

MINI PROJECT REPORT

Real Estate Management System

205001102
M R Shrijith

205001106
Siddharth S

205001128
Vishal V

1. PROBLEM STATEMENT

Traditional systems to organize and manage property data aren't enough with the rise in popularity of the rental industry. Property owners thus require software that is capable of handling large amounts of information and also be able to showcase the properties to a larger audience. There is also a need for people who are willing to buy a property to be able to filter out important information according to their interests. If the job of a broker is taken care of by a software, it makes the job easier for both the parties (owner/buyer) involved.

The Real Estate Management System website caters to property owners, buyers, and moderators by facilitating their corresponding needs. Buyers can search and select various properties sold by authenticated owners, as approved by moderators.

2. SOFTWARE REQUIREMENTS

2.1 FUNCTIONAL REQUIREMENTS

Functional requirements outline what services the system presents to the user. Following is a list of functionalities of the system. There are registered people in the system.

2.2.1. Login: Buyer, Owner and Moderator can login to the system via their respective usernames and passwords.

2.2.2. Buyer:

- View properties- The buyer can view the properties up for sale in and around his location from the existing database.

- Search for properties using filters- Filters such as Price Range, Number of Rooms, Area, etc can be used by the buyer to hasten his decision making process.

- Select properties- The buyer then selects the property based on their liking.

- Contact Owner- Upon selecting the property, the buyer can contact the owner whose phone number and email address is displayed.

2.2.3. Owner:

- Add property- The owner can add the property for sale to the existing database.

- Delete property- If the owner changes his mind about selling the house, the property can be deleted from the database.

- Display- Displays property details such as :

1. Name

2. Address

3. Price

4. Property Documents

5. Area/Blueprint

6. Number of bedrooms

7. Type

- Restrictions- Defines the ground rules of the property.

2.2.4. Moderator:

- Authentication- Approve/Reject property sales from owners after authentication.

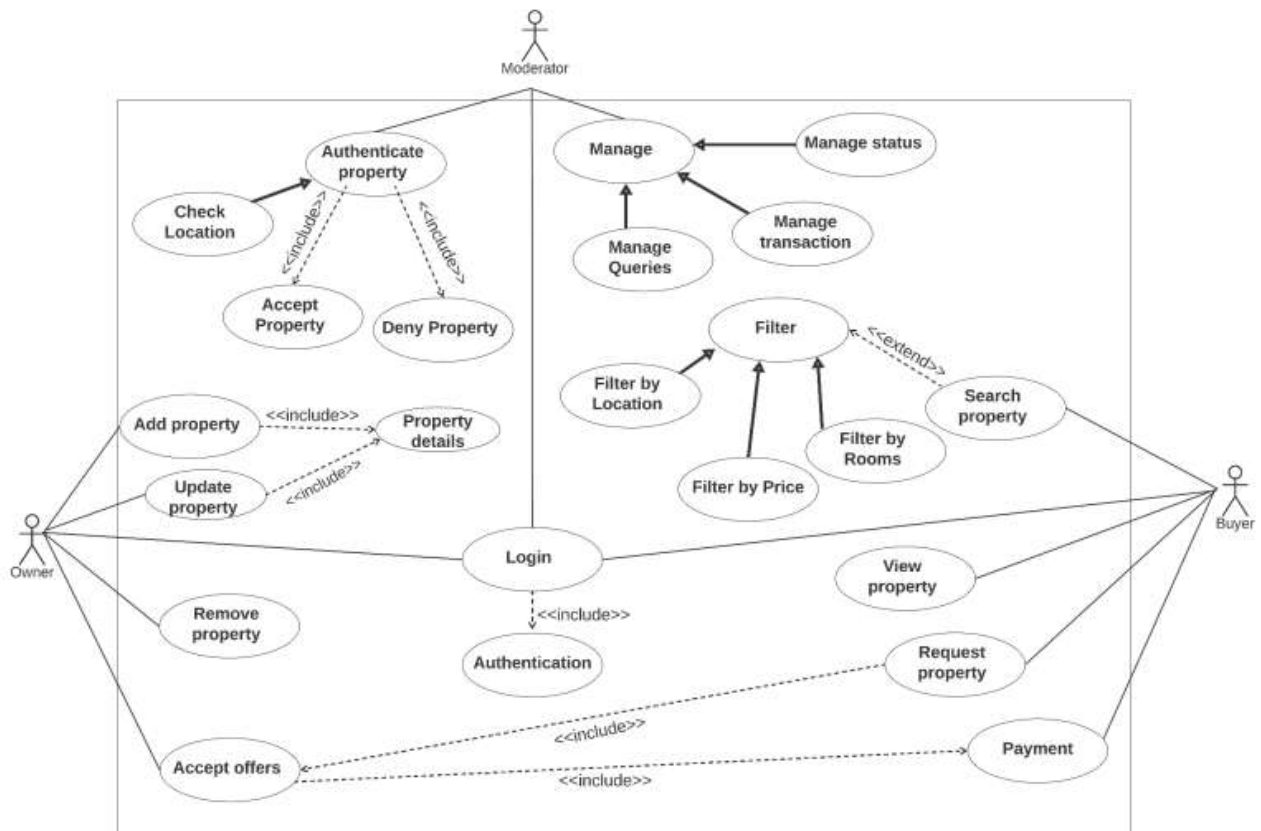
- Mediator- Connect buyer and owner once the offer is accepted.

2.2 Non- Functional Requirements

These are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

1. Security: Credentials of users are stored after encryption and hashing are done.
2. Capacity: System can not accommodate large no. of users, texts or images as data are stored in a free-of-cost-database that has limited storage.
3. Usability: UI/UX is easy to understand.
4. Performance: System may take time to load images as images are stored in an online cloud.

3. System Design:



Main success scenario:

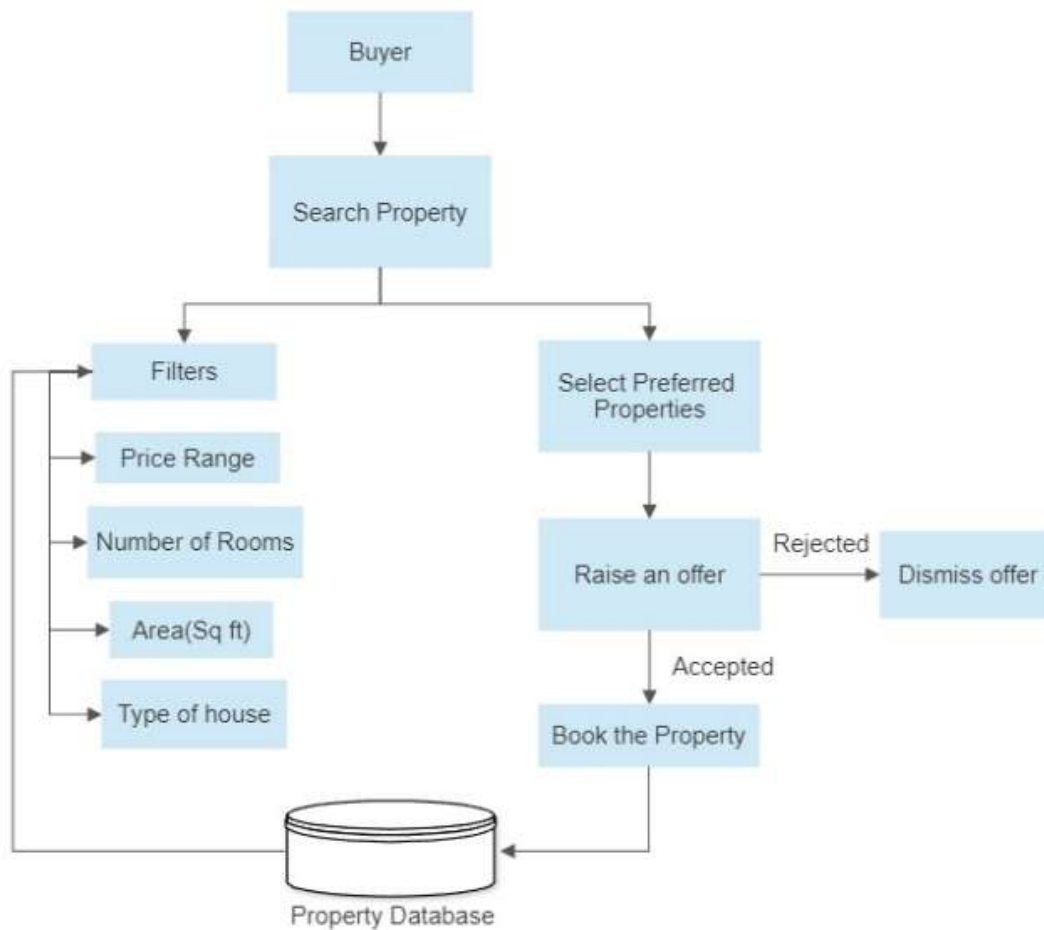
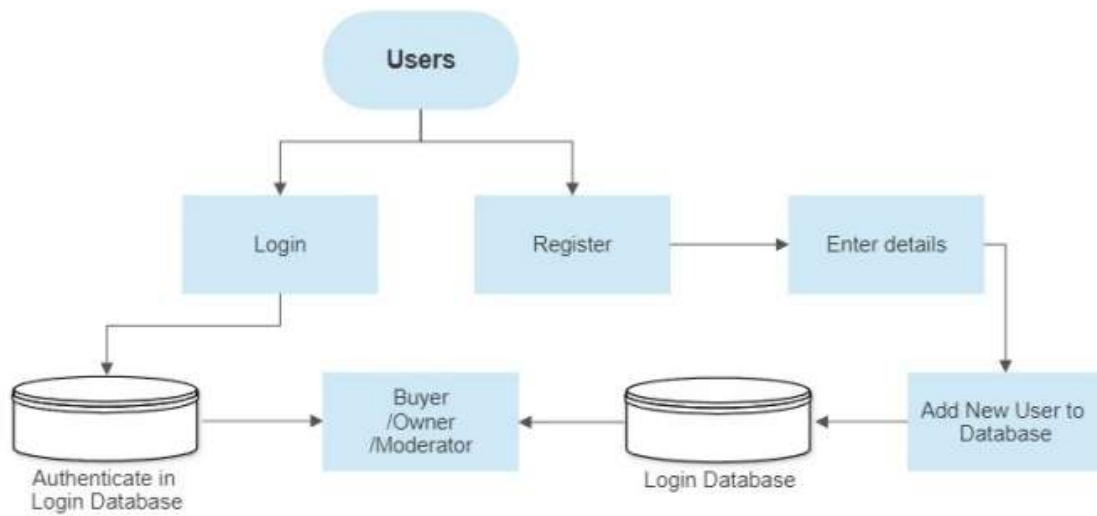
1. Owner adds a new property “for sale” on the website.
2. The moderator adds the property to the database after verifying all the details and checking its authenticity.
3. The buyer searches for a property to buy through various filter options given to him.
4. The buyer views all the different properties that satisfy the criteria, and selects one of the properties and contacts the buyer

