****

**Project -Title: Homify**

**Submitted by**

Taukir Ahmad Raju

ID: 191-35-425

Department of Software Engineering

Daffodil International University

**Supervised by**

Afsana Begum

Assistant Professor & Coordinator M.Sc

Department of Software Engineering

Daffodil International University

This project report has been submitted in fulfillment of the requirements for the Degree of Bachelor of Science in Software Engineering.

© All right Reserved by Daffodil International University

Date of Submission: 6th July 2023

**ABSTRACT**

Homify is a real estate app that connects homeowners who are looking to rent out their properties with potential renters. The app has two user ends, one for the public and one for homeowners. The public can see rental posts from homeowners and contact them through various means such as email, call, or messaging by creating a user account. Homeowners can store information about their properties, including bills, renter information, and make monthly bill invoices for renters. The app stores this information in a database and sends instant SMS to renters' contact numbers. Homeowners can view graphical representations of their property information and assign a manager to manage their dashboard if they have multiple properties.

Here are some of the key features of Homify:

* Easy to use interface for both homeowners and renters
* Secure database for storing property information
* Instant SMS notifications for renters
* Graphical representations of property information
* Ability to assign a manager to manage your dashboard

Homify is a convenient and secure way to rent out your property. With its easy-to-use interface and secure database, Homify makes it easy to find tenants and manage your rental properties.

Table of Contents

[CHAPTER 1 1](#_Toc139387680)

[INTRODUCTION 1](#_Toc139387681)

[1.1 Purpose 1](#_Toc139387682)

[1.2 Scope 1](#_Toc139387683)

[1.3 Target Audience 2](#_Toc139387684)

[1.4 Project Schedule 2](#_Toc139387685)

[1.4.1 Release Plan And Milestone 2](#_Toc139387686)

[CHAPTER 2 3](#_Toc139387687)

[REQUIREMENT ANALYSIS 3](#_Toc139387688)

[2.1 Functional Requirements: 3](#_Toc139387689)

[2.1.1 General User 3](#_Toc139387690)

[2.1.2 Home Owner 4](#_Toc139387691)

[2.1.3 Admin 5](#_Toc139387692)

[2.2 Non-Functional Requirements: 6](#_Toc139387693)

[CHAPTER 3 7](#_Toc139387694)

[SYSTEM OVERVIEW 7](#_Toc139387695)

[3.1 System Architecture 7](#_Toc139387696)

[3.2 Development Model 8](#_Toc139387697)

[3.3 Use Case Diagram 9](#_Toc139387698)

[3.4 Activity Diagram 10](#_Toc139387699)

[3.5 Sequence Diagram 16](#_Toc139387700)

[3.6 Database Modeling 21](#_Toc139387701)

[3.7 Technology Stack 22](#_Toc139387702)

[CHAPTER 4 23](#_Toc139387703)

[USER INTERFACE 23](#_Toc139387704)

[4.1 Public User Interface 23](#_Toc139387705)

[4.1.1 Property Listings 23](#_Toc139387706)

[4.1.2 Search and Filters 24](#_Toc139387707)

[4.1.3 Contact Homeowners 24](#_Toc139387708)

[4.1.4 User Account 25](#_Toc139387709)

[4.1.5 Live Chat 25](#_Toc139387710)

[4.2 Homeowner Interface 26](#_Toc139387711)

[4.2.1 Homeowner Dashboard 26](#_Toc139387712)

[4.2.2 Property Management 26](#_Toc139387713)

[4.2.3 Renter Management 26](#_Toc139387714)

[4.2.4 Billing and Invoicing 27](#_Toc139387715)

[4.2.5 Homeowner Profile 28](#_Toc139387716)

[4.2.6 Home Information 28](#_Toc139387717)

[4.2.7 Post Section 29](#_Toc139387718)

[4.3 Admin Interface 29](#_Toc139387719)

[4.3.1 Admin Dashboard 29](#_Toc139387720)

[4.3.2 User Management 30](#_Toc139387721)

[4.3.3 House Management 30](#_Toc139387722)

[4.3.4 Post Management 31](#_Toc139387723)

[CHAPTER 5 32](#_Toc139387724)

[SYSTEM TESTING 32](#_Toc139387725)

[5.1 Testing Phases 32](#_Toc139387726)

[5.1 Testing Schedule 33](#_Toc139387727)

[CHAPTER 6 34](#_Toc139387728)

[Conclusion 34](#_Toc139387729)

[Frequently Asked Questions (FAQ) 35](#_Toc139387730)

[References 36](#_Toc139387731)

List of Figures

USE CASE DIAGRAM

[Figure 3.3 1 Use Case Diagram 9](#_Toc139386784)

ACTIVITY DIAGRAM

[Figure 3.4 1 User login 10](#_Toc139386771)

[Figure 3.4 2 User Registration 11](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386772)

[Figure 3.4 3 Search Post 11](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386773)

[Figure 3.4 4 Check Bill 11](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386774)

[Figure 3.4 5 Create Home 12](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386775)

[Figure 3.4 6 Live chat 12](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386776)

[Figure 3.4 7 Create Renter 12](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386777)

[Figure 3.4 8 Create Apartment 12](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386778)

[Figure 3.4 9 Assign Renter 13](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386779)

[Figure 3.4 10 Post Apartment 13](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386780)

[Figure 3.4 11 Make Bill 14](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386781)

[Figure 3.4 12 Create Admin 15](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386782)

[Figure 3.4 13 Verify House 15](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139386783)

SEQUENCE DIAGRAM

[Figure 3.5 1 Login 16](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387414)

[Figure 3.5 2 Registration 17](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387415)

[Figure 3.5 3 Search Post 17](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387416)

[Figure 3.5 4 Post Apartment 18](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387417)

[Figure 3.5 5 Create Apartment 18](#_Toc139387418)

[Figure 3.5 6 Assign Renter 19](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387419)

[Figure 3.5 7 Create Bill 19](file:///C:\Users\Toukir\Desktop\HomeFy%20Updated\Project_documentation.docx#_Toc139387420)

[Figure 3.5 8 Verify House 20](#_Toc139387421)

[Figure 3.5 9 Create Admin 20](#_Toc139387422)

## CHAPTER 1

## INTRODUCTION

Welcome to Homify, a powerful real estate app that connects people who want to rent out their homes with people who might be interested in renting them. Homify wants to make it easier to find and rent homes by giving people a platform that is easy to use, helps people talk to each other, and makes property management easier.

### 1.1 Purpose

The Homify app was designed to simplify the process of renting a home by making it easier for landlords to advertise their properties and for tenants to find and interact with them. The software serves as a hub for all of a landlord's administrative needs, allowing them to keep track of their properties, their bills, their tenants, and their monthly billing. At the same time, tenants can look for available rentals and get in touch with landlords via a selection of channels (email, phone, instant message).

### 1.2 Scope

Homeowners and renters alike can benefit from Homify's attention to the rental sector. The software is useful for homeowners because it allows them to advertise their homes, keep track of relevant rental data, and connect with prospective tenants. However, tenants can use the app to look for available rentals and contact property owners. The app also has an admin section for managing and monitoring the system as a whole, including users, properties, reports, and analytics.

### 1.3 Target Audience

The Homify app targets two primary user groups:

* Homeowners: People that have property and are looking to rent it out. Homify allows them to advertise their rentals, handle payments and other administrative tasks, and connect with prospective tenants.
* Renters: People who want to rent an apartment or house. They can use Homify to look at rental listings, get in touch with homeowners, and start the process of renting.

In addition, the app caters to system administrators who have the authority to manage and monitor the entire system, ensuring its smooth operation.

Homify aims to provide a user-friendly and efficient platform that benefits both homeowners and renters, simplifying the rental process and fostering effective communication between parties.

### 1.4 Project Schedule

Due to the short deadline, I created a schedule to ensure that the project was completed on time. The schedule outlined the tasks that needed to be completed and the actions that needed to be taken. I used Trello to track my progress.

### 1.4.1 Release Plan And Milestone

**Table: Release Plan**

|  |  |  |
| --- | --- | --- |
| **Activities** | **Duration in Week** | **Total Week** |
| Research | W1, W2 | 2 |
| Planning | W2, W3, W4, W5 | 4 |
| Design | W5, W6, W7, W8 | 4 |
| Development | W9, W10, W11, W12 | 4 |
| Testing | W13, W14 | 2 |
| Documentation | W14, W15, W16 | 3 |
| Software release | W17 | 1 |

## CHAPTER 2

## REQUIREMENT ANALYSIS

### 2.1 Functional Requirements:

Functional requirements are features or functions of a product that developers must build in order to help users do their jobs. So, it's important to make them clear to both the development team and the people who have an interest in the project. Most of the time, functional requirements describe how a system works in certain situations.

### 2.1.1 General User

|  |  |
| --- | --- |
| **Requirement ID** | **FR.G.1** |
| **Requirement Name** | Searching available home |
| **Description** | General users can find their preferred home as per their needs. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.G.2** |
| **Requirement Name** | Email/Call home owner |
| **Description** | General users can directly emailed/call to home owner for more information |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.G.3** |
| **Requirement Name** | Live chat |
| **Description** | General users can talk directly with home Owner via live chat |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.G.4** |
| **Requirement Name** | Login/Register |
| **Description** | General users must login/register through phone no, email and password. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.G.5** |
| **Requirement Name** | Bill Query |
| **Description** | General users can get brief information about the bill which they paid. |

### 2.1.2 Home Owner

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.1** |
| **Requirement Name** | Login/Register |
| **Description** | Home Owner must login/register using phone number, email and password |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.2** |
| **Requirement Name** | Update home information |
| **Description** | Home Owner can easily update home related information  (Like: location, numbers of apartment, apartment rent etc.) |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.3** |
| **Requirement Name** | Create/Update user information |
| **Description** | If renter does not have any user account, then home Owner can open one account for him/her and update his/her info. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.4** |
| **Requirement Name** | Assign renter |
| **Description** | Home Owner can assign the new renter to his designated apartment. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.5** |
| **Requirement Name** | Create/Update apartments information |
| **Description** | Home Owner can create new apartment and update them  (Like: rent, water bill, gas bill etc.) |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.6** |
| **Requirement Name** | Generate monthly bills |
| **Description** | Home Owner can update and enter the monthly bills of every renter who lives in his house. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.7** |
| **Requirement Name** | Generate temporary bills |
| **Description** | Home Owner can update temporary bills.  (Like: electricity bill, WIFI bill , others service charge etc.) |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.H.8** |
| **Requirement Name** | Live chat |
| **Description** | Home Owner can talk directly with renter, support center via live chat. |

### 2.1.3 Admin

|  |  |
| --- | --- |
| **Requirement ID** | **FR.A.1** |
| **Requirement Name** | System Management |
| **Description** | Admins should have the ability to manage and oversee the entire system, including user management, property listings, and system configuration. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.A.2** |
| **Requirement Name** | User and Property Management |
| **Description** | Admins should be able to manage user accounts, including creating, updating, and deleting accounts as necessary. They should also have the ability to manage property listings and perform administrative tasks related to properties. |

|  |  |
| --- | --- |
| **Requirement ID** | **FR.A.3** |
| **Requirement Name** | Reporting and Analytics |
| **Description** | The app should provide reporting and analytics capabilities to admins, allowing them to generate reports, monitor system activities, and gain insights into the performance of the platform. |

### 2.2 Non-Functional Requirements:

Nonfunctional Requirements (NFRs) are meant to describe "system qualities," which are different parts of a system that don't have anything to do with how it works.

