



**Cégep de la Gaspésie
et des Îles**



RENT APARTMENT

It's Everything You Want To Stay!

Apartment Rental Application

**420-ENW-MT: PROJECT REPORT
(INTERNSHIP)**

By: Vidya Sagar Mulkanoori (1895253)
Alekyia Valisetty (1892744)
Anilkumar Vilasagar (1898475)
Akvinder Kaur (1895975)
Sagar Dawan (1895739)
Sukhdeepsinghbrar (1795558)

SUBMITTED TO:
Sakkaravarthi Ramanathan
Gilles-Philippe Grégoire
Sakshi Sharma
SilviyaPaskaleva
Harshkumar Dave

Software Requirements Specification

Document

Version: 1

Date: 24/05/2020

Table of Contents

1. Introduction	3-4
2. System Analysis	4
3. User Characterises	5
4. Specific Requirements	6-8
4.1 Functional Requirements	6-7
4.2 Non Functional Requirements	7-8
4.3 Other Requirements	8
4.3.1 Software Requirements	8
4.3.2 Hardware Requirements	8
5. Analysis Models	9-14
5.1 Use case Diagram	9-12
5.2 Class Diagram	13
5.3 ER Diagram	14
6. Prototype Screens	14-26
6.1 Splash Screen	15
6.2 Login Screen	16

6.3 Register Screen	17
6.4 Password Screen	18
6.5 Description Screen	19
6.6 User Dashboard	20-28
6.6.1 User Favourites	20
6.6.2 User Maps	21
6.6.3 User Search	22
6.6.4 User Home	23
6.6.5 User Property	24
6.6.6 User Contact Owner	25
6.6.7 User Account	26
6.6.8 User Settings	27
6.6.9 User Profile	28
6.7 Renter Dashboard	29-34
6.7.1 Renter Listing	29
6.7.2 Renter Home	30
6.7.3 Renter Add new	31
6.7.4 Renter Settings	32
6.7.5 Renter Account	33
6.7.6 Renter edit Profile	34
6.8 Admin Dashboard	35-39
6.8.1 Admin Users	35
6.8.2 Admin Property	36
6.8.3 Admin Map	37
6.8.4 Admin Home	38
6.8.5 Admin Delete User	39

1. Introduction

The Apartment rental project is designed for allowing the users in finding the right type of apartment according to their lifestyle and also finding the right type of renter for the apartment or property. It needs a good amount of effort and research in getting a right apartment. Finding accommodation in Quebec can be a real challenge, especially in the cities and during this pandemic. One essential step in the process of getting hold of an apartment or room is to write an outstanding rental application. Due to this, being a mobile based application, any user can get trusted and transparent apartments for rent on a Google maps. The User can then click on any apartment to display additional information about the apartment such as the price, apartment type, address and description. The process of choosing and leasing an apartment is sometimes a difficult, challenging, and very stressful task.

The Gaspésie system group requires an application to help the user of finding an apartment. If you move frequently, or you're unsure of how long you'll be staying in the area, renting could be the better choice. This application can be used by such kind of people. The user can also add a filter so that only apartments in a given price range are displayed and also apartments at a particular location are displayed. The user can also save the apartments to a favourites list to review them later. On the other hand, a User must create an account if they wish to put an apartment for rent. When logged in, the user has access to an extra feature which allows them to add an apartment on the map with the information mentioned above. Logged in users can also update their profiles for contact information.

In order to make the best possible decision for yourself, you must take into consideration a variety of issues. The two most important factors of finding an apartment are the amount that you are willing to spend on rent each month and the neighbourhood in which the apartment is located. Choosing a place to live is one of the

most important decisions that you will have to make. Before you start your search for an apartment you will want to decide how much is acceptable to spend on rent. You will want to take into consideration the amenities that you are looking for in an apartment, as these will cost extra per month. User can also login as an admin if the user is registered as one. Admin has ability to verify the apartment listings, delete apartment listings which consist of fake data/ incomplete data. Modify user's profile which consist of in-appropriate images.

2. System Analysis

1. Title: Analysis and Design of an Apartment Rental Mobile Application

2. The Problem Statement: The Gaspésie Systems Group wants you to develop an app to ease the process of finding an apartment for people living anywhere in Québec.

(a) Overall goals of the app allows the user to visualize efficient apartments for rent on a map. The user can then click any apartment to display additional information such as price, address, description, contact information and more.

(b) Scope of the project IN-Scope: This will include only users and their requests regarding apartments. The Apartment Rental Mobile Application includes to find an apartment for people living anywhere in Quebec.

3. User Characterises

There are 3 types of users of the System, 1) User, 2) Renter and 3) Admin.

The user should be able to perform the following functions:

- User can view the posted apartments.
- User can contact the Renter by Phone.
- User can contact by Email.
- User can find the direction in Google Maps.
- User can filter the Apartment according to the price and location.
- User can update his profile details.

The function of the Renter are as follows:

- Renter can post an apartment by giving exact location.
- Renter can post multiple apartment listings.
- Renter can view all his posted listings in Google Maps.

The Admin Manages the system and their functions are as follows:

- Admin has ability to either approve or Reject.
- Admin can manage user filtering fake data.

4. Specific Requirements

4.1. Functional Requirements

The system will be password-protected. Apartment Rental Application will be a multi-user system where every user must log in. This application needs to perform the following functions:

Requirement ID	Requirement Statement	MUST/COULD/SHOULD/WOULD	Comments
FR1	The application must have login screen	MUST	App must have the login page
FR2	The application must have register screen and users must register by entering details	MUST	User shall have to register before using the application
FR3	Users must login application with valid credentials	MUST	The system shall have to take the valid details to login
FR4	Users could manage their profile	COULD	The system shall edit the profile to store in the database
FR5	Users could manage Forgot password	COULD	The system shall manage to recreate the password to store in the database
FR6	Renter can add apartments with details which include (price, location, rooms, etc)	MUST	The system shall add the data according to the renter requirement
FR7	Renter could edit apartments	COULD	The system shall edit details to store in database
FR8	Renter could delete apartments	COULD	The system shall delete data and store in the

			database
FR9	Renter could see user requests	COULD	The system shall allow renter to see user requests
FR10	Users could view apartments	COULD	The system shall allow user to view apartments
FR11	Users could select and view desired apartment details	COULD	The system shall allow users to select and view desired apartment from the database
FR12	Users could search based on price criteria	COULD	The system shall allow user to search according to the price they want to.

4.2. Non-functional requirements

Requirements on usability, reliability, performance, supportability, security, recovery, interface, implementation, operation, and legal. It describes aspects of the system that are concerned with how the system provides the functional requirements. They are:

- NF1-The application must be a Mobile-based application.
- NF2-Menus should be organized in a hierarchical manner (usability)
- NF3-The application should be user friendly
- NF4-Application must be with readable content
- NF5-The application should be reliable to perform the business, i.e. when a user performs some important action it should be acknowledged with confirmation.
- NF6-The application must be providing a help and support menu in all interfaces for the user to interact with the system.
- NF7-All the application data should be secured and be encrypted.
- NF8-The application should have response time for every instruction conducted within time of 60 seconds.

- NF9- Application should be able to maintain mass number of customers on the server at once without crashing.
- NF10- Application must be able to transform data quickly (Speedy performance).
- NF11-Application should be backed up daily. (Back up)

4.3. Other Requirements

4.3.1. Software Requirements

For developing the application the following are the software requirements

1. Android Development Tool version 3.6.3
2. Android SDK for Android ADT (Recent version).

Technologies and Languages used to Develop

1. Android
2. Java
3. Front end: XML
4. Back end Database: Firebase

Debugger and Emulator

1. Android Dalvik Debug Monitor service
2. Android Emulator (Android Virtual Device)

4.3.2. Hardware Requirements

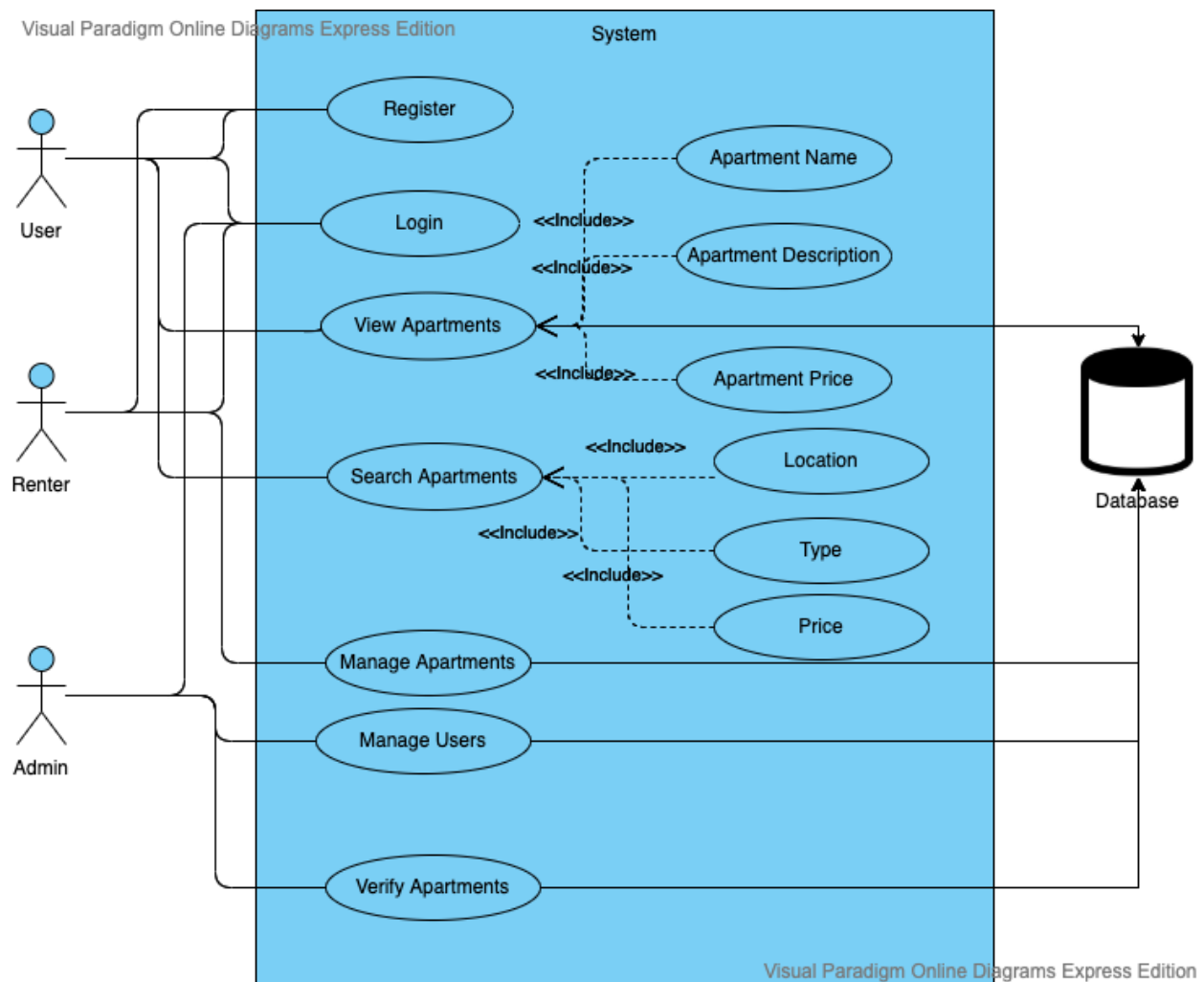
For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM: 256 MB
- Space on Hard Disk: minimum 512MB

5. ANALYSIS MODELS

5.1 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.



