**­HOSTEL SEEKING FORUM**

PROJECT REPORT

Submitted by

Anjali Krishna 20419018

Aparna Unnikrishnan 20419024

Ashana 20419026

In partial fulfillment for the award of the degree of Bachelor Of Technology Degree in Information Technology.



**SCHOOL OF ENGINEERING**

**COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY KOCHI-22**

**JUNE 2021**

**COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY SCHOOL OF ENGINEERING**

**COCHIN-22**

****

**CERTIFICATE**

It is certified that this report titled “HOSTEL SEEKING FORUM” is a bonafide record of the mini project work done by Anjali Krishna , Aparna Unnikrishnan , Ashana and is forwarded towards the fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology from Cochin University of Science And Technology during the year 2021.

Mrs. Sahiya Riyas , Dr .Shelbi Joseph Dr. Binsu C. Kovoor

Project Guide Head of the Department

**ACKNOWLEDGEMENT**

The project on the topic “**HOSTEL SEEKING FORUM**” was taken as a part of the curriculum for the award of B. Tech degree in Information Technology Engineering.

We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavour to a successful culmination. We extend our sincere and heartfelt thanks to our esteemed guide, **MRS. SAHIYA RIYAS** , for providing us with the right guidance and advice at the crucial junctures and for showing us the right way. We also take this opportunity to express a deep sense of gratitude to **Mr. MAHESH BABU** and **JIM** from **JimShapedCoding**. We gratefully acknowledge the support extended by Head of Division , **DR. BINSU C KOVOOR** and we are also obliged to the staff members of the division for their cooperation during the period of our project.

Lastly, we would like to thank our friends and family for the support and encouragement they have given us during the course of our work.

**ABSTRACT**

Hostel Seeking Forum is an RDBMS project which is done as a finding tool for students.

The main aim of this project is to help the new comers who join the college and others(hostel owners) to be familiar with a new platform to make an advertisement .Here, they are free to make an account and seek for their hostels. We also provide a platform for hostel owners to post their hostel details , like rent , location , availability of rooms ,curfew, and a small description about their particular hostel on the website .

Students can look up information about the hostels near CUSAT and choose the right one. This will provide a quick access of hostels and save the time.

**CONTENTS**

1. **INTRODUCTION**
   1. **OBJECTIVES**
   2. **BACKGROUND OF PROJECT**
   3. **PROBLEM STATEMENT**
2. **LITERATURE SURVEY**
   1. **EXISTING SYSTEM**
   2. **PROPOSED SYSTEM**
3. **SYSTEM REQUIREMENT SPECIFICATIONS**
   1. **HARDWARE SPECIFICATION**
   2. **SOFTWARE SPECIFICATION**
4. **DESIGN**
   1. **E-R DIAGRAM**
   2. **TABLE DESIGN**
   3. **DATAFLOW DIAGRAMS**
   4. **STUCTURE OF WEBSITE**
5. **SYSTEM IMPLEMENTATION**
   1. **MODULE DESCRIPTION**
   2. **SNAPSHOTS**
6. **RESULT AND DISCUSSION**
7. **CONCLUSION AND FUTURE SCOPE**
8. **REFERENCES**
9. **APPENDIX**

**1.INTRODUCTION**

**1.1 OBJECTIVES**

The main objective of our project is to help students who are in a search of hostels and others(hostel owners) to be familiar with a new platform to make an advertisement .This could be used as a finding tool so that they can definitely save time for their needs.

**1.2 BACKGROUND OF PROJECT**

We come to this project by observing the problems of new comers to our college especially the North Indians ,who are facing difficulties in search of a good hostel. Also in some colleges including CUSAT do not provide the hostel facilities for first year students. So here, they are free to make an account and seek for their hostel details .We also provide a platform for hostel owners to post their hostel details , like rent , location , availability of rooms ,curfew, and a small description about their particular hostel on the website . They can register and login using their username and password for adding details and maintain them for future use.

So for those who are in a quick search of a hostel , this is a finding tool for them. This will save the time with the details of nearest hostels in a finger tip.

* 1. **PROBLEM STATEMENT**

When we join college it is very much important to find a hostel for long distance students as some colleges do not provide the hostel facilities for first year students. Also we can observe that students from different geographical locations have difficulty in finding a proper hostel to get accommodated. That’s why we have come up with a platform which is not only going to help these students but also is very profitable for the hostel owners. This could save their time by signing up to our website and seek the hostel details within a limited time.

1. **LITERATURE SURVEY**

**2.1 EXISTING SYSTEM**

The existing system is manual based and need lot of efforts and consume enough time. In the existing system, we have to enquire and gather the data from our seniors and batchmates. It may lead to wastage of huge amount of time and still not getting any satisfaction.

DISADVANTAGES :

* Time consuming
* Wastage of human power and energy
* Difficulty in gathering data
* Low security
* Difficulty in finding tenants for hostels
* No proper platform to advertise their hostel

**2.2 PROPOSED SYSTEM**

The proposed system is having many advantages over the existing system. It requires less overhead and is very efficient. The proposed system deals with a proper seeking forum for students and for hostel owners, where they can advertise their hostels or PG s.

**3. SYSTEM REQUIREMENT SPECIFICATIONS**

**3.1HARDWARE SPECIFICATION**

The section of hardware configuration is an important task related to the software development insufficient random access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

Processor : Intel Core i3 and above

Processor speed : 1.6 GHz

Cache size : Up to 3.9 GHz

Network connection : WiFi connection or mobile data

Hard disk : 1 TB

Monitor : 1920 X 1080

**3.2 SOFTWARE SPECIFICATION**

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system.

This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allows the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Frontend : HTML, CSS and Bootsrap

Backend : Python, Flask and Sqlite3

Operating System : Windows 10

**Frontend:**

1.HTML

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

2.CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

3.Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

**Backend:**

1.Python

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation.

2.Flask

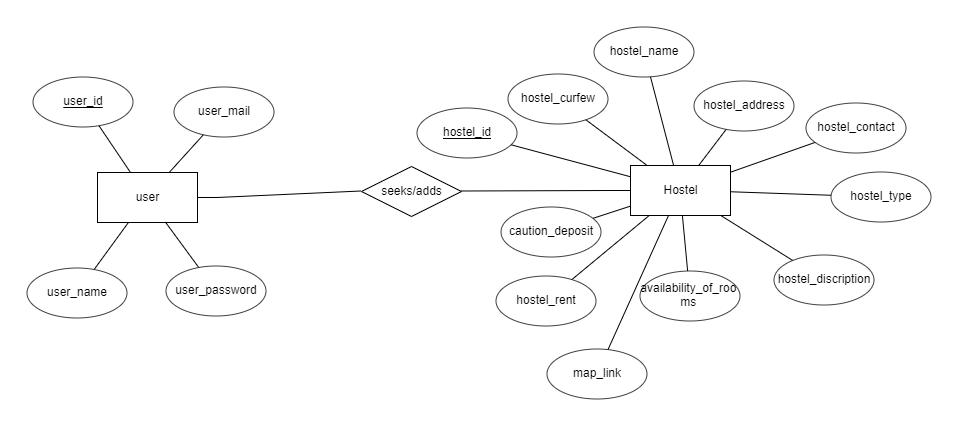
Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

3.Sqlite3

SQLite is a relational database management system contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. SQLite generally follows PostgreSQL syntax.

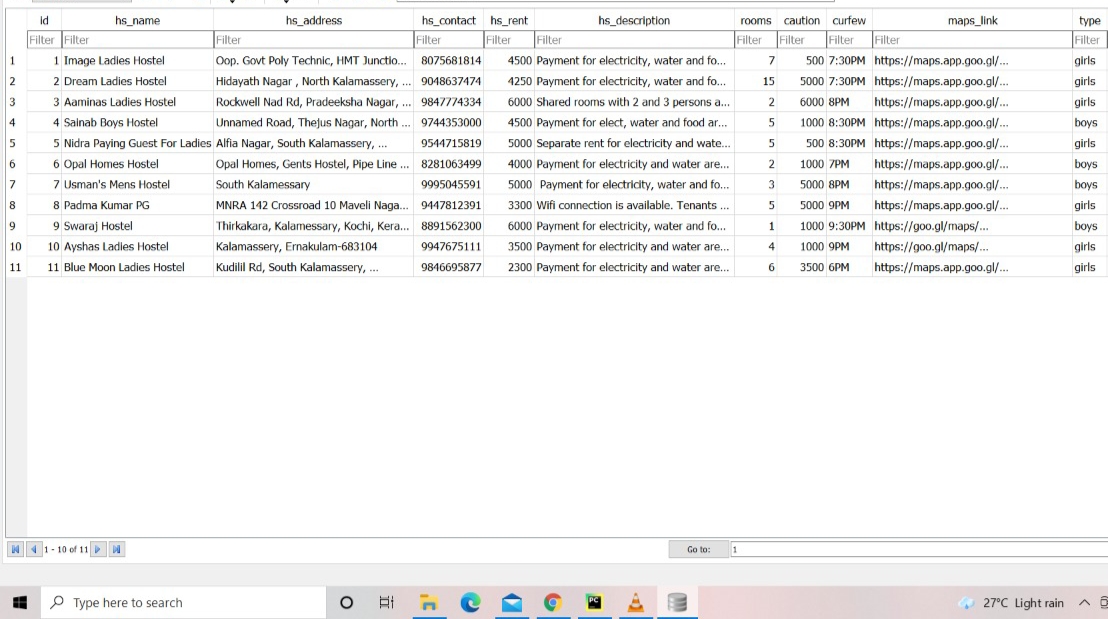
**3. DESIGN**

**4.1 E-R DIAGRAM**

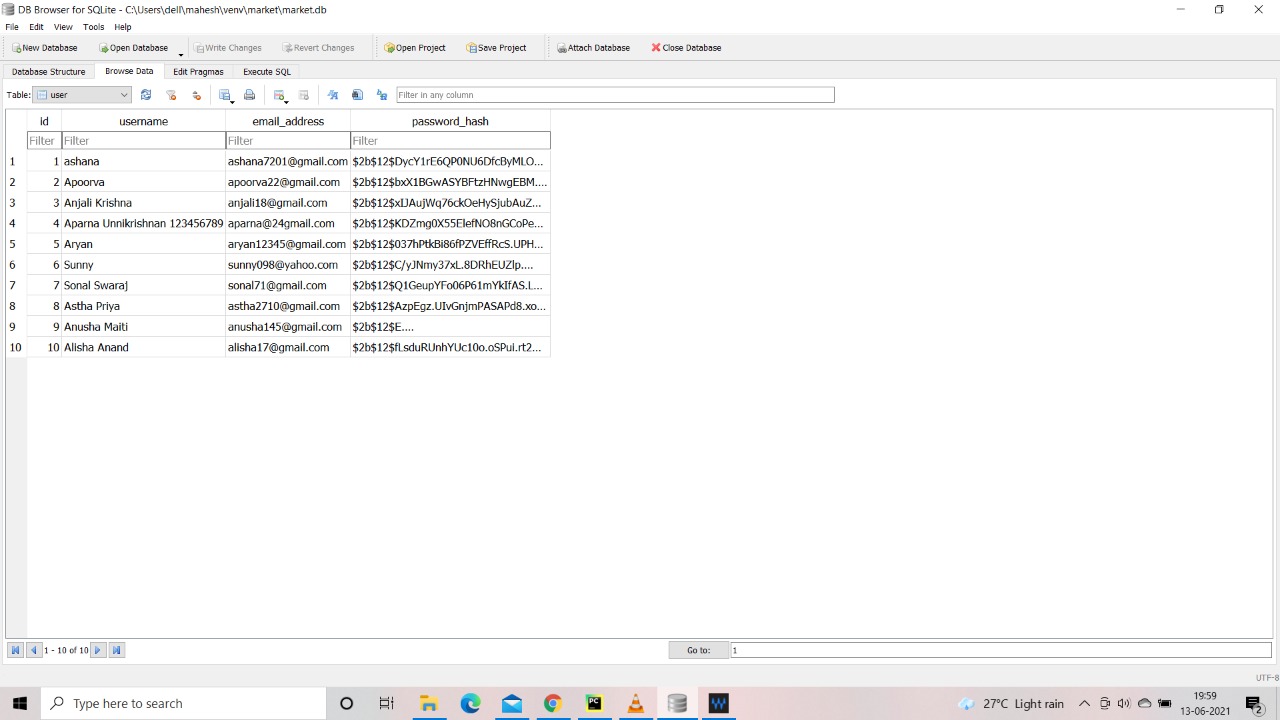


**4.2 TABLE DESIGN**

Database Table for hostel details :



Database table for users :



**DATABASE DESIGN**

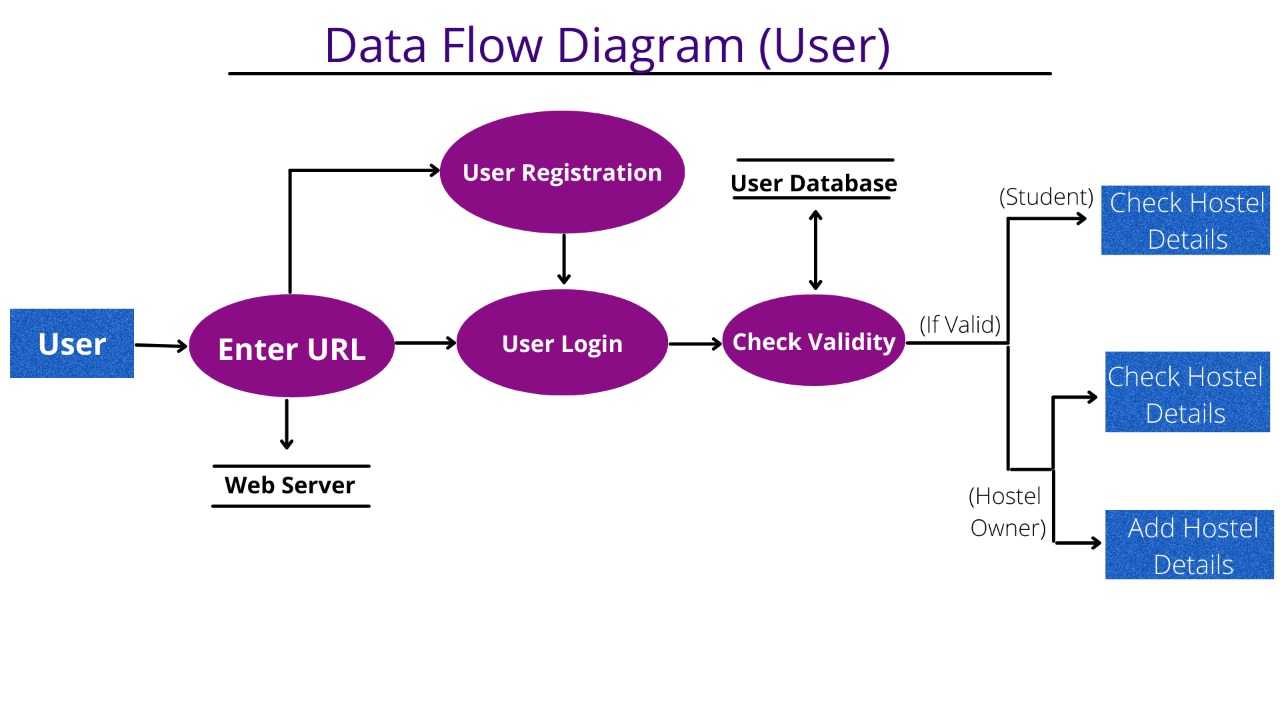
|  |  |
| --- | --- |
| Name | Type |
| user\_id | int |
| user\_mail | nchar |
| user\_name | varchar |
| user\_password | nchar |