* **Usability**: The app should provide an intuitive and user-friendly interface for General users to easily navigate and search for rental properties.
* **Performance**: The browsing and search functionalities should be fast and responsive, providing a smooth user experience for general users.
* **Compatibility**: The app should be compatible with various devices and browsers, ensuring that general users can access it from different platforms.
* **Data Accuracy**: The app should ensure that the rental posts and property information displayed to general users are up-to-date and accurate.
* **Security**: General users' personal information, such as email addresses and contact details, should be protected and kept confidential.

## CHAPTER 3

## SYSTEM OVERVIEW

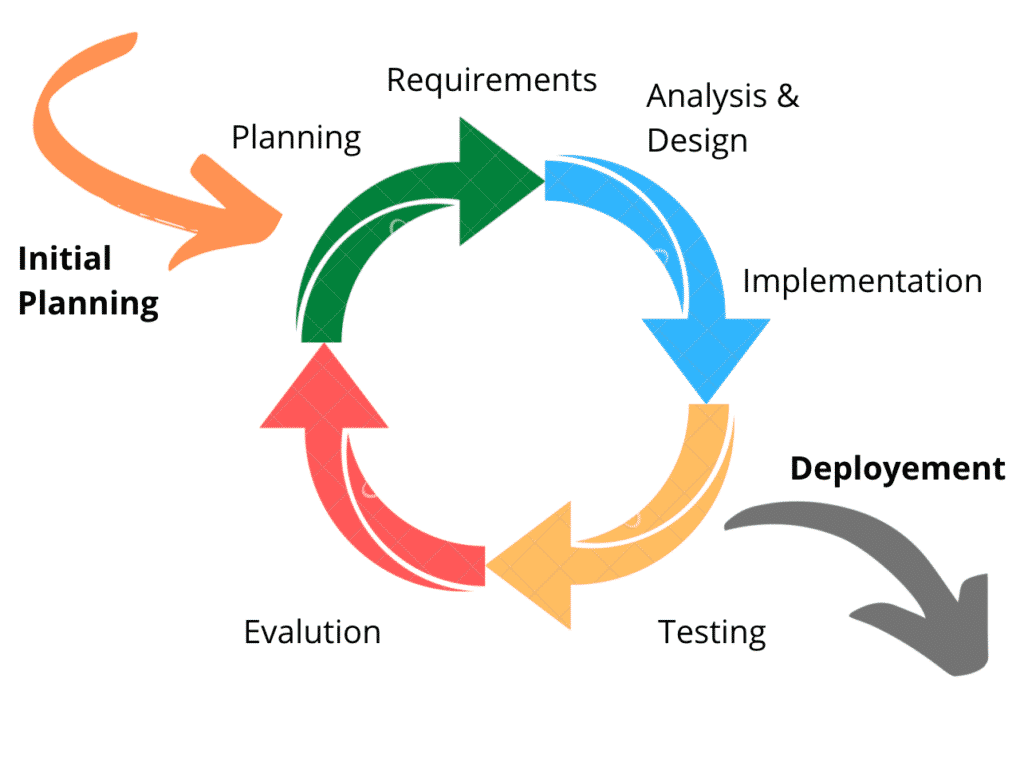
### 3.1 System Architecture

The Homify app is built on a scalable and modular architecture that ensures efficient performance and easy maintenance. The system architecture consists of the following key components:

1. **Front-End**: Homify's front end is built with HTML, CSS, JavaScript, React, Redux, Axios, and RTK Query. These technologies are used to make a dynamic and responsive user interface that lets general users, homeowners, and administrators interact with the app and get to their own features.
2. **Back-End**: Homify's back end is constructed with Node.js and Express, which offer a reliable and fast server-side environment. JSON Web Tokens (JWT) and Passport.js are two of the technologies used for authentication and authorization. The back end processes incoming requests, executes business logic, interacts with the database, and makes itself available to the front end via a RESTful API.
3. **Database**: Homify use MongoDB as the database management system for its platform. MongoDB is a NoSQL database that provides flexibility, scalability, and quick data retrieval among its many benefits. It holds a wide variety of data elements, such as user profiles, rental agreements, bills, and invoices, among other things. In addition, Cloudinary is employed as an image storage solution, which enables streamlined maintenance and retrieval of property photographs.
4. **External Services**: Homify is able to integrate third-party services such as SMS gateways, which allow for quick notifications to be sent to renters, and mapping systems, which provide location-based functionality. The functioning of the app is improved by these services, which also make the app's user experience more seamless.

### 3.2 Development Model

The Homify app has been built using the Agile development framework. Functionality is delivered in iterative and incremental cycles using agile approaches like Scrum and Kanban. This method promotes adaptability, regular feedback, and user participation in the development process. Agile methods like user stories, sprints, and retrospectives guarantee that the software can adapt to its users' and stakeholders' changing requirements.



### 3.3 Use Case Diagram

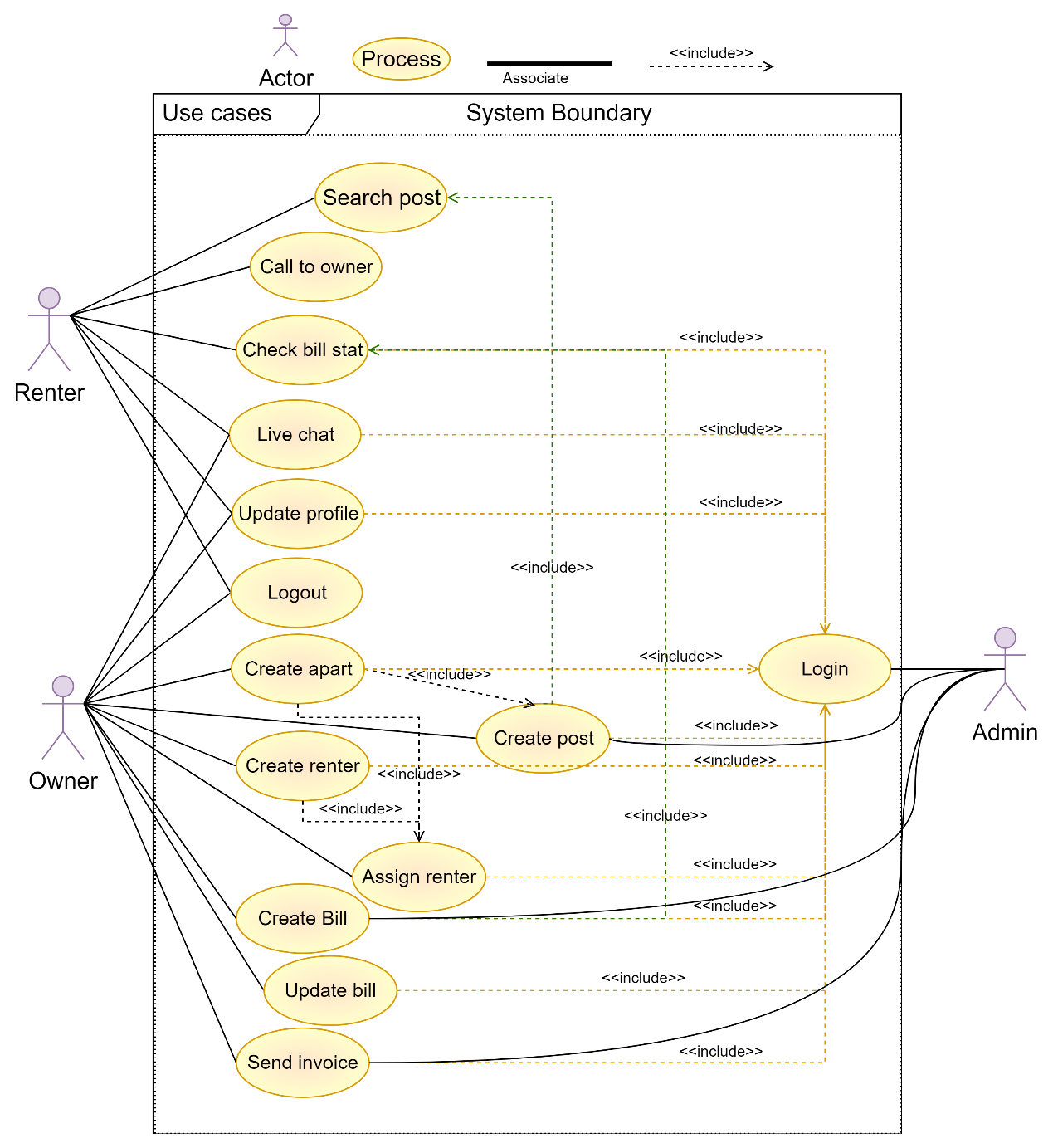
A use case diagram shows how the actors (users and the system) and the different use cases (functions) in the Homify app interact and relate to each other. It gives a general description of how the app works and who is involved. The use case diagram helps figure out the different user roles, how they work together, and what the app's main features are.

Figure 3.3 1 Use Case Diagram

### 3.4 Activity Diagram

An activity diagram shows how the different tasks or processes in the Homify app work together. It shows visually the steps, decisions, and actions that happen during certain processes, like registering a user or listing a property. The activity diagram makes the workflow clear and helps find any possible bottlenecks or places where things could be better.

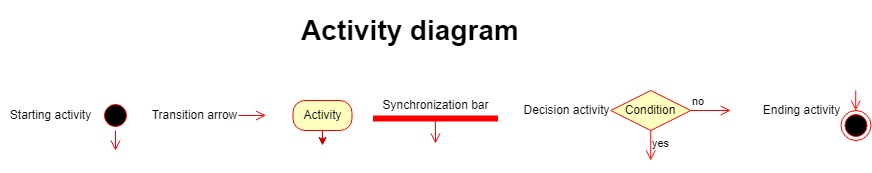
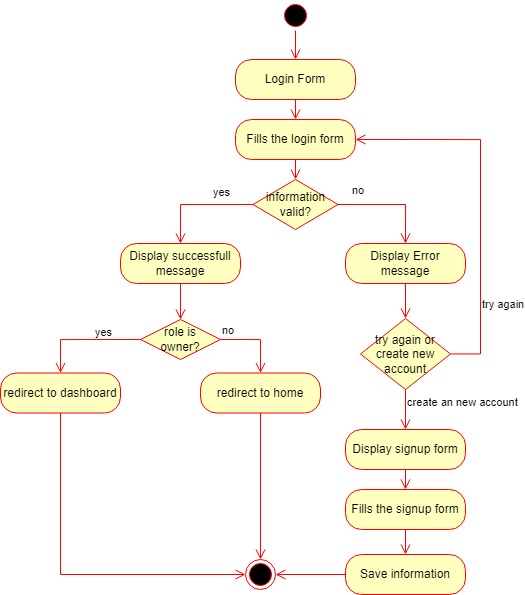


