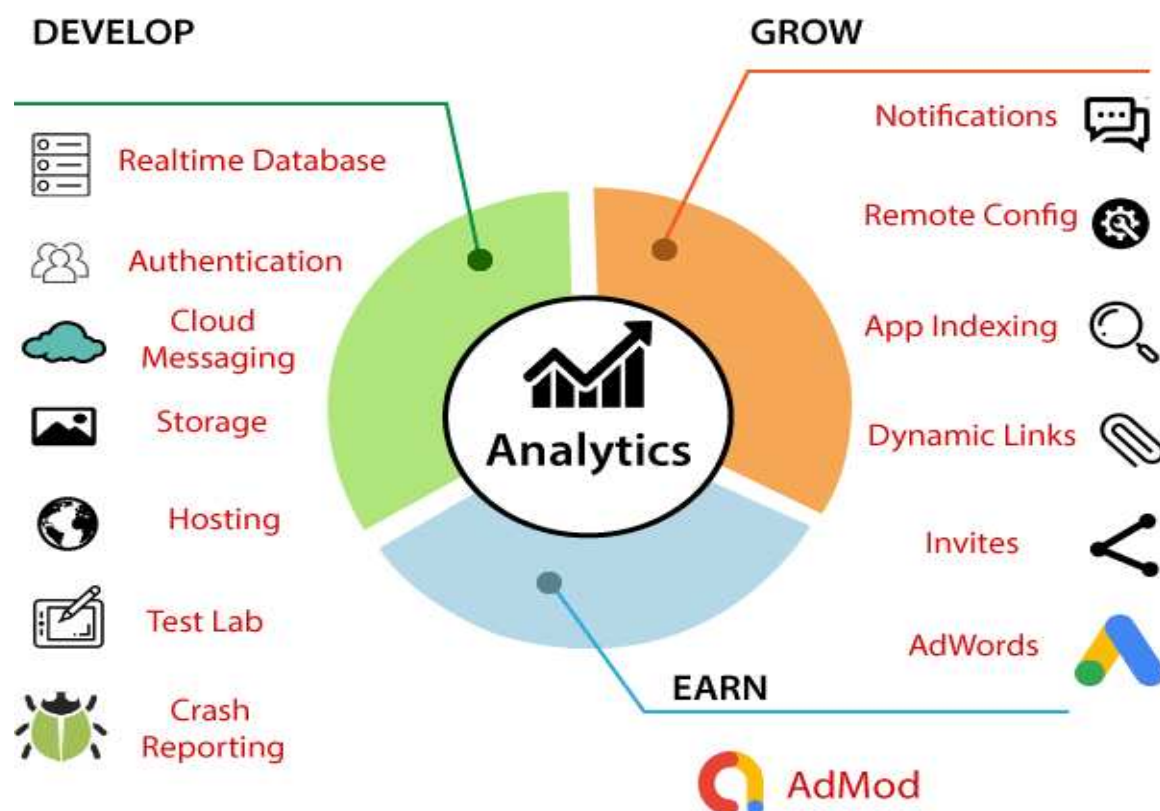
Traditional



Firebase



Google Firebase is Google-backed application development software which allows developers to develop **Android, IOS, and Web apps**. For reporting and fixing app crashes, tracking analytics, creating marketing and product experiments, firebase provides several tools.

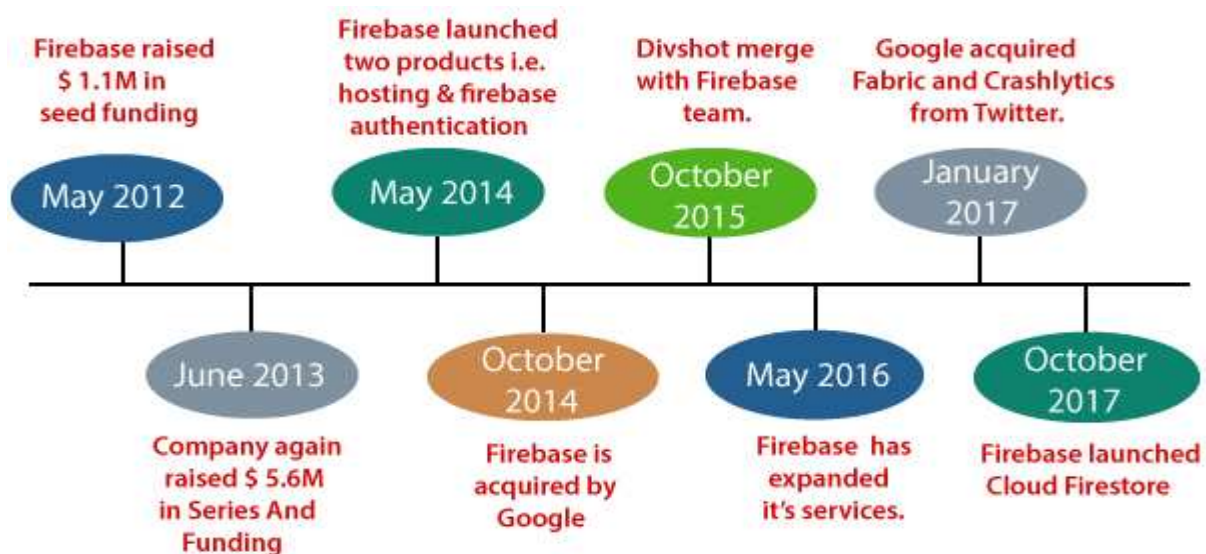


Firebase has three main services, i.e., a real-time database, user authentication, and hosting. We can use these services with the help of the Firebase iOS SDK to create apps without writing any server code.

History of Firebase:

Firebase evolved from Envolv. Envolv is a prior startup founded by James Tamplin and Andrew Lee in 2011. Envolv provided developers an API which allowed the integration of online chat functionality into their websites. After releasing the chat service, it found that the envlove was being used to pass application data, which were not chat messages. Developers used Envolv to sync application to separate the real-time architecture and the chat system which powered it. In September 2011, Tamplin and Lee founded firebase as a separate company. It was lastly launched to the public in April 2012.

Firebase Real-time Database was the first product of firebase. It is an API which syncs application data across Android, iOS, and Web devices. It gets stored on Firebase's cloud. Then the firebase real-time database helps the developers to build real-time, collaborative applications.



Why use Firebase?

- Firebase manages real-time data in the database. So, it easily and quickly exchanges the data to and from the database. Hence, for developing mobile apps such as live streaming, chat messaging, etc., we can use Firebase.
- Firebase allows syncing real-time data across all devices - iOS, Android, and Web - without refreshing the screen.

- Firebase provides integration to Google Advertising, Ad-Mob, Data Studio, Big Query Double Click, Play Store, and Slack to develop our apps with efficient and accurate management and maintenance.
- Everything from databases, analytics to crash reports are included in Firebase. So, the app development team can stay focused on improving the user experience.
- Firebase applications can be deployed over a secured connection to the firebase server.
- Firebase offers a simple control dashboard.
- It offers an number of useful services to choose from.

Pros and Cons of Firebase:

Firebase has a lot of pros or advantages. Apart from the advantages, it has disadvantages too. Let's take a look at these advantages and disadvantages:

Pros:

- Firebase is a real-time database.
- It has massive storage size potential.
- Firebase is serverless.
- It is highly secure.
- It is the most advanced hosted BaaS solution.
- It has minimal setup.
- It provides three-way data binding via angular fire.
- It provides simple serialization of app state.
- We can easily access data, files, auth, and more.
- There is no server infrastructure required to power apps with data.
- It has JSON storage, which means no barrier between data and objects.

Cons:

- Firebase is not widely used, or battle-tested for enterprises.
- It has very limited querying and indexing.
- It provides no aggregation.
- It has no map-reduce functionality.
- It cannot query or list users or stored files.

4.TESTING IMPLEMENTATION

The term implementation has different meanings ranging from the conversion of a basic application to a complete replacement of a computer system. The procedures however, are virtually the same. Implementation includes all those activities that take place to convert from old system to new. The new system may be totally new replacing an existing manual or automated system or it may be major modification to an existing system. The method of implementation and time scale to be adopted is found out initially. Proper implementation is essential to provide a reliable system to meet organization requirement.

4.1 UNIT TESTING:

Introduction:

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure. In object-oriented programming, a unit is often an entire interface, such as a class, but could be an individual method. Unit tests are short code fragments created by programmers or occasionally by white box testers during the development process. It forms the basis for component testing. Ideally, each test case is independent from the others. Substitutes such as method stubs, mock objects, fakes, and test harnesses can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

Benefits:

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict, written contract that the piece of code must satisfy. As a result, it affords several benefits.

1) Find problems early: Unit testing finds problems early in the development cycle. In test-driven development (TDD), which is frequently used in both extreme programming and scrum, unit tests are created before the code itself is written. When the tests pass, that code is considered complete. The same unit tests are run against that function frequently as the larger code base is developed either as the code is changed or via an automated process with the build. If the unit tests fail, it is considered to be a bug either in the changed code or the tests themselves. The unit tests then allow the location of the fault or failure to be easily traced. Since the unit tests alert the development team of the problem before handing the code off to testers or clients, it is still early in the development process.

2) Facilitates Change: Unit testing allows the programmer to refactor code or upgrade system libraries at a later date, and make sure the module still works correctly (e.g., in regression testing). The procedure is to write test cases for all functions and methods so that whenever a change causes a fault, it can be quickly identified. Unit tests detect changes which may break a design contract.

3) Simplifies Integration: Unit testing may reduce uncertainty in the units themselves and can be used in a bottom-up testing style approach. By testing the parts of a program first and then testing the sum of its parts, integration testing becomes much easier.

4) Documentation: Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit, and how to use it, can look at the unit tests to gain a basic understanding of the unit's interface (API). Unit test cases embody characteristics that are critical to the success of the unit. These characteristics can indicate appropriate/inappropriate use of a unit as well as negative behaviours that are to be trapped by the unit. A unit test case, in and of itself, documents these critical characteristics, although many software development environments do not rely solely upon code to document the product in development.

4.2 INTEGRATION TESTING:

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

Purpose:

The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items. These "design items", i.e., assemblages (or groups of units), are exercised through their interfaces using black-box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface. Test cases are constructed to test whether all the components within assemblages interact correctly, for example across procedure calls or process activations, and this is done after testing individual modules, i.e., unit testing. The overall idea is a "building block" approach, in which verified assemblages are added to a verified base which is then used to support the integration testing of further assemblages. Software integration testing is performed according to the software development life cycle (SDLC) after module and functional tests. The cross-dependencies for software integration testing are: schedule for integration testing, strategy and selection of the tools used for integration, define the cyclomathical complexity of the software and software architecture, reusability of modules and life-cycle and versioning management. Some different types of integration testing are big-bang, top-down, and bottom-up, mixed (sandwich) and risky-hardest. Other Integration Patterns are: collaboration integration, backbone integration, layer integration, client-server integration, distributed services integration and high-frequency integration.

Big Bang:

In the big-bang approach, most of the developed modules are coupled together to form a complete software system or major part of the system and then used for integration testing. This method is very effective for saving time in the integration testing process. However, if the test cases and their results are not recorded properly, the entire integration process will be more complicated and may prevent the testing team from achieving the goal of integration testing. A type of big-bang integration testing is called "usage model testing" which can be used in both software and hardware integration testing. The basis behind this type of integration testing is to run user-like workloads in integrated user-like environments. In doing the testing in this manner, the environment is proofed, while the individual components are proofed indirectly through their use. Usage Model testing takes an optimistic approach to testing, because it

expects to have few problems with the individual components. The strategy relies heavily on the component developers to do the isolated unit testing for their product. The goal of the strategy is to avoid redoing the testing done by the developers, and instead flesh-out problems caused by the interaction of the components in the environment. For integration testing, Usage Model testing can be more efficient and provides better test coverage than traditional focused functional integration testing. To be more efficient and accurate, care must be used in defining the user-like workloads for creating realistic scenarios in exercising the environment. This gives confidence that the integrated environment will work as expected for the target customers.

Top-down And Bottom-up:

Bottom-up testing is an approach to integrated testing where the lowest level components are tested first, then used to facilitate the testing of higher level components. The process is repeated until the component at the top of the hierarchy is tested. All the bottom or low-level modules, procedures or functions are integrated and then tested. After the integration testing of lower level integrated modules, the next level of modules will be formed and can be used for integration testing. This approach is helpful only when all or most of the modules of the same development level are ready. This method also helps to determine the levels of software developed and makes it easier to report testing progress in the form of a percentage. Top-down testing is an approach to integrated testing where the top integrated modules are tested and the branch of the module is tested step by step until the end of the related module. Sandwich testing is an approach to combine top down testing with bottom up testing.

4.3 SOFTWARE VERIFICATION AND VALIDATION:

Introduction:

In software project management, software testing, and software engineering, verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfill its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle. Validation checks that the product design satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements. This is done through dynamic testing and other forms of review. Verification and validation are not the same thing, although they are often confused. Boehm succinctly expressed the difference between

- Validation: Are we building the right product?
- Verification: Are we building the product right?

Software Verification: The process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.

Software Validation: The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements.

