PROPERTY DEALING WEB APP

DISSERTATION

SUBMITTED IN PARTIAL FULLFILMENT OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF SCIENCE (2018-21)

IN

COMPUTER SCIENCE

Guided by:

DR. RAKHI GARG
(Associate Professor)
AND
DR. SARVESH PANDEY
(Assistant Professor)

Submitted by:

SHIVANI SINGH
Roll no: 18229CMP003
AND
AKRITI SINGH
Roll no: 18229CMP008

DEPARTMENT OF COMPUTER SCIENCE

MAHILA MAHA VIDYALAYA

BANARAS HINDU UNIVERSITY

CERTIFICATE

THIS IS TO CERTIFY THAT THE PROJECT WORK WHICH IS ENTITLED AS “PROPERTY DEALING WEB APP” IS JOINTLY SUBMITTED BY AKRITI SINGH AND SHIVANI SINGH AS A PART OF B.Sc. COMPUTER SCIENCE HONS (2018-2021) OF BANARAS HINDU UNIVERSITY, IS A BONAFIDE WORK CARRIED OUT UNDER DIRECT SUPERVISION AND GUIDANCE.

SUPERVISED BY

DR. RAKHI GARG
COMPUTER SCIENCE DEPARTMENT
MAHILA MAHAVIDYALAYA
BANARAS HINDU UNIVERSITY

DR. SARVESH PANDEY
COMPUTER SCIENCE DEPARTMENT
MAHILA MAHAVIDYALAYA
BANARAS HINDU UNIVERSITY

DECLARATION

WE HEREBY DECLARE THAT THIS PROJECT REPORT THE “PROPERTY DEALING WEB APP” IS SUBMITTED FOR PARTIAL FULLFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE TO THE DEPARTMENT OF COMPUTER SCIENCE IS A RECORD OF ORIGINAL WORKDONE BY US UNDER THE GUIDANCE OF DR. RAKHI GARG AND DR. SARVESH PANDEY.

THE DATA AND INFORMATION GIVEN IN THE REPORT IS BEST OF OUR KNOWLEDGE.

AKRITI SINGH (18229CMP008)

SHIVANI SINGH (18229CMP003)

TABLE OF CONTENTS

EXECUTIVE SUMMARY

‘HOME SWEET HOME’ is a web application designed for people in search of home according to their requirements.

Coming in a new city and meeting several brokers to find a suitable place to live is laborious and time taking work and even if you find a suitable house there’s possibility to have problems with budget.

So here we propose this site ‘HOME SWEET HOME’ which can be used by new buyers, house sellers and for renting purpose. People can come and search for a home in specific location which best matches their requirements. We also have a facility that will predict total amount anyone might have to pay according to their needs, this will give them an idea about actual price of houses and help them to know if the price asked by owner is appreciable or not.

To make things easy for people there will be some best matches according to your need and budget.

CHAPTER-1

INTRODUCTION

Objective

- To provide a GUI for end-user who is looking for buying/ selling/ renting any housing asset.
- Assist user in selecting the most suitable property based on his/her custom requirements through ML model.

Additional Features

- To predict the price of the property so that the user can verify that the price asked by the owner is appreciable or not.
- Suggesting the best property matches if there is no such property which has the exact same features as asked by the user.

Motivation

When someone wants to buy a new house and they don't know market value of properties it's quite difficult for them to know whether the price asked from them is genuine or not so we have built a feature which gives an estimated price they might have to pay according to their requirements. This will help them to check price asked from them is not absurd. It's even possible that sometimes there is no match according to your requirements then in that case we are giving some best matches for you.

Methodology

1. Identifying the prerequisite of project:

- As we know necessity is the mother of inventions, so what let us think over this site should be clear. So, our first step towards this project was problem formulation.
- Deciding the tools to implement this project.

2. Identifying various components and entities present in system:

- Deciding the attributes to be kept in an entity.

3. Identifying all the relation between various entities:

- Identifying the type of relation and deriving a relational mapping between various entities.

4. Creating a database:

5. Using queries to generate desired results and display it on user interface by means of a server – scripting language and mark-up language.

6. Deciding a machine learning algorithm for backend used in predicting prices.

Advantages

1. Easy for people to find best preferred home.
2. Customer will get a predicted price they should pay.

Limitations

1. Have to check multiple times if you want multiple filters for one option.
2. Have to search last seen property again.

CHAPTER-2

SOFTWARE REQUIREMENT AND SPECIFICATIONS

Technology Used

(1) Hardware interface

System Configuration: 4GB RAM

Operating System: 64-bit Windows 10

(2) Software interface

Front-end: HTML, CSS, JavaScript, Bootstrap

Back-end: Flask (python framework), Pyodbc

Tools: GitHub

DBMS: SSMS

HTML and CSS

HTML stands for Hyper Text Markup Language and CSS stands for Cascading Style Sheets. These are crucial technologies for creating web pages.HTML supplies the structure of the page, and CSS the layout, for diversity of devices. Together with scripting and graphics, HTML and CSS are the fundamental of building Web applications and Web pages.

JavaScript

JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive. Where HTML and CSS are languages that give structure and style to web pages, JavaScript gives web pages interactive elements that engage a user.

Bootstrap

Bootstrap is the most popular open-source front-end framework. Bootstrap is a collection of CSS classes and JavaScript function and it is used for responsive design and building responsive, mobile-first site and application. It generally works on a grid system for creating page layout with the help of rows and columns and it supports all browsers for creating responsive websites.

Flask

Flask is a micro web framework written in python. It is called 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 library provide common functions. Flask supports extensions that can add application features as if they were implemented in flask.

Pyodbc

Pyodbc is open-source python module that makes accessing ODBC databases simple (ODBC-OPEN DATABASE CONNECTIVITY). ODBC is an interface that makes it possible for application to access data from variety of database management systems.

GitHub

GitHub is code hosting platform for collaboration and version control. It provides access control, source code management and several collaborations.

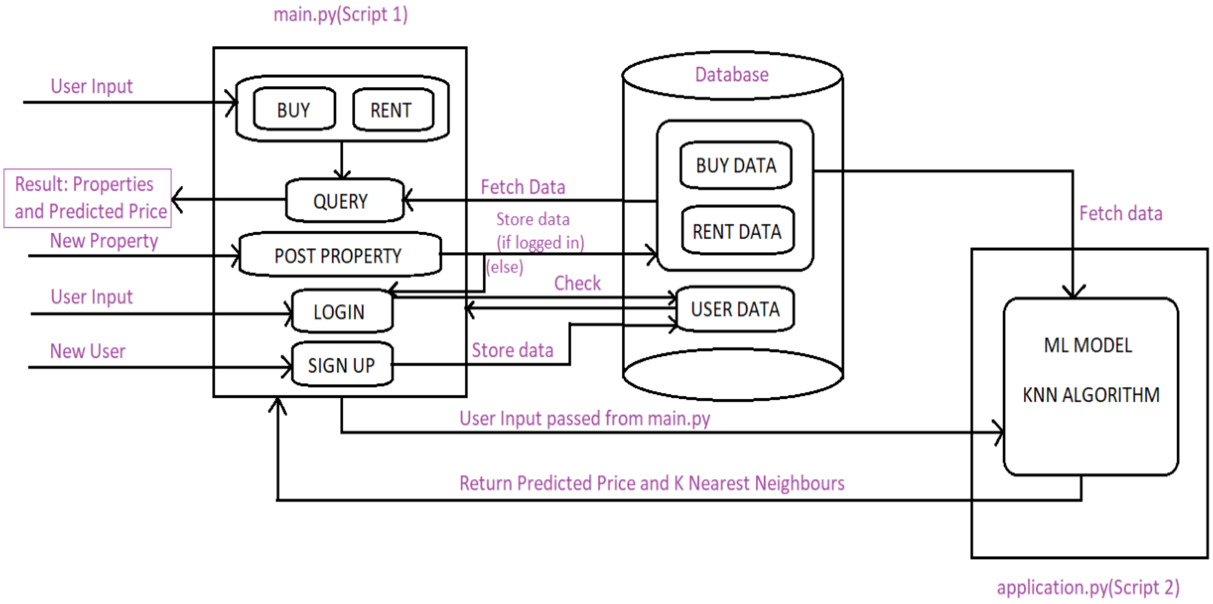
SSMS

SQL Server Management Studio (SSMS) is an integrated environment for managing any SQL infrastructure.

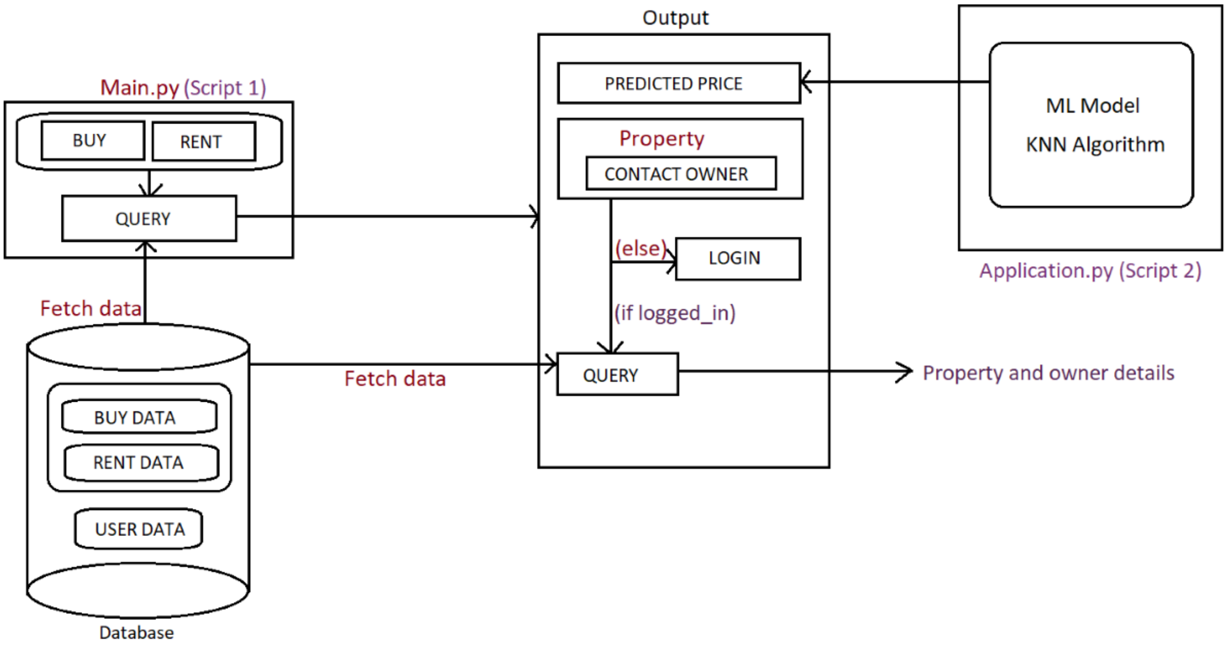
CHAPTER-3

DESIGN OF PROPOSED SYSTEM

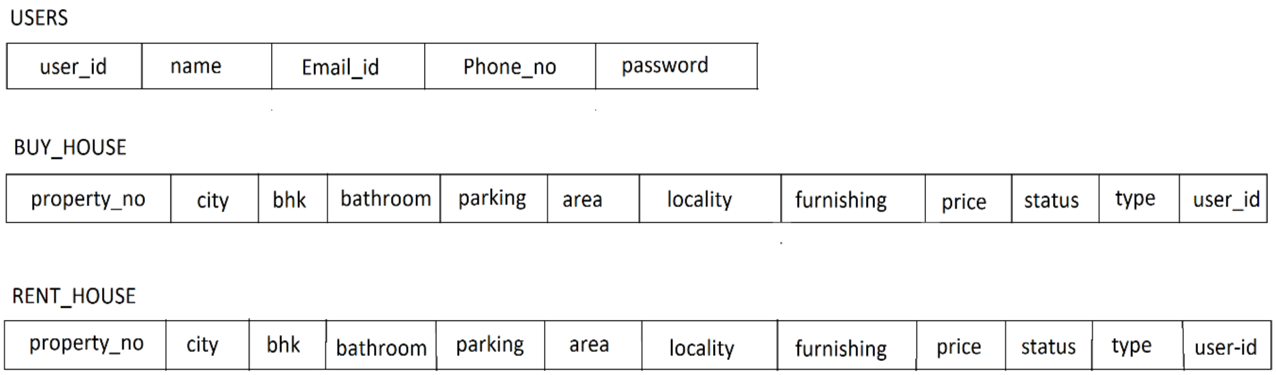
OVERVIEW OF PROJECT



EXECUTION OF QUERY



ER-Schema



USERS

user_id	name	Email_id	Phone_no	password

BUY_HOUSE

property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user_id

RENT_HOUSE

property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user-id

Primary keys:

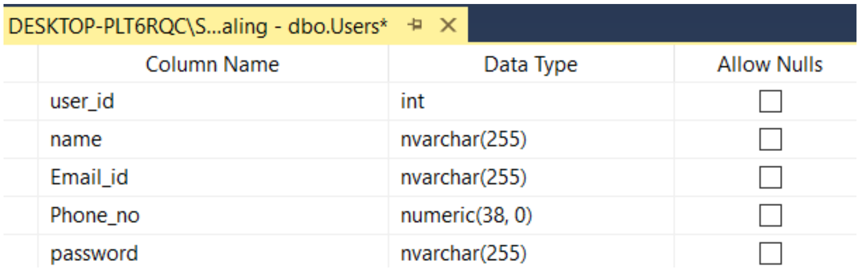
1. Users table: uesr_id
2. Buy_house table: property_no
3. Rent_house table: property_no

Foreign keys:

1. Buy_house table: uesr_id
2. Rent_house table: uesr_id

THE SCHEMA STRUCTURE: -

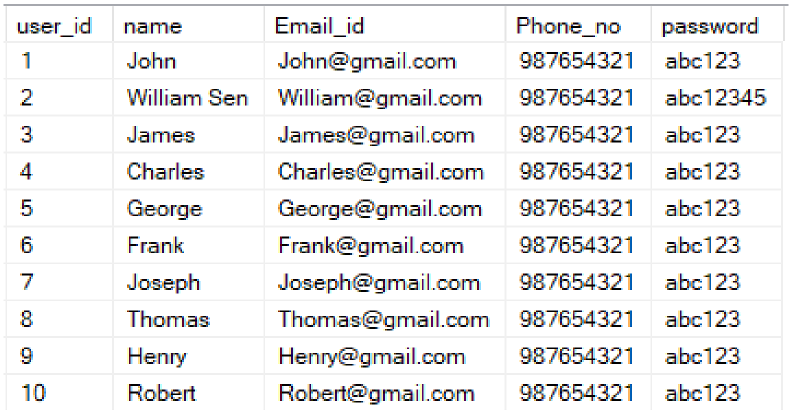
1) Users



DESKTOP-PLT6RQC\S...aling - dbo.Users*

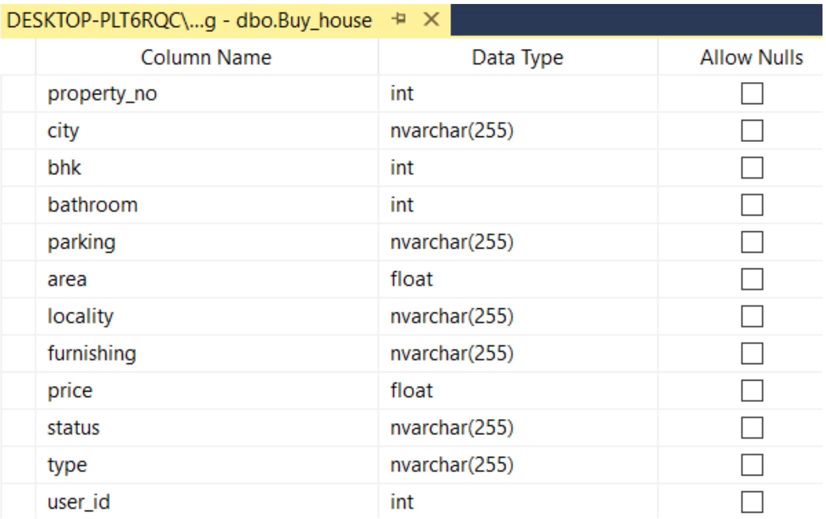
Column Name	Data Type	Allow Nulls
user_id	int	☐
name	nvarchar(255)	☐
Email_id	nvarchar(255)	☐
Phone_no	numeric(38, 0)	☐
password	nvarchar(255)	☐

About Users



user_id	name	Email_id	Phone_no	password
1	John	John@gmail.com	987654321	abc123
2	William Sen	William@gmail.com	987654321	abc12345
3	James	James@gmail.com	987654321	abc123
4	Charles	Charles@gmail.com	987654321	abc123
5	George	George@gmail.com	987654321	abc123
6	Frank	Frank@gmail.com	987654321	abc123
7	Joseph	Joseph@gmail.com	987654321	abc123
8	Thomas	Thomas@gmail.com	987654321	abc123
9	Henry	Henry@gmail.com	987654321	abc123
10	Robert	Robert@gmail.com	987654321	abc123

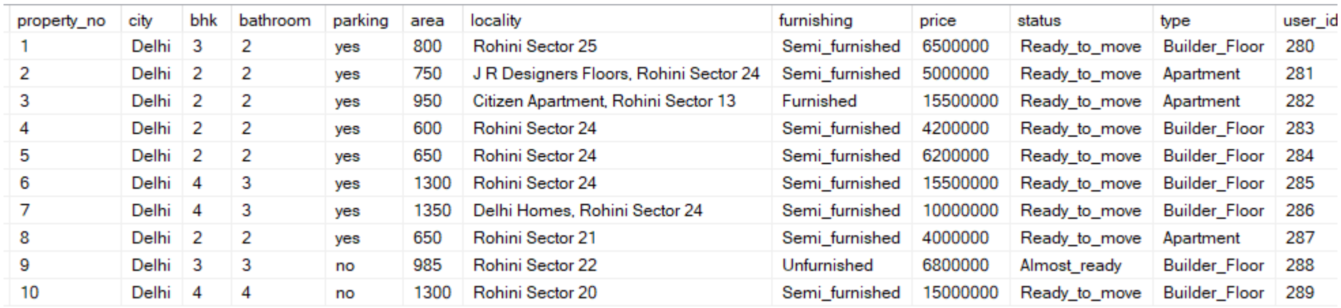
2) Buy_house



DESKTOP-PLT6RQC\...g - dbo.Buy_house

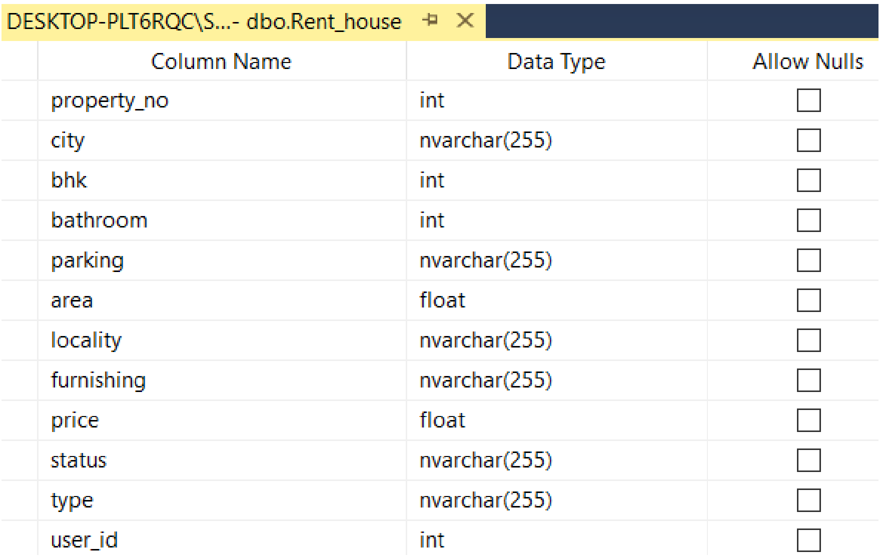
Column Name	Data Type	Allow Nulls
property_no	int	☐
city	nvarchar(255)	☐
bhk	int	☐
bathroom	int	☐
parking	nvarchar(255)	☐
area	float	☐
locality	nvarchar(255)	☐
furnishing	nvarchar(255)	☐
price	float	☐
status	nvarchar(255)	☐
type	nvarchar(255)	☐
user_id	int	☐

