Mini Project On

Roommate Finder Application

By

Shubham Patil (2021510049)
Devvrat Suktel (2021510065)

Under the guidance of

Internal Supervisor

Prof. Harshil Kanakia

Department of Master Of Computer Application
Sardar Patel Institute of Technology
Autonomous Institute Affiliated to Mumbai University

2020-21

CERTIFICATE OF APPROVAL

This is to certify that the following students

Shubham Patil (2021510049)
Devvrat Suktel (2021510065)

Have satisfactorily carried out work on the project
entitled

" Roomie"

Towards the fulfilment of project, as laid down
by
Sardar Patel Institute of Technology
during year
2021-22.

Project Guide:
Prof. Harshil Kanakia

PROJECT APPROVAL CERTIFICATE

This is to certify that the following students

Shubham Patil (2021510049)
Devvrat Suktel (2021510065)

Have successfully completed the Project report on

"Roomie ",

which is found to be satisfactory and is approved

at

SARDAR PATEL INSTITUTE OF TECHNOLOGY,
ANDHERI (W), MUMBAI

INTERNAL EXAMINER EXTERNAL EXAMINER

HEAD OF DEPARTMENT PRINCIPAL

Contents

Abstract

The world of Internet is growing rapidly, many applications that previously created on the desktop start moving to the web. Many applications could be accessed anytime and anywhere easily using Internet.People looking for a place to stay face lot of problems roomie helps to solve these issues.

Roomie is a mobile application that helps people easily find a perfect room/ roommate either in their new location or current location. Most of the people around the world are choosing to live together sharing a place with some trusted person can be a brilliant living experience and a great way to save money so we can all live better! Roomie is making the experience of sharing a home easy - safe.

Objectives

The Web based Application "Recode" is used

- To provide Users a user friendly platform for users to find ideal roommate .
- To provide a convenient communication between roomies - rentee.
- To provide a facility to view ,Edit Post Rooms in the application
- To provide verified room and users information that is trustworthy.

List of Figures

List of Tables

1 Introduction

1.1 Problem Definition

A large percentage of youth are constantly migrating from small towns to big cities for education and employment. In most cases, they are faced with the challenges such as high rent, security deposit and deal with landlords who are still averse to rent out the apartments to bachelors. Therefore, the shared accommodation has been the alternate for people faced with such difficulties with renting an apartment of their own. But finding a proper room/roommate is a hassle and having the wrong roommates can be a nightmare. After doing the user interview, some paints and frustrations were noted.
Those are:

- Hard to find someone having similar religion lifestyle.
- communication issues to arrange a meeting.
- Difficult to find a budget friendly room.
- fake images and listing with fake details.
- Hard to find people that are trustworthy.

1.2 Objectives and Scope

1.2.1 Objectives

The Web based application "Roomie" is
- To provide Users a user friendly platform for finding Rooms Roommates.
- To provide a convenient communication between roommies-rentees
- To provide a facility to view ,Edit n Post Room Details in the application.
- To provide verified room and users information that is trustworthy.

1.2.2 Scope

The Users can view ,post rooms and contact rentees.

In the application users will create their own Account post details if they have a room and other users can view these rooms and contact if possible match is found.

Our System leverages lamp stack technology for real-time updation within the app for intuitive and user friendly interface.

1.3 Existing System

Most of the existing applications have no authentic information available. Take a lot of time to find a perfect roommate, Agencies charge a lot of money to confirm room.

Some of the disadvantages of existing system are as follows :

- No contact details of landlords and roommate,
- most landlords don't want to rent flats to bachelors,
- No ads available for budget friendly rooms.

1.4 Proposed System

1.5 expectation

- Need a shared room to decrease cost.
- roommate having similar lifestyle.
- finding a roommate in a very short time.
- no authentic information available
- Trustworthy and verified users

1.6 Features

- Well planned filter to refine the best results.
- Map or geolocation tracking system.
- Post ads option
- message and video call option.
- organized listing.

Some of the advantages of our system are as follows :

- User Friendly
  Roomie will provide a clean user friendly roommate finding interface.
  user will be able to see who has authentic room

- trustworthy features
  Users can view ,edit , post their rooms in the application with prooper verification.
- Easy Access
  user will be able to access someone's room and start contact just by a single click of a button.

1.7 System Requirements

- Hardware Requirements on Server Side

Table 1.5.1: Hardware Requirements on Server Side

Processor	Dual Core Processor or Above
RAM	Minimum 4 GB RAM
Storage	Minimum 10 GB Hard Disk Space for smooth run

- Hardware Requirements on Client Side

Table 1.5.2: Hardware Requirements on Client Side

Device	Android Device with Touch Screen minimum 5” inch Display
Processor	Dual Core Processor or Above
RAM	Minimum 2 GB RAM
Storage	Minimum 250 MB Storage Space

- Software Requirements on Server Side

Table 1.5.3: Software Requirements on Server Side

Operating System	OS Independent
Database	browser cache

- Software Requirements on Client Side

Table 1.5.3: Software Requirements on Client Side

Operating System	Android/IOS Smartphone
Server	Not Required

2 Software Requirement Specification (SRS) and Design

2.1 Purpose

The purpose of our project is to develop an UI application that can help user collaborate with other users to write code within a single file.
This can save lots of time setting up your own development environment. inviting and sharing code with others becomes a tedious task. recode eliminates this hassle and directly unables one to code.

2.2 Definition

To build a Recode Application so the user can have an easy code collaboration among them.

2.3 Overall Description

2.3.1 Product Functions

The product function includes:

1. Authentication: Users are required to enter room ID and create a room which itself creates a room ID automatically
2. Profile: This will contain information regarding the user
3. Room: This will contain information regarding room and room ID marks.
4. contact roommate: contact other users and view thier contact details

2.4 references

IEEE standard -830 -1998, Software Engineering.

2.5 document overview

This document will be used for design purpose by the developer and design team. It will be the basis for validating the final delivered system.

2.6 Intended audience

This document will be used for design purpose by the developer and design team. It will be the basis for validating the final delivered system.