From Testing Perspective:

- Fault – wrong or missing function in the code.
- Failure – the manifestation of a fault during execution.
- Malfunction – according to its specification the system does not meet its specified functionality.

4.4 SYSTEM TESTING:

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

System testing is performed on the entire system in the context of a Functional Requirement Specification(s) (FRS) and/or a System Requirement Specification (SRS). System testing tests not only the design, but also the behaviour and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software/hardware requirements specification(s).

5.SYSTEM DESIGN

5.1 Table Design:

Database Table

Firestore Database

jobhunt

Go to docs

Project Overview

Build

- Authentication
- Firestore Database
- Realtime Database
- Storage
- Hosting
- Functions
- Machine Learning

Release & Monitor

Crashlytics, Performance, Test La...

Analytics

Dashboard, Realtime, Events, Conv...

Extensions

Spark

Free \$0/month

Upgrade

Realtime Database

Data Rules Backups Usage


https://jobhunt-3daf3-default-rtdb.firebaseio.com/

Employer_Details

- MWizpOUE8AmXsGlcwsU
 - address: "102/North Cotton Road"
 - city: "Bangalore "
 - description: "An experienced and fully qualified candidates w..."
 - dtm: "Date:Wednesday, 14 Apr 2021, time:10:30 AM"
 - email: "7904662375"
 - exp: "20210414"
 - id: "-MWizpOUE8AmXsGlcwsU"
 - name: "VPRN Software Corporations"
 - num: "2"
 - qualification: "Web designer, Software Developer"
- MWthnXaUXTInWOLmhkk
- MXWqPrQShhq9ORP0kOy

User_Profile

Database location: United States (us-central1)

 **Firebase**

Project Overview

Build

Authentication

Firestore Database

Realtime Database

Storage

Hosting

Functions

Machine Learning

Release & Monitor

Analytics

Extensions

Spark

Free \$0/month

Upgrade

jobhunt

Go to docs

Realtime Database

DataRulesBackupsUsage

https://jobhunt-3daf3-default-rtdb.firebaseio.com/

User_Profile

-MWxbR-NTg97Eqn8cg8L

address: "fmf"

age: "dj Dr"

branch: "hdd"

city: "ncnfd"

college_nameee: "bdhd"

current_location: "bdnd"

degree: "shf"

degree_percentage: "bdn"

email: "9994026550"

gender: "MaLe"

id: "-MWxbR-NTg97Eqn8cg8L"

name: "hssh"

profileimage: "https://firebasestorage.googleapis.com/v0/b/job..."

resumeimage: "https://firebasestorage.googleapis.com/v0/b/job..."

Database location: United States (us-central1)

5.2 Data Flow Diagram:

Data flow is the movement of origin to a specific destination indicated by lines and arrows. Data flow diagram is the graphical representation of a data movement, processes and files used in the support of an information system.

Data Flow Diagram Symbols:

All data flow diagrams include four main elements: entity, process, data store and data flow.

External Entity – Also known as actors, sources or sinks, and terminators, external entities produce and consume data that flows between the entity and the system being diagrammed. These data flows are the inputs and outputs of the DFD. Since they are external to the system being analyzed, these entities are typically placed at the boundaries of the diagram. They can represent another system or indicate a subsystem.

Process – An activity that changes or transforms data flows. Since they transform incoming data to outgoing data, all processes must have inputs and outputs on a DFD. This symbol is given a simple name based on its function, such as “Ship Order,” rather than being labeled “process” on a diagram. In Gane -Sarson notation, a rectangular box is used and may be labeled with a reference number, location of where in the system the process occurs and a short title that describes its function. Processes are typically oriented from top to bottom and left to right on a data flow diagram.

Data Store – A data store does not generate any operations but simply holds data for later access. Data stores could consist of files held long term or a batch of documents stored briefly while they wait to be processed. Input flows to a data store include information or operations that change the stored data. Output flows would be data retrieved from the store.

Data Flow – Movement of data between external entities, processes and data stores is represented with an arrow symbol, which indicates the direction of flow. This data could be electronic, written or verbal. Input and output data flows are labeled based on the type of data or its associated process or data store, and this name is written alongside the arrow.

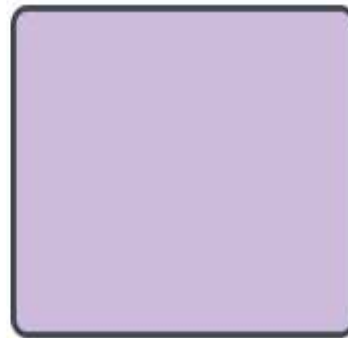
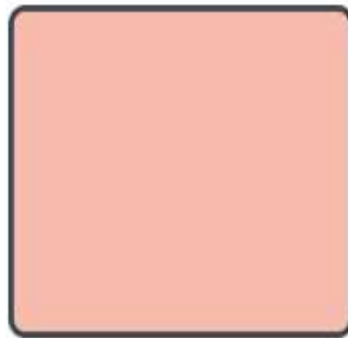
Source/Destination:

Notation

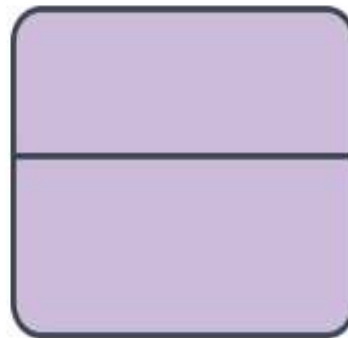
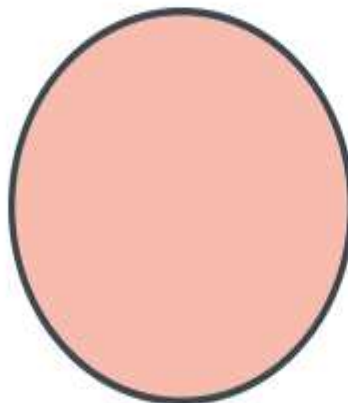
Yourdon and Coad

Gane and Sarson

External Entity



Process



Data Store

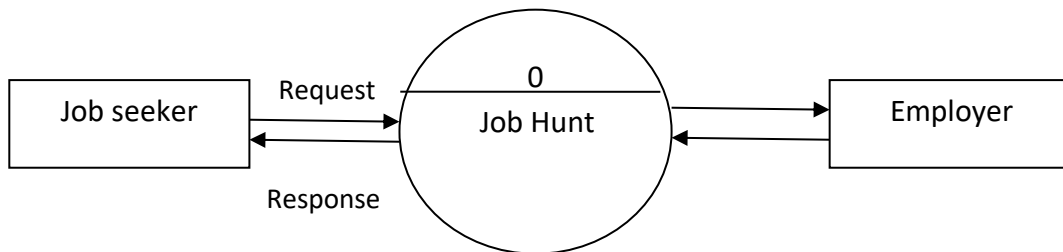


Data Flow

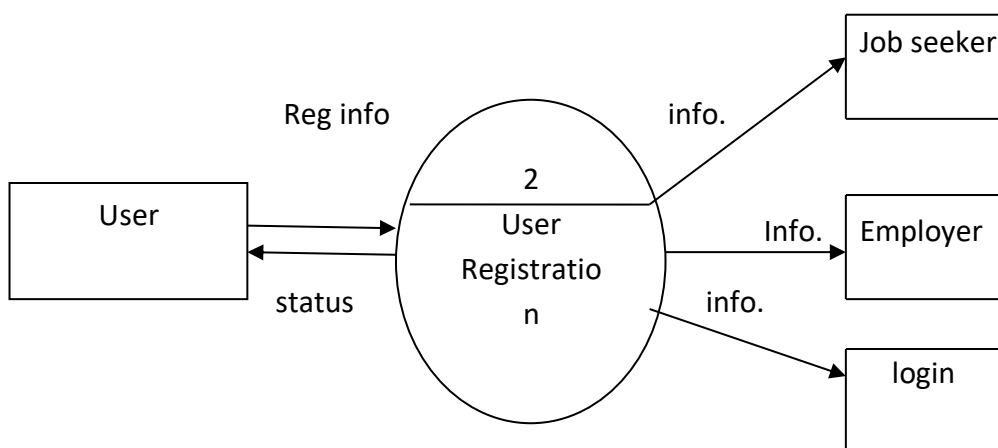
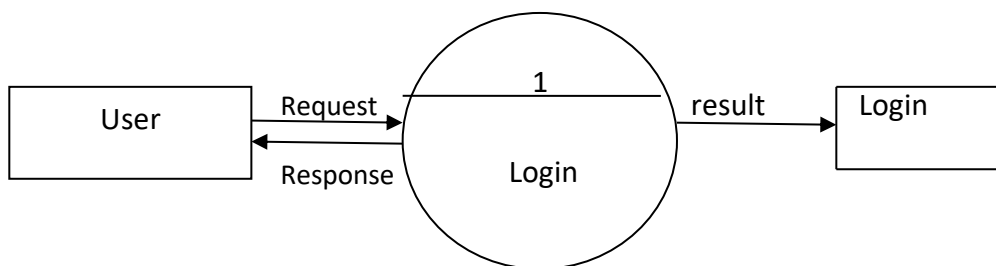


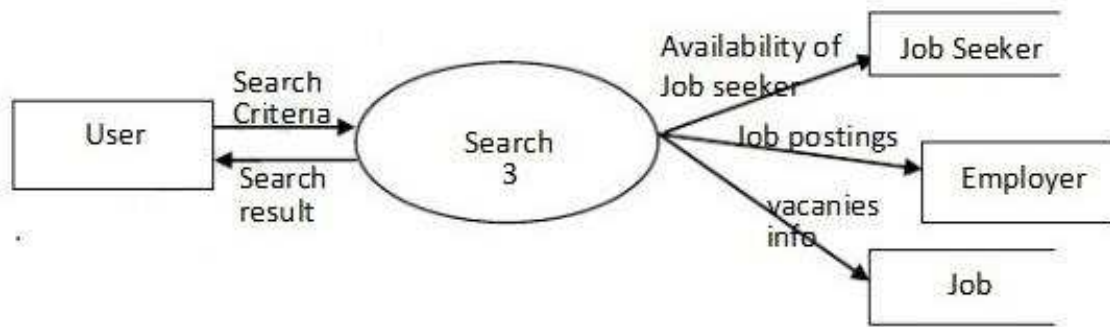
DATA FLOW DIAGRAM

Level: 0

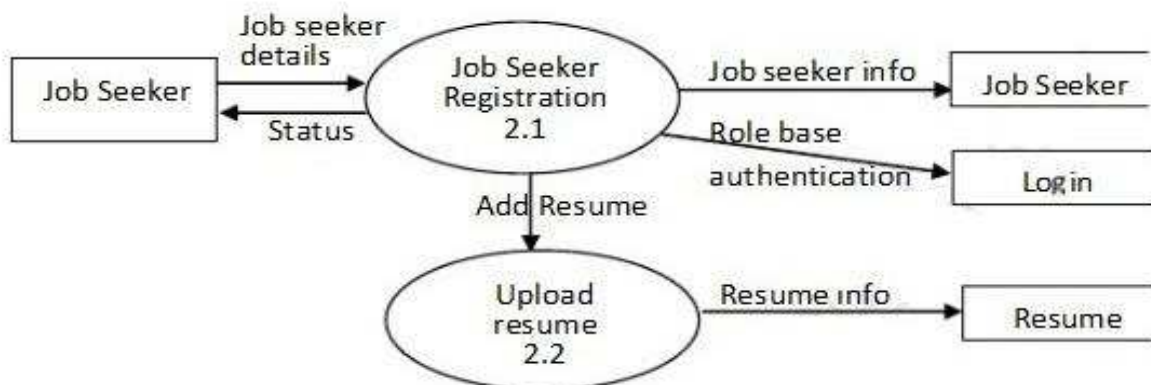
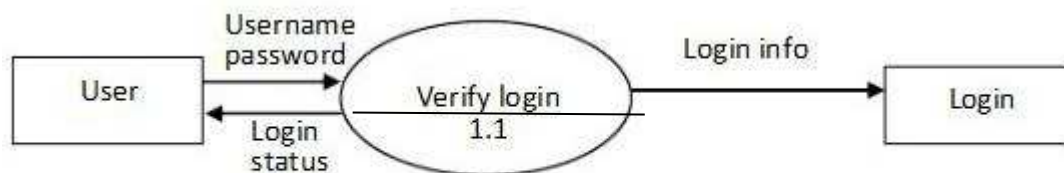