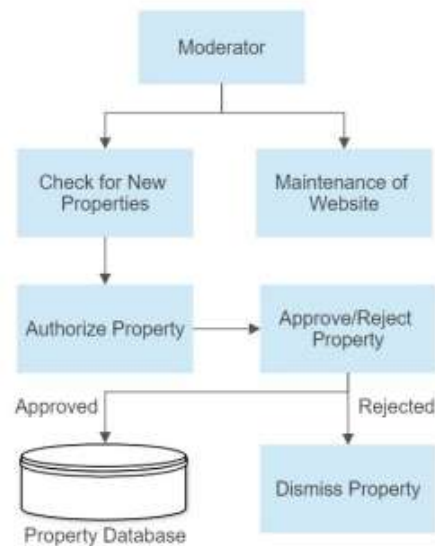
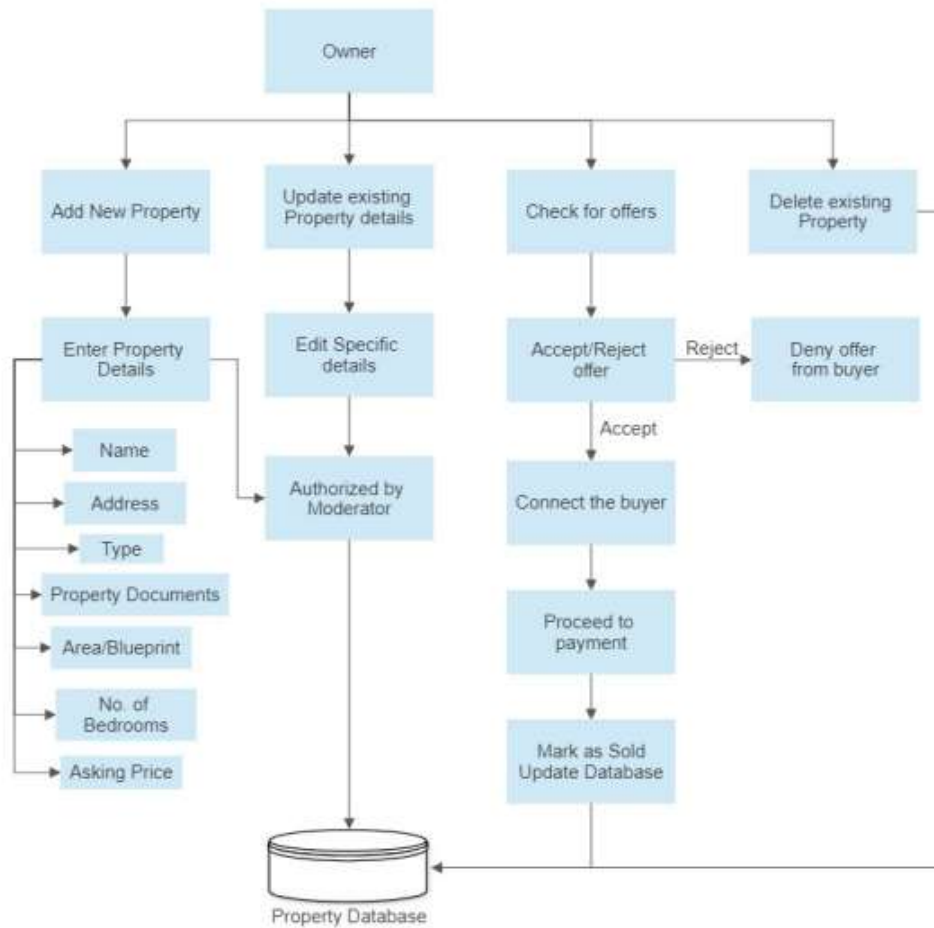
Alternate Success Scenario:

1. The buyer contacts more than one owner at a time.
2. The owner updates the details of the property.
3. The moderator can reject invalid properties.

Failure Scenario:

1. Login credentials are invalid / User already exists
2. Owner details displayed are not valid while the buyer tries to contact them.
3. Even after deleting the property from the database, it is still displayed in the list of properties.





5. Logic used for each module:

Login: New user details is inserted in the users database and during login,

Email and password is verified in the database using servlet. If credentials are valid, the user is redirected to the home page.

Sell Properties: The properties added by the owners and authenticated by moderators are displayed to all buyers.

Searchbox: The buyer can search for properties by city, state and pincode.

Filters: There are various filters for properties, ex:- looking to: rent/sell, by area slider and price range, with every selection, the query is modified to retrieve the relevant search results.

View my property: All properties of the owner which are added to the database are displayed.

7.USAGE OF NEW TOOLS:

The MySQL Database having user and property details is used, due to its flexible and cost effective method of managing information. For the Front end design, JSP and CSS are used in structuring and styling the web page while JS and AJAX is used to make it responsive. For the backend, we used Tomcat Server along with Java Servlets.

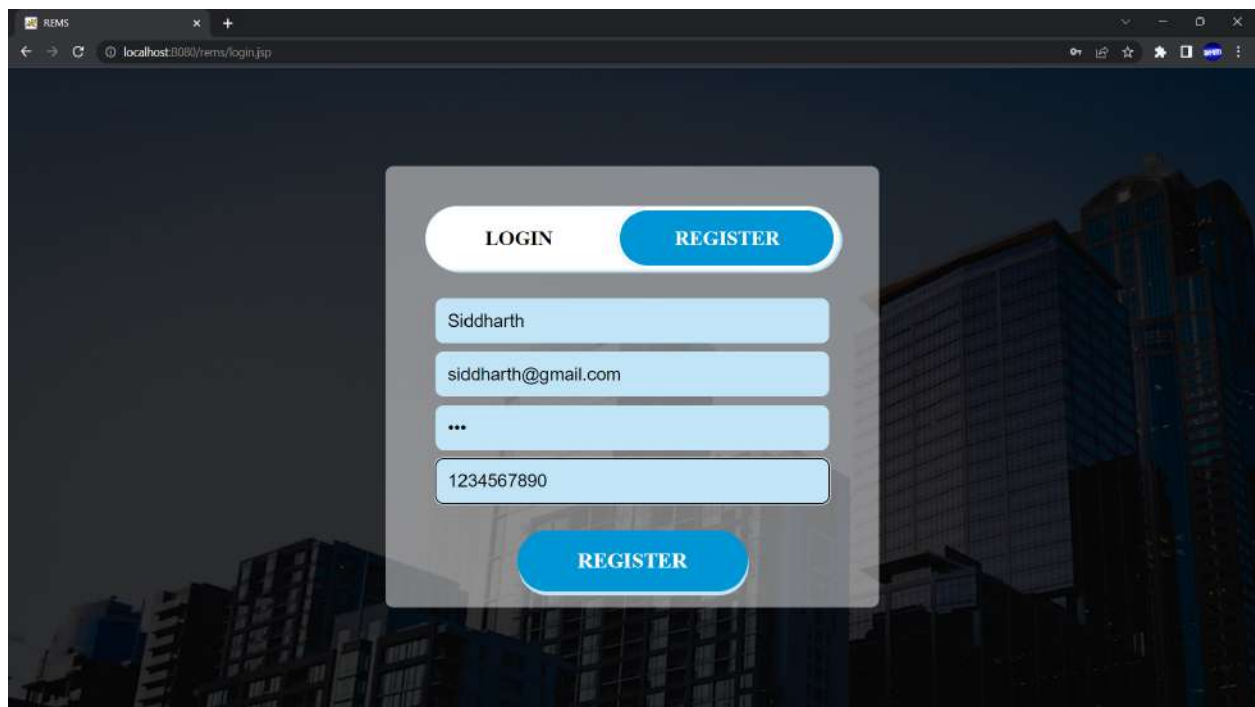
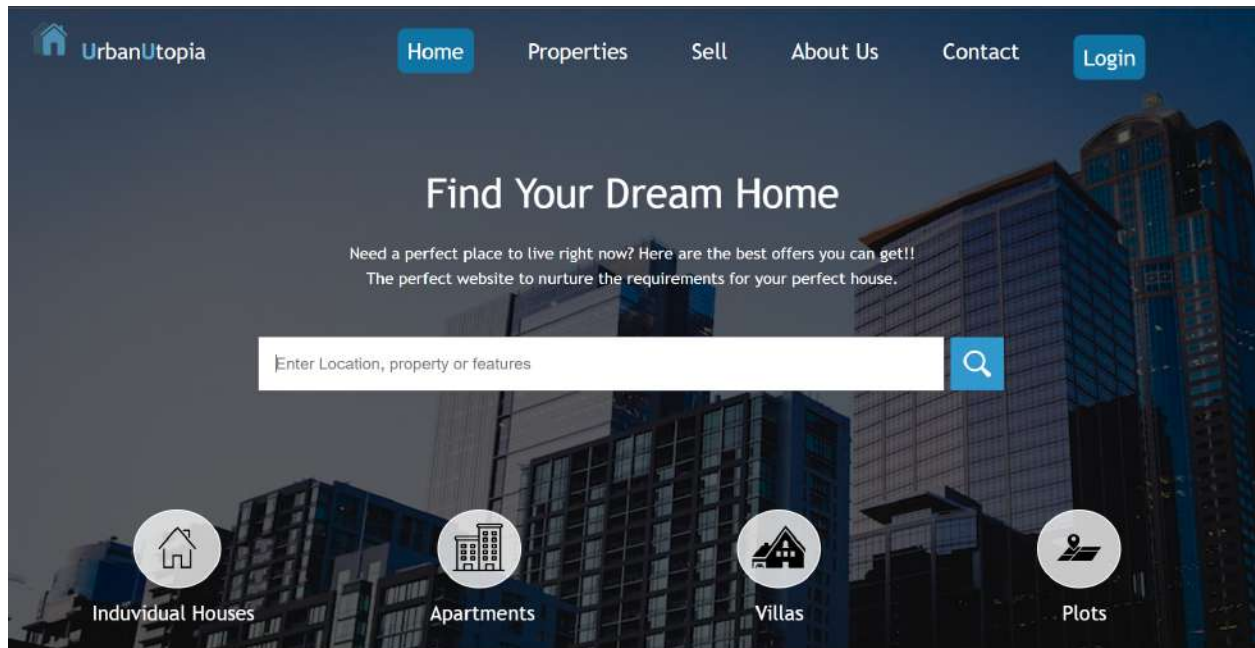
- HTML
- CSS
- SQL

