



Islington college
(इस्लिंग्टन कॉलेज)

Module Code & Module Title
CS6P05NI Final Year Project

Assessment Weightage & Type
25% FYP Interim Report

Semester

2022 Autumn/Spring

Project Title: My Room

Student Name: Anish Aryal

London Met ID: 20048825

College ID: np01cp4s210084

Internal Supervisor: Aadesh Tandukar

External Supervisor: Brij Sharma

Assignment Due Date: 28th Dec, 2022

Assignment Submission Date: 28th Dec, 2022

Word Count (Where Required): 3753

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked.

I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

SUMMERY

This project involves the development of a web application for a property rental and sales platform. The project has made significant progress so far, with the initial phase and several subsequent tasks being completed. These tasks include research and planning, gathering resources, developing a risk management plan, creating a Gantt chart, designing the system architecture and user interface, and preparing various documents and diagrams. The project is currently in the development phase, with work underway on the login/register module. The remaining tasks, which include the development of various pages and features, finetuning and testing of APIs, and final testing and reviews, are yet to be started. The project is on track to be completed by April 2023 and the conclusion has been reached that the remaining tasks will be completed efficiently and on schedule.

Table of Contents

| | |
|---|-----------|
| SUMMERY..... | 2 |
| Table of Figures..... | 7 |
| 1. Introduction | 1 |
| 1.1. Problem Scenario | 1 |
| 1.2. Project as a solution..... | 2 |
| 1.3. Aims and Objectives | 2 |
| 1.4. Structure of Report | 3 |
| 1.4.1. Introduction..... | 3 |
| 1.4.2. Background/Literature review | 3 |
| 1.4.3. Development to date | 3 |
| 1.4.4. Analysis of progress | 3 |
| 1.4.5. Future work | 3 |
| 1.4.6. References | 3 |
| 1.4.7. Appendix | 3 |
| 2. Background/Literature Review..... | 4 |
| 2.1. Similar Projects..... | 5 |
| 2.1.1. Room Finder Nepal | 5 |
| 2.1.2. Gharbeti..... | 6 |
| 2.1.3. DalayDai..... | 7 |
| 2.1.4. Hamro rental..... | 8 |
| 2.2. Comparison Between the Systems | 9 |
| 2.3. Methodology..... | 10 |
| 2.3.1. Considered Methodology | 10 |
| 2.3.2. Chosen Methodology | 11 |
| 2.4. Resource Requirement | 12 |
| 3. Development to date..... | 13 |
| 3.1. Use case Diagram..... | 13 |
| 3.2. SRS Document..... | 13 |
| 3.3. Wireframes..... | 14 |
| 3.3.1. Wireframe for the whole system | 14 |
| 3.3.2. Wireframe for the individual component system..... | 14 |

| | | |
|-----------|--|----|
| 3.4. | Initial Entity Relationship Diagram | 15 |
| 4. | Analysis of Progress | 16 |
| 4.1. | Progress Table | 16 |
| 4.2. | Progress Review | 17 |
| 4.3. | Progress Timeline | 19 |
| 4.4. | Action Plan | 19 |
| 4.4.1. | Changes made | 19 |
| 5. | Future Work | 20 |
| 5.1. | Development of the login/register module | 20 |
| 5.2. | Develop Landing page | 21 |
| 5.3. | Develop Homepage | 21 |
| 5.4. | Develop Property Listing page/form | 21 |
| 5.5. | Develop Property Listing page/form | 21 |
| 5.6. | Develop Property Listing page | 21 |
| 5.7. | Develop Chat Module | 22 |
| 5.8. | Finetuning and testing APIs | 22 |
| 5.9. | Final Test and review | 22 |
| 5.10. | Submission of Final Report to RTE | 23 |
| 6. | References | 23 |
| 7. | Bibliography | 25 |
| 8. | Appendix | 26 |
| 8.1. | SRS Document | 26 |
| 8.1.1. | Project Title: My Room | 26 |
| 8.1.2. | Category: Web Application | 26 |
| 8.1.3. | Introduction | 26 |
| 8.1.4. | Overall Description | 27 |
| i. | Near-by schools/ markets and hospitals can be seen easily in the map for property | 31 |
| 8.2. | Resource Requirement | 52 |