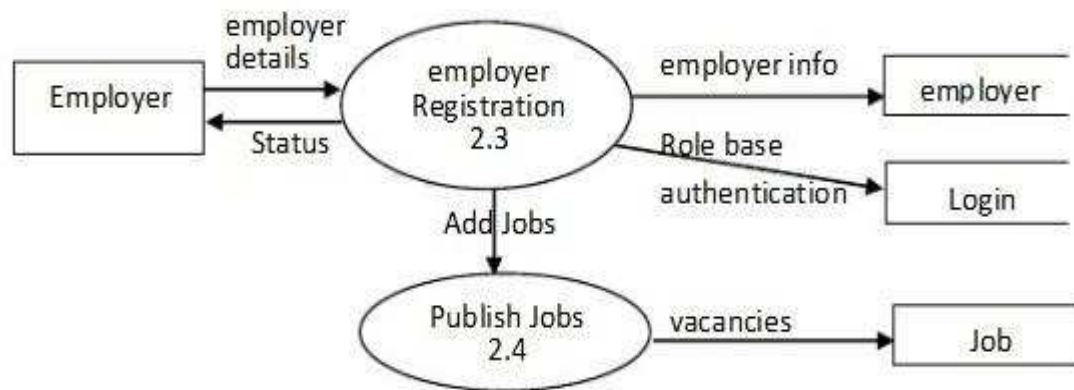
Level: 1



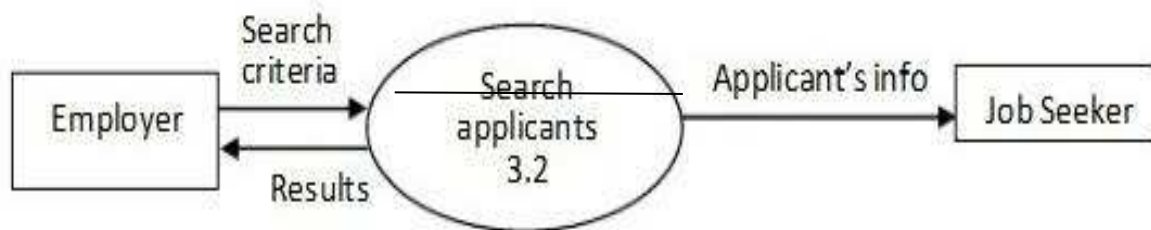


Level: 2

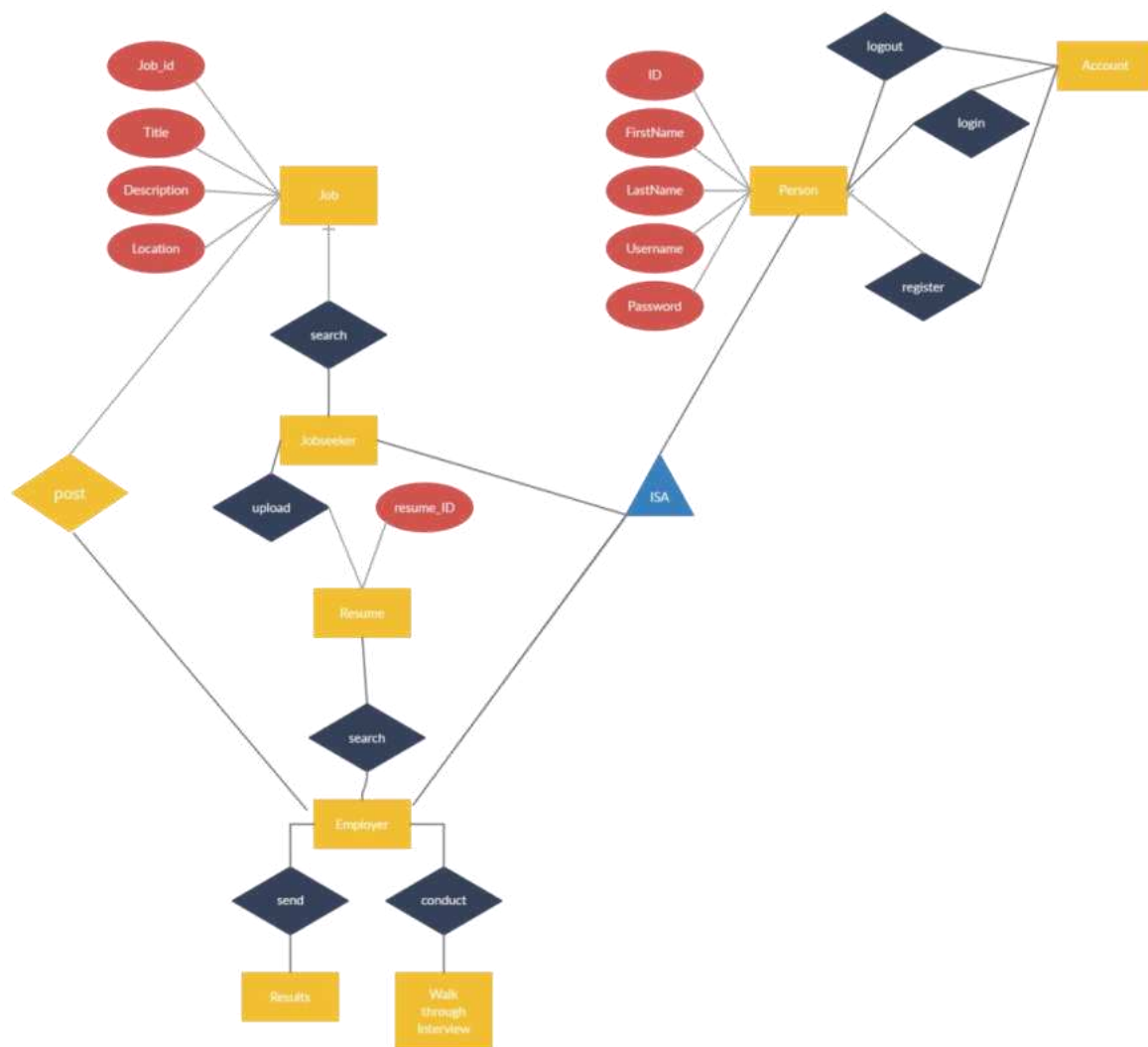




Level: 3



Level: 4

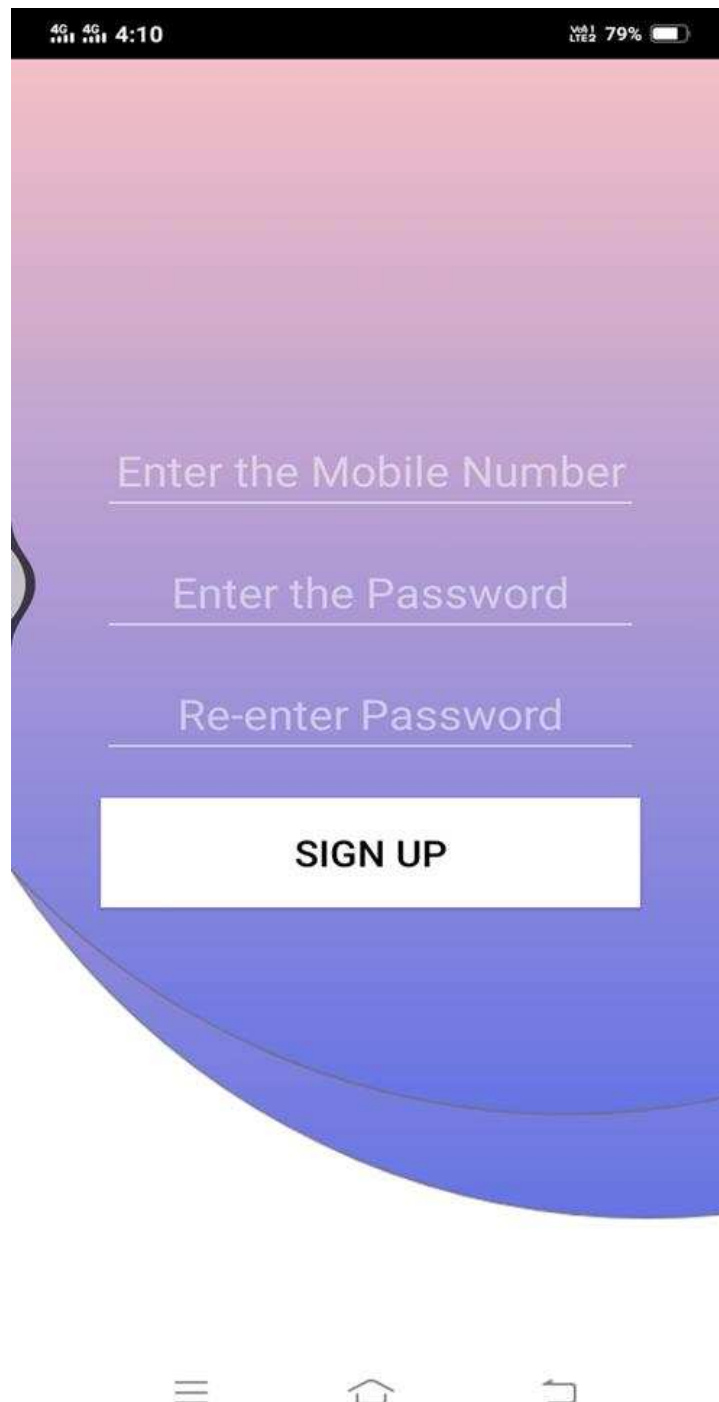


5.3 FORM DESIGN:

WELCOME PAGE



SIGN UP PAGE



4G 4G 4:10 VoLTE 79%

Enter the Mobile Number

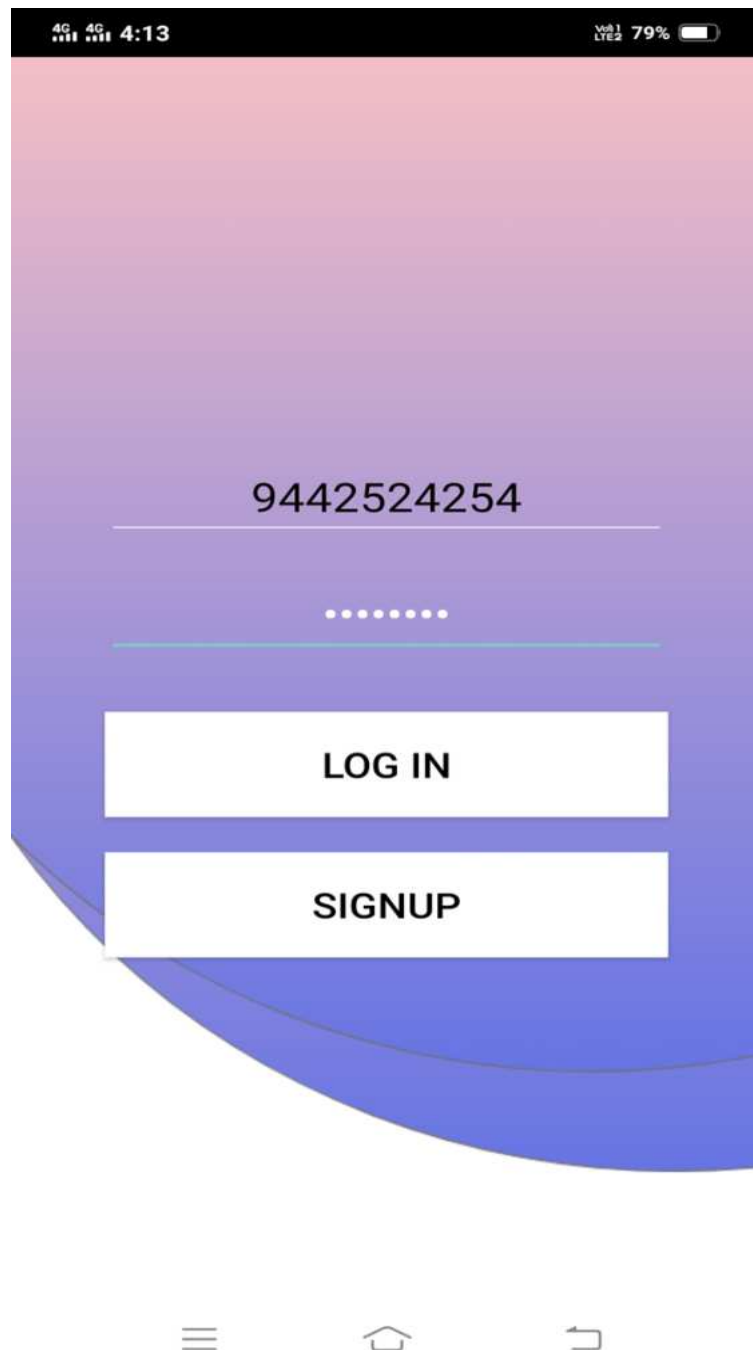
Enter the Password

Re-enter Password

SIGN UP

≡ ⌂ ↩

LOGIN PAGE



A mobile application login page with a purple-to-pink gradient background. The page features a status bar at the top with '4G LTE2' and '79%' battery. The main content area has a text input field containing '9442524254', a password input field with seven dots, and two buttons labeled 'LOG IN' and 'SIGNUP'. At the bottom, there are three navigation icons: a hamburger menu, a home icon, and a back arrow.

4G LTE2 79% 4:13

9442524254

.....