Database for user

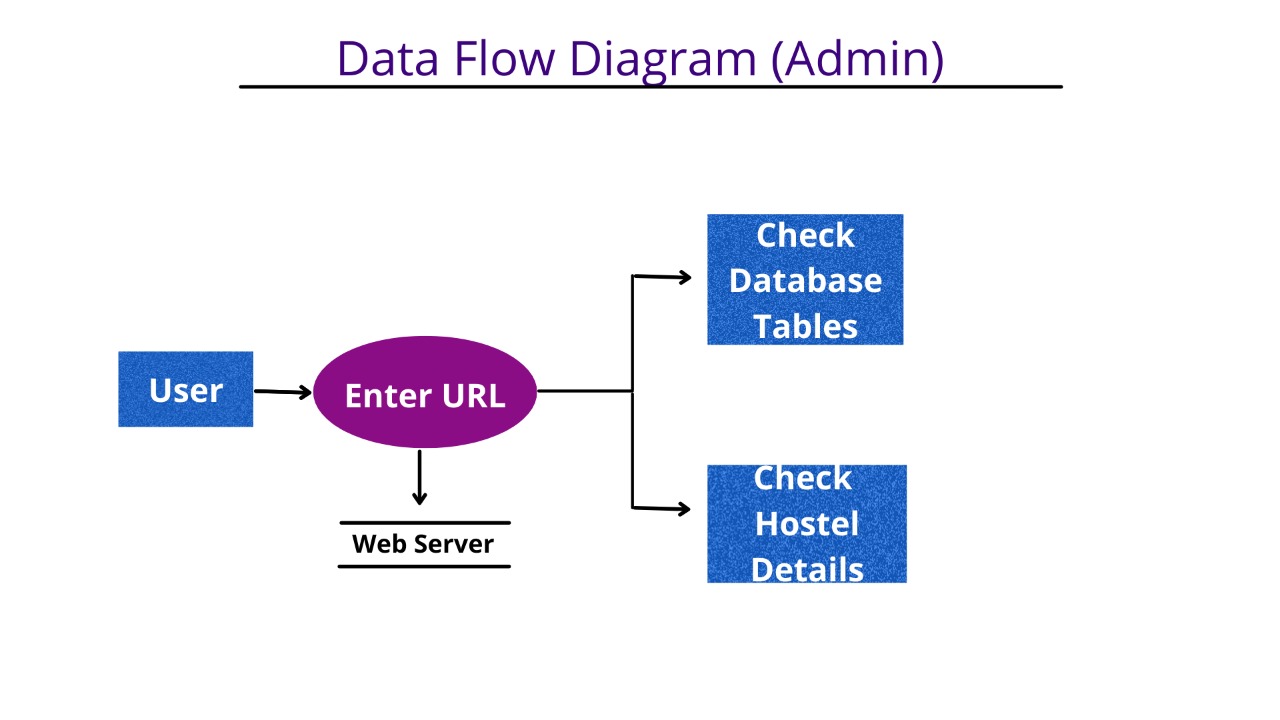
|  |  |
| --- | --- |
| Name | Type |
| hostel\_name | varchar |
| hostel\_address | nchar |
| hostel\_contact | int |
| hostel\_curfew | int |
| caution\_deposit | float |
| hostel\_rent | float |
| hostel\_discription | varchar |
| availability\_of\_rooms | int |
| hostel\_id | int |
| hostel\_type | varchar |
| map\_link | ncahr |

Database table for hostel

**4.3 DATAFLOW DIAGRAMS**

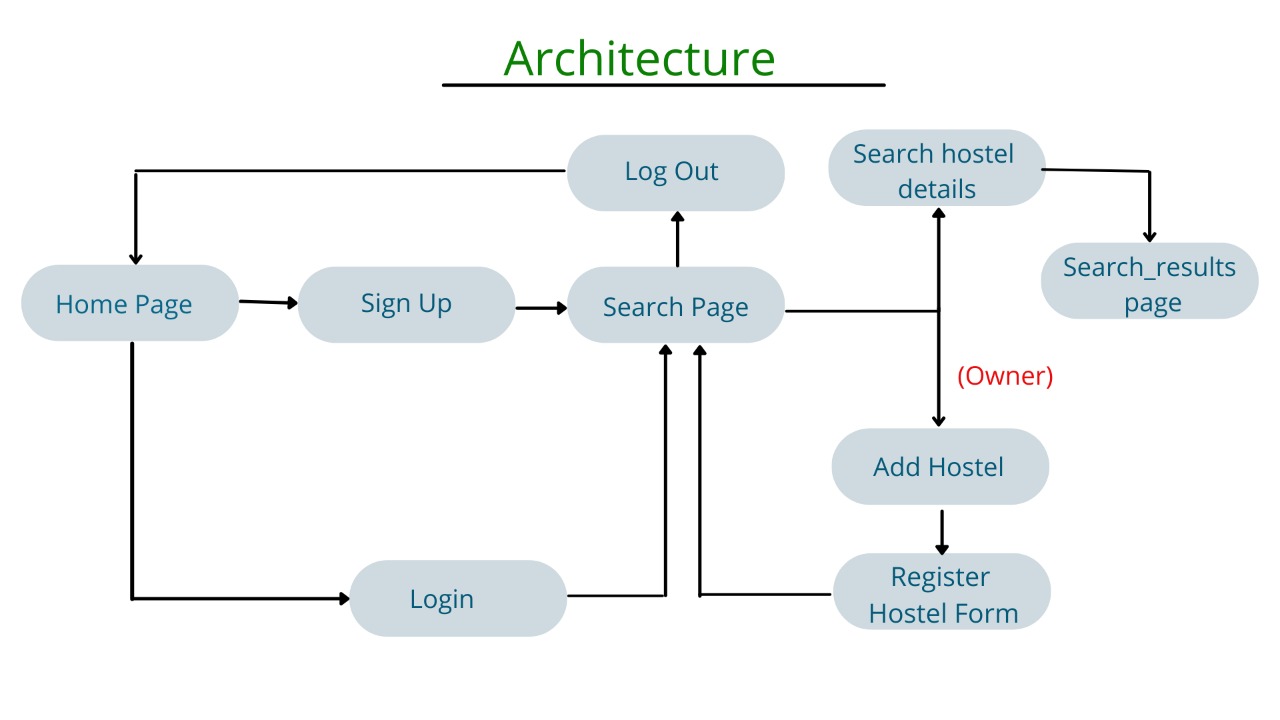


DFD for User



DFD for Admin

**4.4 STRUCTURE OF WEBSITE**

****

**5. SYSTEM IMPLEMENTATION**

**5.1MODULE DESCRIPTION**

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over, an evaluation of change over methods. Apart from planning major task of preparing the implementation are education and training of users. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

**5.1.1 SYSTEM TESTING**

As the part of system testing we execute the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance. Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The various tests performed are unit testing, integration testing and user acceptance testing.

**5.1.1.1 UNIT TESTING**

The software units in the system is are modules and routines that are assembled and integrated to perform a specific function. As a part of unit testing we executed the program for individual modules independently. This enables, to detect errors in coding and logic that are contained within each of the three module. This testing includes entering data that is filling forms and ascertaining if the value matches to the type and entered into the database. The various controls are tested to ensure that each performs its action as required.

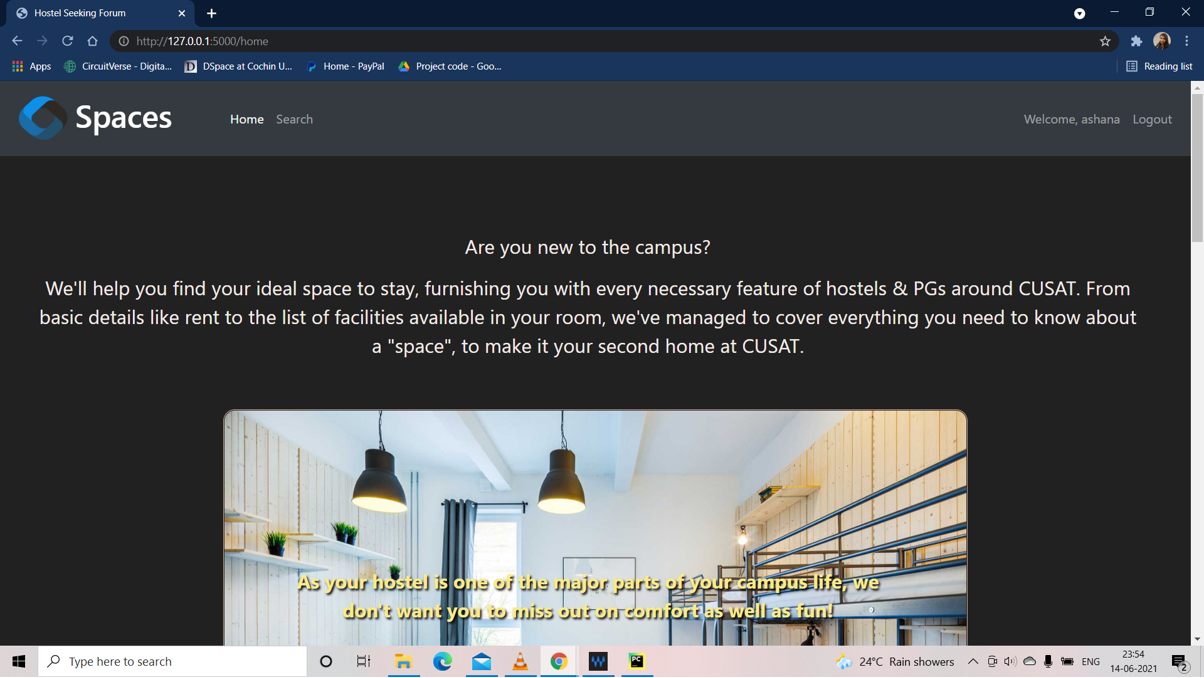
**5.1.1.2 INTEGRATION TESTING**

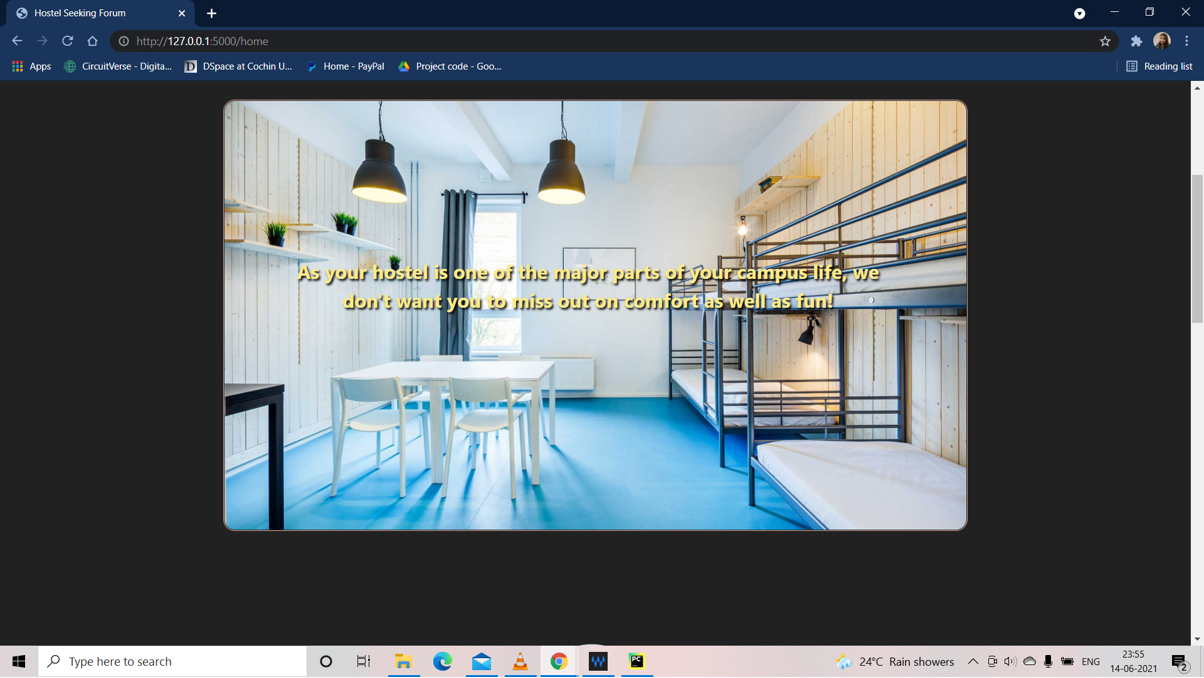
Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here the admin module, sec module and student module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