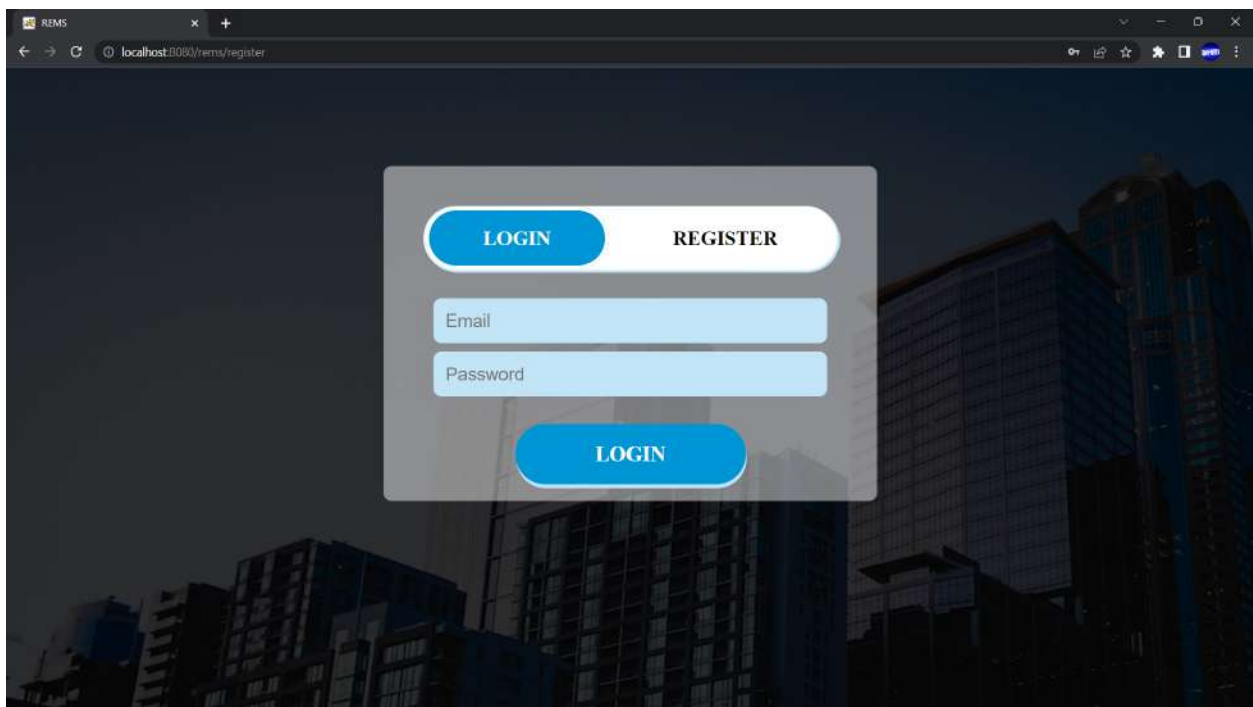
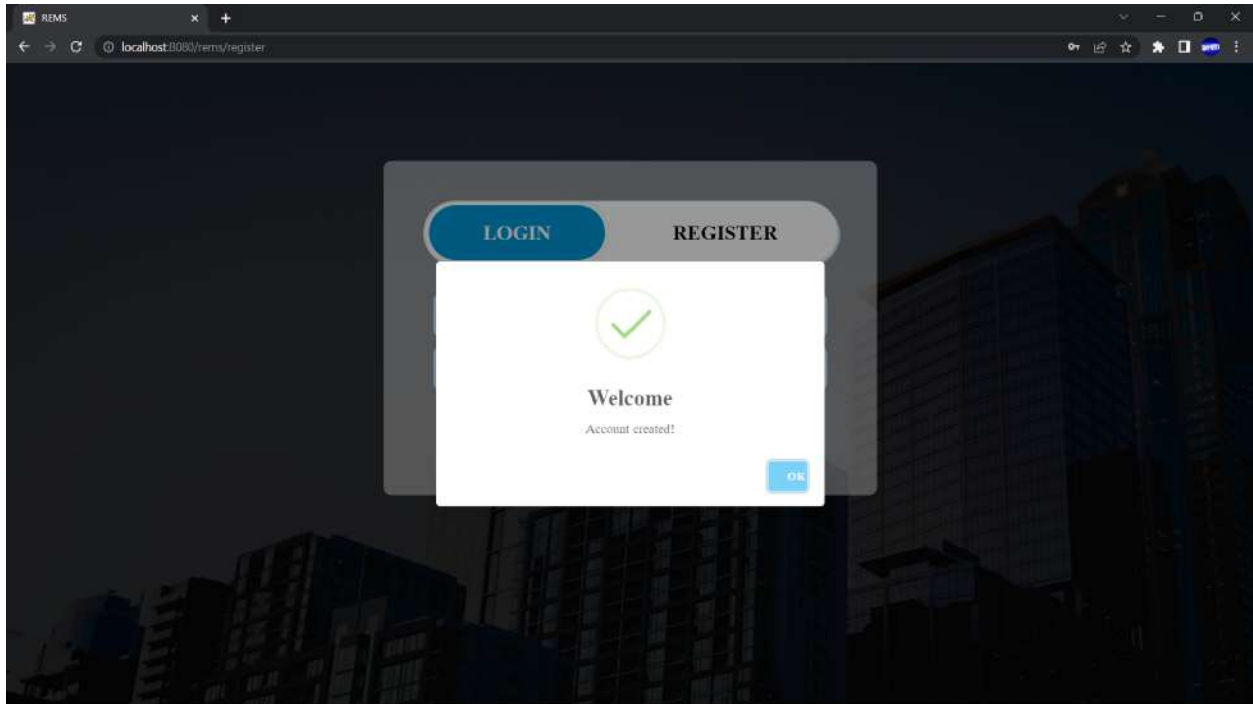
- JS
- AJAX
- JSP
- JAVA

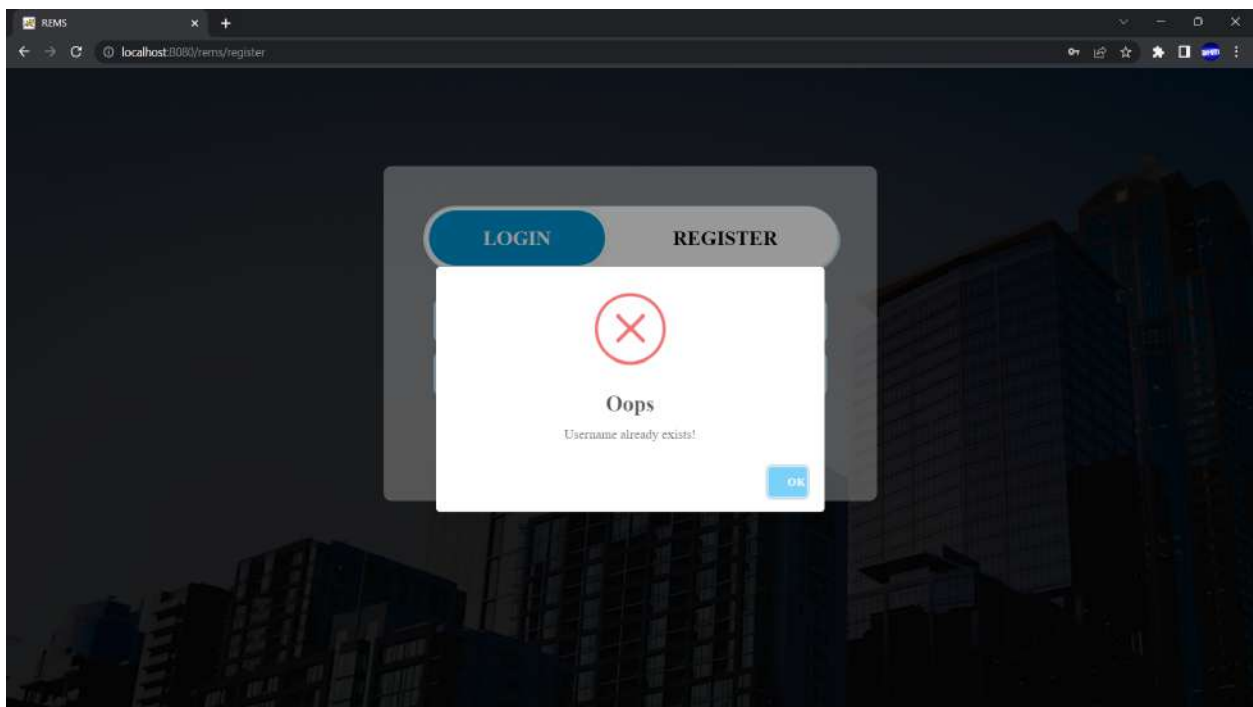
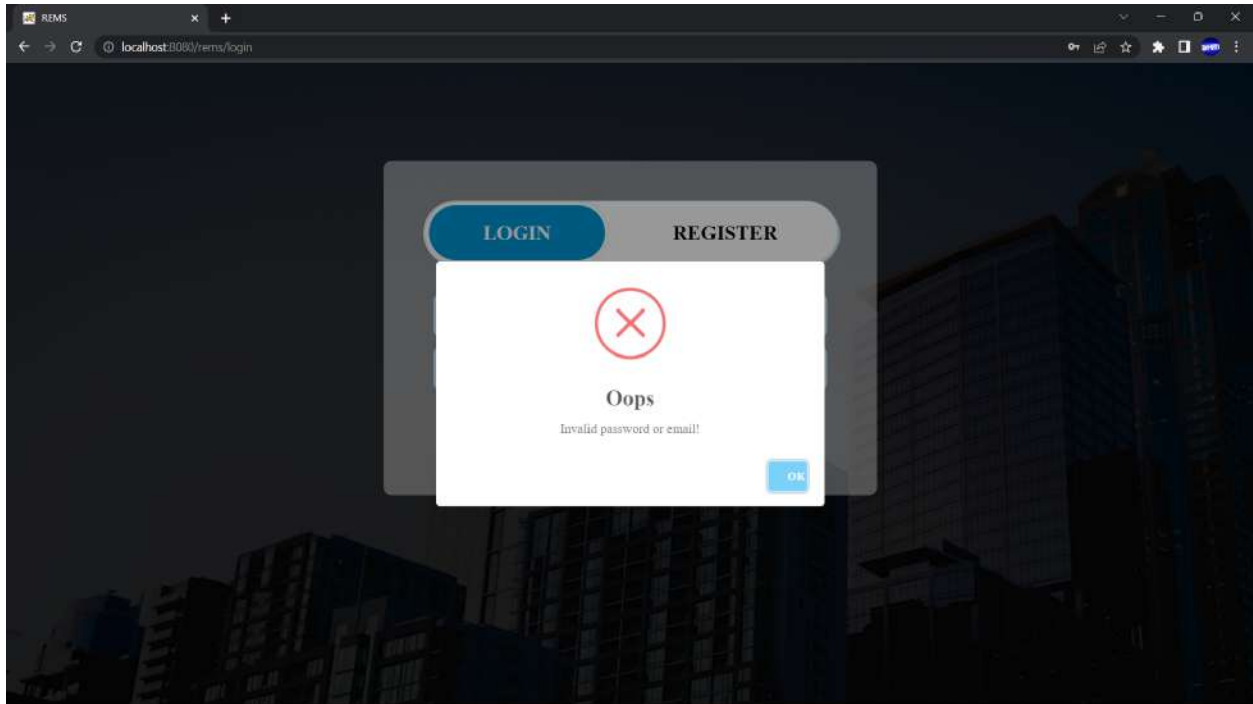
8.FEATURES NOT USED:

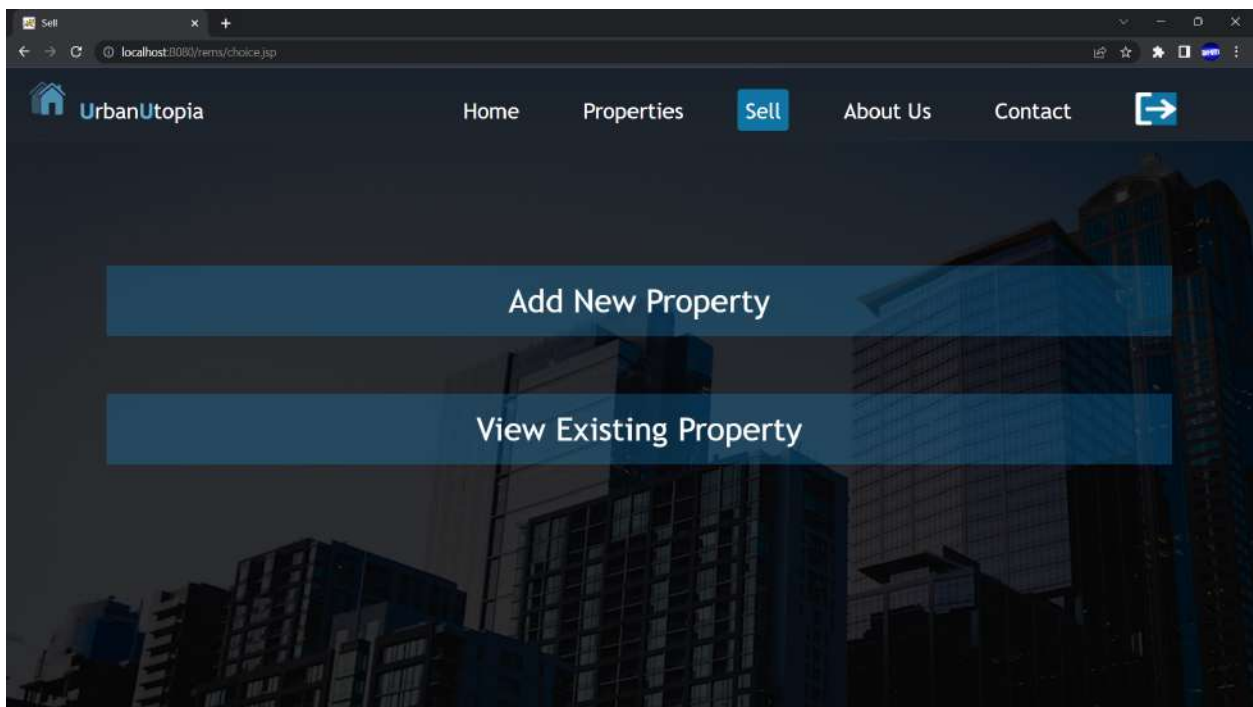
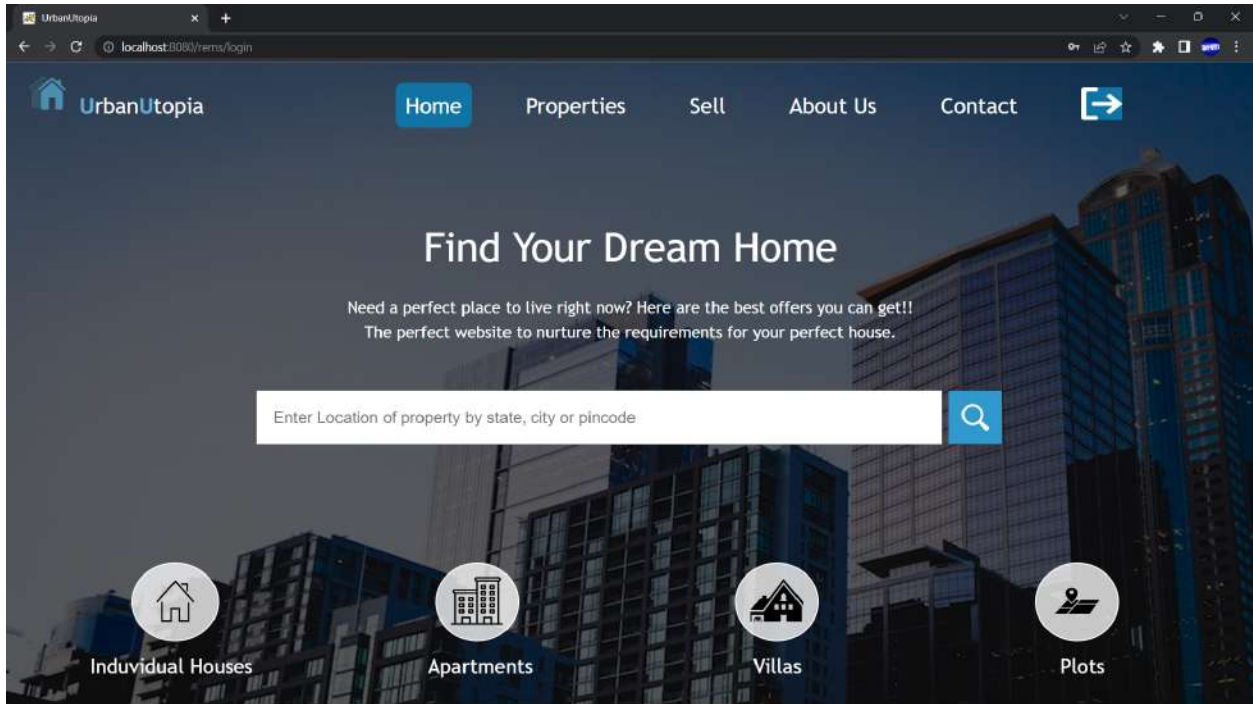
- React was not used to implement the front-end. It can be used for better adaptability.
- MongoDB Atlas was not used to host our database. It can be used in the future for better scalability.
- Our application was not deployed.
- Moderation was not included. It can be used in the future when the number of users increases.

Screenshots:









UrbanUtopia

Home Properties **Sell** About Us Contact

Property Name
Radiance

Mobile number
9876543210

Looking to
☒ Sell ☐ Rent

Type of property
☐ Individual House ☐ Apartment ☒ Villa ☐ Plot

Location Details

Door number

UrbanUtopia

Home Properties **Sell** About Us Contact

Location Details

Door number
12

Address of property
East mada street, Mylapore

City
Chennai

State
Tamil Nadu

Pincode
600004

Property Details

No. of bedrooms
☒ 1 ☐ 2 ☐ 3 ☐ 4

Sell

localhost:8080/remu/choice.jsp

UrbanUtopia

Home Properties **Sell** About Us Contact

Property Details

No. of bedrooms
☐ 1 ☐ 2 ☒ 3 ☐ 4

Area of plot

Furnishing
☐ Furnished ☒ Semi-Furnished ☐ Un-Furnished

Floor Details

Availability
☒ Ready to move ☐ Under Construction

Sell

My Drive - Google Drive

localhost:8080/remu/choice.jsp

UrbanUtopia

Home Properties **Sell** About Us Contact

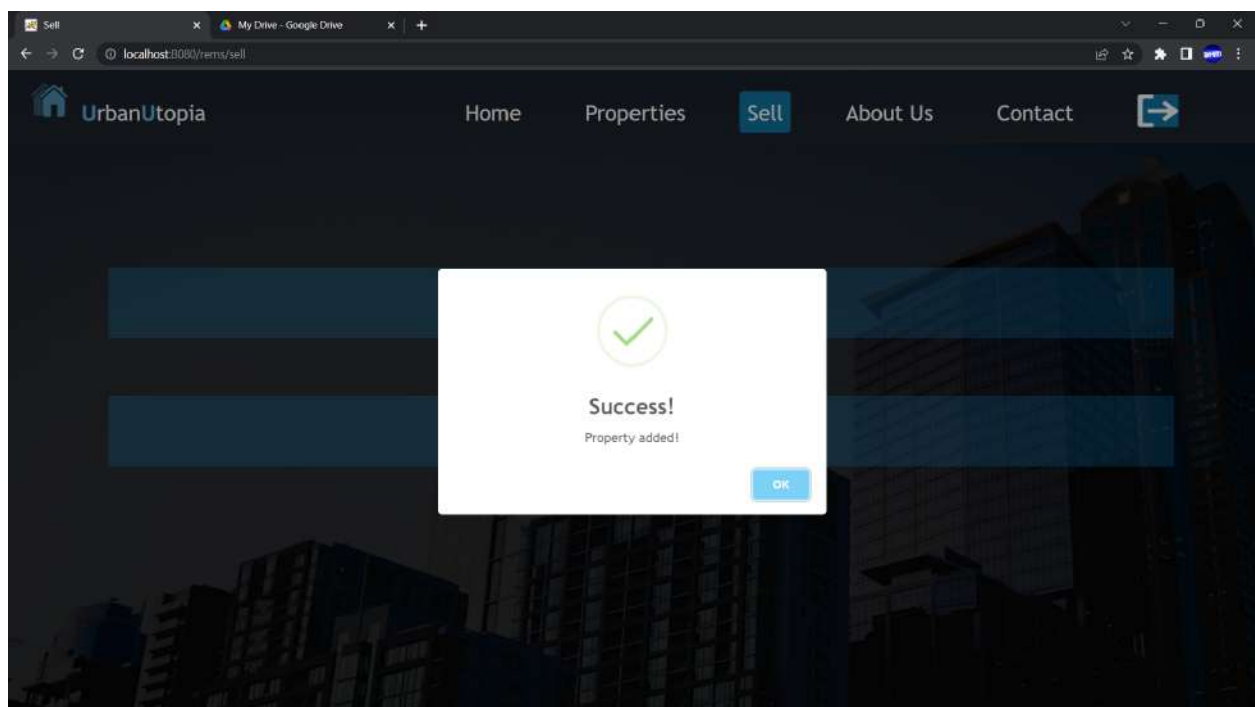
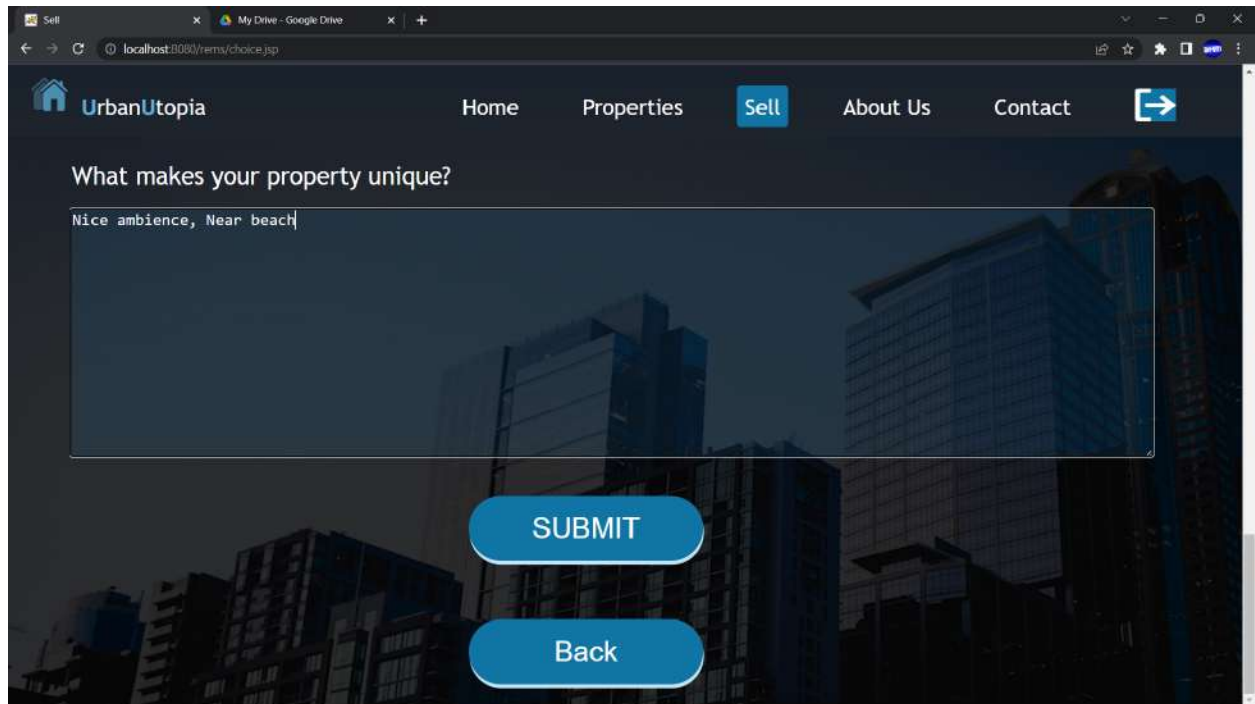
Property Images and Documents