LOG IN

SIGNUP

≡ 🏠 ↩

HOME PAGE



JOB DESCRIPTION PAGE

11:53

4G

85%

Your Ad

Job Description

Name of the company/Office

Qualification Required

Address

City

No of Candidates required

Description

date

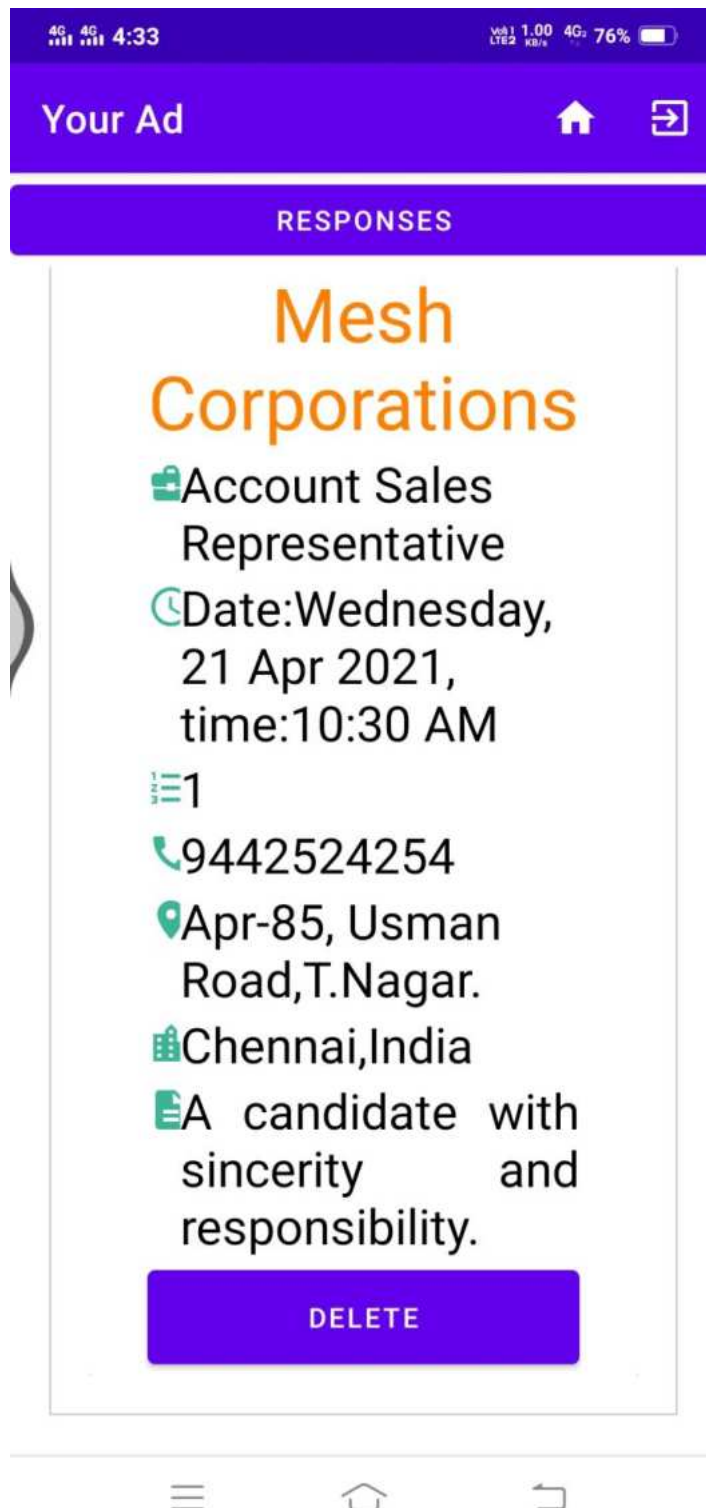
time

DATE PICKER

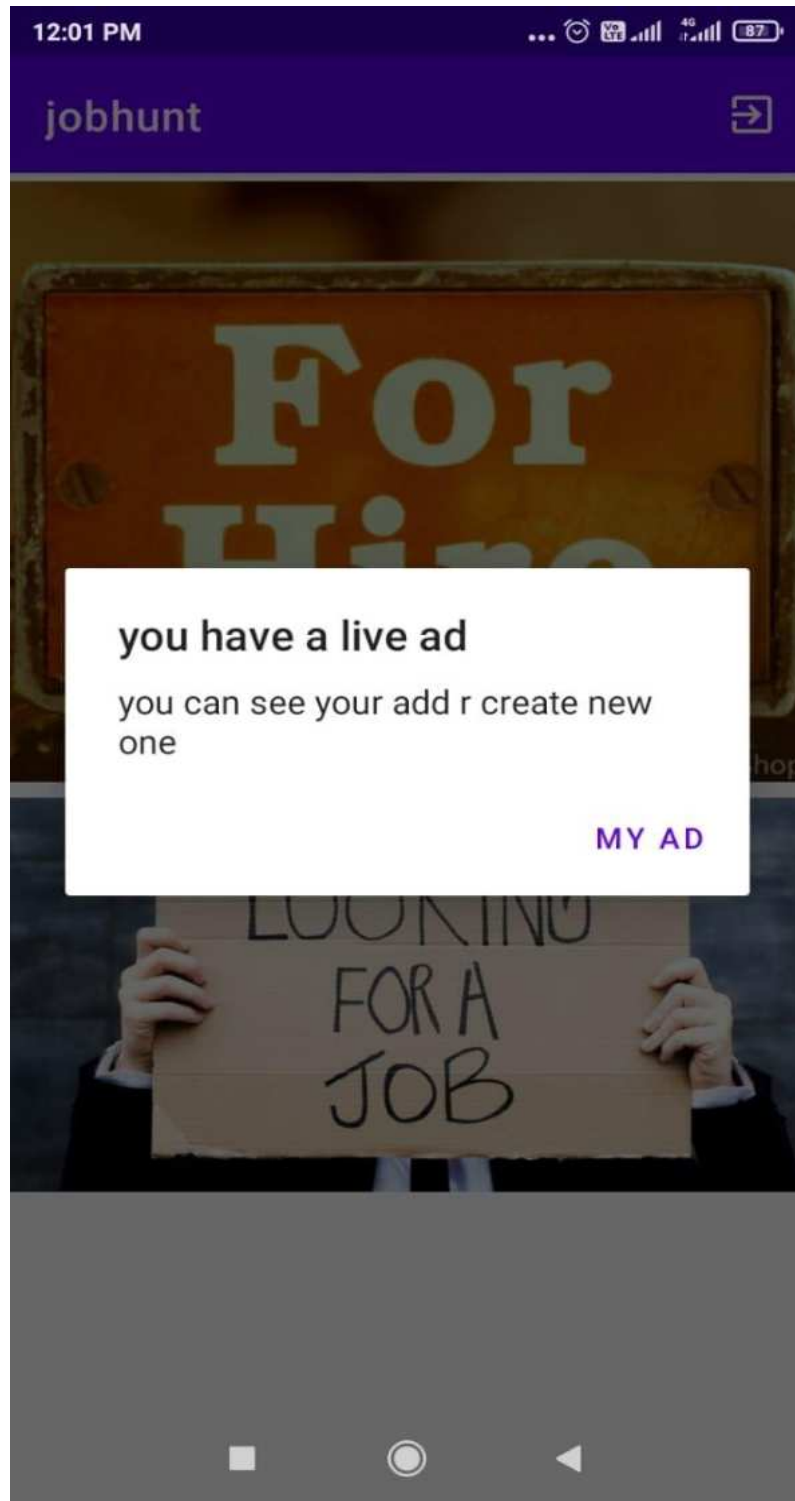
TIME PICKER

SUBMIT

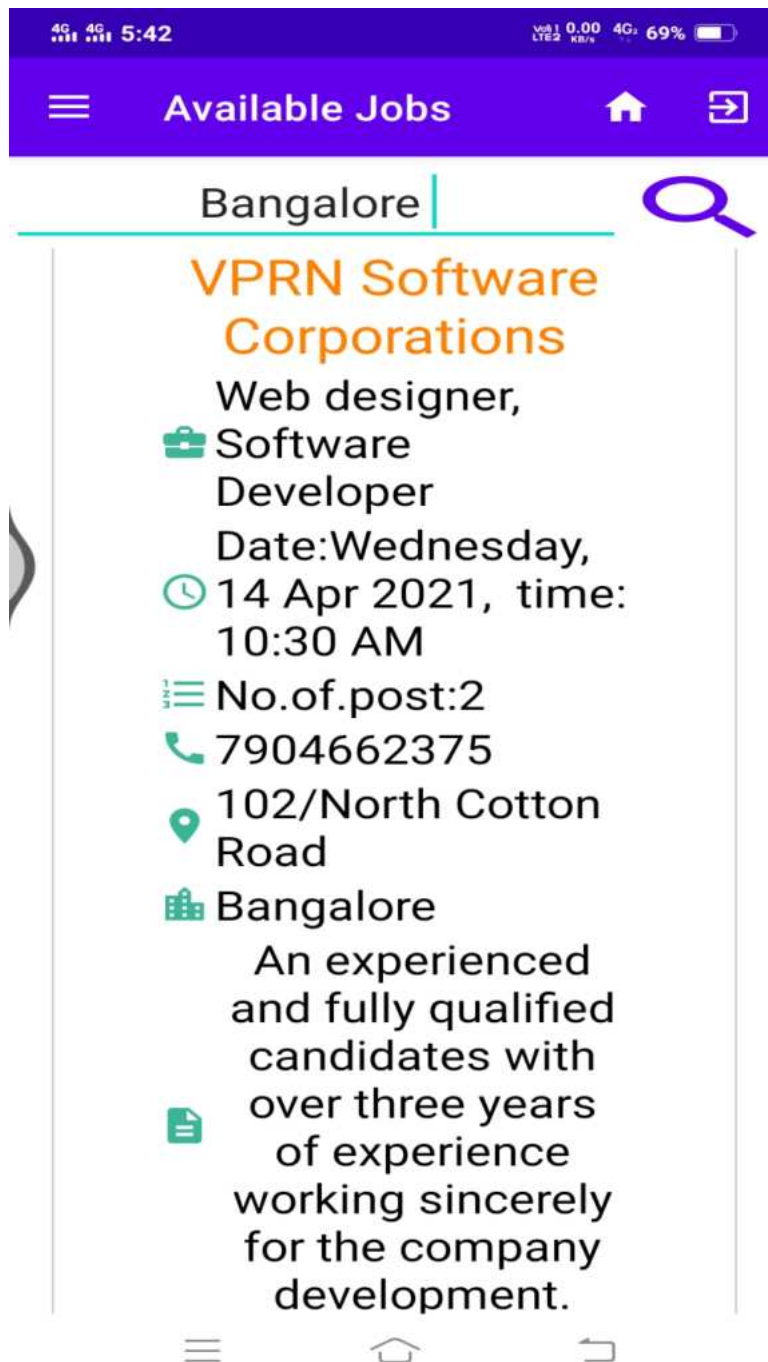
ADVERTISEMENT PAGE



ADVERTISEMENT NOTIFICATION PAGE



AVAILABLE JOBS PAGE WITH SEARCH FUNCTION




APPLICANT DETAILS WITH SMS OPTION

4G 4G 5:39
LTE2 0.00 KB/s 4G 69%

Your Ad

Profile image
resume image



WORK EXPERIENCE

SENIOR JAVASCRIPT DEVELOPER 10/2018 - 03/2019

- Developed and maintained web applications using JavaScript, HTML, and CSS.
- Collaborated with the design team to create user-friendly interfaces.
- Optimized application performance and reduced load times.
- Implemented security measures to protect user data.
- Mentored junior developers and provided technical support.

SENIOR JAVASCRIPT DEVELOPER 03/2017 - 09/2018

- Developed and maintained web applications using JavaScript, HTML, and CSS.
- Collaborated with the design team to create user-friendly interfaces.
- Optimized application performance and reduced load times.
- Implemented security measures to protect user data.
- Mentored junior developers and provided technical support.

EDUCATION

GOVERNMENT UNIVERSITY 2013 - 2016

PROFESSIONAL SKILLS

- Strong understanding of JavaScript, HTML, and CSS.
- Experience with web development frameworks like React and Angular.
- Strong problem-solving and analytical skills.
- Excellent communication and teamwork skills.
- Ability to learn new technologies quickly.
- Strong attention to detail and quality assurance.
- Experience with version control systems like Git.