| | |
|--|-----------|
| 8.3. High Level Use Case Diagram..... | 54 |
| 8.4. Wireframes..... | 62 |
| 8.4.1. Landing Page Wireframe | 62 |
| 8.4.2. Login Page | 63 |
| 8.4.3. Register Page..... | 64 |
| 8.4.4. Home Page | 65 |
| 8.4.5. Property Description Page | 66 |
| 8.4.6. Nav-link Room Page | 67 |
| 8.4.7. Nav-link Room Page | 68 |
| 8.5. Work Break Down Structure | 69 |
| 8.6. Gantt Chart..... | 70 |
| 8.7. Milestones | 71 |

Table of Tables

| | |
|---|----|
| Table 1: Comparison Between Similar systems..... | 9 |
| Table 2: Progress Table according to the Gantt chart | 17 |
| Table 3: Register Function Functional Requirements | 32 |
| Table 4: Login Feature Functional Requirement..... | 33 |
| Table 5: Property Listing feature functional requirement | 34 |
| Table 6: Map Location Feature Functional Requirement..... | 35 |
| Table 7: Comment Feature Functional Requirement | 36 |
| Table 8: Nearby schools/hospitals Functional Requirement..... | 37 |
| Table 9: Advance Search Feature Functional Requirement..... | 38 |
| Table 10: Chat feature Functional requirement..... | 39 |
| Table 11: Report feature functional requirement..... | 40 |
| Table 12: Admin review functional requirement | 41 |
| Table 13: Expiry time feature functional requirement..... | 42 |
| Table 14: Ban user functional requirement..... | 43 |
| Table 15: Warn user functional requirement | 44 |
| Table 16: Actors of the web application..... | 54 |
| Table 17: High level use case for register | 55 |
| Table 18: High Level Use case diagram for Login | 56 |
| Table 19: High Level Use case Diagram for property Listing..... | 57 |
| Table 20: High Level Use case diagram for map Location | 57 |
| Table 21: High Level use case diagram for comment..... | 58 |
| Table 22: High Level Use case diagram for show nearby schools/hospitals | 58 |
| Table 23: High Level Use case diagram for Advance search | 59 |
| Table 24: High Level Use Case Diagram for Chat Feature | 59 |
| Table 25: High Level Use Case Diagram for Report Feature | 60 |
| Table 26: High Level Use Case Diagram for Report Review Feature | 60 |
| Table 27: High Level Use Case Diagram for Ban user feature..... | 61 |
| Table 28: High Level Use Case Diagram for Warn user feature | 61 |

Table of Figures

| | |
|--|----|
| Figure 1: (Room Finder Nepal, 2022) Room Finder Nepal | 5 |
| Figure 2: (Gharbeti, 2022) Gharbeti..... | 6 |
| Figure 3: (Dalay Dai, 2022) Dalay Dai..... | 7 |
| Figure 4: (Hamro Rental, 2022) Hamro Rental | 8 |
| Figure 5: (tech-ict, 2022)Evolutionary Prototyping | 12 |
| Figure 6: Use case Diagram | 13 |
| Figure 7: Wireframe for the whole system..... | 14 |
| Figure 8: Initial ERD for the system | 15 |
| Figure 9: (Anon., 2022)MongoDB Logo..... | 19 |
| Figure 10: System Perspective | 29 |
| Figure 11: Flowchart for the user as seller | 46 |
| Figure 12: Flowchart of user as buyer | 47 |
| Figure 13: Flowchart for admin | 48 |
| Figure 14: Register page UI | 49 |
| Figure 15: Login Page UI | 50 |
| Figure 16: Landing page Ui | 51 |
| Figure 17: Vs code Logo | 52 |
| Figure 18: (Bootstrap, 2022) Bootstrap Logo | 53 |
| Figure 19: (Reactjs ORg, 2022)React Logo | 53 |
| Figure 20: (nodejs, 2022) Nodejs Logo | 54 |
| Figure 21: Wireframe for Landing Page..... | 62 |
| Figure 22: Wireframe for Login Page..... | 63 |
| Figure 23: Wireframe for Register Page..... | 64 |
| Figure 24: Wireframe for Homepage | 65 |
| Figure 25: Wireframe for Property Description Page | 66 |
| Figure 26: Wireframe for sidebar's Room menu item | 67 |
| Figure 27: Wireframe for Sidebar's Apartments menu item..... | 68 |
| Figure 28: Work Breakdown Structure..... | 69 |
| Figure 29: Gantt Chart..... | 70 |
| Figure 30: Milestones | 71 |