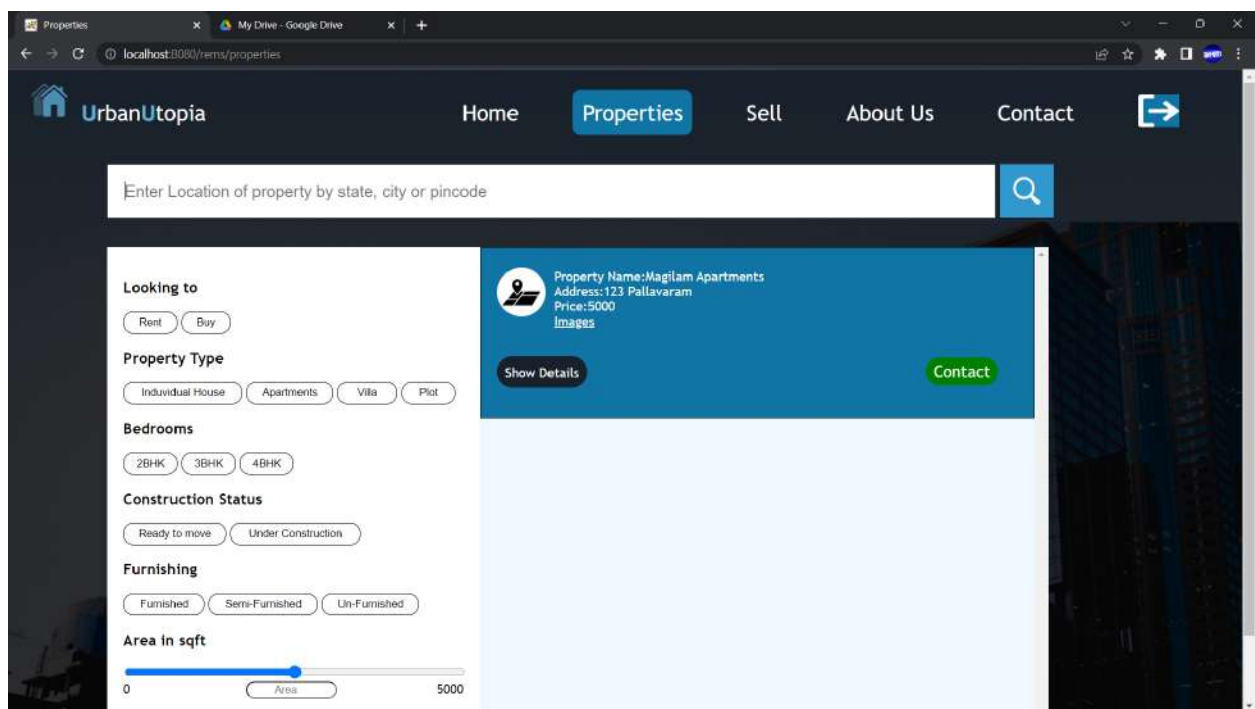
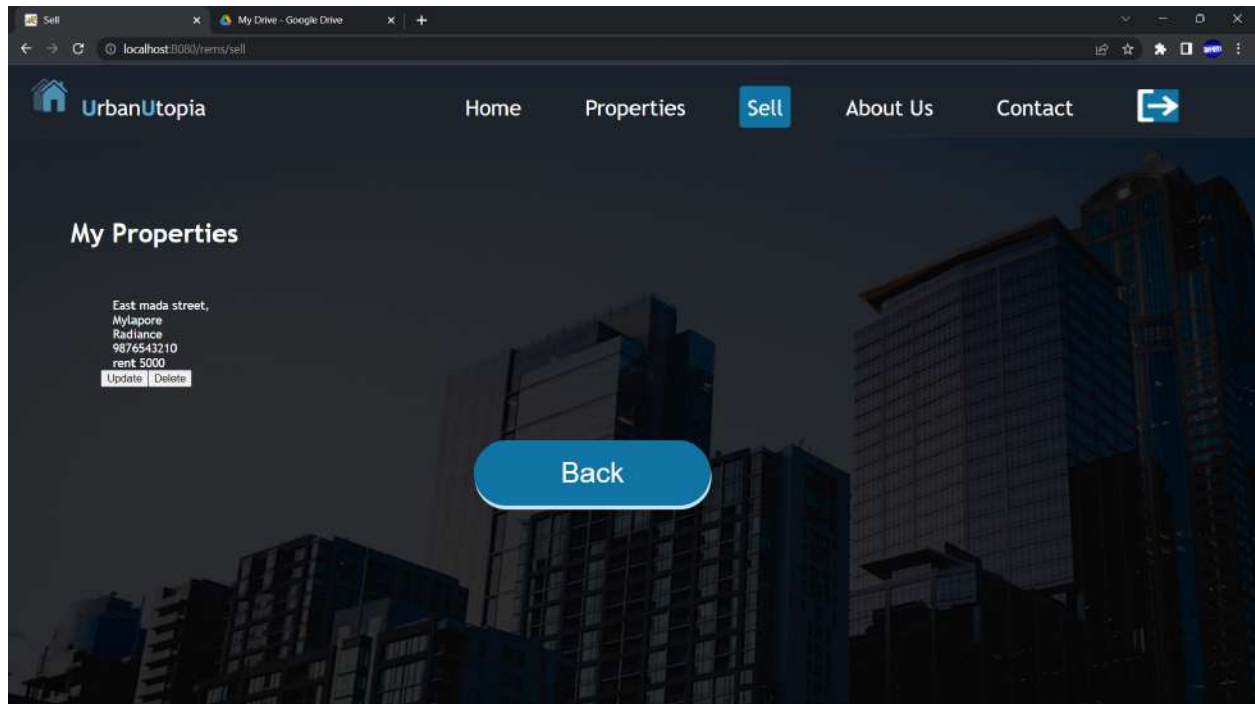
Upload google drive link containing the property images and documents

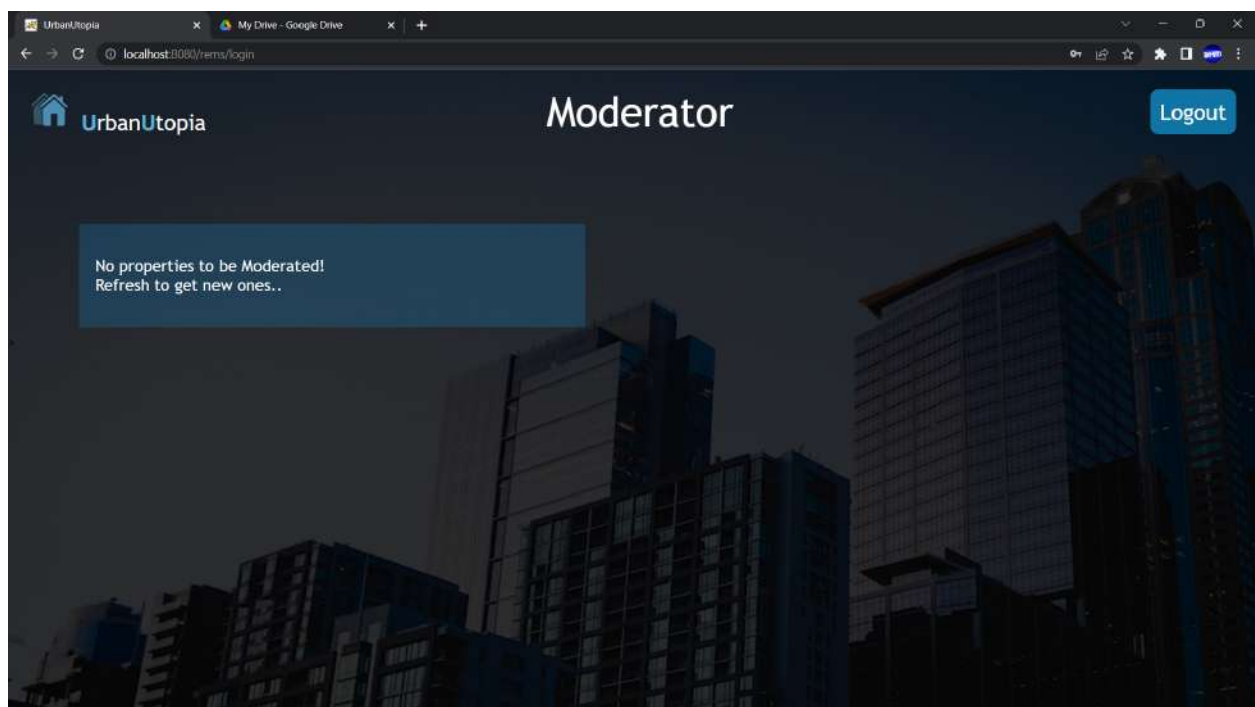
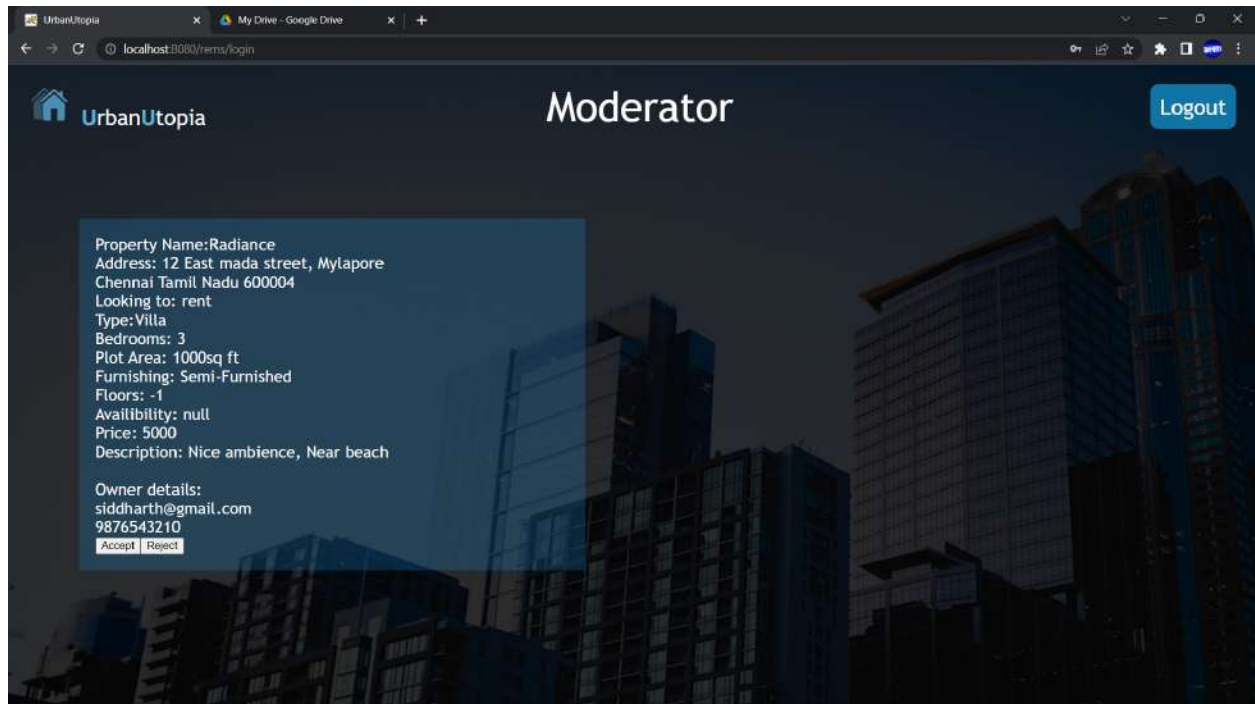
Property Price