Figure 3.4 1 User login

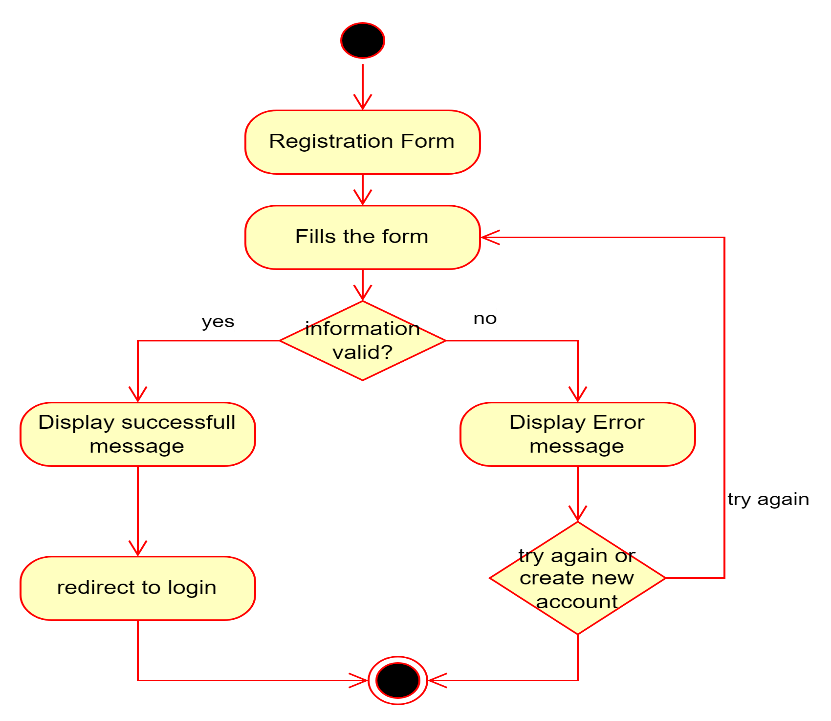


Figure 3.4 2 User Registration

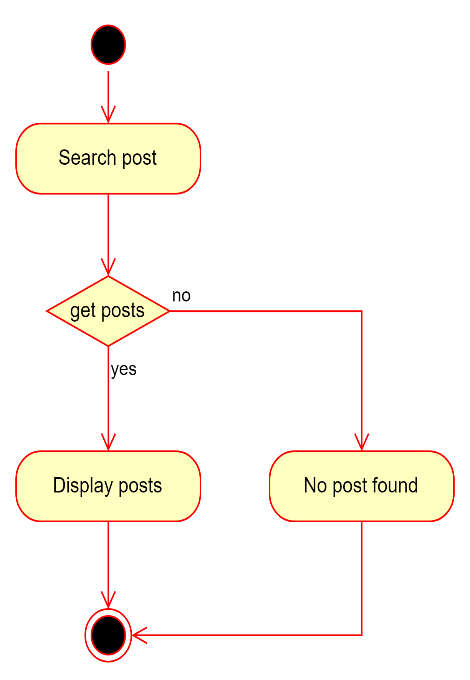


Figure 3.4 3 Search Post

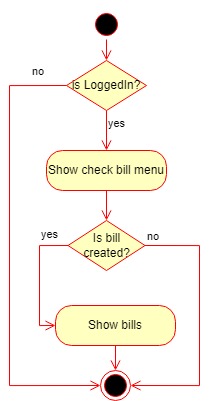


Figure 3.4 4 Check Bill

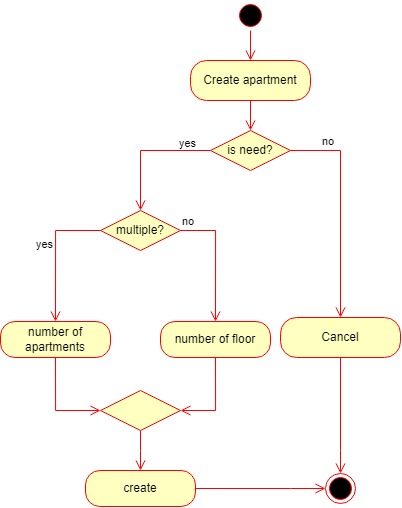
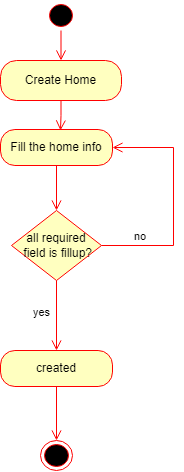


Figure 3.4 5 Create Home

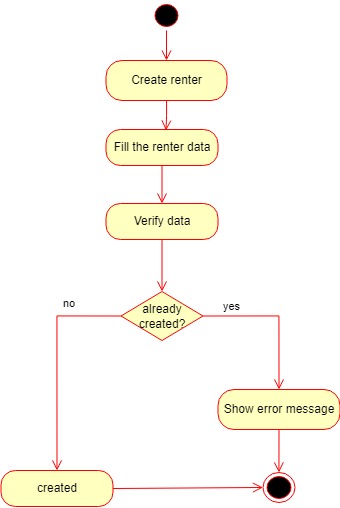
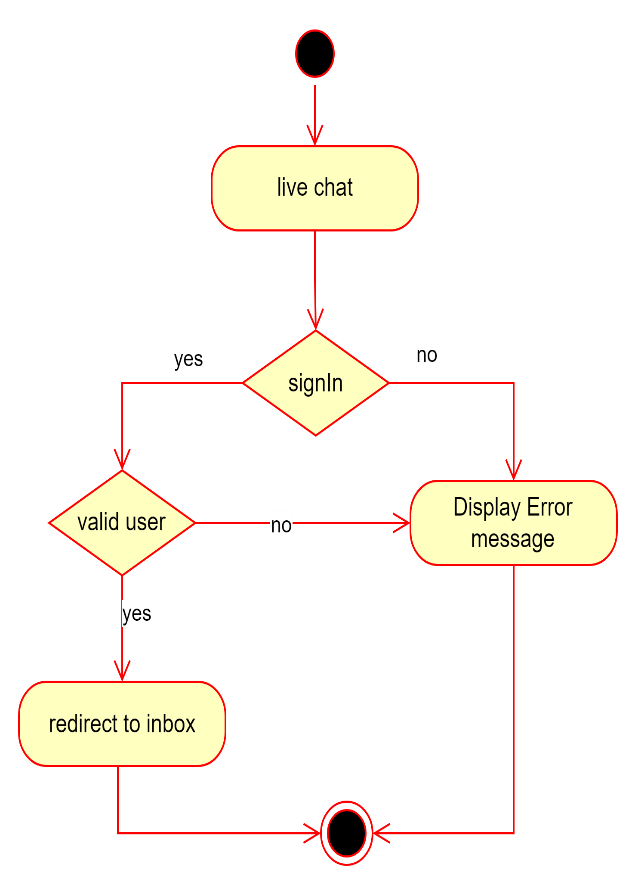


Figure 3.4 6 Live chat

Figure 3.4 7 Create Renter

Figure 3.4 8 Create Apartment

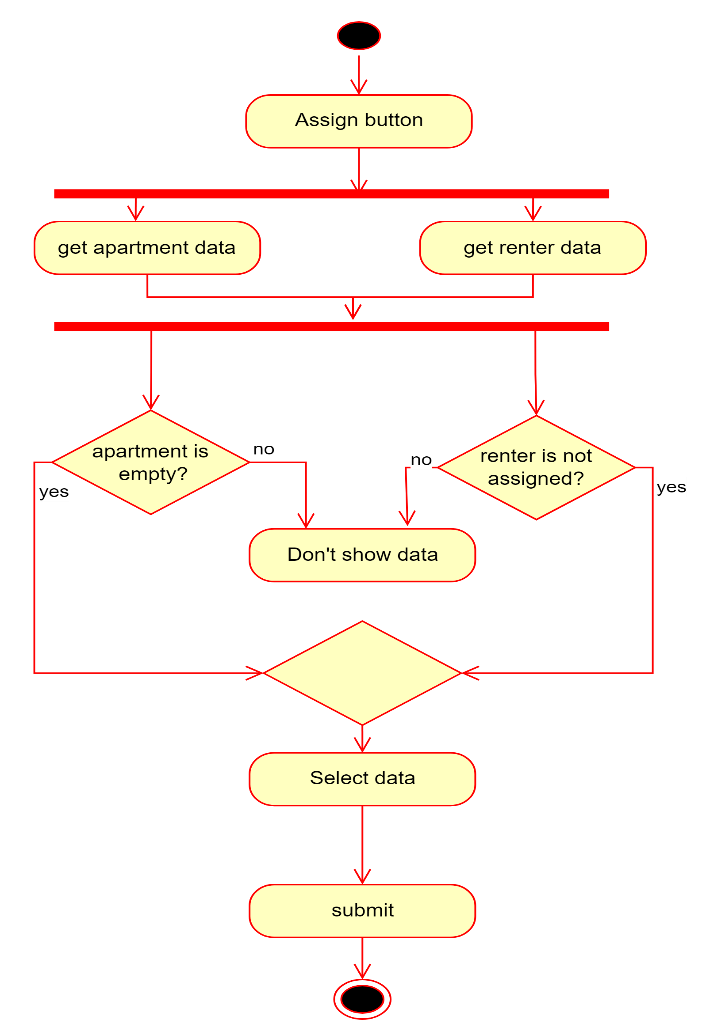
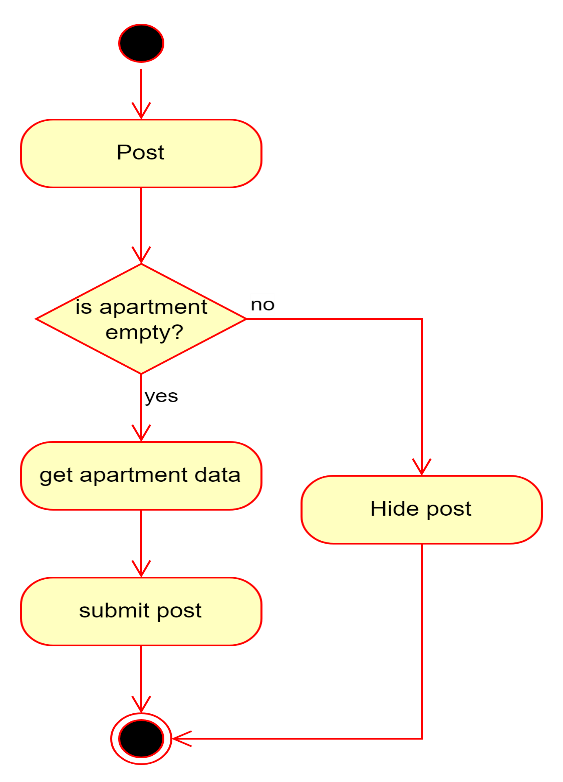