2.7 SRS team members

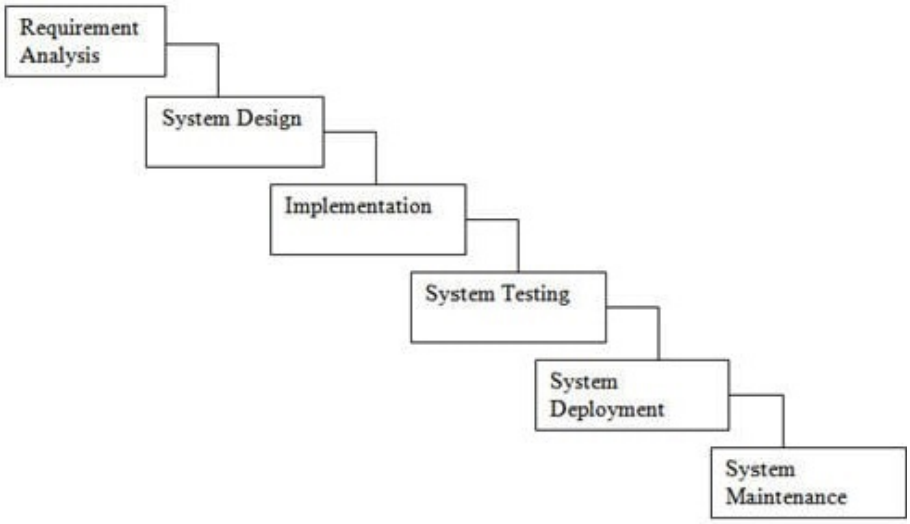
The document is written by
Shubham Patil(2021510049) and devvrat suktel(2021510065)

3 Project Analysis and Design

3.1 Methodologies Adapted

In Waterfall model, very less customer interaction is involved during the development of the product. Once the product is ready then only it can be demonstrated to the end users.

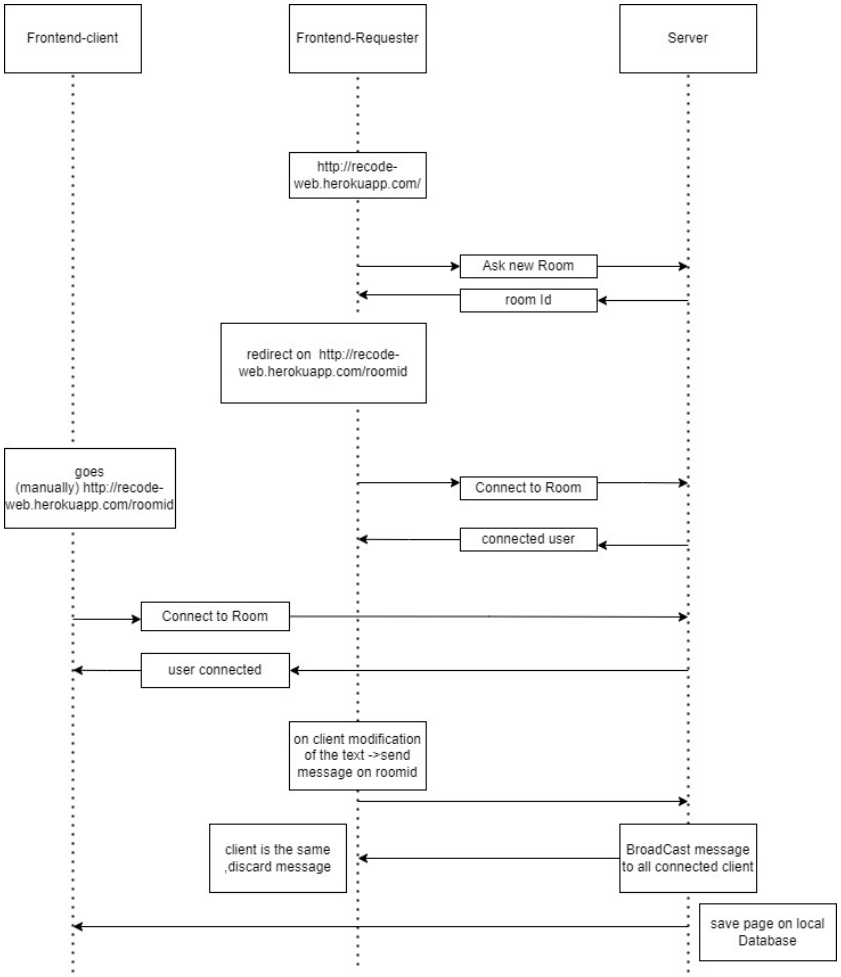
Once the product is developed and if any failure occurs then the cost of such issues is very high, because we need to update everything from document till the logic.