SEND SMS

Name: Dharun
Age: 25/03/1994 ,26
Sex: Male
Degree: BBA
Branch: Business and
Location: Marketing
Address: Chennai
City: 25,Arcot Road
Valasaravakkam

5.4 SOURCE CODE:

//Applied_Job.java

```
package com.example.jobhunt;
public class Applied_Jobs {
String user_id;
    public Applied_Jobs() {
    }
    public Applied_Jobs(String user_id) {
        this.user_id = user_id;
    }
    public String getUser_id() {
        return user_id;
    }
    public void setUser_id(String user_id) {
        this.user_id = user_id;
    }
}
```

//Employer_Details.java

```
package com.example.jobhunt;
public class Employer_Details {
    private String
name,qualification,address,dttm,num,exp,city,email,id,description;
    public Employer_Details() {
    }
    public Employer_Details(String name, String qualification,
String address, String dttm, String num, String exp, String city,
String email, String id, String description) {
        this.name = name;
        this.qualification = qualification;
        this.address = address;
        this.dttm = dttm;
        this.num = num;
        this.exp = exp;
        this.city = city;
        this.email = email;
        this.id = id;
        this.description = description;
    }
    public String getDescription() {return description;}
    public void setDescription(String description) {
        this.description = description;
    }
    public String getId() {return id;}
    public void setId(String id) {this.id = id;}
    public String getCity() {return city;}
    public void setCity(String city) {this.city = city;}
    public String getEmail() {return email;}
    public void setEmail(String email) {this.email = email;}
}
```

```

    public String getExp() {return exp;}
    public void setExp(String exp) {this.exp = exp;}
    public String getAddress() {return address;}
    public void setAddress(String address) {this.address =
address;}
    public String getDttm() {return dttm;}
    public void setDttm(String dttm) {this.dttm =dttm;}
    public String getName() {return name;}
    public void setName(String name) {this.name=name;}
    public String getNum() {return num;}
    public void setNum(String num) {this.num = num;}
    public String getQualification() {
        return qualification;
    }
    public void setQualification(String qualification) {
        this.qualification = qualification;
    }
}

```

//Employportalupload.java

```

package com.example.jobhunt;
class Employportalupload {
    String name; String qualification; String address; String dttm;
String num;
    String exp;
    String city;
    String email;
    String id;
    String description;

    public Employportalupload(String namee, String qualificationn,
String addresss, String dttmm, String numm, String exp, String
city, String email, String id, String description) {
        this.name=namee;
        this.qualification=qualificationn;
        this.address=addresss;
        this.dttm=dttmm;
        this.num=numm;
        this.exp=exp;
        this.city=city;
        this.email=email;
        this.id=id;
        this.description=description;    }
    public String getDescription() {
        return description;    }
    public void setDescription(String description) {
        this.description = description;    }
    public String getId() {
        return id;    }
    public void setId(String id) {
        this.id = id;    }
    public String getEmail() {

```

```

        return email;    }
    public void setEmail(String email) {
        this.email = email;    }
    public String getCity() {
        return city;    }
    public void setCity(String city) {
        this.city = city;    }
    public String getExp() {
        return exp;    }
    public void setExp(String exp) {
        this.exp = exp;    }
    public String getName() {
        return name;    }
    public void setName(String name) {
        this.name = name;    }
    public String getQualification() {
        return qualification;    }
    public void setQualification(String qualification) {
        this.qualification = qualification;
    }
    public String getAddress() {
        return address;    }
    public void setAddress(String address) {
        this.address = address;    }
    public String getDttm() {
        return dttm;    }
    public void setDttm(String dttm) {
        this.dttm = dttm;    }
    public String getNum() {
        return num;    }
    public void setNum(String num) {
        this.num = num;
    }
}

```

//Home.java

```

package com.example.jobhunt;
import ...
public class Home extends AppCompatActivity {
    ImageView hire,work;
    String a1;
    private Query query,query1,query2;
    private SharedPreferences.Editor editor;
    private SharedPreferences sharedPreferences;
    DatabaseReference
    databaseReference,databaseReference1,databaseReference2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);
        sharedPreferences=getSharedPreferences("alreadylogged",
        Context.MODE_PRIVATE);
    }
}

```

```

        a1=sharedPreferences.getString("phonenumber","");
        databaseReference=
FirebaseDatabase.getInstance().getReference("user");
        query=databaseReference.orderByChild("email").equalTo(a1);
        databaseReference1=
FirebaseDatabase.getInstance().getReference("Employer_Details");

query1=databaseReference1.orderByChild("email").equalTo(a1);
databaseReference2=FirebaseDatabase.getInstance().getReference("User_Profile");
query2=databaseReference2.orderByChild("email").equalTo(a1);
        editor=sharedPreferences.edit();
        hire=(ImageView)findViewById(R.id.work_hire);

        work=(ImageView)findViewById(R.id.work_employee);
        work.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                /* Intent i1=new Intent(Home.this,JobList.class);
                startActivity(i1);*/
                query2.addValueEventListener(new
ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                        if (dataSnapshot.getChildrenCount() == 0) {
                            AlertDialog.Builder builder=new
AlertDialog.Builder(Home.this);
                            builder.setTitle("user profile");
                            builder.setMessage("For accessing list of jobs your
profile is must");
                            builder.setPositiveButton("Enter My Profile", new
DialogInterface.OnClickListener() {
                                @Override
                                public void onClick(DialogInterface dialogInterface, int i) {
                                    Intent i1=new Intent(Home.this,userProfile.class);
                                    startActivity(i1);
                                }
                            });
                            builder.create().show();
                        }
                        else
                        {
                            Intent i1=new Intent(Home.this,JobList.class);
                            startActivity(i1);
                        }
                    }
                @Override
                public void onCancelled(@NonNull DatabaseError databaseError) {
                    Toast.makeText(getApplicationContext(),databaseError.getClass().toString(),Toast.LENGTH_SHORT).show();
                }
            }
        });

```



```

        });
    }
});
hire.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        query1.addValueEventListener(new ValueEventListener()
        {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                if (dataSnapshot.getChildrenCount() == 0) {
                    Intent i1=new Intent(Home.this,Employportal.class);
                    startActivity(i1);
                }
            }
            else
            {
                AlertDialog.Builder builder=new
                AlertDialog.Builder(Home.this);
                builder.setTitle("you have a live ad");
                builder.setMessage("you can see your add r create new one");
                builder.setPositiveButton("My ad", new
                DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        Intent i1=new Intent(Home.this,Job_existing.class);
                        startActivity(i1);
                    }
                });
                /*
                builder.setNegativeButton("new ad", new
                DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface
                    dialogInterface, int i) {
                        Intent i1=new
                        Intent(Home.this,Employportal.class);
                        startActivity(i1);
                    }
                });*/
                builder.create().show();
            }
        }
    }
    @Override
    public void onCancelled(@NonNull DatabaseError
    databaseError) {
        Toast.makeText(getApplicationContext(),databaseError.getClass().toS
        tring(),Toast.LENGTH_SHORT).show();
    }
});
});

```

```

        }

    });
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_actionbar, menu);
    return true;
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    int id=item.getItemId();
    if (id==R.id.logout)
    {
        editor.putString("phonenumber", "");
        editor.commit();
        Intent i1 = new Intent(Home.this, logIn.class);
        startActivity(i1);
        finish();
    }
    return super.onOptionsItemSelected(item);
}
}

```

//Job_existing.java

```

package com.example.jobhunt;
import ...

public class Job_existing extends AppCompatActivity {
    private RecyclerView recyclerView_job;
    private Query query;
    ArrayList<Employer_Details> list;
    private SharedPreferences.Editor editor;
    private SharedPreferences sharedPreferences;
    private String phonenumber;
    private Employer_Details emp;
    private ViewHolderJobExisting adapter;
    Button responses;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_job_existing);
        getSupportActionBar().setTitle("job existing");
        responses=findViewById(R.id.job_response);
        sharedPreferences = getSharedPreferences("alreadylogged",
Context.MODE_PRIVATE);
        editor = sharedPreferences.edit();
        phonenumber=sharedPreferences.getString("phonenumber","");

        recyclerView_job=(RecyclerView)findViewById(R.id.recycler_jobexisti
ng);
        recyclerView_job.setLayoutManager(new

```

```

LinearLayoutManager(this));
        recyclerView_job.addItemDecoration(new
DividerItemDecoration(recyclerView_job.getContext(),
DividerItemDecoration.VERTICAL));
        query=
FirebaseDatabase.getInstance().getReference("Employer_Details").ord
erByChild("email").equalTo(phonenummer);
        list=new ArrayList<>();
        getSupportActionBar().setTitle("Your Ad");
        editor = sharedPreferences.edit();
        query.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
                list.clear();
                for (DataSnapshot
dataSnapshot1:dataSnapshot.getChildren()) {
                    emp =
dataSnapshot1.getValue(Employer_Details.class);
                    list.add(emp);

                }
            }
        });
        adapter=new ViewHolderJobExisting(Job_existing.this,list);
        recyclerView_job.setAdapter(adapter);

        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

        }

    });
    responses.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {

        }

    });

    FirebaseDatabase.getInstance().getReference("Applied_Jobs").child(p
honenummer).addListenerForSingleValueEvent(new ValueEventListener()
{
    @Override
    public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
        if (dataSnapshot.getChildrenCount() == 0)
        {

        }

        Toast.makeText(getApplicationContext(),"There are no
responses",Toast.LENGTH_SHORT).show();
        }else
        {
            Intent i1=new
            Intent(Job_existing.this,JobResponses.class);

```

```

        startActivity(i1);
    }

}

@Override
public void onCancelled(@NonNull DatabaseError
databaseError) {

}

});
}

});
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_logandhome, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {

    int id=item.getItemId();
    if (id==R.id.logout)
    {

        editor.putString("phonenummer", "");
        editor.commit();
        Intent i1 = new Intent(Job_existing.this, logIn.class);
        startActivity(i1);
        finish();
    }
    if (id==R.id.home1)
    {
        Intent i1 = new Intent(Job_existing.this, Home.class);
        startActivity(i1);
        finish();
    }
    return super.onOptionsItemSelected(item);
}

@Override
public void onBackPressed()
{
    int backButtonCount=0;
    backButtonCount++;
    if(backButtonCount == 1)
    {
        Intent i1=new Intent(Job_existing.this,Home.class);
        startActivity(i1);
        finish();
    }
}

```

```
}  
}
```

//JobList.java

```
package com.example.jobhunt;  
import ...  
public class JobList extends AppCompatActivity implements  
NavigationView.OnNavigationItemSelectedListener {  
    Query query;  
    Button search_button;  
    EditText search;  
    String search_text="",id;  
    RecyclerView recyclerView_job;  
    private Query query1,query2;  
    private DatabaseReference databaseReference;  
    User_Profile user_profile;  
    ArrayList<Employer_Details> list;  
    Employer_Details emp;  
    ViewHolderJob adapter;  
    private SimpleDateFormat sdf;  
    ImageView profile,resume;  
    TextView  
name,age,sex,degree,branch,location,address,city,cgpa,college;  
    private ActionBarDrawerToggle toggle;  
    DrawerLayout drawerLayout;  
    NavigationView navigationView;  
    Button test;  
    private DatabaseReference databaseReference1;  
    String currentDateandTime,phonenummer,i,r;  
    private SharedPreferences.Editor editor;  
    private SharedPreferences sharedPreferences;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_job_list);  
        sharedPreferences=getSharedPreferences("alreadylogged",  
Context.MODE_PRIVATE);  
        phonenummer=sharedPreferences.getString("phonenummer","");  
        editor = sharedPreferences.edit();  
        databaseReference=  
FirebaseDatabase.getInstance().getReference("User_Profile");  
  
        query2=databaseReference.orderByChild("email").equalTo(phonenummer)  
        ;  
  
        drawerLayout=findViewById(R.id.user_drawer);  
        name=findViewById(R.id.drawer_name_matrimony);  
        age=findViewById(R.id.drawer_age_matrimony);  
        sex=findViewById(R.id.drawer_sex_matrimony);  
        degree=findViewById(R.id.drawer_degree_matrimony);  
        branch=findViewById(R.id.branch_name_matrimony);
```

```

        location=findViewById(R.id.drawer_locaton_matrimony);
        address=findViewById(R.id.drawer_address_matrimony);
        city=findViewById(R.id.drawer_city_matrimony);
        cgpa=findViewById(R.id.branch_cgpa_matrimony);
        college=findViewById(R.id.drawer_college_matrimony);
        profile=findViewById(R.id.profileimage_drawer_matrimony);
        resume=findViewById(R.id.resumeimage_drawer_matrimony);

        getSupportActionBar().setTitle("Job List");
        toggle=new
ActionBarDrawerToggle(JobList.this,drawerLayout,R.string.open,R.str
ing.close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

navigationView=(NavigationView)findViewById(R.id.matrimony_navigati
on);
        navigationView.setNavigationItemSelectedListener(this);
        View v1=navigationView.getHeaderView(0);

        name=v1.findViewById(R.id.drawer_name_matrimony);
        age=v1.findViewById(R.id.drawer_age_matrimony);
        sex=v1.findViewById(R.id.drawer_sex_matrimony);
        degree=v1.findViewById(R.id.drawer_degree_matrimony);
        branch=v1.findViewById(R.id.branch_name_matrimony);
        location=v1.findViewById(R.id.drawer_locaton_matrimony);
        address=v1.findViewById(R.id.drawer_address_matrimony);
        city=v1.findViewById(R.id.drawer_city_matrimony);
        cgpa=v1.findViewById(R.id.branch_cgpa_matrimony);
        college=v1.findViewById(R.id.drawer_college_matrimony);

        profile=v1.findViewById(R.id.profileimage_drawer_matrimony);
        resume=v1.findViewById(R.id.resumeimage_drawer_matrimony);

//Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
        drawer();

        search_button=findViewById(R.id.search_button);
        search=findViewById(R.id.search);

recyclerView_job=(RecyclerView)findViewById(R.id.recycler_job);
        recyclerView_job.setLayoutManager(new
LinearLayoutManager(this));
        recyclerView_job.addItemDecoration(new
DividerItemDecoration(recyclerView_job.getContext(),
DividerItemDecoration.VERTICAL));
        databaseReference=
FirebaseDatabase.getInstance().getReference("Employer_Details");