Description

UC-1: REGISTER

Primary Actor	User/Renter/Admin
Secondary Actor	Database
Pre-Conditions	User is not registered
Success Guarantee	User is successfully registered with application
Main Success	<ul style="list-style-type: none">• User Opens the App• User enters the details in Signup page• User select his Role• User Register with App
Extensions	User is not created if email is already exists in the Database

UC-2: LOGIN

Primary Actor	User/Admin
Secondary Actor	Database
Pre-Conditions	User is Registered
Success Guarantee	User successfully signed in
Main Success	<ul style="list-style-type: none">• User enters his credentials• App verifies the credentials• User is successfully redirected to his respective dashboard
Extensions	<ul style="list-style-type: none">• Minimum Password length is 6 character• Error is displayed when user enters wrong password

UC-3: VIEW APARTMENT

Primary Actor	User
Secondary Actor	Renter/Admin/Database
Pre-Conditions	User is Logged in
Success Guarantee	User gets all the details of the Apartment
Main Success	<ul style="list-style-type: none">• User can get all the details of listed apartment• User can contact the listing Renter

UC-4: SEARCH APARTMENT

Primary Actor	User
Secondary Actor	Database
Pre-Conditions	User is Logged in
Success Guarantee	User act all the filtered results
Main Success	<ul style="list-style-type: none">• User filter according to price and location• User get all the results filtered
Extensions	“NO RESULTS” is displayed No apartment available in the filter

UC-5: MANAGE APARTMENT

Primary Actor	Renter
Secondary Actor	Database
Pre-Conditions	Renter is logged in
Success Guarantee	Renter has Successfully modified the listing data
Main Success	<ul style="list-style-type: none">• Renter can add new apartment• Renter can Remove the apartment• Renter can Edit the Details

UC-6: MANAGE USER

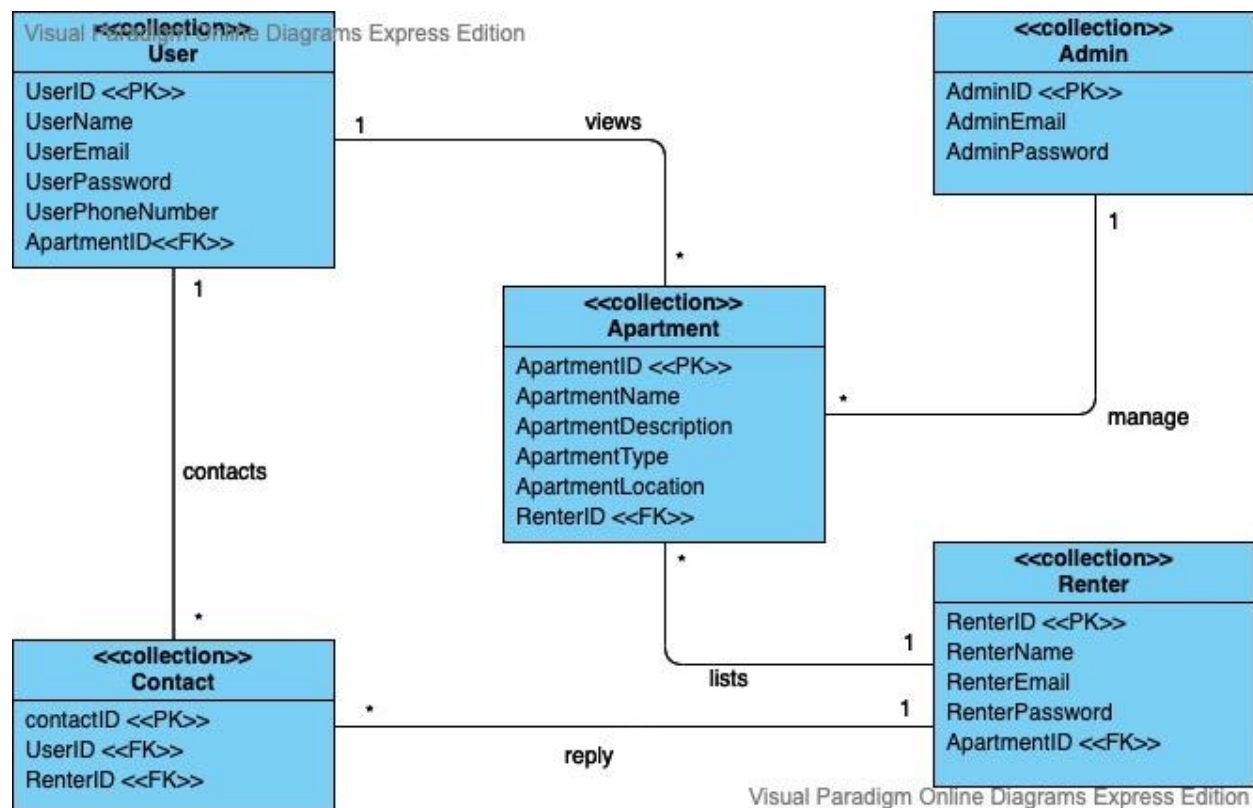
Primary Actor	Admin
Secondary Actor	Database
Pre-Conditions	Admin is Logged in
Success Guarantee	Admin is able to modify user data
Main Success	<ul style="list-style-type: none">• Admin can modify user details• Admin can Approve or Remove Profile with fake data.

UC-7: VERIFY APARTMENT

Primary Actor	Admin
Secondary Actor	Database
Pre-Conditions	Admin is logged in
Success Guarantee	Admin has Successfully Verified Apartment listing
Main Success	<ul style="list-style-type: none">• Admin can view all the posted apartment• Admin can filter apartments with fake data• Admin can modify the details

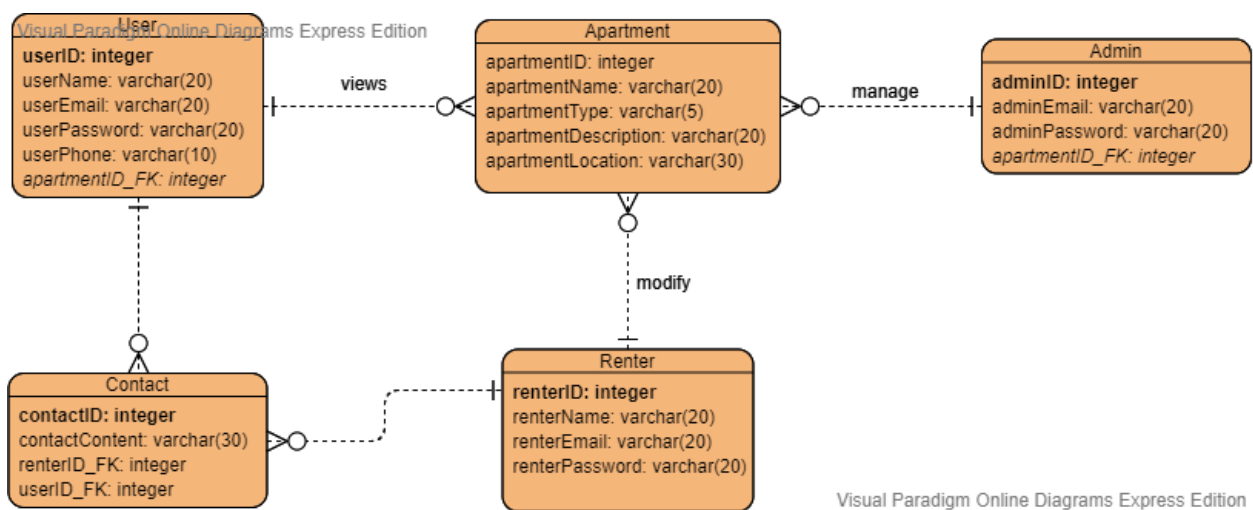
5.2 Class Diagram

Class diagrams show the classes of the system, their interrelationships and the operations and attributes of the classes. Class diagrams are used for a wide variety of purposes. It shows relationships between classes, objects, attributes, and operations. Class Diagram defines the types of objects in the system and the different types of relationships that exist among them. It gives a high-level view of an application.



5.3 ER Diagram

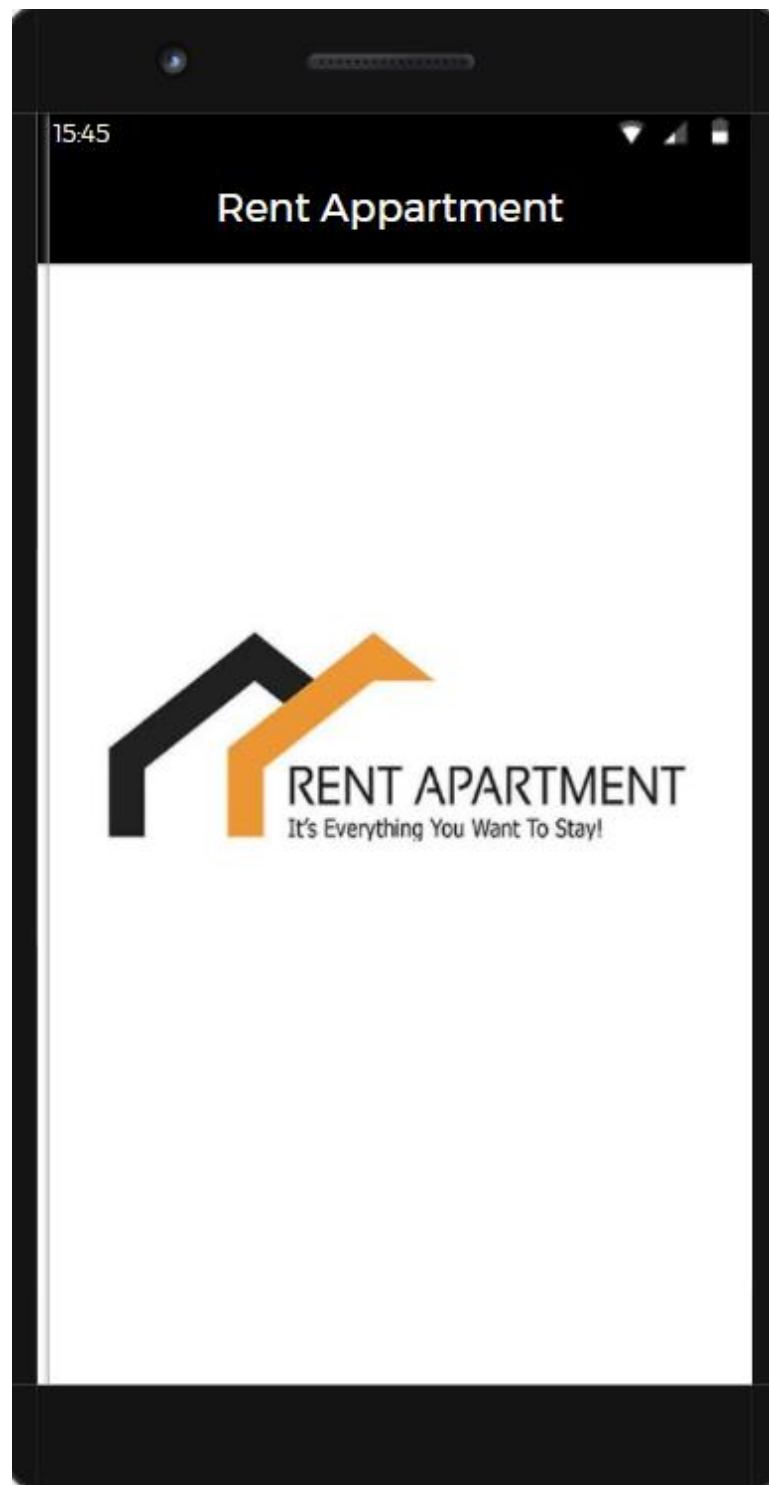
An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems.



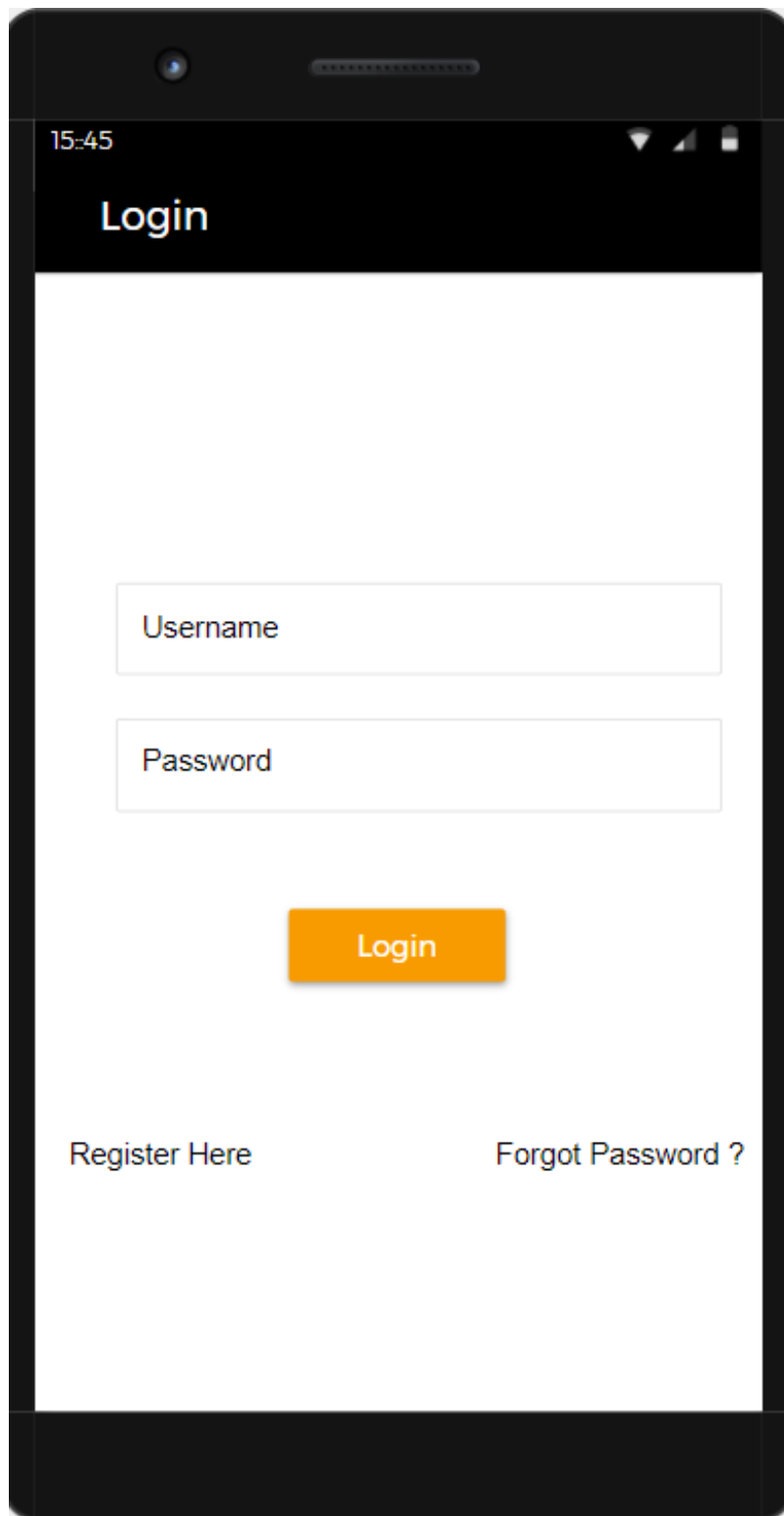
6. Prototype Screens

Prototypes are interactive mock-up of websites and mobile applications designed to look and feel like the real thing. They're much quicker to make, and although they aren't full-featured, they allow us to spot mistakes before we build the real thing. The Rent Apartment application is designed with a simple user interface with the purpose of providing great usability for users. As shown in the figure below.