3.1.1: Diagrammatic Representation of Waterfall Model

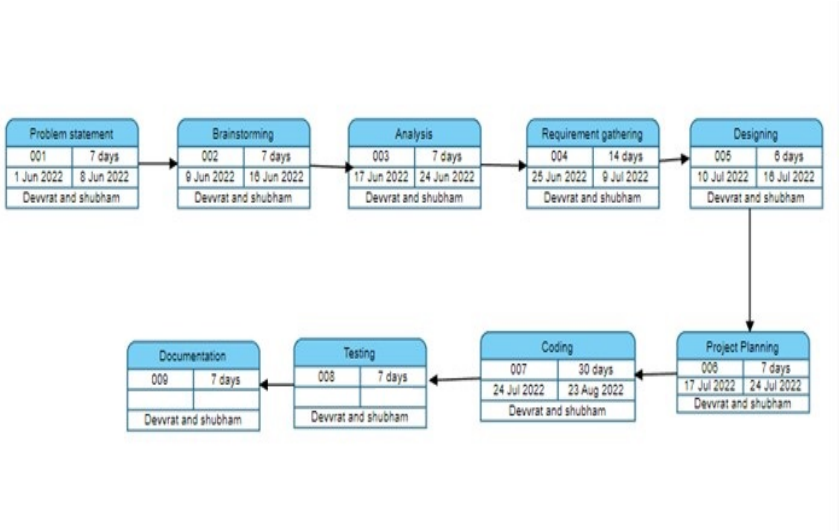
3.2 Modules

3.2.1 Sequence Diagram



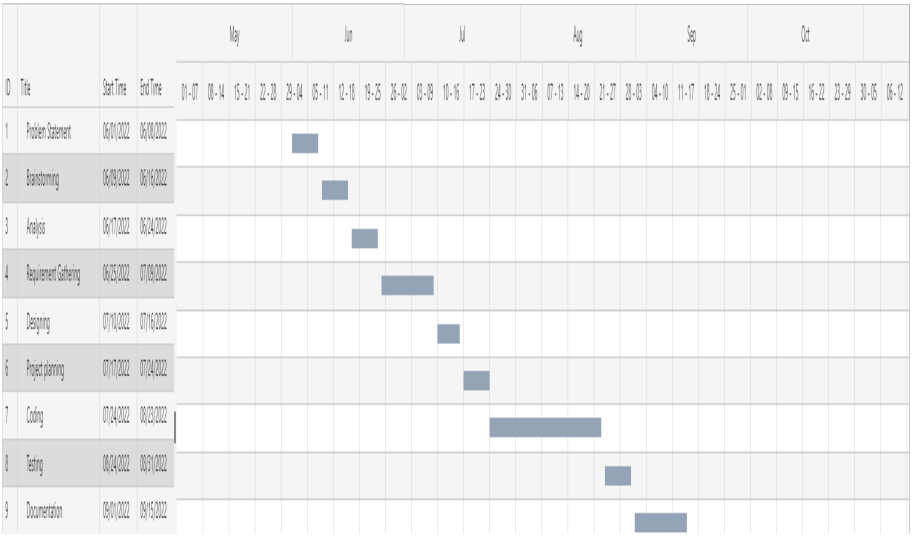
3.2.1: Sequence Diagram

3.2.2 Pert Chart



3.2.2: Pert Chart

3.2.3 Gantt Chart



3.2.3: Gantt Chart

3.3 Use cases

Use Cases:

1. Create Room
2. View Room
3. Delete Room
4. contact user

Table 4.2.1: Use Case Table - create room

Use Case ID	1
Use Case Name	Create room
Actor	User
Pre-Condition	They should have an active internet connection
Post-Condition	User can enter room
Flow of events	create room ,Post details.

Table 4.2.2: Use Case Table - View room

Use Case ID	2
Use Case Name	enter Room
Actor	user
Pre-Condition	They must have a room ID to view a room
Post-Condition	User can enter the room and view code.
Flow of events	create room,view Room ,make changes,post again

Table 4.2.3: Use Case Table - deleter room

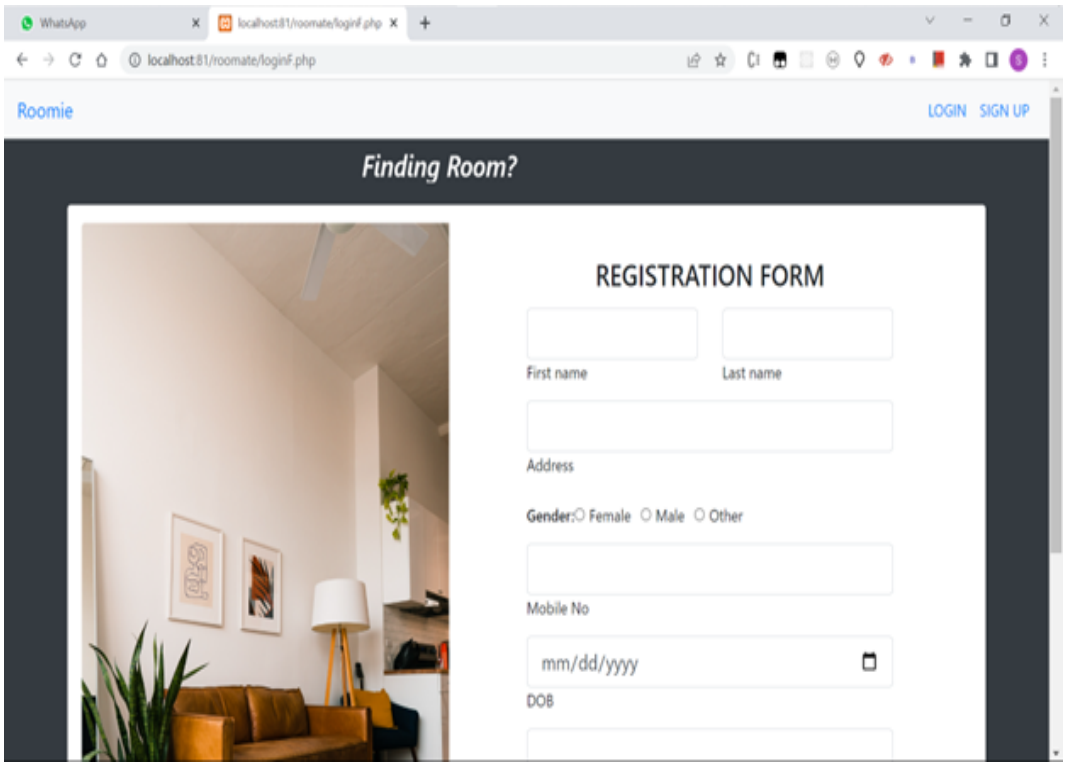
Use Case ID	3
Use Case Name	delete room
Actor	User-1,user-2
Pre-Condition	room member
Post-Condition	User can view or delete the room

Table 4.2.4: Use Case Table - contact user

Use Case ID	4
Use Case Name	contact user
Actor	User-1, User-2
Pre-Condition	Login
Post-Condition	User can view and contact other users from the details