```

```

query=FirebaseDatabase.getInstance().getReference("Employer_Details
");
    list=new ArrayList<>();
    getSupportActionBar().setTitle("Available Jobs");
    resume.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i1=new
Intent(JobList.this,ImageFullView.class);
            i1.putExtra("image",r);
            startActivity(i1);

        }
    });
    profile.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i1=new
Intent(JobList.this,ImageFullView.class);
            i1.putExtra("image",i);
            startActivity(i1);

        }
    });
    check(search_text);
    search_button.setOnClickListener(new View.OnClickListener()
{
        @Override
        public void onClick(View view) {
            search_text=search.getText().toString();
            check(search_text);
            search_text="";

            //
query=FirebaseDatabase.getInstance().getReference("Employer_Details
").orderByChild("city").equalTo(search_text);
        }
    });
}
private void check(String search_text)
{
    String city=search_text;
    sdf = new SimpleDateFormat("yyyyMMdd");
    currentDateandTime = sdf.format(new Date());
    query.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
            list.clear();
            for (DataSnapshot
dataSnapshot1:dataSnapshot.getChildren()) {

```

```

        emp = dataSnapshot1.getValue(Employer_Details.class);
        if
(Integer.parseInt(currentDateandTime)<=Integer.parseInt(emp.getExp(
))) {
            if (city.equals(emp.getCity())) {
                list.add(emp);
            }
            else if (city.equals(""))
            {
                list.add(emp);
            }
        }
        else if
(Integer.parseInt(currentDateandTime)>Integer.parseInt(emp.getExp(
))) {
            dataSnapshot1.getRef().removeValue();
            Query query1 =
FirebaseDatabase.getInstance().getReference("Applied_Jobs").orderBy
Child(phonenummer);
            query1.addValueEventListener(new
ValueEventListener() {
                @Override
                public void onDataChange(@NonNull
DataSnapshot dataSnapshot2) {

                    for (DataSnapshot
dataSnapshot3:dataSnapshot2.getChildren()) {
                        dataSnapshot3.getRef().removeValue();
                        //Toast.makeText(context, "The AD successfully
deleted", Toast.LENGTH_SHORT).show();
                    }
                }
                @Override
                public void onCancelled(@NonNull
DatabaseError databaseError) {
                    Toast.makeText(getApplicationContext(),databaseError.toString(),Toa
st.LENGTH_SHORT).show();
                }
            });
        }
        adapter=new
ViewHolderJob(JobList.this,list,phonenummer);
        recyclerView_job.setAdapter(adapter);
    }
    @Override
    public void onCancelled(@NonNull DatabaseError
databaseError) {

    }
});

```



```

    }

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item)
    {
        return false;
    }
    private void drawer()
    {
        query2.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
                for (DataSnapshot
dataSnapshot1:dataSnapshot.getChildren()) {
                    user_profile =
dataSnapshot1.getValue(User_Profile.class);
                    i=user_profile.getProfileimage();
                    r=user_profile.getResumeimage();

Picasso.with(getApplicationContext()).load(user_profile.getProfilei
mage()).into(profile);
Picasso.with(getApplicationContext()).load(user_profile.getResumeim
age()).into(resume);
                    name.setText(user_profile.getName());
                    age.setText(user_profile.getAge());
                    degree.setText(user_profile.getDegree());
                    sex.setText(user_profile.getGender());
                    branch.setText(user_profile.getBranch());
location.setText(user_profile.getCurrent_location());
cgpa.setText(user_profile.getDegree_percentage());
college.setText(user_profile.getCollege_namee());
                    city.setText(user_profile.getCity());
                    address.setText(user_profile.getAddress());
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError
databaseError) {

        }
    });
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_Logandhome,menu);
    return true;
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    if (toggle.onOptionsItemSelected(item))

```

```

        {
            return true;
        }
        int id=item.getItemId();
        if (id==R.id.Logout)
        {
            editor.putString("phonenumber", "");
            editor.commit();
            Intent i1 = new Intent(JobList.this, logIn.class);
            startActivity(i1);
            finish();
        }
        if (id==R.id.home1)
        {
            Intent i1 = new Intent(JobList.this, Home.class);
            startActivity(i1);
            finish();
        }
        return super.onOptionsItemSelected(item);
    }
    @Override
    public void onBackPressed()
    {
        int backButtonCount=0;
        backButtonCount++;
        if(backButtonCount == 1)
        {
            Intent i1=new Intent(JobList.this,Home.class);
            startActivity(i1);
            finish();
        }
    }
}

```

//JobResponses.java

```

package com.example.jobhunt;
import ...
public class JobResponses extends AppCompatActivity {
    private RecyclerView recyclerView_job;
    private Query query,query1;
    ArrayList<User_Profile> list;
    private SharedPreferences.Editor editor;
    private SharedPreferences sharedPreferences;
    private String phonenumber;
    private User_Profile emp;
    Applied_Jobs applied_jobs;
    String id;
    private ViewHolderJobResponse adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_job_responses);
    }
}

```

```

        getSupportActionBar().setTitle("Job Responces");

        sharedPreferences = getSharedPreferences("alreadylogged",
Context.MODE_PRIVATE);
        editor = sharedPreferences.edit();
        phonenumber=sharedPreferences.getString("onenumber", "");

recyclerView_job=(RecyclerView)findViewById(R.id.jobResponserecycle
rview);
        recyclerView_job.setLayoutManager(new
LinearLayoutManager(this));
        recyclerView_job.addItemDecoration(new
DividerItemDecoration(recyclerView_job.getContext(),
DividerItemDecoration.VERTICAL));
        query=
FirebaseDatabase.getInstance().getReference("Applied_Jobs").child(p
honenumber);
        query1=
FirebaseDatabase.getInstance().getReference("User_Profile");
        list=new ArrayList<>();
        getSupportActionBar().setTitle("Your Ad");
        query.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
                list.clear();
                for (DataSnapshot
dataSnapshot1:dataSnapshot.getChildren()) {
                    applied_jobs =
dataSnapshot1.getValue(Applied_Jobs.class);
                    query1.orderByChild("email").equalTo(applied_jobs.getUser_id()).add
ValueEventListener(new ValueEventListener() {
                        @Override
                        public void onDataChange(@NonNull
DataSnapshot snapshot) {
                            for (DataSnapshot
dataSnapshot2:snapshot.getChildren()) {
                                emp =
dataSnapshot2.getValue(User_Profile.class);
                                list.add(emp);
                            }
                        }
                    });
                }
            }
        });
        adapter=new
ViewHolderJobResponse(JobResponses.this,list,onenumber);
        recyclerView_job.setAdapter(adapter);
    }
    @Override
    public void onCancelled(@NonNull
DatabaseError error) {
    }
    });
}

```

```

        }
        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

        }

    });
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_Logandhome,menu);
    return true;
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    int id=item.getItemId();
    if (id==R.id.Logout)
    {
        editor.putString("phonenummer", "");
        editor.commit();

        Intent i1 = new Intent(JobResponses.this, logIn.class);
        startActivity(i1);
        finish();
    }
    if (id==R.id.home1)
    {
        Intent i1 = new Intent(JobResponses.this, Home.class);
        startActivity(i1);
        finish();
    }
    return super.onOptionsItemSelected(item);
}
}

```

//logIn.java

```

package com.example.jobhunt;
import ...
public class logIn extends AppCompatActivity {
    Button login,verify1;
    EditText email,password;
    private DatabaseReference mDatabase;
    private Query query;
    private user user1;
    private SharedPreferences sharedPreferences;
    private SharedPreferences.Editor editor;
    String verify,username,id;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```

        setContentView(R.layout.activity_log_in);
        verify1=findViewById(R.id.verify);
        sharedPreferences=getSharedPreferences("alreadylogged",
Context.MODE_PRIVATE);
        editor=sharedPreferences.edit();
        mDatabase =
FirebaseDatabase.getInstance().getReference("user");
        login=findViewById(R.id.login_auth);
        email=findViewById(R.id.email);
        password=findViewById(R.id.password);
        String s1=sharedPreferences.getString("phonenumber","");
        if (!s1.equals(""))
        {
            Intent i1=new Intent(this,Home.class);
            startActivity(i1);
            finish();
        }
        verify1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent i1=new Intent(logIn.this,SignUp.class);
                startActivity(i1);
            }
        });
        login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (email.getText().length()==10) {
                    if (password.getText().length()>=6)
                    {

FirebaseDatabase.getInstance().getReference("user").orderByChild("e
mail").equalTo(email.getText().toString()).addListenerForSingleValu
eEvent(new ValueEventListener() {
                            @Override
                            public void onDataChange(@NonNull
DataSnapshot dataSnapshot) {

                                if (dataSnapshot.getChildrenCount() == 0) {
                                    Toast.makeText(getApplicationContext(), "The entered Phone Number
doesn't exists", Toast.LENGTH_SHORT).show();
                                } else {
                                    for (DataSnapshot dataSnapshot1 :
dataSnapshot.getChildren()) {
                                        user1 = dataSnapshot1.getValue(user.class);
                                        verify = user1.getPassword();
                                        username = user1.getEmail();
                                        id = user1.getId();
                                    }
                                    if (verify.equals(password.getText().toString())) {
                                        editor.putString("phonenumber", username);

```

```

        editor.putString("ID", id);
        editor.commit();
        Intent i1 = new Intent(logIn.this, Home.class);
            startActivity(i1);
            finish();
        } else {

Toast.makeText(getApplicationContext(), "enter the correct
password", Toast.LENGTH_SHORT).show();
        }
        //
        Toast.makeText(getApplicationContext(), verify, Toast.LENGTH_SHORT).s
how();

    }
}
@Override
    public void onCancelled(@NonNull
DatabaseError databaseError) {
        Toast.makeText(getApplicationContext(),
databaseError.getClass().toString(), Toast.LENGTH_SHORT).show();
    }
});
}
    else
    {
        Toast.makeText(getApplicationContext(), "password must be greater
than 6 characters", Toast.LENGTH_SHORT).show();
    }
}
    else
    {
        Toast.makeText(getApplicationContext(), "Enter
The 10 digit mobile number", Toast.LENGTH_SHORT).show();
    }
}
});
@Override
    public void onBackPressed()
    {
        int backButtonCount=0;
        backButtonCount++;
        if(backButtonCount == 1)
        {
            Intent i1=new Intent(logIn.this, MainActivity.class);
            startActivity(i1);
            finish();
        }
    }
}
}

```

//MainActivity.java

```

package com.example.jobhunt;
import ...
public class MainActivity extends AppCompatActivity {
    Button textView;
    ImageView top,bottom;
    private Animation frombottom,frombottom1;
    private Handler mHandler = new Handler();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        top=findViewById(R.id.top);
        bottom=findViewById(R.id.bottom);
        frombottom= AnimationUtils.loadAnimation(this,R.anim.frombottom);
        frombottom1=AnimationUtils.loadAnimation(this,R.anim.frombottom1);
        bottom.setAnimation(frombottom1);
        top.setAnimation(frombottom);
        mHandler.postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent intent = new Intent(MainActivity.this,
logIn.class);
                startActivity(intent);
                finish();
            }
        }, 4000); // 4 seconds
        /*textView=findViewById(R.id.text);
        textView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent i1=new Intent(MainActivity.this,logIn.class);
                startActivity(i1);
            }
        });*/
    }
}

```

//SignUp.java

```

package com.example.jobhunt;
import ...
public class SignUp extends AppCompatActivity {
    EditText email,password,confirm;
    Button signup;
    private DatabaseReference mDatabase;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sign_up);
        email=findViewById(R.id.email_signup);
        password=findViewById(R.id.password_signup);
        confirm=findViewById(R.id.confirm_password);
        signup=findViewById(R.id.signup_auth);
    }
}

```

```

        mDatabase =
        FirebaseDatabase.getInstance().getReference("user");
        signup.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (email.getText().length()==10) {
                    if (password.getText().length()>=6) {
                        FirebaseDatabase.getInstance().getReference("user").orderByChild("email").equalTo(email.getText().toString()).addListenerForSingleValueEvent(new ValueEventListener() {
                            @Override
                            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                                if (dataSnapshot.getChildrenCount() == 0) {
                                    String id = mDatabase.push().getKey();
                                    String mail = email.getText().toString();
                                    String pass = password.getText().toString();
                                    if (pass.equals(confirm.getText().toString())) {
                                        mDatabase.child(id);
                                        mDatabase.child(id).child("email").setValue(mail);
                                        mDatabase.child(id).child("password").setValue(pass);
                                        mDatabase.child(id).child("id").setValue(id);
                                        Intent i1 = new Intent(SignUp.this, logIn.class);
                                        startActivity(i1);
                                        finish();
                                    } else {
                                        Toast.makeText(getApplicationContext(), "password and confirm password doesn't match", Toast.LENGTH_SHORT).show();
                                    }
                                }
                                //
                                Toast.makeText(getApplicationContext(), verify, Toast.LENGTH_SHORT).show();
                            } else {
                                Toast.makeText(getApplicationContext(), "The entered Phone Number already exists", Toast.LENGTH_SHORT).show();
                            }
                        }
                    }
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
                Toast.makeText(getApplicationContext(), databaseError.getClass().toString(), Toast.LENGTH_SHORT).show();
            }
        });
    }
    else
    {
        Toast.makeText(getApplicationContext(), "password must be greater