1. Introduction

First thing a person search for while planning to migrate from one place to another is a place to stay after reaching the destination. A room/apartment is mostly preferred by the students and working professionals who constantly need to travel from one place to another. However, there is no reliable platform for you to find a room/apartment of your choice. This complicates things a lot as you cannot shift to a new place without finding a home to stay and searching on the physical site may not be possible for most people.

The product being built for this project is room/apartment finding web application which will help the customers to negotiate or make a deal the property directly with the owner and to make the user experience as effective and efficient as possible this project will have maximum time limit/expiry-date of 1 month on the ads of the property posted by the user. A user can be a customer and seller as he/she can buy a property and sell their own property.

1.1. Problem Scenario

- I. People need to roam physically to find a room or apartment either to buy or rent.
- II. There are few websites which claim to help people to find the property of their type however, most of them are run by the agencies however they are not active on the website.
- III. There is no dedicated platform for the customer to find the property they need without any
- IV. As there are no alternatives for finding the property the customer need, finding the room for people who are planning to shift to major cities of Nepal face a lot of problems especially for students and working professionals as they don't have their established source of income and time on their hand to roam around to search physically.

V. Most websites in Nepal which claims to help find people property to live in they got too old ads like 5–6-year-old which makes the experience for the customer troublesome.

1.2. Project as a solution

This project will act as the best possible solution for the above-mentioned problem in the context of Nepal. With the time limit on the ad of property posted by the seller, customer can benefit the most as they will get to know how old the ad is and since there will no ads that are of before 30-day customer can surf the website with ease to find the property they need.

People can directly interact with the owner by using the contact info given by the owner on the post or in the comment section of the property ad's details. The details on the ad include property details, owner details, property location and picture of property and surrounding. There will also be the embedded map which shows the location of the property so the customer can easily go and visit the owner in person and deal with them. Maps will further show the nearby schools, Hospitals, Markets near the property.

1.3. Aims and Objectives

The aim of this project is to create a room finding web application. Which can solve the above mention problems people are facing while finding the rooms in major cities of Nepal.

The objectives which are the basis of the completion of the project are:

- To learn about web application and their workings.
- To use build a room finder web application at the end of the project
- To understand and implement usage of database along with web application.
- To learn about user interface.
- To understand about API programming.
- To deliver a complete functional product at the end of the project

1.4. Structure of Report

1.4.1. Introduction

Introduction consists of information about project, problem scenario, project as a solution, aims and objectives of the project.

1.4.2. Background/Literature review

This section includes the researches done on this topic and some of the similar systems like MY Room and comparison between those systems with own's project.

1.4.3. Development to date

In Development to date, progress of project and some diagrams which supports the progress is kept.

1.4.4. Analysis of progress

Analysis of progress chapter include the progress analysis of the project. How the progress is going is the progress is following the plans that was submitted in proposal and action plans for new changes made from proposal.

1.4.5. Future work

Future work includes the remaining task of this project to be carried out in future.