About Buy_house



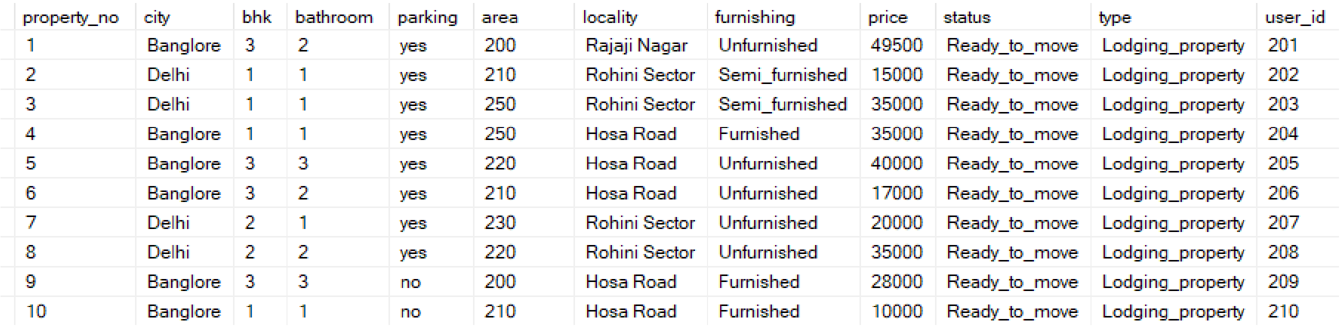
property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user_id
1	Delhi	3	2	yes	800	Rohini Sector 25	Semi_furnished	6500000	Ready_to_move	Builder_Floor	280
2	Delhi	2	2	yes	750	J R Designers Floors, Rohini Sector 24	Semi_furnished	5000000	Ready_to_move	Apartment	281
3	Delhi	2	2	yes	950	Citizen Apartment, Rohini Sector 13	Furnished	15500000	Ready_to_move	Apartment	282
4	Delhi	2	2	yes	600	Rohini Sector 24	Semi_furnished	4200000	Ready_to_move	Builder_Floor	283
5	Delhi	2	2	yes	650	Rohini Sector 24	Semi_furnished	6200000	Ready_to_move	Builder_Floor	284
6	Delhi	4	3	yes	1300	Rohini Sector 24	Semi_furnished	15500000	Ready_to_move	Builder_Floor	285
7	Delhi	4	3	yes	1350	Delhi Homes, Rohini Sector 24	Semi_furnished	10000000	Ready_to_move	Builder_Floor	286
8	Delhi	2	2	yes	650	Rohini Sector 21	Semi_furnished	4000000	Ready_to_move	Apartment	287
9	Delhi	3	3	no	985	Rohini Sector 22	Unfurnished	6800000	Almost_ready	Builder_Floor	288
10	Delhi	4	4	no	1300	Rohini Sector 20	Semi_furnished	15000000	Ready_to_move	Builder_Floor	289

3) Rent_house



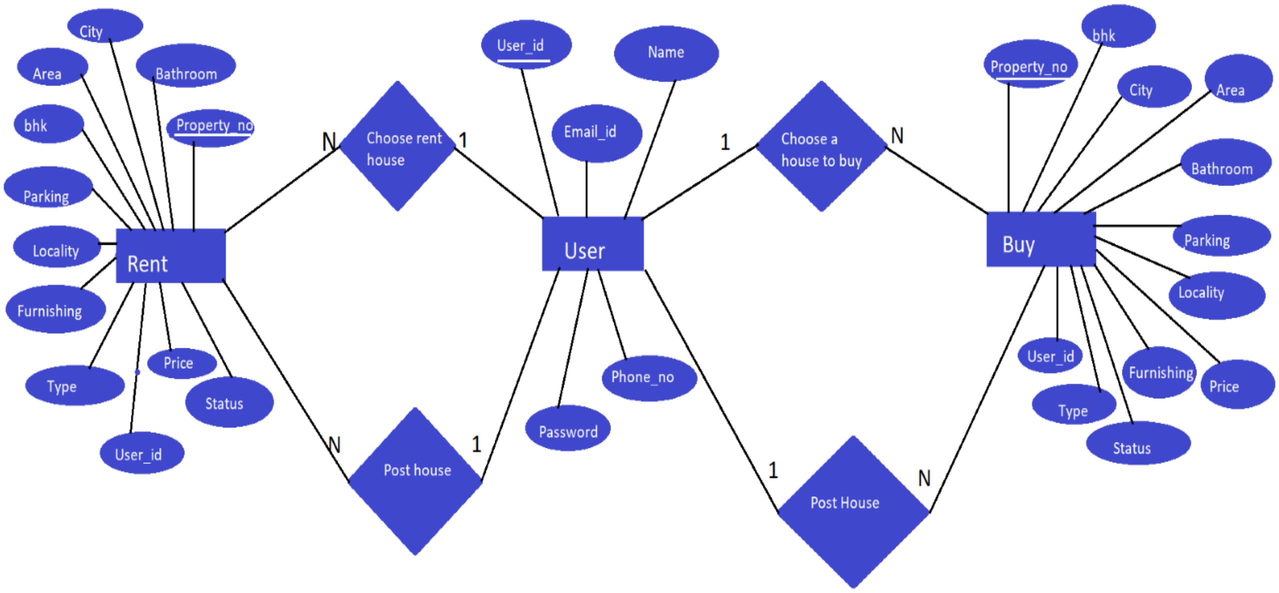
Column Name	Data Type	Allow Nulls
property_no	int	□
city	nvarchar(255)	□
bhk	int	□
bathroom	int	□
parking	nvarchar(255)	□
area	float	□
locality	nvarchar(255)	□
furnishing	nvarchar(255)	□
price	float	□
status	nvarchar(255)	□
type	nvarchar(255)	□
user_id	int	□

About Rent_house



property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user_id
1	Banglore	3	2	yes	200	Rajaji Nagar	Unfurnished	49500	Ready_to_move	Lodging_property	201
2	Delhi	1	1	yes	210	Rohini Sector	Semi_furnished	15000	Ready_to_move	Lodging_property	202
3	Delhi	1	1	yes	250	Rohini Sector	Semi_furnished	35000	Ready_to_move	Lodging_property	203
4	Banglore	1	1	yes	250	Hosa Road	Furnished	35000	Ready_to_move	Lodging_property	204
5	Banglore	3	3	yes	220	Hosa Road	Unfurnished	40000	Ready_to_move	Lodging_property	205
6	Banglore	3	2	yes	210	Hosa Road	Unfurnished	17000	Ready_to_move	Lodging_property	206
7	Delhi	2	1	yes	230	Rohini Sector	Unfurnished	20000	Ready_to_move	Lodging_property	207
8	Delhi	2	2	yes	220	Rohini Sector	Unfurnished	35000	Ready_to_move	Lodging_property	208
9	Banglore	3	3	no	200	Hosa Road	Furnished	28000	Ready_to_move	Lodging_property	209
10	Banglore	1	1	no	210	Hosa Road	Furnished	10000	Ready_to_move	Lodging_property	210

ER-Diagram



Requirements for project:

1. Flask
2. NumPy
3. Pandas
4. Scikit-learn
5. Pyodbc

CHAPTER-4

WORKING OF PROJECT

FLASK

We have used flask framework to develop this property dealing web application. This web frameworks provide routing technique so that user can remember the URLs. It is useful to access the web page directly without navigating from the home page. It is done through the following route () decorator, to bind the URL to a function.

ROUTES IN MAIN.PY SCRIPT (FRONT-END)

- ❑ **"/ "and "/rent"** - These URLs are associated with the buy () and rent () function which is the home page of our website where user can search their required properties.

- ❑ **"/signup"** - This URL is associated with the signup () function which will render the signup.html template where user can enter his/her details to join as a user.

❑ **"/login"** - This URL is associated with the login () function which will render the login.html template where user can enter his/her details which is further verified from the database. If the user has already signed up than a secret key is generated and the user successfully get logged in in that session.

❑ **"/logout"** - This URL will release the session variable and the user will be redirected to the home page.

❑ **"/post" -** This URL is associated with the post () function which will render the post.html template where user can enter the details of the property, he/she wanted to post.

❑ **"/profile" -** This URL is associated with the profile () function which will render the profile.html template where user can edit his/her details.

❑ **"/user_saleprop" and "/user_rentprop" -** This URL is associated with the user_saleprop () and user_rentprop () function which will render the userprop.html template where user can view the properties he posted for sale or rent. He can also edit the details of the property or delete it.

❑ **"/edit_prop/<pro_for>/<user_id>/<prop_no>" -** This is a dynamic URL associated with the editprop () function which will render the editprop.html template where user can edit the details of his/her property.

❑ **"/delete_prop/<pro_for>/<prop_no>" -** This is a dynamic URL which will delete the property and will redirect the user to "/user_saleprop".

❑ **"/buy_own_pro" -** This URL is associated with the buy_own_pro () function which will render the cardview.html template where user can view the properties for sale with a particular filter that it is an owner property. (There are more such routes available which will filter the properties with some particular conditions.)

❑ **"/propview/<pro_for>/<user_id>/<prop_no>" -** This is a dynamic URL associated with the propview () function which will render the propview.html template when the user is logged_in otherwise redirect the user to "/login". Here user can view the details of property and its owner.

ML MODEL

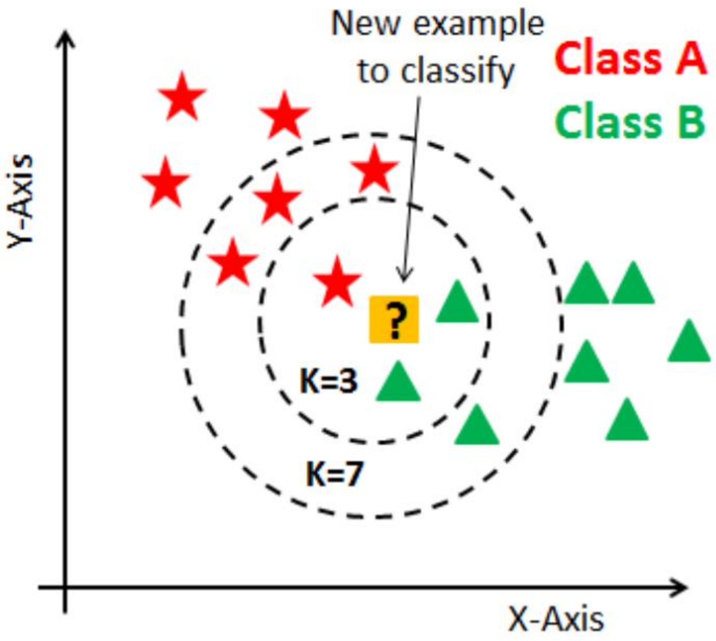
KNN ALGORITHM (K Nearest Neighbour)

In backend we are using KNN algorithm. It is a supervised algorithm and used to solve two types of problems:

1) Classification problem

2) Regression problem.

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm.



Steps to solve a KNN problem:

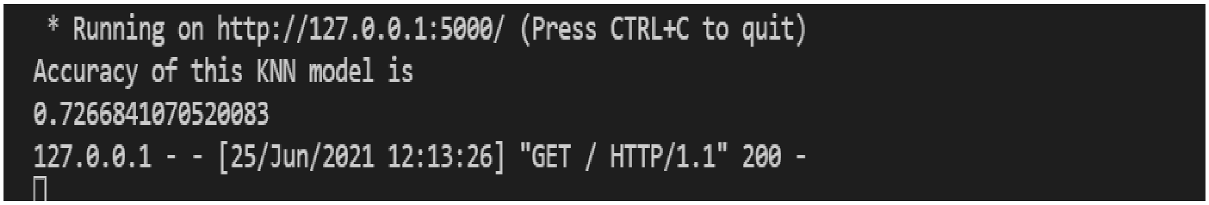
1. Load data set.
2. Split data in training set and testing set.
3. Choose a value of k that best suits your training

   data. (k is total number of neighbours)

4. For each data point find distance between this new data point and already stored data point.
5. Sort the collection of distances from smallest to largest then choose first k elements for which distance is minimum.
6. Get values of selected k entries.
7. If regression problem, return mean of k values.
8. If classification problem, return mode of k values.

For predicting price, we are using KNN to solve regression type of problem.

Accuracy of our KNN model is 0.72.



```
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
Accuracy of this KNN model is
0.7266841070520083
127.0.0.1 - - [25/Jun/2021 12:13:26] "GET / HTTP/1.1" 200 -
```

Working in Back-end

To implement KNN model first we have to import some modules:

1) Flask
2) NumPy
3) Pandas
4) Json
5) From scikit-learn

import-

```
from sklearn.impute import SimpleImputer

from sklearn.model_selection import train_test_split

from sklearn.neighbors import KNeighborsRegressor

from sklearn.metrics import mean_squared_error

from math import sqrt

from sklearn import neighbors
```

APPLICATION.PY SCRIPT (BACK-END)

Working of functions present in backend

1. **def index()**

- Fetched data from database.
- Dropped columns that are not required.
- Dealt with categorical values.
- Divided data in independent and dependent variables.
- Handled missing data.
- Split data in training data and testing set.

2. **def drop()**

- This function drops unrequired columns.

3. **def Best_value_of_k()**

- return best value of k according to data set for which error between predicted price from model and actual price is minimum.

4. **def input_data()**

- This function stores user input and return it to neighbouring_prices()so that it can predict k nearest neighbours.

5. **def neighbouring_prices()**

- Use user input from input_data () and return index value of k properties closest to new data point and return these values to index().

CHAPTER-5

IMPLEMENTATION AND OUTPUT

The home page of our web application “HOME SWEET HOME” consist of a navbar having 4 items named as Home, Post Property, Login and Sign Up. We also have a form to enter the details of the property the user wants to search.

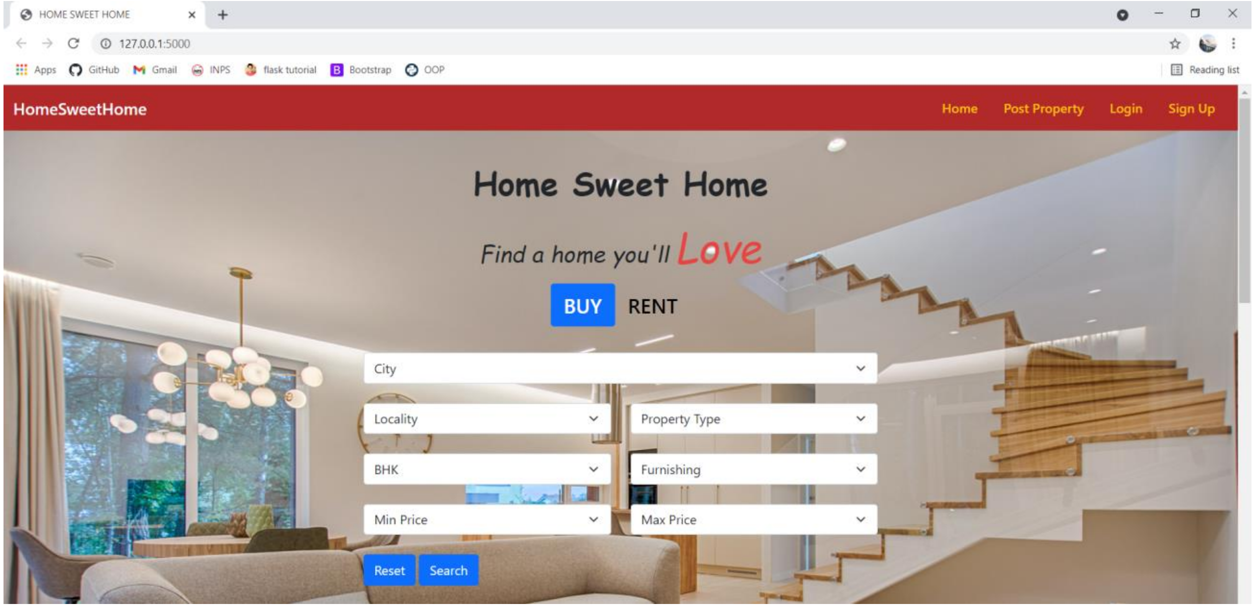
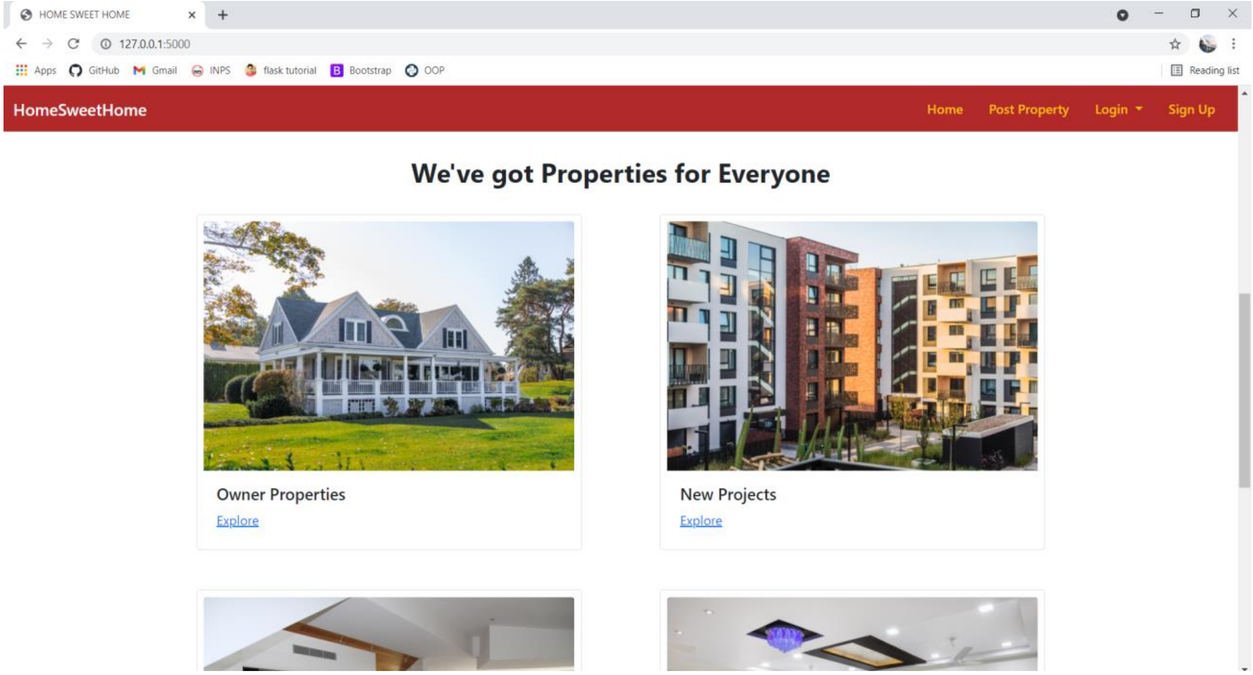


Figure 1. Home page (Buy section)

After that we have some cards which display properties according to particular filters such as ‘Owner Properties’, ‘New Projects’,

‘Ready to move-in’, and ‘Furnished Properties’.



We've got Properties for Everyone

Owner Properties

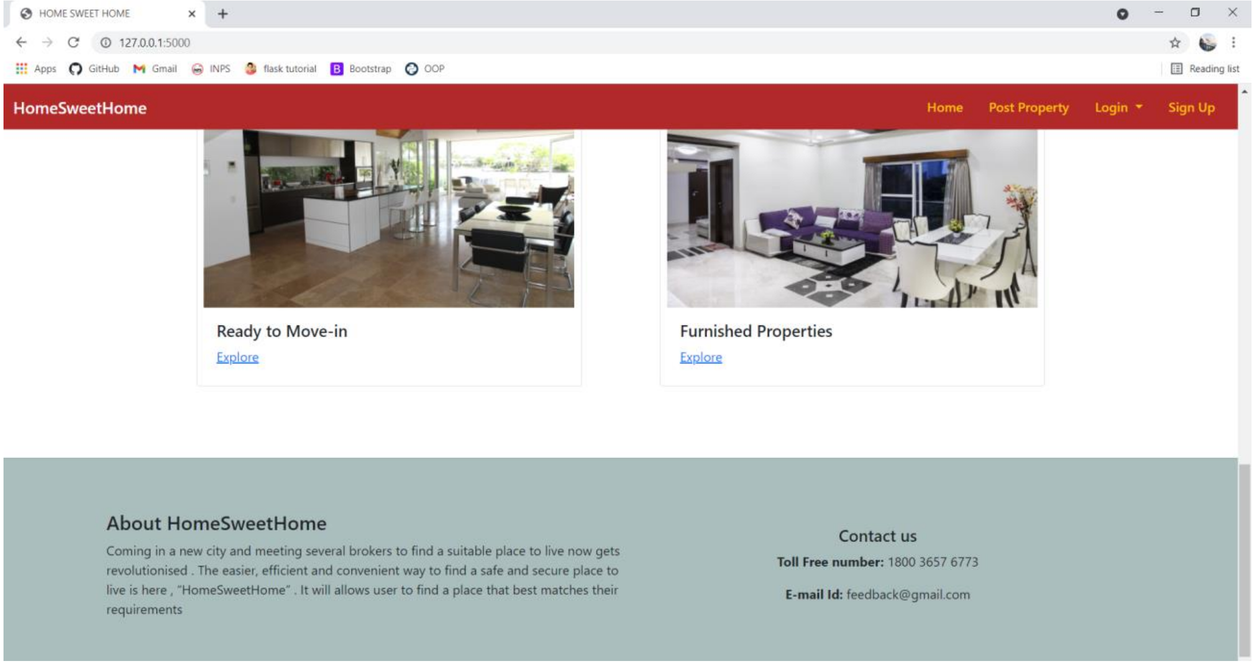
Explore

New Projects

Explore

Figure 2. Cards (Buy Section)

At the end we have some details about our application and also our contact details.