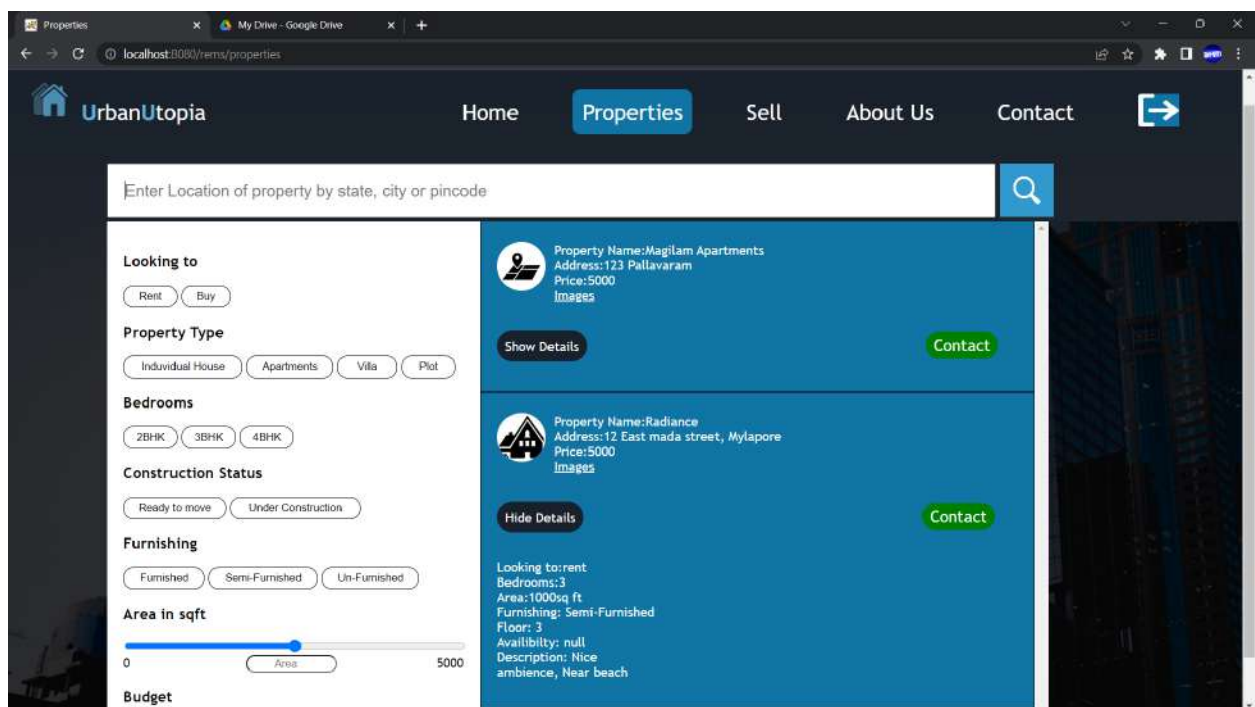
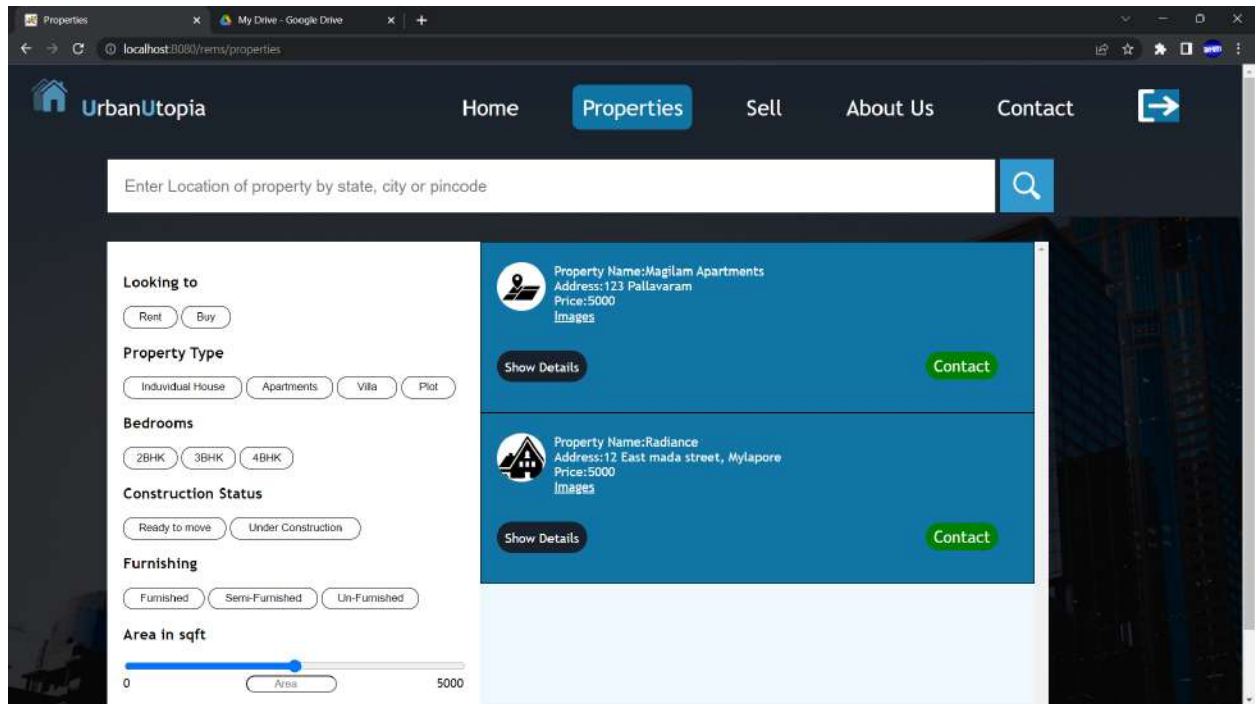
Pricing

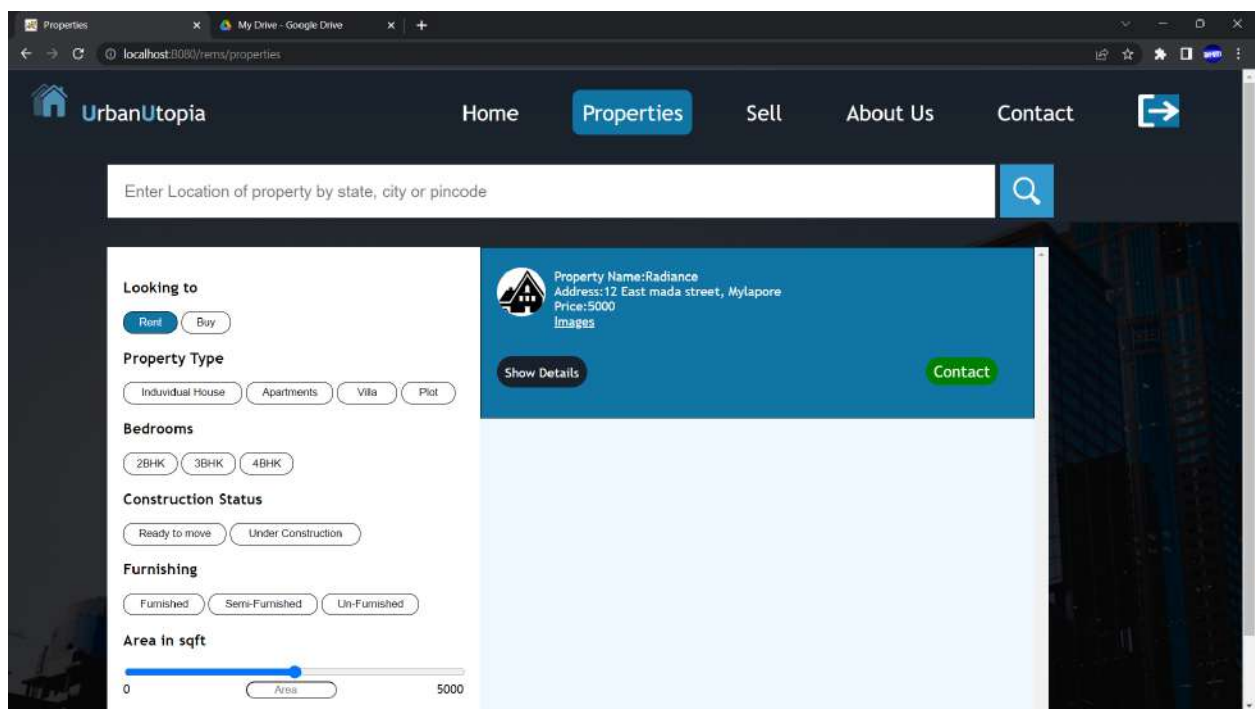
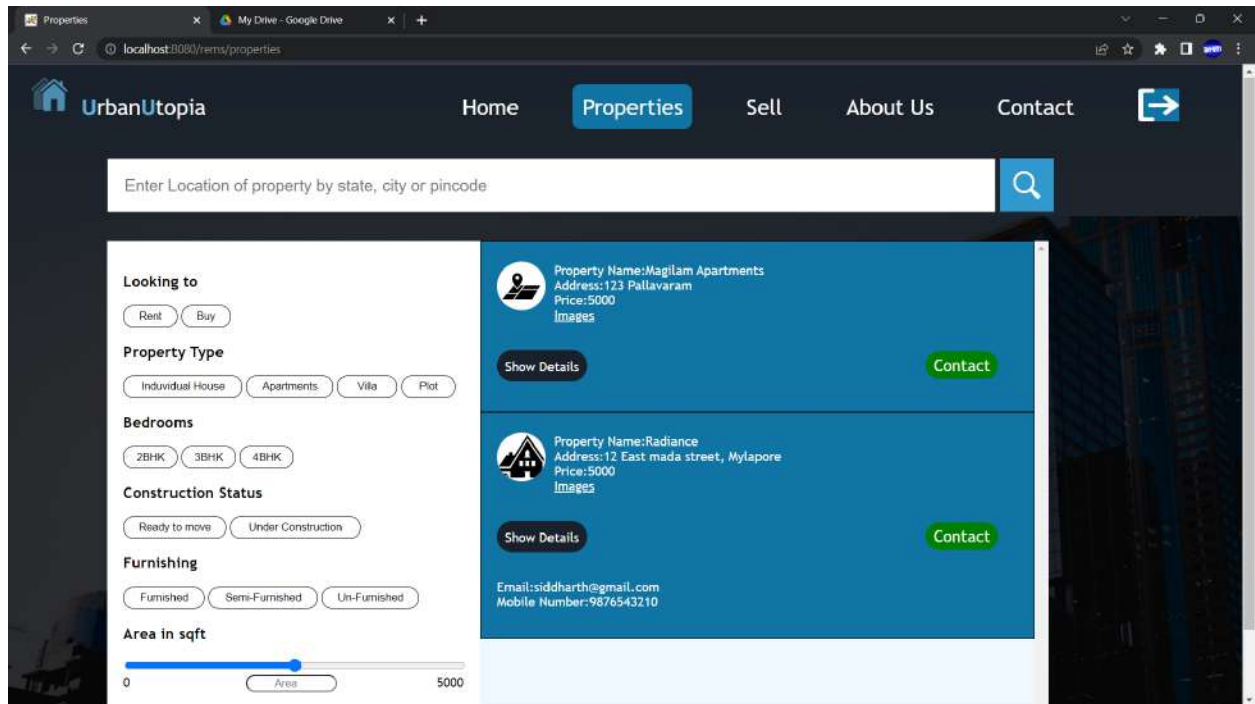
What makes your property unique?