1.4.6. References

All the references taken while making this report and project is present in this section

1.4.7. Appendix

In appendix sections, it contains the brief description of SRS document, high level use case diagram and wireframes with screenshot. Some user interface designs entity relationship diagram and some contents from the previous report (proposal)

2. Background/Literature Review

A room/apartment finder web application is a tool that allows users to search for rental properties, such as apartments, rooms, or houses, in a specific location. These types of applications are commonly used by people who are looking for a place to live, either temporarily or permanently. Users can consider various factors when searching for a rental property, including location, price, size, and amenities. Some room/apartment finder web applications may also offer features like the ability to filter search results by these criteria, view photos and virtual tours of the properties, and contact landlords or property managers directly through the application.

The use of online platforms can make the process of searching for a rental property more efficient by allowing users to quickly and easily compare different properties and contact landlords or property managers directly. This can also increase the transparency and fairness of the rental market, as users can access a wider range of properties and landlords can find qualified tenants more easily.

In addition to benefits for users and landlords, room/apartment finder web applications can also be beneficial for cities and communities. These types of applications can help reduce vacancy rates and improve the overall supply and demand balance in the rental market, and can also increase the mobility of residents by making it easier for people to find and move to new rental properties. Overall, room/apartment finder web applications provide a convenient and efficient way for people to find rental properties and can have a positive impact on the rental market and communities.

2.1. Similar Projects

2.1.1. Room Finder Nepal

Roomsfindernepal is a web-based application. Rooms Finder Nepal is located at: Manamaiju, Kathmandu, Nepal 46000. According to alexa, roomsfindernepal.com has a global rank of #6991232 and it has some SEO issue. This site allows people to search for the property people need in their web. (Room Finder Nepal, 2022)