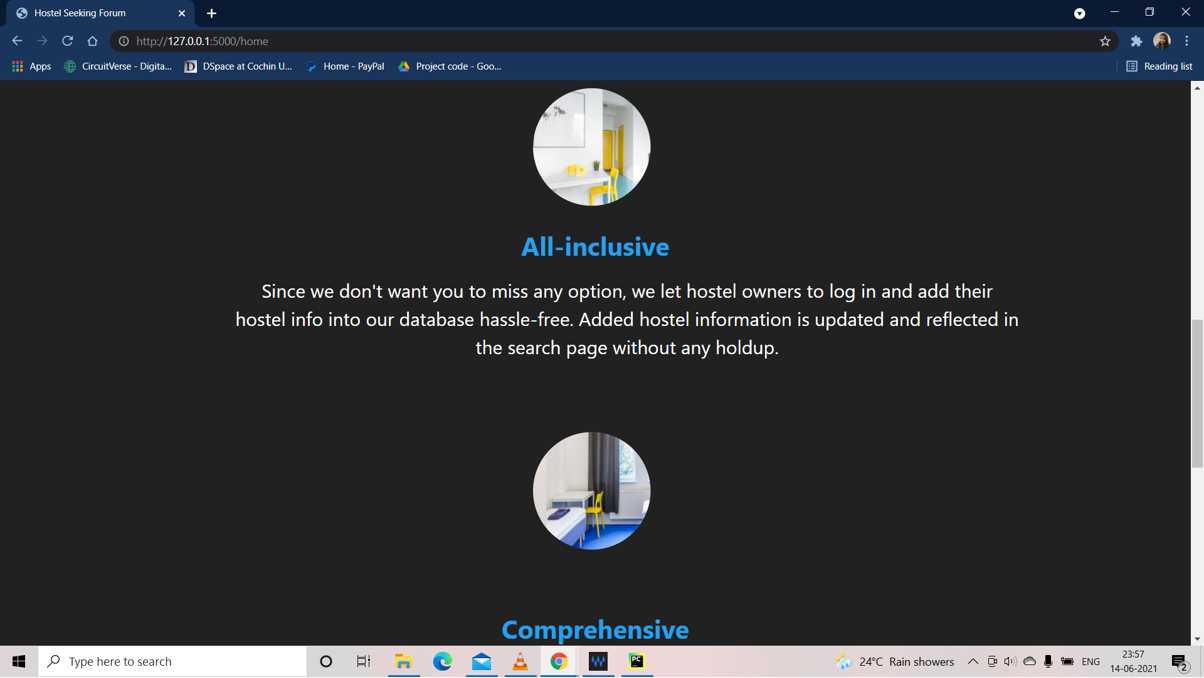
**5.1.1.3 USER ACCEPTANCE TESTING**

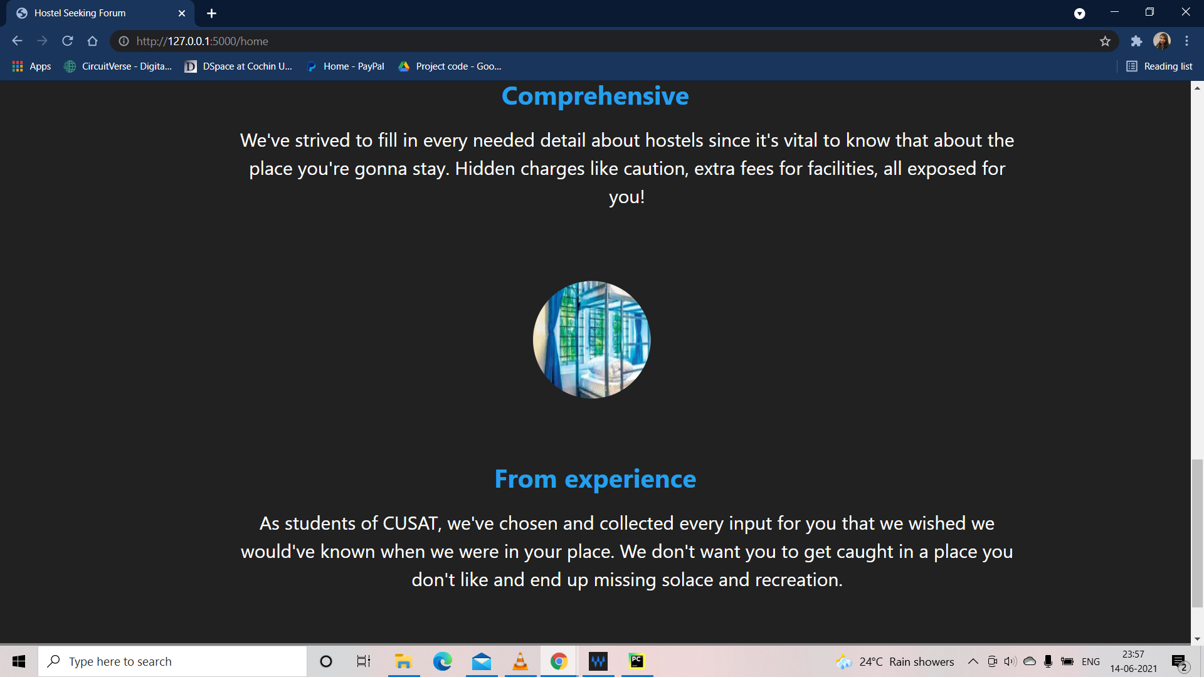
User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping the records of applicants and making changes to the details .