About HomeSweetHome

Coming in a new city and meeting several brokers to find a suitable place to live now gets revolutionised . The easier, efficient and convenient way to find a safe and secure place to live is here , "HomeSweetHome" . It will allows user to find a place that best matches their requirements

Contact us

Toll Free number: 1800 3657 6773

E-mail Id: feedback@gmail.com

Figure 3. Footer (Buy Section)

Our home page consists of 2 sections, one for buy and another for rent. Here is the rent section.

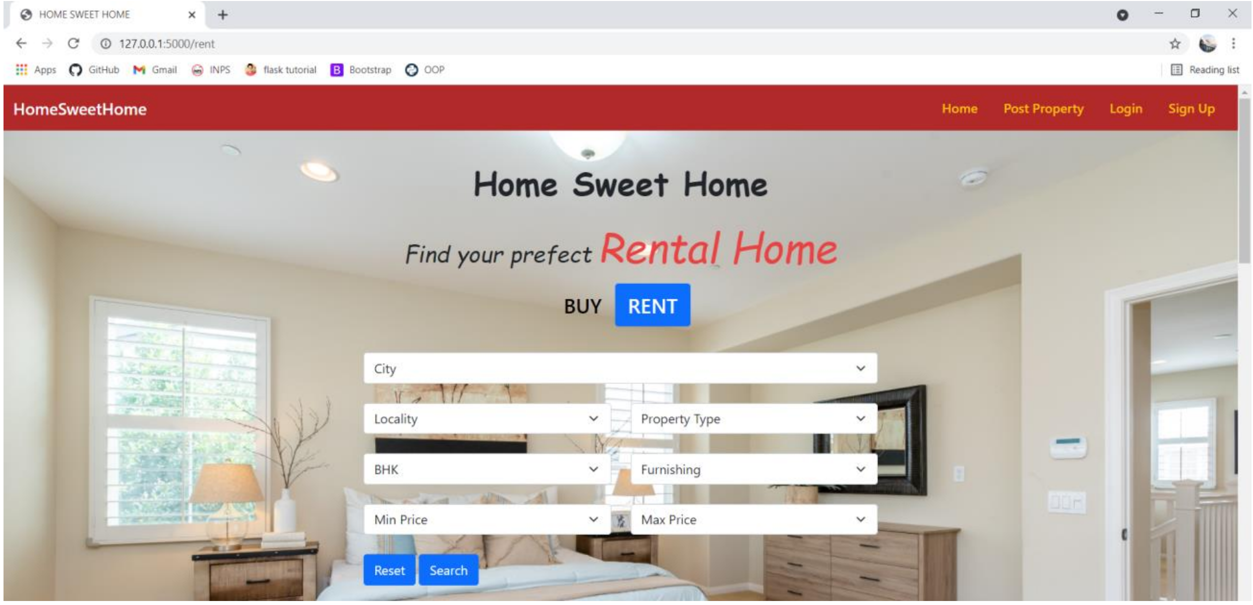


Figure 4. Home Page (Rent Section)

In rent section we have one extra card from buy section named as ‘Lodging Properties’.

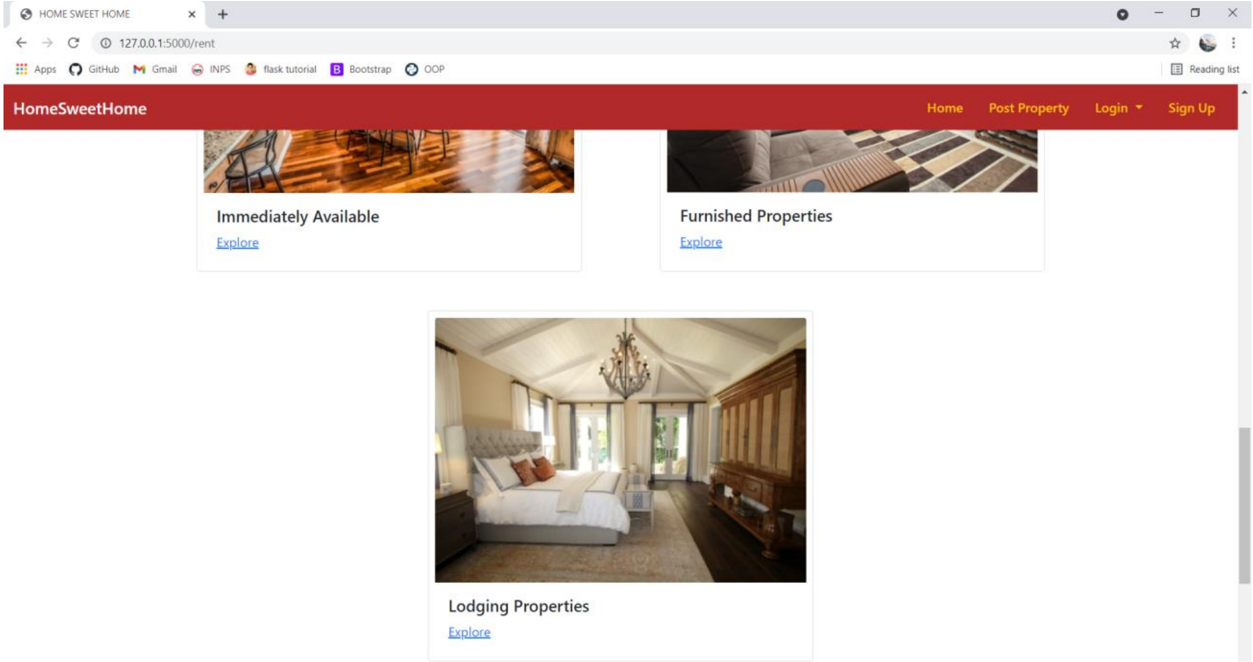


Figure 5. Lodging Properties (Rent Section)

This is our Sign-Up page where new user can enter the details to register.

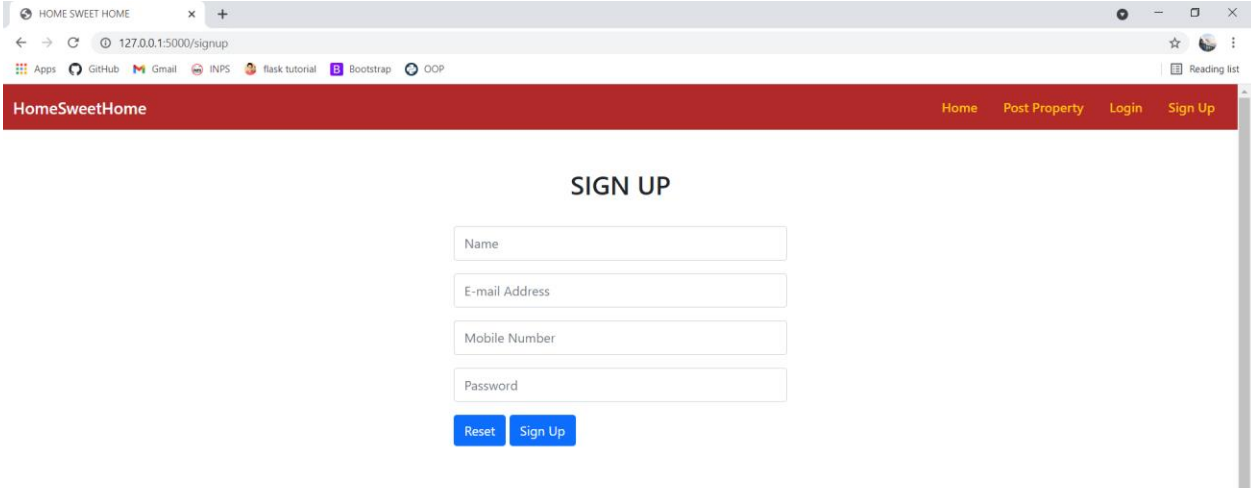
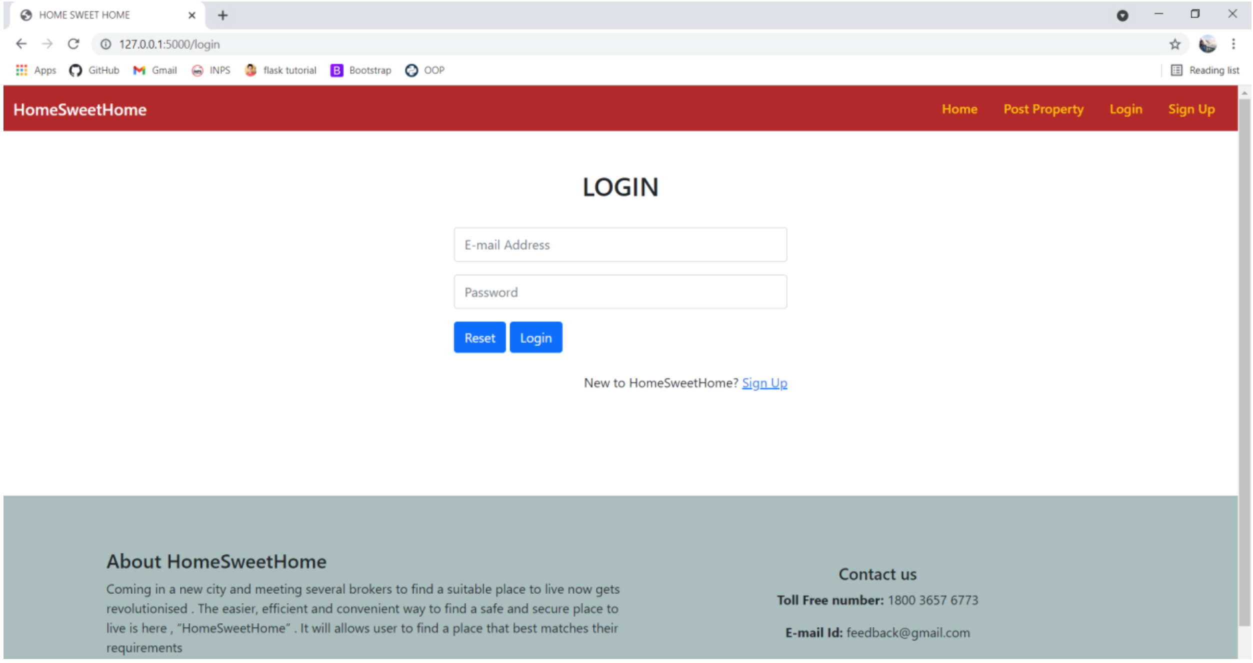


Figure 6. Sign-up Page

This is our Login page where user enter the details to login.



LOGIN

New to HomeSweetHome? Sign Up

About HomeSweetHome

Coming in a new city and meeting several brokers to find a suitable place to live now gets revolutionised . The easier, efficient and convenient way to find a safe and secure place to live is here , "HomeSweetHome" . It will allows user to find a place that best matches their requirements

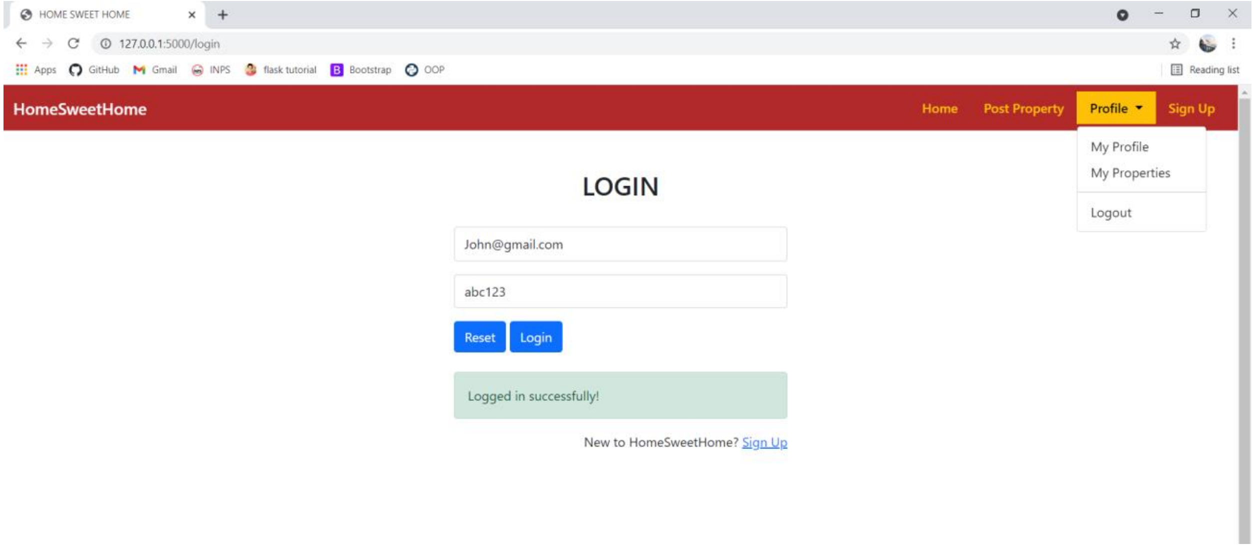
Contact us

Toll Free number: 1800 3657 6773

E-mail Id: feedback@gmail.com

Figure 7. Login Page

Success message when the user gets logged in.



LOGIN

John@gmail.com

abc123

Logged in successfully!

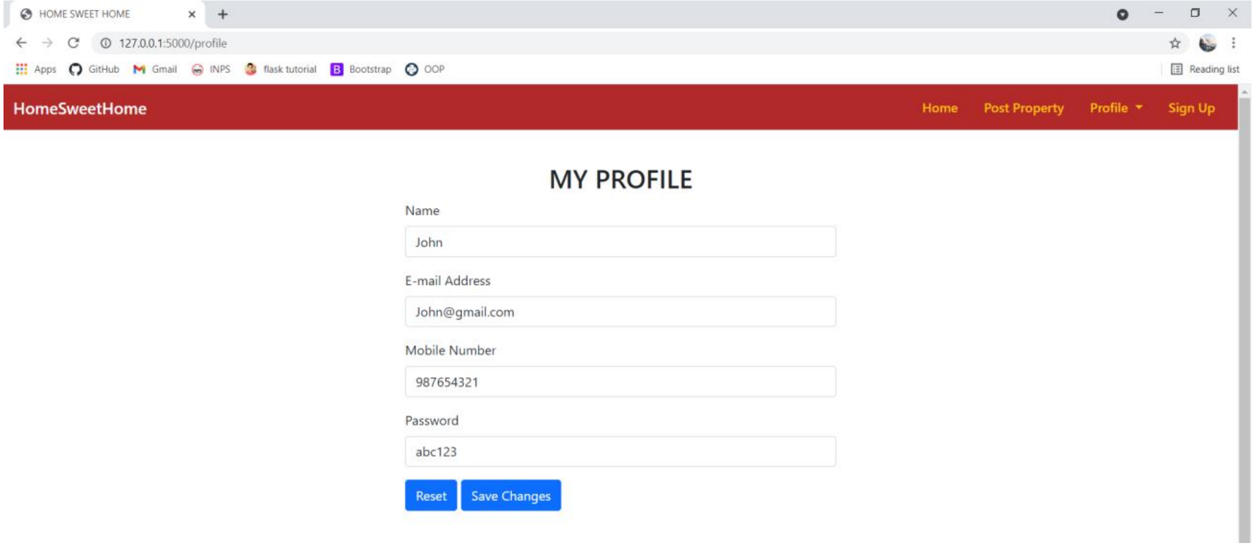
New to HomeSweetHome? Sign Up

Figure 8. Login Page with success message

After logging in successfully the ‘Login’ button in navbar will be replaced with ‘Profile’ dropdown list which have 3 options: My Profile, My Properties and Logout.

After logout the user will be redirected to our home page.

User can change their details from ‘My Profile’.



MY PROFILE

Name

John

E-mail Address

John@gmail.com

Mobile Number

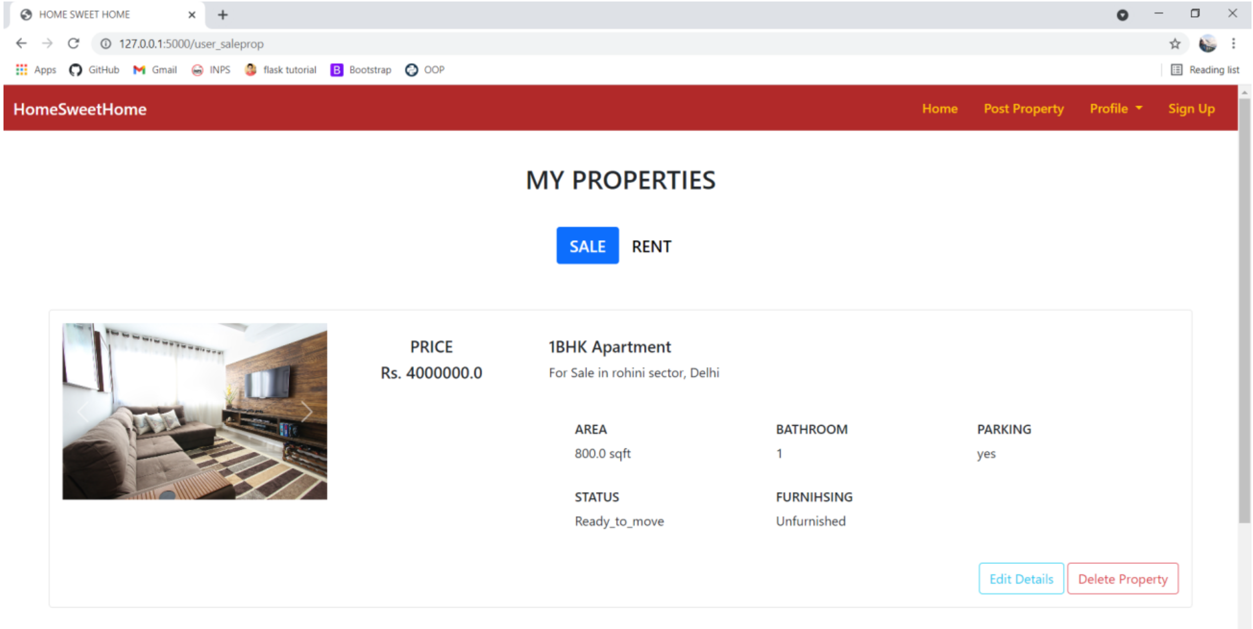
987654321

Password

abc123

Figure 9. My Profile Page

User can see the Properties they posted for sale or rent.



MY PROPERTIES

PRICE

Rs. 4000000.0

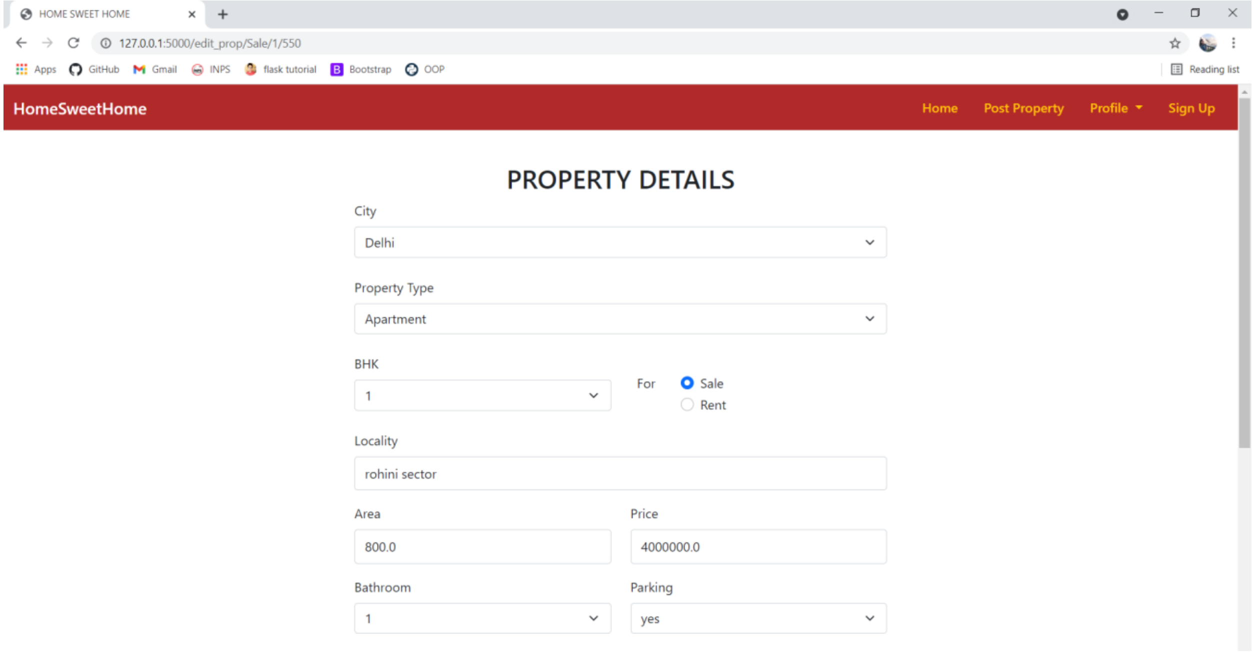
1BHK Apartment

For Sale in rohini sector, Delhi

AREA	BATHROOM	PARKING
800.0 sqft	1	yes
STATUS	FURNIHSING	
Ready_to_move	Unfurnished	

Figure 10. My Properties Page

User can also change the details of the property they posted.