6.1 Splash Screen



6.2 Login Screen



A mobile application login screen mockup. The screen has a dark header bar at the top with a status bar above it showing the time 15:45 and signal/battery icons. The word "Login" is displayed in white text on the dark header. Below the header is a large white rectangular area containing the login form. The form consists of two white input fields with thin grey borders. The first field is labeled "Username" and the second is labeled "Password". Below these fields is a solid orange button with the word "Login" in white text. At the bottom of the white area, there are two links: "Register Here" on the left and "Forgot Password ?" on the right. The entire screen is framed by a dark border representing the phone's bezel.

15:45

Login

Username

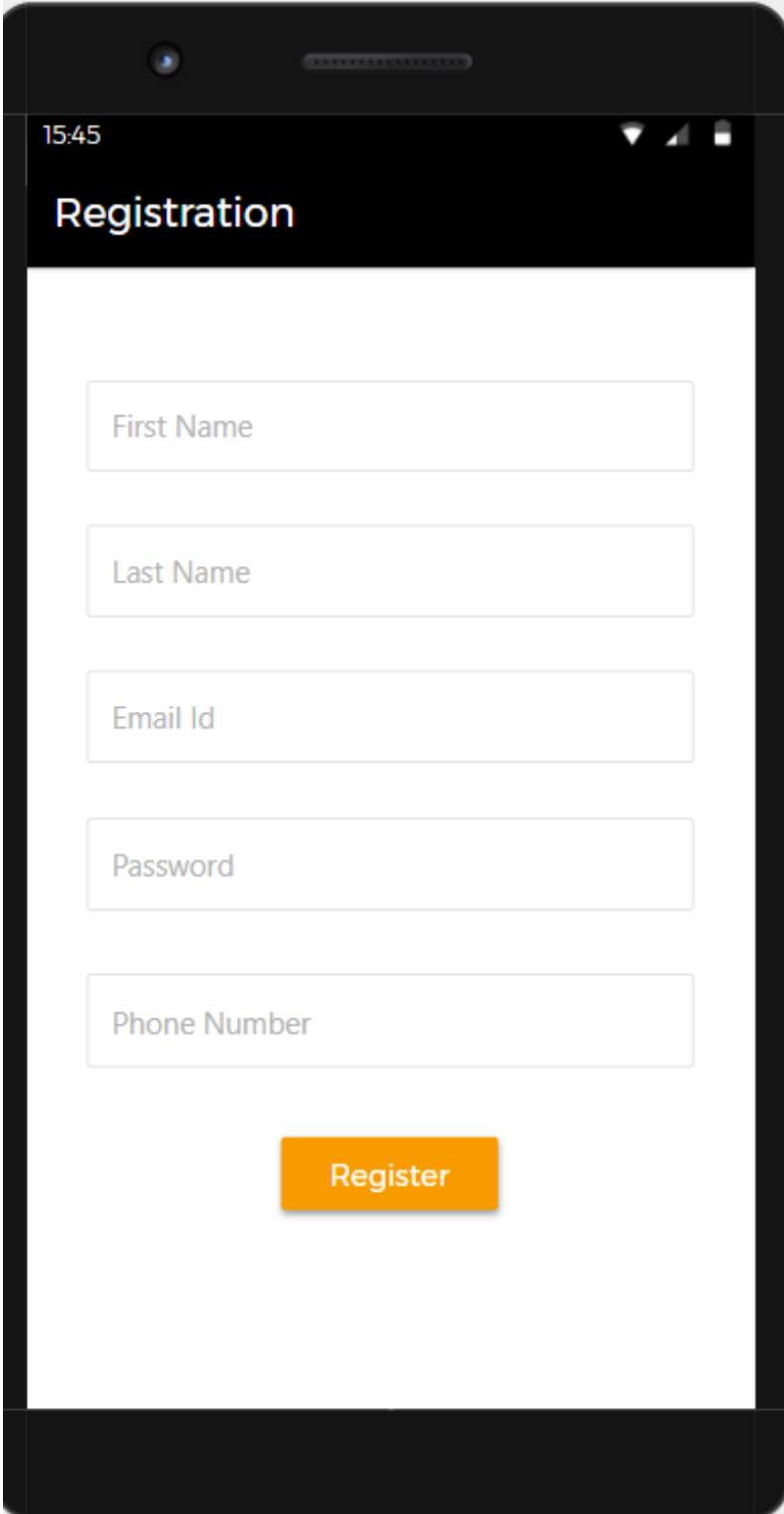
Password

Login

Register Here

Forgot Password ?

6.3 Register Screen

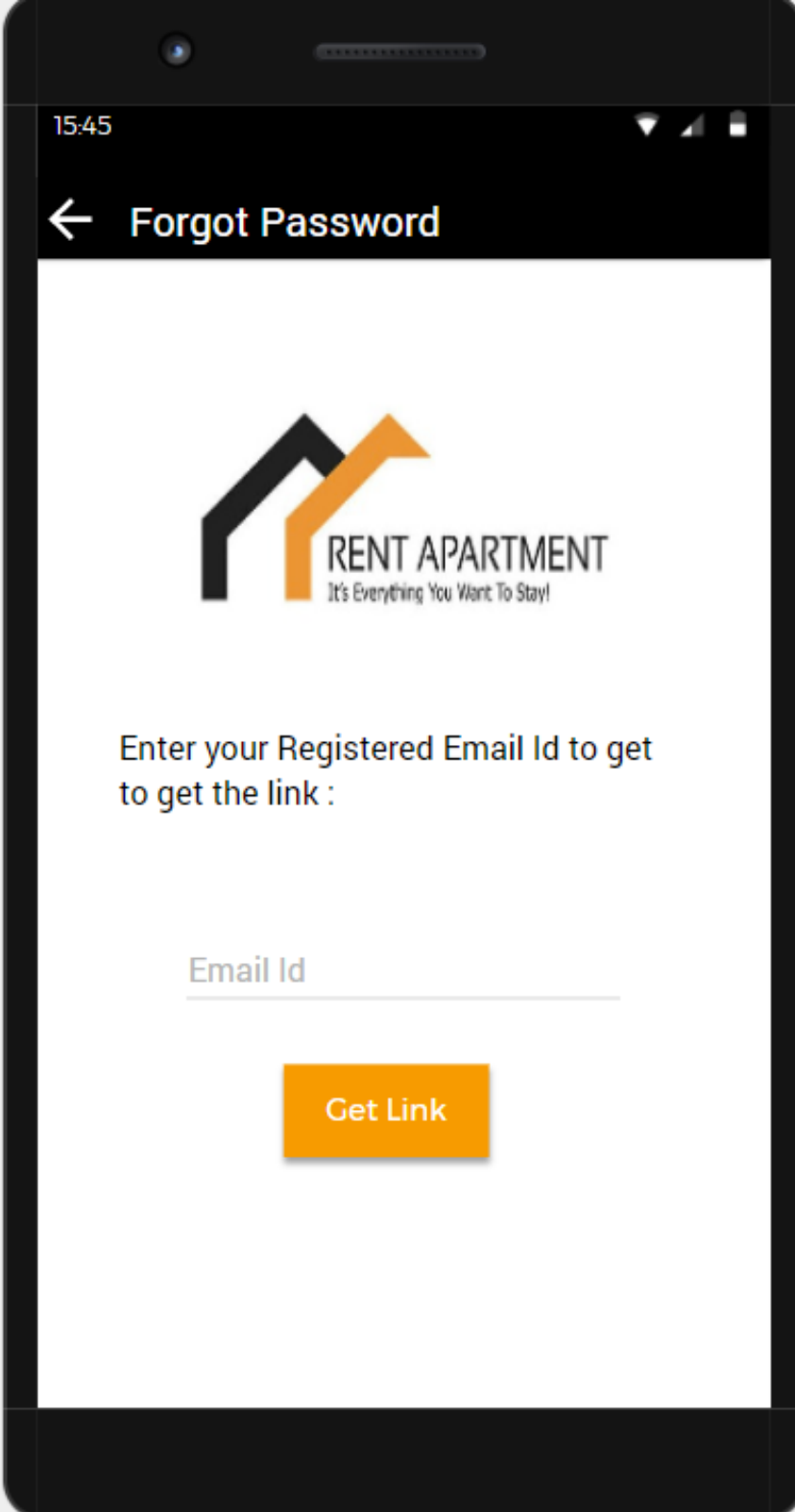
A mobile application registration screen. At the top, a black status bar shows the time 15:45 and signal icons. Below this is a black header bar with the word "Registration" in white. The main area is white and contains five text input fields stacked vertically, each with a light gray placeholder label: "First Name", "Last Name", "Email Id", "Password", and "Phone Number". At the bottom of the form is a solid orange button with the word "Register" in white text. The entire screen is framed by a black border.

15:45

Registration


Register

6.4 Password Reset

A mobile application interface for a password reset process. The screen is framed by a black border. At the top, a black header bar contains a white back arrow and the text "Forgot Password". Above this header, a status bar shows the time "15:45" and signal icons. The main content area is white. It features a logo for "RENT APARTMENT" with the tagline "It's Everything You Want To Stay!". Below the logo, the text "Enter your Registered Email Id to get to get the link :" is displayed. Underneath this text is a text input field with the placeholder "Email Id". At the bottom of the input area is an orange button with the text "Get Link".

15:45

← Forgot Password



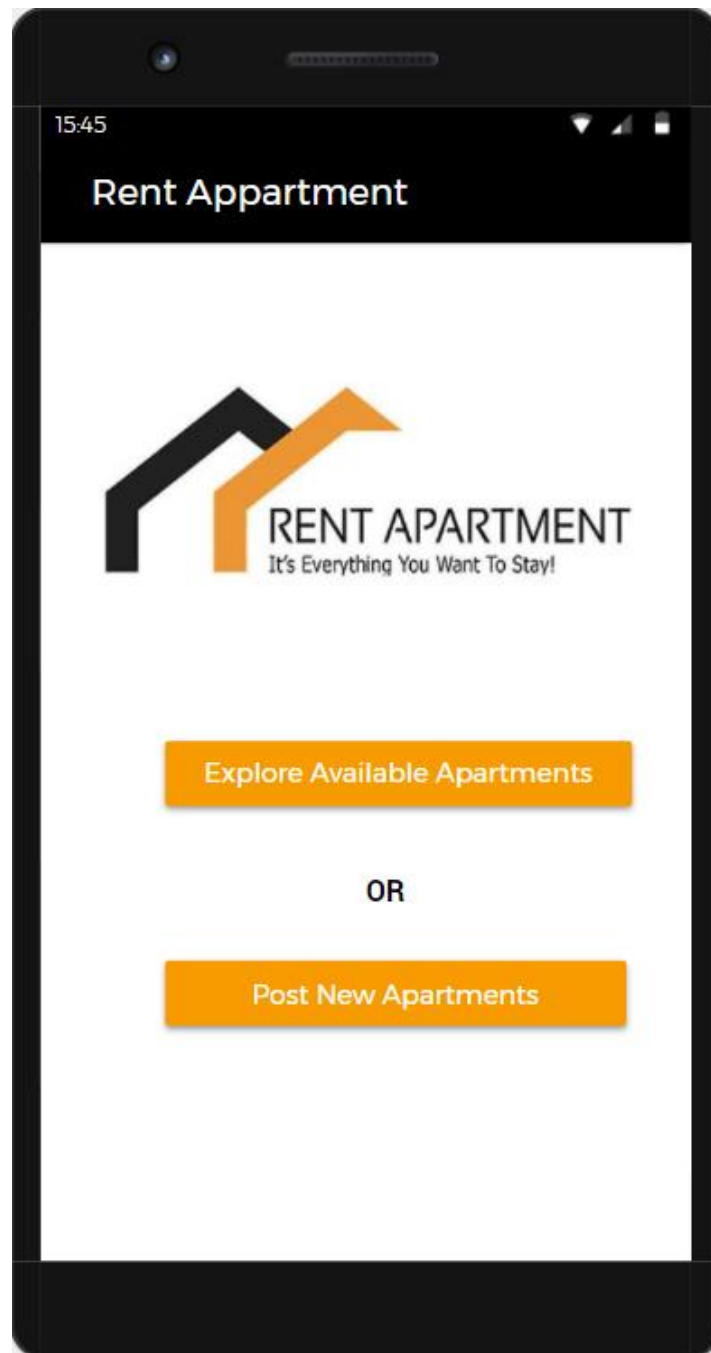
RENT APARTMENT
It's Everything You Want To Stay!