Figure 3.4 9 Assign Renter

Figure 3.4 10 Post Apartment

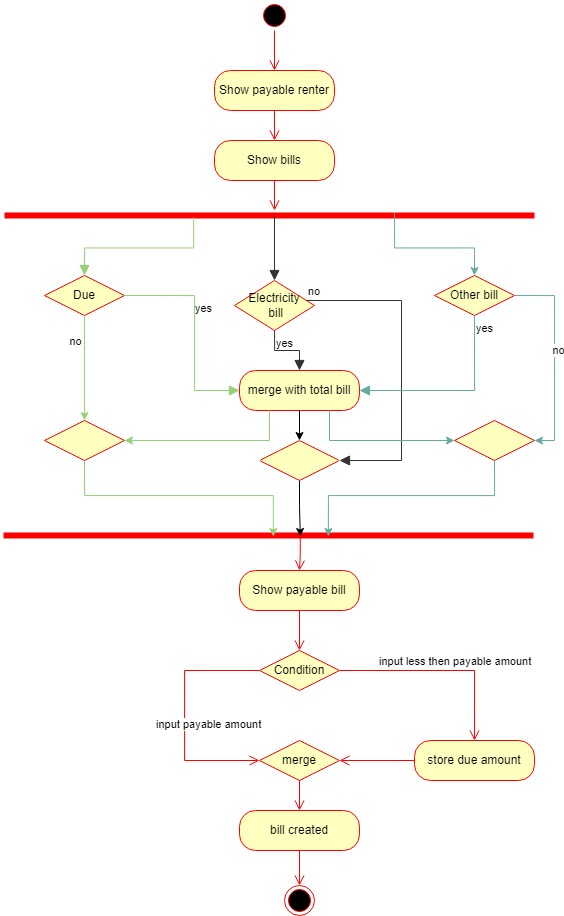


Figure 3.4 11 Make Bill

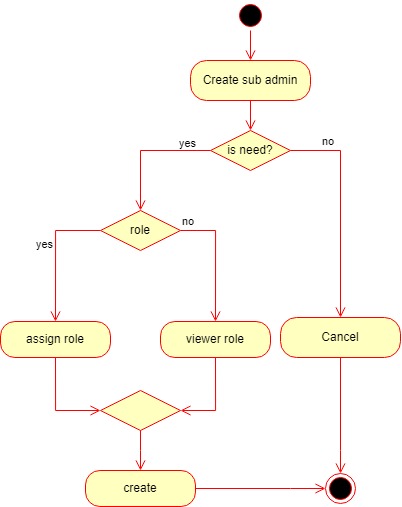


Figure 3.4 12 Create Admin

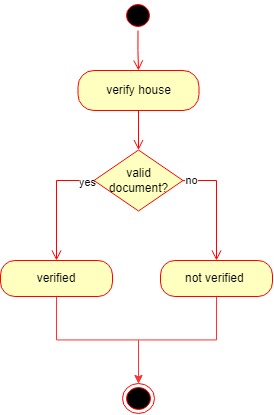
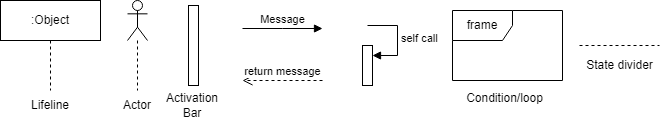


Figure 3.4 13 Verify House

### 3.5 Sequence Diagram

A sequence diagram shows the order in which the different parts or actors of the Homify app interact with each other. It shows how different parts communicate to each other and share information when certain things happen, like when a user contacts a homeowner or when a homeowner makes a monthly bill invoice. The sequence diagram helps you see how things work together and figure out how the system works.



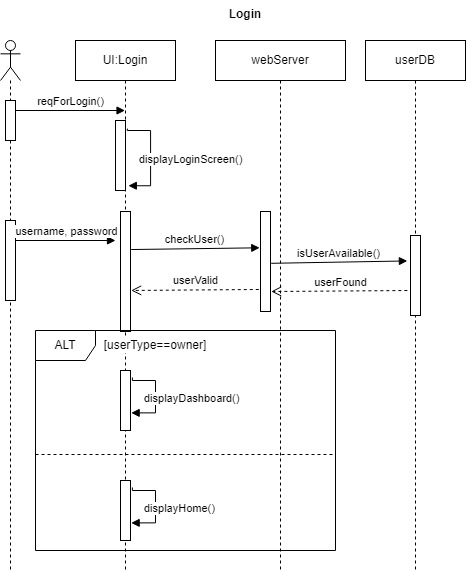
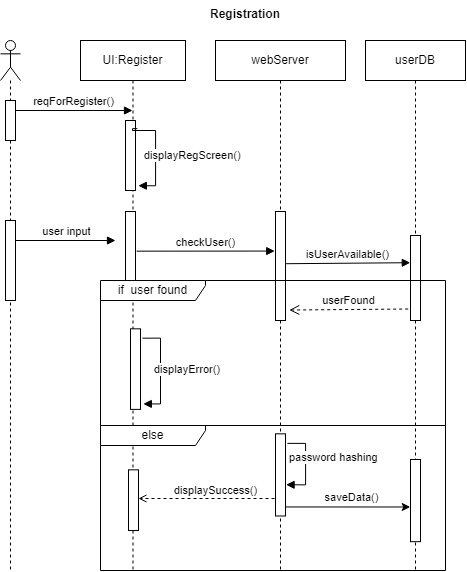


Figure 3.5 1 Login

Figure 3.5 2 Registration



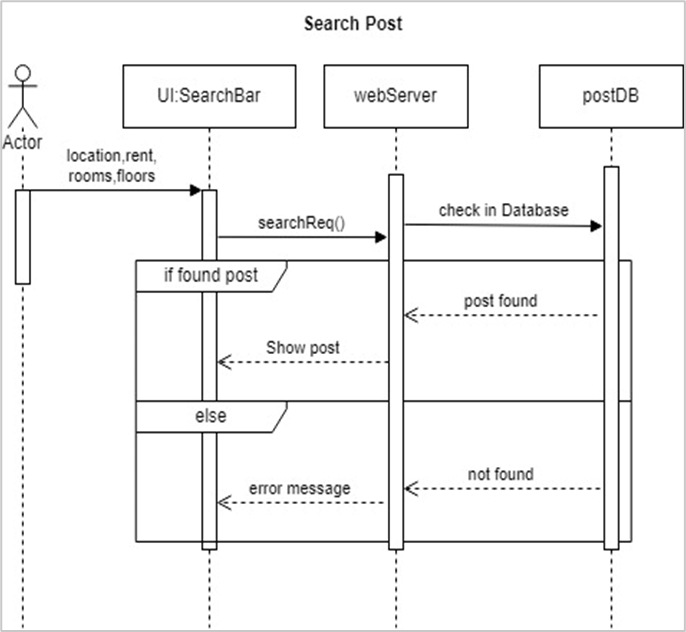


Figure 3.5 3 Search Post

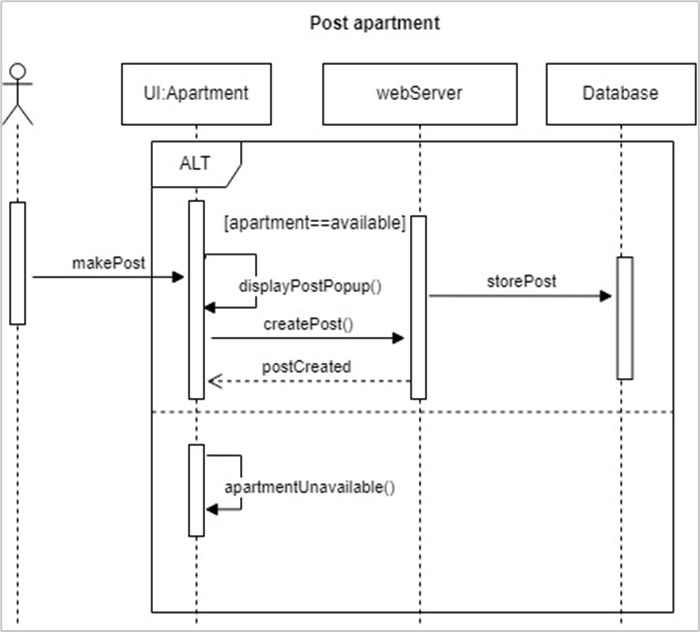


Figure 3.5 4 Post Apartment

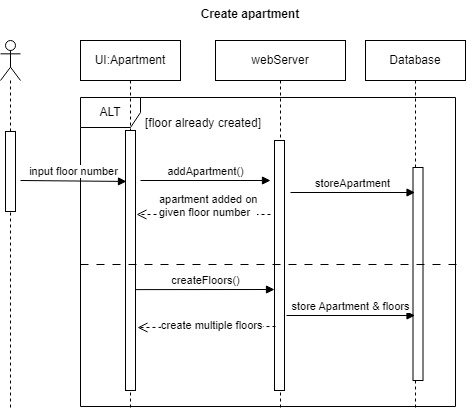


Figure 3.5 5 Create Apartment

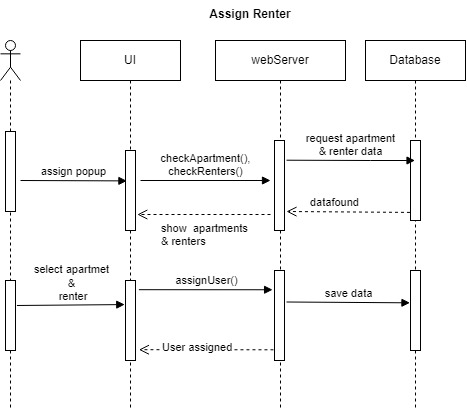


Figure 3.5 6 Assign Renter

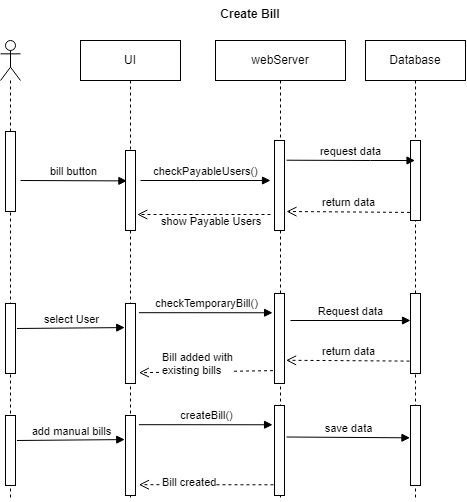


Figure 3.5 7 Create Bill

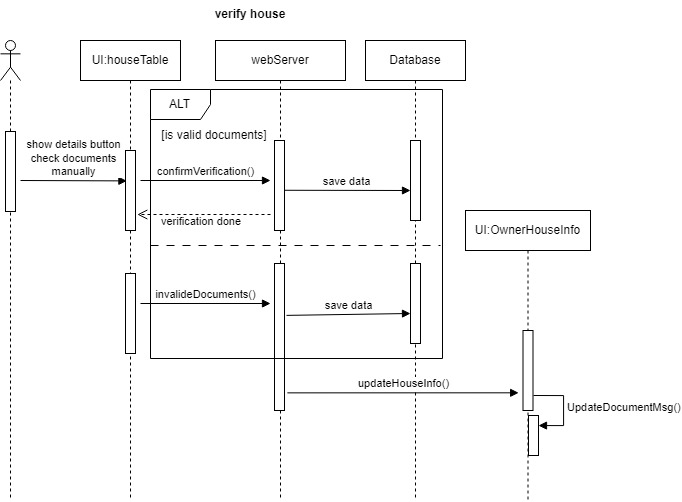


Figure 3.5 8 Verify House

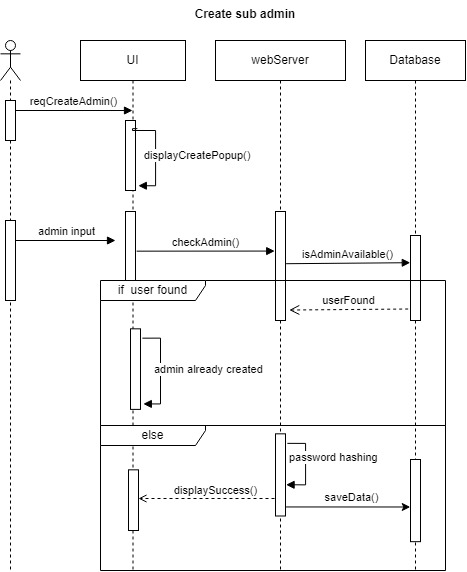


Figure 3.5 9 Create Admin

### 3.6 Database Modeling

Homify's database is based on the document-based approach of MongoDB. It organizes and stores data using collections and documents. The data model shows how entities like "renter," "owner," "house," "address," "bill," and "posts" are connected. The flexible schema of MongoDB makes it easy to adapt to changing needs and quickly find the data you need.