PROPERTY DETAILS

City: Delhi

Property Type: Apartment

BHK: 1

For: ● Sale ○ Rent

Locality: rohini sector

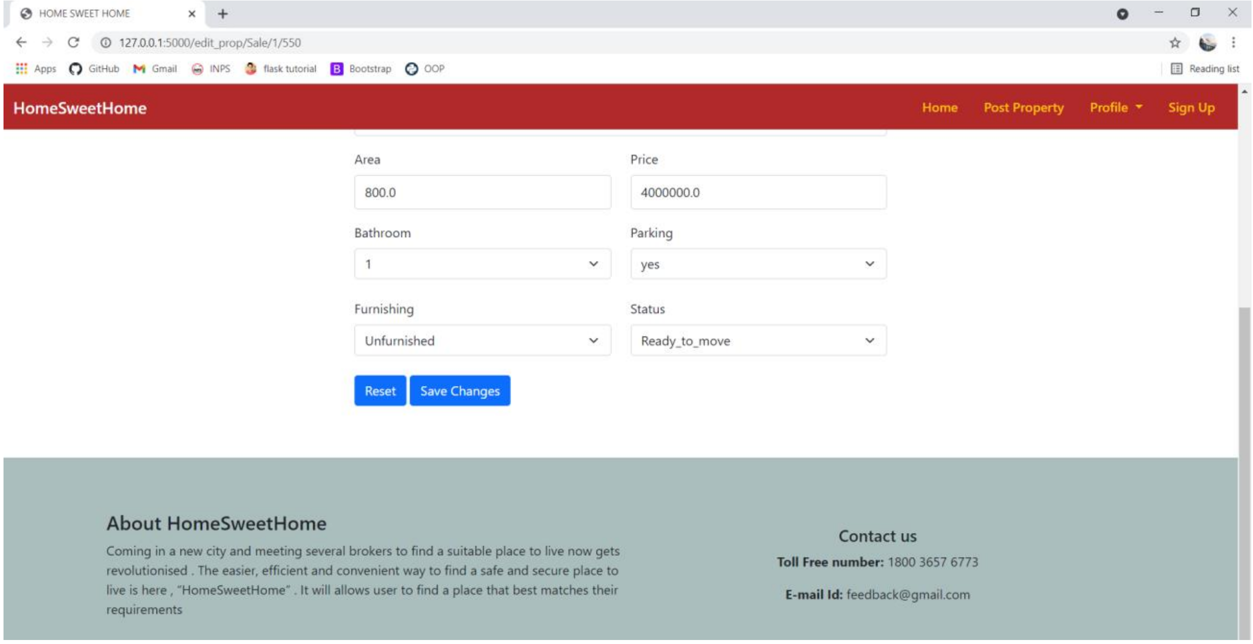
Area: 800.0

Price: 4000000.0

Bathroom: 1

Parking: yes

Figure 11. Property details Page



Area: 800.0

Price: 4000000.0

Bathroom: 1

Parking: yes

Furnishing: Unfurnished

Status: Ready_to_move

About HomeSweetHome

Coming in a new city and meeting several brokers to find a suitable place to live now gets revolutionised . The easier, efficient and convenient way to find a safe and secure place to live is here , "HomeSweetHome" . It will allows user to find a place that best matches their requirements

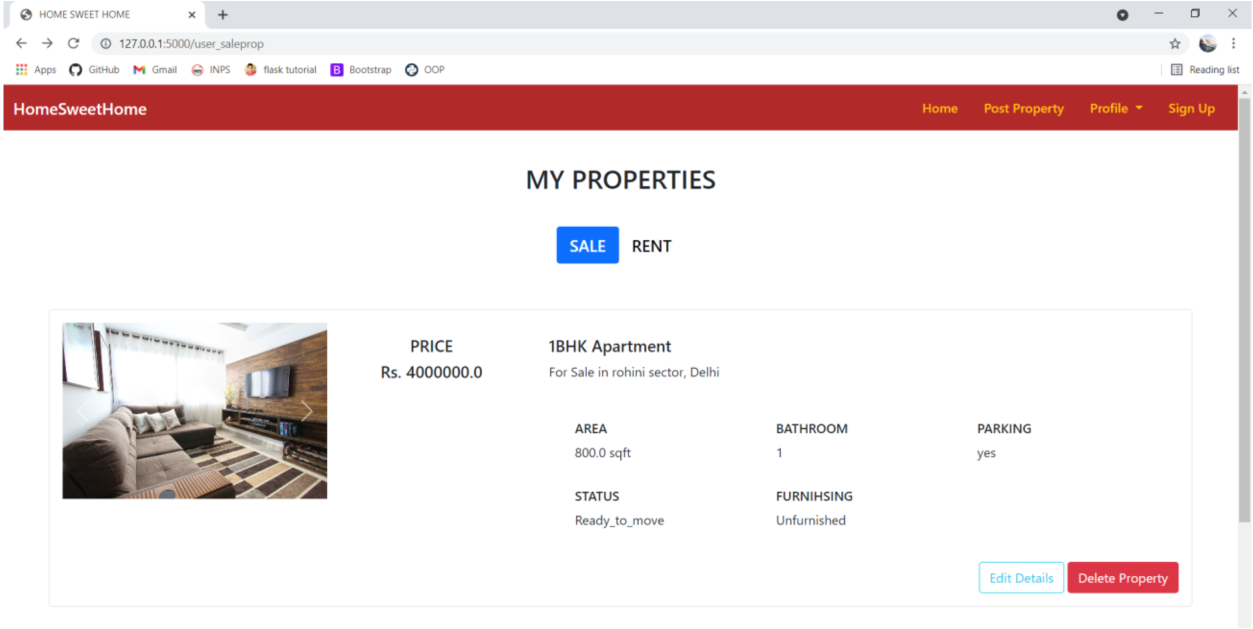
Contact us

Toll Free number: 1800 3657 6773

E-mail Id: feedback@gmail.com

Figure 12. Edit Property details

User can also delete the property posted by them.



MY PROPERTIES

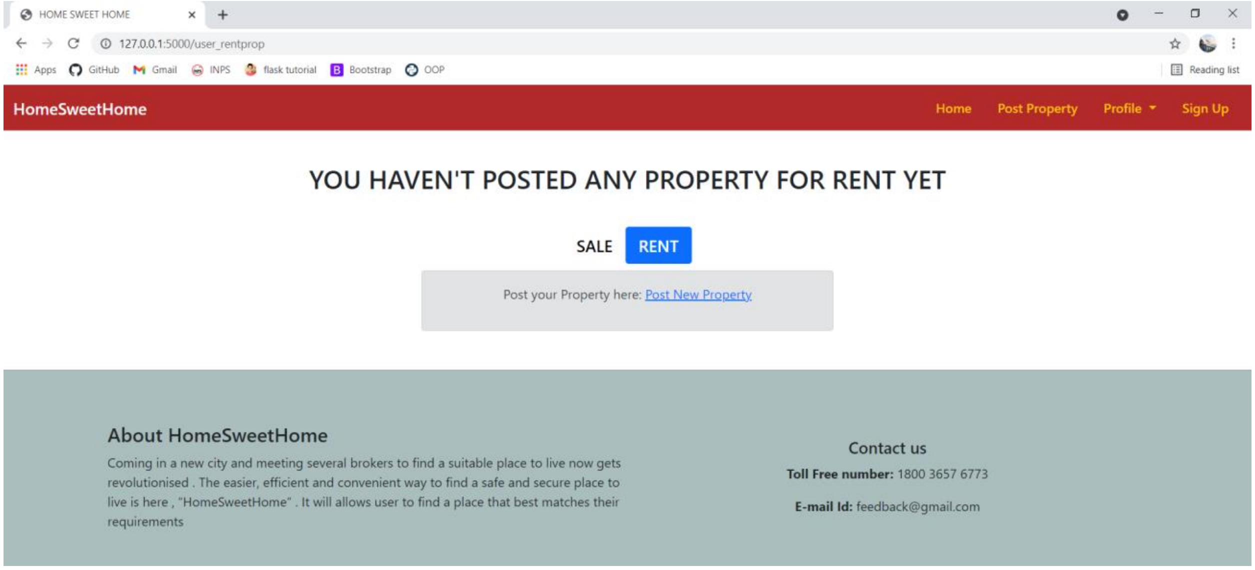
PRICE
Rs. 4000000.0

1BHK Apartment
For Sale in rohini sector, Delhi

AREA: 800.0 sqft
BATHROOM: 1
PARKING: yes
STATUS: Ready_to_move
FURNIHSING: Unfurnished

Figure 13. Delete Property Feature

If the user has not posted any property.



YOU HAVEN'T POSTED ANY PROPERTY FOR RENT YET

Post your Property here: Post New Property

About HomeSweetHome

Coming in a new city and meeting several brokers to find a suitable place to live now gets revolutionised . The easier, efficient and convenient way to find a safe and secure place to live is here , "HomeSweetHome" . It will allows user to find a place that best matches their requirements

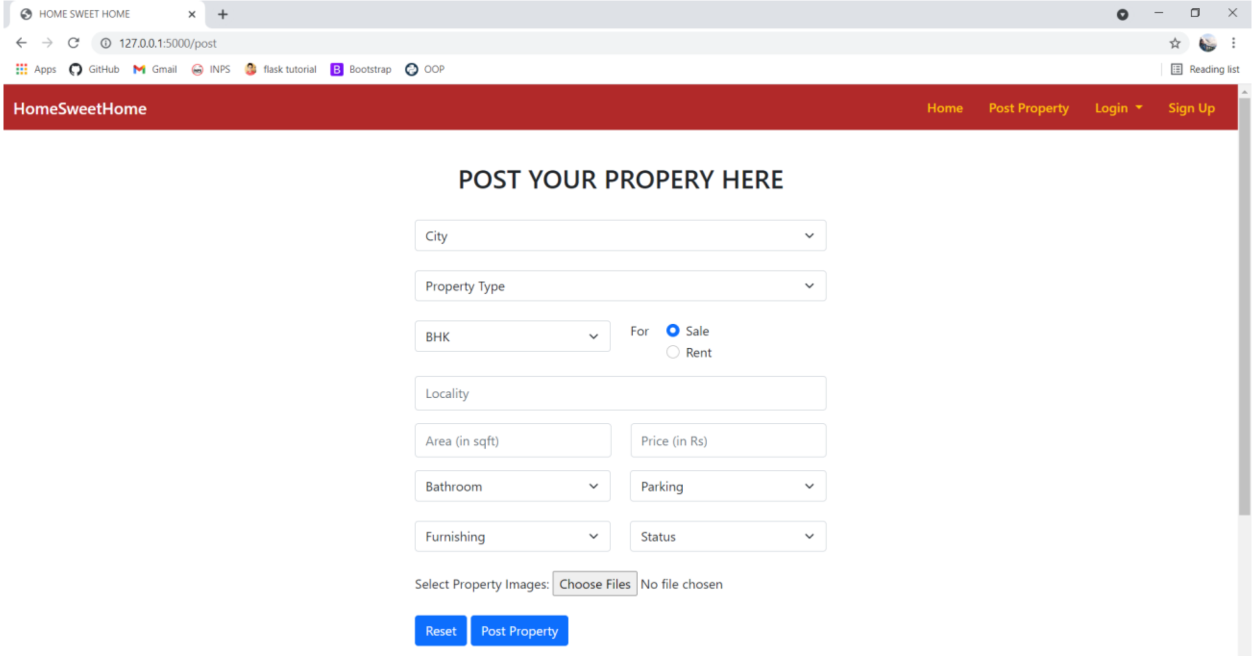
Contact us

Toll Free number: 1800 3657 6773

E-mail Id: feedback@gmail.com

Figure 14. My properties page (Haven’t posted any property)

If the user wants to post new property.



POST YOUR PROPERY HERE

City

Property Type

BHK

For ● Sale ○ Rent

Locality

Area (in sqft)

Price (in Rs)

Bathroom

Parking

Furnishing

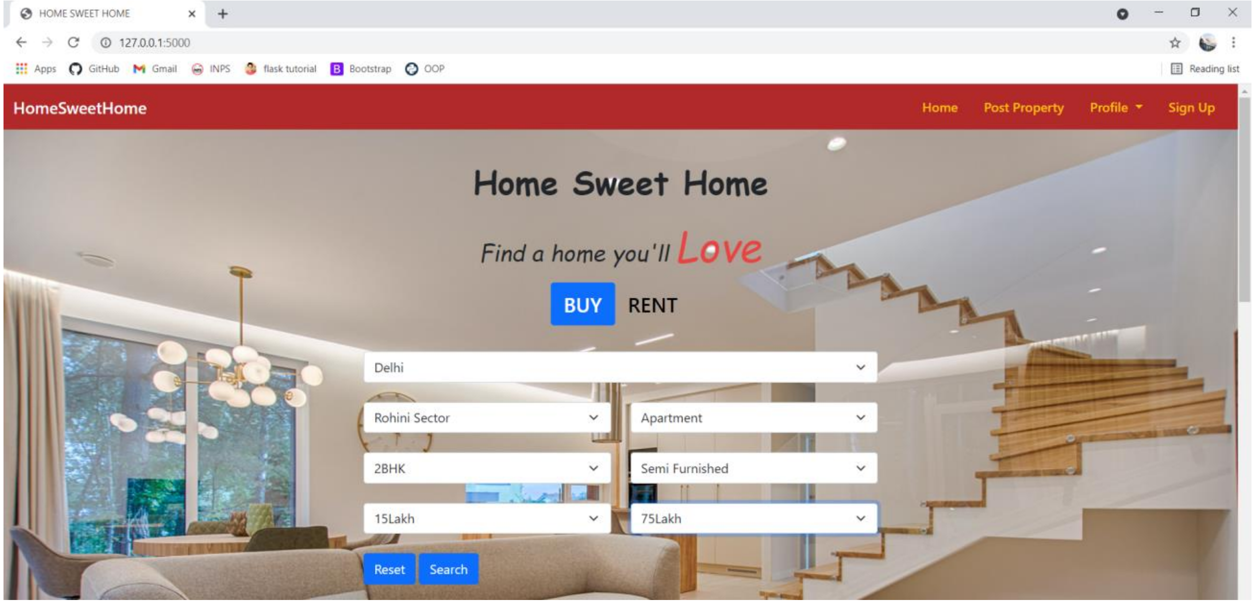
Status

Select Property Images: Choose Files No file chosen

Figure 15. Post Property Page

Now if user wants to search a property, then they have to fill the form according to their requirements.

Example 1. When the exact requirements of the user are fulfilled.



Delhi

Rohini Sector

Apartment

2BHK

Semi Furnished

15Lakh

75Lakh

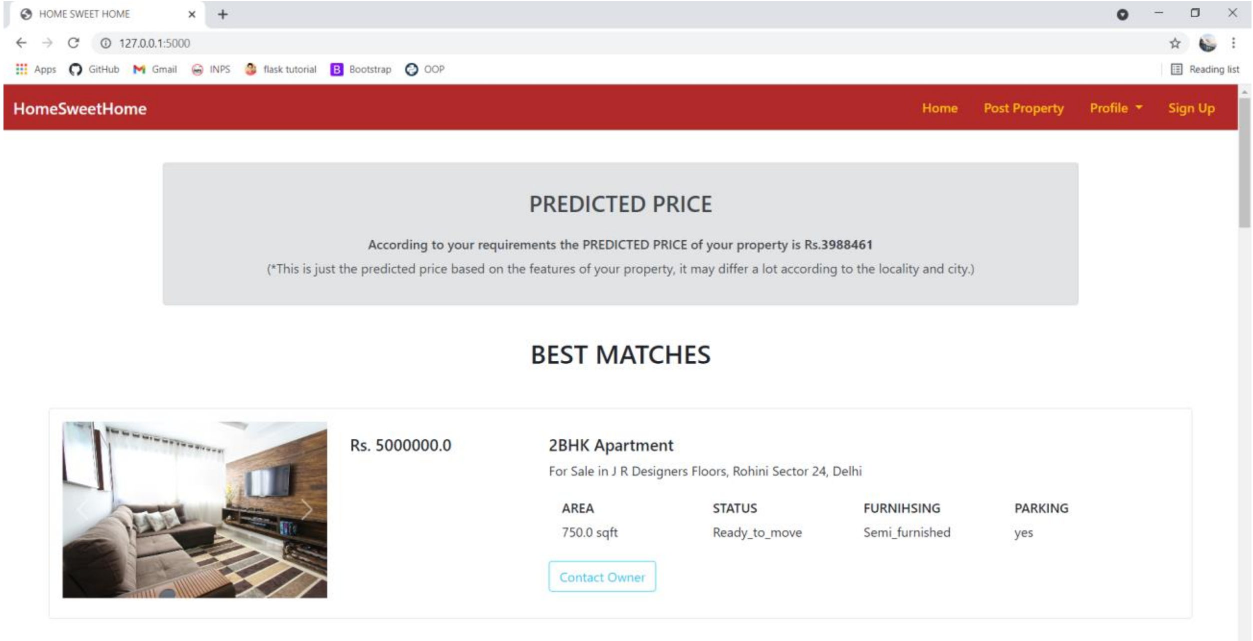
Figure 16. Credentials in search menu

Output of the previous search:

- Predicted price of the property according to user requirements.
- Best matched properties from the database.

Now if a user is interested in any property, then he can know further details of owner and property from 'CONTACT OWNER' option.

(For this user must be logged in otherwise he will be redirected to the login page)



PREDICTED PRICE

According to your requirements the PREDICTED PRICE of your property is **Rs.3988461**

(*This is just the predicted price based on the features of your property, it may differ a lot according to the locality and city.)

BEST MATCHES

Rs. 5000000.0

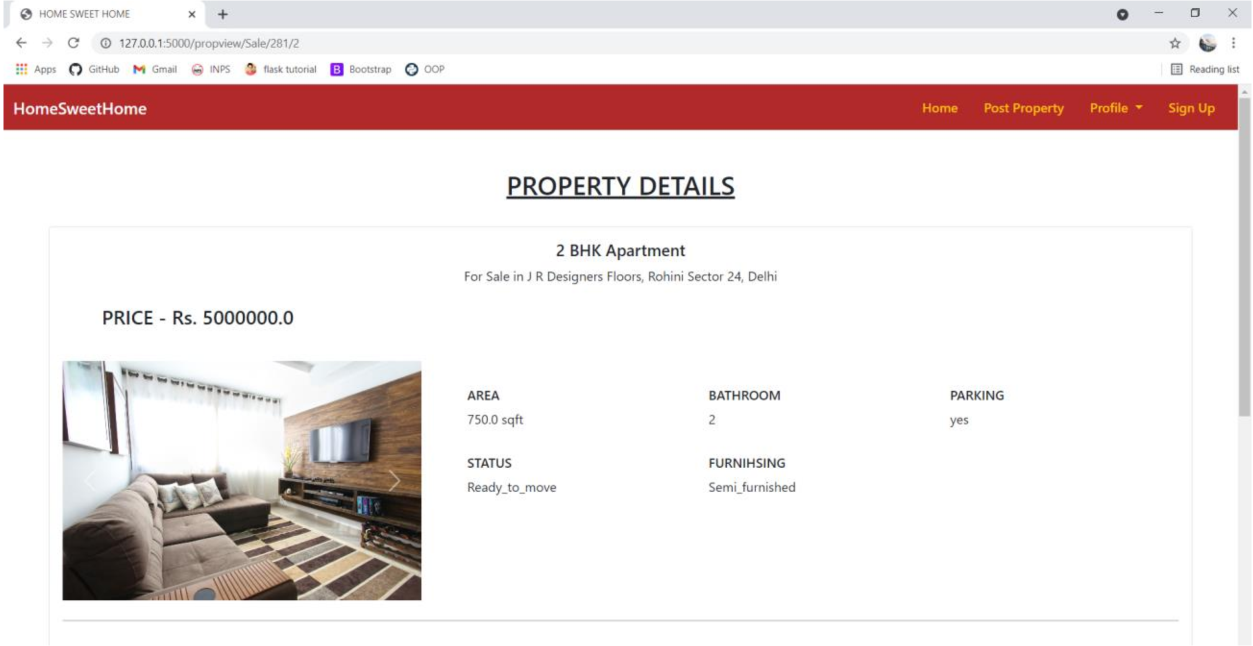
2BHK Apartment

For Sale in J R Designers Floors, Rohini Sector 24, Delhi

AREA	STATUS	FURNIHSING	PARKING
750.0 sqft	Ready_to_move	Semi_furnished	yes

Figure 17. Output after search

Property details after user choose 'contact owner' option.