Enter your Registered Email Id to get to get the link :

Email Id

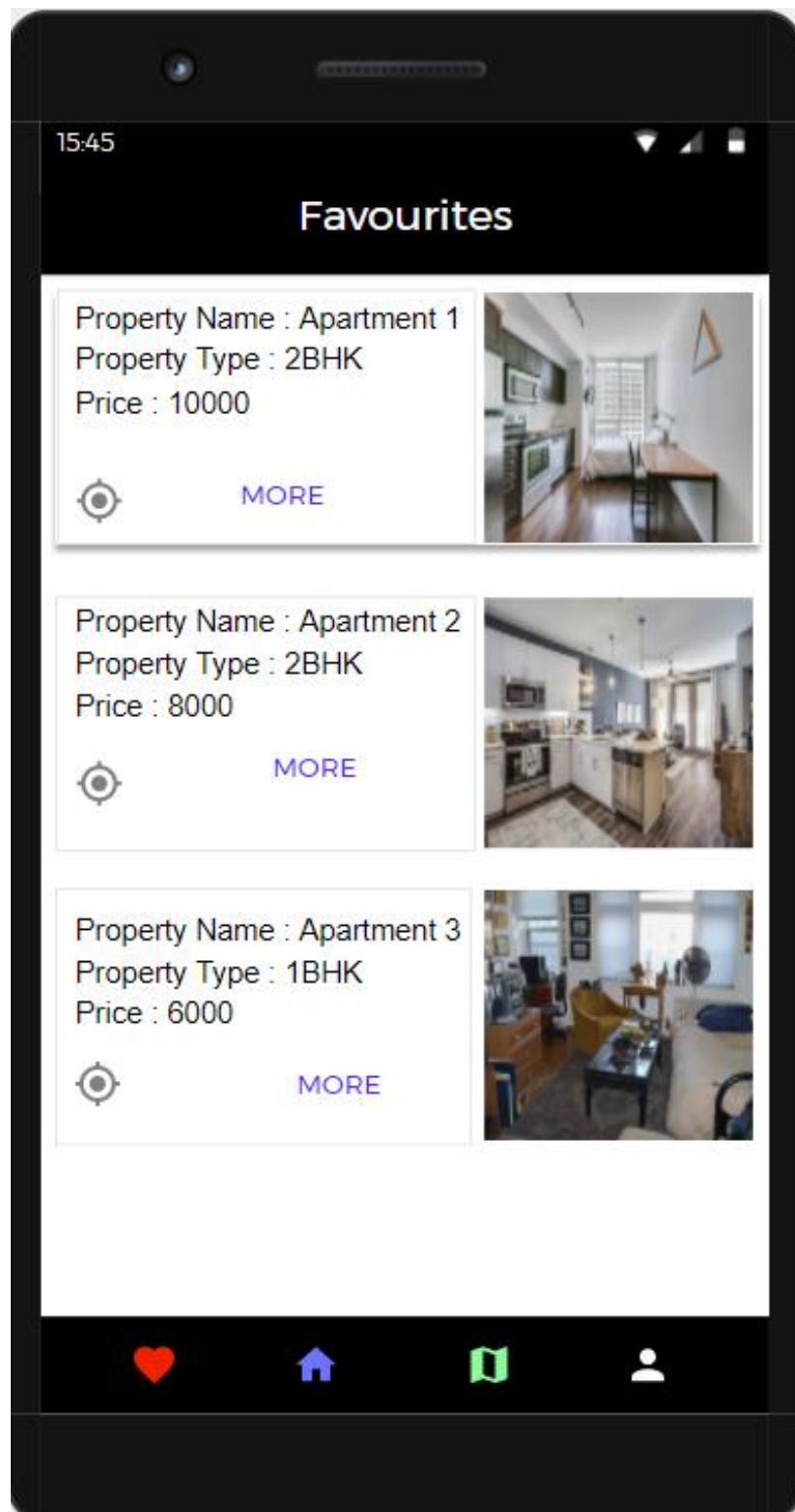
Get Link

6.5 Descision Screen

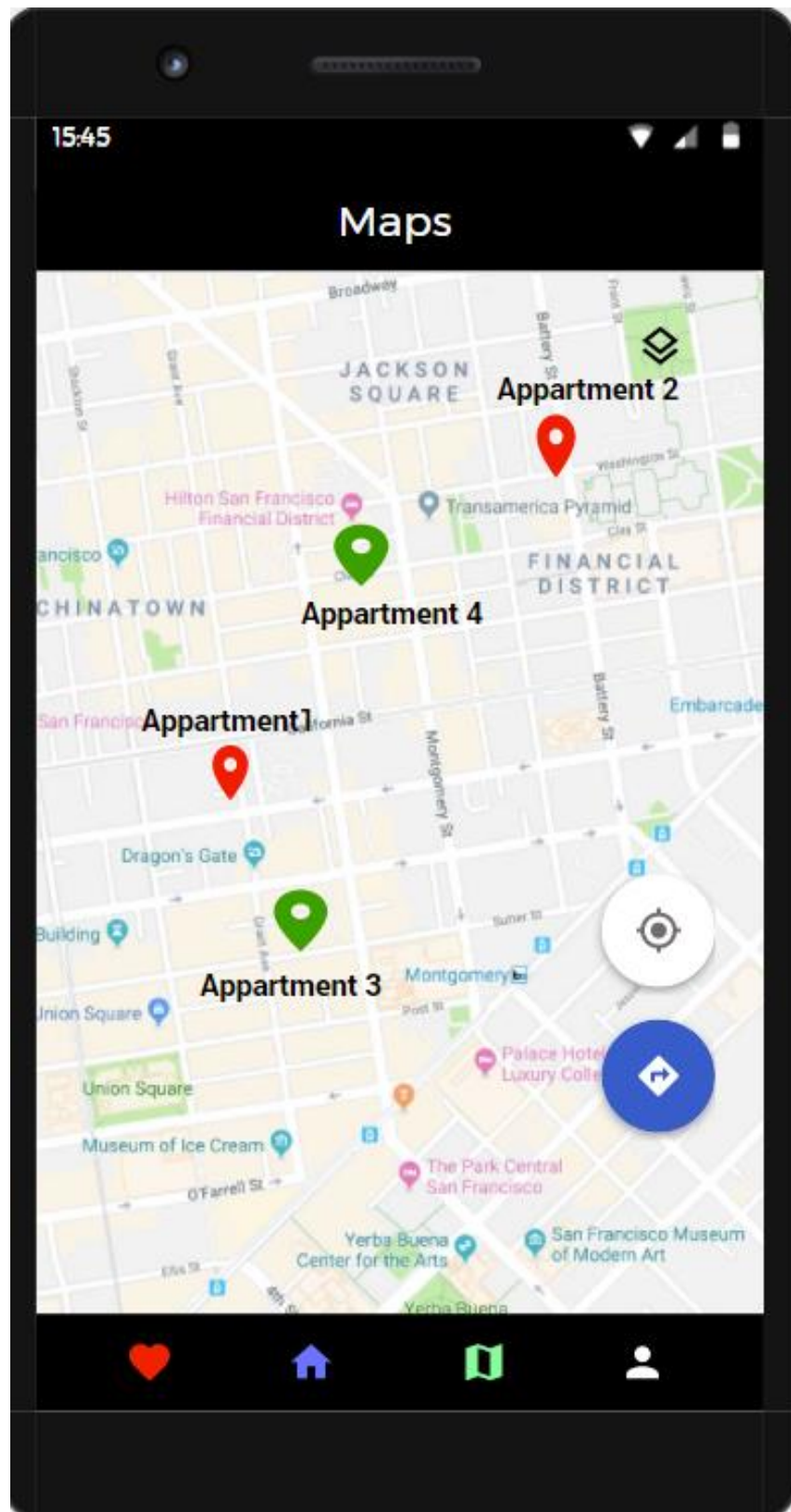


6.6 User Dashboard

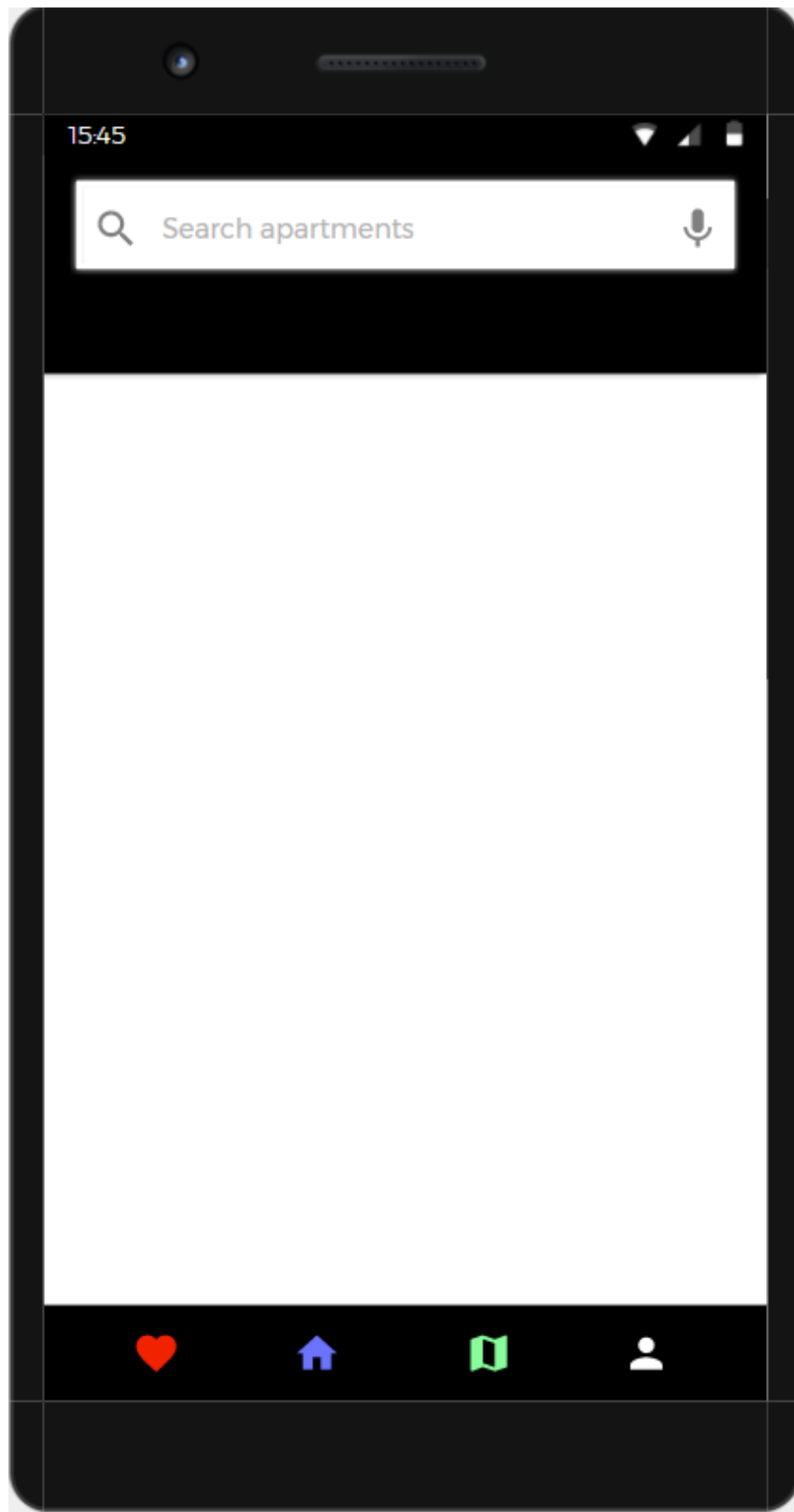
6.6.1 User Favourites



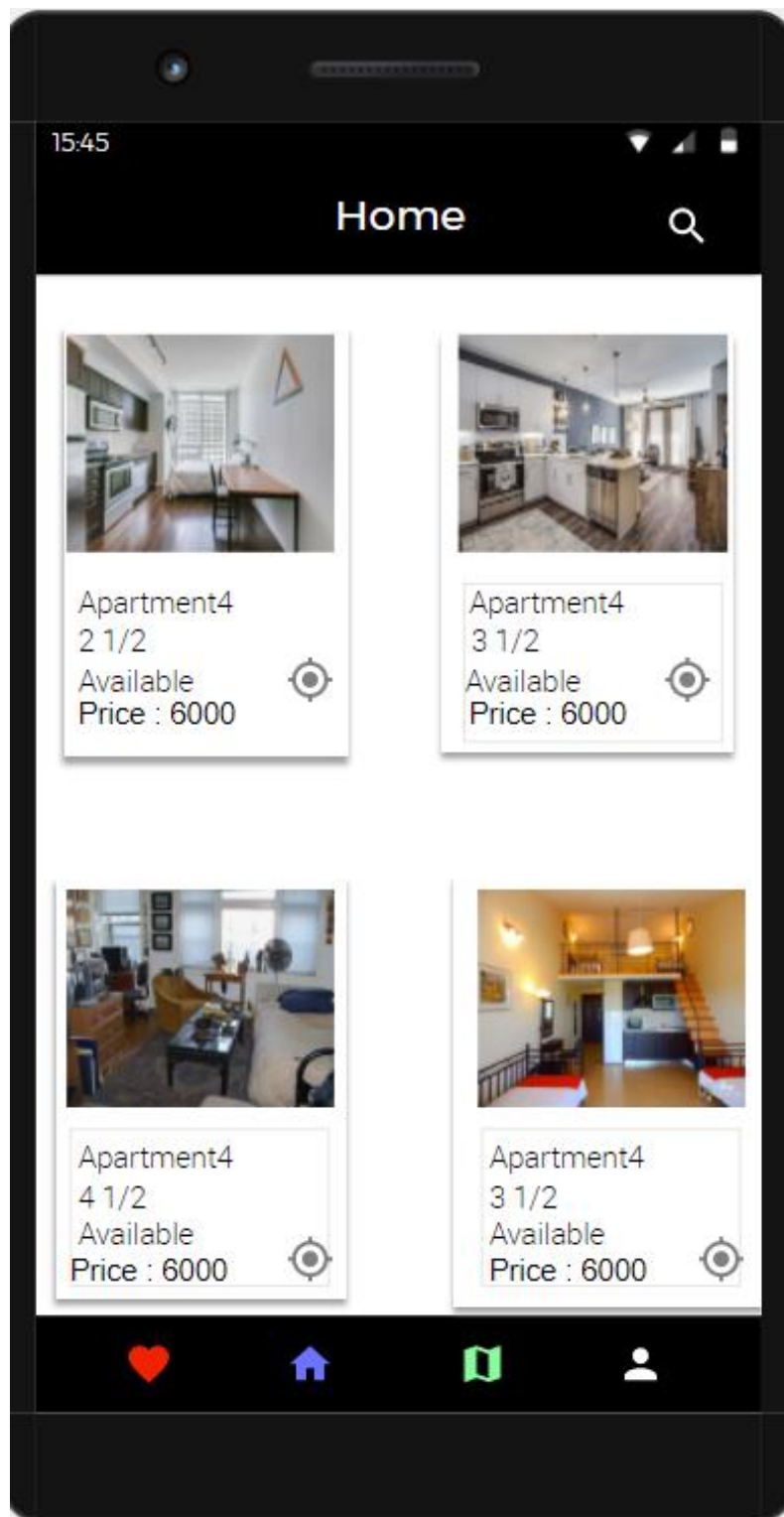
6.6.2 User Maps



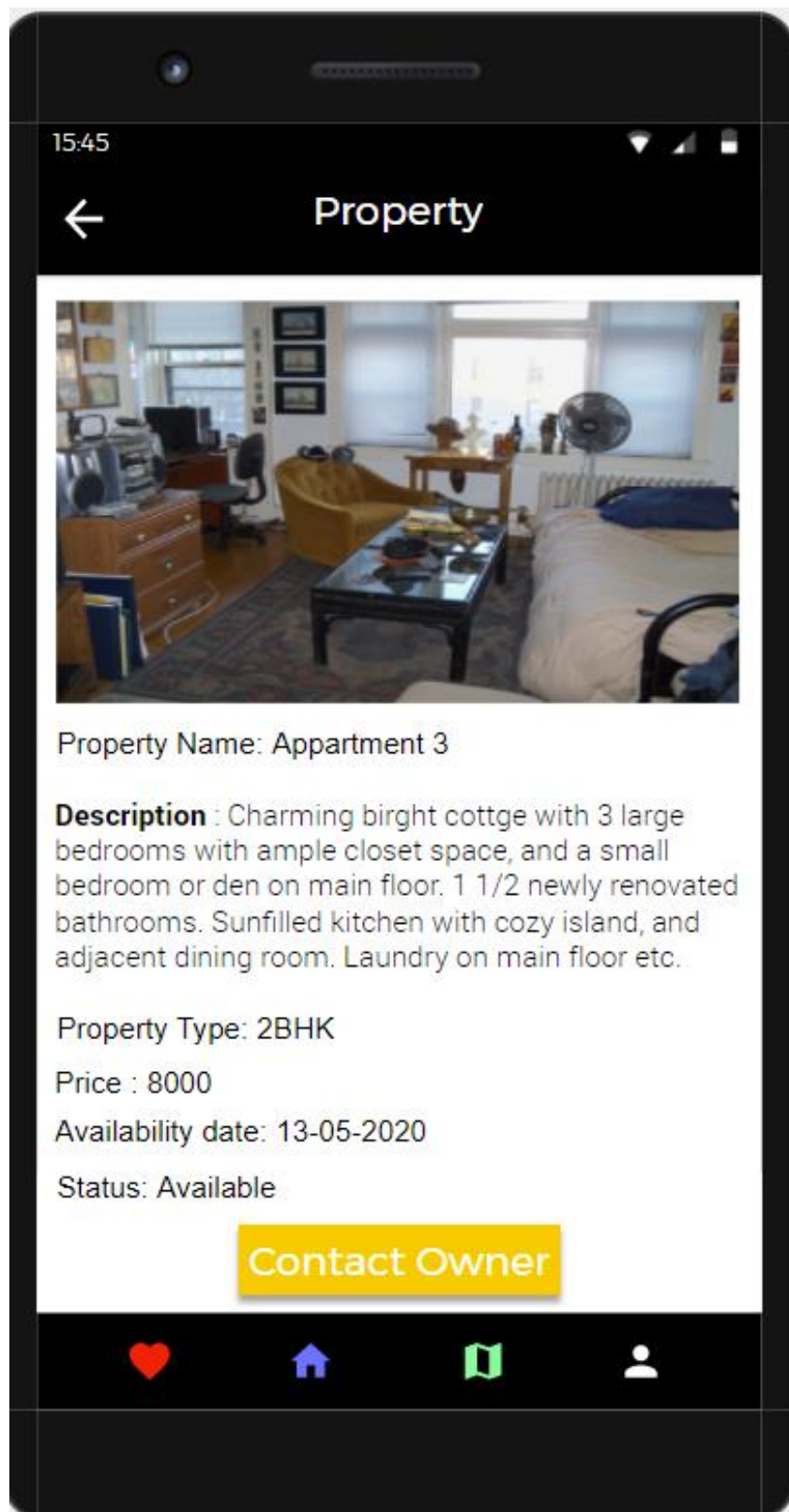
6.6.3 User Search



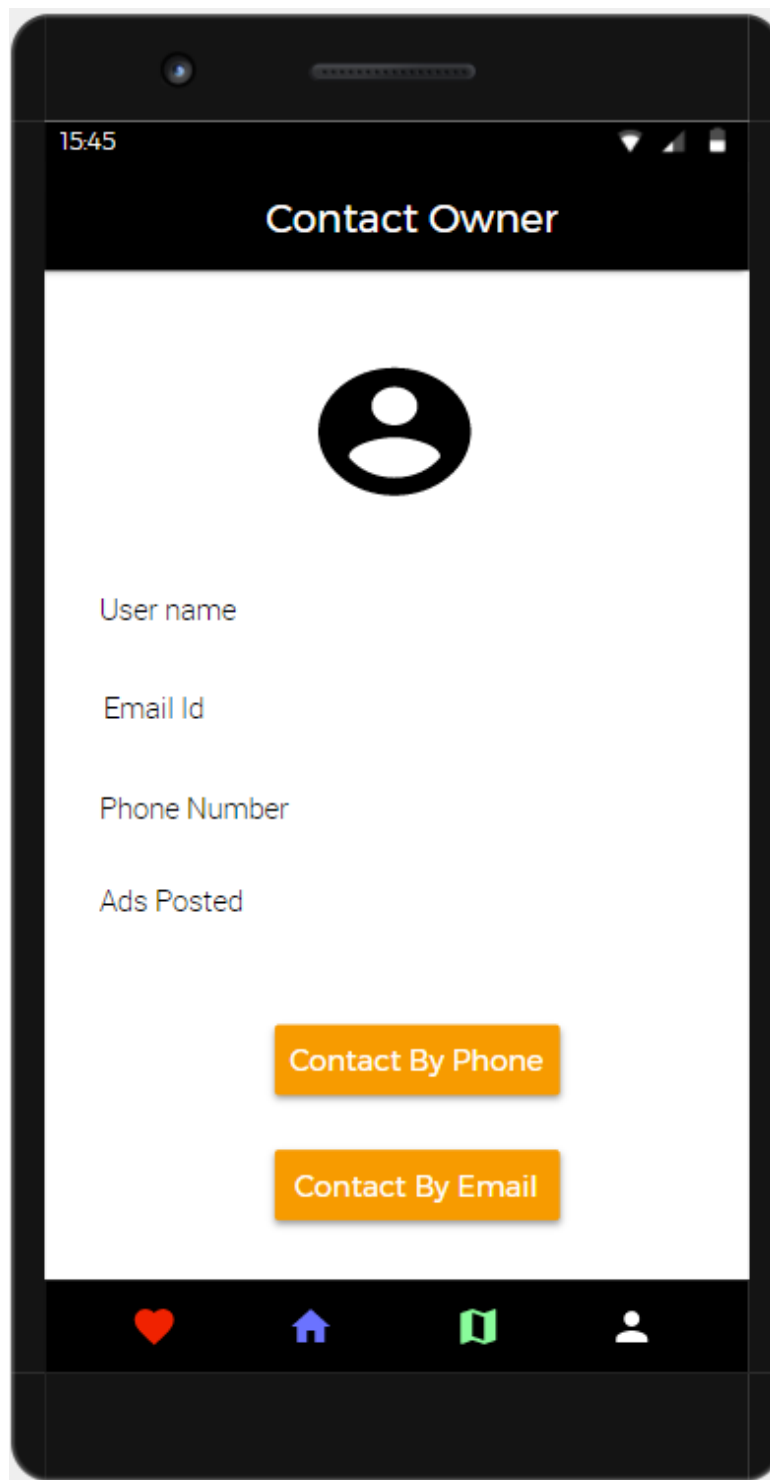
6.6.4 User Home



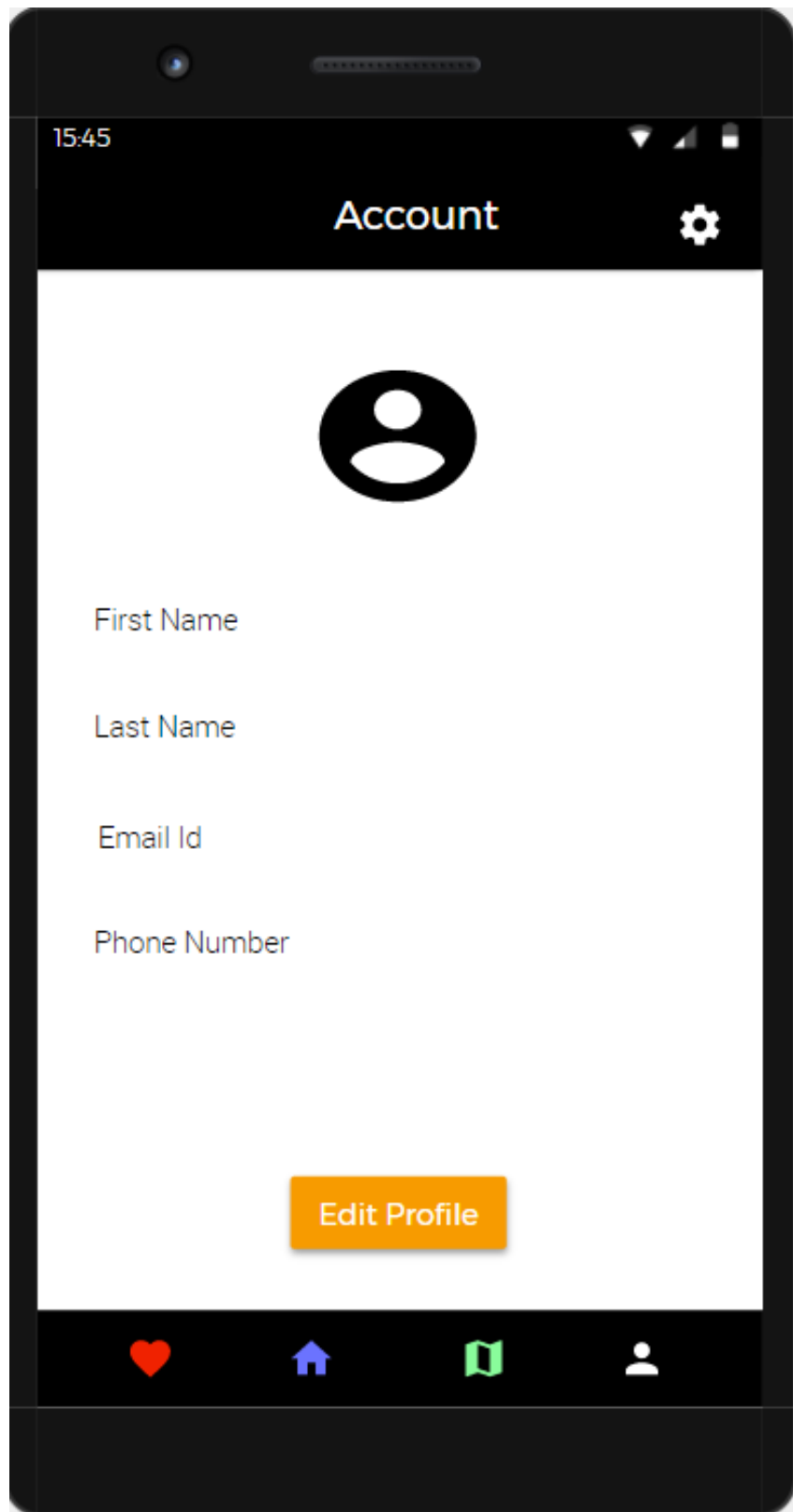
6.6.5 User Property



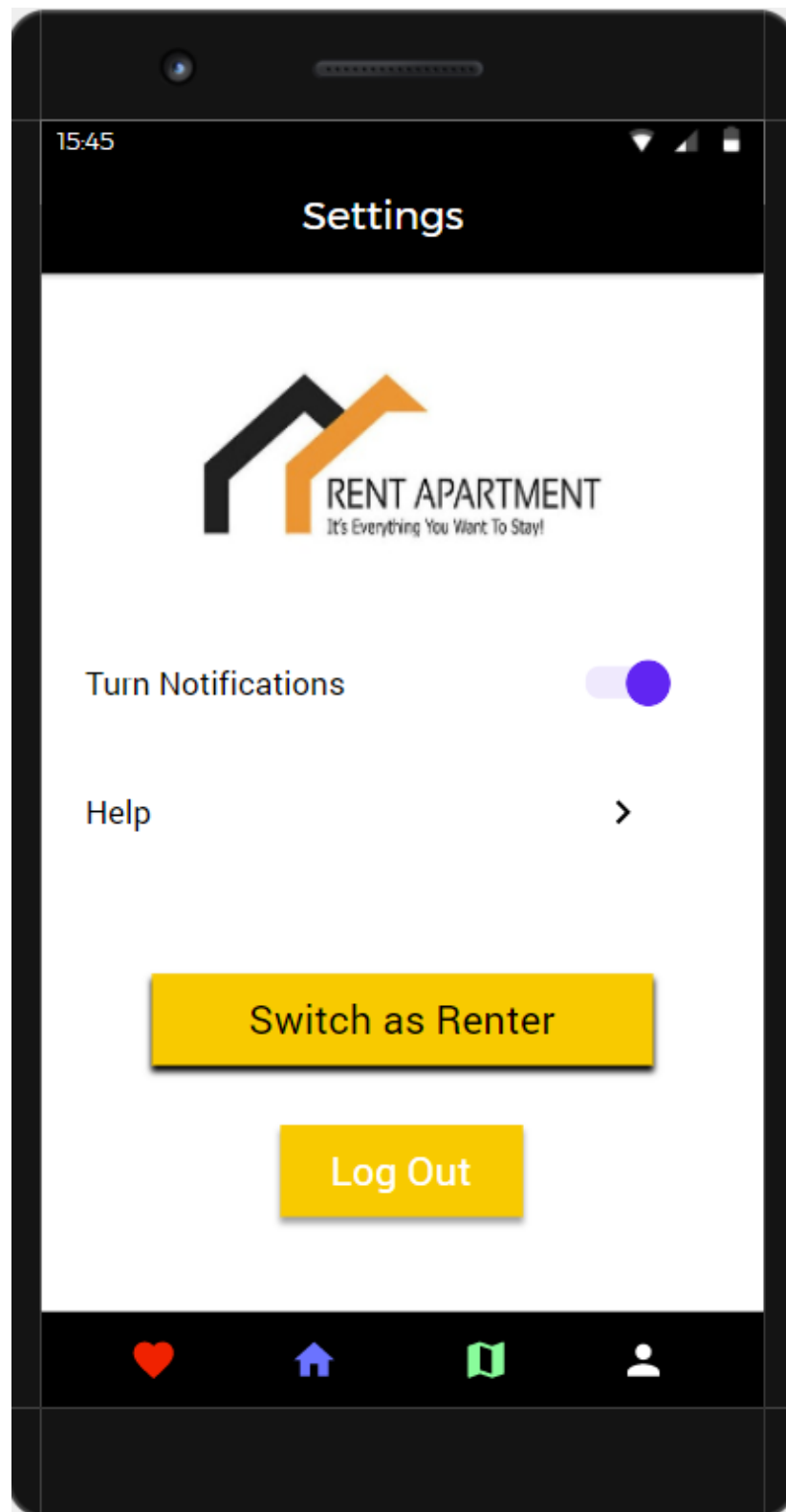
6.6.6 User Contact Owner



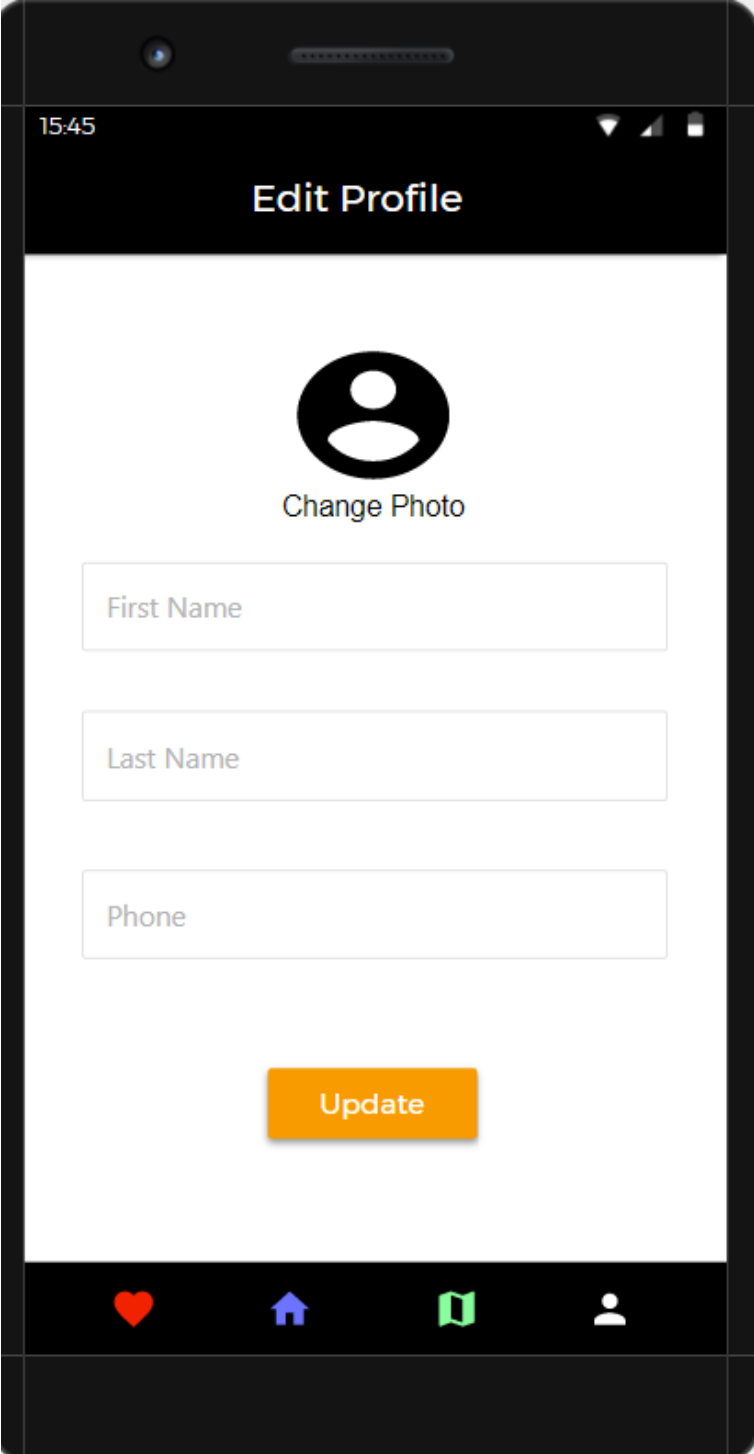
6.6.7 User Account



6.6.8 User Settings




6.6.9 Edit Profile



The image shows a mobile application interface for editing a user profile. The screen has a dark header bar at the top with the title "Edit Profile" in white. Below the header, the status bar shows the time "15:45" and various system icons. The main content area is white and contains a large black circular icon representing a person. Below this icon is the text "Change Photo". There are three white input fields with light gray borders, each containing a placeholder label: "First Name", "Last Name", and "Phone". Below these fields is a prominent orange button with the text "Update" in white. At the bottom of the screen is a dark navigation bar with four icons: a red heart, a blue house, a green book, and a white person icon.