### 3.7 Technology Stack

**Front-End:**

* HTML
* CSS
* Tailwindcss
* Mantine
* JavaScript
* React
* React table
* Redux
* Axios
* RTK Query

**Back-End:**

* Node.js
* Express
* JWT (JSON Web Tokens)
* CryptoJs
* Passport.js
* RESTful API

**Database:**

* MongoDB (NoSQL database)

**External Services:**

* Cloudinary (Image storage)
* SMS gateway (for instant notifications)
* Socket.io (for instant messages and notifications)

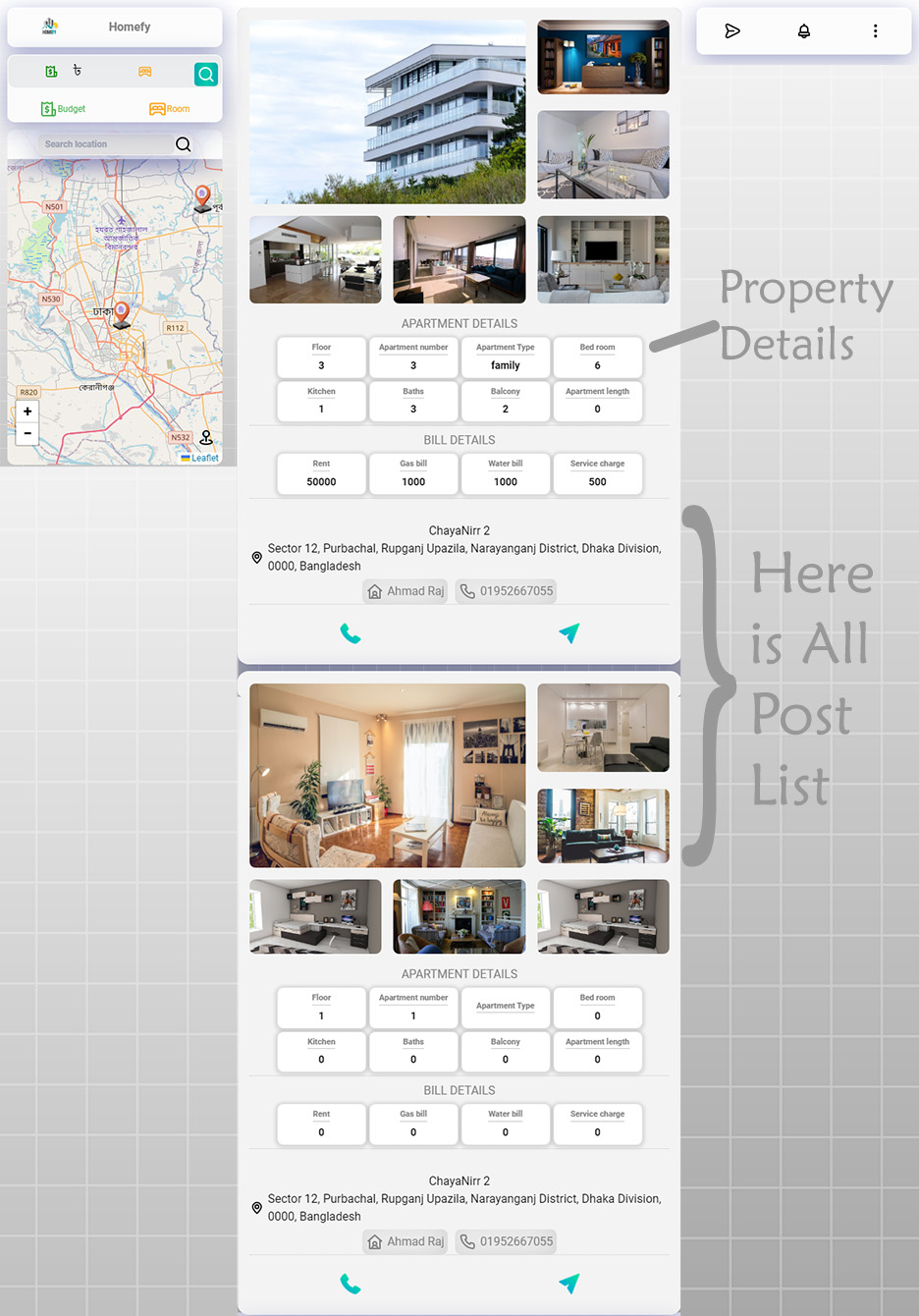
## CHAPTER 4

## USER INTERFACE

### 4.1 Public User Interface

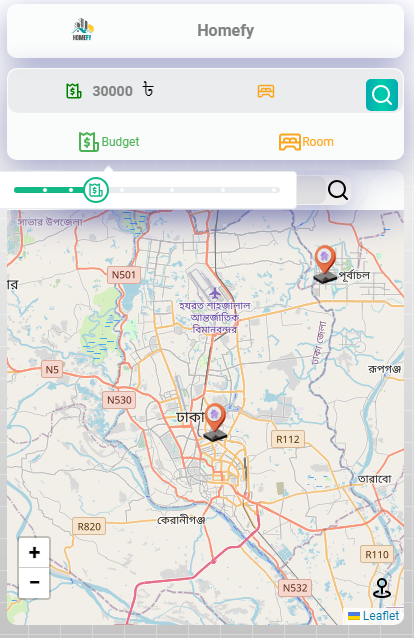
The public user interface is designed for users who are searching for rental properties. It provides the following features and functionalities:

### 4.1.1 Property Listings

Users can look through a list of properties that are for rent. The listings include important details like where the property is, how big it is, what amenities it has, how much it costs to rent, and pictures.

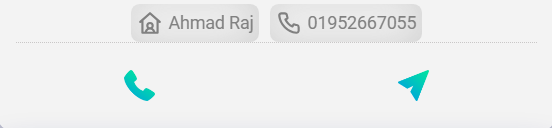
### 4.1.2 Search and Filters

Users can perform searches based on their preferences, such as location, rental price range, property type, etc. The app provides filters to narrow down the search results and find properties that match their criteria.



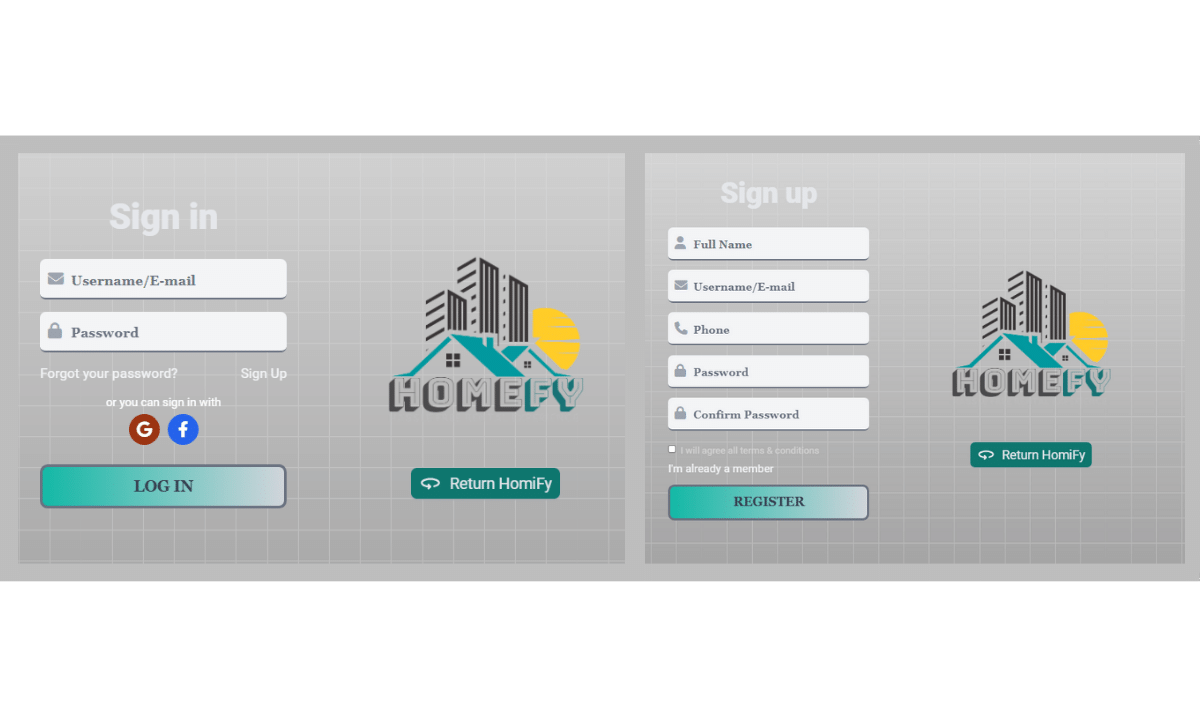
### 4.1.3 Contact Homeowners

Users can view detailed property descriptions and contact the homeowners for more information. Contact options include email, phone calls, and messaging.



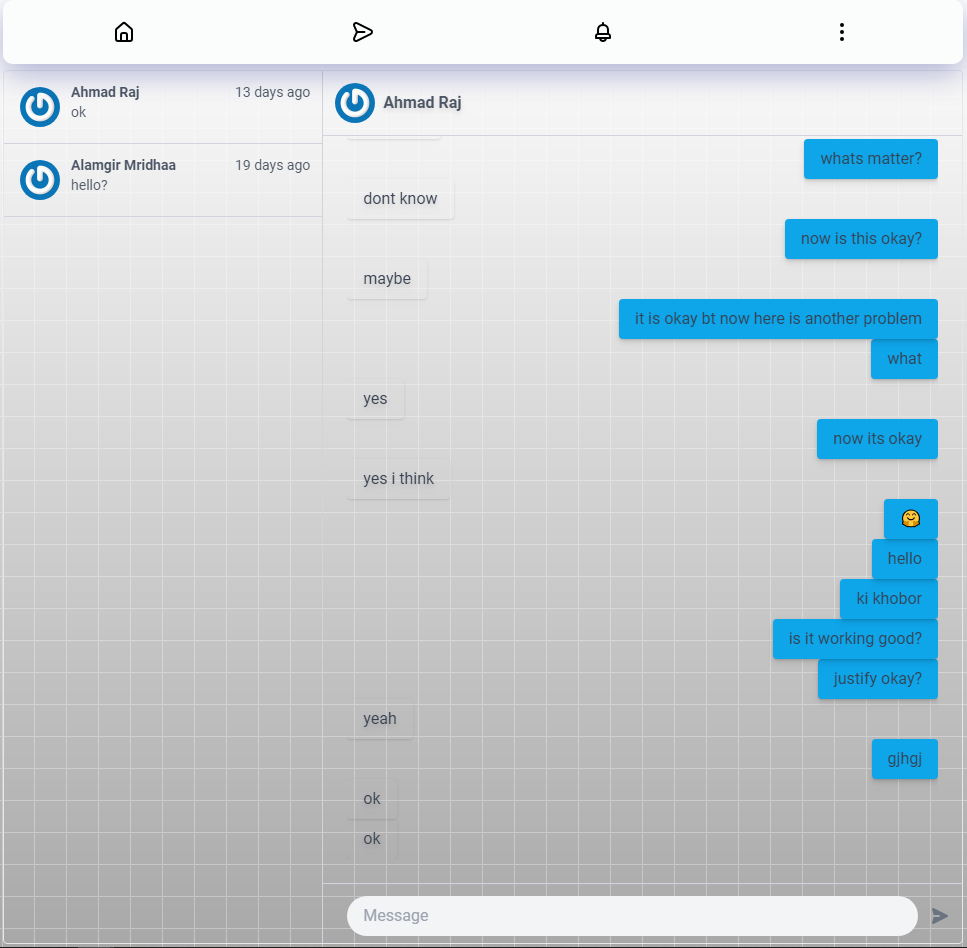
### 4.1.4 User Account

Public users can create an account to save their personal information, track their transaction history with homeowners, and receive notifications from property owners, and many more.





