

# Development of Online Based House Renting Web Application

By

**Raju Miah**  
CSE1901016171

**Turjaun Akter Onika**  
CSE1901016003

**Tamanna Akter**  
CSE1901016147

**MD. Jahid Hasan**  
CSE1901016009

Supervised by  
**Sadia Tasnim Barsha**

Submitted in partial fulfillment of the requirements for the degree of  
Bachelor of Science in Computer Science and Engineering



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
SONARGAON UNIVERSITY (SU)**

**January - 2023**

# APPROVAL

The Project Report “**Development of Online Based House Renting Web Application**” submitted by Raju Miah (ID: CSE1901016171), Turjaun Akter Onika (ID: CSE1901016003), Tamanna Akter (ID: CSE1901016147), MD. Jahid Hasan (ID: CSE1901016009) to the Department of Computer Science and Engineering, Sonargaon University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

## Board of Examiners

-----

**Sadia Tasnim Barsha**

Lecturer,

Department of Computer Science and Engineering

Sonargaon University (SU)

**Supervisor**

-----

(Examiner Name & Signature)

Department of Computer Science and Engineering

Sonargaon University (SU)

**Examiner 1**

-----

(Examiner Name & Signature)

Department of Computer Science and Engineering

Sonargaon University (SU)

**Examiner 2**

-----

(Examiner Name & Signature)

Department of Computer Science and Engineering

Sonargaon University (SU)

**Examiner 3**

# DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Sadia Tasnim Barsha**, Lecturer, Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this project has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

Signature

-----  
**(Sadia Tasnim Barsha)**  
**Supervisor**

-----  
Raju Miah  
ID: CSE1901016171

-----  
Turjaun Akter Onika  
ID: CSE1901016003

-----  
Tamanna Akter  
ID: CSE1901016147

-----  
MD. Jahid Hasan  
ID: CSE1901016009

# **ABSTRACT**

We propose rent house system with the PHP, JavaScript, HTML5, CSS3, CodeIgniter 4, jQuery and MySQL technologies for users. With the current paradigm shift in technological field, there is an urgent need to embrace and appreciate power of technology. This website helps user to register individual home or apartment to assist you in finding the perfect rental home or property. Also, we can find your next rental search view in your targeted area. There will be facility for a user to upload room details (if he is a landlord) and to search for rooms available on rent (if he wants to be a tenant).

# ACKNOWLEDGMENT

At the very beginning, I would like to express my deepest gratitude to the Almighty Allah for giving me the ability and the strength to finish the task successfully within the schedule time.

I am auspicious that I had the kind association as well as supervision of **Sadia Tasnim Barsha**, Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out my project.

We thankfully acknowledge the support and inspiration that we received from our teachers specially **Bulbul Ahmed**, Associate Professor & Head, Computer Science and Engineering Department.

We would also like to thank our classmates for letting my defense be an enjoyable moment and for your brilliant comment and suggestions on this project.

Finally, I am deeply grateful to my parents for their supports, appreciation, encouragement, and keen interest in my academic achievements.

# LIST OF ABBREVIATIONS

CMS	Content Management System
CSS	Cascading Style Sheets
DFD	Data Flow Diagram
DOM	Document Object Model
ECMAS	European Computer Manufacturers Association Script
ES6	ECMAScript 6
ERD	Entity Relationship Diagram
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
JS	JavaScript
PHP	Hypertext Preprocessor
RDBMS	Relational Database Management System
SQL	Structured Query Language
URL	Uniform Resource Locator

# TABLE OF CONTENTS

<b><u>Title</u></b>	<b><u>Page No.</u></b>
<b>APPROVAL</b> .....	<b>i</b>
<b>DECLARATION</b> .....	<b>ii</b>
<b>ABSTRACT</b> .....	<b>iii</b>
<b>ACKNOWLEDGEMENT</b> .....	<b>iv</b>
<b>LIST OF ABBREVIATION</b> .....	<b>v</b>
<b>CHAPTER 1</b>	<b>1 – 3</b>
<b>INTRODUCTION &amp; PROPOSE OF THE PROJECT</b>	
1.1 Introduction.....	1
1.2 Motivation.....	2
1.3 Objective.....	2
1.4 Statement of the Problem.....	3
1.5 Advantage of the System.....	3
<b>CHAPTER 2</b>	<b>4 – 6</b>
<b>BACKGROUND INFORMATION AND DEVELOPMENT PROCESS</b>	
2.1 Background.....	4
2.2 Process of Development.....	4-5
2.2.1 Waterfall Model.....	5
2.3 Relational Database Management System.....	6
<b>CHAPTER 3</b>	<b>7-17</b>
<b>SOFTWARE REQUIRMENTS AND SPECIFICATION</b>	
3.1 Software Requirments.....	8 –15
3.1.1 PHP.....	9 - 10
3.1.2 CodeIgniter 4.....	10-11
3.1.3 JavaScript.....	11-12
3.1.4 MySql.....	12
3.1.5 JQuery.....	13
3.1.6 HTML.....	14
3.1.7 CSS.....	14-15
3.2 Functional Requirement.....	15-16

3.3 Non-Functional Requirement.....	16-17
<b>CHAPTER 4</b>	<b>18-23</b>
<b>SOFTWARE DESIGN</b>	
4.1 Data Flow Diagram (DFD).....	18-19
4.2 Entity Relationship Diagram (ERD).....	20 - 21
4.3 Use Case Diagram.....	22-23
<b>CHAPTER 5</b>	<b>24 - 35</b>
<b>SOFTWARE OUTCOME AND IMPLEMENTATION</b>	
5.1 Home Page.....	24 - 25
5.2 Rent House Page.....	26
5.3 Contact us Page.....	27
5.4 Sign in.....	27 - 28
5.5 Sign up.....	29
5.6 House Owner.....	30 - 31
5.6.1 Your post.....	30
5.6.2 Post feature.....	30
5.6.2.1 Add post.....	30
5.6.2.2 Modify post.....	31
5.7 Renters.....	32
5.7.1 Bookmark.....	32
5.8 Admin panel.....	32 - 33
5.8.1 Dashboard.....	32
5.8.2 Notifications.....	33
5.8.3 Active Users.....	33
5.9 Message.....	34
5.10 View profile.....	34
5.11 Update profile.....	35
5.12 Logout.....	35



<b>CHAPTER 6</b>	<b>36</b>
CONCLUSION AND FUTURE WORKS	
6.1 Conclusion.....	36
6.2 Future Work.....	36
<b>REFERENCES</b> .....	<b>37</b>

# LIST OF FIGURES

---

<b><u>Figure No.</u></b>	<b><u>Title</u></b>	<b><u>Page No.</u></b>
Figure 2.2.1	Block Diagram of Waterfall Model	5
Figure 4.1	DFD Diagram	19
Figure 4.2	ER Diagram	21
Figure 4.3	Use Case Diagram	23
Figure 5.1	Home Page	25
Figure 5.2	Rent House page	26
Figure 5.3	Contact Us page	27
Figure 5.4	Sign in Page	28
Figure 5.5	Sign up Page	29
Figure 5.6.1	Your Posts	30
Figure 5.6.2.1	Add Post	31
Figure 5.6.2.2	Post Modify	31
Figure 5.8.1	Admin Dashboard	32
Figure 5.8.3	Active Users	33
Figure 5.9	Messages	34
Figure 5.10	View Profile	34
Figure 5.11	Update Profile	35

# CHAPTER 1

## INTRODUCTION & PURPOSES OF THE PROJECT

---

### 1.1 Introduction

Rental house management has become an important factor in modern society, hence the need to have a rental house management system. If you are a stranger in some city and want to rent a house, then it is difficult to find a suitable house in time. This is the main motivation behind the Online House Rental Management System project development. In our project, we have three stakeholders (Admin who controls everything, Owner who owns a property, and a renter). The website has three different panels for them. If any issue happens with the website, so a user can contact the admin using the Contact us form. And admin will get all the users' messages in the admin panel. Admin has some more power like to delete a profile in case if something goes wrong. The renters have a renter panel where a renter can browse all the posts, can bookmark a rent post if the renter likes, and can communicate with the owners. Also, this website has a house owner panel where an owner can post their property details. The owner can update and delete a post if he wishes.

Tech Stacks we have used in our project:

Frontend: HTML5, CSS3, Bootstrap, JavaScript, jQuery, PHP

Full Stack Framework: CodeIgniter 4

Database: MySQL

## **1.2 Motivation**

According to estimates from the United Nations, the global population was approximately 7.9 billion as of mid-2021. That's not the issue, the issue is that not everybody owns a house. They may have their own reasons, but everyone needs a shade to live under the sun. Some people have their own home, but they need to live outside of their town for their reasons. The reason can be, they are job holders, and their office is far away from home, for a student whose school, college, university is far from home, sometime in different country, some people want to go for a vacation etc. Now there are some people also who has a huge space to share and wants to earn some money by renting their properties. But finding host isn't an easy task without internet. Without internet it's very hard to find a home in demand in short period of time. It's not impossible to find home on demand but it's matter of long time and extra pain for health and mind. House rental service on internet can be useful here. In house renting website, host who wants to rent, sell their property adds their property, and the renters can find a shelter for them as they are looking.

## **1.3 Objectives of the project**

- i To analyze the challenges facing management of rental properties.
- ii To provide a friendly system for landlords to list their available properties.
- iii Tenants can search and find rental houses that meet their criteria.
- iv The system also aims to improve communication and organization between landlords and tenants.

## 1.4 Statement of the Problem

Metro city is a very wide city, therefore, to rent or finding a house in this city is taken as a problem. The other problem is increasing the number of people come to City from day to day for different reasons and they want to rent a house. Then they face many problems to get a rental house.

Generally, the main and observed problems are:

- **Manpower wastage:** Renters lose their energy by finding house on foot from place to place even they cannot get house because most of houses are reserved, and they cannot know where they can get free house exactly.
- **Wastage of money:** Renters waste their money for coordinators payment and for taxi to go to the office.
- **Coordinators or brokers false information:** Coordinators give uncertain information for renter when the renters will find the house with coordinators since the renter did not get the house as he/she expect.
- **Data Redundancy:** There is redundancy in renting new houses. Coordinators or brokers may take payment from different peoples, and they will give the house for the one who gives more money for them this leads to conflict among them.