15:45

Edit Profile



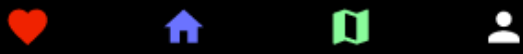
Change Photo

First Name

Last Name

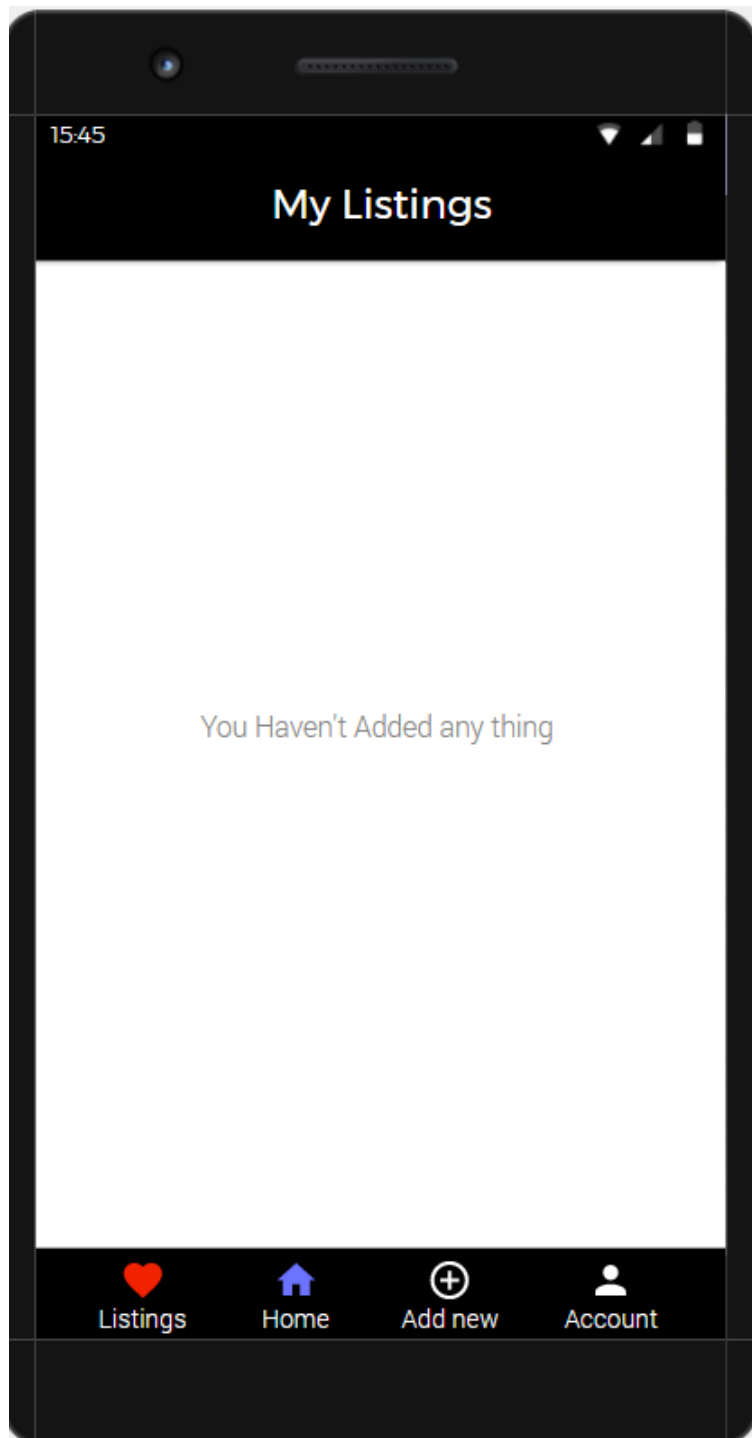
Phone

Update

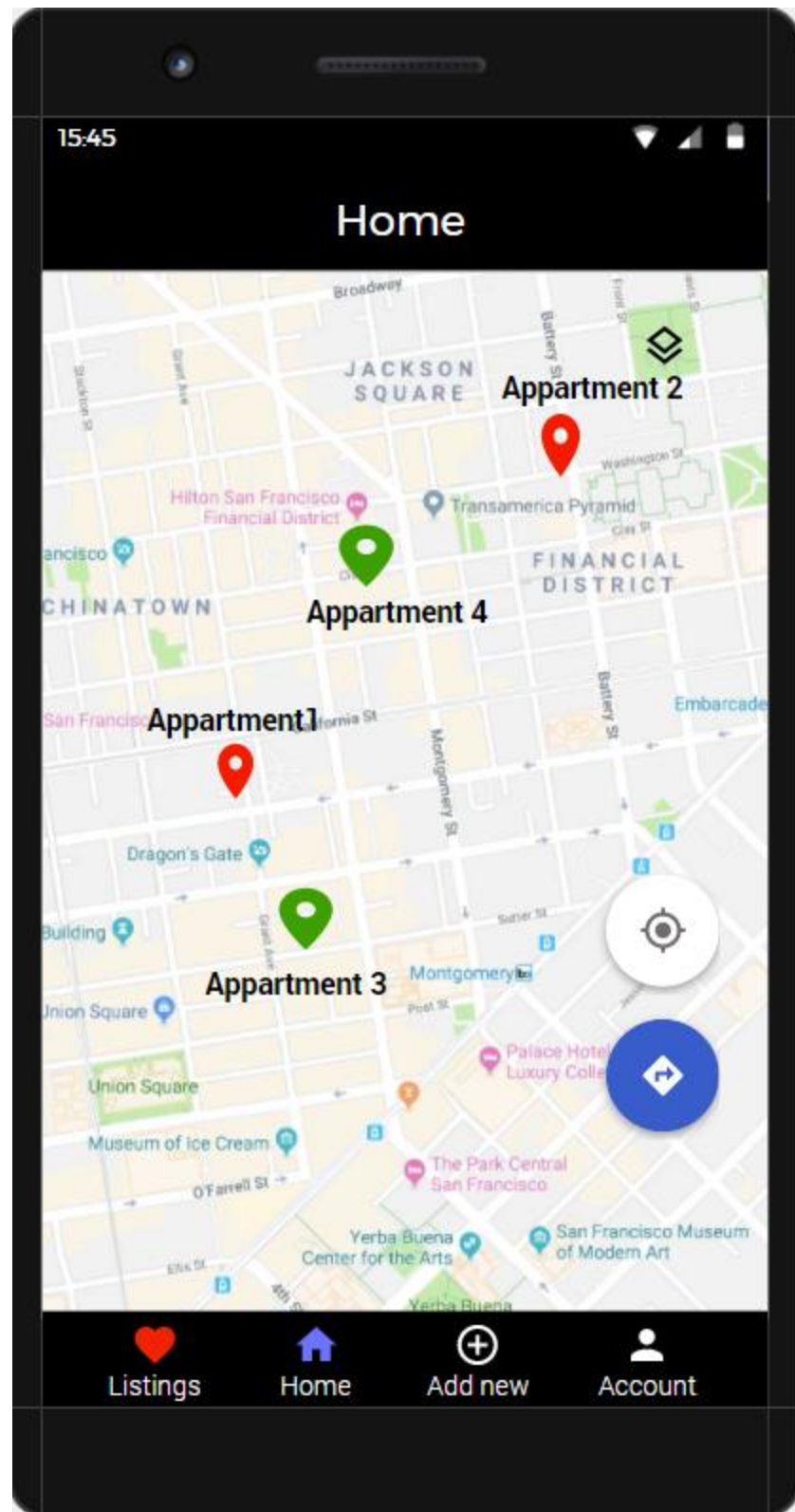


6.7 Renter Dashboard

6.7.1 Renter Listing



6.7.2 Renter Home



6.7.3 Renter Add New

The image shows a mobile application interface for adding a new property. The screen has a dark header with the title "Add Property" in white. Below the header, there are five white input fields with gray placeholder text: "Property Name", "Property Type", "Price", "Availability", and "Description". Below these fields are two dark gray buttons with white text: "+ Latitude/ Longitude" and "+ Image". At the bottom of the form area is a prominent orange button with the text "Add Property". The bottom of the screen features a dark navigation bar with four icons and labels: a red heart icon for "Listings", a blue house icon for "Home", a white plus icon for "Add new", and a white person icon for "Account". The status bar at the top shows the time "15:45" and various system icons.