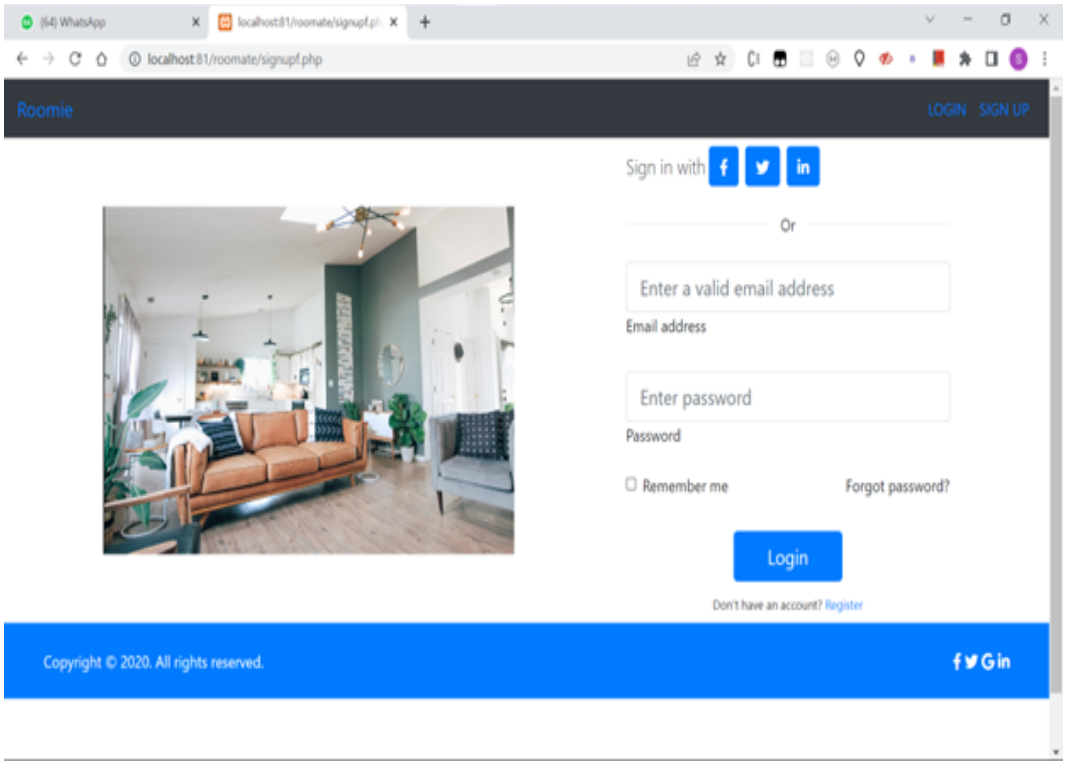
4 Project Implementation and Testing

4.1 register user



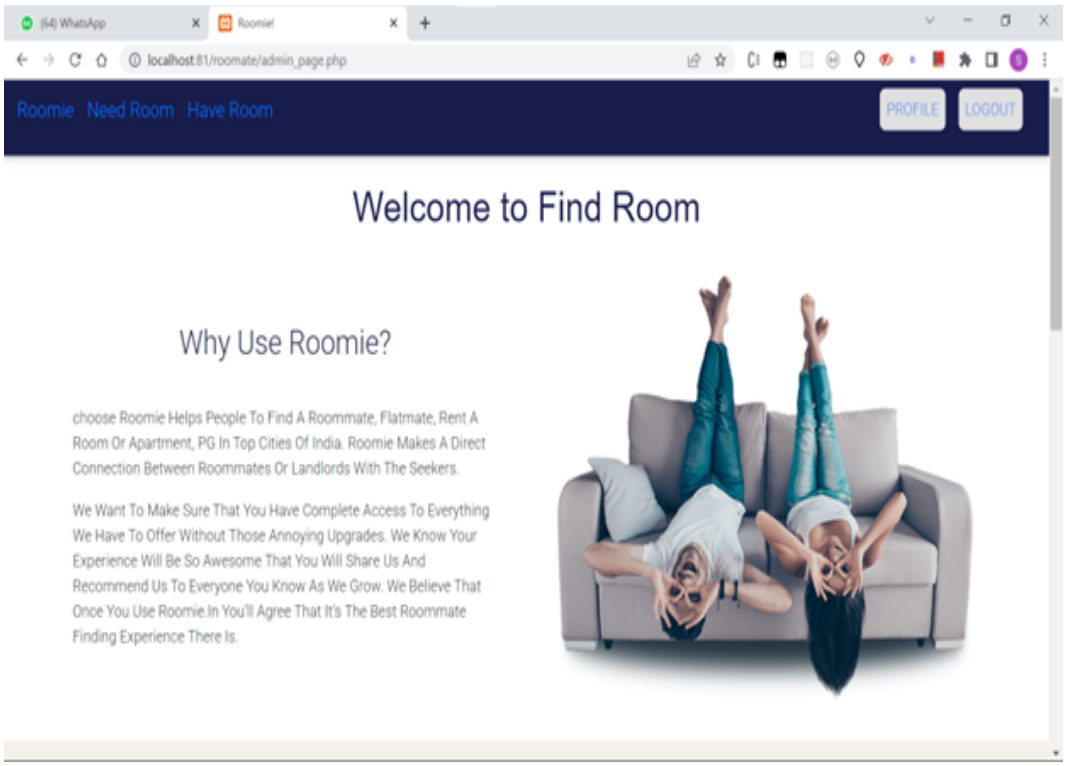
4.1.1: registration form

4.2 login page



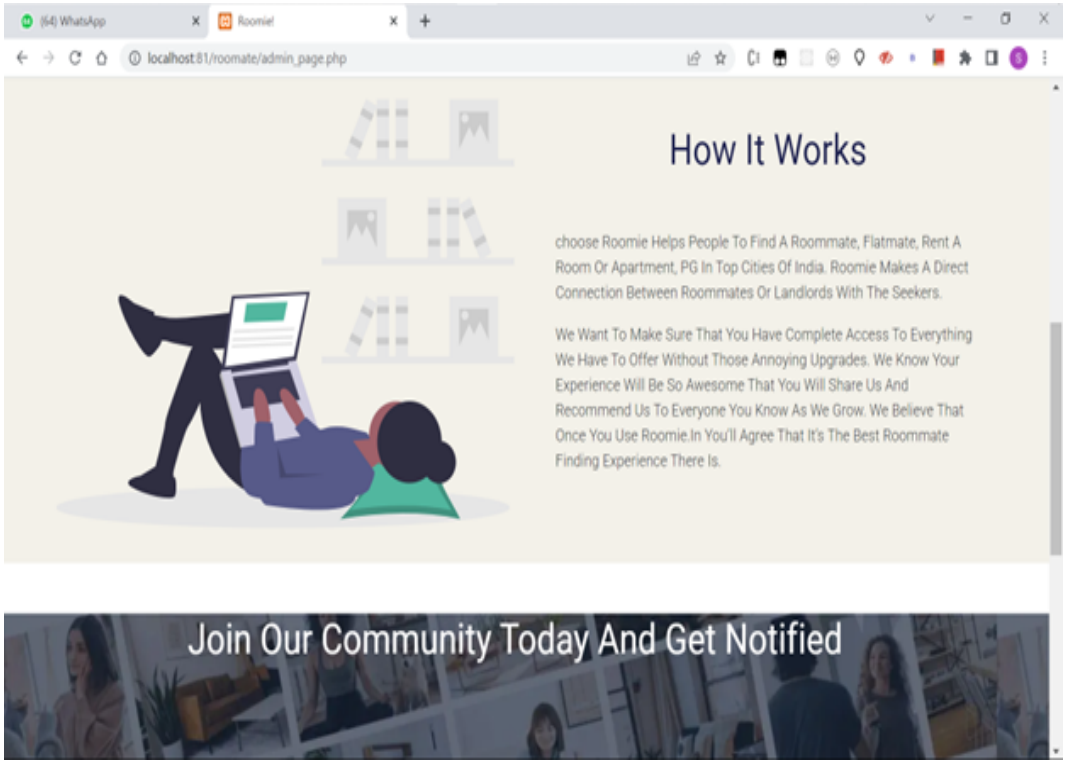
4.2.1: join roomie

4.3 Home



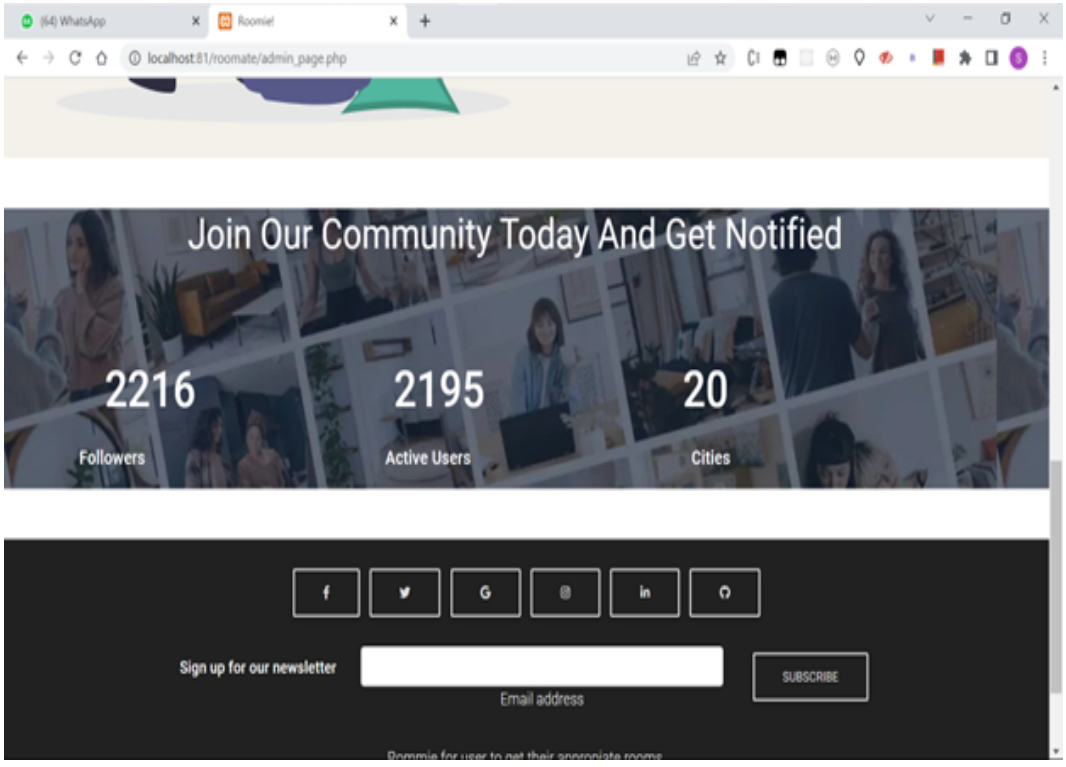
4.3.1: Home

4.4 how it works



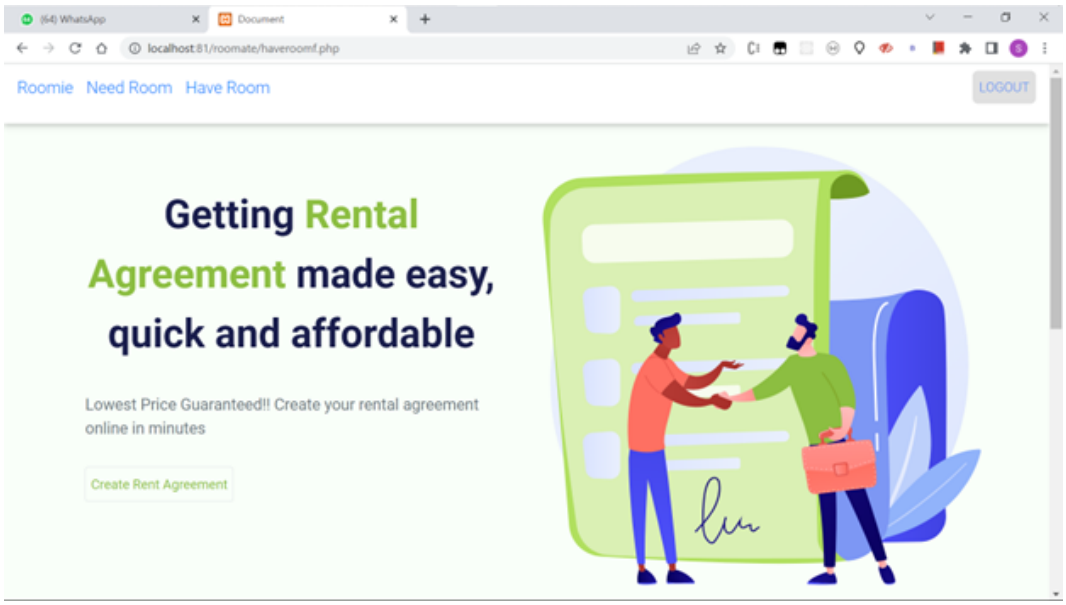
4.4.1: how it works View

4.5 full website



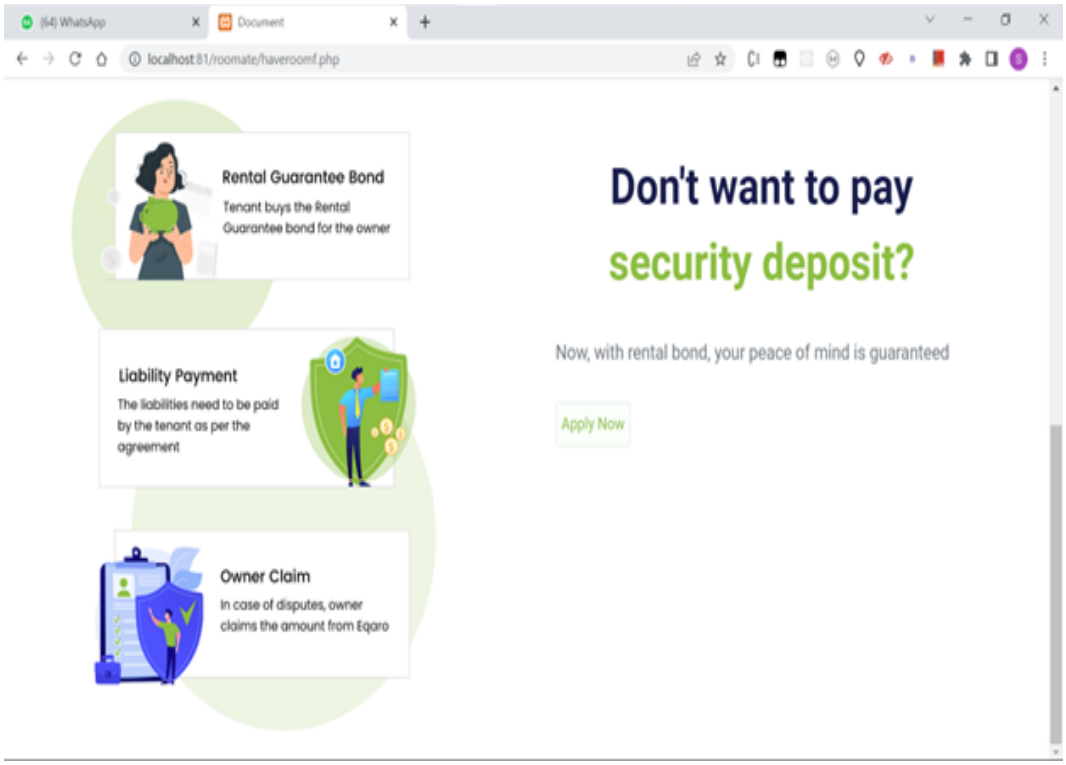
4.5.1: website

4.6 have a room



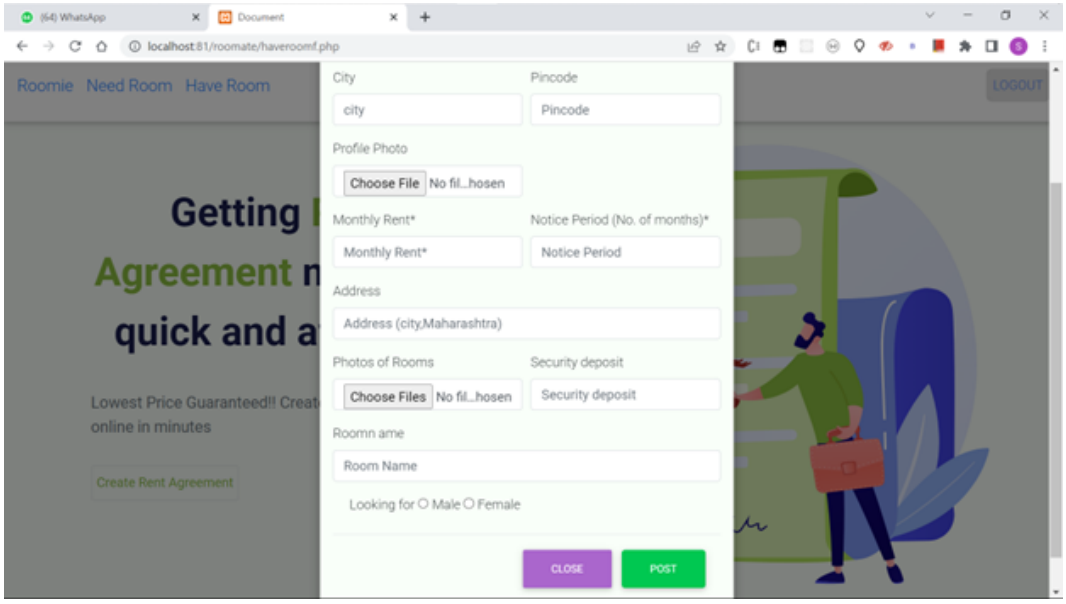
4.6.1: have room View

4.7 have a room page



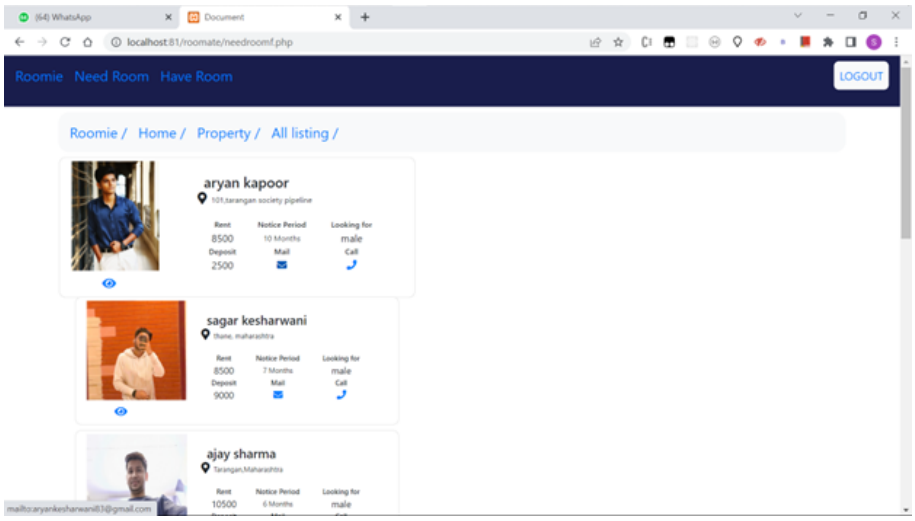
4.7.1: have room page

4.8 room details form

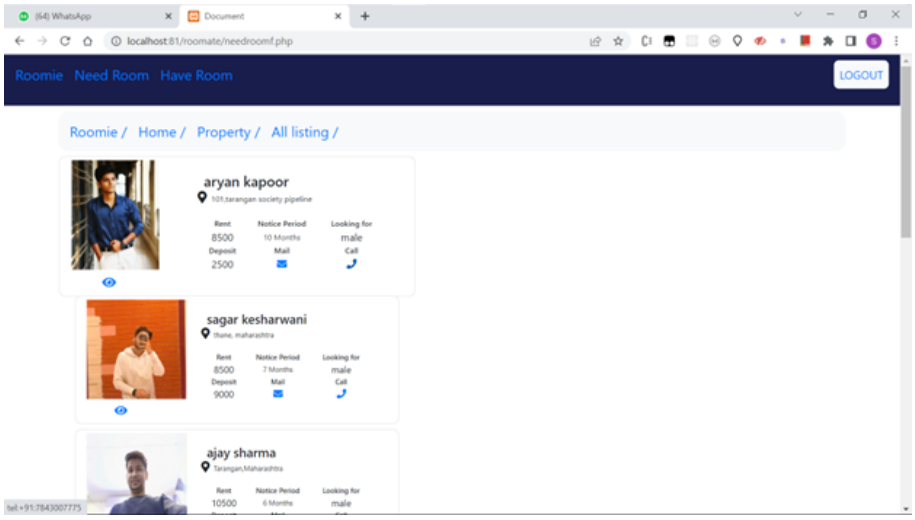


4.8.1: room details

4.9 rooms dashboard

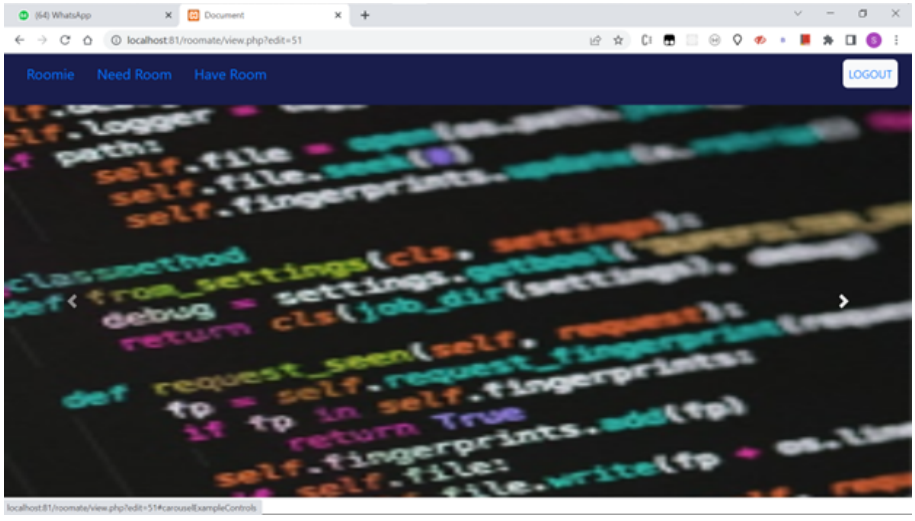


4.10 dashboard

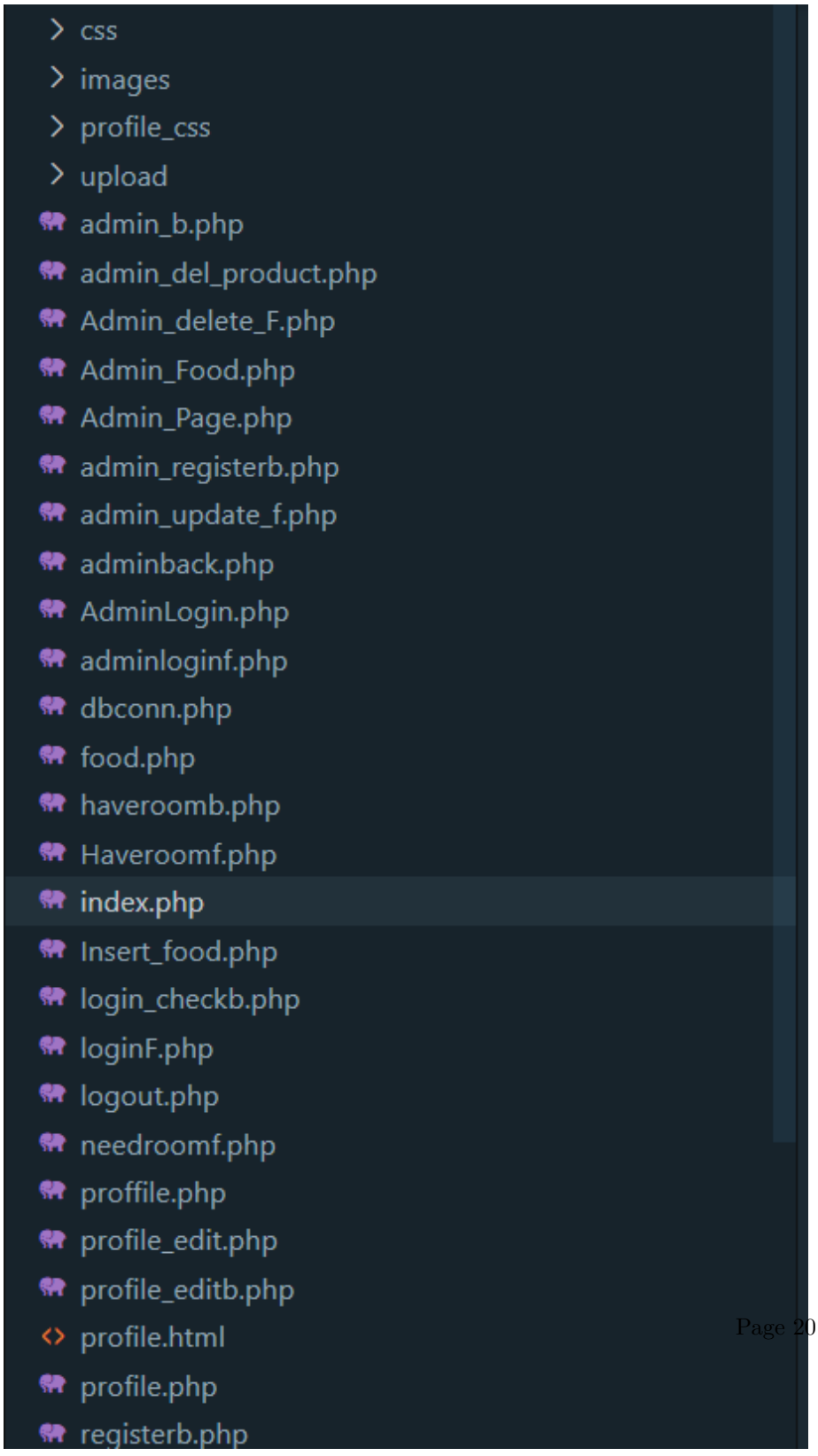


4.11 room pictures

4.12 room pictures



4.13 Code


css
images
profile_css
upload
admin_b.php
admin_del_product.php
Admin_delete_F.php
Admin_Food.php
Admin_Page.php
admin_registerb.php
admin_update_f.php
adminback.php
AdminLogin.php
adminloginf.php
dbconn.php
food.php
haveroomb.php
Haveroomf.php