## 1.5 Advantage of the System

1. This website will help the user to register individual home or apartment to asset you in finding the perfect rental home or property.
2. Our website is to help the students who get admitted in colleges far away from their homes to find the best environment for their studies by just looking in their phones.
3. **Cost Saving:** This project will save the students from paying too much rent. They can easily compare prices and choose the best accommodation.
4. **Time Saving:** The students can easily search available rooms on their phones and saving their precious time.
5. **No Travelling:** There is no need to travel and visit different locations in search of rental rooms. This will be a one-minute job.

# CHAPTER 2

## BACKGROUND INFORMATION AND DEVELOPMENT PROCESS

### **2.1 Background**

Dhaka is one of the beautiful city and capital of Bangladesh. The city is found in the Middle parts of Bangladesh. Which volume is 306.4 square kilo meter. Currently Dhaka's 2023 population is now estimated at 31,233,652. Dhaka has grown by 606,802 in the last year. Since the city is wide it is harder to find houses for renting in such wide city. In Dhaka city the population growth is increasing day to day that means different branch offices are opened. All company's needs employees and staff to work in their company and these employees need house to serve their life. There are many universities in Dhaka City. As a student that study far away from home and don't get hostel, we must find a nearby house to rent. House rental that are nearby to the colleges, and University are usually become student choice. The student who wants to rent a house will hardly find a suitable with them. The owner usually not advertises their house as they just spread the house to the resident mouth to mouth the student must ask the resident around to find the available rent house and it will take so much time. The purpose of this system is to help the student to find rent house easily. So, we have started to develop new web-based application of house renting relative with current problem of the city we have started by studying the detail of the system. This system will show the available house to rent and make comparison to help the student in making their choice.

### **2.2 Process of Development**

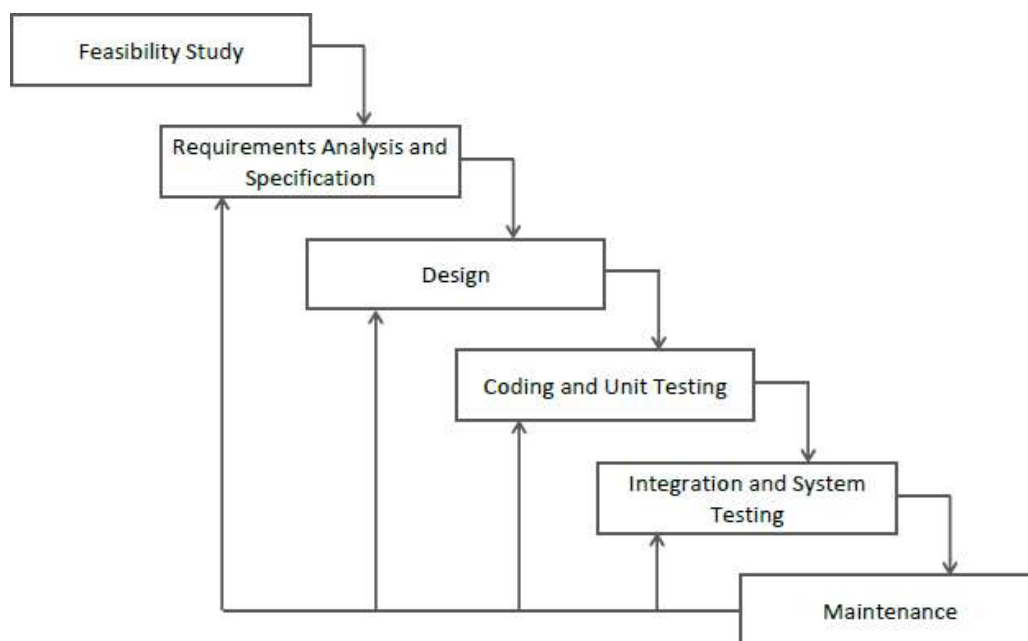
The term methodology means the technique and procedure of the system adopted for conducting a research study. It shows how data will be collected and the tools for collecting data, system methodology, the proposed system input and output, users, and systems development tools. System development is a technique that is used to show how the proposed system will be developed. In this case, the methodology used will be a waterfall model. It is included in the stages that the developer will use when developing the system. It is a sequential model, and the name is waterfall. The developer must finish with one stage

before going to the next stage. It contains the feasibility study, analysis phase, design phase, coding phase, testing phase, implementation phase and finally the maintenance phase.

It is a simple model and easy to use and understand to make a system. With waterfall development-based methodologies, the analysts and users proceed to work sequentially from one phase to the next phase. The deliverables from each phase are voluminous and are presented to the project sponsor for approval as the project moves from phase to phase. Once the client has approved the phase, the next phase begins with database design. It is at this stage that we will identify any faults before moving to the next stage. The output of this stage is the design specification which is used in the next stage of implementation.

### 2.2.1 Waterfall Model

Figure 2.2.1 shows the software development cycle. Basically, you design the system, code it and test it. When an error is found, it must be determined whether the error is a design error or not. Coding errors can quickly be fixed, but design errors may take longer.



**Figure 2.2.1:** Block Diagram of Waterfall Model

## 2.3 Relational Database Management System

Relational database is a type of database that organizes data into one or more tables (or "relations") of rows and columns, with a unique identifier for each row. The tables are connected by relationships, which are defined by foreign keys, allowing data in different tables to be related to one another. The most popular type of relational database management system is the SQL database. Examples of SQL databases include MySQL, SQL Server, and PostgreSQL.

MySQL is a popular open source relational database management system (RDBMS) choice for web-based applications. Developers, database administrators and DevOps teams use MySQL to build and manage next-generation web- and cloud-based applications.

In a relational database, the relationship between a house owner and renters would likely be represented by three separate tables: one for house owners and one for renters and one table for property information. The house owner table would likely have columns for information about the house owner such as name, contact information, and ownership details. The renter table would have columns for information about the renters, such as name, contact information, rental history, and the rental property they are currently living in. To establish a relationship between these three tables, we can add a foreign key column in the renter and property information table that references the primary key of the house owner table. This foreign key column would contain the unique identifier of the house owner associated with each renter, effectively linking the three tables, and allowing us to easily identify the house owner for a given renter or the renters for a given house owner. This is a simple example the Database will be much more complex with multiple tables and relations.



# CHAPTER 3

## SOFTWARE REQUIREMENTS AND SPECIFICATIONS

---

Software requirements and specifications are documents that describe the functionality, performance, and design constraints of a software system. They define what the software should do, as well as any constraints or limitations on its operation.

The process of defining software requirements typically involves identifying the stakeholders of the system (such as users, developers, and management), and gathering their input on what the software should do and how it should behave. This information is then used to create a set of requirements that the software must meet.

There are several different types of requirements that are typically included in software requirements and specifications documents, including:

- **Functional Requirements:** These describe the specific functions and features that the software must provide.
- **Performance Requirements:** These describe the specific performance characteristics that the software must meet, such as response times and memory usage.
- **Design Constraints:** These describe any constraints on the design of the software such as platform-specific requirements or limitations on the use of certain technologies.
- **User Interface Requirements:** This describes the User interaction, layout, design and user experience.
- **Usability Requirements:** This describes the accessibility and ease of use of the software for different users and groups.
- **Security Requirements:** These describe the security measures that must be in place to protect the software and its users from unauthorized access or other security threats.
- **Acceptance Criteria:** These are a set of conditions that must be met for the software to be considered complete and ready for release.

Once the requirements have been defined, they are used as the basis for designing, building, and testing the software. As the software is developed, it is tested against the requirements to ensure that it meets them, and any issues or bugs are identified and resolved.

It's worth noting that the software requirements and specifications can change throughout the development process, due to feedback from stakeholders, or the emergence of new requirements. The process of managing and updating these requirements is known as requirements management.

Overall, software requirements and specifications play a critical role in the development of any software system, providing a clear and detailed understating of what the software should do, and helping to ensure that it meets the needs of its users.

### **3.1 Software Requirements**

Software requirements are a set of statements that describe what a software system must do to meet the needs of its users and stakeholders. They are typically gathered through a process of requirements elicitation, which involves identifying the stakeholders of the system and gathering their input on what the software should do and how it should behave.

There are several different types of software requirements, including:

- **Functional Requirements:** These describe the specific functions and features that the software must provide. They detail the system's capabilities, what it should do and what should be possible for the users to accomplish with it.
- **Non-Functional Requirements:** These describe the broader characteristics of the software, such as performance, security, usability, and maintainability.
- **Business Requirements:** These describe the specific needs of the business or organization that the software is being developed for.
- **User Requirements:** These describe the specific needs and expectations of the end users of the software.

It's important to note that requirements are often documented in natural language, and should be clear, complete, and testable. A good requirement should be:

- Specific
- Measurable
- Achievable
- Relevant
- Time-bound

Once the requirements have been gathered, they are analyzed, refined, and organized to create a set of software requirements and specifications that the software must meet. These requirements serve as the basis for designing, building, and testing the software.

It's also worth noting that requirements can change during the development process due to changes in business or user needs, or the emergence of new technologies. The process of managing and updating these requirements is known as requirements management.

Overall, software requirements play a critical role in the development of any software system by providing a clear and detailed understanding of what the software should do and helping to ensure that it meets the needs of its users.

### **3.1.1 PHP**

PHP (Hypertext Preprocessor) is a popular open-source programming language that is commonly used for web development. It is especially well-suited for server-side scripting and is used to build a wide variety of web-based systems, from simple websites to complex web applications.

PHP code is typically embedded within HTML code and is executed on the server side before the page is sent to the user's web browser. This allows developers to create dynamic web pages that can change based on user input or other conditions. PHP supports a wide range of data types, including numbers, strings, arrays, and objects. It also has several built-in functions for common tasks, such as connecting to a database, manipulating strings, and performing mathematical operations. In addition to its built-in functions, PHP also has many libraries and frameworks that can be used to extend its functionality and simplify development.