```



```

        than 6 characters",Toast.LENGTH_SHORT).show();
    }
    }
    else
    {
        Toast.makeText(getApplicationContext(),"Enter
The 10 digit mobile number",Toast.LENGTH_SHORT).show();
    }
}
});
}
}
}

```

//UserProfile.java

```

package com.example.jobhunt;
class Upload {
String
Name,Address,Timing,Contact_number,Category,Firmname,Description,Pr
oprietor_name,horoscopedescription,imageurl,city;
    public Upload(String name) {
        this.Name=name;
    }
    public Upload() {
    }
    public Upload(String address, String firmname, String
description, String proprietor_name, String city) {
        Address = address;
        Firmname = firmname;
        Description = description;
        Proprietor_name = proprietor_name;
        this.city = city;
    }
    public Upload(String Firmname, String address, String
contact_number, String Category, String Description, String
Proprietor_name) {
        this.Firmname = Firmname;
        this.Category=Category;
        this.Address=address;
        this.Contact_number=contact_number;
        this.Description=Description;
        this.Proprietor_name=Proprietor_name;
    }
    public Upload(String address, String contact_number, String
category, String firmname, String description, String
proprietor_name, String imageurl, String city) {
        Address = address;
        Contact_number = contact_number;
        Category = category;
        Firmname = firmname;
        Description = description;
        Proprietor_name = proprietor_name;
        this.imageurl = imageurl;
    }
}

```

```

        this.city = city;
    }
    public Upload(String horoscopedescription, String imageurl) {
        this.horoscopedescription = horoscopedescription;
        this.imageurl = imageurl;
    }
    public String getCity() {
        return city;    }
    public void setCity(String city) {
        this.city = city;    }
    public String getImageurl() {
        return imageurl;    }
    public void setImageurl(String imageurl) {
        this.imageurl = imageurl;    }
    public String getFirmname() {
        return Firmname;    }
    public void setFirmname(String firmname) {
        Firmname = firmname;    }
    public String getCategory() {
        return Category;    }
    public void setCategory(String category) {
        Category = category;    }
    public String getName() {
        return Name;    }
    public void setName(String name) {
        Name = name;    }
    public String getAddress() {
        return Address;    }
    public void setAddress(String address) {
        Address = address;    }
    public String getDescription() {
        return Description;    }
    public void setDescription(String description) {
        Description = description;    }
    public String getProprietor_name() {
        return Proprietor_name;    }
    public void setProprietor_name(String proprietor_name) {
        Proprietor_name = proprietor_name;    }
    public String getContact_number() {
        return Contact_number;    }
    public void setContact_number(String contact_number) {
        Contact_number = contact_number;
    }
}

```

//ViewHolderJob.java

```

package com.example.jobhunt;
import android.content.Context;
import android.content.DialogInterface;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

```

```

import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AlertDialog;
import androidx.recyclerview.widget.RecyclerView;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;
import java.util.ArrayList;
public class ViewHolderJob extends
RecyclerView.Adapter<com.example.jobhunt.ViewHolderJob.myViewHolder
> {
    private Context context;
    private ArrayList<Employer_Details> emp;
    String phonenumber;
    private DatabaseReference
mDatabase=FirebaseDatabase.getInstance().getReference("Applied_Jobs
");
    private Query query,query1;
    public ViewHolderJob(Context context,
ArrayList<Employer_Details> emp) {
        this.context = context;
        this.emp = emp;
    }
    public ViewHolderJob(Context context,
ArrayList<Employer_Details> emp, String phonenumber) {
        this.context = context;
        this.emp = emp;
        this.phonenumber = phonenumber;
    }
    @NonNull
    @Override
    public com.example.jobhunt.ViewHolderJob.myViewHolder
onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        return new
myViewHolder(LayoutInflater.from(context).inflate(R.layout.job_item
, parent, false));
    }
    @Override
    public void onBindViewHolder(@NonNull
com.example.jobhunt.ViewHolderJob.myViewHolder holder, int
position) {
        //
query1=mDatabase.child(phonenumber).orderByChild("orderid").equalTo
(emp.get(position).getId());
        holder.name_job.setText(emp.get(position).getName());

```

```

holder.qualification_job.setText(emp.get(position).getQualification
());
holder.date_job.setText(emp.get(position).getDttm());
holder.number_job.setText("No.ofpost:"+emp.get(position).getNum());
holder.address_job.setText(emp.get(position).getAddress());
holder.city.setText(emp.get(position).getCity());
holder.phonenumber.setText(emp.get(position).getEmail());
holder.description.setText(emp.get(position).getDescription());
holder.apply.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String id=emp.get(position).getEmail();
        query=FirebaseDatabase.getInstance().getReference("Applied_Jobs").c
hild(phonenumber).orderByChild("user_id").equalTo(id);

        FirebaseDatabase.getInstance().getReference("Applied_Jobs").child(p
hononenumber).orderByChild("user_id").equalTo(id).addListenerForSingl
eValueEvent(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
                if (dataSnapshot.getChildrenCount() == 0)
                {
                    String id1 = mDatabase.push().getKey();
                    mDatabase.child(phonenumber).child(id1).child("user_id").setValue(i
d);
                    holder.apply.setText("Applied");
                    Toast.makeText(context,"you have
successfully applied to this job",Toast.LENGTH_SHORT).show();
                }else
                {
                    Toast.makeText(context,"you have
already applied to this job",Toast.LENGTH_SHORT).show();
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError
databaseError) {
                Toast.makeText(context,databaseError.toString(),Toast.LENGTH_SHORT)
.show();
            }
        });
    }
});

FirebaseDatabase.getInstance().getReference("Applied_Jobs").child(p
hononenumber).orderByChild("user_id").equalTo(emp.get(position).getEm
ail()).addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
        if (dataSnapshot.getChildrenCount() == 0)

```

```

        {
            holder.apply.setText("Apply");
        }else
        {
            holder.apply.setText("Already Applied");
        }

    }

    @Override
    public void onCancelled(@NonNull DatabaseError
databaseError) {
    Toast.makeText(context,databaseError.toString(),Toast.LENGTH_SHORT)
        .show();
    }
    });
    //
    Picasso.with(context).load(list.get(position).getImageurl()).into(h
older.job_image);
    /*
    FirebaseDatabase.getInstance().getReference("job_fav").child(phonen
umber).orderByChild("orderid").equalTo(emp.get(position).getId()).a
ddListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
            if (dataSnapshot.getChildrenCount() == 0)
            {

holder.favour.setBackgroundResource(R.drawable.ic_baseline_favorite
_border_24);
            }else
            {
holder.favour.setBackgroundResource(R.drawable.favourite1);
            }
        }
        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

        }
    });
    holder.favour.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View view) {
        //
        Toast.makeText(context,phonenumber,Toast.LENGTH_SHORT).show();
        FirebaseDatabase.getInstance().getReference("user").orderByChild("i
d").equalTo(phonenumber).addValueEventListener(new
ValueEventListener() {

```

```

        @Override
        public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {

    FirebaseDatabase.getInstance().getReference("job_fav").child(phonen
umber).orderByChild("orderid").equalTo(emp.get(position).getId()).a
ddListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull
DataSnapshot dataSnapshot) {
            if (dataSnapshot.getChildrenCount()
== 0)
                {
                    String id=mDatabase.push().getKey();
mDatabase.child(phonenumber).child(id).child("orderid").setValue(em
p.get(position).getId());
holder.favour.setBackgroundResource(R.drawable.favourite1);
                }else
                {
                    AlertDialog.Builder builder=new AlertDialog.Builder(context);
                    builder.setTitle("Remove user from favourites");
                    builder.setMessage("Do you want to remove this user from
favourite?");
                    builder.setPositiveButton("yes", new
DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface dialogInterface, int i) {
mDatabase.child(phonenumber).orderByChild("orderid").equalTo(emp.ge
t(position).getId()).addListenerForSingleValueEvent(new
ValueEventListener() {
                            @Override
                            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                                for (DataSnapshot appleSnapshot: dataSnapshot.getChildren()) {
                                    appleSnapshot.getRef().removeValue();
                                    holder.favour.setBackgroundResource(R.drawable.ic_baseline_favorite
_border_24);
                                    Toast.makeText(context,"profile removed from favourites
successfully", Toast.LENGTH_SHORT).show();
                                }
                            }
                        }
                    });

                    @Override
                    public void onCancelled(@NonNull DatabaseError databaseError) {
                        }
                });
            }
        });

        builder.setNegativeButton("No", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {

```

```

        }
    });
    builder.create().show();
    //
holder.favourite.setBackgroundResource(R.drawable.ic_favorite_borde
r_black_24dp);
    }
}
@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
}
});
// String id=mDatabase.push().getKey();
//mDatabase.child(phonenummer).child(id).child("mobile").setValue(u
p.get(position).getCellno());
}
@Override
public void onCancelled(@NonNull DatabaseError
databaseError) {
}
});
}
});*/
}

@Override
public int getItemCount() {
    return emp.size();
}
public static class myViewHolder extends
RecyclerView.ViewHolder {
    TextView
name_job,qualification_job,date_job,number_job,address_job,city,pho
nenummer,description;
    Button apply;
    //ImageView job_image;
    public myViewHolder(@NonNull View itemView) {
        super(itemView);
        name_job=(TextView)itemView.findViewById(R.id.name_job);
        qualification_job=(TextView)itemView.findViewById(R.id.qualificatio
n_job);
        date_job=(TextView)itemView.findViewById(R.id.date_job);
        number_job=(TextView)itemView.findViewById(R.id.number_job);
        address_job=(TextView)itemView.findViewById(R.id.address_job);
        city=(TextView)itemView.findViewById(R.id.city_job_item);
        phonenummer=(TextView)itemView.findViewById(R.id.phonenummer_job);
        description=(TextView)itemView.findViewById(R.id.description_job_it
em);
        //job_image=(ImageView)itemView.findViewById(R.id.job_image);
        //favour=(Button)itemView.findViewById(R.id.favour);
        apply=(Button)itemView.findViewById(R.id.apply_job);
    }
}

```

```

    }
}

}

//Work_Portal.java
package com.example.jobhunt;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class Work_Portal extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_work_portal);
    }
}

```

//res/layout/activity_employ_portal.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scrollbars="vertical"
    tools:context=".Employportal">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:background="#ffffff">
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical">
                <TextView
                    android:layout_width="300dp"
                    android:layout_height="wrap_content"
                    android:text="Job Description "
                    android:gravity="center"
                    android:textColor="#000000"
                    android:layout_marginTop="10dp"
                    android:layout_marginBottom="50dp"
                    android:textSize="30sp"
                    android:layout_gravity="center"/>
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
</ScrollView>

```



```

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_gravity="center">
    <TextView
        android:id="@+id/t2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name of the company/Office"
        android:textColor="#000000"
        android:textSize="25sp"
        android:layout_marginLeft="10dp"/>
    <EditText
        android:textSize="23sp"
        android:id="@+id/q1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:background="@drawable/border1"
        android:layout_marginLeft="10dp"
        android:layout_marginRight="10dp"
        android:layout_marginBottom="5dp"/>
    </LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView
        android:id="@+id/t3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Qualification Required"
        android:textColor="#000000"
        android:textSize="25sp"
        android:layout_marginLeft="10dp"/>
    <EditText
        android:textSize="23sp"
        android:id="@+id/q2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:background="@drawable/border1"
        android:layout_marginLeft="10dp"
        android:layout_marginRight="10dp"
        android:layout_marginBottom="5dp"/>
    </LinearLayout>
</LinearLayout>

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <TextView
            android:id="@+id/tvEmail"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Address"
            android:textColor="#000000"
            android:textSize="25sp"
            android:layout_marginLeft="10dp"/>
        <EditText
            android:textSize="23sp"
            android:id="@+id/q3"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPersonName"
            android:background="@drawable/border1"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginBottom="5dp"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <TextView
            android:id="@+id/city"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="City"
            android:textSize="25sp"
            android:textColor="#000000"
            android:layout_marginLeft="10dp"/>
        <EditText
            android:textSize="23sp"
            android:id="@+id/city_job"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPersonName"
            android:background="@drawable/border1"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginBottom="5dp"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

```

```

        <TextView
            android:id="@+id/t7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="No of Candidates required"
            android:textColor="#000000"
            android:textSize="25sp"
            android:layout_marginLeft="10dp"/>
        <EditText
            android:textSize="23sp"
            android:id="@+id/q5"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="number"
            android:background="@drawable/border1"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginBottom="5dp"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <TextView
            android:id="@+id/t1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Description"
            android:textSize="25sp"
            android:textColor="#000000"
            android:layout_marginLeft="10dp"/>
        <EditText
            android:textSize="23sp"
            android:id="@+id/q4"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPersonName"
            android:background="@drawable/border1"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginBottom="5dp"/>
    </LinearLayout>
</LinearLayout>

```

//res/layout/activity_home.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"