PROPERTY DETAILS

2 BHK Apartment

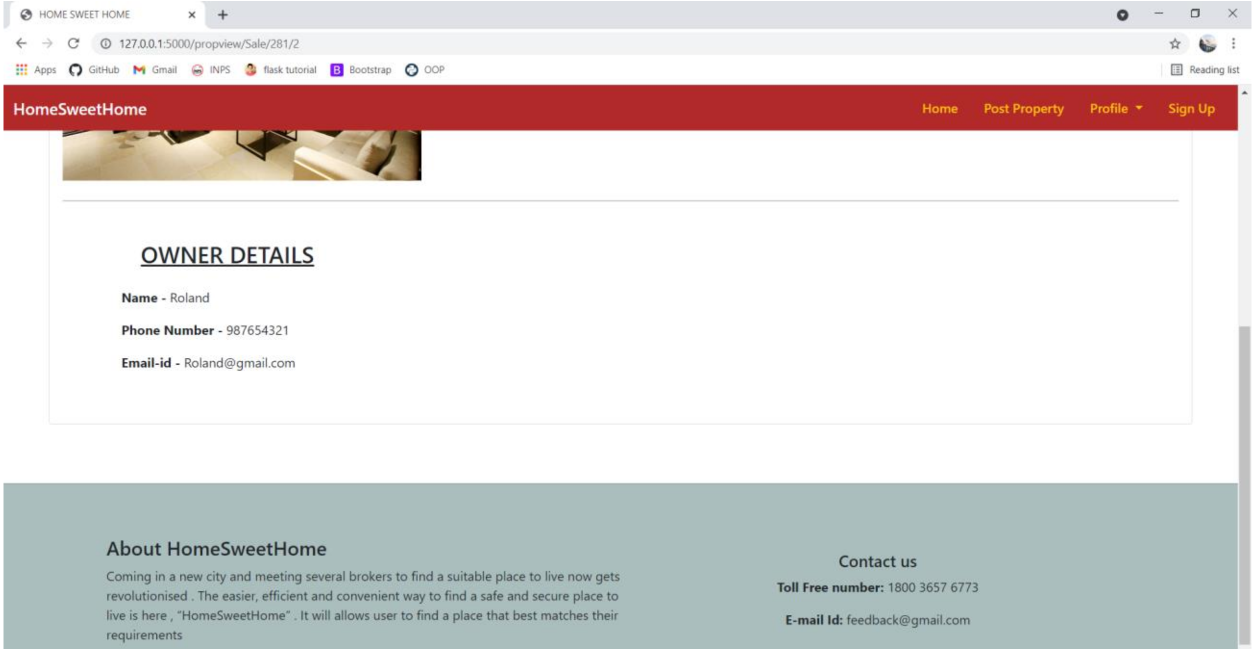
For Sale in J R Designers Floors, Rohini Sector 24, Delhi

PRICE - Rs. 5000000.0

AREA	BATHROOM	PARKING
750.0 sqft	2	yes
STATUS	FURNIHSING	
Ready_to_move	Semi_furnished	

Figure 18. Property details

Owner details after user choose 'contact owner' option.



OWNER DETAILS

Name - Roland

Phone Number - 987654321

Email-id - Roland@gmail.com

About HomeSweetHome

Coming in a new city and meeting several brokers to find a suitable place to live now gets revolutionised . The easier, efficient and convenient way to find a safe and secure place to live is here , "HomeSweetHome" . It will allows user to find a place that best matches their requirements

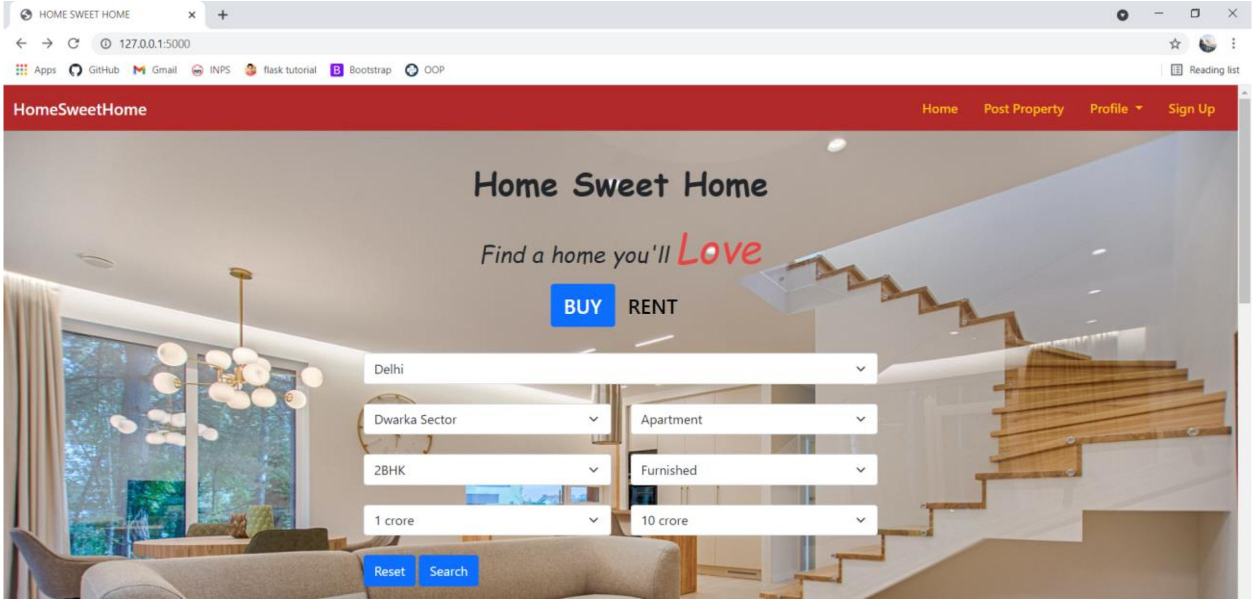
Contact us

Toll Free number: 1800 3657 6773

E-mail Id: feedback@gmail.com

Figure 19. Owner details

Example 2. When the exact requirements of the user are not fulfilled.



Delhi

Dwarka Sector

Apartment

2BHK

Furnished

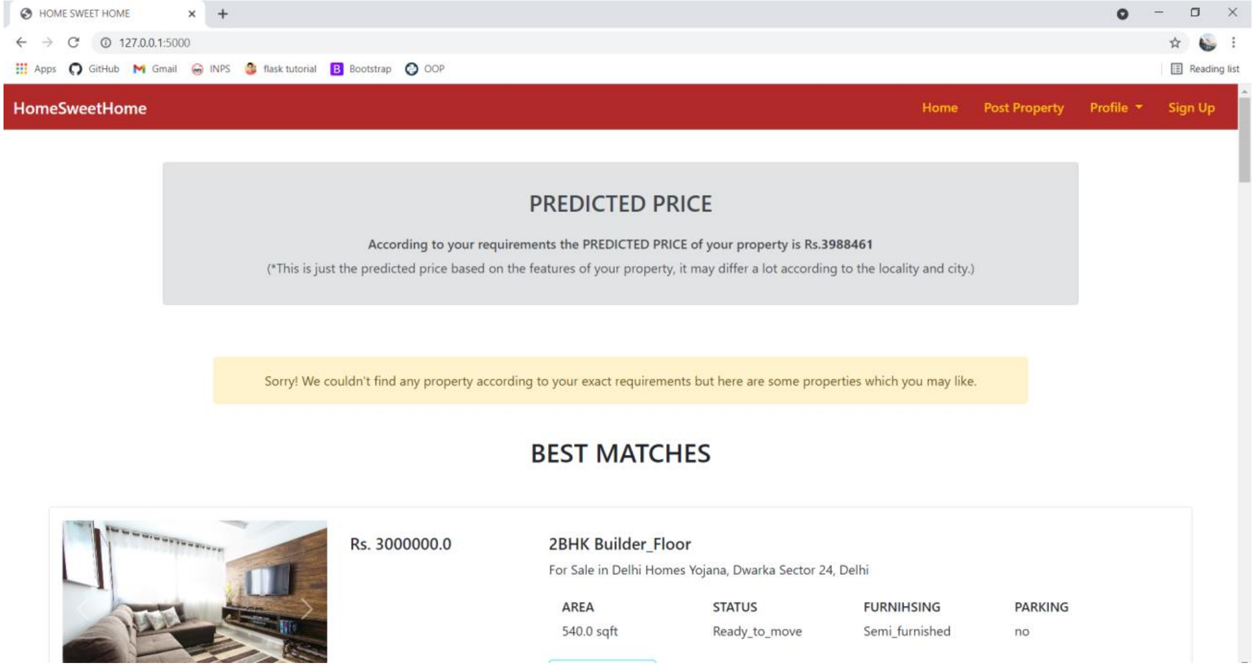
1 crore

10 crore

Figure 20. Credentials in search menu

Output of the previous search:

- Predicted price of the property according to user requirements.
- K nearest properties returned from the ML Model.



PREDICTED PRICE

According to your requirements the PREDICTED PRICE of your property is Rs.3988461

(*This is just the predicted price based on the features of your property, it may differ a lot according to the locality and city.)

Sorry! We couldn't find any property according to your exact requirements but here are some properties which you may like.

BEST MATCHES

Rs. 3000000.0

2BHK Builder_Floor

For Sale in Delhi Homes Yojana, Dwarka Sector 24, Delhi

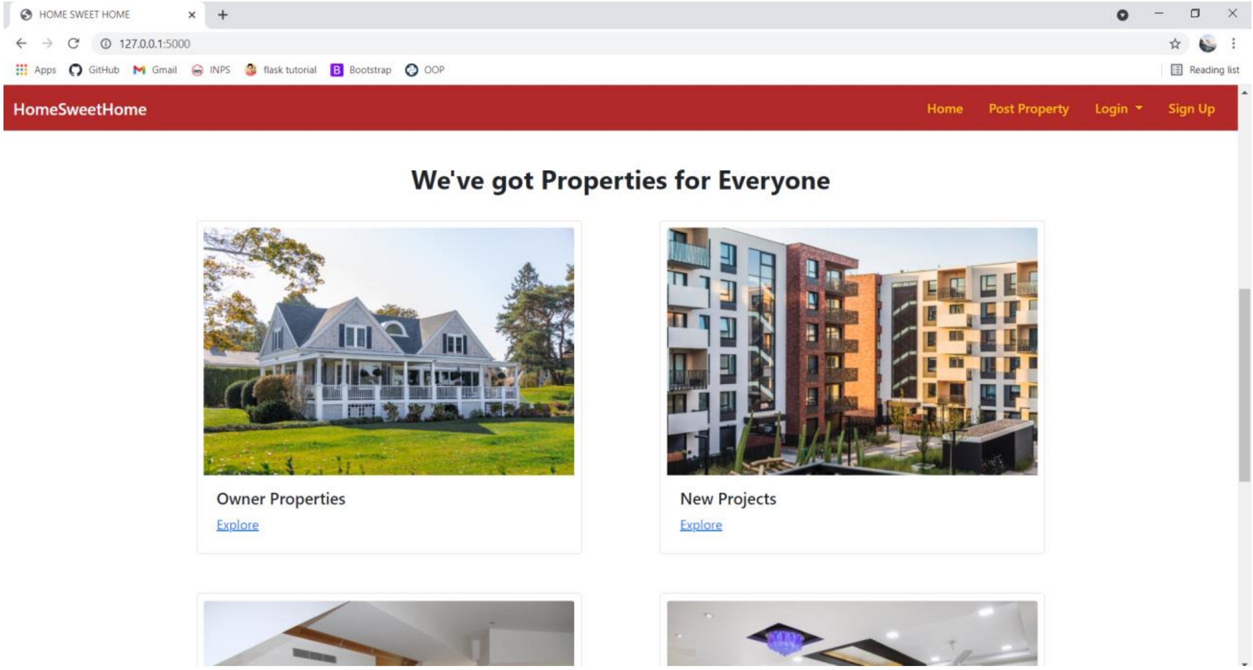
AREA	STATUS	FURNIHSING	PARKING
540.0 sqft	Ready_to_move	Semi_furnished	no

Figure 21. Output after search

Now if a user is interested in any property, then he can know further details of owner and property from 'CONTACT OWNER' option as shown earlier.

Card options are also shown on the home page.

If user wants to see the properties under that condition, he will select the explore option.



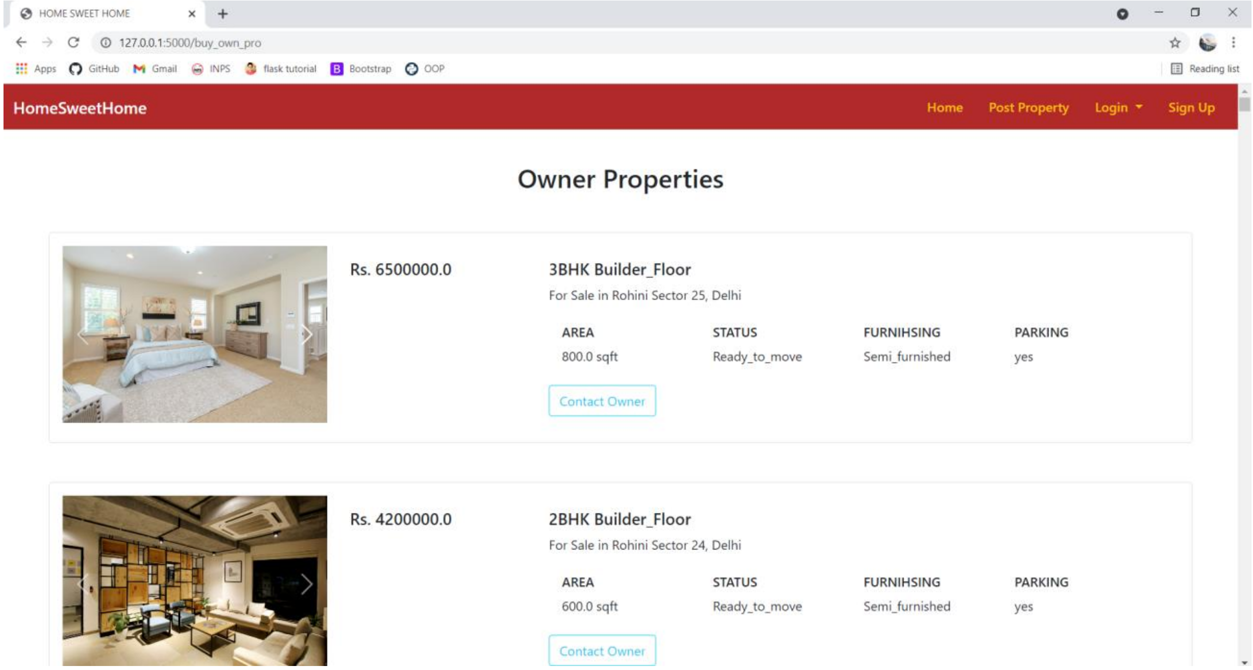
We've got Properties for Everyone

Owner Properties
Explore

New Projects
Explore

Figure 22. Cards on home page

Output after user selects explore option under owner properties category.



Owner Properties

Rs. 6500000.0

3BHK Builder_Floor

For Sale in Rohini Sector 25, Delhi

AREA	STATUS	FURNIHSING	PARKING
800.0 sqft	Ready_to_move	Semi_furnished	yes

Contact Owner

Rs. 4200000.0

2BHK Builder_Floor

For Sale in Rohini Sector 24, Delhi

AREA	STATUS	FURNIHSING	PARKING
600.0 sqft	Ready_to_move	Semi_furnished	yes

Contact Owner

Figure 23. Output after explore in cards

So, this was our web app, which is made with a purpose of making your home search easier, efficient and convenient. Hope the user find his dream house using this application.

CHAPTER-6

CONCLUSION AND FUTURE WORK

CONCLUSION

- Minimizes user efforts and their time.
- Fast Processing.
- Provides estimated price to verify price asked by owner is reasonable or not.
- Gives some suggestion in case there is no such property that exactly matches with user's requirement.

FUTURE WORK

- Bookmark facility will be further implemented so that user can see the properties they are interested in whenever they want.
- Facility to apply multiple filters in the suggested properties.
- We can add new features as and when we require.

CHAPTER-7

APPENDICES (Source Code)