These include popular frameworks like Laravel and CodeIgniter, which provide a structure for building web applications and include many useful features out of the box. PHP can be run on a variety of platforms, including Linux, Windows, and macOS. It is typically used in conjunction with a web server such as Apache or Nginx and a database management system such as MySQL or PostgreSQL.

PHP have been around for a long time, since 1995 and it's evolved to version 8 as of 2021 and its often used with other technologies such as HTML, CSS, and JavaScript to create dynamic, interactive web pages and web application, also it's very popular for creating CMS such as WordPress, Drupal, Joomla, and many more. Overall, PHP is a versatile, powerful, and widely used programming language that makes it easy to create dynamic web pages and web-based applications.

### **3.1.2 CodeIgniter 4**

CodeIgniter is a popular open-source web application framework for building dynamic web sites using the PHP programming language. CodeIgniter 4 is the latest version of the framework, which was released in 2020.

One of the main advantages of CodeIgniter 4 is its lightweight and minimalistic design, which makes it easy to learn and use for developers of all skill levels. It has a simple, elegant, and well-organized structure, which makes it easy to organize and maintain your code.

CodeIgniter 4 has several improvements and new features over the previous version, including:

- Namespaces and PSR-4 autoloading support
- Support for PHP 7.2 and above
- Improved security features and encryption
- A new built-in caching library
- Better routing and URL management
- Improved support for modes, views, and controllers
- Support for more database engines

The framework provides a range of libraries and helper classes for common tasks such as working with databases, handling form inputs, sending email, and more. Additionally, it is easy to extend the framework to add custom functionality, and there are many third-party libraries available that can be used to add additional features and capabilities to your application. CodeIgniter 4 also has a backward compatibility with version 3 and it's kept as much as possible so you can upgrade your application with minimal changes. CodeIgniter 4 is also supported by a large and active community, which makes it easy to find help and support when needed. There are many resources available online including tutorials, documentation, and forums, where you can learn more about the framework and get help with specific problems. Overall, CodeIgniter 4 is a powerful, flexible, and lightweight web application framework that makes it easy to build dynamic web applications with PHP. It has a simple and intuitive structure, making it an easy to use and efficient choice for developers.

### **3.1.3 JavaScript**

JavaScript is a programming language that is commonly used to add interactivity and other dynamic features to websites. It is primarily used on the client side (i.e., in web browsers), although it can also be run on the server side using platforms like Node.js.

JavaScript is a versatile and powerful language that can be used for a wide variety of tasks, including:

- Creating interactive forms and user interfaces
- Validating user input
- Animating elements on a web page
- Creating complex interactive user interfaces and data visualizations
- Building web and mobile applications

JavaScript is a high-level, interpreted programming language, which means that it is easy to learn and use, and its code can be written and executed directly without the need for a compiler.

JavaScript code is typically written within HTML and CSS code and is executed by the user's web browser. JavaScript can manipulate HTML and CSS, and can also make asynchronous requests to the server, allowing for a more dynamic and responsive user experience.

JavaScript can also be used in combination with other languages and technologies to build complex web applications. For example, it is commonly used in conjunction with CSS and HTML to create interactive user interfaces, and with frameworks like AngularJS, ReactJS, and Vue.js for building complex web and mobile applications.

It is also worth noting that JavaScript has a standardized version called ECMAScript (ECMAScript 6 and higher called ES6) which defined new features and improve the language and make it more consistent across different environments, and constantly evolving.

JavaScript have a large and active community with many resources and libraries to help developers in their work, some of the well-known libraries are jQuery, Lodash, and Moment.js and many more.

Overall, JavaScript is a powerful, versatile, and widely used programming language that is essential for creating interactive and dynamic web pages and web applications. Its popularity has been growing fast and it's considered one of the most important programming languages to learn.

### **3.1.4 MySQL**

MySQL is a popular open-source relational database management system (RDBMS). It is widely used for web and mobile applications, as well as for data warehousing and logging applications. MySQL is known for its reliability, robustness, and ease of use, making it a popular choice for many businesses and organizations.

MySQL stores data in tables, which consist of rows and columns. Each row represents a single record, and each column represents a field in that record. MySQL supports a wide range of data types, including numbers, strings, and dates. You can use SQL (Structured Query Language) to query and manipulate the data in a MySQL database. SQL is a powerful, yet simple, language that is used to perform tasks such as inserting, updating, and retrieving data from a database.

MySQL also offers several advanced features, such as support for stored procedures and triggers, which allow you to create complex business logic within the database itself. It also supports for replication and clustering, to improve performance and availability, which is critical for mission critical applications.

There are multiple versions of MySQL, and it's also possible to get the software from multiple vendor (Oracle, MariaDB, Percona Server, etc.) with different set of features.

### 3.1.5 jQuery

jQuery is a JavaScript library that makes it easier to work with HTML documents, handle events, create animations, and perform other common JavaScript tasks. It was created by John Resig in 2006 with the goal of simplifying the process of writing JavaScript code.

One of the key features of jQuery is its use of a chainable syntax, which makes it easy to apply multiple actions to a single selection of elements. For example, the following code selects all the elements on a page with the class "example" and sets their background color to red:

```
$(".example").css("background-color", "red"); 
```

jQuery also provides many built-in functions for common tasks such as traversing the DOM, manipulating elements, and handling events. Some examples include:

- `$(selector).click(function)`: Attaches a click event to the elements selected by the given selector
- `$(selector).hide()`: Hides the elements selected by the given selector
- `$(selector).val()`: Gets or sets the value of the elements selected by the given selector (e.g. for form inputs)
- `$(selector).addClass(className)`: Adds the given class to the elements selected by the given selector

jQuery also has an extensive plugin ecosystem, which allows developers to easily add new functionality to their projects. Some examples include carousels, modals, and image sliders.

It is important to note that while jQuery is widely used, the vanilla JavaScript has improved, now providing more functionality than before. Thus, many developers are leveraging these new features and often find they don't need to use jQuery on their projects.

### 3.1.6 HTML

HTML (Hypertext Markup Language) is the standard language used to create web pages. It consists of a series of elements, represented by tags that define the structure and content of the page.

HTML documents are made up of two main parts: the head and the body. The head contains Meta information about the document, such as the title of the page and links to external resources like CSS stylesheets and JavaScript files. The body contains the content of the page, such as text, images, and links.

HTML elements can also have attributes, which provide additional information about the element. For example, the `<a>` element, which creates a hyperlink, has a `href` attribute that specifies the URL of the linked page:

```
<a href="https://www.example.com"> Visit Example </a>
```

HTML has evolved a lot since it was first proposed in 1989. Now HTML5 includes new elements and attributes to handle new media, such as video and audio, and to provide new features, such as form validation and the ability to work offline.

### 3.1.7 CSS

CSS (Cascading Style Sheets) is a stylesheets language used for describing the presentation of a document written in a markup language. While HTML is used to structure the content of a web page, CSS is used to control the layout, colors, fonts, and other visual aspects of the page.

CSS styles are defined in stylesheets, which are separate files that can be linked to a web page using the `<link>` element in the HTML `<head>`. Styles can also be defined directly in the HTML document using the `<style>` element or included in the `style` attribute of individual elements.



CSS selectors are used to select the elements on a web page that a particular rule should be applied to. In the example above, the p selector selects all <p> elements. Other types of selectors include:

- **Class selectors:** select elements based on the value of their class attribute. For example, .my-class would select all elements with the class "my-class".
- **ID selectors:** select elements based on the value of their id attribute. For example, #my-id would select the element with the ID "my-id".
- **Attribute selectors:** select elements based on the presence or value of a particular attribute. For example, a[href='https://www.example.com'] would select all <a> elements with a href attribute set to "https://www.example.com".

CSS also provides a wide range of properties that can be used to control the layout, colors, fonts, and other visual aspects of a web page. Some examples include:

- **display:** controls how an element is displayed on the page (e.g., as a block or inline element)
- **position:** controls the positioning of an element on the page (e.g., as a relative or absolute element)
- **width and height:** control the size of an element
- **color and background-color:** controls the text and background colors of an element
- **font-size and font-family:** controls the size and typeface of text

CSS has also evolved, with new specifications and features like CSS2, CSS3. Now, it also allows more advanced layout, animation, and responsive design.

CSS is an essential building block of modern web development, it gives developers a powerful way to create a consistent look and feel across a website and to adapt it to different screen sizes, devices, and user contexts.

## 3.2 Functional Requirements

Functional requirements are the specific features and capabilities that a system or product must have to meet the needs of its users. They describe what the system is supposed to do and how it should do it. These requirements are usually specified by stakeholders such as the product owner, users, or customer and they are typically used as the basis for design and development decisions.

Functional requirements can be further categorized into several types, including:

- Functional requirements can be further categorized into several types, including Performance requirements: These define the performance characteristics of the system, such as response time, throughput, and reliability.
- Data requirements: These define the types of data that the system must store, manage, and process.
- Functional capabilities: These define the specific functionality that the system must provide, such as data entry, search, and reporting.
- Security requirements: These define the measures that the system must take to ensure the confidentiality, integrity, and availability of data.
- User interface requirements: These define the layout, structure, and design of the user interface, such as buttons, menus, and forms.
- Compliance requirements: These define any regulatory or compliance requirements that the system must meet, such as accessibility, data protection, and security standards.

When writing functional requirements, it is important to use clear and concise language, and to be as specific as possible. This will make it easier to understand the requirements and to implement them correctly. Additionally, it is important to work closely with the stakeholders, including users and customers, to ensure that the requirements accurately reflect their needs and that they are properly prioritized and scheduled for development.

Functional requirements are a critical part of system development, as they provide a clear understanding of the system's capabilities and constraints, and they provide a basis for testing and validation of the system. The requirements must be clearly understood and implemented to create a successful system.

### **3.3 Non-Functional Requirements**

Non-functional requirements are requirements that specify how well a system should perform in terms of its quality attributes, such as performance, security, usability, and maintainability. These requirements describe the system's characteristics and constraints, but they do not describe specific functions or behaviors.