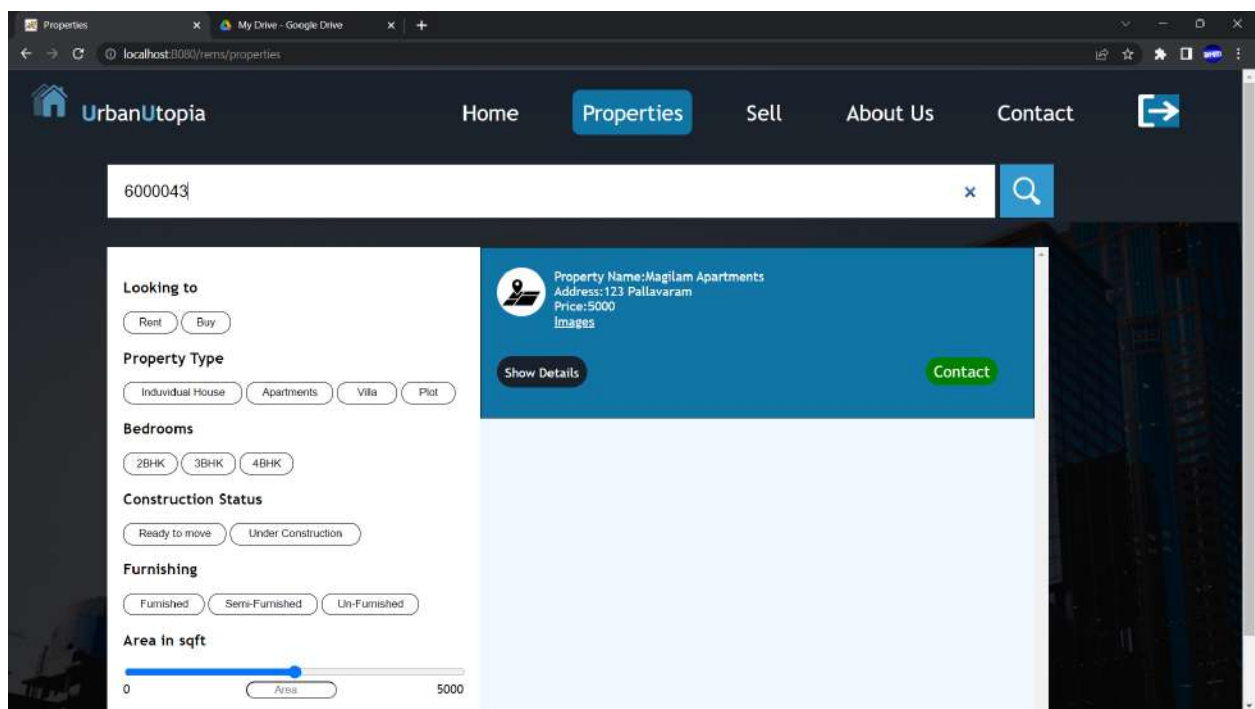
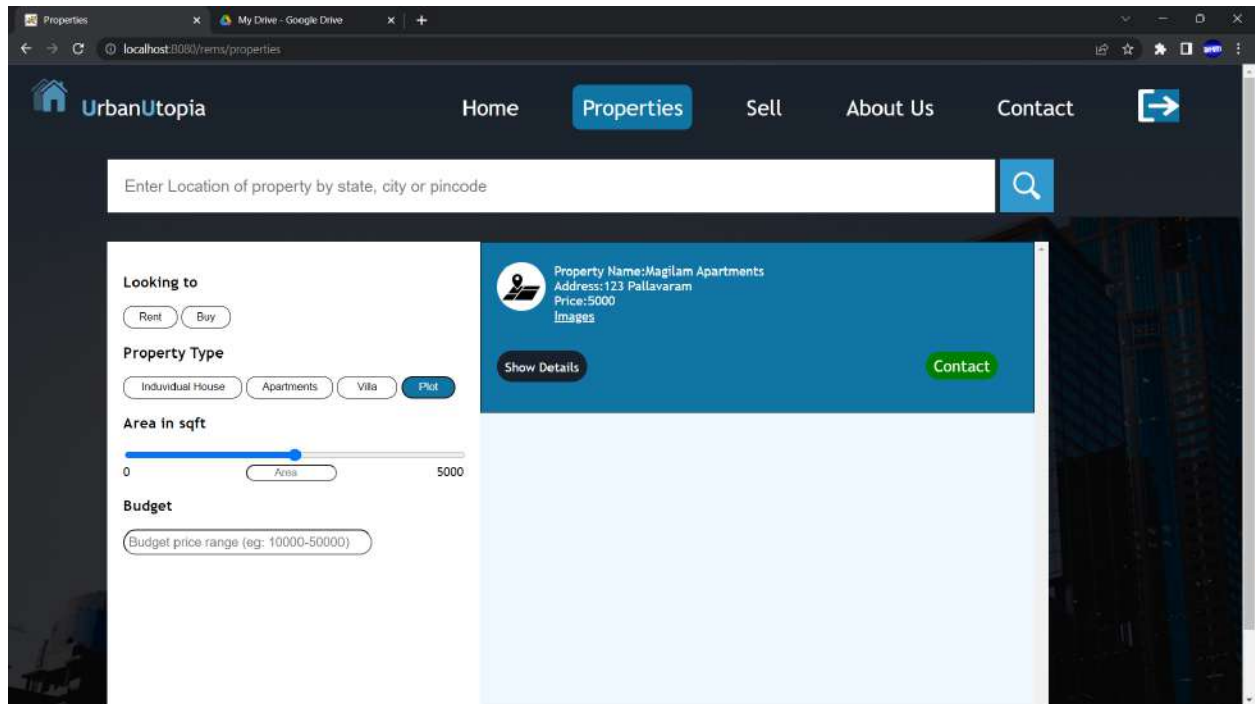


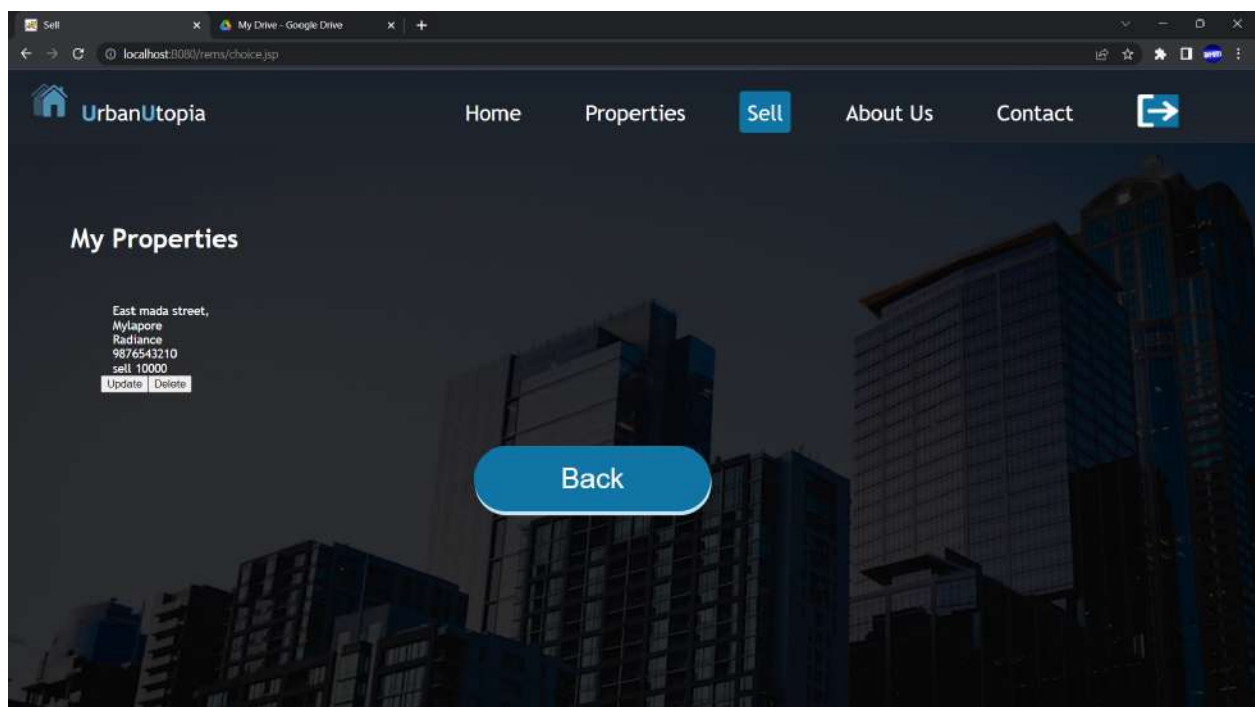
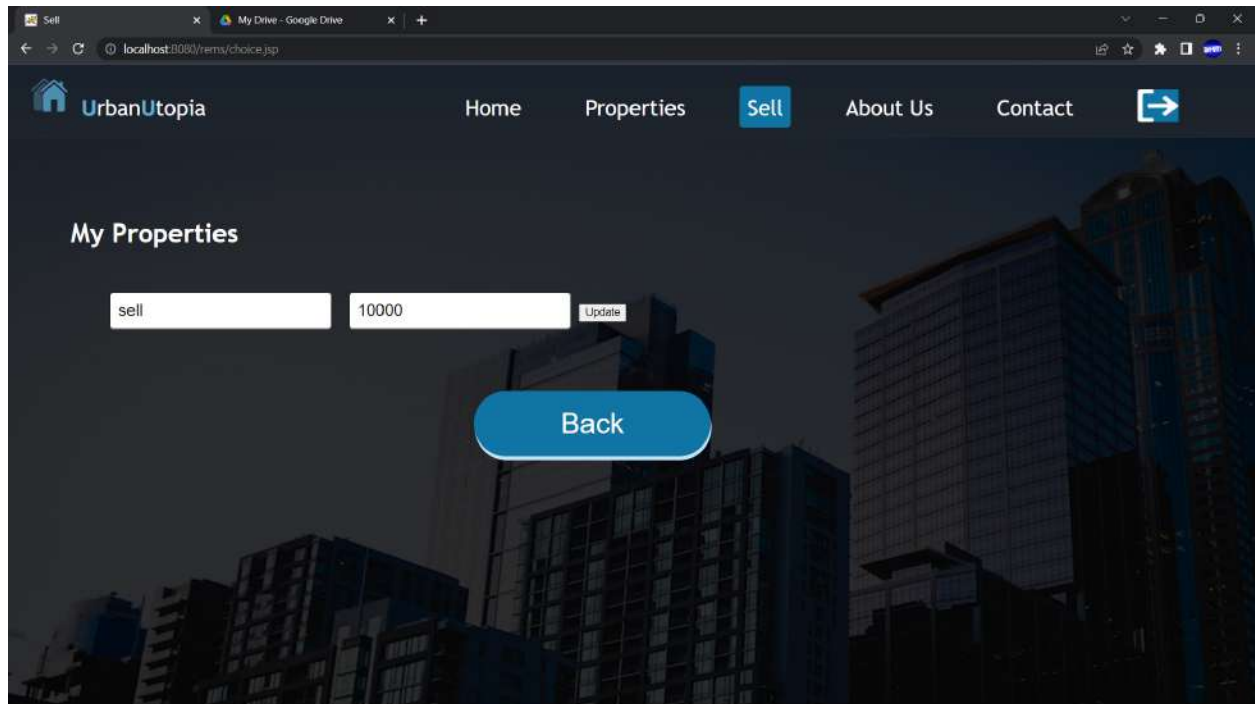