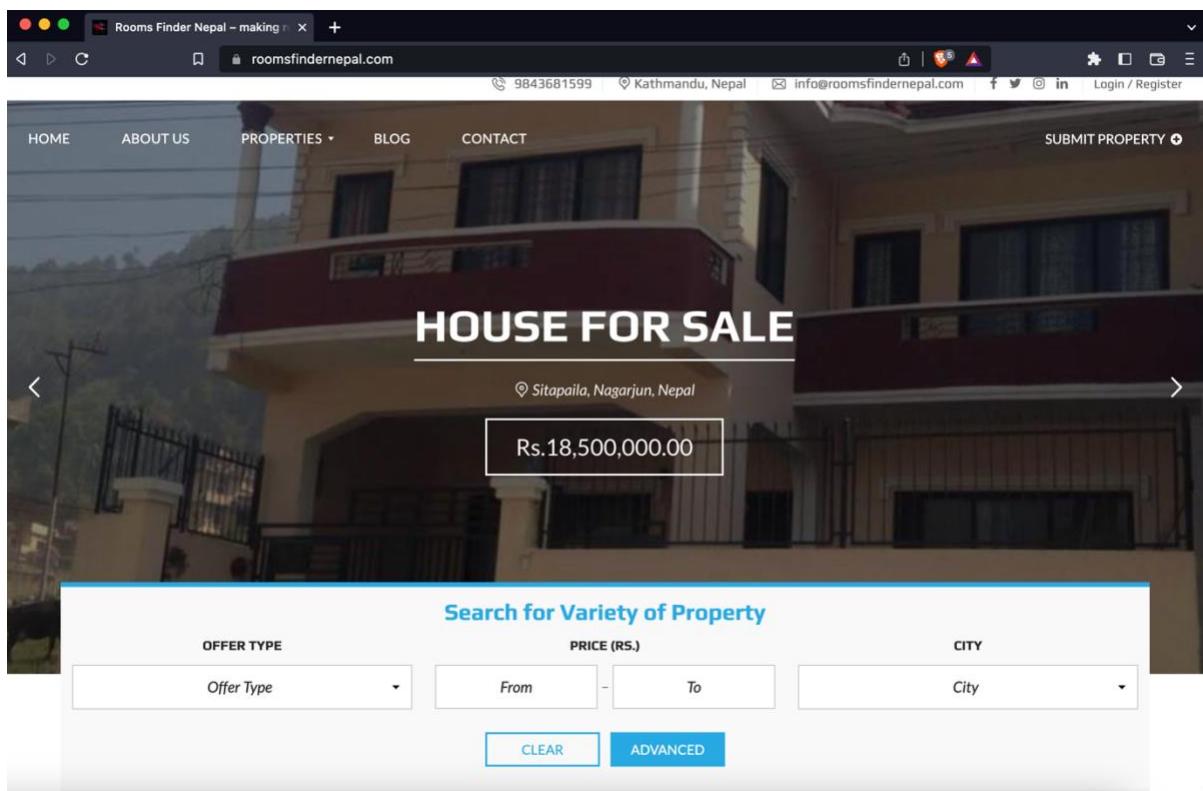


Figure 1: (Room Finder Nepal, 2022) Room Finder Nepal

2.1.2. Gharbeti

Gharbeti is a web platform that provides a comprehensive list of available rental properties in Nepal, including buildings, apartments, rooms, hostels, and land. Users can search and book these properties online, and the list is regularly updated with detailed information. It offers an efficient way for people to find and book the perfect rental property. (Gharbeti, 2022).