Here are some examples of non-functional requirements:

- Performance requirements: These define the system's performance characteristics, such as response time, throughput, and scalability. For example, a website may have a requirement for a response time of less than 2 seconds for 90% of all requests.
- Security requirements: These define the system's security characteristics, such as confidentiality, integrity, and availability. For example, a payment system may have a requirement for end-to-end encryption of all sensitive data.
- Usability requirements: These define the system's usability characteristics, such as ease of use, learnability, and accessibility. For example, an e-commerce website may have a requirement that it be usable by people with disabilities.
- Maintainability requirements: These define the system's maintainability characteristics, such as ease of modification, testing, and troubleshooting. For example, a software application may have a requirement for automated testing to facilitate the change process.
- Scalability requirements: These define the system's ability to handle an increasing amount of work or users, such as number of concurrent users, data volume or geographic distribution.
- Compatibility requirements: These define the compatibility of the system with other systems or devices.

Non-functional requirements are often more difficult to specify than functional requirements, as they often depend on the specific context and use case of the system. Additionally, non-functional requirements may conflict with one another, and trade-offs may need to be made.

It's important to clearly define and prioritize non-functional requirements, and to validate that they are met during the testing and deployment process. Failing to consider these requirements can lead to a system that is difficult to use, unreliable, or difficult to maintain. Non-functional requirements play a critical role in the success of a project because they describe how well the system must perform, how easy it is to use, and how easy it is to maintain. They provide a basis for testing and validation, and for determining the system's overall quality.

# CHAPTER 4

## SOFTWARE DESIGN

---

Software design is the process of defining the architecture, components, interfaces, and other characteristics of a software system. It is the process of creating a plan or blueprint that describes how the software will meet the needs of its users, and how the different components of the software will work together. The goal of software design is to produce a representation of the software that can be used to guide its construction, testing, and maintenance.

There are many different approaches to software design, each with their own strengths and weaknesses. Some of the more popular approaches include:

- Object-oriented design, which focuses on the development of reusable objects and classes, and the interactions between them.
- Component-based design, which focuses on the development of reusable software components that can be easily combined and reused in different systems.
- Service-oriented design, which focuses on the development of loosely, coupled services that can be easily integrated with other systems.
- Event-driven design, which focuses on the development of systems that respond to events, such as user inputs or system changes.
- Agile design, which emphasizes on adaptability, rapid prototyping, and iteration.

The specific approach and techniques used will depend on the requirements of the software and the goals of the project.

### 4.1 DFD

A Data Flow Diagram (DFD) is a graphical representation of the flow of data through a system. It is used to model the flow of information in a process or system, including the inputs, processes, and outputs. The goal of a DFD is to show the flow of data through a system, from external entities to internal processes and data stores, and out to external entities again.

DFD's are used in many different fields, including software engineering, information systems, and business process modeling. They are particularly useful for modeling complex

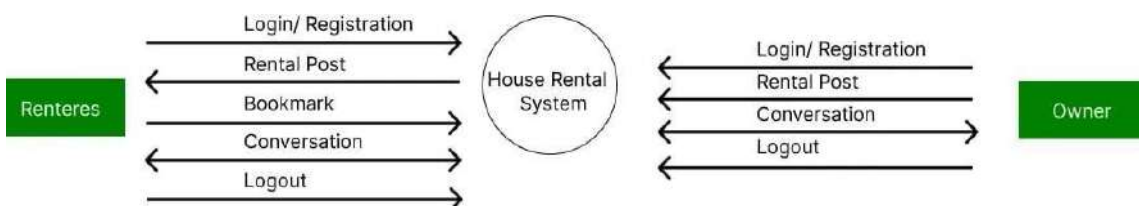
systems, as they provide a clear, visual representation of the flow of data and can help identify potential bottlenecks and areas of improvement.

DFD's are divided into two types:

- Context Diagram:
  - Context diagrams are also called Level 0 diagrams.
  - Context diagrams are also called Level 0 diagrams. It depicts the system boundary by showing the inputs and outputs that the system receives and sends, respectively.
  - It is the highest-level DFD and provides an overall view of the entire system.
- Level 1 DFD:
  - A level 1 DFD is a detailed view of the single process that was represented in the context diagram.
  - Level 1 DFD breaks down the process into smaller sub-processes.
  - Level 1 DFD provides a detailed view of what the system does and how it does it.

DFD's use a standardized notation that includes circles (representing data sources or destinations), rectangles (representing processes), and arrows (representing data flows). The notation is used to show how data flows through the system and how it is transformed by the processes that it passes through.

**Figure 4.1** shows '0' Level Diagram of Rental house systems represent a basic overview of whole system or process being analyzed or modeled.



**Figure 4.1:** DFD Diagram

## 4.2 ERD

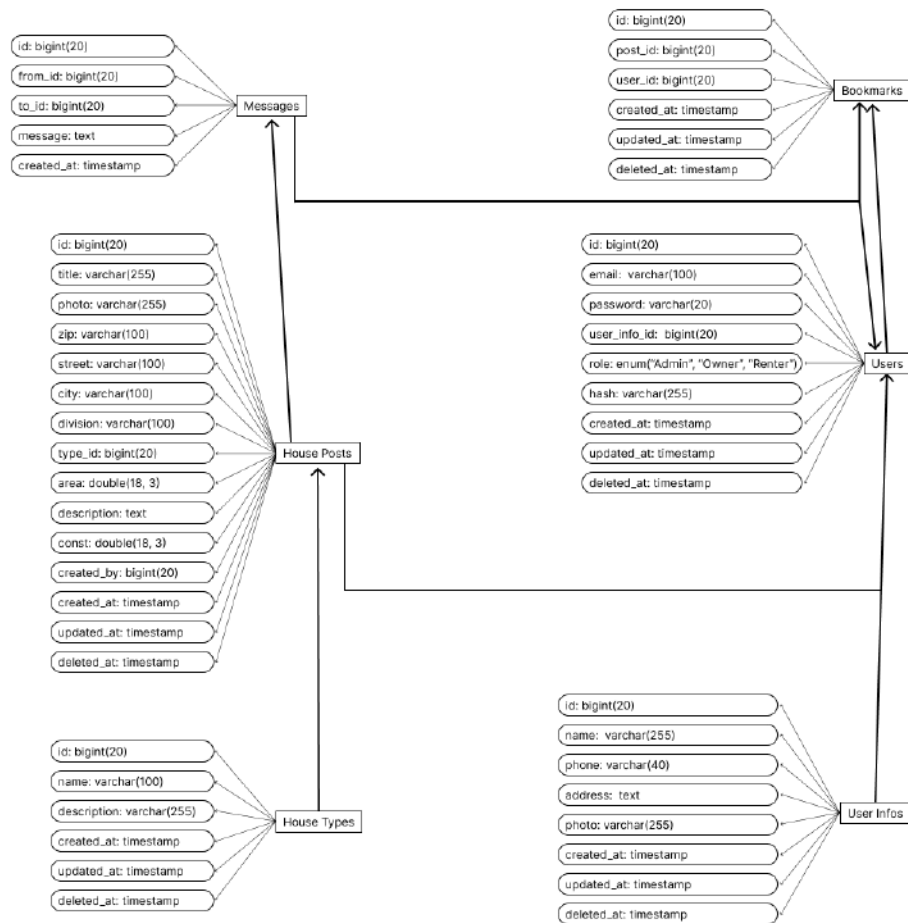
An ERD, or Entity Relationship Diagram, is a graphical representation of entities and their relationships to each other. It is used in the field of software engineering and database design to represent the structure of a database. The goal of an ERD is to show the relationships between entities, including how data is stored and retrieved, and to provide a visual representation of the data model.

ERDs use a standardized notation that includes three main components: entities, attributes, and relationships.

- Entities
  - Represented by rectangles in the ERD.
  - It represents the object or concept in the real world.
  - Examples: Customers, Orders, Products etc.
  
- Attributes:
  - Represented by ovals in the ERD.
  - It describes the characteristics of an entity.
  - Examples: customer name, order date, product price.
  
- Relationships:
  - Represented by diamond shapes in the ERD.
  - It defines how two or more entities are related to each other.
  - Examples: A customer makes many orders; A product belongs to a specific category.

ERD's are useful in various fields such as software engineering and database design because they help to visually represent the structure of data and how it is organized, making it easier to understand, design and develop a database. They also can be useful for system analysis and for documenting existing systems.

**Figure 4.2** Shows ER Diagram of Online house Rental System. An Entity Relationship Diagram (ERD) shows how entities (such as renter, owner, objects, or concepts etc.) relate to each other in a particular system.



\*

**Figure 4.2:** ER Diagram

### 4.3 Use Case Diagram

A Use Case Diagram is a type of behavioral diagram in the Unified Modeling Language (UML) that shows the relationships between actors and use cases within a system. The goal of a use case diagram is to show a high-level view of the functionality of a system, and to identify the relationships between the actors (people or systems) that interact with the system, and the use cases (the specific actions or functions that the system can perform).