Project GitHub Repo - https://github.com/Akriti1811/Property-Dealing-Flask_App

SCRIPTS

SCRIPT-1 (main.py)

```
from flask import Flask, render_template, request, session, g, redirect, url_for

import pyodbc

import pandas as pd

from pandas import DataFrame

import os

import json

import application



app = Flask(__name__)

app.secret_key = os.urandom(24)
```

```
conn = pyodbc.connect("Driver={SQL Server Native Client 11.0};"
                         "Server=DESKTOP-PLT6RQC\SQLEXPRESS;"
                        # "Server=LAPTOP-EVDFGGHS\SQLEXPRESS;
                         "Database=Property_dealing;"
                         "Trusted_Connection=yes;")


cursor = conn.cursor()



@app.route("/",methods=["POST","GET"])
def buy():


    f=open('buy_input.json')  # using json file to store user input to be used by ML model
    data=f.read()
    ss=json.loads(data)
    f.close()

    pro_for="Sale"
    heading="BEST MATCHES"
    error=None
    if request.method == 'POST':
        city=request.form["city"]
        prop=request.form["property_type"]
        bhk=request.form["bhk"]
        locality=request.form["locality"]
        min_price=request.form["Min_price"]
        max_price=request.form["Max_price"]

        furnish=request.form["furnishing"]


        # Updating json file

        ss['city']=city

        ss['propert_y']=prop

        ss['bhk']=bhk

        ss['min_price']=min_price

        ss['max_price']=max_price

        ss['furnishing']=furnish

        ss['variable']=0

        g=open('buy_input.json',"w")

        json.dump(ss, g)

        g.close()

        predicted_price=(application.index())

        price=(int)(predicted_price[0])


        print("property_no :")

        array=[]

        for i in range (predicted_price[1].size):

            array.append(int(predicted_price[1][i]))


        print(array)

        placeholders = ", ".join(["?"] * len(array))

        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Buy_house
WHERE city = ? AND type = ? AND bhk = ? AND furnishing = ? AND price BETWEEN ?
AND ? AND locality LIKE '%'+?+'%' ''',city, prop, bhk, furnish, min_price,
max_price, locality)

        sql_query=query.fetchall()
        if cursor.rowcount == 0:

            sql='''SELECT * FROM Property_dealing.dbo.Buy_house WHERE property_no
IN (''' + placeholders + ")"

            query = cursor.execute(sql, array)

            sql_query=query.fetchall()

            error="true"

        return render_template('searchview.html',data=sql_query ,pro_for=pro_for
,heading=heading,error=error,price=price)

        cursor.close()

    return render_template('buy.html')



@app.route("/rent",methods=["POST","GET"])

def rent():


    f=open('buy_input.json')

    data=f.read()

    ss=json.loads(data)




    pro_for="Rent"

    heading="BEST MATCHES"

    error=None

    if request.method == "POST":

        r_city=request.form["city"]

        r_prop=request.form["property_type"]

        r_bhk=request.form["bhk"]

        r_locality=request.form["locality"]

        r_min_price=request.form["min_price"]
```

```
r_max_price=request.form["max_price"]
r_furnish=request.form["furnishing"]
r_min_price=request.form["min_price"]
r_max_price=request.form["max_price"]

# Updating json file
ss['city']=r_city
ss['propert_y']=r_prop
ss['bhk']=r_bhk
ss['min_price']=r_min_price
ss['max_price']=r_max_price
ss['furnishing']=r_furnish
ss['variable']=1
g=open('buy_input.json',"w")
json.dump(ss, g)
g.close()


predicted_price=(application.index())
price=(int)(predicted_price[0])
print("nearest id's:")


print("property_no :")
array=[]

for i in range (predicted_price[1].size):
    array.append(int(predicted_price[1][i]))
print(array)
placeholders = ", ".join(["?"] * len(array))
```

```
        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Rent_house 
WHERE city = ? AND type = ? AND bhk = ? AND furnishing = ? AND price BETWEEN ? 
AND ? AND locality LIKE '%'+?+'%' ''',r_city, r_prop, r_bhk, r_furnish, 
r_min_price, r_max_price,r_locality)
        sql_query=query.fetchall()
        print(type(sql_query))
        if cursor.rowcount == 0:
            sql='''SELECT * FROM Property_dealing.dbo.Rent_house WHERE 
property_no  IN (''' + placeholders + ")"
            query = cursor.execute(sql, array)
            sql_query=query.fetchall()
            error="true"

        return render_template('searchview.html',data=sql_query ,pro_for=pro_for 
,heading=heading,error=error,price=price)
        cursor.close()
    return render_template('rent.html')


@app.route("/signup", methods=['GET', 'POST'])
def signup():
    message=None
    error=None
    if request.method == "POST":
        name=request.form["Name"]
        Email=request.form["Email"]
        number=request.form["Number"]
        password=request.form["Password"]
        num= int(number)
        if num<1000000000 or num>9999999999:
            error= "Number not valid! Please check again."
        else:
            message="Signed-up Successfully!"
        query= cursor.execute("SELECT MAX(user_id) FROM
Property_dealing.dbo.Users")
        last_id =query.fetchone()
        last=int(last_id[0])
        user_id=last+1
        cursor.execute('''INSERT INTO Property_dealing.dbo.Users (user_id, name,
Email_id, Phone_no, password) VALUES(?, ?, ?, ?, ?)''',user_id, name, Email,
number, password)
        conn.commit()
    return render_template('signup.html',message=message, error=error)


@app.route("/login", methods=['GET', 'POST'])
def login():
    message = None
    error = None
    if request.method == 'POST' and 'Email' in request.form and 'Password' in
request.form:
        Email = request.form['Email']
        password = request.form['Password']
        session.pop('loggedin',None)
        session.pop('userid',None)
        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Users WHERE
email_id = ? AND password = ? ''',Email, password)
        account =query.fetchone()
        if account:
            session['loggedin'] = True
            session['userid'] = account[0]
            message = "Logged in successfully!"

        else:

            error = "Error: Invalid Credentials. Please try again."



    return render_template('login.html', message=message,error=error)



@app.route("/logout")

def logout():

    session.pop('loggedin',None)

    session.pop('userid',None)

    return render_template("buy.html")



@app.route("/profile",methods=["POST","GET"])

def profile():

    if g.loggedin:

        user_id=session['userid']

        message=None

        if request.method == "POST":

            name=request.form["Name"]

            Email=request.form["Email"]

            number=request.form["Number"]

            password=request.form["Password"]

            cursor.execute('''UPDATE Property_dealing.dbo.Users SET  name= ?,
Email_id= ?, Phone_no= ?, password=? WHERE user_id= ?''', name, Email, number,
password,user_id)

            conn.commit()

            message="Changes saved Successfully!"
        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Users WHERE 
user_id= ? ''',user_id)

        sql_query=query.fetchone()

        return render_template('profile.html',data=sql_query,message=message)

    return redirect(url_for('login'))


@app.route("/user_saleprop",methods=["POST","GET"])

def user_saleprop():

    if g.loggedin:

        heading= "MY PROPERTIES"

        pro_for="Sale"

        sale="true"

        error=None

        user_id=session['userid']

        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Buy_house 
WHERE  user_id= ? ''',user_id)

        sql_query=query.fetchall()

        if cursor.rowcount == 0:

            heading="YOU HAVEN'T POSTED ANY PROPERTY FOR SALE YET"

            error="true"

        return 
render_template('userprop.html',data=sql_query,heading=heading,error=error,pro_fo
r=pro_for,Sale=sale)

    return redirect(url_for('login'))



@app.route("/user_rentprop",methods=["POST","GET"])

def user_rentprop():

    if g.loggedin:

        heading= "MY PROPERTIES"

        pro_for="Rent"
        rent="true"

        error=None

        user_id=session['userid']

        query= cursor.execute('''SELECT * FROM Property_dealing.dbo.Rent_house
WHERE  user_id= ? ''',user_id)

        sql_query=query.fetchall()

        if cursor.rowcount == 0:

            heading="YOU HAVEN'T POSTED ANY PROPERTY FOR RENT YET"

            error="true"

        return
render_template('userprop.html',data=sql_query,heading=heading,error=error,pro_fo
r=pro_for,Rent=rent)

    return redirect(url_for('login'))



@app.route("/edit_prop/<pro_for>/<user_id>/<prop_no>",methods=["POST","GET"])

def edit_prop(user_id=None,pro_for=None,prop_no=None):

    heading= "MY PROPERTIES"

    user_id=session['userid']

    message=None

    sale=None

    rent=None

    p_no=int(prop_no)

    if request.method == "POST":

        city=request.form["city"]

        prop_type=request.form["property_type"]

        bhk=request.form["bhk"]

        purpose=request.form["gridRadios"]

        locality=request.form["locality"]

        area=request.form["area"]
        price=request.form["price"]

        bathroom=request.form["bathroom"]

        parking=request.form["parking"]

        furnishing=request.form["furnishing"]

        status=request.form["status"]

        if prop_type=="Lodging_property":

            purpose="Rent"

        if purpose=="Sale":

            cursor.execute('''UPDATE Property_dealing.dbo.Buy_house SET city= ? ,
bhk= ? , bathroom= ? , parking= ? , area= ? , locality= ? , furnishing= ? ,
price= ?, status= ?, type= ? WHERE property_no= ?''', city, bhk, bathroom,
parking, area, locality, furnishing, price, status, prop_type, prop_no)

        else:

             cursor.execute('''UPDATE Property_dealing.dbo.Rent_house SET city= ?
, bhk= ? , bathroom= ? , parking= ? , area= ? , locality= ? , furnishing= ? ,
price= ?, status= ?, type= ? WHERE property_no= ?''', city, bhk, bathroom,
parking, area, locality, furnishing, price, status, prop_type, prop_no)



        conn.commit()

        message="Changes saved Successfully!"



    if pro_for=='Sale':

        sale="true"

        query= cursor.execute('''SELECT * FROM
Property_dealing.dbo.Buy_house,Property_dealing.dbo.Users WHERE
Property_dealing.dbo.Users.user_id=Property_dealing.dbo.Buy_house.user_id AND
Property_dealing.dbo.Users.user_id= ? AND
Property_dealing.dbo.Buy_house.property_no= ? ''',user_id,prop_no)

    else:

        rent="true"

        query= cursor.execute('''SELECT * FROM
Property_dealing.dbo.Rent_house,Property_dealing.dbo.Users WHERE
Property_dealing.dbo.Users.user_id=Property_dealing.dbo.Rent_house.user_id AND Property_dealing.dbo.Users.user_id= ? AND Property_dealing.dbo.Rent_house.property_no= ? ''',user_id,prop_no)

    sql_query=query.fetchone()

    return render_template('editprop.html',data=sql_query,message=message,heading=heading,sale=sale,rent=rent)



@app.route("/delete_prop/<pro_for>/<prop_no>",methods=["POST","GET"])

def delete_prop(pro_for=None,prop_no=None):

    if pro_for=="Sale":

        cursor.execute('''DELETE FROM Property_dealing.dbo.Buy_house WHERE property_no= ? ''',prop_no)

    else:

        cursor.execute('''DELETE FROM Property_dealing.dbo.Rent_house WHERE property_no= ? ''',prop_no)

    conn.commit()

    return redirect(url_for('user_saleprop'))



@app.route("/post",methods=["POST","GET"])

def post():

    if g.loggedin:

        message=None

        if request.method == "POST":

            city=request.form["city"]

            prop_type=request.form["property_type"]

            bhk=request.form["bhk"]

            purpose=request.form["gridRadios"]

            locality=request.form["locality"]

            area=request.form["area"]
            price=request.form["price"]
            bathroom=request.form["bathroom"]
            parking=request.form["parking"]
            furnishing=request.form["furnishing"]
            status=request.form["status"]
            user_id=session['userid']
            message = "You have posted your property successfully!"
            if prop_type=="Lodging_property":
                purpose="Rent"
            if purpose=="Sale":
                query= cursor.execute("SELECT MAX(property_no) FROM 
Property_dealing.dbo.Buy_house")
                last_no =query.fetchone()
                last=int(last_no[0])
                prop_no=last+1
                cursor.execute('''INSERT INTO Property_dealing.dbo.Buy_house 
(property_no, city, bhk, bathroom, parking, area, locality, furnishing, price, 
status, type, user_id) VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)''',prop_no, 
city, bhk, bathroom, parking, area, locality, furnishing, price, status, 
prop_type, user_id)
                conn.commit()
            else:
                query= cursor.execute("SELECT MAX(property_no) FROM 
Property_dealing.dbo.Rent_house")
                last_no =query.fetchone()
                last=int(last_no[0])
                prop_no=last+1
                cursor.execute('''INSERT INTO Property_dealing.dbo.Rent_house 
(property_no, city, bhk, bathroom, parking, area, locality, furnishing, price, 
status, type, user_id) VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)''',prop_no, 
city, bhk, bathroom, parking, area, locality, furnishing, price, status, 
prop_type, user_id)
                conn.commit()
```

```
        return render_template('post.html', message=message)
    return redirect(url_for('login'))


@app.route("/buy_own_pro")
def buy_own_pro():
    pro_for="Sale"
    heading="Owner Properties"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Buy_house
WHERE type='Builder_Floor'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)


@app.route("/buy_new_pro")
def buy_new_pro():
    pro_for="Sale"
    heading="New Projects"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Buy_house
WHERE type='Apartment'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)


@app.route("/buy_ready")
def buy_ready():
    pro_for="Sale"
    heading="Ready to Move-in Properties"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Buy_house
WHERE status='Ready_to_move'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)
```

```
@app.route("/buy_furnished")

def buy_furnished():

    pro_for="Sale"

    heading="Furnished Properties"

    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Buy_house
WHERE furnishing='Furnished'",conn)

    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)



@app.route("/rent_own_pro")

def rent_own_pro():

    pro_for="Rent"

    heading="Owner Properties"

    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Rent_house
WHERE type='Builder_Floor'",conn)

    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)



@app.route("/rent_new_pro")

def rent_new_pro():

    pro_for="Rent"

    heading="New Projects"

    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Rent_house
WHERE type='Apartment'",conn)

    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)



@app.route("/rent_ready")
```

```
def rent_ready():
    pro_for="Rent"
    heading="Immediately Available Properties"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Rent_house
WHERE status='Ready_to_move'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)


@app.route("/rent_furnished")
def rent_furnished():
    pro_for="Rent"
    heading="Furnished Properties"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Rent_house
WHERE furnishing='Furnished'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)


@app.route("/rent_lodging")
def rent_lodging():
    pro_for="Rent"
    heading="Lodging Properties"
    sql_query= pd.read_sql_query("SELECT * FROM Property_dealing.dbo.Rent_house
WHERE type='Lodging_property'",conn)
    return render_template('cardview.html',data=sql_query ,pro_for=pro_for
,heading=heading)



@app.route("/propview/<pro_for>/<user_id>/<prop_no>", methods=['GET', 'POST'])
def propview(user_id=None,pro_for=None,prop_no=None):
    if g.loggedin:
        if pro_for=='Sale':

            query= cursor.execute('''SELECT * FROM
Property_dealing.dbo.Buy_house,Property_dealing.dbo.Users WHERE
Property_dealing.dbo.Users.user_id=Property_dealing.dbo.Buy_house.user_id AND
Property_dealing.dbo.Users.user_id= ? AND
Property_dealing.dbo.Buy_house.property_no= ? ''',user_id,prop_no)

        else:

            query= cursor.execute('''SELECT * FROM
Property_dealing.dbo.Rent_house,Property_dealing.dbo.Users WHERE
Property_dealing.dbo.Users.user_id=Property_dealing.dbo.Rent_house.user_id AND
Property_dealing.dbo.Users.user_id= ? AND
Property_dealing.dbo.Rent_house.property_no= ? ''',user_id,prop_no)


        sql_query=query.fetchone()

        return render_template('propview.html',data=sql_query,pro_for=pro_for)

    return redirect(url_for('login'))



@app.before_request

def before_request():

    g.loggedin=None

    if 'loggedin' in session:

        g.loggedin = session['loggedin']




if __name__ == "__main__":

    app.run(host="127.0.0.1", port=8080, debug=True)
```

SCRIPT-2 (application.py)

```
from flask import Flask
import numpy as np
import pandas as pd


from sklearn.impute import SimpleImputer # used for handling missing data
from sklearn.model_selection import train_test_split # used for splitting training and testing data
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error
from math import sqrt
from sklearn import neighbors


import pyodbc
import json


conn = pyodbc.connect("Driver={SQL Server Native Client 11.0};"
                        "Server=DESKTOP-PLT6RQC\SQLEXPRESS;"
                        # "Server=LAPTOP-EVDFGGHS\SQLEXPRESS;"
                        #  "Server=DESKTOP-TS4AFA1;"
                         "Database=Property_dealing;"
                         "Trusted_Connection=yes;")
```

```
cursor = conn.cursor()


app=Flask(__name__)
@app.route("/")
def index():


    open_file=open('buy_input.json',"r")
    data_file=open_file.read()
    data=json.loads(data_file)

    if(data['variable']==0):
        df=pd.read_sql_query("SELECT * FROM dbo.Buy_house",conn)
    else:
        df=pd.read_sql_query("SELECT * FROM dbo.Rent_house",conn)
    # print(df)
    area=df['area'].mean()

    H=drop_fun(df)
    M=pd.get_dummies(H) # For categorical data
    y = M.iloc[:,3].values  #indepemdent variable
    M=M.drop("price",axis='columns')


    # attributes to determine dependent variable
    if(data['variable']==0):
        X= M.iloc[:,0:8].values # for buy data
    else:

        X = M.iloc[:,0:9].values # for rent data




    imputer = SimpleImputer(missing_values=np.nan, strategy='mean') #Handle
missing data

    imputer = imputer.fit(X)

    X = imputer.transform(X)


    # CONVERTING numpyarray INTO dataframe

    if(data['variable']==0):

X=pd.DataFrame(X,columns=['bhk','bathroom','area','furnishing_Furnished','furnish
ing_Semi_furnished','furnishing_Unfurnished','type_Apartment','type_Builder_Floor
'])

    else:

X=pd.DataFrame(X,columns=['bhk','bathroom','area','furnishing_Furnished','furnish
ing_Semi_furnished','furnishing_Unfurnished','type_Apartment','type_Builder_Floor
','type_Lodging_property'])



    y=pd.DataFrame(y,columns=['price'])





    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.32,
random_state=0)


    g=best_value_of_k(X_train,y_train,X_test,y_test) #Best value of k to find
number is neighbour
```

```
    ids=neighbouring_prices(g,X,y,area)    # GIVES NEAREST NEIGHBOUR ID'S

    nearest_negh_df=pd.DataFrame()  # EMPTY DATAFRAME


    for i in range(g):                  # STORE NEAREST NEIGHBOUR IN A DATAFRAME
        pf=df.iloc[ids[i]]
        temp=nearest_negh_df.append(pf,ignore_index=True)
        nearest_negh_df=temp

    means=nearest_negh_df['price'].mean()
    mn=nearest_negh_df['property_no']

    # return X.to_html()
    # return nearest_negh_df.to_html()
    return means,mn





def drop_fun(L):
    M= L.drop("property_no",axis='columns')
    M= M.drop("city",axis='columns')
    M= M.drop("locality",axis='columns')


    M= M.drop("status",axis='columns')


    M=M.drop("parking",axis='columns')
    M=M.drop("user_id",axis='columns')

    return M



def best_value_of_k(X_train,y_train,X_test,y_test):
    rmse_val = [] #to store rmse values for different k
    for k in range(20):
        k=k+1
        model=neighbors.KNeighborsRegressor(n_neighbors=k)
        model.fit(X_train, y_train)  #fit the model
        pred=model.predict(X_test) #make prediction on test set
        error = sqrt(mean_squared_error(y_test,pred)) #calculate rmse
        rmse_val.append(error) #store rmse values
        # print('RMSE value for k= ' , k , 'is:', error)


    p=min(rmse_val)
    # print(p)
    # print(model.score(X_test,y_test))
    for m in range(20):
        if(rmse_val[m]==p):
            print(m)
            t=m

    return t



def neighbouring_prices(g,X,y,area):
    Z=input_data(area)
    distances = np.linalg.norm(X - Z, axis='1')


    nearest_neighbour_ids = distances.argsort()[:g]
    # print( nearest_neighbour_ids)
    return nearest_neighbour_ids

    # nearest_neighbour_price = y[nearest_neighbour_ids]
    # means=np.mean(nearest_neighbour_price)
    # return nearest_neighbour_ids
    # return means,nearest_neighbour_ids


def input_data(Area):
    open_file=open('buy_input.json')
    data_file=open_file.read()
    data=json.loads(data_file)

    bathroom=0
    area=(int(data['bhk'])*270)
    # area=Area/2
    if(data['bhk']==1):
        bathroom=1
    elif(data['bhk']==2):
            bathroom=2
    elif (data['bhk']>='3'):
            bathroom=3
```

```
if(data['variable']==0):

    Z=np.array([3,2,9,1,0,1,0,1])

    if(data['furnishing']=='Furnished'):
        Z[3]=1
        Z[4]=0
        Z[5]=0

    elif(data['furnishing']=='Semi_furnished'):
        Z[3]=0
        Z[4]=1
        Z[5]=0

    elif(data['furnishing']=='Unfurnished'):


        Z[3]=0
        Z[4]=0
        Z[5]=1



    if(data['propert_y']=='Builder_Floor'):
        Z[6]=0
        Z[7]=1
    elif (data['propert_y']=='Apartment'):
        Z[6]=1
        Z[7]=0

else:

    Z=np.array([3,2,9,1,0,1,0,1,0])


    if(data['furnishing']=='Furnished'):
        Z[3]=1
        Z[4]=0
        Z[5]=0



    elif(data['furnishing']=='Semi_furnished'):
        Z[3]=0
        Z[4]=1
        Z[5]=0




    elif(data['furnishing']=='Unfurnished'):
        Z[3]=0
        Z[4]=0
        Z[5]=1


    if(data['propert_y']=='Apartment'):
        Z[6]=1
        Z[7]=0
        Z[8]=0
    elif (data['propert_y']=='Builder_Floor'):
        Z[6]=0
        Z[7]=1
        Z[8]=0

    elif(data['propert_y']=='Lodging_Property'):
        Z[6]=0
        Z[7]=0
        Z[8]=1

Z[0]=int(data['bhk'])
Z[1]=bathroom
Z[2]=area

return Z
```

TEMPLATES

1. Layout.html

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
  <meta name="author" content="">
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">


  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-eOJMYsd53ii+scO/bJGFsiCZc+5NDVN2yr8+0RDqr0Ql0h+rP48ckxlpbzKgwra6" crossorigin="anonymous">


  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/js/bootstrap.bundle.min.js" integrity="sha384-JEW9xMcG8R+pH31jmWH6WWP0WintQrMb4s7ZOdauHnUtxwoG2vI5DkLtS3qm9Ekf" crossorigin="anonymous"></script>


  <title>HOME SWEET HOME</title>


  <style>
   .highlight{
    font-size: 50px;
    color:rgb(233, 67, 67);
   }
   .header-2{
    font-family:cursive;
   }
   .header-rent{
    background-image: url(/static/images/rent-bg.jpg);
    background-size: cover;
   }
   .header-buy{
        background-image: url(/static/images/buy-background.jpg);

        background-size: cover;

      }

      .nav-item{

        font-size: 25px;

        font-weight: 600;

      }

      .nav-item-1{

        font-weight: 600;

      }

      .nav-item-2{

        font-size: 20px;

        font-weight: 500;

      }

    </style>




  </head>



  <body>


    <section>


      <nav class="navbar fixed-top navbar-light " style="background-color:
#b12929;">

        <div class="container-fluid">

          <a class="navbar-brand" style="color: #ffffff;font-weight: 600;">
HomeSweetHome</a>
```

```
     <ul class="nav justify-content-end">

      <li class="nav-item-1">

       <a class="nav-link btn-outline-warning" href="/">Home</a>

      </li>

      <li class="nav-item-1">

       <a class="nav-link btn-outline-warning" href="/post">Post
Property</a>

      </li>

      {% if session['loggedin'] %}

      <li class="nav-item-1 dropdown" >

       <a class="nav-link dropdown-toggle btn-outline-warning"
id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-
expanded="false">