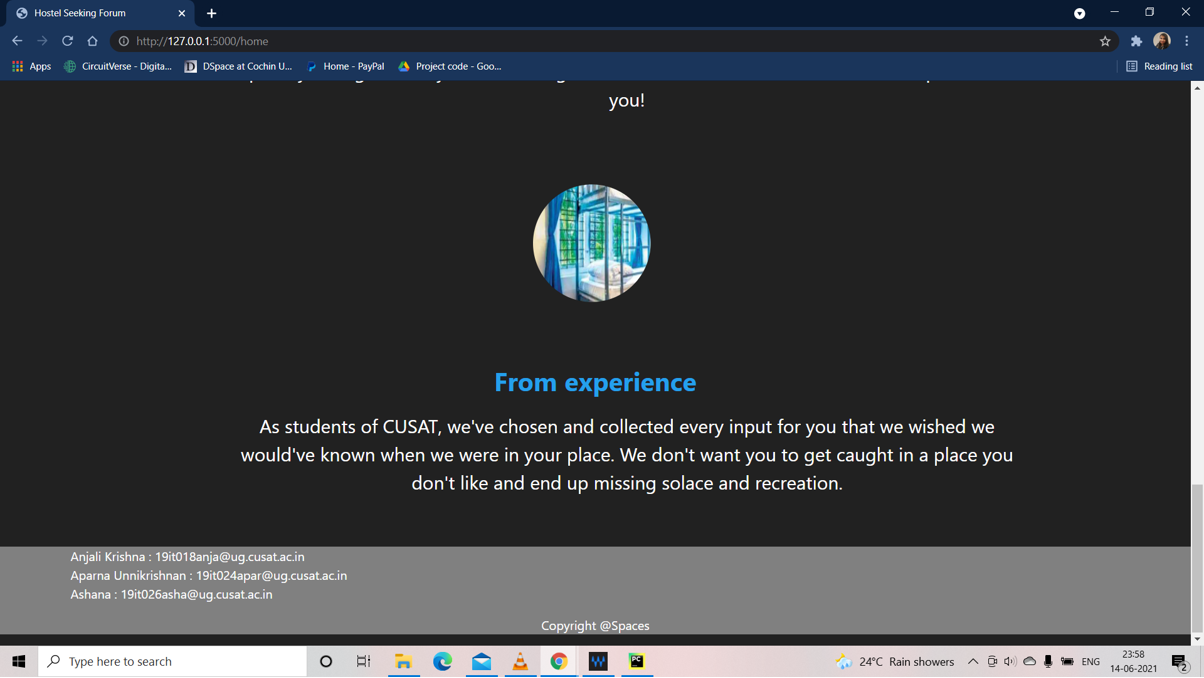
**5.2 SNAPSHOTS**



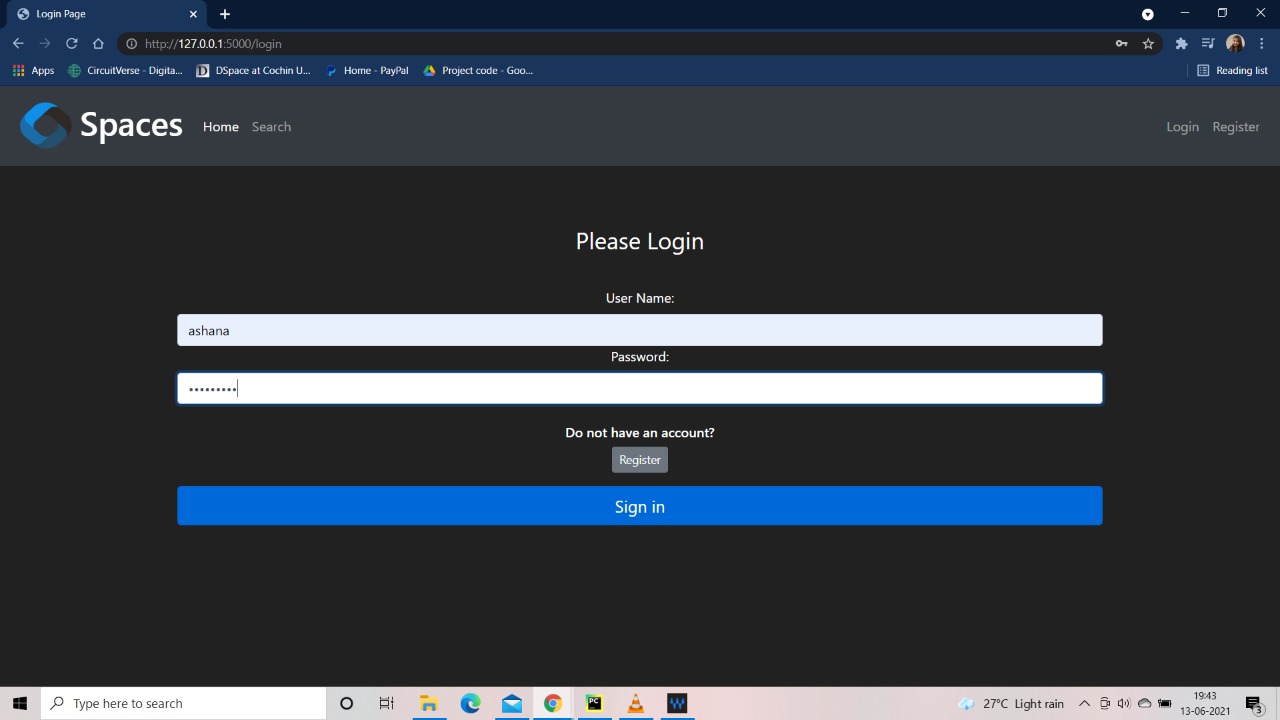


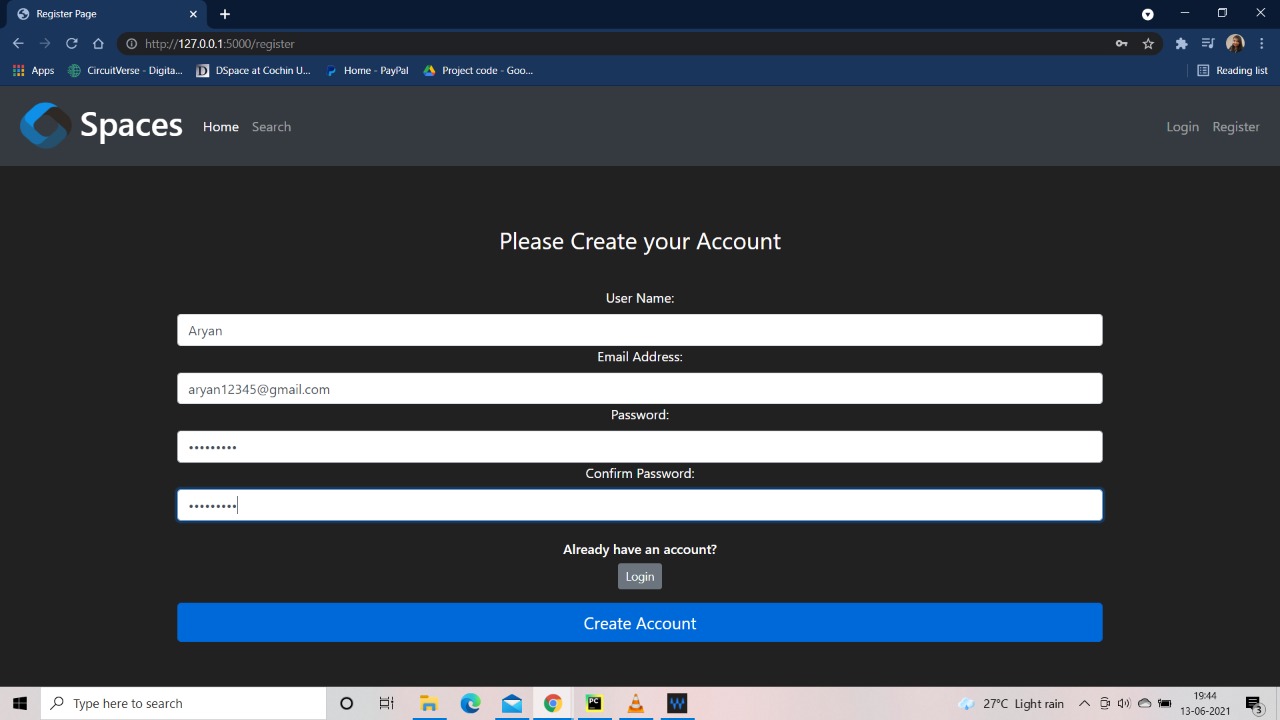




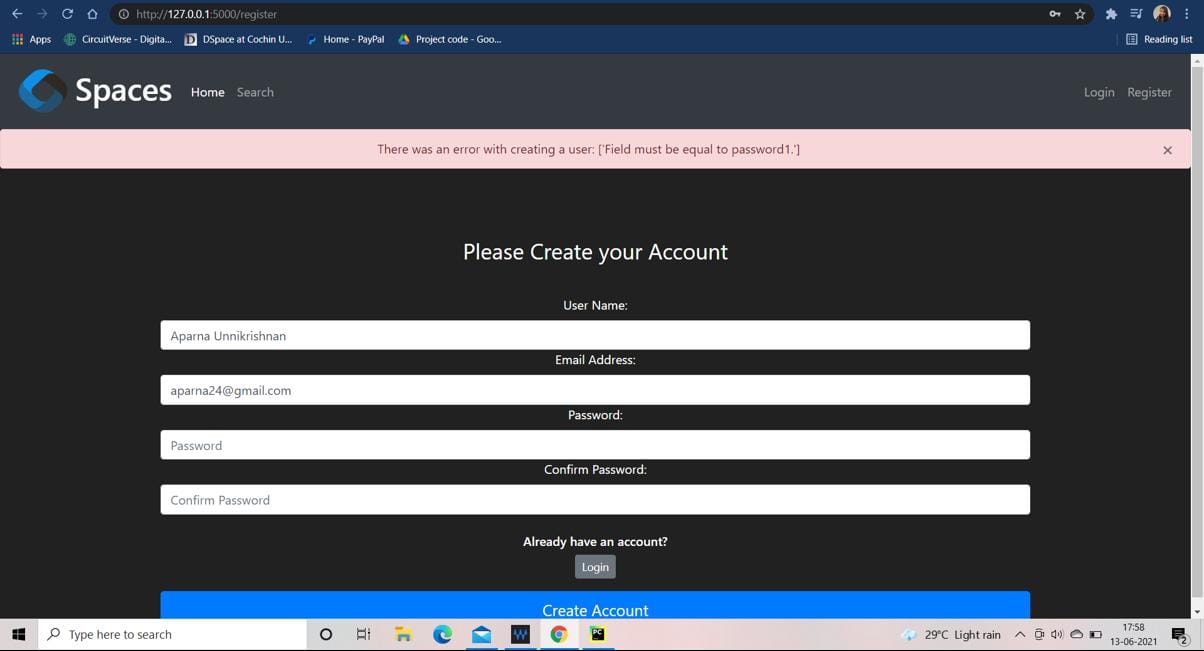


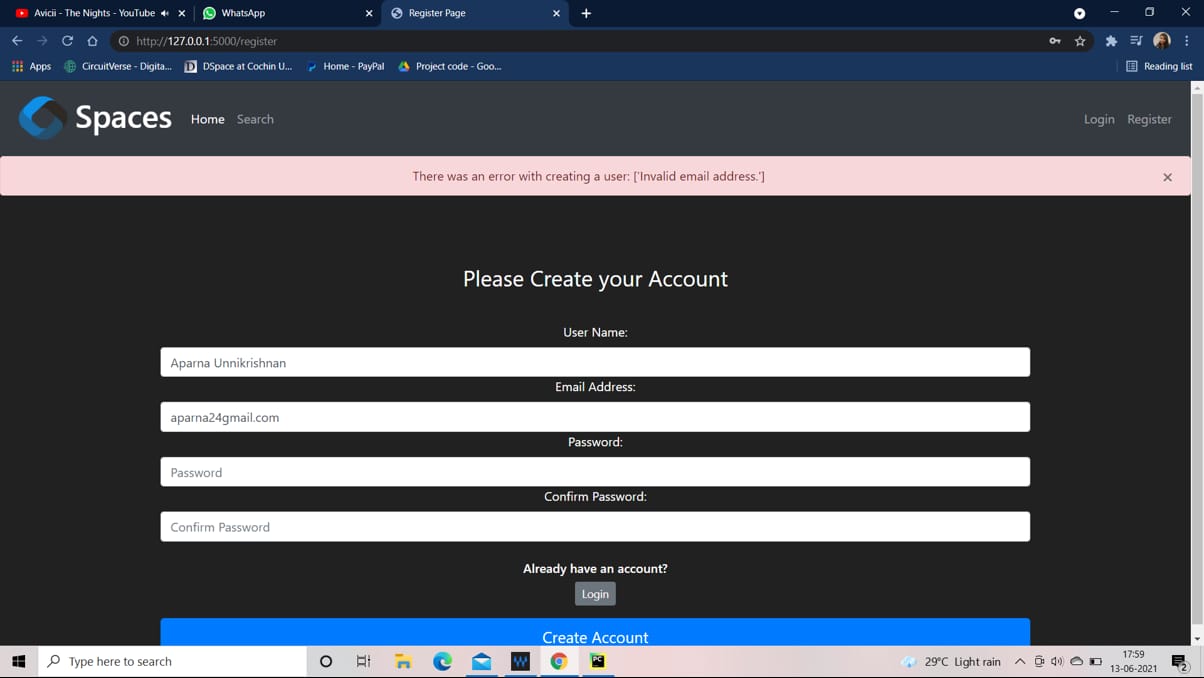
5.2.1 Snapshot of home page

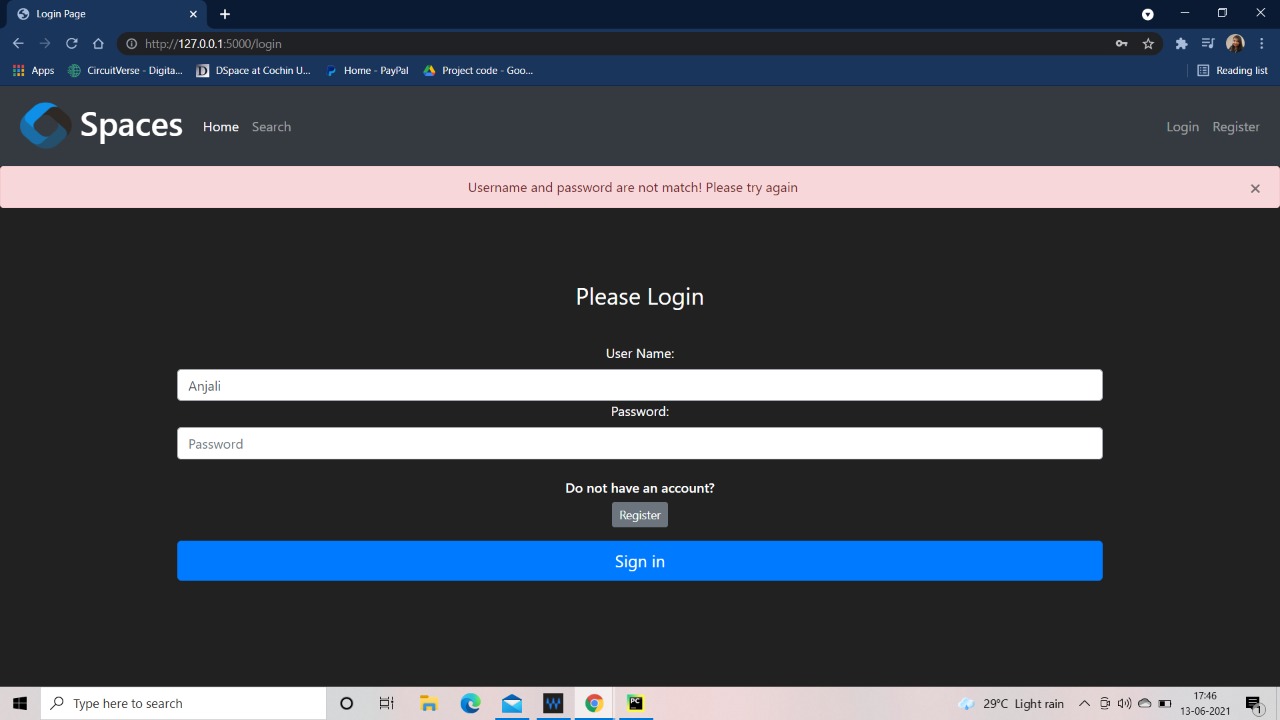




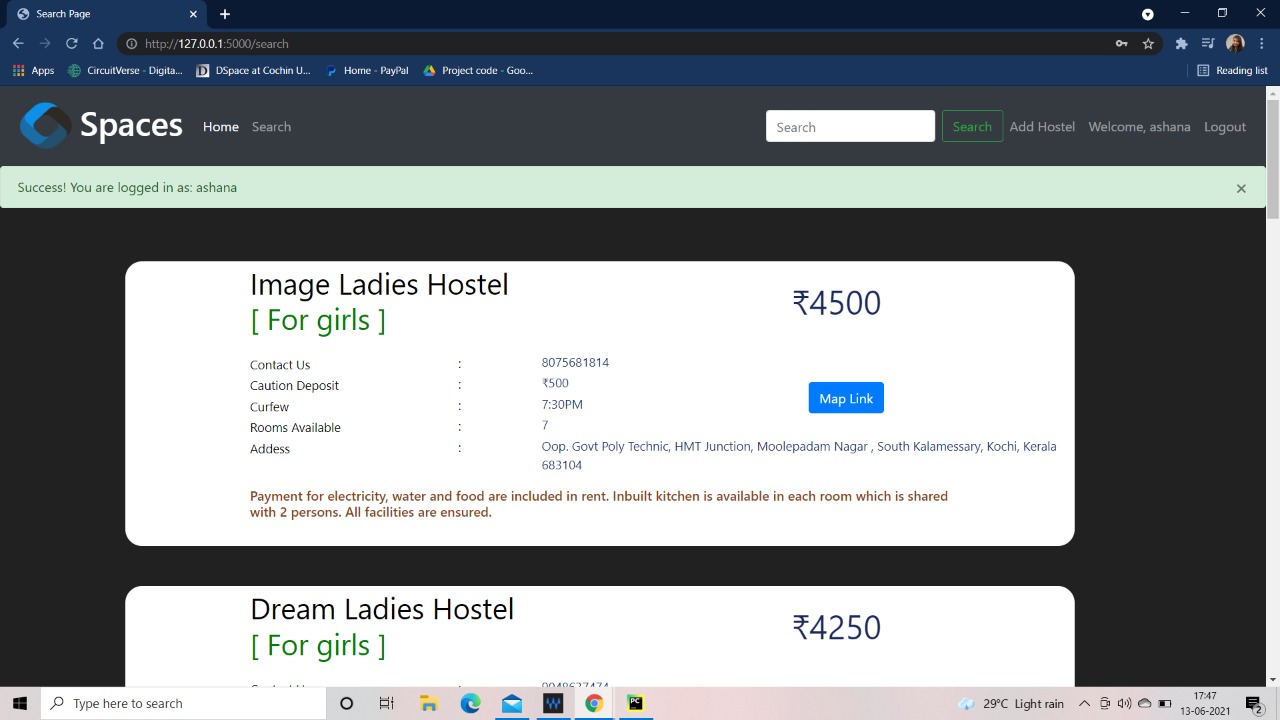
5.2.2 Snapshots of login and register

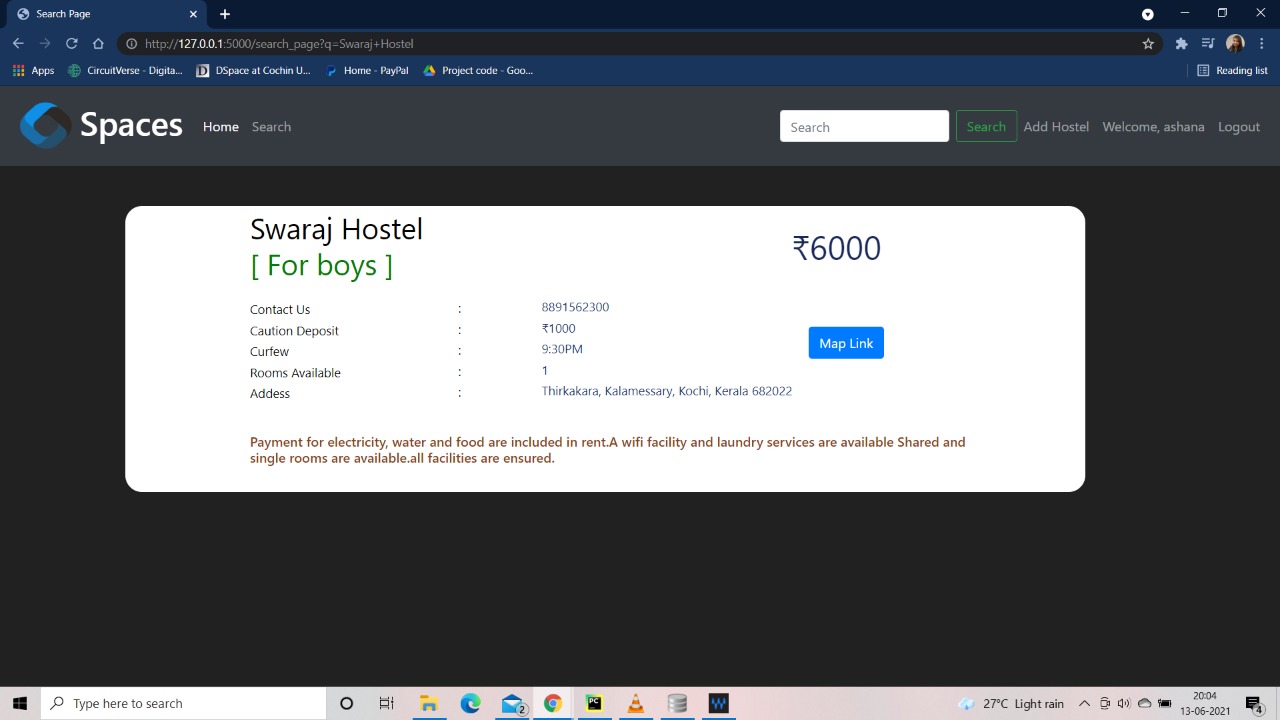




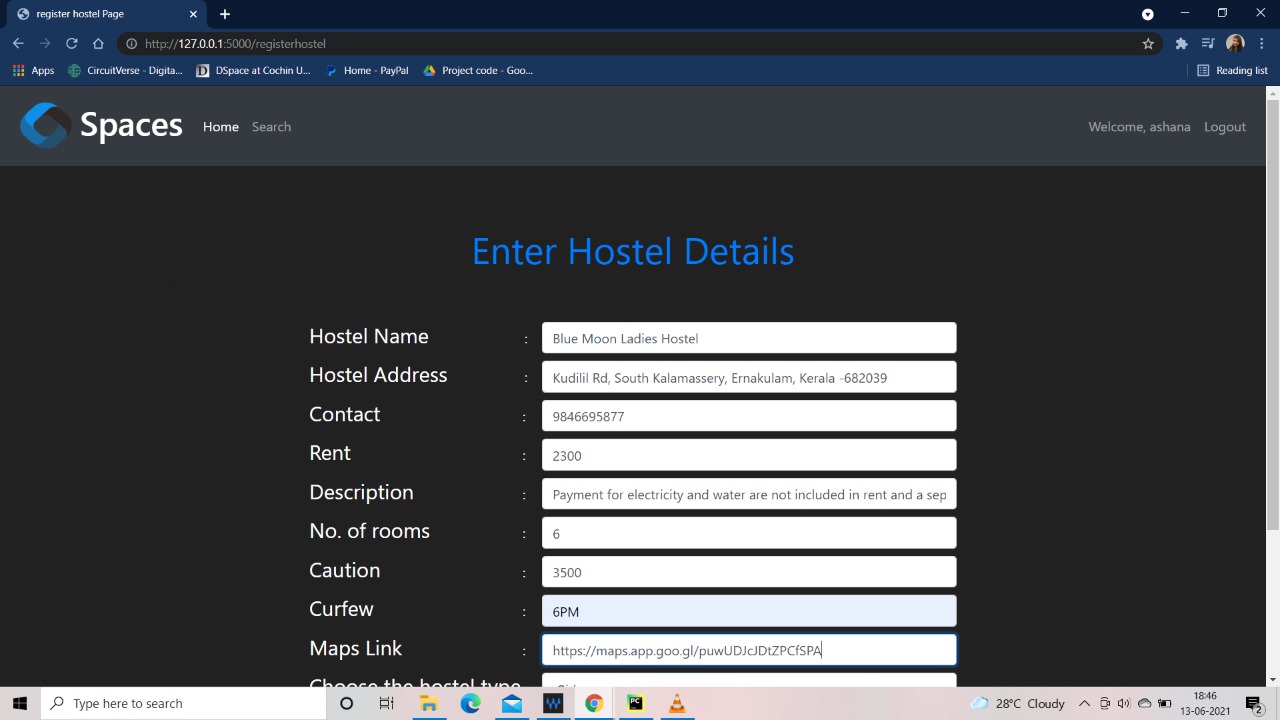


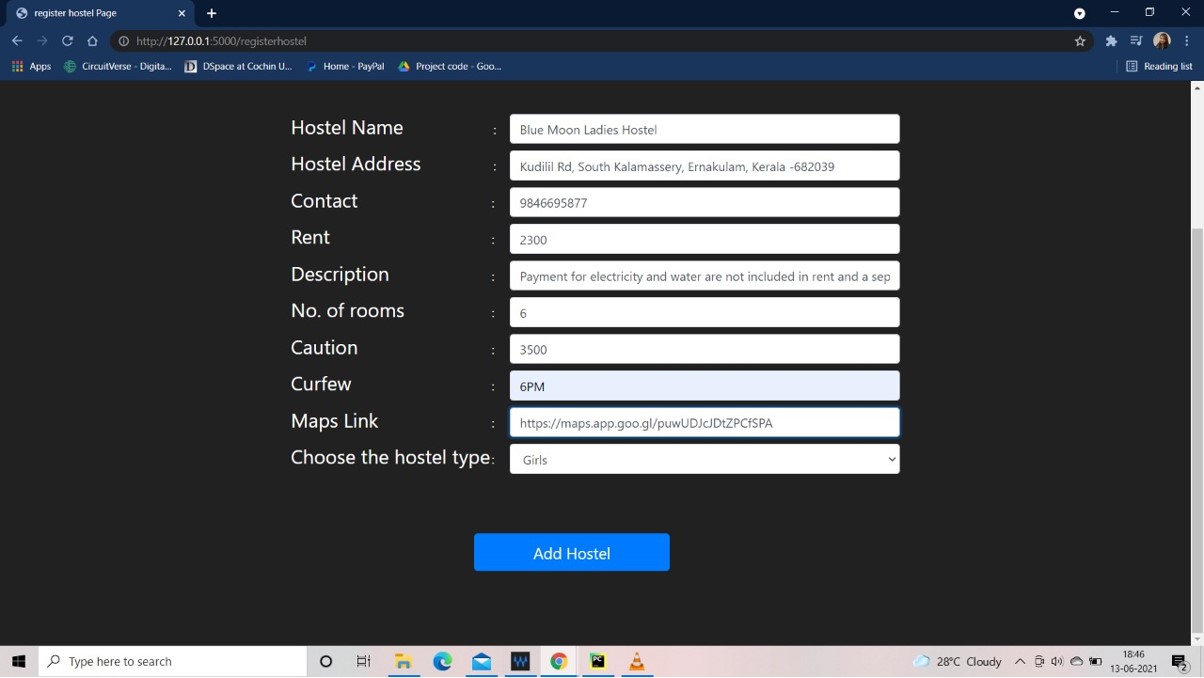
5.2.3 Snapshots of validators

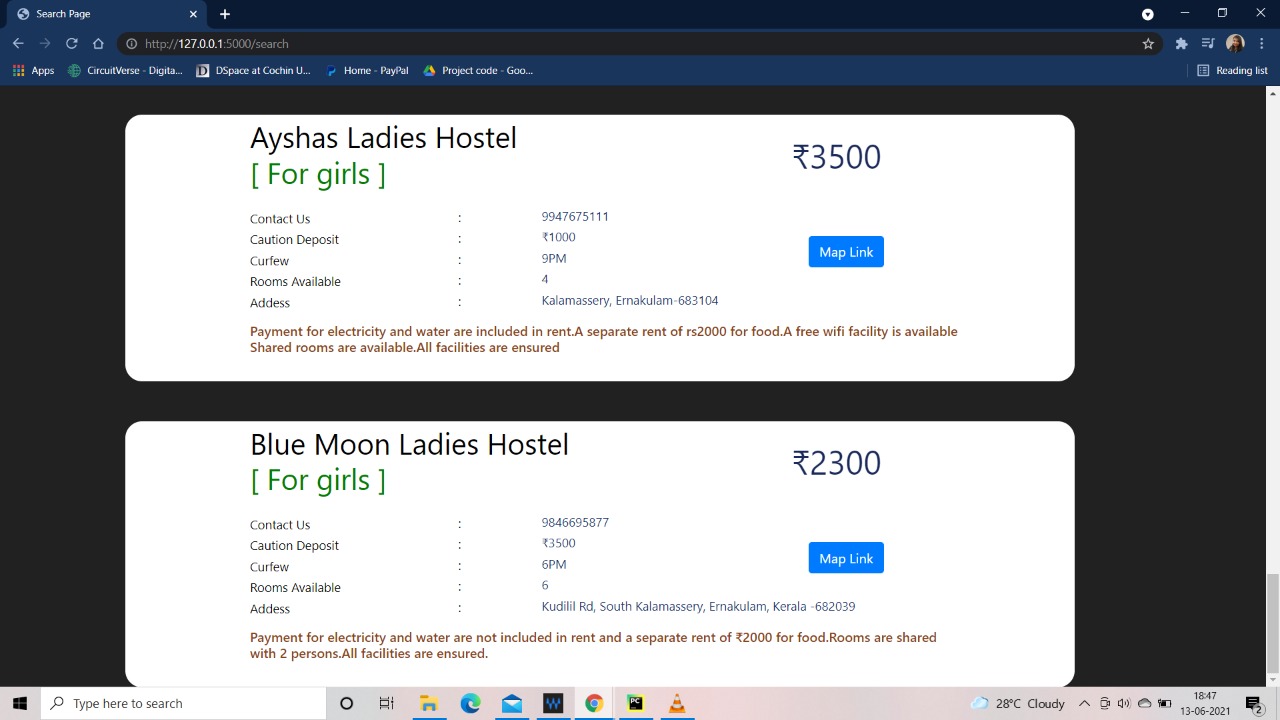




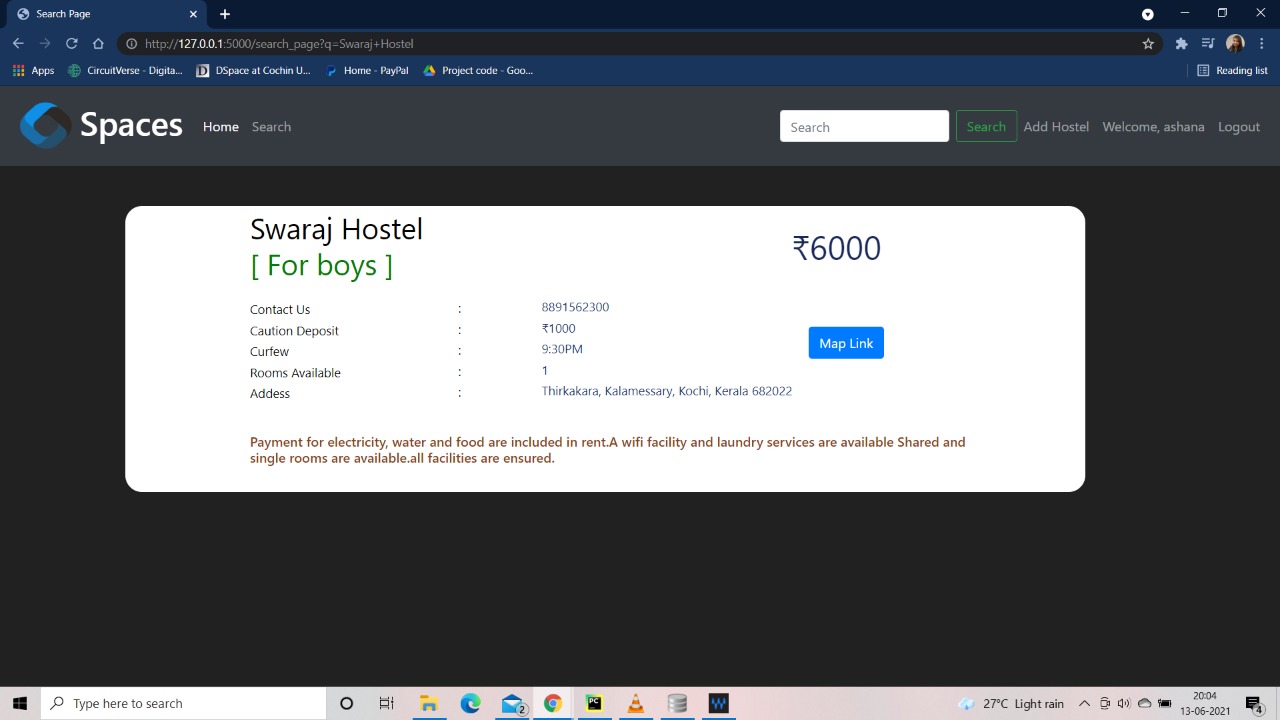
5.2.4 Snapshot of search page



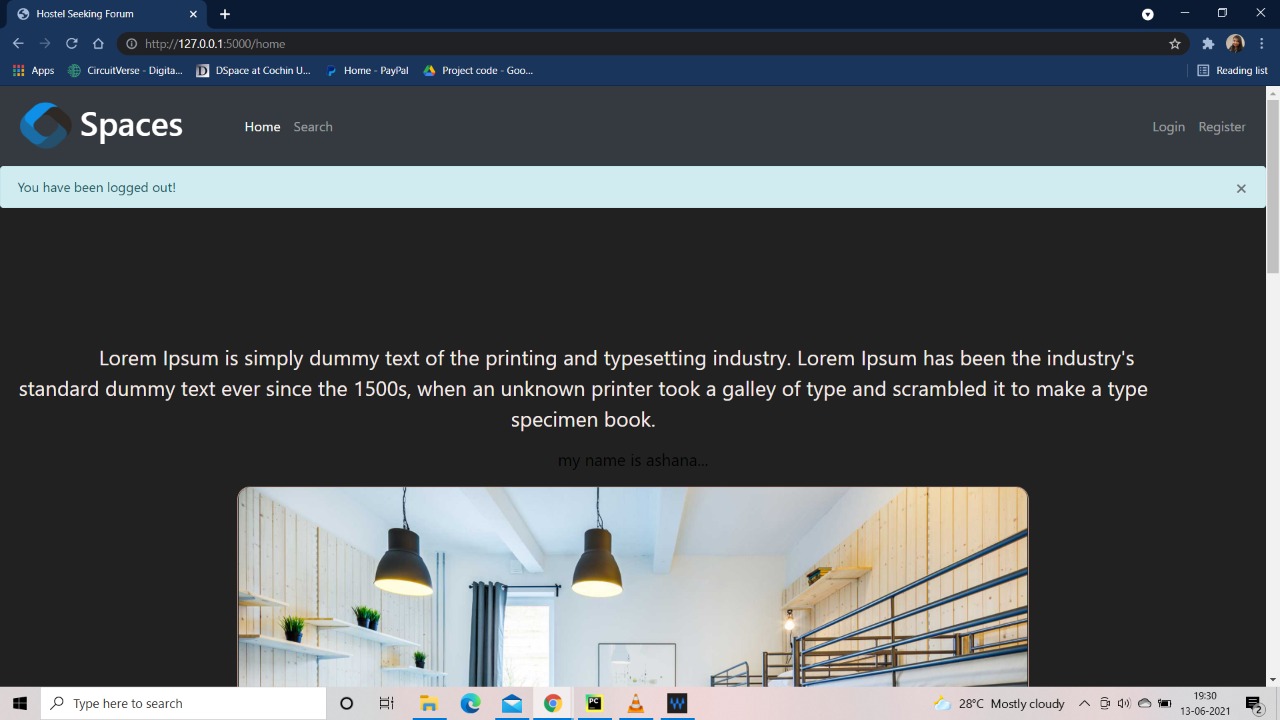




5.2.5 Snapshots of adding hostel details



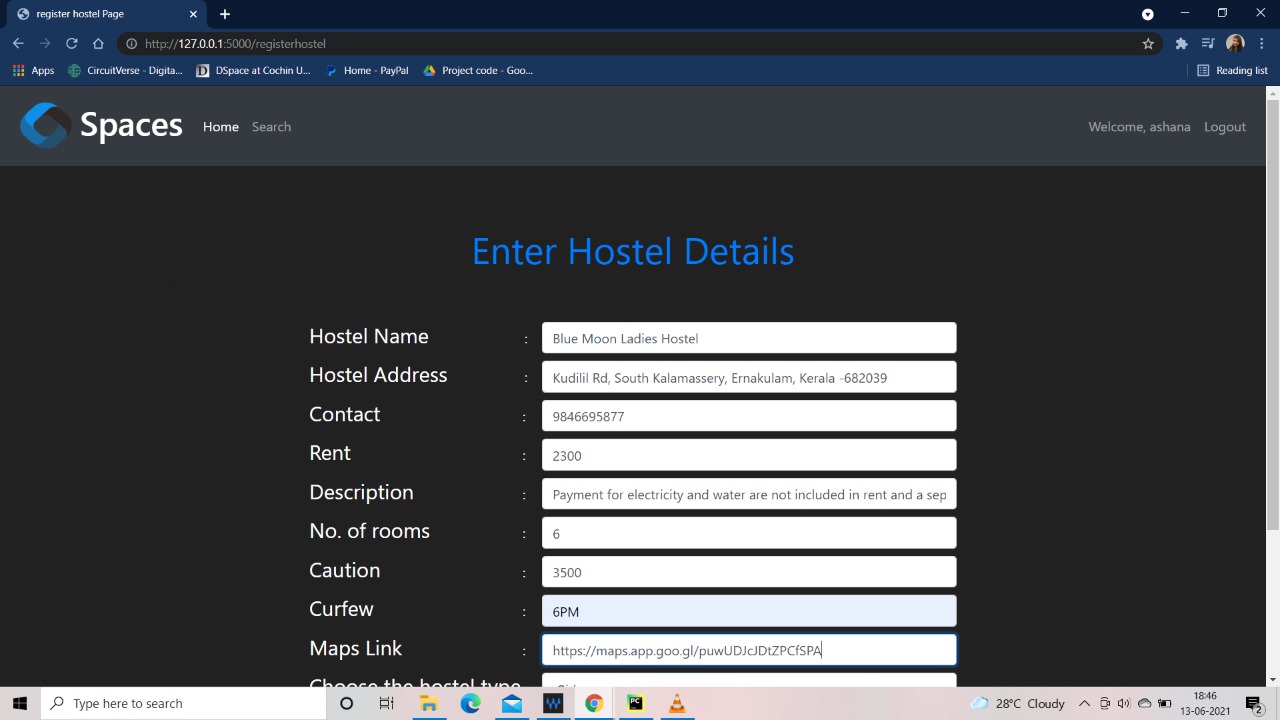
5.2.6 Snapshot of search function



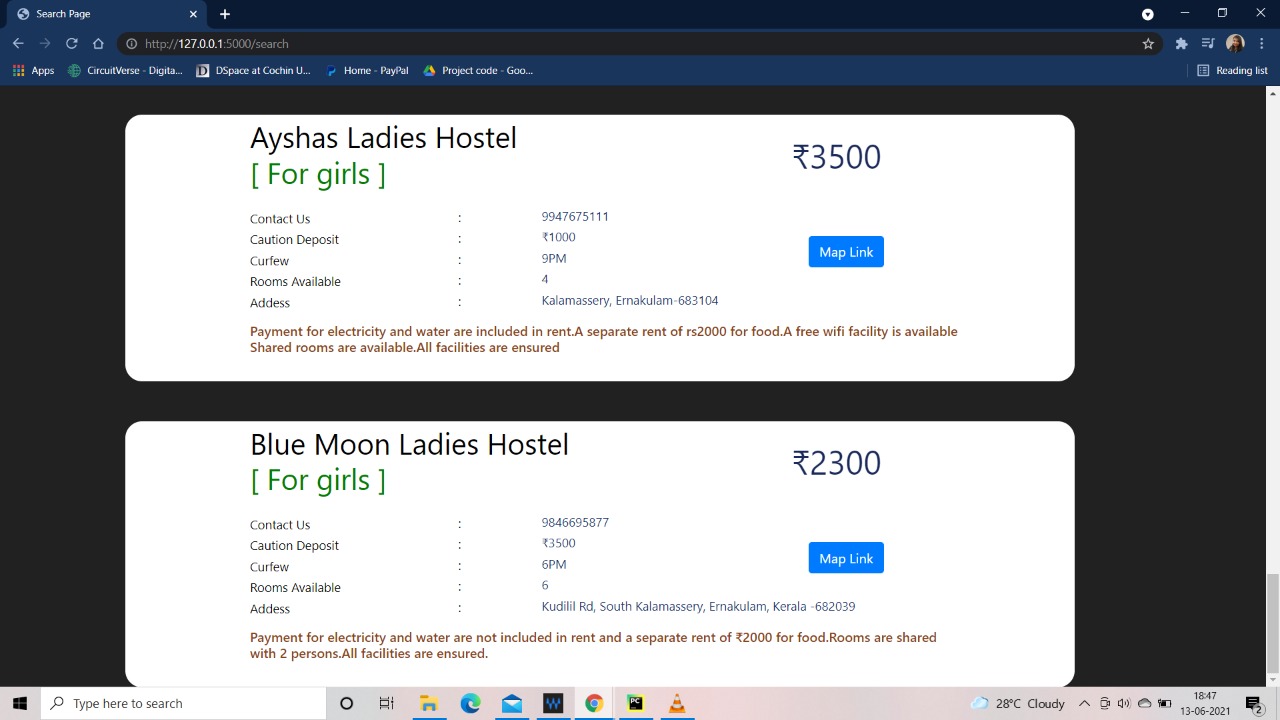
5.2.7 Snapshot of log out button

**6. RESULT AND DISCUSSION**

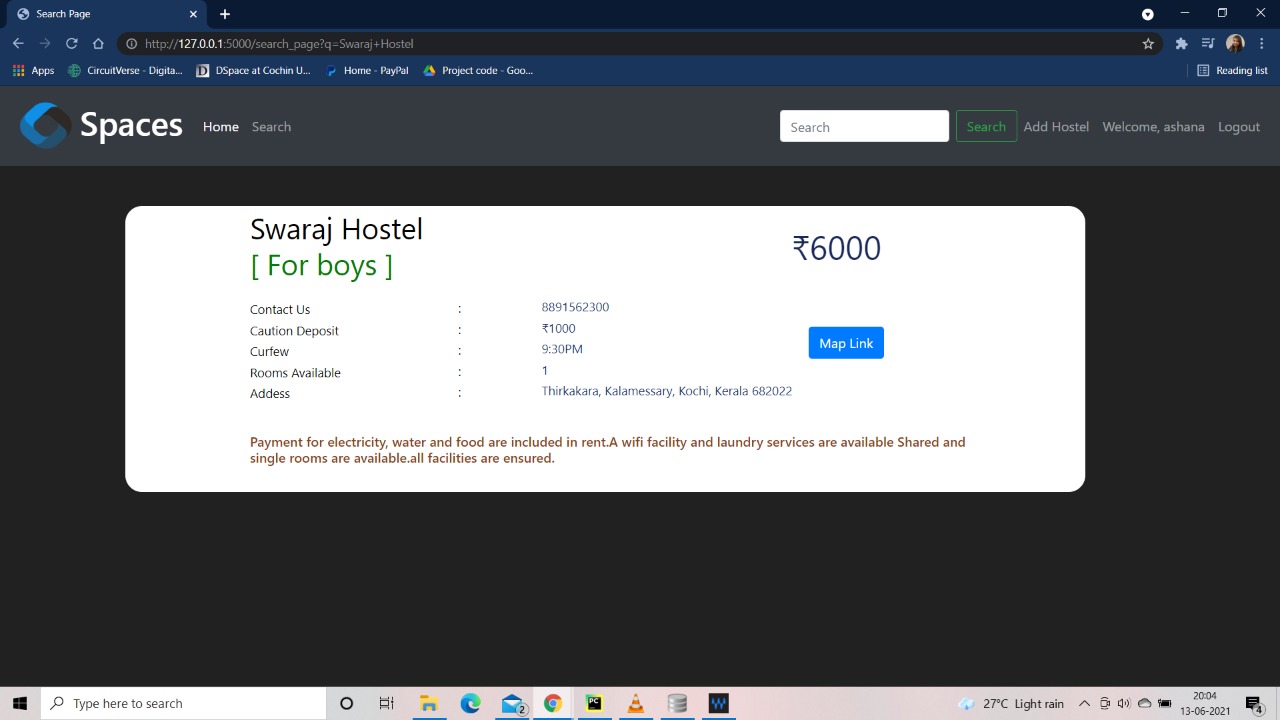
Overall performance of our system is quite simple and fast. Whenever the hostel owners needs to give their hostel on rent they could use this platform to advertise their hostel. Similarly, students can seek hostels from this forum .Users simply have to sign up and if the user is an owner they can add their hostel details via clicking on the Add Hostel button and submitting the hostel details form correctly.



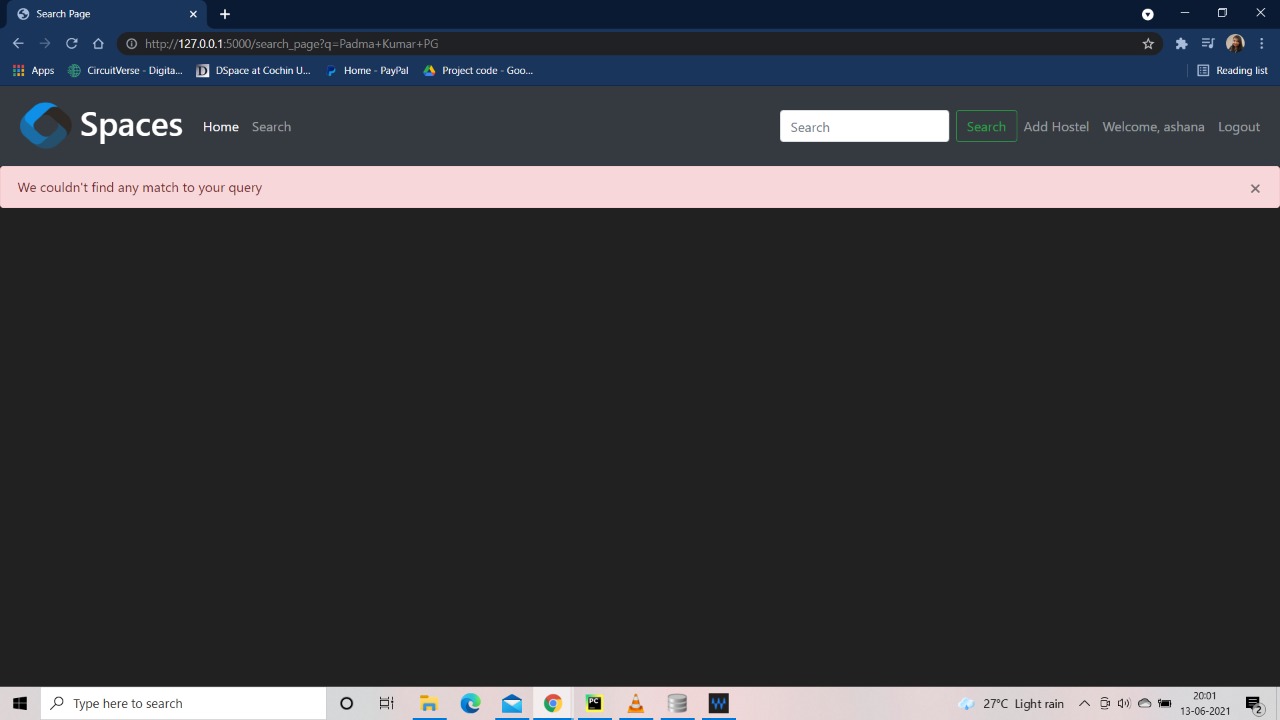
After the successful submission of our hostel details form by the owner , the hostel details will directly be shown on the search page.



We have also provided a search function for students so that they can search the hostels by their names.



If the data entered in the search button does not match with any name of the hostels that are in the database then a flash message “We couldn’t find any match to your query” is displayed on the screen.



**7. CONCLUSION AND FUTURE SCOPE**

To conclude the description about the project, it is developed using HTML, CSS, Bootstrap, Python, Flask and Sqlite3. It is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement. HOSTEL SEEKING FORUM is very useful for students as well as hostel owners. This hostel finding software is designed for people who are in a search of hostels or PG s. For the past few years the numbers of educational institutions are increasing rapidly. Thereby the numbers of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the problems on finding a hostel for students and advertisement platform for hostel owners. Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

**8. REFERENCES**

**1.** [**https://stackoverflow.com/**](https://stackoverflow.com/)

**2.** [**https://www.w3schools.com/**](https://www.w3schools.com/)

**3.** [**https://youtu.be/byHcYRpMgI4**](https://youtu.be/byHcYRpMgI4)

**4.** [**https://youtu.be/UqRawhLNE-0**](https://youtu.be/UqRawhLNE-0)

**9. APPENDIX**

**SOURCE CODE**

**BACKEND PART**

1. **run.py**