A use case diagram typically includes the following components:

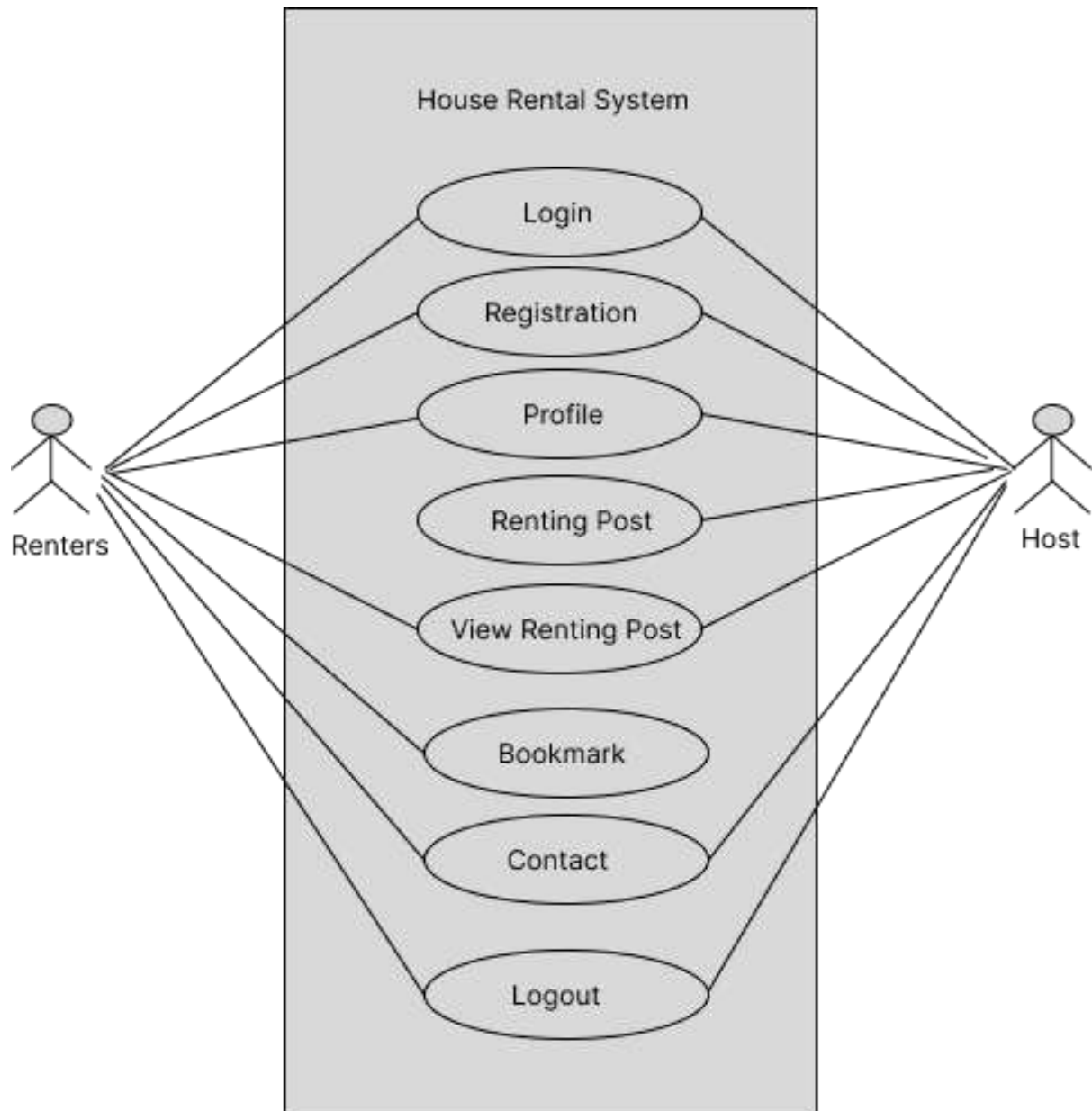
- **Actors:** Represented by stick figures in the diagram. They are external entities that interact with the system. Examples include a customer, an administrator, or another system.
- **Use cases:** Represented by oval shapes in the diagram. They are the actions or functions that the system can perform. Examples include placing an order, viewing a report, or resetting a password.
- **System boundary:** Represented by a rectangle that encloses the use cases and actors. It defines the scope of the system and separates it from the outside world.
- **Relationships:** Represented by lines connecting the actors and use cases. The lines indicate how the actors and use cases interact with each other. There are two types of relationships:
  - **Association:** A line connecting an actor to a use case, representing that the actor performs the use case.
  - **Include:** A dashed line connecting one use case to another, representing that the first use case includes the functionality of the second use case.

Use case diagrams are a useful tool for capturing the functional requirements of a system, and can be used during the planning, analysis, and design phases of a project. They can help identify the specific actions or functions that a system must perform, and the actors that will interact with the system. They can also provide a clear and concise view of the system's functionality, making it easier for developers, stakeholders, and users to understand and communicate about the system's requirements.



In this system, there are two actors where one is internal agent, and another is external agent. The internal agent is the system owner while the external agent is the tenant.

**Figure 4.3** shows how the actors integrate with the system-



**Figure 4.3:** Use case diagram

# CHAPTER 5

## SOFTWARE OUTCOME AND IMPLEMENTATION

---

Software outcome refers to the result or output produced by a software program or application, while implementation refers to the process of putting a software program or system into action. This includes the installation, configuration, and testing of the software, as well as any necessary training or support for users. The implementation process is critical to ensuring that the software can meet the needs and requirements of the users, and that it can be used effectively and efficiently. The outcome of a software program can vary depending on the purpose and design of the program and how well the implementation process is done.

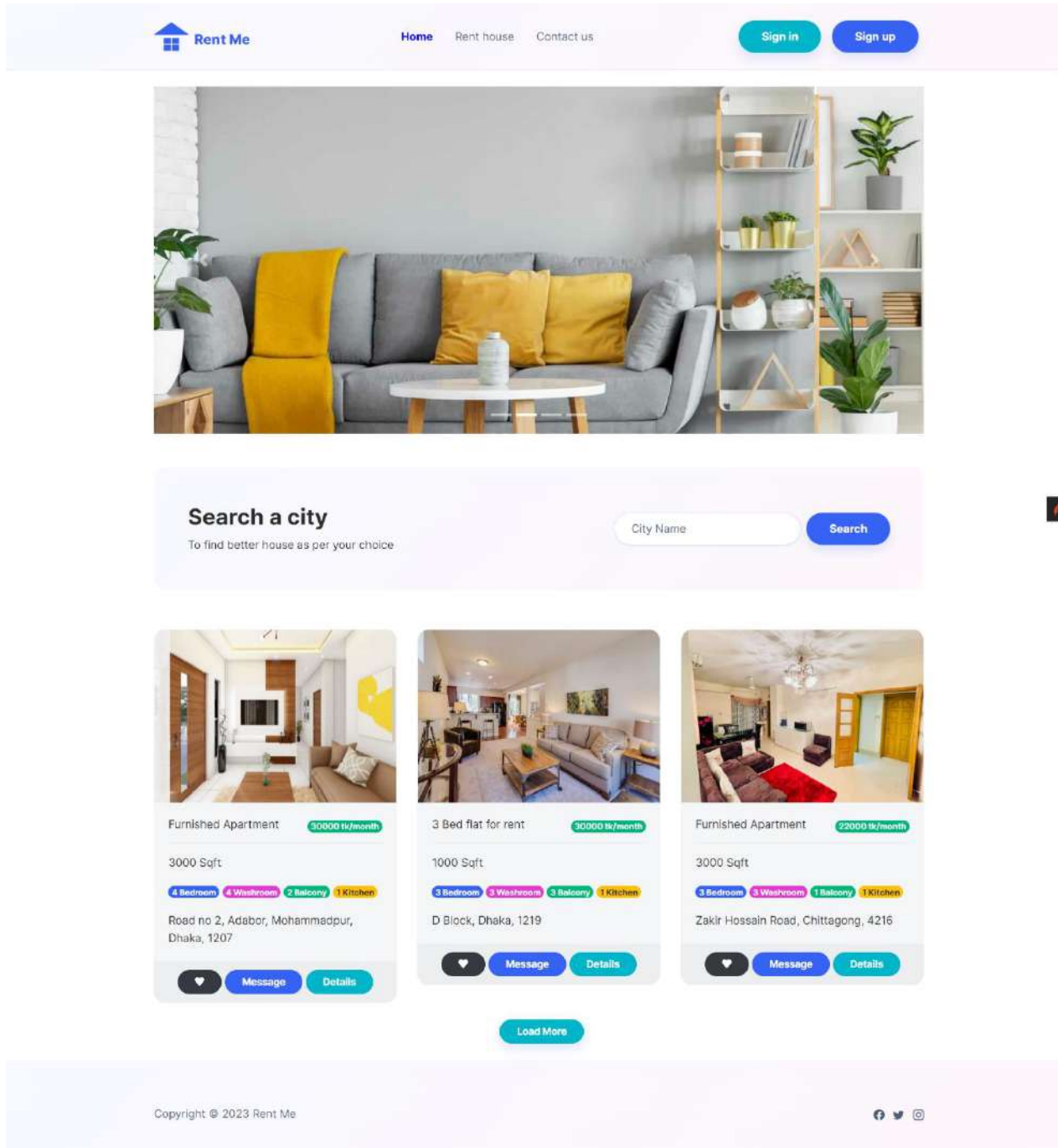
### 5.1 Home page

Homepage is often the first point of contact for users visiting a website or application, and it plays a crucial role in providing a positive user experience. The homepage should be designed to be easily navigable, visually appealing, and provide users with a clear understanding of the purpose and content of the website or application. Depending on the purpose of the website or application, the homepage may include the following elements:

- Navigation menu: Allows users to easily access different sections of the website or application.
- Branding: Displays the name and logo of the website or application.
- Calls to action: Encourages users to take specific actions, such as signing up, purchasing a product, or making a reservation.
- Featured content: Highlights important information or promotions.
- Search bar: Allows users to quickly search for specific information or content.
- Social media links: Connects users to the website or application's social media accounts.
- News or blog feed: Keeps users up to date with the latest news or updates.
- Contact information: Provides users with the means to contact the website or application's owner or support team.
- Login or Sign in button: Allows registered users to access their account.

The design and layout of the homepage should be consistent with the overall aesthetic of the website or application, and it should be optimized for all devices, including desktop and mobile. The homepage should be able to communicate the value proposition of the website or application and make it easy for the users to find what they are looking for.