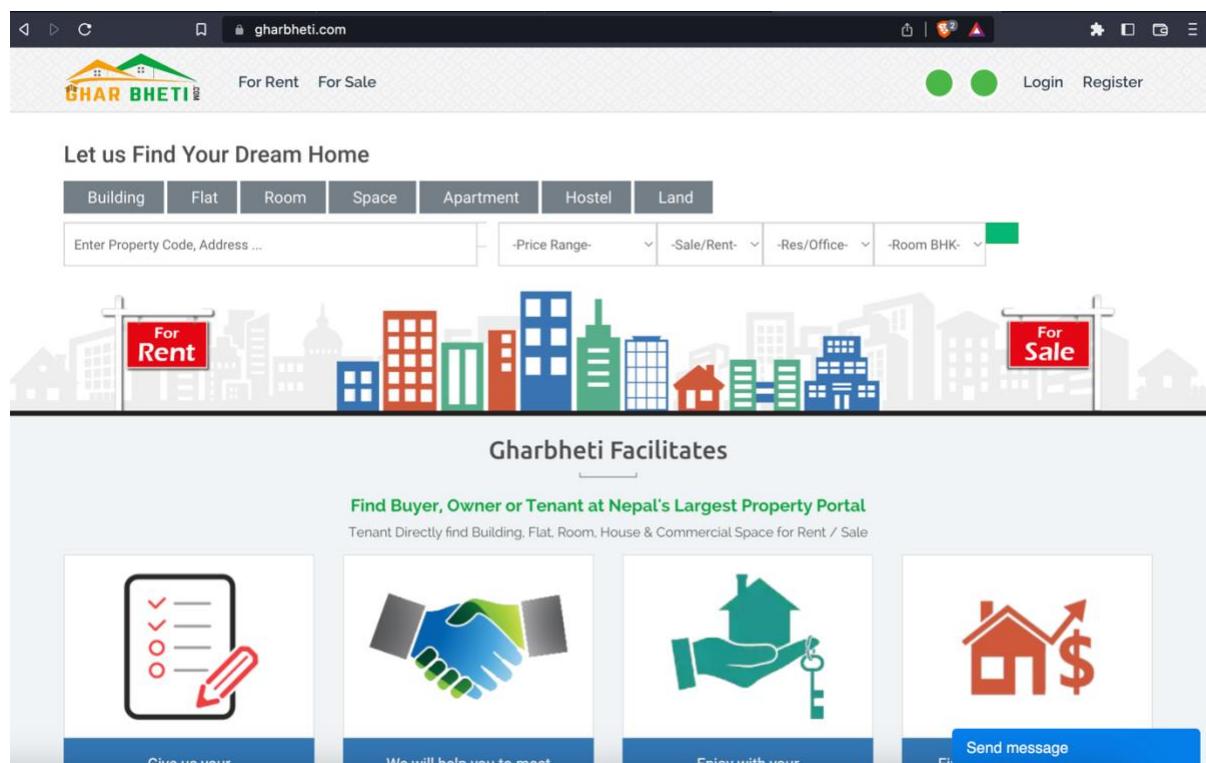


Figure 2: (Gharbeti, 2022) Gharbeti

2.1.3. DalayDai

DalayDai.com is a free online platform for buying and selling Houses, Lands and everything Real Estate. dalaydai.com makes, searching a perfect Home or a perfect plot of Land for Nepali people effortless and easy like it has never been. With frequent updates to the platform (Dalay Dai, 2022).

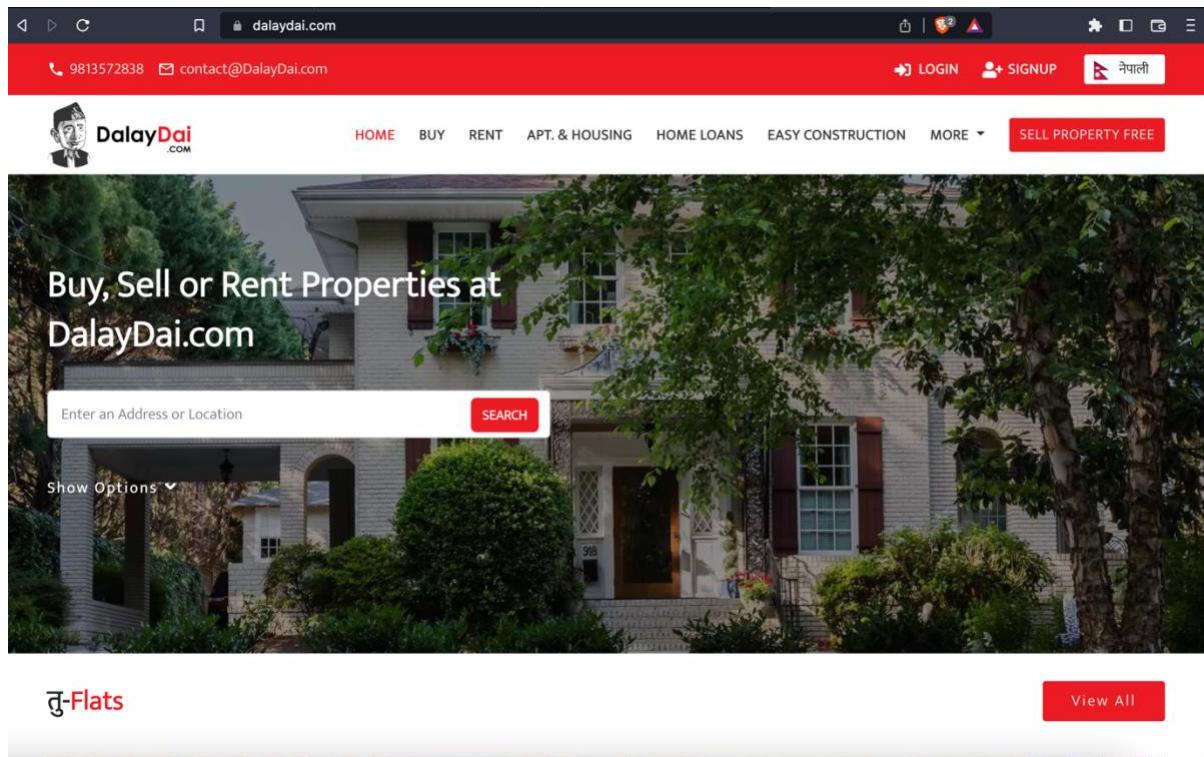


Figure 3: (Dalay Dai, 2022) Dalay Dai

2.1.4. Hamro rental

Hamro Rental is a free online platform for buying and selling Houses, Lands, Apartments Shops. However, this site seems to be inactive of long time and the user experience while using this site is also not that great. (Hamro Rental, 2022)