index.php
Insert_food.php
login_checkb.php
loginF.php
logout.php
needroomf.php
proffile.php
profile_edit.php
profile_editb.php
profile.html
profile.php
registerb.php

4.14 Code

index.php

```php
<?php
session_start();
  if(!isset($_SESSION["username"])){
      header("Location:loginF.php");
  }
?>

<!DOCTYPE html>
<html>



<head>
  <titile> </titile>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet" href="css/style.css">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css">
  <!-- Google Fonts -->
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap">
  <!-- Bootstrap core CSS -->
  <link  rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.5.0/css/bootstrap.min.css" >
  <!-- Material Design Bootstrap -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/mdbootstrap/4.19.1/css/mdb.min.css" >




</head>

<body>

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
```

4.15 Code

```php

<nav class="navbar navbar-expand-lg navbar-red-green bg-dark">
    <a class="navbar-brand bg-white px-4 py-2"  style="border-radius: 7px;" href="index.php">PETIFY </a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav ml-auto">


        <li class="nav-item active">
          <a class="nav-link" href="index.php">HOME <span class="sr-only">(current)</span></a>
        </li>
        <div class="dropdown ">
        <a class="nav-link dropdown-toggle" href="PAGES.HTML" id="navbarDropdown" role="button" data-toggle="dropdown"
            aria-haspopup="true" aria-expanded="false">
            PAGES
          </a>
            <ul class="dropdown-menu multi-level bg-dark" role="menu" aria-labelledby="dropdownMenu">

                <li class="dropdown-submenu bg-dark">
                <a class="dropdown-item bg-light" href="food.php"> Foods </a>
                  </li>

                </li>
```

4.16 Code

loginF.php

```php
<?php
Session_start();

?>
<!DOCTYPE html>
<html>

<head>
    <titile> </titile>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <!-- <link rel="stylesheet" href="images/style_LOGIN.css"> -->
    <style>
    .card-registration .select-input.form-control[readonly]:not([disabled]) {
        font-size: 1rem;
        line-height: 2.15;
        padding-left: .75em;
        padding-right: .75em;
    }

    .card-registration .select-arrow {
        top: 13px;