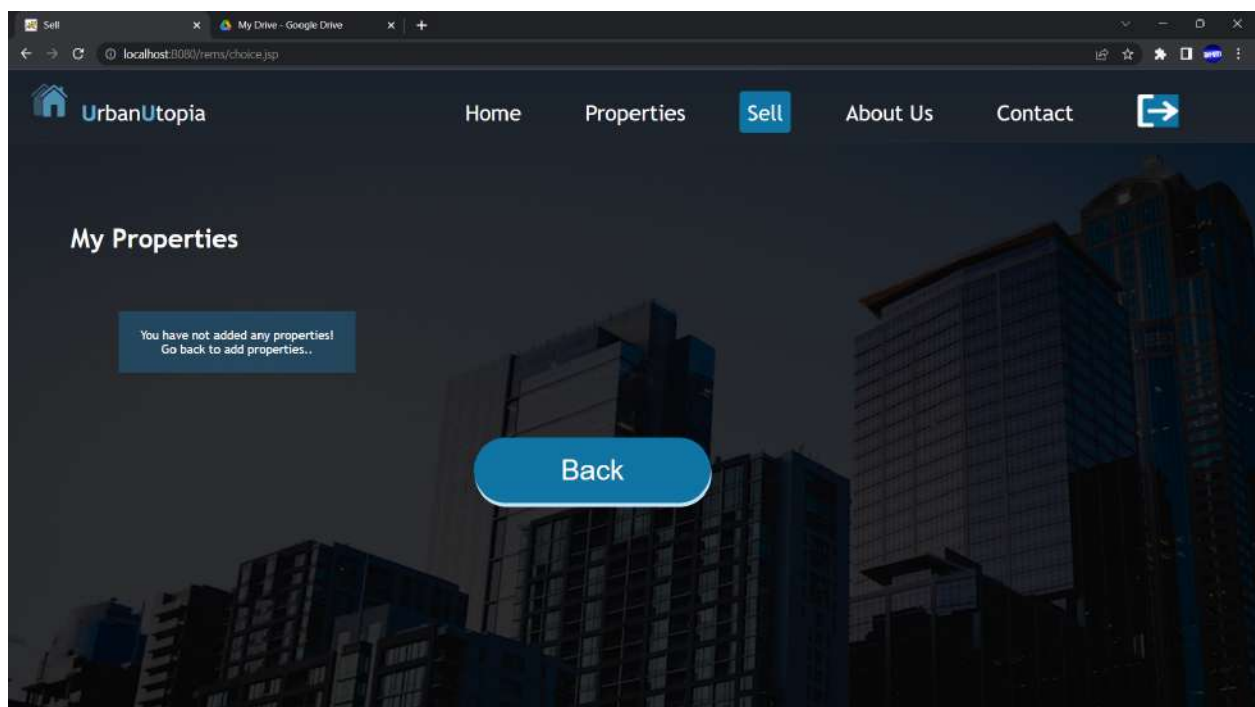
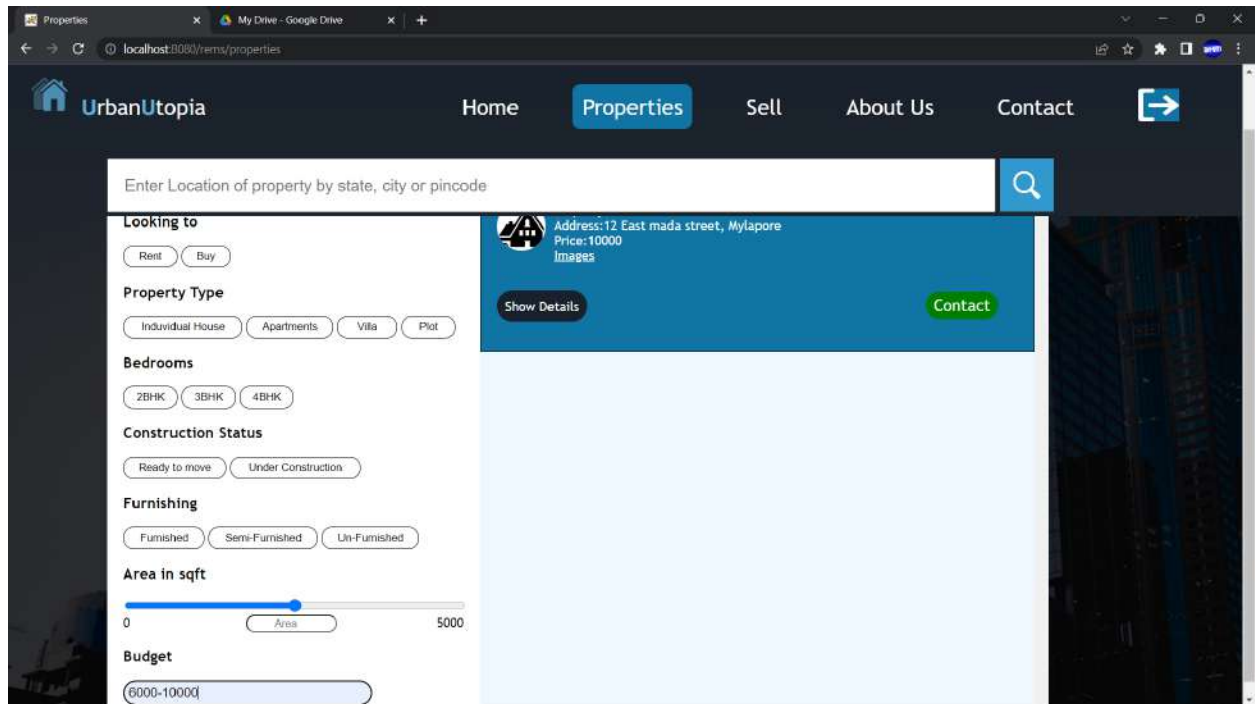


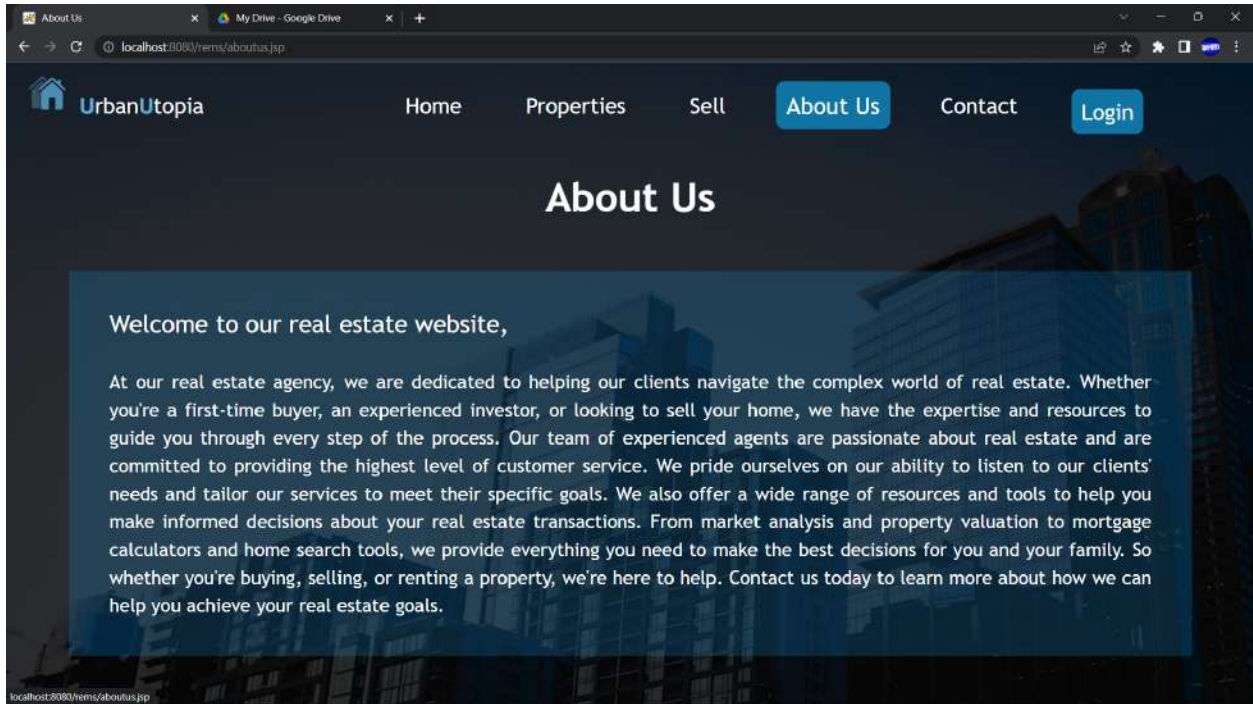


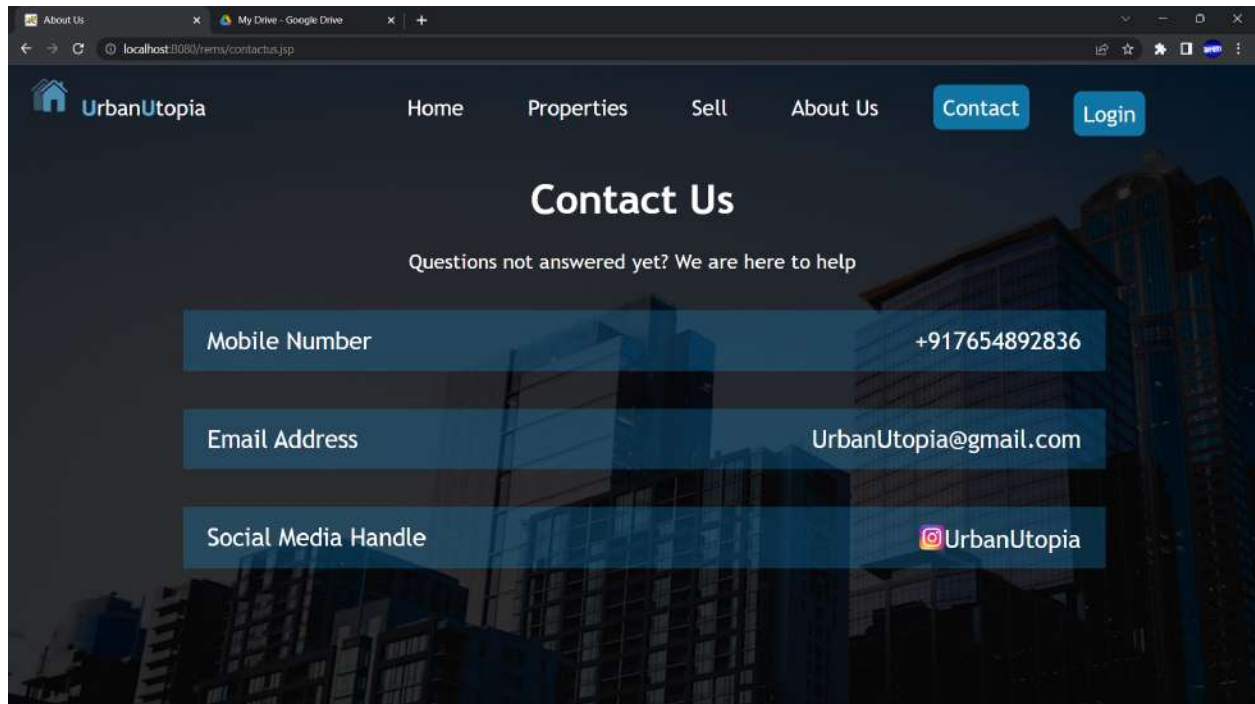












```
mysql> desc users;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| name  | varchar(25) | YES |  | NULL |  |
| email | varchar(30) | NO  | PRI | NULL |  |
| password | varchar(25) | YES |  | NULL |  |
| phone | bigint(20) | YES |  | NULL |  |
| moderator | varchar(3) | YES |  | no |  |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> desc sellproperty;
```

Field	Type	Null	Key	Default	Extra
name	varchar(20)	YES		NULL	
mobile_no	varchar(20)	YES		NULL	
looking_to	varchar(4)	YES		NULL	
property_type	varchar(20)	YES		NULL	
door_no	int(11)	YES		NULL	
address	varchar(70)	YES		NULL	
city	varchar(15)	YES		NULL	
state	varchar(15)	YES		NULL	
pincode	int(11)	YES		NULL	
bedrooms	int(11)	YES		NULL	
plot_area	int(11)	YES		NULL	
furnishing	varchar(20)	YES		NULL	
total_floors	int(11)	YES		NULL	
floor	int(11)	YES		NULL	
plotlength	int(11)	YES		NULL	
plotbreadth	int(11)	YES		NULL	
availability	varchar(20)	YES		NULL	
expected_price	int(11)	YES		NULL	
description	varchar(120)	YES		NULL	
status	varchar(20)	YES		NULL	
id	varchar(20)	NO	PRI	NULL	
location	varchar(50)	YES		NULL	
countid	varchar(10)	NO	PRI	NULL	
drive_link	varchar(170)	YES		NULL	

```
24 rows in set (0.00 sec)
```