from market import app  
  
#Checks if the run.py file has executed directly and not imported  
if \_\_name\_\_ == '\_\_main\_\_':  
    app.run(debug=True)

1. **\_\_init\_\_.py**
2. from flask import Flask  
   from flask\_sqlalchemy import SQLAlchemy  
   from flask\_bcrypt import Bcrypt  
   from flask\_login import LoginManager  
     
     
   app = Flask(\_\_name\_\_)  
   app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///market.db'  
   app.config['SECRET\_KEY'] = 'ec9439cfc6c796ae2029594d'  
   db = SQLAlchemy(app)  
     
   bcrypt = Bcrypt(app)  
   login\_manager = LoginManager(app)  
   login\_manager.login\_view = "login\_page"  
   login\_manager.login\_message\_category = "info"  
   from market import routes

**3.forms.py**

1. from flask\_wtf import FlaskForm  
   from wtforms import StringField, IntegerField, SelectField, PasswordField, SubmitField  
   from wtforms.validators import Length, EqualTo, Email, DataRequired, InputRequired, ValidationError  
   from market.models import User  
     
     
   class RegisterForm(FlaskForm):  
       def validate\_username(self, username\_to\_check):  
           user = User.query.filter\_by(username=username\_to\_check.data).first()  
           if user:  
               raise ValidationError('Username already exists! Please try a different username')  
     
       def validate\_email\_address(self, email\_address\_to\_check):  
           email\_address = User.query.filter\_by(email\_address=email\_address\_to\_check.data).first()  
           if email\_address:  
               raise ValidationError('Email Address already exists! Please try a different email address')  
     
       username = StringField(label='User Name:', validators=[Length(min=2, max=30), DataRequired()])  
       email\_address = StringField(label='Email Address:', validators=[Email(), DataRequired()])  
       password1 = PasswordField(label='Password:', validators=[Length(min=6), DataRequired()])  
       password2 = PasswordField(label='Confirm Password:', validators=[EqualTo('password1'), DataRequired()])  
       submit = SubmitField(label='Create Account')  
     
     
   class LoginForm(FlaskForm):  
       username = StringField(label='User Name:', validators=[DataRequired()])  
       password = PasswordField(label='Password:', validators=[DataRequired()])  
       submit = SubmitField(label='Sign in')  
     
     
   class RegisterHostel(FlaskForm):  
       hs\_name = StringField('Hostel Name')  
       hs\_address = StringField('Hostel Address')  
       hs\_contact = IntegerField('Contact')  
       hs\_rent = IntegerField('Rent')  
       hs\_description = StringField('Description')  
       rooms = IntegerField('No. of rooms')  