        Profile

       </a>

       <ul class="dropdown-menu" aria-labelledby="navbarDropdown">

        <li><a class="dropdown-item" href="/profile">My Profile</a></li>

        <li><a class="dropdown-item" href="/user_saleprop">My
Properties</a></li>

        <li><hr class="dropdown-divider"></li>

        <li><a class="dropdown-item" href="/logout">Logout</a></li>

       </ul>

      </li>

      {% else %}

      <li class="nav-item-1">

       <a class="nav-link btn-outline-warning" href="/login">Login</a>

      </li>

      {% endif %}


      <li class="nav-item-1">
```

```
          <a class="nav-link btn-outline-warning" href="/signup">Sign Up</a>
        </li>
      </ul>
    </div>
  </nav>

</section>

{% block body %} {% endblock %}

<!-- Footer  -->
<section>

    <footer >
      <div style="background-color: rgb(170, 190, 190);">
        <hr>
        <div class="d-flex flex-wrap justify-content-center">
          <div class="col-md-5 col-12 text-left py-5">
            <h4>About HomeSweetHome</h4>
            <p>Coming in a new city and meeting several brokers to find
              a suitable place to live now gets revolutionised . The
              easier, efficient and convenient way to find a safe and
              secure place to live is here , “HomeSweetHome” . It will
              allows user to find a place that best matches their
              requirements</p>
          </div>
          <div class="col-md-5 col-12 p-2 my-auto text-center">
        <h5>Contact us</h5>
        <p><strong>Toll Free number: </strong> 1800 3657 6773</p>
        <p><strong>E-mail Id: </strong> feedback@gmail.com</p>
       </div>
      </div>
     </div>
    </footer>

  </section>

  <script>
   $(document).ready(function () {
    $("#city").change(function () {
     var val = $(this).val();
      if (val == "Delhi")
      {
       $("#locality").html("<option selected disabled
value=''>Locality</option><option value='Rohini Sector'>Rohini
Sector</option><option value='Lajpat Nagar'>Lajpat Nagar</option><option
value='Dwarka Sector'>Dwarka Sector</option><option value='Budh Vihar'>Budh
Vihar</option>><option value='New Friends Colony'>New Friends Colony</option>");
      }
      else if (val == "Banglore")
      {
       $("#locality").html("<option selected disabled
value=''>Locality</option><option value='Jayanagar'>Jayanagar</option><option
value='Koramangala'>Koramangala</option><option
value='Malleshwaram'>Malleshwaram</option><option value='Hosa Road'>Hosa
Road</option><option value='Electronic City'>Electronic City</option>");
      }
      else if (val == "")
      {
           $("#Locality").html("<option value=''>Locality</option>");
      }
    });
   });
  </script>


  </body>
</html>
```

2. Buy.html

```
    {% extends "layout.html" %}
    {% block body %}


    <!-- content -->
    <section class="header-buy">


      <div class="header-2 text-center">
        <div class=" col-12 mt-5 pt-5">
          <div class="heading ">
            <h1><strong>Home Sweet Home</strong></h1>
          </div>
          <div class="subtitle pt-3">
            <h3><i>Find a home you'll <span
class="highlight">Love</span></i></h3>
          </div>
        </div>
      </div>


      <nav class="navbar navbar-expand-lg justify-content-center" >
        <ul class="nav nav-pills ">
          <li class="nav-item">
            <a class="nav-link active" aria-current="true" href="/">BUY</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" style="color:black" href="/rent">RENT</a>
          </li>
        </ul>
      </nav><br>



      <div class="search-bar d-flex justify-content-center">



        <form class="col-5" action="" method="POST">
          <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
          <div>
            <select required id="city" class="form-select" aria-label="Default
select example" name="city">
              <option selected disabled value="">City</option>
              <option value="Banglore">Banglore</option>
              <option value="Delhi">Delhi</option>
            </select><br>
          </div>


          <div class="row">
          <div class="col">

            <select id="locality" class="form-select" aria-label="Default select
example" name="locality">

              <option selected disabled value="">Locality</option>

            </select><br>

          </div>


          <div class="col">

            <select required class="form-select" aria-label="Default select
example" name="property_type">

              <option selected disabled value="">Property Type </option>

              <option value="Apartment">Apartment</option>

              <option value="Builder_Floor">Builder_Floor</option>

            </select><br>

          </div>

        </div>

          <div class="row">

          <div class="col">

            <select required class="form-select" aria-label="Default select
example" name="bhk">

              <option selected disabled value="">BHK</option>

              <option value="1">1BHK</option>

              <option value="2">2BHK</option>

              <option value="3">3BHK</option>

              <option value="4">4BHK</option>

              <option value="5">5BHK</option>

            </select><br>

          </div>


          <div class="col">
```

```
            <select required class="form-select" aria-label="Default select 
example" name="furnishing">
              <option selected disabled value ="">Furnishing</option>
              <option value="Furnished">Furnished</option>
              <option value="Semi_furnished">Semi Furnished</option>
              <option value="Unfurnished">Unfurnished</option>
            </select><br>
          </div>
        </div>
          <div>
            <div class="row">
              <div class="col">
                <select required class="form-select" aria-label="Default select 
example" name="Min_price">
                  <option selected disabled value="">Min Price</option>
                  <option value="100000">1Lakh</option>
                  <option value="300000">3Lakh</option>
                  <option value="500000">5Lakh</option>
                  <option value="700000">7Lakh</option>
                  <option value="900000">9Lakh</option>
                  <option value="1500000">15Lakh</option>
                  <option value="2500000">25Lakh</option>
                  <option value="3500000">35Lakh</option>
                  <option value="5000000">50Lakh</option>
                  <option value="7500000">75Lakh</option>
                  <option value="10000000">1 crore</option>
                  <option value="20000000">2 crore</option>
                  <option value="50000000">5 crore</option>
                  <option value="100000000">10 crore</option>
                </select>
              </div>
              <div class="col">
                <select required class="form-select" aria-label="Default select
example" name="Max_price">
                  <option selected disabled value="">Max Price</option>
                  <option value="100000">2Lakh</option>
                  <option value="300000">3Lakh</option>
                  <option value="500000">5Lakh</option>
                  <option value="700000">7Lakh</option>
                  <option value="900000">9Lakh</option>
                  <option value="1500000">15Lakh</option>
                  <option value="2500000">25Lakh</option>
                  <option value="3500000">35Lakh</option>
                  <option value="5000000">50Lakh</option>
                  <option value="7500000">75Lakh</option>
                  <option value="10000000">1 crore</option>
                  <option value="20000000">2 crore</option>
                  <option value="50000000">5 crore</option>
                  <option value="100000000">10 crore</option>
                </select><br>
              </div>
            </div>
          </div>
          <div class="row">
            <div >
              <button type="reset" class="btn btn-primary ">Reset</button>
              <button href="/" type="submit" class="btn btn-
primary">Search</button>
            </div>
          </div><br>
```

```
        </form>

      </div>


    </section>


      <div class="content px-5 py-5">
        <h2 class="text-center"><strong>We've got Properties for Everyone</strong></h2>
        <div class="d-flex flex-wrap justify-content-center">

          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >
            <img src="/static/images/own house buy.jpg" class="card-img-top" alt="...">
            <div class="card-body">
              <h5 class="card-title">Owner Properties</h5>
              <a href="/buy_own_pro" class="card-link">Explore</a>
            </div>
          </div>

          <div class="card col-md-4 col-12  p-2 mx-5 my-4" >
            <img src="/static/images/new pro buy.jpg" class="card-img-top" alt="...">
            <div class="card-body">
              <h5 class="card-title">New Projects</h5>
              <a href="/buy_new_pro" class="card-link">Explore</a>
            </div>
          </div>


          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >
            <img src="/static/images/ready to move.jpg" class="card-img-top"
alt="...">
            <div class="card-body">
              <h5 class="card-title">Ready to Move-in</h5>
              <a href="/buy_ready" class="card-link">Explore</a>
            </div>
          </div>


          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >
            <img src="/static/images/furnish buy.jpg" class="card-img-top"
alt="...">
            <div class="card-body">
              <h5 class="card-title">Furnished Properties</h5>
              <a href="/buy_furnished" class="card-link">Explore</a>
            </div>
          </div>
        </div>
      </div>


     {% endblock %}
```

3. Rent.html

```
    {% extends "layout.html" %}
    {% block body %}
    <!-- content -->


    <section class="header-rent">

      <div class="header-2 text-center">
        <div class=" col-12 mt-5  pt-5">
          <div class="heading ">
            <h1><strong>Home Sweet Home</strong></h1>
          </div>
          <div class="subtitle pt-3">
            <h3><i>Find your prefect <span class="highlight">Rental
Home</span></i></h3>
          </div>
        </div>
      </div>

      <nav class="navbar navbar-expand-lg justify-content-center" >
        <ul class="nav nav-pills ">
          <li class="nav-item">
            <a class="nav-link" style="color:black" href="/">BUY</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="true" href="/rent">RENT</a>
          </li>
        </ul>
      </nav><br>
      <div class="search-bar d-flex justify-content-center">


        <form class="col-5" action="" method="POST">
          <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
          <div>
            <select required id="city" class="form-select" aria-label="Default
select example" name="city">
              <option selected disabled value="">City</option>
              <option value="Banglore">Banglore</option>
              <option value="Delhi">Delhi</option>
            </select><br>
          </div>

          <div class="row">
          <div  class="col">
            <select id="locality" class="form-select" aria-label="Default select
example" name="locality">
              <option selected disabled value="">Locality</option>
            </select><br>
          </div>

          <div class="col">
            <select required class="form-select" aria-label="Default select
example" name="property_type">
              <option selected disabled value="">Property Type </option>
              <option value="Apartment">Apartment</option>
              <option value="Builder_Floor">Builder Floor</option>
              <option value="Lodging_property">Lodging property</option>
            </select><br>
          </div>
        </div>
          <div class="row">
          <div class="col">
            <select required class="form-select" aria-label="Default select example" name="bhk">
              <option selected disabled value="">BHK</option>
              <option value="1">1BHK</option>
              <option value="2">2BHK</option>
              <option value="3">3BHK</option>
              <option value="4">4BHK</option>
              <option value="5">5BHK</option>
            </select><br>
          </div>

          <div class="col">
            <select required class="form-select" aria-label="Default select example" name="furnishing">
              <option selected disabled value ="">Furnishing</option>
              <option value="Furnished">Furnished</option>
              <option value="Semi_furnished">Semi Furnished</option>
              <option value="Unfurnished">Unfurnished</option>
            </select><br>
          </div>
        </div>

          <div>
            <div class="row">
```

```
              <div class="col">
                <select required class="form-select" aria-label="Default select
example" name="min_price">
                  <option selected disabled value=""> Min Price</option>
                  <option value="5000">5000</option>
                  <option value="10000">10000</option>
                  <option value="20000">20000</option>
                  <option value="30000">30000</option>
                  <option value="40000">40000</option>
                  <option value="50000">50000</option>
                  <option value="60000">60000</option>
                  <option value="70000">70000</option>
                  <option value="90000">90000</option>
                  <option value="100000">1 lakh</option>
                </select>
              </div>
              <div class="col">
                <select required class="form-select" aria-label="Default select
example" name="max_price">
                  <option selected disabled value="">Max Price</option>
                  <option value="5000">5000</option>
                  <option value="10000">10000</option>
                  <option value="20000">20000</option>
                  <option value="30000">30000</option>
                  <option value="40000">40000</option>
                  <option value="50000">50000</option>
                  <option value="60000">60000</option>
                  <option value="70000">70000</option>
                  <option value="90000">90000</option>
                  <option value="100000">1 lakh</option>
                </select><br>
              </div>
            </div>
          </div>
          <div class="row">
            <div >
              <button type="reset" class="btn btn-primary ">Reset</button>
              <button href="/rent" type="submit" class="btn btn-primary">Search</button>
            </div>
          </div><br>

        </form>

      </div>

    </section>


      <div class="content px-5 py-5">
        <h2 class="text-center"><strong>We've got Properties for Everyone</strong></h2>
        <div class="d-flex flex-wrap justify-content-center">

          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >
            <img src="/static/images/own house rent.jpg" class="card-img-top" alt="...">
            <div class="card-body">
              <h5 class="card-title">Owner Properties</h5>
              <a href="/rent_own_pro" class="card-link">Explore</a>
            </div>

          </div>


          <div class="card col-md-4 col-12  p-2 mx-5 my-4" >

            <img src="/static/images/new projects rent.jpg " class="card-img-top"
alt="...">

            <div class="card-body">

              <h5 class="card-title">New Projects</h5>

              <a href="/rent_new_pro" class="card-link">Explore</a>

            </div>

          </div>



          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >

            <img src="/static/images/immediately av.jpg" class="card-img-top"
alt="...">

            <div class="card-body">

              <h5 class="card-title">Immediately Available</h5>

              <a href="/rent_ready" class="card-link">Explore</a>

            </div>

          </div>



          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >

            <img src="/static/images/furnished rent.jpg" class="card-img-top"
alt="...">

            <div class="card-body">

              <h5 class="card-title">Furnished Properties</h5>

              <a href="/rent_furnished" class="card-link">Explore</a>

            </div>

          </div>
          <div class="card col-md-4 col-12 p-2 mx-5 my-4" >
            <img src="/static/images/lodging_property.jpg" class="card-img-top"
alt="...">
            <div class="card-body">
              <h5 class="card-title">Lodging Properties</h5>
              <a href="/rent_lodging" class="card-link">Explore</a>
            </div>
          </div>


        </div>
      </div>


     {% endblock %}
```

4. Post.html

```
{% extends "layout.html" %}
{% block body %}


<div class="post-property d-flex justify-content-center mt-5 pt-5">


    <form class="col-4" action="" method="POST">
      <div>
        <h2 class="text-center">POST YOUR PROPERY HERE</h2><br>
      </div>
      <div>
```

```
        <select required class="form-select" aria-label="Default select example" name="city">
          <option selected disabled value="">City</option>
          <option value="Ahemdabaad">Ahemdabaad</option>
          <option value="Banglore">Banglore</option>
          <option value="Chennai">Chennai</option>
          <option value="Delhi">Delhi</option>
          <option value="Faridabaad">Faridabaad</option>
          <option value="Gaziabaad">Gaziabaad</option>
          <option value="Greater Noida">Greater Noida</option>
          <option value="Gorakhpur">Gorakhpur</option>
          <option value="Gurgaon">Gurgaon</option>
          <option value="Hyderabaad">Hyderabaad</option>
          <option value="Kolkata">Kolkata</option>
          <option value="Mumbai">Mumbai</option>
          <option value="Noida">Noida</option>
          <option value="Pune">Pune</option>
          <option value="Thane">Thane</option>
          <option value="Varanasi">Varanasi</option>
        </select><br>
      </div>


      <div>
        <select required class="form-select" aria-label="Default select example" name="property_type" >
          <option selected disabled value="">Property Type </option>
          <option value="Apartment">Apartment</option>
          <option value="Builder_Floor">Builder_Floor</option>
          <option value="Lodging_property">Lodging property</option>
        </select><br>
      </div>

      <div class="row">
        <div class="col-md-6">
          <select required class="form-select" aria-label="Default select
example" name="bhk" >
            <option selected disabled value="">BHK</option>
            <option value="1">1BHK</option>
            <option value="2">2BHK</option>
            <option value="3">3BHK</option>
            <option value="4">4BHK</option>
            <option value="5">5BHK</option>
          </select><br>
        </div>
        <div class="col-md-6">
          <fieldset class="row mb-3">
            <legend class="col-form-label col-sm-2 pt-0">For</legend>
            <div class="col-sm-10">
              <div class="form-check">
                <input class="form-check-input" type="radio" name="gridRadios"
id="gridRadios1" value="Sale" checked>
                <label class="form-check-label" for="gridRadios1">
                  Sale
                </label>
              </div>
              <div class="form-check">
                <input class="form-check-input" type="radio" name="gridRadios"
id="gridRadios2" value="Rent">
                <label class="form-check-label" for="gridRadios2">
```

```
                  Rent
                </label>
              </div>
            </div>
          </fieldset>
        </div>
      </div>


      <div class="mb-3">
        <input required type="text" class="form-control py-2"
placeholder="Locality" name="locality" >
      </div>


      <div class="row">
        <div class="col mb-3">
          <input required type="number" class="form-control py-2"
placeholder="Area (in sqft)" name="area" >
        </div>

        <div class="col mb-3">
          <input required type="number" class="form-control py-2"
placeholder="Price (in Rs)" name="price" >
        </div>
      </div>


      <div class="row">
        <div class="col-md-6">
          <select required class="form-select" aria-label="Default select
example" name="bathroom">
```

```
            <option selected disabled value="">Bathroom</option>
            <option value="1">1</option>
            <option value="2">2</option>
            <option value="3">3</option>
            <option value="4">4</option>
            <option value="5">5</option>
          </select><br>
        </div>

        <div class="col-md-6">
          <select required class="form-select" aria-label="Default select
example" name="parking">
            <option selected disabled value="">Parking</option>
            <option value="yes">yes</option>
            <option value="no">no</option>
          </select><br>
        </div>
      </div>


      <div class="row">
        <div class="col-md-6">
          <select required class="form-select" aria-label="Default select
example" name="furnishing">
            <option selected disabled value="">Furnishing</option>
            <option value="Furnished">Furnished</option>
            <option value="Semi Furnished">Semi-Furnished</option>
            <option value="Unfurnished">Unfurnished</option>
          </select><br>
        </div>
        <div class="col-md-6">

          <select required class="form-select" aria-label="Default select
example" name="status">

            <option selected disabled value="">Status</option>

            <option value="Ready_to_move">Ready to move</option>

            <option value="Almost_Ready">Almost Ready</option>

          </select><br>

        </div>

      </div>



      <div>

        <label for="files">Select Property Images:</label>

        <input type="file" id="images" name="images" multiple><br><br>

      </div>


      {% if message %}

        <div class="alert alert-success" role="alert">{{ message }} </div>

      {% endif %}



      <div class="row">

        <div>

          <button type="reset" class="btn btn-primary ">Reset</button>

          <button type="submit" class="btn btn-primary">Post Property </button>

        </div>

      </div>
```

```
    </form>

  </div>


{% endblock %}
```

5. Signup.html

```
{% extends "layout.html" %}
{% block body %}


<div class="mt-3 pt-3">
    <div class=" py-5">
      <h2 class="text-center py-4">SIGN UP</h2>
      <div class="col-md-4 col-12 form-inner pb-5 px-5 mx-auto">
        <form action="" method="post">
            <div class="mb-3">
              <input required type="text" class="form-control py-2"
placeholder="Name" name="Name" value="{{ request.form.Name }}">
            </div>

            <div class="mb-3">
              <input required type="text" class="form-control py-2"
placeholder="E-mail Address" name="Email" value="{{ request.form.Email }}">
            </div>


            <div class="mb-3">
```

```
                <input required type="number" class="form-control py-2"
placeholder="Mobile Number" name="Number" value="{{ request.form.Number }}">
            </div>


            <div class="mb-3">
              <input required type="text" class="form-control py-2"
placeholder="Password" name="Password" value="{{ request.form.Password }}">
            </div>


            {% if message %}
              <div class="alert alert-success" role="alert">{{ message }} </div>
            {% endif %}
            {% if error %}
              <div class="alert alert-danger" role="alert">{{ error }} </div>
            {% endif %}

            <div class="row">
              <div>
                <button type="reset" class="btn btn-primary ">Reset</button>
                <button type="submit" class="btn btn-primary">Sign Up</button>
              </div>
            </div>



        </form>
      </div>
    </div>
  </div>
```

```
{% endblock %}
```

6. Login.html

```
{% extends "layout.html" %}

{% block body %}



<div class="mt-3 pt-3">

    <div class=" py-5" >

      <h2 class="text-center py-4">LOGIN</h2>

      <div class="col-md-4 col-12 form-inner pb-5 px-5 mx-auto">

        <form action="" method="post">



            <div class="mb-3">

              <input required type="text" class="form-control py-2"
placeholder="E-mail Address" name="Email" value="{{ request.form.Email }}">

            </div>



            <div class="mb-3">

              <input required type="text" class="form-control py-2"
placeholder="Password" name="Password" value="{{ request.form.Password }}">

            </div>


            <div class="row">

              <div>

                <button type="reset" class="btn btn-primary ">Reset</button>
                <button type="submit" class="btn btn-primary">Login</button>
              </div>
            </div>
        </form><br>


        {% if message %}
          <div class="alert alert-success" role="alert">{{ message }} </div>
        {% endif %}