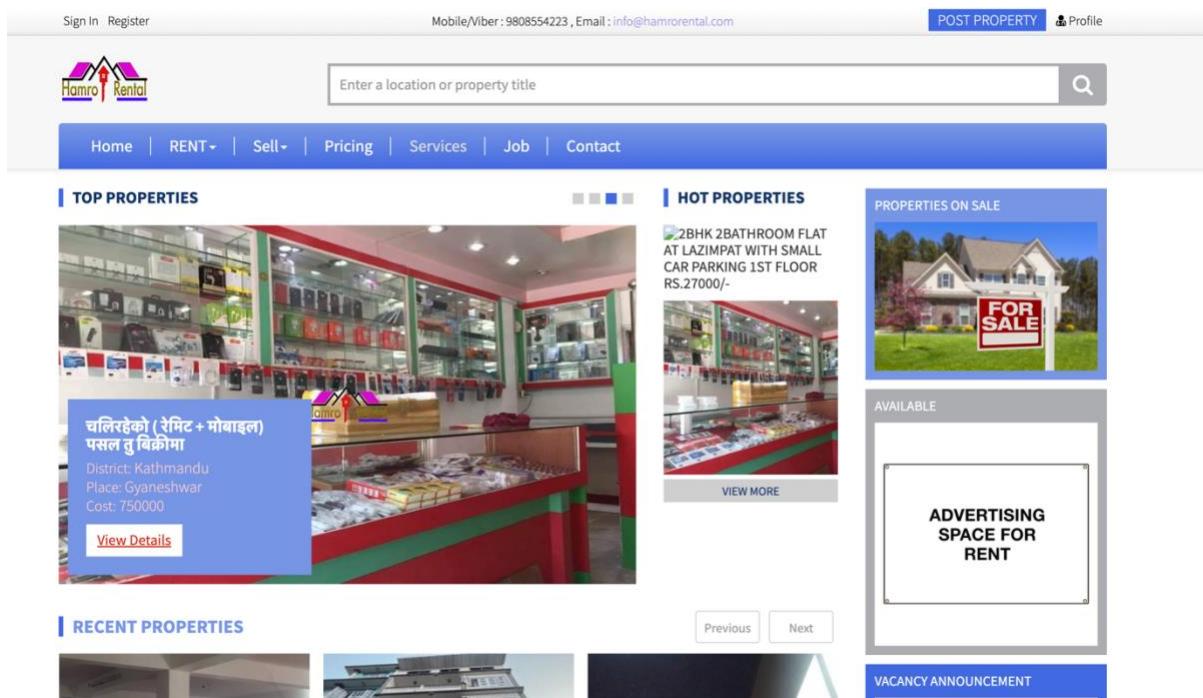


Figure 4: (Hamro Rental, 2022) Hamro Rental

2.2. Comparison Between the Systems

| S.N. | Features | Similar Projects | | | | |
|------|--|-------------------|----------|----------|--------------|---------------------|
| | | Room Finder Nepal | Gharbeti | DalayDai | Hamro Rental | MYRoom (My Project) |
| 1 | Advance Search Feature | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2 | Wishlist | 🚫 | 🚫 | ✓ | ✓ | ✓ |
| 3 | Location | ✓ | ✓ | 🚫 | 🚫 | ✓ |
| 4 | Ad Expiry Time | 🚫 | ✓ | ✓ | 🚫 | ✓ |
| 5 | Comments in ads | ✓ | 🚫 | ✓ | ✓ | ✓ |
| 6 | Realtime Chat | 🚫 | 🚫 | 🚫 | 🚫 | ✓ |
| 7 | Report unreliable ads | 🚫 | 🚫 | 🚫 | 🚫 | ✓ |
| 8 | Pet Allowed Filter | 🚫 | 🚫 | 🚫 | 🚫 | ✓ |
| 9 | Show Nearby Schools, Markets, Hospitals in map view for property | 🚫 | 🚫 | 🚫 | 🚫 | ✓ |
| 10 | Upload Video tour for the property | 🚫 | 🚫 | 🚫 | 🚫 | ✓ |