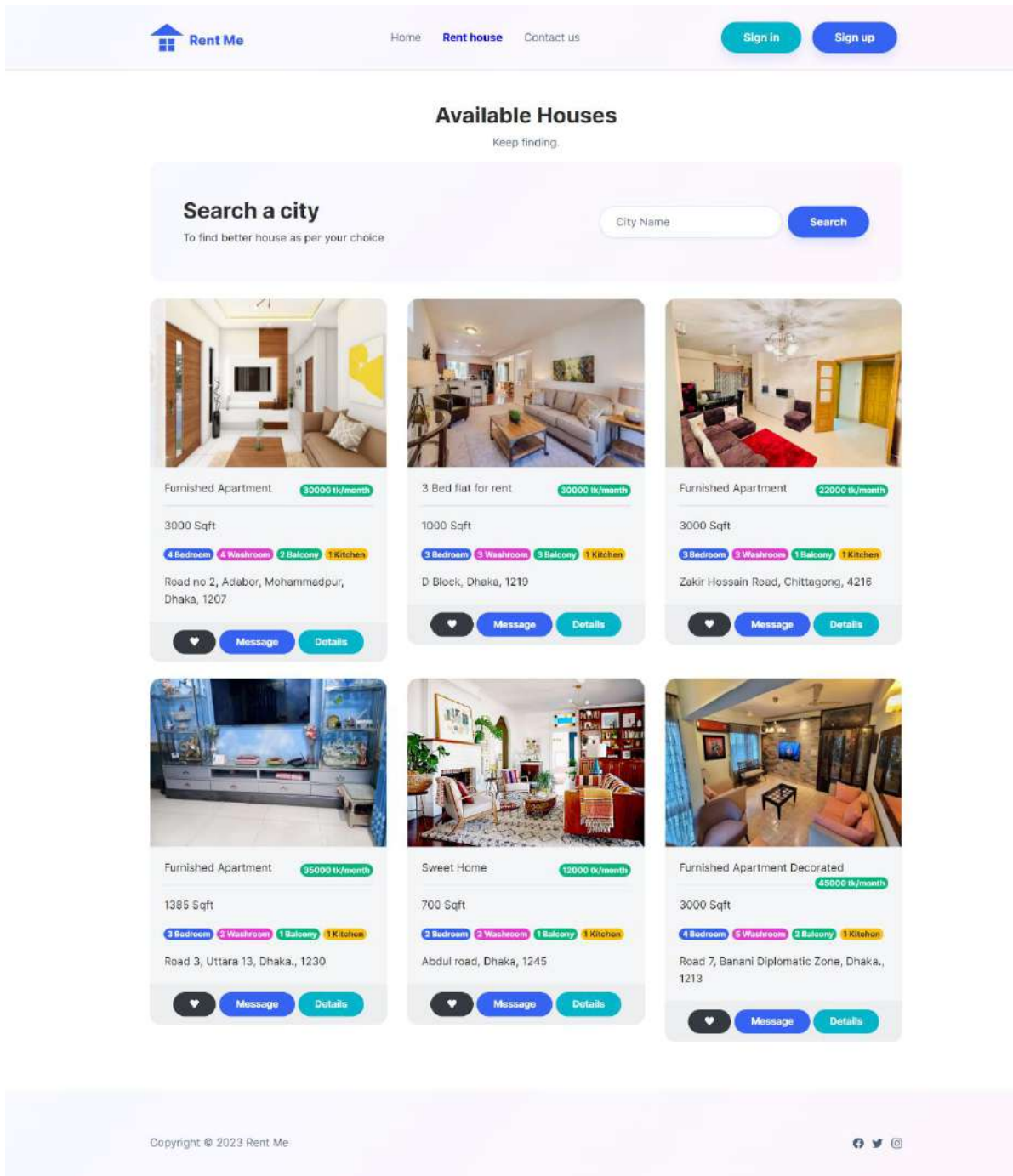
In **Figure 5.1**, shows the home page some Non-Functional requirements (Brand logo, Navigation bar, Slider) and functional requirements (Sign in, sign up, Search bar, Post card, Load more button). Here, Home page shows only three recent posts. If the user wants to see more post, then the user clicks on load more button to see more houses post.



**Figure 5.1:** Home Page

## 5.2 Rent House Page

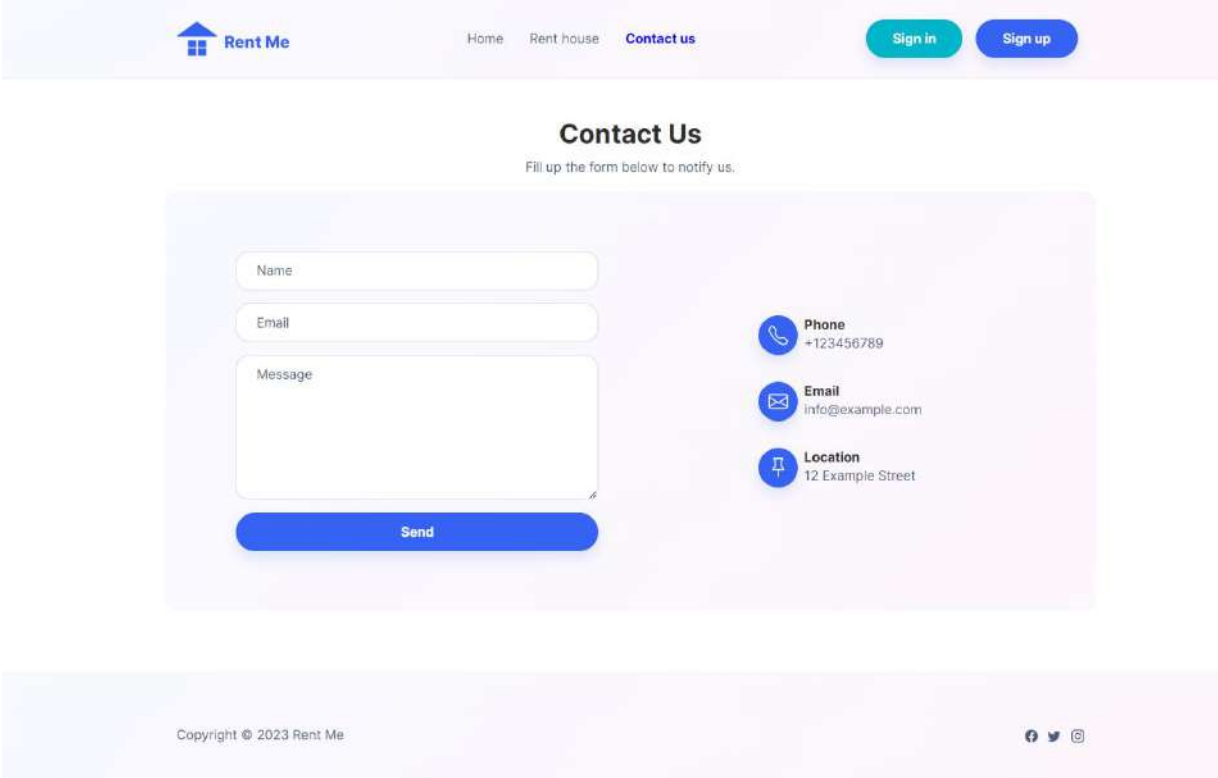
In **Figure 5.2** shows all available houses posted. Here the user sees all the recent posts and gets all details about houses. If the user (Renter) chooses any house, then they can click on the post for view and more details. If the renter wants to contact the house owner in this case, they can contact through the message to rent the houses. If the user wants to save any post to see the later, they can click on the bookmark button. The user can also filter location by searching a city name.



**Figure 5.2:** Rent house page.

## 5.3 Contact Us Page

In **Figure 5.3** shows the system management contact. If the user faces any technical problem in this case the user needs to contact system management IT. To contact the user at first, enter their name and Email then write text on message option and send to fix the problem. Users also contact with voice call.



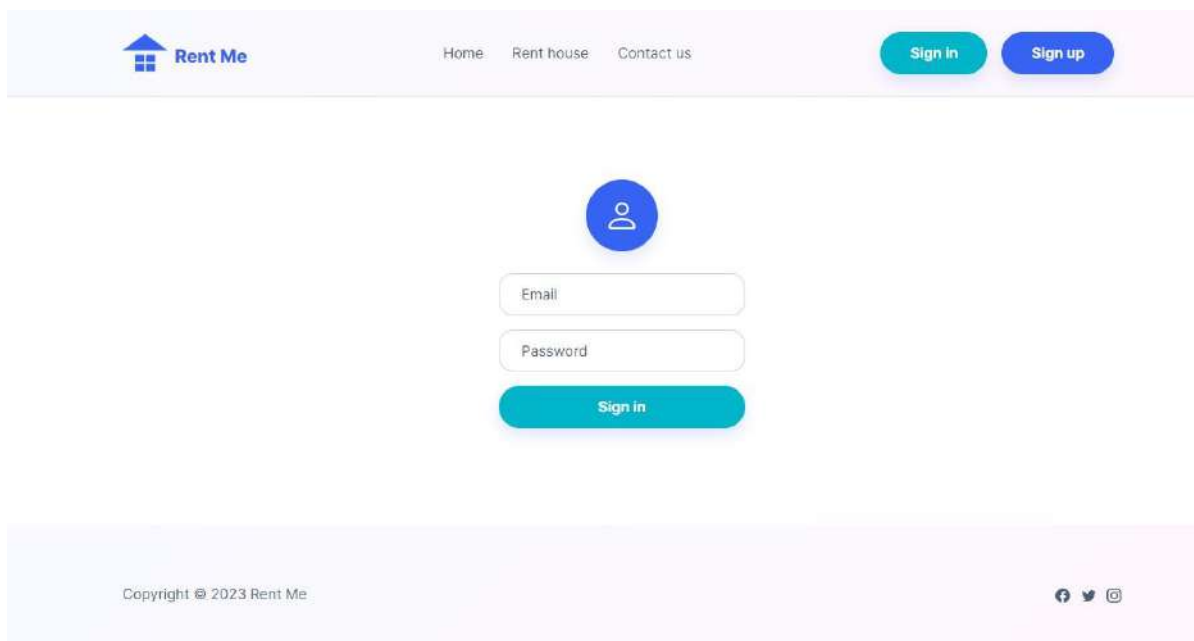
The screenshot displays the 'Contact Us' page for 'Rent Me'. The header includes the 'Rent Me' logo, navigation links for 'Home', 'Rent house', and 'Contact us', and buttons for 'Sign in' and 'Sign up'. The main content area features a contact form with fields for 'Name', 'Email', and 'Message', and a 'Send' button. To the right of the form, contact information is listed: 'Phone' (+123456789), 'Email' (info@example.com), and 'Location' (12 Example Street). The footer contains the copyright notice 'Copyright © 2023 Rent Me' and social media icons for Facebook, Twitter, and Instagram.

**Figure 5.3:** Contact Page

## 5.4 Sign in

Signing in typically refers to the process of entering an email and password to gain access to a website, application, or system. This allows the user to authenticate their identity and access the resources that they are authorized to use.