### 4.1.5 Live Chat

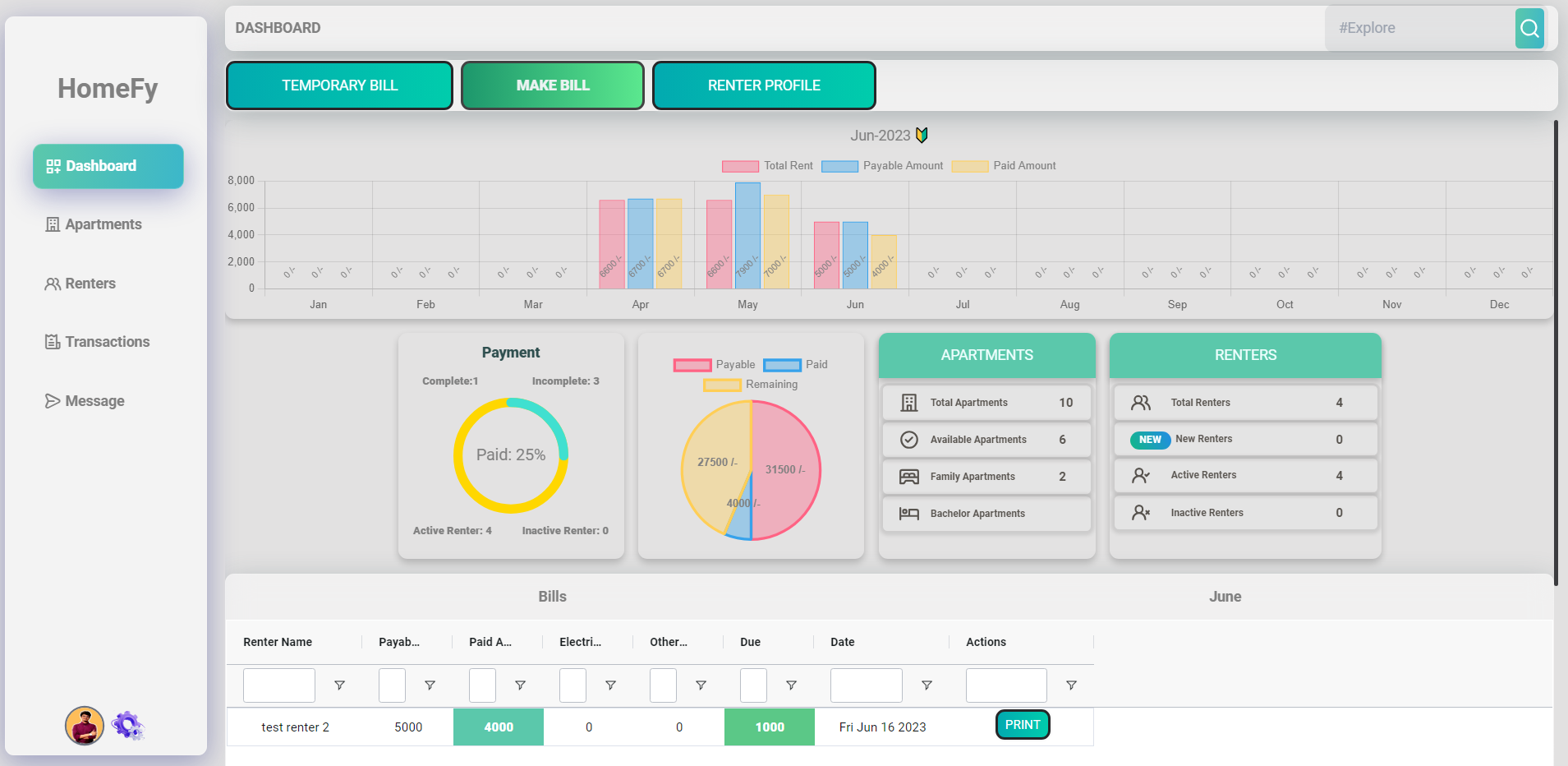
Registered users can live chat with homeowner and help center.

### 4.2 Homeowner Interface

The homeowner interface is designed for property owners who want to rent out their properties. It provides the following features and functionalities:

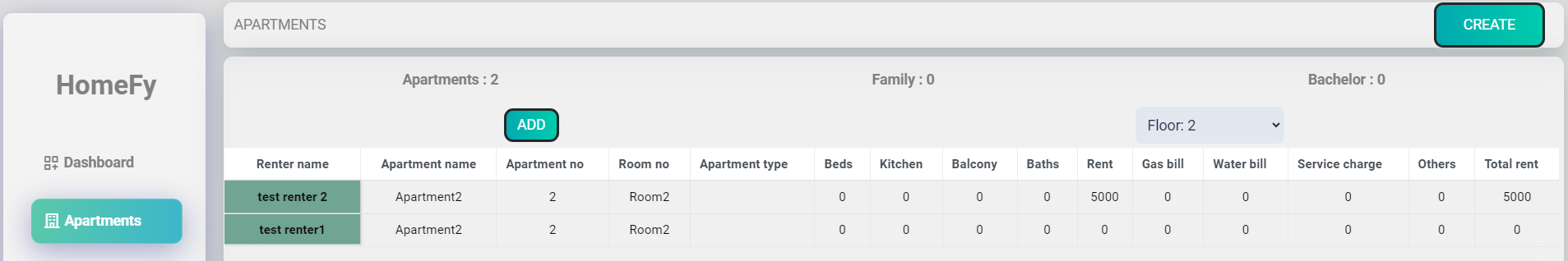
### 4.2.1 Homeowner Dashboard

The homeowner dashboard in the Homify app offers a comprehensive overview of property-related information and statistics. It provides graphical representations and visualizations to help homeowners analyze and track various aspects of their rental properties.



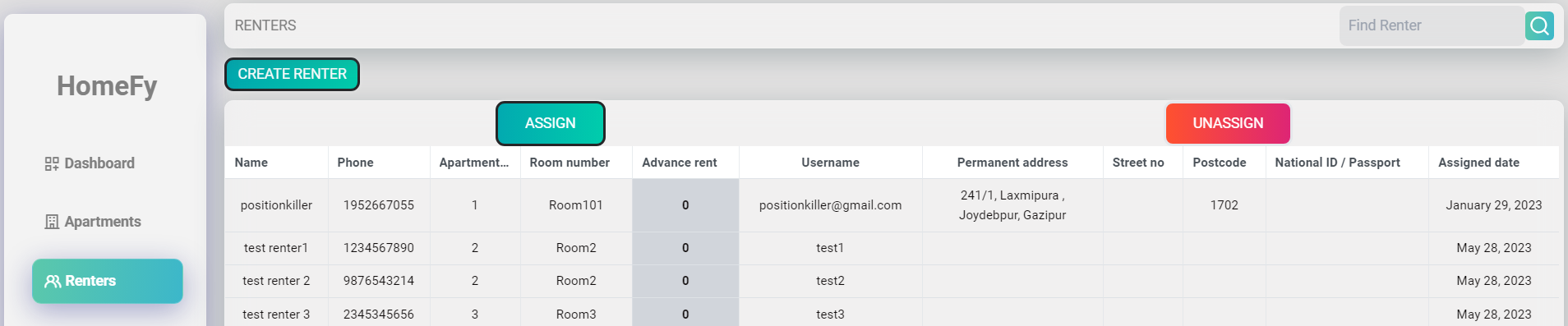
### 4.2.2 Property Management

Homeowners can create and manage listings for their properties. They can input property details such as location, size, and rental terms, and upload images to showcase the property.



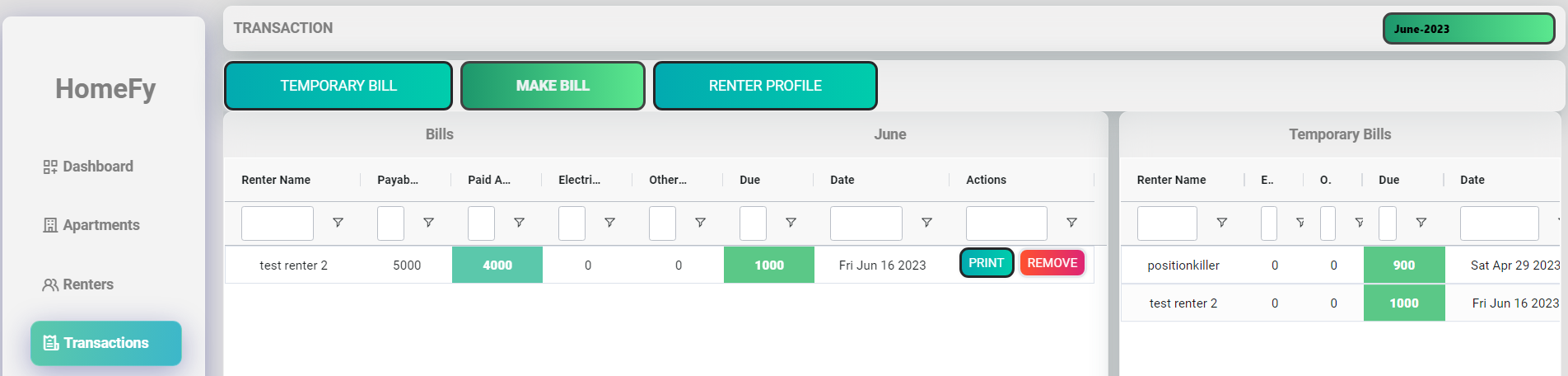
### 4.2.3 Renter Management

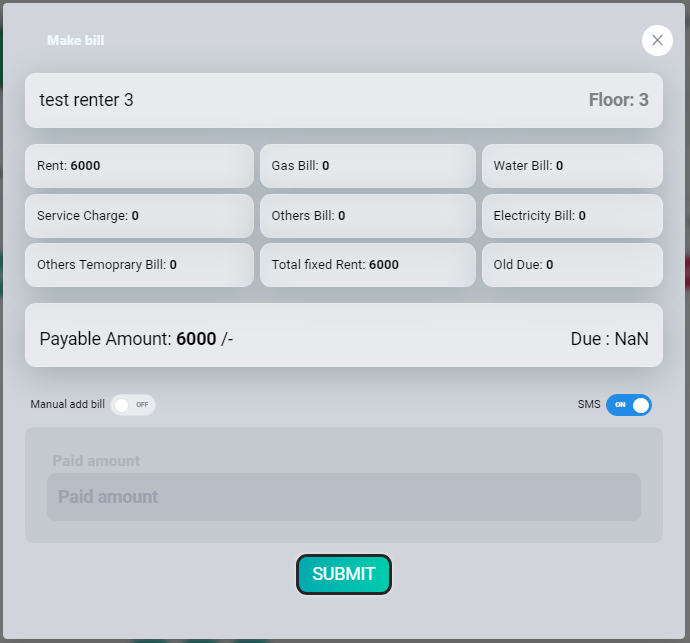
Homeowner can manage renter, including creating new accounts, assigning or unassigning apartment and remove from home.