       caution = IntegerField('Caution')  
       curfew = StringField('Curfew')  
       maps\_link = StringField('Maps Link')  
       type = SelectField('Choose the hostel type', [ InputRequired()], choices=[ ('girls', 'Girls'),('boys', 'Boys') ])  
       submit = SubmitField('Add Hostel')

**4.models.py**

from market import db, login\_manager  
from market import bcrypt  
from flask\_login import UserMixin  
  
  
@login\_manager.user\_loader  
def load\_user(user\_id):  
    return User.query.get(int(user\_id))  
  
class User(db.Model, UserMixin):  
    id = db.Column(db.Integer(), primary\_key=True)  
    username = db.Column(db.String(length=30), nullable=False, unique=True)  
    email\_address = db.Column(db.String(length=50), nullable=False, unique=True)  
    password\_hash = db.Column(db.String(length=60), nullable=False)  
    hostel = db.relationship('Hostels', backref='owned\_user', lazy=True)  
  
    @property  
    def password(self):  
        return self.password  
  
    @password.setter  
    def password(self, plain\_text\_password):  
        self.password\_hash = bcrypt.generate\_password\_hash(plain\_text\_password).decode('utf-8')  
  
    def check\_password\_correction(self, attempted\_password):  
        return bcrypt.check\_password\_hash(self.password\_hash, attempted\_password)  
  
class Hostels(db.Model):  
    id = db.Column(db.Integer(), primary\_key=True)  
    hs\_name = db.Column(db.String(length=30), nullable=False, unique=True)  
    hs\_address = db.Column(db.String(length=1024), nullable=False, unique=True)  
    hs\_contact = db.Column(db.Integer(), nullable=False)  
    hs\_rent = db.Column(db.Integer(), nullable=False)  
    hs\_description = db.Column(db.String(length=1024), nullable=False)  
    rooms = db.Column(db.Integer(), nullable=False)  
    caution = db.Column(db.Integer(), nullable=False)  
    curfew = db.Column(db.String(length=30), nullable=False)  
    maps\_link = db.Column(db.String(length=1024), nullable=True, unique=True)  
    type = db.Column(db.String(length=30), nullable=False)  
    owner = db.Column(db.Integer(), db.ForeignKey('[user.id](http://user.id/)'))  
    def \_\_repr\_\_(self):  
        return f'Hostels {self.hs\_name}'

1. **routes.py**

from market import app  
from flask import render\_template, redirect, url\_for, flash, request  
from market.models import Hostels, User  
from market.forms import RegisterForm, LoginForm, RegisterHostel  
from market import db  
from flask\_login import login\_user, logout\_user, login\_required  
  
  
@app.route('/')  
@app.route('/home')  
def home\_page():  
    return render\_template('home.html')  
  
@app.route('/search', methods=['GET', 'POST'])  
@login\_required  
def search\_page():  
    hostels = Hostels.query.all()  
    return render\_template('search.html', hostels=hostels)  
  