In **Figure 5.4** shows the login part. There are Multiples Functional Requirements (validation checking, and authentication checking) and Non-Functional Requirements. Here we have taken a form, where we have three fields (email, password, button) Here the email, and password is a required field, this field must be filled with valid email and password. After that, the user clicks on the sign in button, if the user enters valid email and password so the user can proceed to the next step. If not valid, in this case this user isn't an authenticated user. If the user doesn't enter the valid email or password, in this case the user will get an invalid authentication message. If the user forgets password, the user can click on the forget your password button to reset password.



**Figure 5.4:** Sign in Page

## 5.5 Sign up.

The process of signing up for a website or service typically involves providing some basic personal information and creating a username and password. The specific steps can vary depending on the website or service, but generally the process includes the following steps:

- Enter your personal information such as name, and email address.
- Create an email and password. It is important to use a strong and unique password for security.
- Click on the sign-up or register button to complete the process.

Once the user has completed the sign-up process, the user can proceed to the next step.

In **Figure 5.5** shows sign up process of the system. Here, the users sign up as a house owner or as a renter. Need Full name, unique Email, Password, Confirm password as information for sign in. Clicking on the **Sign-up** button and go to next steps. If the users already have an account in this case user click on the login button for sign in.

The screenshot displays the sign-up interface for the 'Rent Me' website. At the top, a navigation bar includes the 'Rent Me' logo, links for 'Home', 'Rent house', and 'Contact us', and buttons for 'Sign in' and 'Sign up'. The central form area features a user profile icon, a role selection section with radio buttons for 'House Owner' and 'Renter', and a form with fields for 'Your Full Name\*', 'Email\*', 'Password\*', and 'Confirm Password\*'. A blue 'Sign up' button is positioned below the form, and a link for 'Already have an account? Log In' is at the bottom. The footer contains the copyright notice 'Copyright © 2023 Rent Me' and social media icons for Facebook, Twitter, and Instagram.

**Figure 5.5:** Sign up Page.

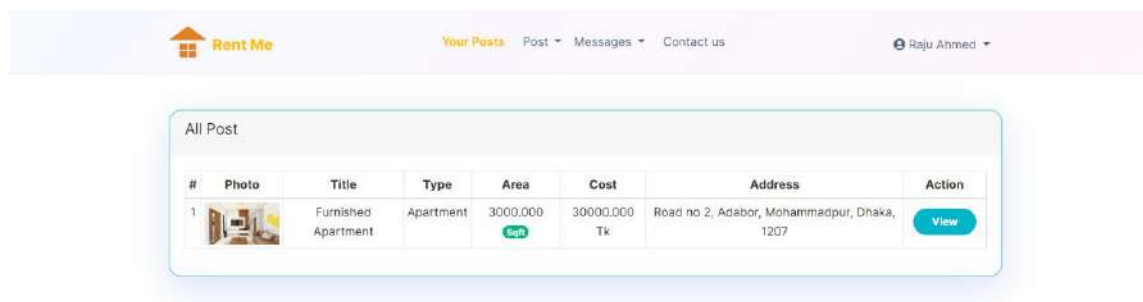
## 5.6 House Owner


House owner is a person who owns a house or other type of residential property. This can include single-family homes, townhouses, condominiums, and apartments. House owners are responsible for the maintenance, upkeep, and any repairs necessary for the property. They may live in the house themselves or rent it out to tenants. In Main host option the house owner can view and update the profile.

### 5.6.1 Your Posts

The specific owner who owns the owner account can see every post he had done before by clicking on the Post List.

**Figure 5.6.1** shows a user post list. Here, the only user (Owner) can see their own post and check how many posts they do by themselves.



#	Photo	Title	Type	Area	Cost	Address	Action
1		Furnished Apartment	Apartment	3000.000 Sqft	30000.000 Tk	Road no 2, Adabor, Mohammadpur, Dhaka, 1207	<a href="#">View</a>

**Figure 5.6.1:** Your Posts

## 5.6.2 Posts

The property owner can rent his property. For that the owner needs to make a post.

### 5.6.2.1 Add Post

To add a property, the owner needs to navigate to add a post Section, here the owner will add all the details about the property.



In **5.6.2.1 Figure** shows add post option to add new post with all details about house by user (owner) for house renting.

The screenshot shows the 'New House Post' form in the Rent Me application. The form is titled 'New House Post' and is located in the center of the page. It contains several input fields for user information and house details. The fields are: House Type (a dropdown menu with 'Select a House Type' as the placeholder), House Title, Cost, Area (sqr. ft.), Zip Code, Street, City, and Description. There is also a 'Choose File' button with 'No file chosen' next to it. To the right of the form, there is a section for 'Room Quantity Information' with input fields for Bed Rooms, WashRooms, Balcony, and Kitchen. At the bottom of the form, there are 'Reset' and 'Submit' buttons. The background of the form is light gray, and the overall page has a light blue header with the 'Rent Me' logo and navigation links.

**Figure 5.6.2.1: Add Post**

### 5.6.2.2 Post Modify

Modify means to edit a post that had been made before already. An owner can edit his past post if he wants to.

**Figure 5.6.2.2** shows post modification. If the user needs any post to edit or delete the owner can do it from the post modify option.

The screenshot shows the 'Edit House Post' form in the Rent Me application. The form is titled 'Edit House Post' and is located in the center of the page. It contains several input fields for user information and house details. The fields are: House Type (a dropdown menu with 'Apartment' as the selected value), House Title (with the value 'Furnished Apartment'), Cost (with the value '30000.000'), Area (sqr. ft.) (with the value '3000.000'), Zip Code (with the value '1207'), Street (with the value 'Road no 2'), City (with the value 'Adabor, Mohammadpur, Dhaka'), and Description. There is also a 'Choose File' button with 'No file chosen' next to it. To the right of the form, there is a section for 'Room Quantity Information' with input fields for Bed Rooms (with the value '4'), WashRooms (with the value '4'), Balcony (with the value '2'), and Kitchen (with the value '1'). At the bottom of the form, there are 'Reset' and 'Update' buttons. The background of the form is light gray, and the overall page has a light blue header with the 'Rent Me' logo and navigation links.

**Figure 5.6.2.2: Post Modify**

## 8.7 Renters

A house renter, also known as a tenant, is a person who rents a house or apartment from a landlord or property owner. Renters typically sign a lease agreement with the landlord, which outlines the terms and conditions of the rental, including the length of the rental period, the amount of rent due, and any rules or restrictions on the use of the property. Renters are responsible for paying rent on time and for maintaining the property in good condition and are usually required to follow any rules or regulations set by the landlord or property management company.

### 5.6.3 Bookmarks

Renters can bookmark or save any renting post by just clicking on the bookmark button (heart shape button). If renters bookmark any post, it will be shown in the Bookmarks tab. There a user can get all the bookmarks that have been saved before.

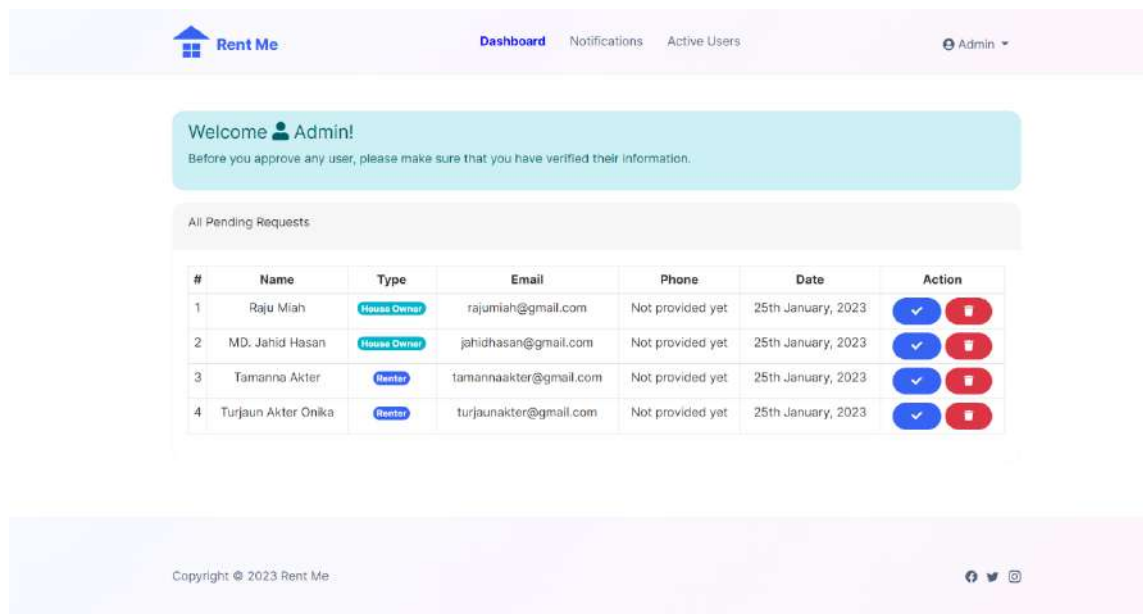
## 5.8 Admin Panel

The admin is a person whose works is to maintain every activity into the website. An admin can delete an account also if finds any violation from users.

### 5.8.1 Dashboard

In dashboard user has the full control to delete an account.

Figure 5.8.1 shows that an admin can approve or delete a user account.



The screenshot displays the Admin Dashboard for 'Rent Me'. The top navigation bar includes 'Dashboard', 'Notifications', 'Active Users', and a user profile 'Admin'. A welcome message for the Admin is shown, followed by a table titled 'All Pending Requests'. The table contains four rows of user requests, each with columns for ID, Name, Type, Email, Phone, Date, and Action. The Action column contains a blue checkmark icon and a red delete icon.









#	Name	Type	Email	Phone	Date	Action
1	Raju Miah	House Owner	rajumiah@gmail.com	Not provided yet	25th January, 2023	 
2	MD. Jahid Hasan	House Owner	jahidhasan@gmail.com	Not provided yet	25th January, 2023	 
3	Tamanno Akter	Renter	tamannaakter@gmail.com	Not provided yet	25th January, 2023	 
4	Turjaun Akter Onika	Renter	turjaunakter@gmail.com	Not provided yet	25th January, 2023	 

Figure 5.8.1: Admin Dashboard

## 5.8.2 Notifications

If any events occur, so the admin will get a notification about that event. An event can be anything like a user tried to contact with the admin for an issue that the user is facing etc.