15:45

Add Property

Property Name

Property Type

Price

Availability

Description

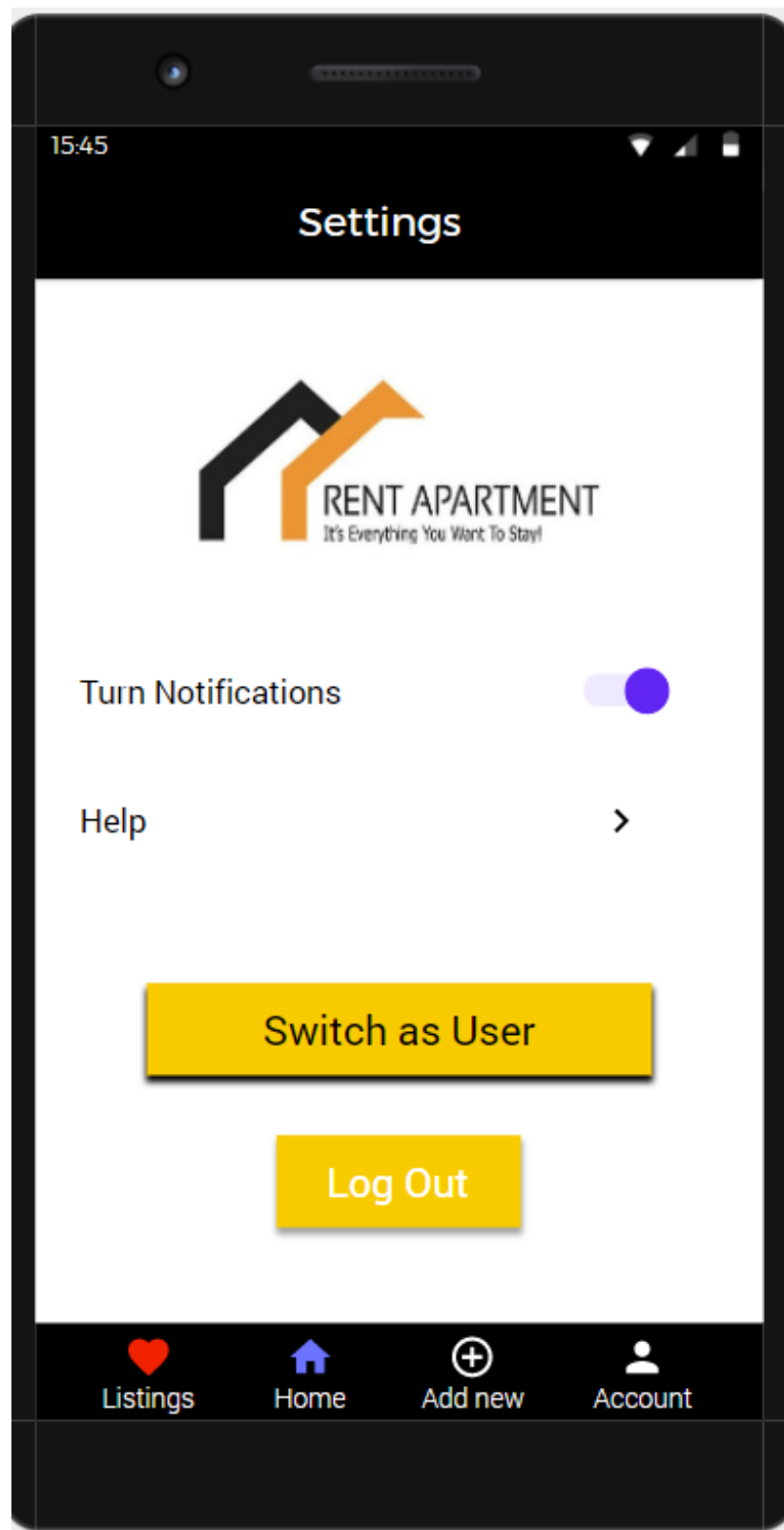
+ Latitude/ Longitude

+ Image

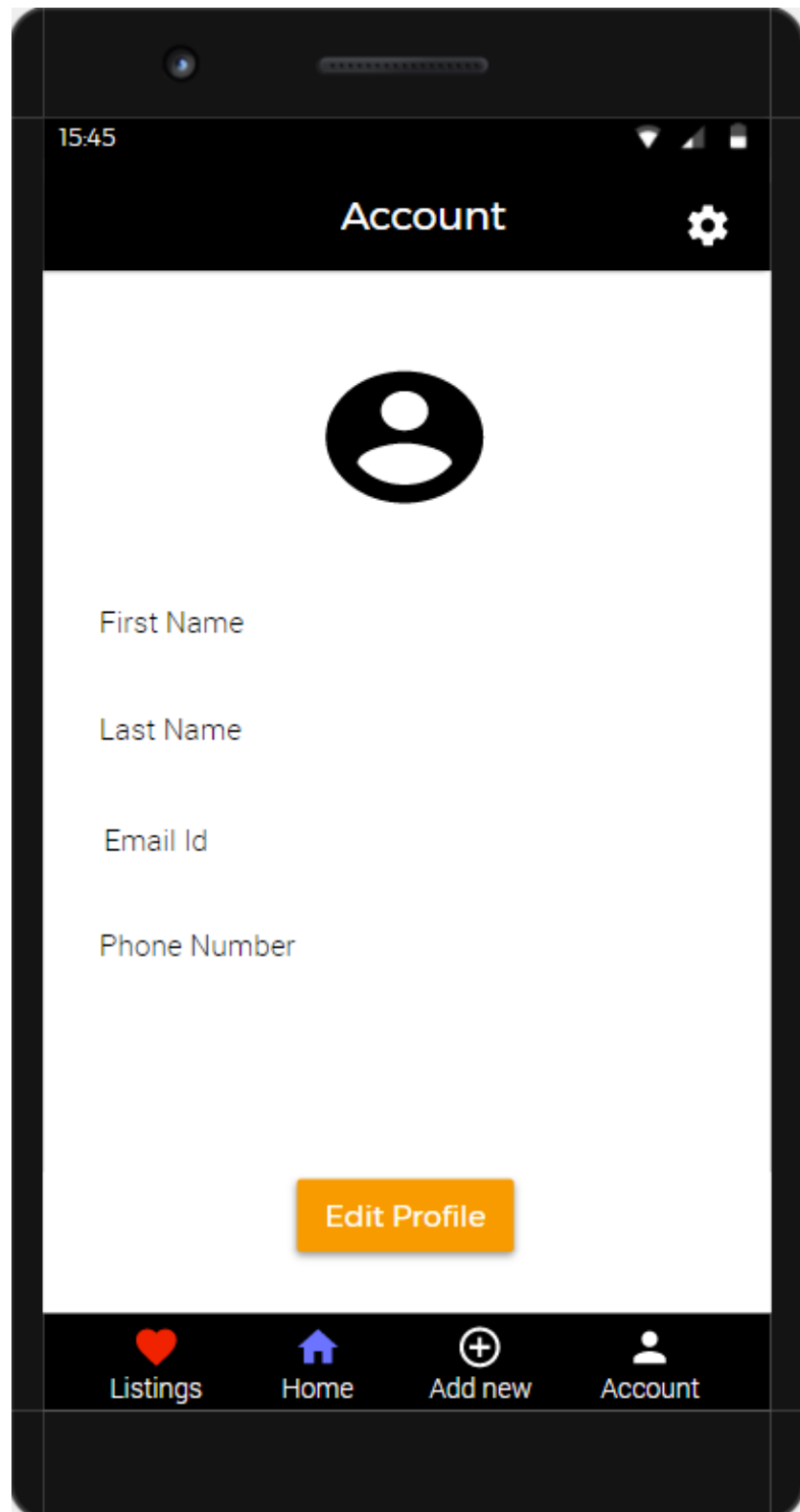
Add Property

Listings Home Add new Account

6.7.4 Renter Settings




6.7.5 Renter Account



6.7.6 Renter Edit Profile

15:45

Edit Profile






Change Photo