@app.route('/register', methods=['GET', 'POST'])  
def register\_page():  
    form = RegisterForm()  
    if form.validate\_on\_submit():  
        user\_to\_create = User(username=form.username.data,  
                              email\_address=form.email\_address.data,  
                              password=form.password1.data)  
        db.session.add(user\_to\_create)  
        db.session.commit()  
        login\_user(user\_to\_create)  
        flash(f"Account created successfully! You are now logged in as {user\_to\_create.username}", category='success')  
        return redirect(url\_for('search\_page'))  
    if form.errors != {}: #If there are not errors from the validations  
        for err\_msg in form.errors.values():  
            flash(f'There was an error with creating a user: {err\_msg}', category='danger')  
  
    return render\_template('register.html', form=form)  
  
@app.route('/login', methods=['GET', 'POST'])  
def login\_page():  
    form = LoginForm()  
    if form.validate\_on\_submit():  
        attempted\_user = User.query.filter\_by(username=form.username.data).first()  
        if attempted\_user and attempted\_user.check\_password\_correction(  
                attempted\_password=form.password.data  
        ):  
            login\_user(attempted\_user)  
            flash(f'Success! You are logged in as: {attempted\_user.username}', category='success')  
            return redirect(url\_for('search\_page'))  
        else:  
            flash('Username and password are not match! Please try again', category='danger')  
  
    return render\_template('login.html', form=form)  
  
@app.route('/logout')  
def logout\_page():  
    logout\_user()  
    flash("You have been logged out!", category='info')  
    return redirect(url\_for("home\_page"))  
  
@app.route('/registerhostel', methods=['GET', 'POST'])  
def add\_hostel():  
    form = RegisterHostel()  
    if form.validate\_on\_submit():  
        hs\_name = request.form['hs\_name']  
        hs\_address = request.form['hs\_address']  
        hs\_contact = request.form['hs\_contact']  
        hs\_rent = request.form['hs\_rent']  
        hs\_description = request.form['hs\_description']  
        rooms = request.form['rooms']  
        caution = request.form['caution']  
        curfew = request.form['curfew']  
        maps\_link = request.form['maps\_link']  
        type = request.form['type']  
        record = Hostels(hs\_name=form.hs\_name.data, hs\_address=form.hs\_address.data, hs\_contact=form.hs\_contact.data, hs\_rent=form.hs\_rent.data, hs\_description=form.hs\_description.data, rooms=form.rooms.data, caution=form.caution.data, curfew=form.curfew.data, maps\_link=form.maps\_link.data, type=form.type.data)  
        db.session.add(record)  
        db.session.commit()  
        return render\_template('search.html')  
    else:  
        return render\_template('registerhostel.html', form=form)  
  
@app.route('/search\_page', methods=['GET', 'POST'])  
def searches():  
    q=request.args.get('q')  
      
    if db.session.query(db.exists().where(Hostels.hs\_name == q)).scalar():  
        form=Hostels.query.filter(Hostels.hs\_name.contains(q))  
        return render\_template('search\_results.html', hostels=form)  
    else:  
        flash('We couldn\'t find any match to your query', category='danger')  
        return render\_template('search\_results.html')

**6.market.db**

**FRONTEND PART**

1. **base.html**
2. <!doctype html>  
   <html lang="en">  
      <head>  
         <!-- Required meta tags -->  
         <meta charset="utf-8">  
         <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">  
         <!-- Bootstrap CSS -->  
         <link rel="stylesheet" href="<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css>" integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2" crossorigin="anonymous">  
         <title>  
             {% block title %}  
     
             {% endblock %}  
         </title>  
          <link rel="stylesheet" type="text/css" href="{{ url\_for('static',filename='style\_home.css') }}">  
          <style>  
              .logo img{  
                   float: left;  
                   width: 80px;  
                   height:80px;  
               }  
          </style>  
      </head>  
      <body>  
       <nav class="navbar navbar-expand-md navbar-dark bg-dark">  
           <div class = "logo">  
               <img src="{{ url\_for('static', filename='images/logo.png')}}">  
          </div>  
         <a class="navbar-brand" href="#"><h1>Spaces</h1></a>  
         <div class="collapse navbar-collapse" id="navbarNav">  
           <ul class="navbar-nav mr-auto">  
               <li class="nav-item active">  
                   <a class="nav-link" href="{{ url\_for('home\_page') }}">Home <span class="sr-only">(current)</span></a>  
               </li>  
               <li class="nav-item">  
                   <a class="nav-link" href="{{ url\_for('search\_page') }}">Search</a>  
               </li>  
           </ul>  
             {% if current\_user.is\_authenticated %}  
               <ul class="navbar-nav">  
                   <li class="nav-item">  
                       <a class="nav-link">Welcome, {{ current\_user.username }}</a>  
                   </li>  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('logout\_page') }}">Logout</a>  
                   </li>  
               </ul>  
             {% else %}  
               <ul class="navbar-nav">  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('login\_page') }}">Login</a>  
                   </li>  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('register\_page') }}">Register</a>  
                   </li>  
               </ul>  
             {% endif %}  
         </div>  
       </nav>  
         {% with messages = get\_flashed\_messages(with\_categories=true) %}  
           {% if messages %}  
              {% for category, message in messages %}  
                 <div class="alert alert-{{ category }}">  
                     <button type="button" class="m1-2 mb-1 close" data-dismiss="alert" aria-label="Close">  
                         <span aria-hidden="true">&times;</span>  
                     </button>  
                     {{ message }}  
                 </div>  
              {% endfor %}  
           {% endif %}  
         {% endwith %}  
         {% block content %}  
     
         {% endblock %}  
         <!-- Future Content here -->  
     
     
     
         <!-- Optional JavaScript -->  
         <!-- jQuery first, then Popper.js, then Bootstrap JS -->  
         <script src='<https://kit.fontawesome.com/a076d05399.js>'></script>  
         <script src="<https://code.jquery.com/jquery-3.5.1.slim.min.js>" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>  
         <script src="<https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js>" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>  
         <script src="<https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js>" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>  
      </body>  
      <style>  
         body {  
         background-color: #212121;  
         color: white  
         }  
      </style>  
   </html>

**2.home.html**

<!doctype html>  
<html lang="en">  
   <head>  
      <!-- Required meta tags -->  
      <meta charset="utf-8">  
      <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">  
      <!-- Bootstrap CSS -->  
      <link rel="stylesheet" href="<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css>" integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2" crossorigin="anonymous">  
      <title>  
          Hostel Seeking Forum  
      </title>  
       <style>  
           .main {  
    width: auto;  
    height: 150px;  
}  
  
  
.logo img{  
    float: left;  
    width: 80px;  
    height:80px;  
}  
.name{  
    color:#d6a99a;  
    font-size: 70px;  
    letter-spacing: 14px;  
    font-family: Comic Sans, Comic Sans MS, cursive;  
}  
  
#tagline{  
    color: #f8dcd2;  
    letter-spacing: 5px;  
    font-size: 18px;  
    font-family: cursive;  
  
}  
  
.text{  
    color: #f4ebe8;  
    font-size: 25px;  
    text-align: center;  
    text-indent: 80px;  
    width: 1400px;  
    margin-left: 10px;  
}  
  
.textans  
{  
     color: #f4ebe8;  
    font-size: 25px;  
    text-align: center;  
    width: 1400px;  
    margin-left : 50px;  
}  
  
  
.picture {  
    text-align: center;  
}  
  
.picture img{  
    border: 1.5px solid #d6a99a;  
}  
  
.para{  
    color: #F8EA8C;  
    position:absolute;  
    margin-top: -350px;  
    text-align: center;  
    width: 800px;  
    margin-left: 350px;  
    font-size: 25px;  
    text-shadow: 2px 2px 3px black;  
}  
  
.alignment{  
    text-align: center;  
    font-size: 33px;  
    color: #24a0ed;  
}  
  
.align{  
    text-align: center;  
    font-size: 24px;  
    color: white;  
    width: 1000px;  
    margin-left: 300px;  
}  
  
.circular\_images{  
    margin-left: 680px;  
    width : 150px;  
    clip-path : circle();  
}  
  
.footer{  
    background-color: #808080;  
}  
ul{  
    margin-left: 50px;  
}  
       </style>  
   </head>  
   <body>  
    <nav class="navbar navbar-expand-md navbar-dark bg-dark">  
       <div class = "logo">  
            <img src="{{ url\_for('static', filename='images/logo.png')}}">  
       </div>  
        <a class="navbar-brand" href="#"><h1>Spaces</h1></a>  
      <div class="collapse navbar-collapse" id="navbarNav">  
        <ul class="navbar-nav mr-auto">  
            <li class="nav-item active">  
                <a class="nav-link" href="{{ url\_for('home\_page') }}">Home <span class="sr-only">(current)</span></a>  
            </li>  
            <li class="nav-item">  
                <a class="nav-link" href="{{ url\_for('search\_page') }}">Search</a>  
            </li>  
        </ul>  
          {% if current\_user.is\_authenticated %}  
            <ul class="navbar-nav">  
                <li class="nav-item">  
                    <a class="nav-link">Welcome, {{ current\_user.username }}</a>  
                </li>  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('logout\_page') }}">Logout</a>  
                </li>  
            </ul>  
          {% else %}  
            <ul class="navbar-nav">  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('login\_page') }}">Login</a>  
                </li>  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('register\_page') }}">Register</a>  
                </li>  
            </ul>  
          {% endif %}  
      </div>  
    </nav>  
      {% with messages = get\_flashed\_messages(with\_categories=true) %}  
        {% if messages %}  
           {% for category, message in messages %}  
              <div class="alert alert-{{ category }}">  
                  <button type="button" class="m1-2 mb-1 close" data-dismiss="alert" aria-label="Close">  
                      <span aria-hidden="true">&times;</span>  
                  </button>  
                  {{ message }}  
              </div>  
           {% endfor %}  
        {% endif %}  
      {% endwith %}  
  
    <div class="content">  
  
        <br><br><br><br>  
        <section>  
            <div>  
                <p class="text">  
                    Are you new to the campus?  
                </p>  
                <p class="textans">  
                     We'll help you find your ideal space to stay, furnishing you with every necessary feature of hostels & PGs around CUSAT. From basic details like rent to the list of facilities available in your room, we've managed to cover everything you need to know about a "space", to make it your second home at CUSAT.  
                </p>  
            </div>  
            <br><br>  
            <div class="picture">  
              <img src="{{ url\_for('static', filename='images/hostel1.jpg') }}" width="950" height="550" align="middle" alt="hostel\_image" style="border-radius: 15px">  
              <p class="para"><b>As your hostel is one of the major parts of your campus life, we don't want you to miss out on comfort as well as fun!</b></p>  
            </div>  
            <br><br><br><br><br>  
            <div>  
                <div class="circular\_images">  
                    <img src="{{ url\_for('static', filename='images/circle1.jpg')}}" width="300" height="200" alt="hostel\_image">  
                </div>  
                <div>  
                    <p class="alignment"><b>All-inclusive</b></p>  
                    <p class="align">Since we don't want you to miss any option, we let hostel owners to log in and add their hostel info into our database hassle-free. Added hostel information is updated and reflected in the search page without any holdup.  
                    </p>  
                </div>  
                <div class="circular\_images">  
                    <img src="{{ url\_for('static', filename='images/hostel-6.jpg')}}" width="400" height="300" alt="hostel\_image">  
                </div>  
                 <div>  
                    <p class="alignment"><b>Comprehensive</b></p>  
                    <p class="align">We've strived to fill in every needed detail about hostels since it's vital to know that about the place you're gonna stay. Hidden charges like caution, extra fees for facilities, all exposed for you!  
                    </p>  
                </div>  
                <div class="circular\_images">  
                    <img src="{{ url\_for('static', filename='images/background.jpg')}}" width="400" height="300" alt="hostel\_image">  
                </div>  
               <div>  
                    <p class="alignment"><b>From experience</b></p>  
                    <p class="align">As students of CUSAT, we've chosen and collected every input for you that we wished we would've known when we were in your place. We don't want you to get caught in a place you don't like and end up missing solace and recreation.  
                    </p>  
                </div>  
            </div>  
        </section>  
        <br><br>  
        <footer class="footer">  
            <ul style="list-style: none;">  
                <li>Anjali Krishna : [19it018anja@ug.cusat.ac.in](mailto:19it018anja@ug.cusat.ac.in)</li>  
                <li>Aparna Unnikrishnan : [19it024apar@ug.cusat.ac.in](mailto:19it024apar@ug.cusat.ac.in)</li>  
                <li>Ashana : [19it026asha@ug.cusat.ac.in](mailto:19it026asha@ug.cusat.ac.in)</li>  
              </ul>  
            <center><p>Copyright @Spaces</p></center>  
        </footer>  
    </div>  
      <!-- Future Content here -->  
  
  
  
      <!-- Optional JavaScript -->  
      <!-- jQuery first, then Popper.js, then Bootstrap JS -->  
      <script src='<https://kit.fontawesome.com/a076d05399.js>'></script>  
      <script src="<https://code.jquery.com/jquery-3.5.1.slim.min.js>" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>  
      <script src="<https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js>" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>  
      <script src="<https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js>" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>  
   </body>  
   <style>  
      body {  
      background-color: #212121;  
      color: white  
      }  
   </style>  
</html>

1. **login.html**
2. {% extends 'base.html' %}  
   {% block title %}  
       Login Page  
   {% endblock %}  
     
   {% block content %}  
   <body class="text-center">  
       <div class="container">  
           <form method="POST" class="form-signin" style="color:white">  
               {{ form.hidden\_tag() }}  
               <br><br><br>  
               <h1 class="h3 mb-3 font-weight-normal">  
                   Please Login  
               </h1>  
               <br>  
               {{ form.username.label() }}  
               {{ form.username(class="form-control", placeholder="User Name") }}  
     
               {{ form.password.label() }}  
               {{ form.password(class="form-control", placeholder="Password") }}  
     
               <br>  
     
     
               <div class="checkbox mb-3">  
                  <h6>Do not have an account?</h6>  
                  <a class="btn btn-sm btn-secondary" href="{{ url\_for('register\_page') }}">Register</a>  
               </div>  
     
               {{ form.submit(class="btn btn-lg btn-block btn-primary") }}  
     
           </form>  
       </div>  
   </body>  
   {% endblock %}

**4.register.html**

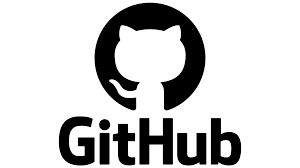
{% extends 'base.html' %}  
{% block title %}  
    Register Page  
{% endblock %}  
  
{% block content %}  
<body class="text-center">  
    <div class="container">  
        <form method="POST" class="form-register" style="color:white">  
            {{ form.hidden\_tag() }}  
            <br><br><br>  
            <h1 class="h3 mb-3 font-weight-normal">  
                Please Create your Account  
            </h1>  
            <br>  
            {{ form.username.label() }}  
            {{ form.username(class="form-control", placeholder="User Name") }}  
  