        {% if error %}
          <div class="alert alert-danger" role="alert">{{ error }} </div>
        {% endif %}


        <div style="text-align: right;">
            <p>New to HomeSweetHome? <a href="/signup"> Sign Up</a></p>
        </div>
      </div>
    </div>
  </div>


{% endblock %}
```

7. Profile.html

```
{% extends "layout.html" %}
{% block body %}
```

```
<div class="pt-5 mx-2 mt-5">
<h2 class="text-center">MY PROFILE</h2>
<div class="col-md-5 col-12 form-inner pb-5 px-5 mx-auto">
    <form action="" method="post">
        <div class="mb-3">
            <label for="Name" class="form-label">Name</label>
            <input type="text" class="form-control" name="Name"
value="{{data[1]}}">
        </div>
        <div class="mb-3">
            <label for="Email" class="form-label">E-mail Address</label>
            <input type="text" class="form-control"  name="Email"
value="{{data[2]}}">
        </div>
        <div class="mb-3">
            <label for="Number" class="form-label">Mobile Number</label>
            <input type="text" class="form-control" name="Number"
value="{{data[3]}}">
        </div>
        <div class="mb-3">
            <label for="Password" class="form-label">Password</label>
            <input type="text" class="form-control"name="Password"
value="{{data[4]}}">
        </div>
        {% if message %}
            <div class="alert alert-success" role="alert">{{ message }} </div>
        {% endif %}
          <div class="row">
            <div>
```

```
              <button type="reset" class="btn btn-primary ">Reset</button>
              <button type="submit" class="btn btn-primary">Save Changes</button>
              </div>
          </div>

    </form>
</div>
</div>


{% endblock %}
```

8. Userprop.html

```
{% extends "layout.html" %}
{% block body %}
<title>MY PROPERTIES</title>
<div class="pt-5 mx-2 mt-5">
<h2 class="text-center">{{ heading }}</h2><br>
{% if Sale %}
<div>
<nav class="navbar navbar-expand-lg justify-content-center" >
    <ul class="nav nav-pills ">
      <li class="nav-item-2">
        <a class="nav-link active" aria-current="true" href="/user_saleprop">SALE</a>
      </li>
      <li class="nav-item-2">
        <a class="nav-link" style="color:black" href="/user_rentprop">RENT</a>
      </li>
    </ul>
</nav>
</div>
{% endif %}
{% if Rent %}
<div>
<nav class="navbar navbar-expand-lg justify-content-center" >
    <ul class="nav nav-pills ">
      <li class="nav-item-2">
        <a class="nav-link " style="color:black" href="/user_saleprop">SALE</a>
      </li>
      <li class="nav-item-2">
        <a class="nav-link active" aria-current="true"
href="/user_rentprop">RENT</a>
      </li>
    </ul>
</nav>
</div>
{% endif %}

{% if error %}
<div class="text-center alert alert-secondary col-md-4 mx-auto mb-5"
role="alert">
    <div class="text-center"><p>Post your Property here: <a href="/post">Post New
Property</a></p></div>
</div>
{% endif %}
    {%for row in data:%}
    <div class="card mx-5 my-5 px-3 py-3">
        <div class="row">
```

```
            <div class="col-md-3">
                <!-- <img src="/static/images/card-cover.jpg" alt="..." width="280" height="215"> -->
                <div id="carouselExampleControls" class="carousel slide" data-bs-ride="carousel">
                    <div class="carousel-inner">
                      <div class="carousel-item active">
                        <img src="/static/images/furnished rent.jpg" class="d-block w-100" alt="...">
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/card-cover.jpg" class="d-block w-100" alt="...">
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/rent-bg.jpg" class="d-block w-100" alt="...">
                      </div>
                    </div>
                    <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="prev">
                      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Previous</span>
                    </button>
                    <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="next">
                      <span class="carousel-control-next-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Next</span>
                    </button>
```

```
                  </div>
            </div>


            <div class="col-md-2 p-3 text-center">
                <h5 >PRICE</h5>
                <h5>Rs. {{ row[8] }}</h5>
            </div>

            <div class="col-md-7">
                <div class="card-body">
                    <h5 class="card-title">{{ row[2] }}BHK {{ row[10] }}</h5>
                    <p class="card-text">For {{ pro_for }} in {{ row[6] }}, {{
row[1] }}</p>


                    <div class=" pt-4">
                        <div>

                            <div class="house-detail-card px-2 py-2">
                                <div class="d-flex flex-wrap">
                                    <div class="col-md-4 col-12 px-4">
                                       <h6>AREA</h6>
                                       <p>{{row[5]}} sqft</p>
                                    </div>

                                    <div class="col-md-4 col-12 px-4">
                                       <h6>BATHROOM</h6>
                                       <p>{{row[3]}}</p>
                                    </div>
                                    <div class="col-md-4 col-12 px-4">
                                       <h6>PARKING</h6>
                                       <p>{{row[4]}}</p>
                                    </div>
                                </div>
                            </div>
                        </div>
                        <div>

                            <div class="house-detail-card px-2 py-2">
                                <div class="d-flex flex-wrap">

                                    <div class="col-md-4 col-12 px-4">
                                       <h6>STATUS</h6>
                                       <p> {{row[9]}}</p>
                                    </div>

                                    <div class="col-md-4 col-12 px-4">
                                       <h6>FURNIHSING</h6>
                                       <p>{{row[7]}}</p>
                                    </div>


                                </div>
                            </div>
                        </div>
                    </div>
```

```
                </div>

            </div>


            <div style="text-align: right;">

                <a href="/edit_prop/{{pro_for}}/{{row[11]}}/{{row[0]}}", class=" btn btn-outline-info">Edit Details</a>

                <a href="/delete_prop/{{pro_for}}/{{row[0]}}", class=" btn btn-outline-danger">Delete Property</a>

            </div>



        </div>

    </div>

    {%endfor%}

</div>



{% endblock %}
```

9. Editprop.html

```
{% extends "layout.html" %}

{% block body %}



<div class="pt-5 mx-2 mt-5">

<h2 class="text-center">PROPERTY DETAILS</h2>

<div class="col-md-6 col-12 form-inner pb-5 px-5 mx-auto">

    <form action="" method="post">
```

```
        <div>
            <label for="city" class="form-label">City</label>
            <select required class="form-select" aria-label="Default select
example" name="city">
              <option selected value="{{data[1]}}">{{data[1]}}</option>
              <option value="Ahemdabaad">Ahemdabaad</option>
              <option value="Banglore">Banglore</option>
              <option value="Chennai">Chennai</option>
              <option value="Delhi">Delhi</option>
              <option value="Faridabaad">Faridabaad</option>
              <option value="Gaziabaad">Gaziabaad</option>
              <option value="Greater Noida">Greater Noida</option>
              <option value="Gorakhpur">Gorakhpur</option>
              <option value="Gurgaon">Gurgaon</option>
              <option value="Hyderabaad">Hyderabaad</option>
              <option value="Kolkata">Kolkata</option>
              <option value="Mumbai">Mumbai</option>
              <option value="Noida">Noida</option>
              <option value="Pune">Pune</option>
              <option value="Thane">Thane</option>
              <option value="Varanasi">Varanasi</option>
            </select><br>
          </div>

          <div>
            <label for="property_type" class="form-label">Property Type</label>
            <select required class="form-select" aria-label="Default select
example" name="property_type" >
              <option selected value="{{data[10]}}">{{data[10]}}</option>
              <option value="Apartment">Apartment</option>
              <option value="Builder_Floor">Builder_Floor</option>
              <option value="Lodging_property">Lodging property</option>
            </select><br>
          </div>

          <div class="row">
            <div class="col-md-6">
              <label for="bhk" class="form-label">BHK</label>
              <select required class="form-select" aria-label="Default select
example" name="bhk" >
                <option selected value="{{data[2]}}">{{data[2]}}</option>
                <option value="1">1BHK</option>
                <option value="2">2BHK</option>
                <option value="3">3BHK</option>
                <option value="4">4BHK</option>
                <option value="5">5BHK</option>
              </select><br>
            </div>
            <div class="col-md-6">
              <fieldset class="row px-2 mt-4 mb-3">
                <legend class="col-form-label col-sm-2 pt-0">For</legend>
                <div class="col-sm-10">
                {% if sale %}
                  <div class="form-check">
                    <input class="form-check-input" type="radio"
name="gridRadios" id="gridRadios1" value="Sale" checked>
                    <label class="form-check-label"
for="gridRadios1">Sale</label>
                  </div>
                  <div class="form-check">
          <input class="form-check-input" type="radio"
name="gridRadios" id="gridRadios2" value="Rent">
          <label class="form-check-label" for="gridRadios2">Rent
</label>
         </div>
        </div>
        {% endif %}
        {% if rent %}
         <div class="form-check">
          <input class="form-check-input" type="radio"
name="gridRadios" id="gridRadios1" value="Sale" checked>
          <label class="form-check-label"
for="gridRadios1">Sale</label>
         </div>
         <div class="form-check">
          <input class="form-check-input" type="radio"
name="gridRadios" id="gridRadios2" value="Rent" checked>
          <label class="form-check-label" for="gridRadios2">Rent
</label>
         </div>
        </div>
        {% endif %}
       </fieldset>
      </div>
     </div>

     <div class="mb-3">
      <label for="locality" class="form-label">Locality</label>
      <input required type="text" class="form-control py-2" name="locality"
value="{{data[6]}}"  >
     </div>
```

```
          <div class="row">
            <div class="col mb-3">
                <label for="area" class="form-label">Area</label>
              <input required type="number" class="form-control py-2" name="area"
value="{{data[5]}}" >
            </div>

            <div class="col mb-3">
                <label for="price" class="form-label">Price</label>
              <input required type="number" class="form-control py-2"
name="price" value="{{data[8]}}"  >
            </div>
          </div>

          <div class="row">
            <div class="col-md-6">
                <label for="bathroom" class="form-label">Bathroom</label>
              <select required class="form-select" aria-label="Default select
example" name="bathroom">
                <option selected value="{{data[3]}}">{{data[3]}}</option>
                <option value="1">1</option>
                <option value="2">2</option>
                <option value="3">3</option>
                <option value="4">4</option>
                <option value="5">5</option>
              </select><br>
            </div>

            <div class="col-md-6">
                <label for="parking" class="form-label">Parking</label>
```

```
       <select required class="form-select" aria-label="Default select
example" name="parking">
        <option selected value="{{data[4]}}">{{data[4]}}</option>
        <option value="yes">yes</option>
        <option value="no">no</option>
       </select><br>
      </div>
     </div>

     <div class="row">
      <div class="col-md-6">
        <label for="furnishing" class="form-label">Furnishing</label>
       <select required class="form-select" aria-label="Default select
example" name="furnishing">
        <option selected value="{{data[7]}}">{{data[7]}}</option>
        <option value="Furnished">Furnished</option>
        <option value="Semi Furnished">Semi-Furnished</option>
        <option value="Unfurnished">Unfurnished</option>
       </select><br>
      </div>

      <div class="col-md-6">
        <label for="status" class="form-label">Status</label>
       <select required class="form-select" aria-label="Default select
example" name="status">
        <option selected value="{{data[9]}}">{{data[9]}}</option>
        <option value="Ready_to_move">Ready to move</option>
        <option value="Almost_Ready">Almost Ready</option>
       </select><br>
      </div>
```

```
          </div>


        {% if message %}
            <div class="alert alert-success" role="alert">{{ message }} </div>
        {% endif %}
          <div class="row">
            <div>
              <button type="reset" class="btn btn-primary ">Reset</button>
              <button type="submit" class="btn btn-primary">Save Changes</button>
              </div>
          </div>

    </form>
</div>
</div>


{% endblock %}
```

10. Cardview.html

```
{% extends "layout.html" %}
{% block body %}
<div class="pt-5 mx-2 mt-5">
<h2 class="text-center">{{ heading }}</h2>
    {%for index, row in data.iterrows()%}
    <div class="card m-5 px-3 py-3">
        <div class="row">
```

```
            <div class="col-md-3">
                <div id="carouselExampleControls" class="carousel slide" data-bs-ride="carousel">
                    <div class="carousel-inner">
                      <div class="carousel-item active">
                        <img src="/static/images/furnished rent.jpg" class="d-block w-100" alt="...">
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/card-cover.jpg" class="d-block w-100" alt="...">
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/rent-bg.jpg" class="d-block w-100" alt="...">
                      </div>
                    </div>
                    <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="prev">
                      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Previous</span>
                    </button>
                    <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="next">
                      <span class="carousel-control-next-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Next</span>
                    </button>
                  </div>
            </div>
            <div class="col-md-2 p-3">
                <h5>Rs. {{ row.price }}</h5>
            </div>


            <div class="col-md-7">
                <div class="card-body">
                    <h5 class="card-title">{{ row.bhk }}BHK {{ row.type }}</h5>
                    <p class="card-text">For {{ pro_for }} in {{ row.locality }},
{{ row.city }}</p>


                    <div class="house-detail-card px-2 py-2">
                        <div class="d-flex flex-wrap">
                            <div class="col-md-3 col-12 px-2">
                               <h6>AREA</h6>
                               <p>{{ row.area }} sqft</p>
                            </div>


                            <div class="col-md-3 col-12 px-2">
                               <h6>STATUS</h6>
                               <p>{{ row.status }}</p>
                            </div>


                            <div class="col-md-3 col-12 px-2">
                               <h6>FURNIHSING</h6>
                               <p>{{ row.furnishing }}</p>
                            </div>


                            <div class="col-md-3 col-12 px-2">
                               <h6>PARKING</h6>
                               <p>{{ row.parking }}</p>
                            </div>
                        </div>
                    </div>



                    <div>
                        <a
href="/propview/{{pro_for}}/{{row.user_id}}/{{row.property_no}}", class=" btn
btn-outline-info">Contact Owner</a>
                    </div>



                    </div>
            </div>



        </div>
    </div>
    {%endfor%}
</div>



  {% endblock %}
```

11. Searchview.html

```
{% extends "layout.html" %}
{% block body %}
<div class="pt-5 mx-2 mt-5">
    <div class="text-center alert alert-secondary col-md-9 mx-auto" role="alert">
        <h3 class="p-3">PREDICTED PRICE</h3>
        <h6>According to your requirements the PREDICTED PRICE of your property
is Rs.<strong>{{ price }}</strong></h6>
        <p>(*This is just the predicted price based on the features of your
property,
            it may differ a lot according to the locality and city.)</p>
    </div>
    <br>
    {% if error %}
        <br><div class="alert alert-warning text-center col-md-8 mx-auto"
role="alert">Sorry! We couldn't find any
            property according to your exact requirements but here are some
properties which you may like.</div><br>
    {% endif %}
    <!-- style="text-align:right" -->
    <h2 class="text-center">{{ heading }}</h2>
    {%for row in data:%}
    <div class="card m-5 px-3 py-3">
        <div class="row">


            <div class="col-md-3">
                <div id="carouselExampleControls" class="carousel slide" data-bs-
ride="carousel">
                    <div class="carousel-inner">
                      <div class="carousel-item active">
                        <img src="/static/images/furnished rent.jpg" class="d-
block w-100" alt="...">
```

```
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/card-cover.jpg" class="d-block w-100" alt="...">
                      </div>
                      <div class="carousel-item">
                        <img src="/static/images/rent-bg.jpg" class="d-block w-100" alt="...">
                      </div>
                    </div>
                    <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="prev">
                      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Previous</span>
                    </button>
                    <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="next">
                      <span class="carousel-control-next-icon" aria-hidden="true"></span>
                      <span class="visually-hidden">Next</span>
                    </button>
                  </div>
            </div>

            <div class="col-md-2 p-3">
                <h5>Rs. {{ row[8] }}</h5>
            </div>

            <div class="col-md-7">
```

```
                <div class="card-body">
                    <h5 class="card-title">{{ row[2] }}BHK {{ row[10] }}</h5>
                    <p class="card-text">For {{ pro_for }} in {{ row[6] }}, {{ row[1] }}</p>


                    <div class="house-detail-card px-2 py-2">
                        <div class="d-flex flex-wrap">
                            <div class="col-md-3 col-12 px-2">
                               <h6>AREA</h6>
                               <p>{{ row[5] }} sqft</p>
                            </div>


                            <div class="col-md-3 col-12 px-2">
                               <h6>STATUS</h6>
                               <p>{{ row[9] }}</p>
                            </div>


                            <div class="col-md-3 col-12 px-2">
                               <h6>FURNIHSING</h6>
                               <p>{{ row[7] }}</p>
                            </div>

                            <div class="col-md-3 col-12 px-2">
                               <h6>PARKING</h6>
                               <p>{{ row[4] }}</p>
                            </div>
                        </div>

                    </div>



                    <div>

                        <a href="/propview/{{pro_for}}/{{row[11]}}/{{row[0]}}",
class=" btn btn-outline-info">Contact Owner</a>

                    </div>

                </div>

            </div>



        </div>

    </div>

    {%endfor%}

</div>



  {% endblock %}
```

## 12.	Propview.html

```
} {% extends "layout.html" %}

{% block body %}



<div class="mt-3 pt-3">

    <div class=" p-5 ">

        <h2 class="text-center py-4"><u>PROPERTY DETAILS</u></h2>

        <div class="card m-2 px-3 py-3">
```

```
            <div class="text-center">
                <h5 class="card-title">{{data[2]}} BHK {{data[10]}}</h5>
                <p class="card-text ">For {{pro_for}} in {{data[6]}},
{{data[1]}}</p>
            </div><br>


            <div>
                <h4 class="px-5">PRICE - Rs. {{data[8]}}</h4>
            </div><br>


            <div class="row py-2">

                <div class="col-md-4">
                    <div id="carouselExampleControls" class="carousel slide"
data-bs-ride="carousel">
                        <div class="carousel-inner">
                          <div class="carousel-item active">
                            <img src="/static/images/furnished rent.jpg"
class="d-block w-100" alt="...">
                          </div>
                          <div class="carousel-item">
                            <img src="/static/images/card-cover.jpg" class="d-
block w-100" alt="...">
                          </div>
                          <div class="carousel-item">
                            <img src="/static/images/rent-bg.jpg" class="d-block
w-100" alt="...">
                          </div>
                        </div>
                        <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="prev">
                          <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                          <span class="visually-hidden">Previous</span>
                        </button>
                        <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleControls" data-bs-slide="next">
                          <span class="carousel-control-next-icon" aria-hidden="true"></span>
                          <span class="visually-hidden">Next</span>
                        </button>
                      </div>
                </div>


                <div class="col-md-8 pt-4">
                    <div>


                        <div class="house-detail-card px-2 py-2">
                            <div class="d-flex flex-wrap">
                                <div class="col-md-4 col-12 px-4">
                                   <h6>AREA</h6>
                                   <p>{{data[5]}} sqft</p>
                                </div>


                                <div class="col-md-4 col-12 px-4">
                                   <h6>BATHROOM</h6>
                                   <p>{{data[3]}}</p>
                                </div>
                                <div class="col-md-4 col-12 px-4">
                                   <h6>PARKING</h6>
                                   <p>{{data[4]}}</p>
                                </div>
                            </div>
                        </div>
                    </div>
                    <div>

                        <div class="house-detail-card px-2 py-2">
                            <div class="d-flex flex-wrap">


                                <div class="col-md-4 col-12 px-4">
                                   <h6>STATUS</h6>
                                   <p> {{data[9]}}</p>
                                </div>

                                <div class="col-md-4 col-12 px-4">
                                   <h6>FURNIHSING</h6>
                                   <p>{{data[7]}}</p>
                                </div>


                            </div>
                        </div>
                    </div>
                </div>
```

```
            </div>


            <hr>
            <div class=" px-5 py-2">
                <h3 class=" px-5 pt-4 pd-2"><u>OWNER DETAILS</h3></u></h3>
                <div class="m-2 px-3 py-3">
                    <p><strong>Name - </strong>{{data[13]}}</p>
                    <p><strong>Phone Number - </strong>{{data[15]}}</p>
                    <p><strong>Email-id - </strong>{{data[14]}}</p>
                </div>
            </div>
        </div>
    </div>
  </div>
{% endblock %}
```

JSON FILE

Buy_input.json

```
{"city": "Banglore", "propert_y": "Apartment", "bhk": "3", "min_price": "90000", 
"max_price": "100000", "furnishing": "Furnished", "variable": 1}
```

CHAPTER-8

REFERENCES

- https://www.w3schools.com/
- https://getbootstrap.com/
- https://towardsdatascience.com/
- https://realpython.com/python-web-applications/
- https://www.youtube.com/
- https://stackoverflow.com/
- https://codeshack.io
- https://pythonspot.com
- https://pynative.com
- https:// https://www.codewithharry.com/videos/web-dev-using-flask-and-python-1