### 4.2.4 Billing and Invoicing

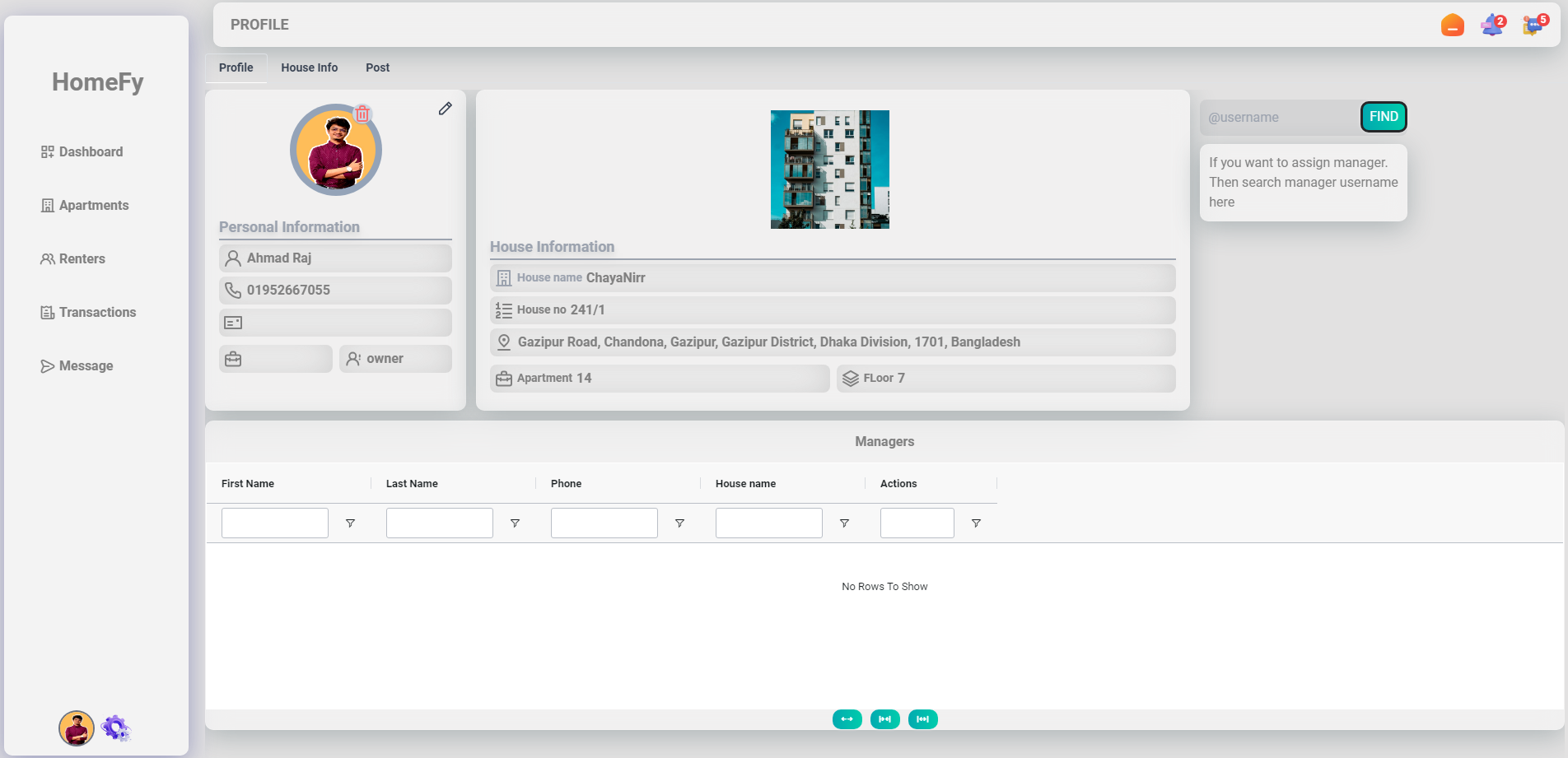
Homeowners can track rental payments, generate monthly bill invoices for renters, and manage bills associated with the property. They can view payment histories and send invoice to renters by SMS.





### 4.2.5 Homeowner Profile

Owner have a personal profile section where they can assign a manager to handle the day-to-day management tasks.



### 4.2.6 Home Information

In home information section homeowner can create multiple home and active single home at a time. Then under this activate home they can manage all things.



### 4.2.7 Post Section

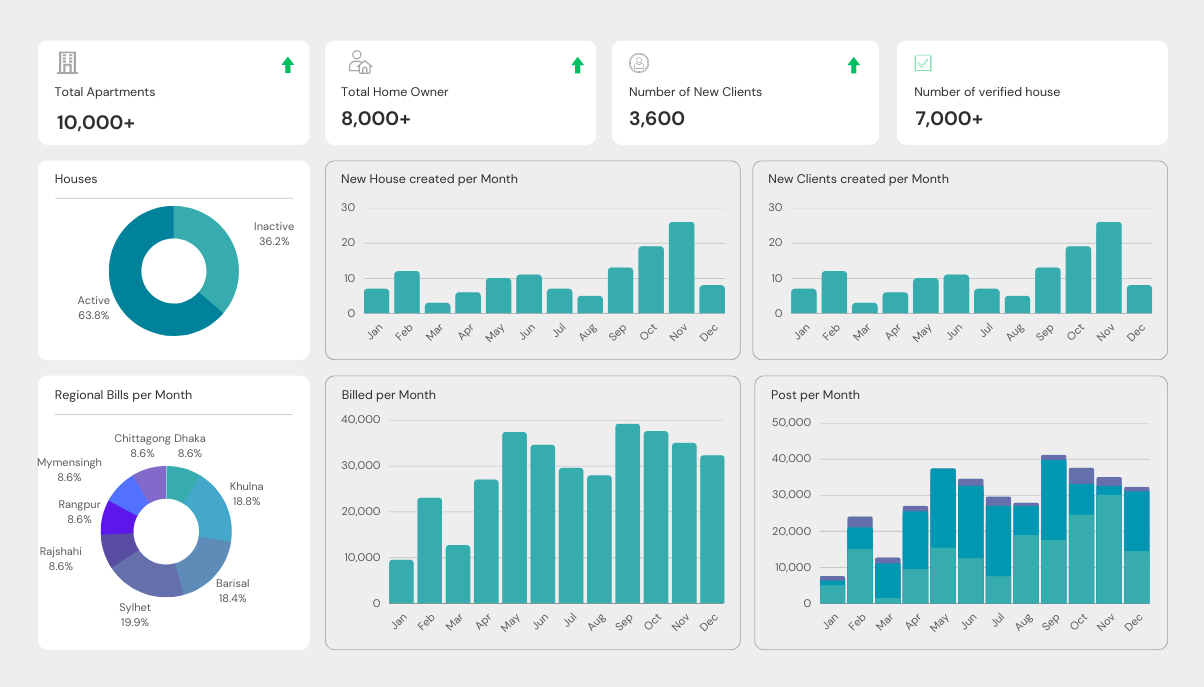
In here all posts are shown under specific home. Homeowner can update active/inactive or delete post from here.



### 4.3 Admin Interface

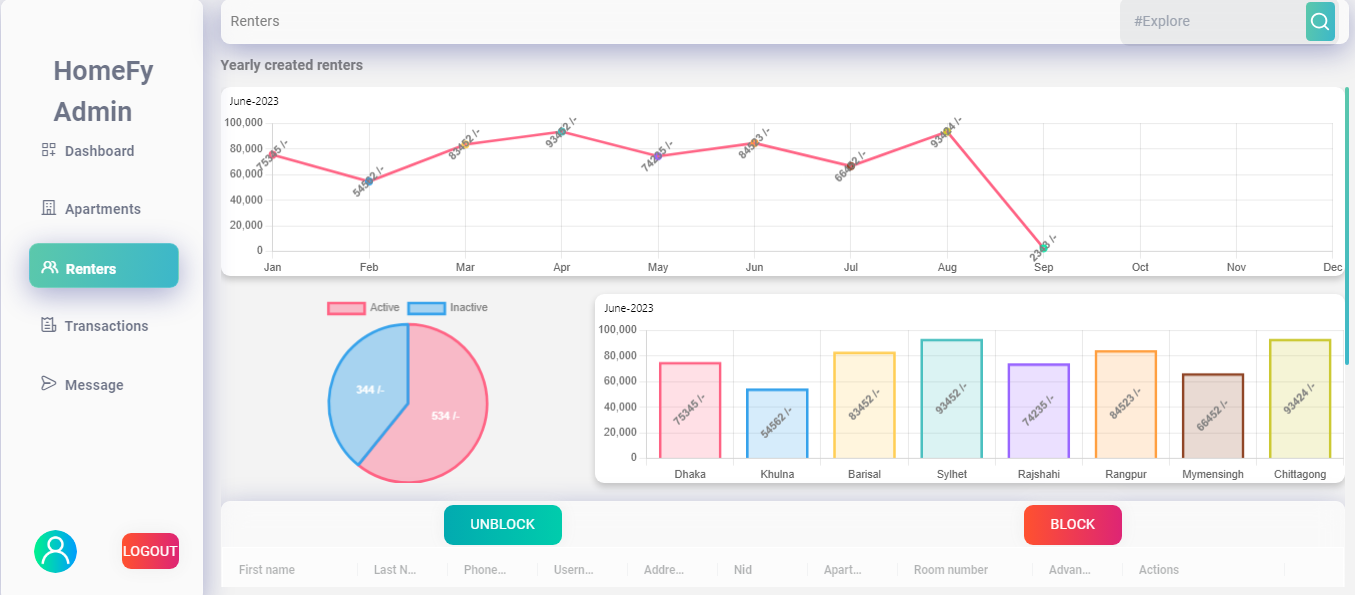
### 4.3.1 Admin Dashboard

The admin dashboard shows all reports and access analytics related to property listings, user activities, and system performance. These insights help in monitoring the app's usage, identifying trends, and making data-driven decisions.



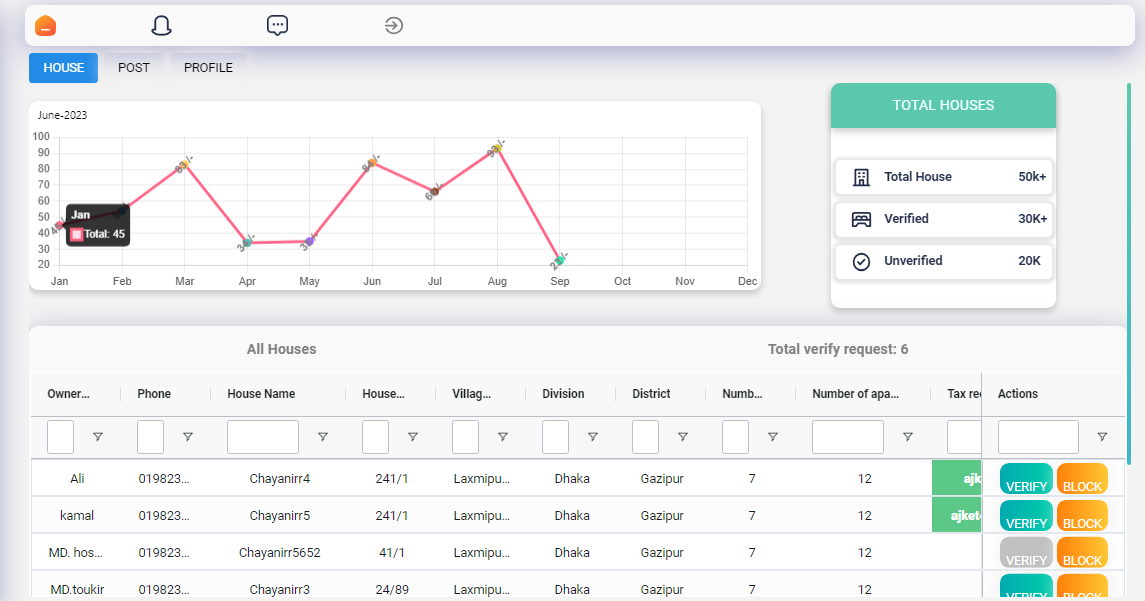
### 4.3.2 User Management

Admin can manage all users. They can block and unblock users for their activities and if some user faces any problem, then they can check their activity.