```

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:layout_weight="2"
        tools:context=".Home">
        <ImageView
            android:id="@+id/work_hire"
            android:layout_gravity="center"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:src="@drawable/hire"
            android:layout_weight="1"/>
        <ImageView
            android:layout_weight="1"
            android:id="@+id/work_employee"
            android:layout_gravity="center"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:src="@drawable/employee"
            android:layout_marginBottom="100dp"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:background="#ffffff"
        android:layout_weight="4"
        android:layout_marginTop="5dp">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_weight="1">
            <TextView
                android:id="@+id/date_text"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="date"
                android:layout_weight="1"
                android:textSize="25sp"
                android:gravity="center"
                android:layout_gravity="end|bottom"
                android:textColor="#000000" />
            </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_weight="1">
            <Button
                android:id="@+id/Timepicker"

```

```

        android:layout_width="wrap_content"
        android:layout_height="60dp"
        android:layout_gravity="center"
        android:layout_weight="1"
        android:layout_marginRight="10dp"
        android:layout_marginLeft="10dp"
        android:text="time picker" />
    </LinearLayout>
    <Button
        android:id="@+id/b1"
        android:layout_marginTop="30dp"
        android:layout_width="220dp"
        android:layout_height="60dp"
        android:text="submit"
        android:textColor="#ffffff"
        android:layout_gravity="center"/>
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:orientation="horizontal"
    android:layout_weight="1"
    android:background="#fbf7ea">
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:orientation="horizontal"
    android:layout_weight="1">
</LinearLayout>
</LinearLayout>
</ScrollView>

```

//res/layout/activity_job_existing.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".Job_existing">
    <Button
        android:id="@+id/job_response"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Responses"/>
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycler_jobexisting"
        android:scrollbars="vertical"
        android:background="#ffffff"

```

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
//res/layout/activity_job_list.xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:id="@+id/user_drawer"
    android:orientation="vertical"
    android:layout_height="match_parent"
    tools:context=".JobList">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
<LinearLayout
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <EditText
        android:textSize="23sp"
        android:gravity="center"
        android:layout_weight="1.5"
        android:id="@+id/search"
        android:hint="Enter The City"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <Button
        android:layout_weight=".3"
        android:background="@drawable/search"
        android:id="@+id/search_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycler_job"
        android:scrollbars="vertical"
        android:divider="#000000"
        android:dividerHeight="4px"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
<com.google.android.material.navigation.NavigationView
    android:id="@+id/matrimony_navigation"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="start"
    app:headerLayout="@layout/check">

```

```
        </com.google.android.material.navigation.NavigationView>
    </androidx.drawerlayout.widget.DrawerLayout>
```

//res/layout/activity_job_response.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".JobResponses">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/jobResponserecyclerview"
        android:layout_width="match_parent"
        android:scrollbars="vertical"
        android:layout_height="match_parent"/>
</LinearLayout>
```

//res/layout/activity_login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@drawable/profile"
    tools:context=".login">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_marginTop="210dp">
        </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:layout_weight="1"
            android:layout_gravity="center">
            <EditText
                android:maxLength="10"
                android:id="@+id/email"
```

```

        android:textSize="23sp"
        android:inputType="phone"
        android:textColor="#000000"
        android:layout_marginTop="15dp"
        android:layout_gravity="center"
        android:layout_width="270dp"
        android:gravity="center"
        android:layout_height="wrap_content"
        android:hint="Enter the Mobile Number"/>
<EditText
    android:textSize="23sp"
    android:id="@+id/password"
    android:inputType="textPassword"
    android:layout_marginTop="15dp"
    android:layout_gravity="center"
    android:layout_width="270dp"
    android:gravity="center"
    android:layout_height="wrap_content"
    android:hint="Enter the password"/>
<Button
    android:textSize="20sp"
    android:id="@+id/login_auth"
    android:layout_width="270dp"
    android:layout_marginTop="30dp"
    android:background="#ffffff"
    android:textColor="#000000"
    android:layout_height="60dp"
    android:layout_gravity="center"
    android:text="log in"/>
<Button
    android:textSize="20sp"
    android:id="@+id/verify"
    android:layout_width="270dp"
    android:layout_marginTop="20dp"
    android:background="#ffffff"
    android:textColor="#000000"
    android:layout_height="60dp"
    android:layout_gravity="center"
    android:text="signup"/>

```

```

</LinearLayout>

```

```

</LinearLayout>

```

```

</LinearLayout>

```

//res/layout/activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>

```

```

<LinearLayout

```

```

    xmlns:android="http://schemas.android.com/apk/res/android"

```

```

    xmlns:app="http://schemas.android.com/apk/res-auto"

```

```

    xmlns:tools="http://schemas.android.com/tools"

```

```

    android:layout_width="match_parent"

```

```

    android:orientation="vertical"

```

```

    android:layout_height="match_parent"

```



```

        android:background="#ffffff"
        android:weightSum="2"
        tools:context=".MainActivity">
<LinearLayout
    android:background="@drawable/prof"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
<ImageView
    android:id="@+id/top"
    android:layout_weight="1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:src="@mipmap/well_foreground"
    android:layout_marginTop="100dp"
    android:layout_gravity="bottom"/>
<ImageView
    android:id="@+id/bottom"
    android:layout_marginTop="-258dp"
    android:layout_gravity="center"
    android:layout_weight="1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:src="@mipmap/job_foreground"/>
</LinearLayout>
</LinearLayout>

```

//res/layout/activity_work_portal.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Work_Portal">
</androidx.constraintlayout.widget.ConstraintLayout>

```

//res/layout/check.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:scrollbars="vertical"
    android:layout_height="match_parent"
    android:background="#ffffff">
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
<LinearLayout
    android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:orientation="vertical">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">
            <TextView
                android:layout_weight="1"
                android:gravity="center"
                android:textColor="#000000"
                android:textSize="20sp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Profile image"/>
            <TextView
                android:gravity="center"
                android:layout_weight="1"
                android:textColor="#000000"
                android:textSize="20sp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="resume image"/>
        </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">
            <ImageView
                android:id="@+id/profileimage_drawer_matrimony"
                android:layout_width="100dp"
                android:layout_height="300dp"
                android:layout_weight="1"
                android:layout_gravity="center"/>
            <ImageView
                android:id="@+id/resumeimage_drawer_matrimony"
                android:layout_width="100dp"
                android:layout_height="300dp"
                android:layout_weight="1"
                android:layout_gravity="center"/>
        </LinearLayout>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <LinearLayout
            android:layout_marginTop="10dp"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_marginLeft="20dp"

```

```

        android:layout_gravity="end">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="Name:"
            android:textColor="#000000"
            android:textSize="20sp"
            android:gravity="start" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Age:"
            android:layout_gravity="start"
            android:textColor="#000000"
            android:textSize="20sp" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="Sex:"
            android:textColor="#000000"
            android:textSize="20sp" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="Degree:"
            android:textColor="#000000"
            android:textSize="20sp" />
    </LinearLayout>
    <LinearLayout
        android:layout_marginTop="10dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_marginLeft="20dp"
        android:layout_gravity="end">
        <TextView
            android:id="@+id/drawer_name_matrimony"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="name"
            android:textColor="#000000"
            android:textSize="20sp"
            android:gravity="center_horizontal" />
        <TextView
            android:id="@+id/drawer_age_matrimony"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"

```

```

        android:layout_gravity="start"
        android:text="age"
        android:textColor="#000000"
        android:textSize="20sp" />
<TextView
    android:id="@+id/drawer_sex_matrimony"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="start"
    android:text="sex"
    android:textColor="#000000"
    android:textSize="20sp" />
<TextView
    android:id="@+id/drawer_degree_matrimony"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="start"
    android:text="degree"
    android:textColor="#000000"
    android:textSize="20sp" />
</LinearLayout>
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <LinearLayout
        android:layout_marginTop="10dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_marginLeft="20dp"
        android:layout_gravity="end">
        <TextView
            android:id="@+id/branch_name_matrimony"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="name"
            android:textColor="#000000"
            android:textSize="20sp"
            android:gravity="center_horizontal" />
        <TextView
            android:id="@+id/drawer_locaton_matrimony"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="age"
            android:textColor="#000000"
            android:textSize="20sp" />
    <TextView

```

```

        android:id="@+id/drawer_address_matrimony"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start"
        android:text="sex"
        android:textColor="#000000"
        android:textSize="20sp" />
    <TextView
        android:id="@+id/drawer_city_matrimony"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start"
        android:text="height"
        android:textColor="#000000"
        android:textSize="20sp" />
    </LinearLayout>
</LinearLayout>
<LinearLayout
    android:layout_marginTop="10dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_marginLeft="20dp"
    android:layout_gravity="end"
    >
    <TextView
        android:id="@+id/branch_cgpa_matrimony"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start"
        android:text="name"
        android:textColor="#000000"
        android:textSize="20sp"
        android:gravity="center_horizontal" />
    <TextView
        android:id="@+id/drawer_college_matrimony"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start"
        android:text="age"
        android:textColor="#000000"
        android:textSize="20sp" />
    </LinearLayout>
</LinearLayout>
</LinearLayout>
</ScrollView>

```

//res/layout/job_item.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout

```

```

xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:padding="20dp"
    android:background="#ffffff"
    android:layout_height="wrap_content">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:background="@drawable/border1">
        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content">
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_weight="8"
                android:orientation="horizontal">
                <LinearLayout
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_weight="6"
                    android:orientation="vertical">
                    <LinearLayout
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
                        android:layout_weight="1"
                        android:orientation="vertical">
                        <TextView
                            android:id="@+id/name_job"
                            android:layout_width="match_parent"
                            android:layout_height="wrap_content"
                            android:layout_marginLeft="30dp"
                            android:layout_marginRight="30dp"
                            android:text="TextView"
                            android:textAlignment="center"
                            android:textColor="#ff8000"
                            android:textSize="28dp" />
                        </LinearLayout>
                    <LinearLayout
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
                        android:orientation="vertical"
                        android:layout_weight="1">
                        <TextView
                            android:id="@+id/address_job"
                            android:layout_width="match_parent"
                            android:layout_height="wrap_content"
                            android:drawableLeft="@drawable/location_vector"
                            android:drawablePadding="2dp"

```

```

        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:text="TextView"
        android:textColor="#000000"
        android:textSize="23dp" />
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_weight="1">
        <TextView
            android:id="@+id/city_job_item"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
android:drawableLeft="@drawable/city_vector"
            android:drawablePadding="2dp"
            android:layout_marginLeft="30dp"
            android:layout_marginRight="30dp"
            android:text="city"
            android:textColor="#000000"
            android:textSize="23dp" />
        </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_weight="1">
            <TextView
                android:id="@+id/description_job_item"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
android:drawableLeft="@drawable/description_vector"
                android:drawablePadding="2dp"
                android:textAlignment="center"
                android:layout_marginLeft="30dp"
                android:layout_marginRight="30dp"
                android:text="city"
                android:textColor="#000000"
                android:textSize="23dp" />
            </LinearLayout>
            <Button
                android:id="@+id/apply_job"
                android:layout_width="match_parent"
android:layout_height="wrap_content"
                android:text="apply"/>
        </LinearLayout>
    </LinearLayout>
</androidx.cardview.widget.CardView>
</LinearLayout>
</LinearLayout>

```

//res/anim/frombottom.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate android:duration="3900"
        android:fromYDelta="100%p"
        android:fromXDelta="0%p"/>
</set>
```

//res/anim/frombottom1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate android:duration="3500"
        android:fromYDelta="100%p"
        android:fromXDelta="0%p"/>
</set>
```

//res/menu/menu_actionbar.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/logout"
        android:icon="@drawable/logout"
        app:showAsAction="always"
        android:title="logout"/>
</menu>
```

//res/menu/menu_logandhome.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/home1"
        android:icon="@drawable/home"
        app:showAsAction="always"
        android:title="home"/>
    <item
        android:id="@+id/logout"
        android:icon="@drawable/logout"
        app:showAsAction="always"
        android:title="logout"/>
</menu>
```


6. FUTURE ENHANCEMENT

This project fulfills the primary requirements of the job seekers and employers. It can be extended in several ways – We can provide recommendations and email updates for new job postings based on the job seeker's search history. Since, the job seekers might be interested in building a strong Resume, we can provide tips and information for the same. We can also provide templates for building the Resumes which might interest most applicants. The mobile application is developed fulfilling the functionalities of job seeker, it can be extended to support functionalities of Employer as well.

7. CONCLUSION

Job Hunt stands as a revolutionizing element in the sphere of recruitment. They act as a communication bridge between applicants and recruiters facilitating their requirements. This application helps organizations to have a greater exposure to the candidate pool and also the job seekers facilitating wide search of jobs matching their interests. The android application provides flexibility to the jobseekers to view the openings and applied jobs without the need to carry a laptop. This application provides an enhanced user experience for both employer and jobseeker. It provides user friendly interface which facilitates in reaching wide range of audience. The application has achieved all the requirements that were initially set in the requirements gathering phase. Starting from requirements elicitation to design, construction, implementation and testing, I have gained a very good experience working with various technologies at every phase. Development of this project boosted my confidence in

mobile and web development.

8. BIBLIOGRAPHY

(1) “An Overview of The Android Architecture”

http://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture

(2) “Android Runtime”

https://en.wikipedia.org/wiki/Android_Runtime

(3) **<https://www.tutorialspoint.com>**

(4) **<https://www.javatpoint.com>**

(5) **<https://www.w3schools.com>**