First Name

Last Name

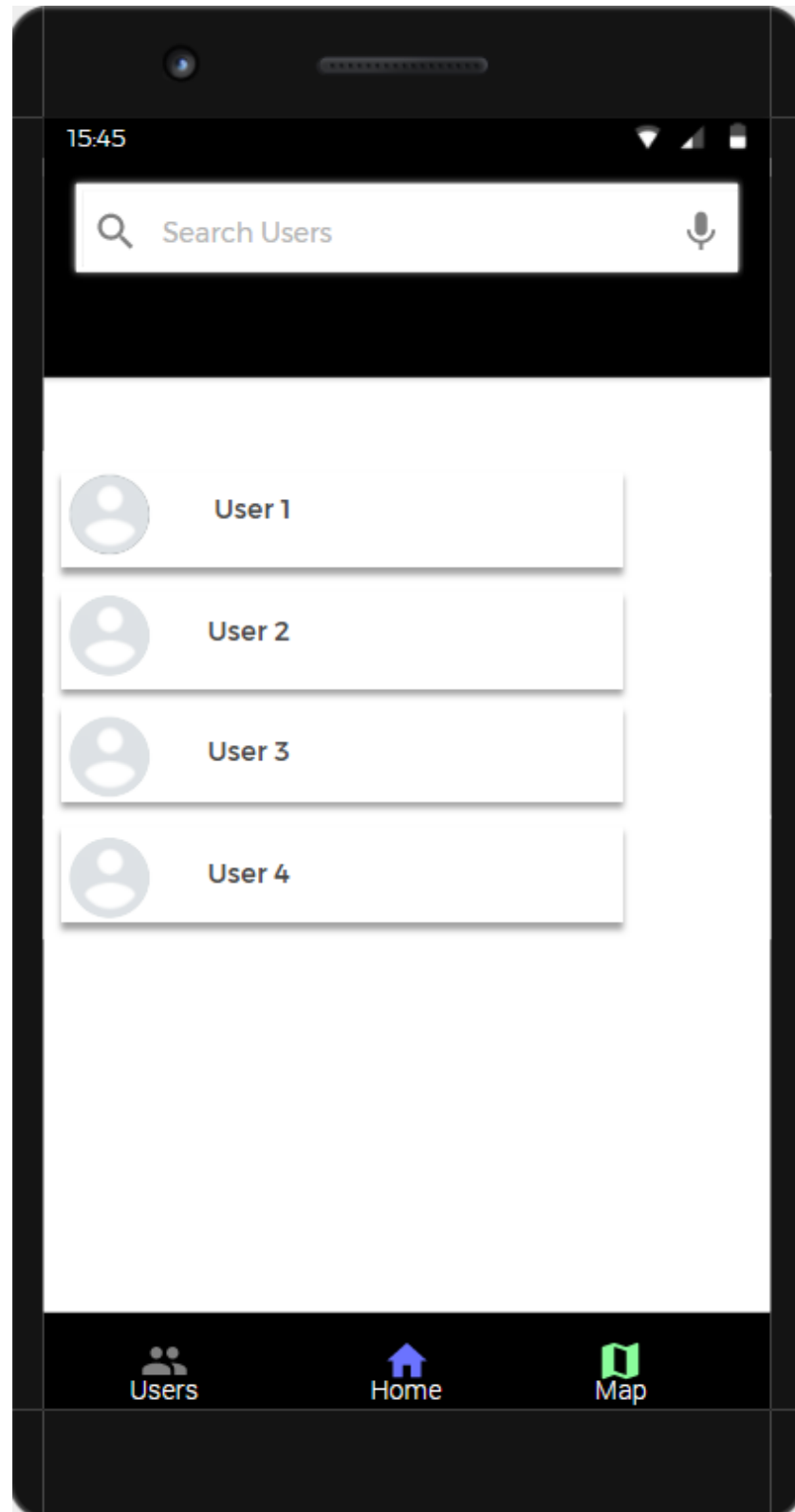
Phone

Update

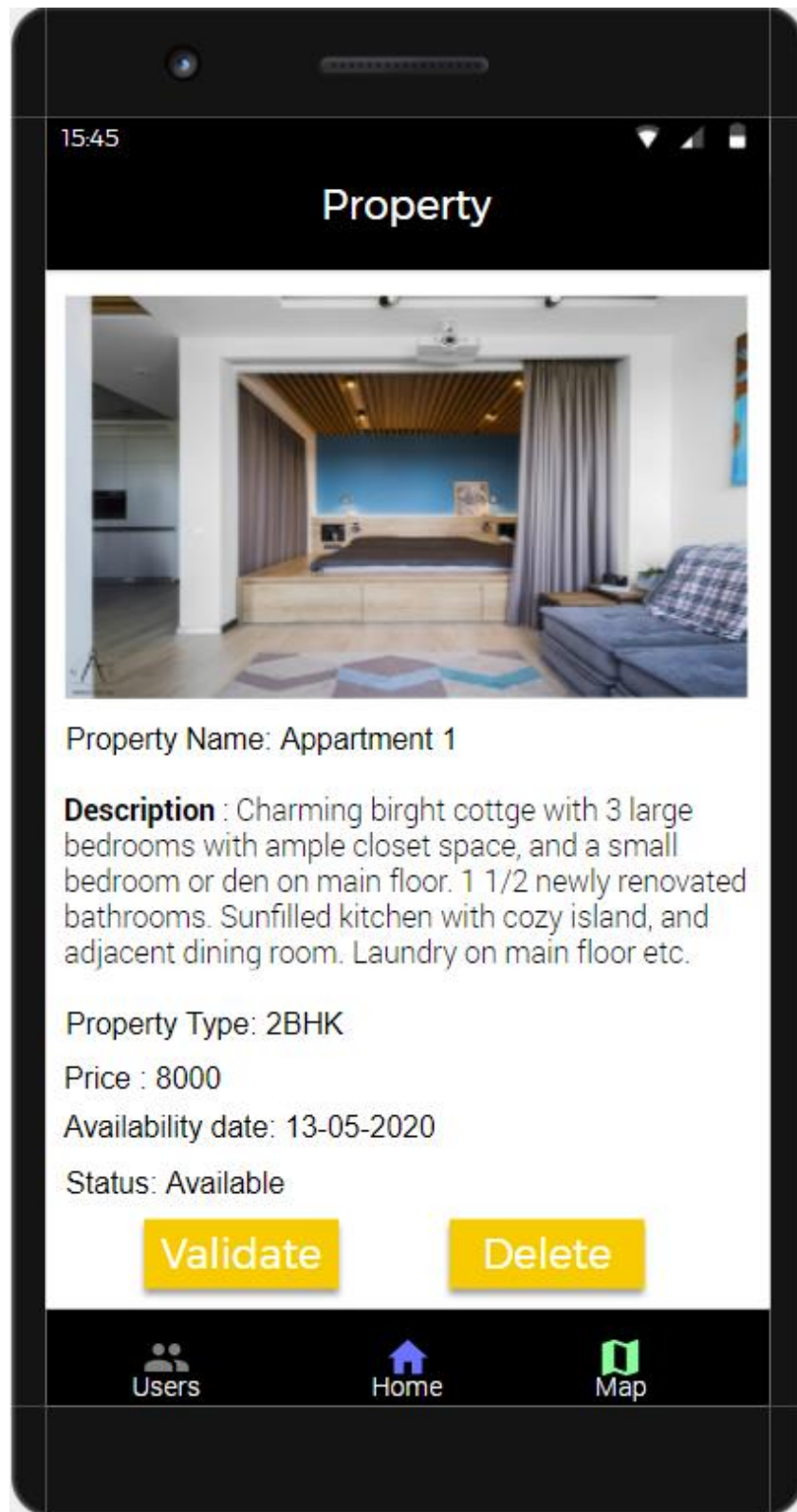
 Listings  Home  Add new  Account

6.8 Admin Dashboard

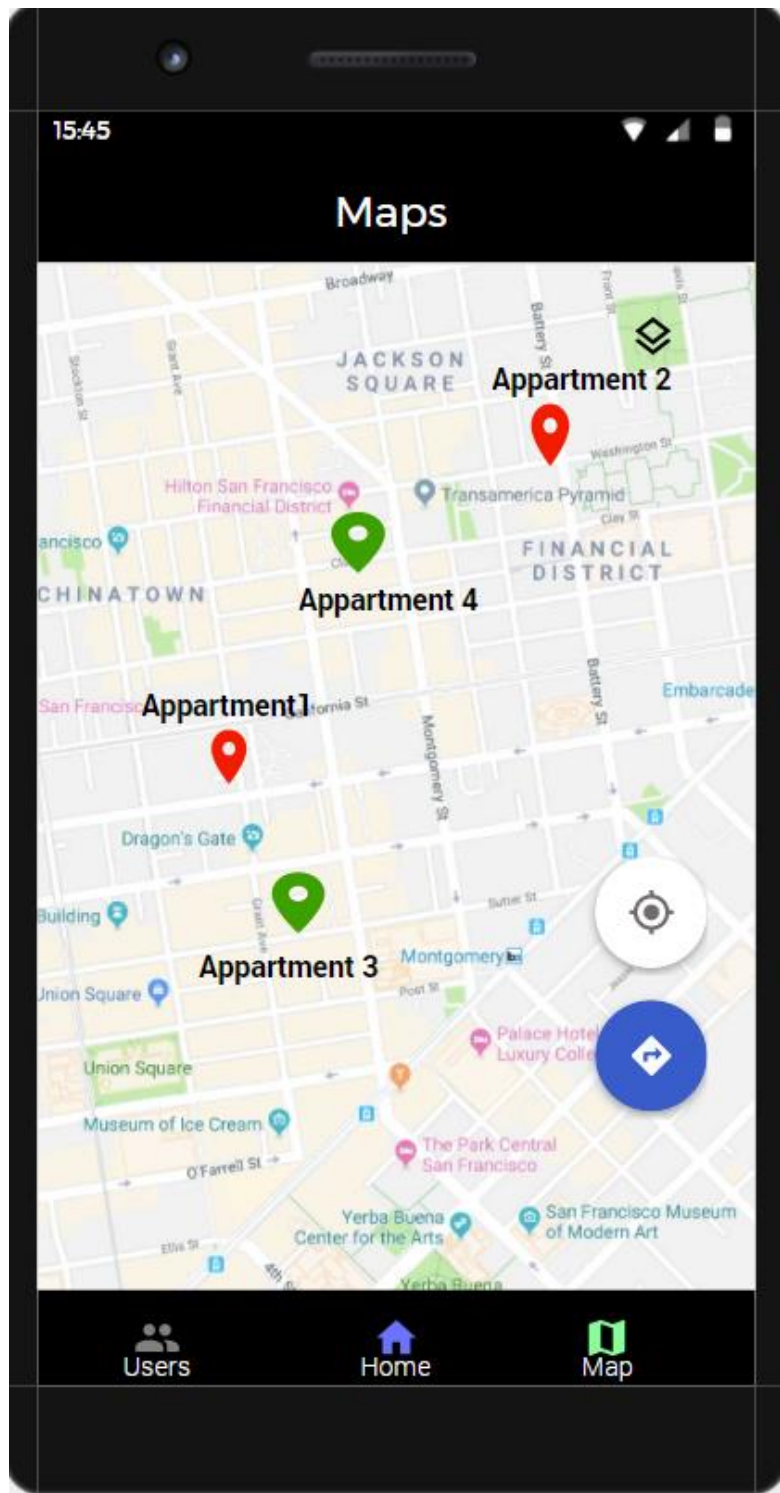
6.8.1 Admin Users



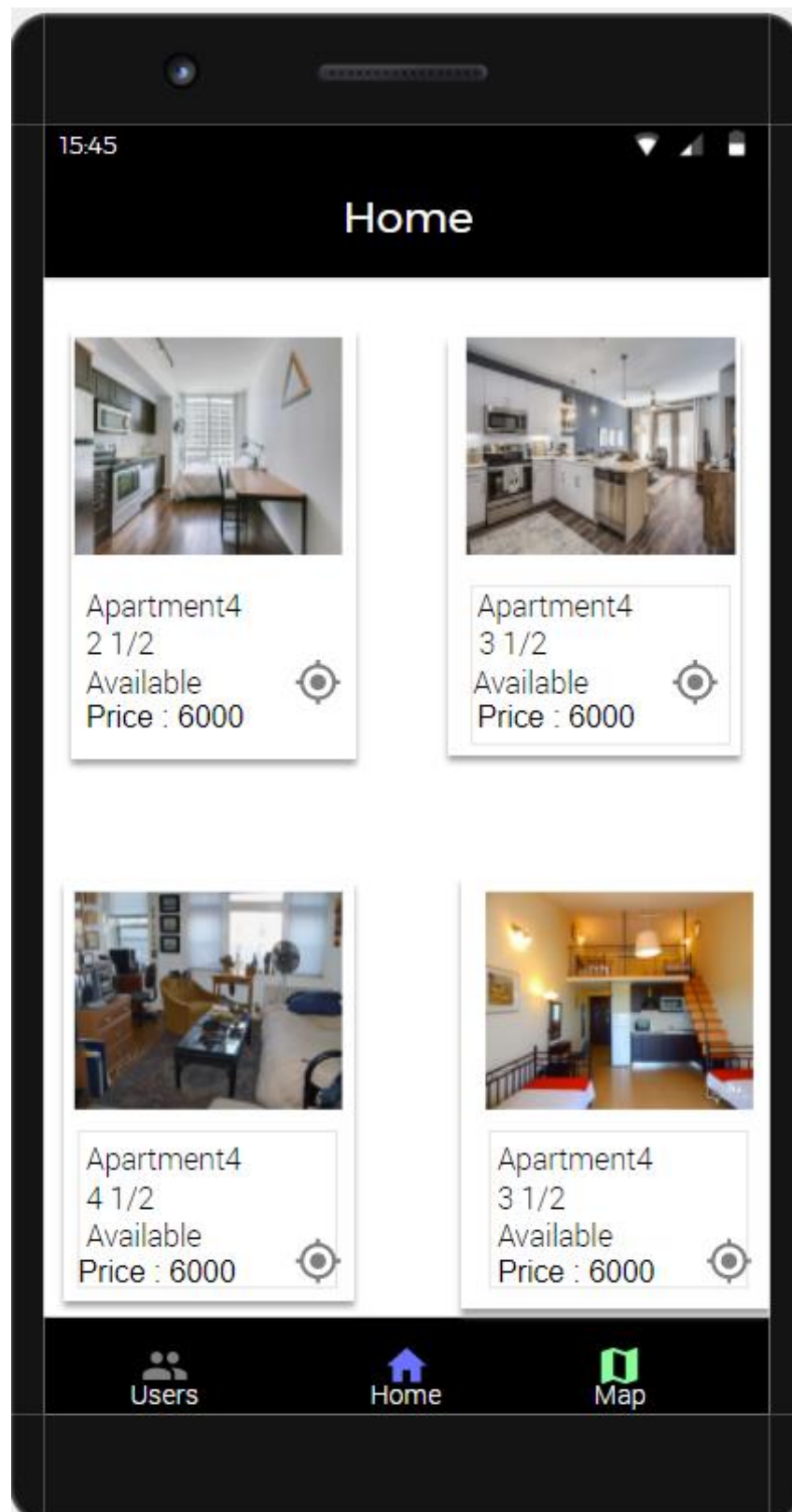
6.8.2 Admin Property



6.8.3 Admin Map



6.8.4 Admin Home



6.8.5 Admin Delete User