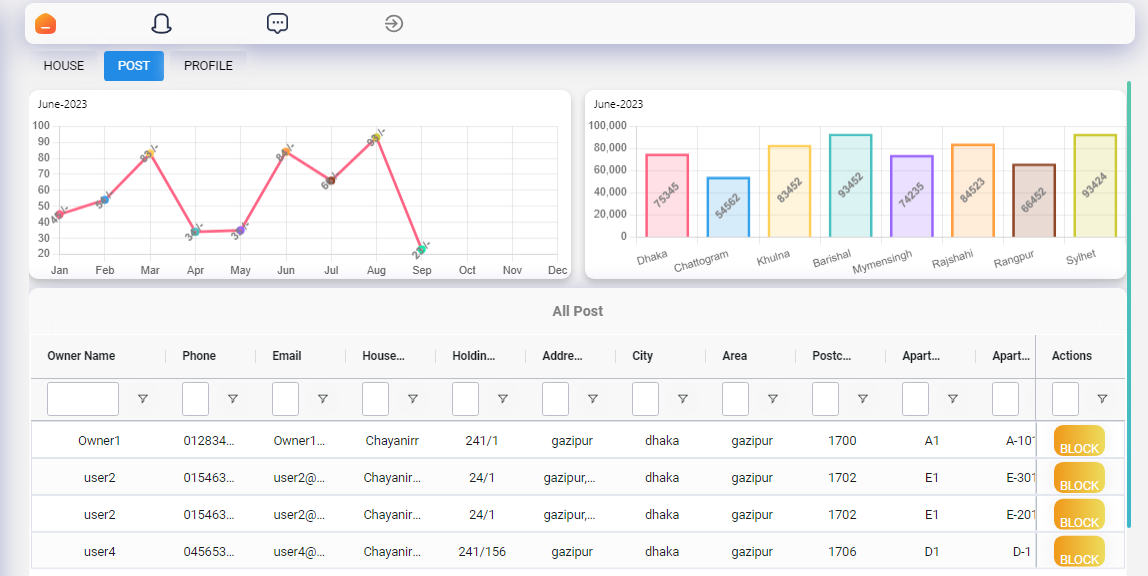
### 4.3.3 House Management

In this section admin can saw all information about houses and they can verify new houses. If submitted document is valid then they approve the house as a verified house.



### 4.3.4 Post Management

Admin can check all post. If any one makes a fake post, then they can block it form this section. Also, they can saw all posts activates form here.



The user interfaces in the Homify app are designed to cater to the specific needs of public users, homeowners, and administrators, providing them with a seamless and intuitive experience while interacting with the app and managing their respective tasks.

## CHAPTER 5

## SYSTEM TESTING

### 5.1 Testing Phases

System testing is one of the most important parts of making the Homify app. It makes sure that all of the system's parts and functions are working as they should, meeting the requirements, and giving the user a good experience. The following methods and approaches are used for testing:

1. **Unit Testing**: Unit tests are conducted to verify the correctness of individual components or modules in isolation. This testing approach ensures that each unit of code functions as expected. Unit tests are written for both front-end and back-end components using appropriate testing frameworks and libraries.
2. **Integration Testing**: Integration testing is performed to validate the proper interaction and communication between different modules and components of the Homify app. It ensures that the integrated system functions as a cohesive unit and that data flows correctly between various layers and services.
3. **Functional Testing**: Functional testing focuses on testing the app's functionalities from the perspective of different user roles, including public users, homeowners, and administrators. It involves verifying that all user actions, such as property search, property listing creation, renter communication, and bill generation, work correctly and produce the expected results.
4. **User Interface (UI) Testing**: UI testing is conducted to ensure that the user interfaces of the Homify app are visually appealing, responsive, and provide a seamless user experience. It involves verifying the layout, navigation, forms, buttons, and other UI elements across different devices and screen sizes.
5. **Performance Testing**: Performance testing is carried out to evaluate the app's performance under various load conditions. It measures response times, throughput, resource utilization, and scalability to identify any performance bottlenecks or areas for optimization. Performance testing helps ensure that the app can handle a large number of concurrent users and deliver a smooth user experience.
6. **Security Testing**: Security testing is performed to identify and mitigate potential vulnerabilities in the Homify app. It involves testing for common security issues such as authentication and authorization flaws, input validation vulnerabilities, and protection against common attacks like SQL injection and cross-site scripting (XSS).
7. **Usability Testing**: Usability testing focuses on evaluating the app's ease of use and user satisfaction. It involves gathering feedback from real users to assess how well the app meets their needs, whether the user interfaces are intuitive and user-friendly, and if any improvements or adjustments are needed to enhance the overall usability.
8. **Regression Testing**: Regression testing is conducted after implementing new features, bug fixes, or modifications to ensure that the existing functionalities have not been affected. It involves retesting previously tested scenarios to confirm that the changes have not introduced any unintended side effects.

### 5.1 Testing Schedule

|  |  |
| --- | --- |
| **Test Phase** | **Time** |
| Planning | 1 W |
| Unit Testing | During development time |
| Component Testing | During development time |
| UI Testing | 1W |
| Usability Testing | 3 Days |
| Performance Testing | 3 Days |
| Database Testing | 1w |
| Performance Testing | 1w |
| Compatibility Testing | 1w |
| Security Testing | 2w |

## CHAPTER 6

## Conclusion

In conclusion, Homify is an app that connects people who want to rent out their homes with people who might be interested in renting them. It makes it easy for people to look for rental properties, communicate to the owners, and handle other renting-related tasks. With its easy-to-use interfaces, wide range of features, and strong functionality, Homify makes the experience of both public users and homeowners smooth and easy.

Homify has done a good job of fulfilling the needs of homeowners and renters, but it has some flaws. One problem that stands out is that there is no way to pay online. Currently, Homify does not help homeowners and renters make online payments to each other. But users can still pay for rentals with traditional methods like cash, checks, and bank transfers. Adding an online payment system to the rental process in the future would make it easier and safer.

Even with this limitation, Homify has a lot of potential for the future. Some possible areas for growth and improvement are as follows:

1. **Online Payment Integration**: By adding a secure online payment system, homeowners and renters would be able to handle rental payments directly through the app, making the process smooth and easy.
2. **Enhanced Communication Features**: Adding more ways to talk, like in-app messaging or video calls, would make it easier for homeowners and renters to talk to each other and help them talk quickly and easily.
3. **Advanced Property Management Tools**: Adding more sophisticated property management features to the homeowner dashboard would provide owners more control over the day-to-day operations of their rental properties.
4. **Localization and International Expansion**: Adding localization features and making the app available in more countries would attract a wider range of users and meet their different rental needs.

By addressing these areas for improvement and adopting new technology, Homify will be able to continue to evolve and provide an amazing rental experience for users, cementing its position as a top real estate app.

In summary, Homify has successfully provided a user-friendly and efficient platform for homeowners and renters to connect and manage rental properties. While it currently lacks an online payment system, there are exciting opportunities for future enhancements and expansions. By continuously refining its features and addressing user needs, Homify has the potential to become a comprehensive and indispensable tool for the real estate rental market.

## Frequently Asked Questions (FAQ)

**Q1: How can I search for rental properties on Homify?**

A: To search for rental properties, simply go to the Homify app and use the search bar or filters to specify your preferences such as location, rental price range, property type, and amenities. The app will display a list of available properties that match your criteria.

**Q2: How can I contact a homeowner regarding a property listing?**

A: Once you find a property of interest, you can contact the homeowner by clicking on the property listing. You will find contact options such as email, phone number, or messaging. Choose the preferred contact method and reach out to the homeowner for further information or to schedule a property viewing.

**Q4: How do I create a property listing as a homeowner?**

A: If you are a homeowner looking to rent out your property, you can create a property listing by logging into your Homify account and accessing the homeowner dashboard. From there, you can provide property details such as location, size, amenities, rental terms, and upload images to showcase your property.

**Q5: How can I track rental payments and manage bills as a homeowner?**

A: Homify provides a billing and invoicing feature for homeowners. You can track rental payments, generate monthly bill invoices for renters, and manage bills associated with your property through the homeowner dashboard. The dashboard will display payment histories, outstanding bills, and upcoming payment reminders.

**Q6: Can I assign a manager to handle my property if I have multiple properties?** A: Yes, if you have multiple properties, you can assign a manager to assist you with property management tasks. The manager will have access to your homeowner dashboard and can help with property listings, communication with renters, and other rental-related activities.

**Q7: How secure is my personal information on Homify?**

A: Homify takes the security and privacy of user information seriously. We implement robust security measures and follow industry best practices to protect your personal data. We do not share your information with third parties without your consent. For more details, please refer to our Privacy Policy.

**Q8: What should I do if I encounter an issue or have a technical problem on Homify?**

A: If you encounter any issues or have technical problems while using Homify, you can reach out to our customer support team. We have a dedicated support channel where you can report the problem and receive assistance. Our team will promptly address your concerns and provide the necessary support.

These are just a few examples of common user queries that can be included in the FAQ section of the Homify app. The FAQ section serves as a helpful resource for users, addressing their most frequently asked questions and providing them with quick and accurate answers to enhance their experience with the app.

## ReferencesTop of Form

During the development of the Homify app, the following resources and references were utilized:

1. Agile Manifesto: <https://agilemanifesto.org/>
   * The Agile Manifesto was referred to for understanding the principles and values of the Agile development model, which guided the development process of the Homify app.
2. React Documentation: <https://reactjs.org/docs>
   * The official React documentation was used as a reference to understand React's concepts, components, and best practices, which played a crucial role in building the front-end of the Homify app.
3. Node.js Documentation: <https://nodejs.org/en/docs/>
   * The Node.js documentation was referenced to explore Node.js capabilities, APIs, and modules, which formed the foundation of the Homify app's back-end development.
4. MongoDB Documentation: <https://docs.mongodb.com/>
   * The MongoDB documentation provided insights into MongoDB's document-oriented database system, helping in designing and implementing the database structure of the Homify app.
5. Cloudinary Documentation: <https://cloudinary.com/documentation>
   * The Cloudinary documentation was consulted to understand the features and integration options for storing and managing images in the cloud, which was utilized for image storage in the Homify app.
6. HTML, CSS, JavaScript Documentation:
   * Standard web development documentation and references were utilized to implement HTML, CSS, and JavaScript features, ensuring the app's responsiveness, styling, and interactivity.
7. Redux Documentation: <https://redux.js.org/>

* The Redux documentation served as a valuable resource for understanding the concepts and implementation of Redux, a state management library used in the Homify app. It provided guidance on managing application state, actions, reducers, and store configuration.

1. Tailwind CSS Documentation: <https://tailwindcss.com/docs>

* The Tailwind CSS documentation was used as a reference for understanding the utility-first CSS framework and its usage in the Homify app. It provided detailed documentation on utility classes, responsive design, customizing styles, and optimizing CSS workflow.

1. OpenAI's ChatGPT: <https://www.openai.com/>

* OpenAI's ChatGPT, a language model powered by GPT-3.5, was utilized as a valuable resource throughout the documentation process for the Homify app. ChatGPT provided real-time assistance, generated content, and answered questions related to various aspects of the project, including system overview, requirements analysis, technology stack, and more.