    }
    </style>
</head>


<body>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
    </script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <nav class="navbar navbar-expand-lg navbar-red-green bg-light">
        <a class="navbar-brand" href="signup.php">PETIFY </a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
            aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
```

4.17 Code

signupF.php

```php
<?php
Session_start();

?>
<!DOCTYPE html>
<html>

<head>
    <titile> </titile>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.0/css/all.min.css"
        integrity="sha512-xh6O/CkQoPOWDdYTDqeRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSCcJHgXloTyT2A=="
        crossorigin="anonymous" referrerpolicy="no-referrer" />
    <style>
    .divider:after,
    .divider:before {
        content: "";
        flex: 1;
        height: 1px;
        background: #eee;
    }

    .h-custom {
        height: calc(100% - 73px);
    }

    @media (max-width: 450px) {
        .h-custom {
            height: 100%;
        }
    }
    </style>
    <!-- <link rel="stylesheet" href="images/style.css"> -->
</head>
```

4.18 Test Cases

Test Case ID	Test Case Name	Test Data	Expected Output	Actual Output	Result
1	User clicks crate room	Unique room id is should be generated	Unique room id is generated	Unique room id is generated	pass
2	User gives name and clicks join	Name should be displayed inside the room	Username shown in the name	Username shown in the name	pass
3	User writes in the editor	Able to write in the editor	Types correctly	Types correctly	pass
4	all user can make changes	Change reflected	Change reflected in all users system	Change reflected in all users system	pass
5	Dynamic typing	All can type at same time	All changes are made concurrently	Changes are made one after the other	fail

5 User Manual

Part 1 – post room

create a room to get started with writing about the room to share it with other users

Part 2 – view room

view a already live room using room dashboard

Part 3 – contact

contact landlords room owners

6 Bibliography

6.1 Web References

[1.] https://www.youtube.com/user/Firebase
[2.] https://stackoverflow.com/
[3.] https://www.draw.io/
[4.] https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/
[5.] https://www.geeksforgeeks.org/