## 5.8.3 Active Users

Admin can see that how many users are active now.

**Figure 5.8.3** shows that an admin can view all the active users.

All the active users.  
You can view their profile and contact them.

All Active Users

#	Name	Type	Email	Phone	Date
1	Admin	Admin	admin@rentme.com	017001122331	22nd January, 2023
2	Raju Ahmed	House Owner	host@rentme.com	01715562612	12th January, 2023
3	Raju Miah	Renter	raju@gmail.com	01739722222	14th January, 2023
4	Nilufa Jaman	Renter	nilufa@gmail.com	Not provided yet	22nd January, 2023
5	Imran Najir	Renter	imran@gmail.com	Not provided yet	25th January, 2023
6	Ahmed Raj	Renter	ahmed@gmail.com	Not provided yet	25th January, 2023

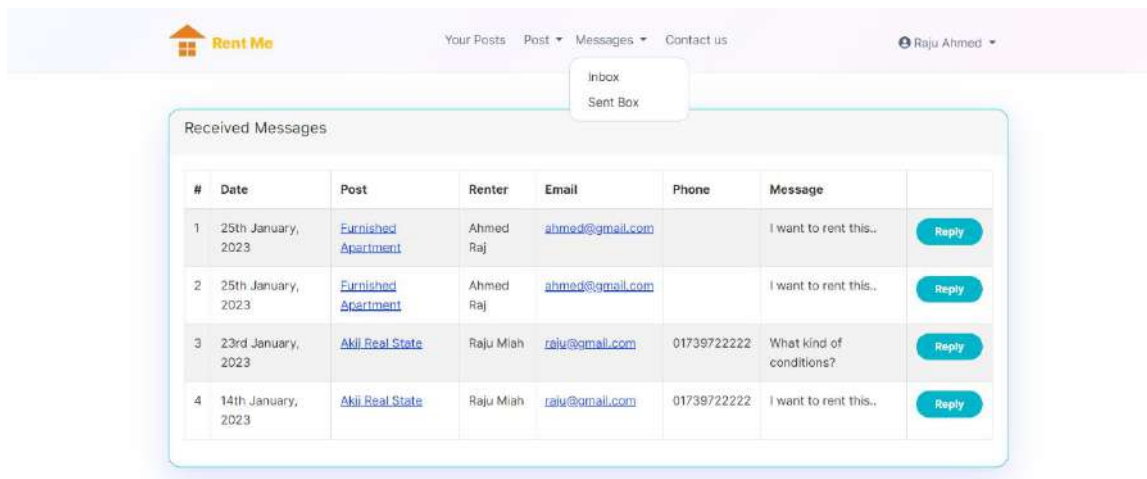
Copyright © 2023 Rent Me

**Figure 5.8.3:** Active Users | Admin Pane

## 5.9 Messages

Users (House owners and Renters) can communicate with each other using the bidirectional communication system. Here, on Inbox, user will get all the incoming message, and in Sent Box, user can view all the Outgoing messages.

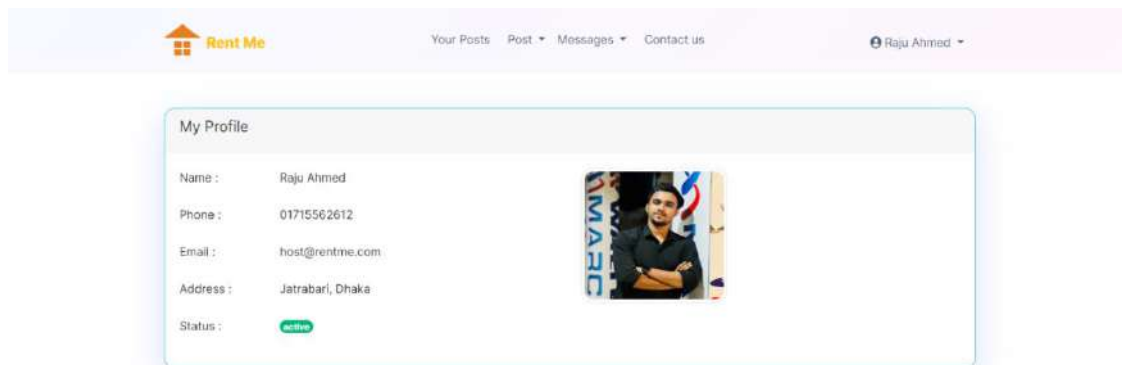
**Figure** shows a chat box. Here, the house owner and tenant contact each other for more house information and for rent.



**Figure 5.9: Messages**

## 5.10 View Profile

**Figure** shows the owner profile. Here the owner can view the own profile.

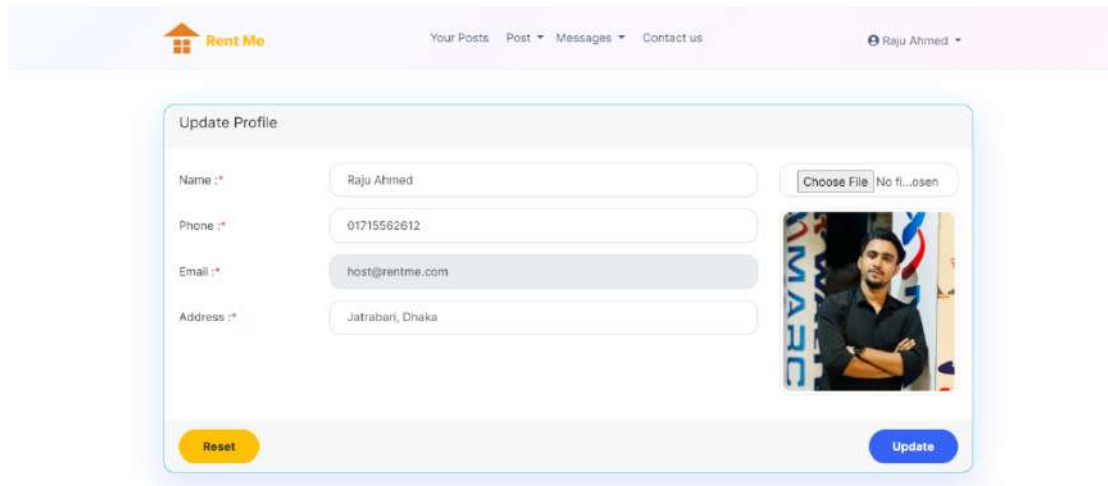


**Figure 5.10: View Profile**

## 5.11 Update Profile

If all the required field has been filled, the user can click on the update button. Or if user want to empty all the field so user need to click on the reset button.

In **Figure 5.11** Shows owner update profile. Here, If the owner needs to update their personal information, they can do it.



The screenshot shows the 'Update Profile' form within the Rent Me application. The form is titled 'Update Profile' and contains the following fields and elements:

- Name :** Input field containing 'Raju Ahmed'. To the right is a file upload button labeled 'Choose File' and 'No file chosen'.
- Phone :** Input field containing '01715562612'.
- Email :** Input field containing 'host@rentme.com'.
- Address :** Input field containing 'Jatrabari, Dhaka'.
- Profile Picture:** A small image of a man with arms crossed, wearing a dark shirt, positioned to the right of the form.
- Buttons:** A yellow 'Reset' button at the bottom left and a blue 'Update' button at the bottom right.

**Figure 5.11:** Update profile

## 5.12 Logout

Pressing on the logout button, the user will be logged out from user account.

# CHAPTER 6

## CONCLUSION AND FUTURE WORK

---

### 6.1 Conclusion

Our mission is to simplify the experience of connecting users & properties. We do this by delivering an integrated solution that brings together the right renter, with the right property, at the time, eliminating friction and driving efficiency. In 21 century the most common problem we can see is rental houses for strangers in the new city. So, we find out the modern solutions we build this website to solve the most common problem. By using this website, you can find out your desirable houses. Our website will give you a smooth user experience it has an easy-to-understand UI/UX design. It is fully optimized and has a 0.5% crashing issue.

### 6.2 Future Work

- Adding Google map so that users can easily find their exact location in a short time.
- Making Mobile Application for customer panel.
- Adding video with two or more photo of a house in the post.
- Adding Payment system for renting house on online.
- Adding different kind of services, for example Quick SMS, Quick Mail, Notification system for any kind of action.

## REFERENCES

- [1] Nandhini, R., Mounika, K., Muthu Subhashini, S. and Suganthi, S. (2018) Rental Home System for Nearest Place. International Journal of Pure and Applied Mathematics, 19, 1681.
- [2] Dipta Voumick, Prince Deb, Sourav Sutradhar, Mohammad Monirujjaman Khan. Development of Online Based Smart House Renting. Journal of Software Engineering and Applications, 14, 2021
- [3] Khan, F. (2011) Rental Housing. United Nations Human Settlements Programme.
- [4] Kamila Sommer, P.S.R.V. (2011) Run-Up in the House Price-Rent Ratio: How?
- [5] Whitten, Jeffrey L.; Lonnie D. Bentley, Kevin C. Dittman. (2003). Systems Analysis and Design Methods. 6th edition. ISBN 0-256-19906-X.
- [6] Wiegers, Karl; Beatty, Joy (2013). Software Requirements (3rd ed.). Microsoft Press. ISBN 978-0-7356-7966-5.
- [7] Scott W. Ambler. The Object Primer 3rd Edition Agile Model Driven Development with UML 2.
- [8] Beynon-Davies, Paul (2004). Database Systems. Basingstoke, UK: Palgrave: Houndmills. ISBN 978-1403916013.
- [9] Jacobson, I., Christerson M., Jonsson P., Övergaard G., (1992). Object-Oriented Software Engineering - A Use Case Driven Approach, Addison-Wesley.
- [10] Alakeson, V. (2011) Making a Rented House a Home: Housing Solutions for “Generation Rent”. Resolution Foundation, London, 4-42