            {{ form.email\_address.label() }}  
            {{ form.email\_address(class="form-control", placeholder="Email Address") }}  
  
            {{ form.password1.label() }}  
            {{ form.password1(class="form-control", placeholder="Password") }}  
  
            {{ form.password2.label() }}  
            {{ form.password2(class="form-control", placeholder="Confirm Password") }}  
  
            <br>  
  
            <div class="checkbox mb-3">  
               <h6>Already have an account?</h6>  
               <a class="btn btn-sm btn-secondary" href="{{ url\_for('login\_page') }}">Login</a>  
            </div>  
  
            {{ form.submit(class="btn btn-lg btn-block btn-primary") }}  
  
        </form>  
    </div>  
</body>  
{% endblock %}

1. **registerhostel.html**
2. {% extends 'base.html' %}  
   {% block title %}  
       register hostel Page  
   {% endblock %}  
     
   {% block content %}  
   <body class="text-center">  
       <div class="container">  
           <form method="POST" class="form-signin" style="color:white">  
               {{ form.hidden\_tag() }}  
               <br><br><br>  
               <h1 class="h3 mb-3 font-weight-normal" id="name">  
                   Enter Hostel Details  
               </h1>  
               <hr>  
               <br>  
           <div class="pic">  
               <!--<img src="{{ url\_for('static', filename='images/hostel-6.jpg')}}" width="600" height="400" alt="hostel\_image">-->  
               <table align="center">  
                   <tr>  
                       <td class="label">{{ form.hs\_name.label() }}</td>  
                       <td> :&nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.hs\_name(class="form-control", placeholder="Hostel Name") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.hs\_address.label() }}</td>  
                       <td> :&nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.hs\_address(class="form-control", placeholder="Address") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.hs\_contact.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.hs\_contact(class="form-control", placeholder="Contact Number") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.hs\_rent.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.hs\_rent(class="form-control", placeholder="Rent") }}</td>  
                   <tr>  
                       <td class="label">{{ form.hs\_description.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.hs\_description(class="form-control", placeholder="Description") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.rooms.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.rooms(class="form-control", placeholder="Rooms") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.caution.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.caution(class="form-control", placeholder="Caution Deposit") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.curfew.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.curfew(class="form-control", placeholder="Curfew") }}</td>  
                   </tr>  
                   <tr>  
                       <td class="label">{{ form.maps\_link.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.maps\_link(class="form-control", placeholder="Maps Link") }}</td>  
                   <tr>  
                       <td class="label">{{ form.type.label() }}</td>  
                       <td> : &nbsp;&nbsp;&nbsp; </td>  
                       <td class="data">{{ form.type(class="form-control", placeholder="Girls or Boys") }}</td>  
                   </tr>  
               </table>  
           </div>  
               <br><br><br>  
               <div class="submit">  
                   {{ form.submit(class="btn btn-lg btn-block btn-primary") }}  
               </div>  
               <br><br><br><br>  
           </form>  
       </div>  
   </body>  
   <style>  
       .pic img{  
       border-radius: 20px;  
       }  
     
       #name{  
       font-size: 45px;  
       color: #007bff;  
       }  
     
       .label{  
       font-size: 25px;  
       color: white;  
       text-align: left;  
       }  
     
       .data{  
       width: 500px;  
       }  
       .submit{  
       width: 250px;  
       margin-left: 400px;  
       }  
   </style>  
   {% endblock %}

**6.search.html**

<!doctype html>  
<html lang="en">  
   <head>  
      <!-- Required meta tags -->  
      <meta charset="utf-8">  
      <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">  
      <!-- Bootstrap CSS -->  
      <link rel="stylesheet" href="<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css>" integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2" crossorigin="anonymous">  
      <title>  
          Search Page  
      </title>  
       <link rel="stylesheet" href="<https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css>">  
       <style>  
           .logo img{  
                float: left;  
                width: 80px;  
                height:80px;  
            }  
            .hostel{  
            margin-left:150px;  
            width:75%;  
            background-color: white;  
            border-radius: 20px;  
            }  
  
            #name{  
            font-size: 35px;  
            margin-left: 150px;  
            color: black;  
            }  
  
            .contact{  
            color : black;  
            margin-left: 100px;  
            font-size: 15px;  
            margin-top: -10px  
            }  
  
            .result a{  
            color:#213571 ;  
            }  
  
            .colon{  
            color : black;  
            margin-left: 350px;  
            font-size: 15px;  
            margin-top: -40px;  
            }  
  
            .result{  
            color:#213571 ;  
            margin-left: 450px;  
            font-size: 15px;  
            margin-top: -40px;  
            }  
  
            .member{  
            margin-top:-70px;  
            margin-left:50px;  
            }  
  
            #rent{  
            color : #1a2a5a;  
            margin-left: 800px;  
            margin-top: -50px;  
            font-size: 40px;  
            }  
  
            #type{  
            margin-left: 150px;  
            color: rgb(0,128,0);  
            margin-top:-75px;  
            font-size: 35px;  
            }  
  
            .des{  
            font-size: 20px;  
            width: 75%;  
            margin-left: 150px;  
            color: #8b4513 ;  
            }  
                #link{  
                margin-left: 820px;  
                margin-top:50px;  
            }  
  
  
       </style>  
   </head>  
   <body>  
    <nav class="navbar navbar-expand-md navbar-dark bg-dark">  
        <div class = "logo">  
            <img src="{{ url\_for('static', filename='images/logo.png')}}">  
       </div>  
      <a class="navbar-brand" ><h1>Spaces</h1></a>  
      <div class="collapse navbar-collapse" id="navbarNav">  
        <ul class="navbar-nav mr-auto">  
            <li class="nav-item active">  
                <a class="nav-link" href="{{ url\_for('home\_page') }}">Home <span class="sr-only">(current)</span></a>  
            </li>  
            <li class="nav-item">  
                <a class="nav-link" href="{{ url\_for('search\_page') }}">Search</a>  
            </li>  
        </ul>  
          {% if current\_user.is\_authenticated %}  
  
            <form class="form-inline" method="GET" action="{{url\_for('searches')}}">  
                <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search" name="q">  
                <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>  
            </form>  
            <ul class="navbar-nav">  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('add\_hostel') }}">Add Hostel</a>  
                </li>  
                <li class="nav-item">  
                    <a class="nav-link">Welcome, {{ current\_user.username }}</a>  
                </li>  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('logout\_page') }}">Logout</a>  
                </li>  
            </ul>  
          {% else %}  
            <ul class="navbar-nav">  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('login\_page') }}">Login</a>  
                </li>  
                <li class="nav-item">  
                    <a class="nav-link" href="{{ url\_for('register\_page') }}">Register</a>  
                </li>  
            </ul>  
          {% endif %}  
      </div>  
    </nav>  
      {% with messages = get\_flashed\_messages(with\_categories=true) %}  
        {% if messages %}  
           {% for category, message in messages %}  
              <div class="alert alert-{{ category }}">  
                  <button type="button" class="m1-2 mb-1 close" data-dismiss="alert" aria-label="Close">  
                      <span aria-hidden="true">&times;</span>  
                  </button>  
                  {{ message }}  
              </div>  
           {% endfor %}  
        {% endif %}  
      {% endwith %}  
        {% for hostel in hostels %}  
            <br><br>  
            <div class="hostel">  
                <p id="name">{{ hostel.hs\_name }}</p>  
  
                    <p id="rent">₹{{ hostel.hs\_rent }}</p>  
                    <br>  
                    <p id="type">[ For {{ hostel.type }} ]</p>  
                    <div id="link">  
                        <a href="{{ hostel.maps\_link }}"><button type="button" class="btn btn-primary">Map Link</button></a>  
                    </div>  
                    <div class="member">  
                        <p class="contact">Contact Us</p><p class="colon"> : </p><p class="result">{{ hostel.hs\_contact }}</p>  
                        <p class="contact">Caution Deposit</p><p class="colon"> : </p><p class="result">₹{{ hostel.caution }}</p>  
                        <p class="contact">Curfew</p><p class="colon"> : </p><p class="result">{{ hostel.curfew }}</p>  
                        <p class="contact">Rooms Available</p><p class="colon"> : </p><p class="result">{{ hostel.rooms }}</p>  
                        <p class="contact">Addess</p><p class="colon"> : </p><p class="result">{{ hostel.hs\_address }}</p>  
                    </div>  
                    <div class="des">  
                        <h6>{{ hostel.hs\_description }}</h6>  
                    </div>  
                    <br>  
            </div>  
            {% endfor %}  
  
  
      <!-- Future Content here -->  
  
  
  
      <!-- Optional JavaScript -->  
      <!-- jQuery first, then Popper.js, then Bootstrap JS -->  
      <script src='<https://kit.fontawesome.com/a076d05399.js>'></script>  
      <script src="<https://code.jquery.com/jquery-3.5.1.slim.min.js>" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>  
      <script src="<https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js>" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>  
      <script src="<https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js>" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>  
   </body>  
   <style>  
      body {  
      background-color: #212121;  
      color: white  
      }  
   </style>  
</html>

1. **search\_results.html**
2. <!doctype html>  
   <html lang="en">  
      <head>  
         <!-- Required meta tags -->  
         <meta charset="utf-8">  
         <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">  
         <!-- Bootstrap CSS -->  
         <link rel="stylesheet" href="<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css>" integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2" crossorigin="anonymous">  
         <title>  
             Search Page  
         </title>  
          <link rel="stylesheet" href="<https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css>">  
          <style>  
              .logo img{  
                   float: left;  
                   width: 80px;  
                   height:80px;  
               }  
               .hostel{  
               margin-left:150px;  
               width:75%;  
               background-color: white;  
               border-radius: 20px;  
               }  
     
               #name{  
               font-size: 35px;  
               margin-left: 150px;  
               color: black;  
               }  
     
               .contact{  
               color : black;  
               margin-left: 100px;  
               font-size: 15px;  
               margin-top: -10px  
               }  
     
               .result a{  
               color:#213571 ;  
               }  
     
               .colon{  
               color : black;  
               margin-left: 350px;  
               font-size: 15px;  
               margin-top: -40px;  
               }  
     
               .result{  
               color:#213571 ;  
               margin-left: 450px;  
               font-size: 15px;  
               margin-top: -40px;  
               }  
     
               .member{  
               margin-top:-70px;  
               margin-left:50px;  
               }  
     
               #rent{  
               color : #1a2a5a;  
               margin-left: 800px;  
               margin-top: -50px;  
               font-size: 40px;  
               }  
     
               #type{  
               margin-left: 150px;  
               color: rgb(0,128,0);  
               margin-top:-75px;  
               font-size: 35px;  
               }  
     
               .des{  
               font-size: 20px;  
               width: 75%;  
               margin-left: 150px;  
               color: #8b4513 ;  
               }  
                   #link{  
                   margin-left: 820px;  
                   margin-top:50px;  
               }  
     
     
          </style>  
      </head>  
      <body>  
       <nav class="navbar navbar-expand-md navbar-dark bg-dark">  
           <div class = "logo">  
               <img src="{{ url\_for('static', filename='images/logo.png')}}">  
          </div>  
         <a class="navbar-brand" ><h1>Spaces</h1></a>  
         <div class="collapse navbar-collapse" id="navbarNav">  
           <ul class="navbar-nav mr-auto">  
               <li class="nav-item active">  
                   <a class="nav-link" href="{{ url\_for('home\_page') }}">Home <span class="sr-only">(current)</span></a>  
               </li>  
               <li class="nav-item">  
                   <a class="nav-link" href="{{ url\_for('search\_page') }}">Search</a>  
               </li>  
           </ul>  
             {% if current\_user.is\_authenticated %}  
     
               <form class="form-inline" method="GET">  
                   <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search" name="q">  
                   <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>  
               </form>  
               <ul class="navbar-nav">  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('add\_hostel') }}">Add Hostel</a>  
                   </li>  
                   <li class="nav-item">  
                       <a class="nav-link">Welcome, {{ current\_user.username }}</a>  
                   </li>  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('logout\_page') }}">Logout</a>  
                   </li>  
               </ul>  
             {% else %}  
               <ul class="navbar-nav">  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('login\_page') }}">Login</a>  
                   </li>  
                   <li class="nav-item">  
                       <a class="nav-link" href="{{ url\_for('register\_page') }}">Register</a>  
                   </li>  
               </ul>  
             {% endif %}  
         </div>  
       </nav>  
         {% with messages = get\_flashed\_messages(with\_categories=true) %}  
           {% if messages %}  
              {% for category, message in messages %}  
                 <div class="alert alert-{{ category }}">  
                     <button type="button" class="m1-2 mb-1 close" data-dismiss="alert" aria-label="Close">  
                         <span aria-hidden="true">&times;</span>  
                     </button>  
                     {{ message }}  
                 </div>  
              {% endfor %}  
           {% endif %}  
         {% endwith %}  
           {% for hostel in hostels %}  
               <br><br>  
               <div class="hostel">  
                   <p id="name">{{ hostel.hs\_name }}</p>  
     
                       <p id="rent">₹{{ hostel.hs\_rent }}</p>  
                       <br>  
                       <p id="type">[ For {{ hostel.type }} ]</p>  
                       <div id="link">  
                           <a href="{{ hostel.maps\_link }}"><button type="button" class="btn btn-primary">Map Link</button></a>  
                       </div>  
                       <div class="member">  
                           <p class="contact">Contact Us</p><p class="colon"> : </p><p class="result">{{ hostel.hs\_contact }}</p>  
                           <p class="contact">Caution Deposit</p><p class="colon"> : </p><p class="result">₹{{ hostel.caution }}</p>  
                           <p class="contact">Curfew</p><p class="colon"> : </p><p class="result">{{ hostel.curfew }}</p>  
                           <p class="contact">Rooms Available</p><p class="colon"> : </p><p class="result">{{ hostel.rooms }}</p>  
                           <p class="contact">Addess</p><p class="colon"> : </p><p class="result">{{ hostel.hs\_address }}</p>  
                       </div>  
                       <br>  
                       <div class="des">  
                           <h6>{{ hostel.hs\_description }}</h6>  
                       </div>  
                       <br>  
               </div>  
               {% endfor %}  
     
     
         <!-- Future Content here -->  
     
     
     
         <!-- Optional JavaScript -->  
         <!-- jQuery first, then Popper.js, then Bootstrap JS -->  
         <script src='<https://kit.fontawesome.com/a076d05399.js>'></script>  
         <script src="<https://code.jquery.com/jquery-3.5.1.slim.min.js>" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>  
         <script src="<https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js>" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>  
         <script src="<https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js>" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>  
      </body>  
      <style>  
         body {  
         background-color: #212121;  
         color: white  
         }  
      </style>  
   </html>





See the whole source code in GitHub : <https://github.com/ashana1647/Spaces-Hostel_Seeking_Forum->