Table 1: Comparison Between Similar systems

2.3. Methodology

2.3.1. Considered Methodology

- **Scrum Methodology**

(Digite, 2019) Scrum is an agile software development technique that relies on incremental and iterative methods. Scrum is an agile framework that is quick, flexible, adaptable, and effective and is made to provide value to the client throughout the course of the project. Through an atmosphere of open communication, shared ownership, and constant improvement, Scrum's main goal is to serve the needs of the client. The development process begins with a general understanding of what needs to be created, then the product owner develops a list of desired qualities that are prioritized (product backlog).

- **RUP**

(Minott, 2022) The Rational Unified Process (RUP) definition, also known as the Unified Process Model, is a web-enabled, object-oriented software development process. Object-oriented software engineering refers to any software development approach that uses visual models organized around objects and iteratively goes through analysis, design, and implementation. The RUP framework was created by International Business Machines Corporation (IBM) in 2003 and was one of the first widely used iterative methods.

2.3.2. Chosen Methodology

The methodology chosen for this project is Evolutionary Prototyping.

Evolutionary Prototyping

The methodology chosen for this project is Evolutionary Prototyping because this methodology provides project with the most flexibility. Since this is the first project, at this big scale working individually. The system might be developed with some modules/features like I had hoped and the work flow might have been disturbed however That will not be the case anymore because this methodology help solve that issue as modules during in the development phase are independent to one another even if the development got stuck on some problems on one of the modules, that module could be kept on hold and also can keep working on other parts. There will also be feedbacks given by the supervisors' after completing a module The system could further be optimized and make it better with the help of supervisors' advice and further research.

The concept is to give the user with an initial prototype. They provide feedback and improvement suggestions. The developer takes action on these and then offers a more improved prototype. The user provides input once more. The procedure is then repeated. As a result, the prototype 'evolves' towards the final system at each iteration. (Teach-itc, 2022)

Advantage of this methodology is that the end user can interact with the prototypes and which increases the user engagement and the

feedback given by the user will help improve the product to its optimal glory. This system will be very efficient to meet the user's requirement.

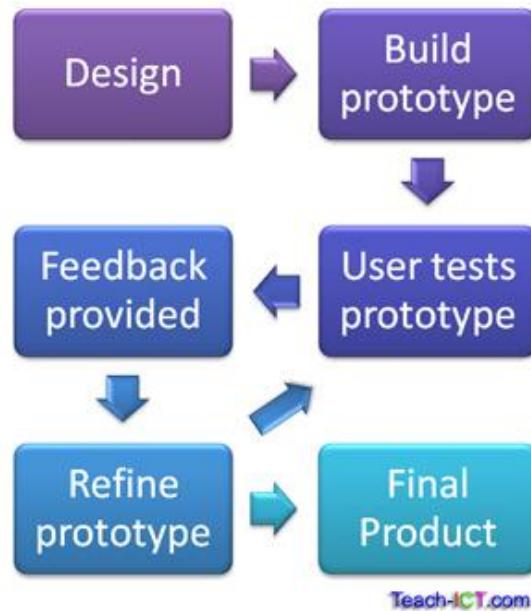


Figure 5: (tech-ict, 2022)Evolutionary Prototyping

2.4. Resource Requirement

These Technological Requirements are explained below in appendix section's [Resource Requirement](#)

Software Requirements

- IDE VS code
- HTML, CSS, JavaScript, Bootstrap and React js will be used for the front-end.
- Node JS, Next JS will be used for backend
- MongoDB will be used for the database.

Hardware Requirement

- A reliable Laptop to work on so MacBook pro will be used for this project

3. Development to date

3.1. Use case Diagram

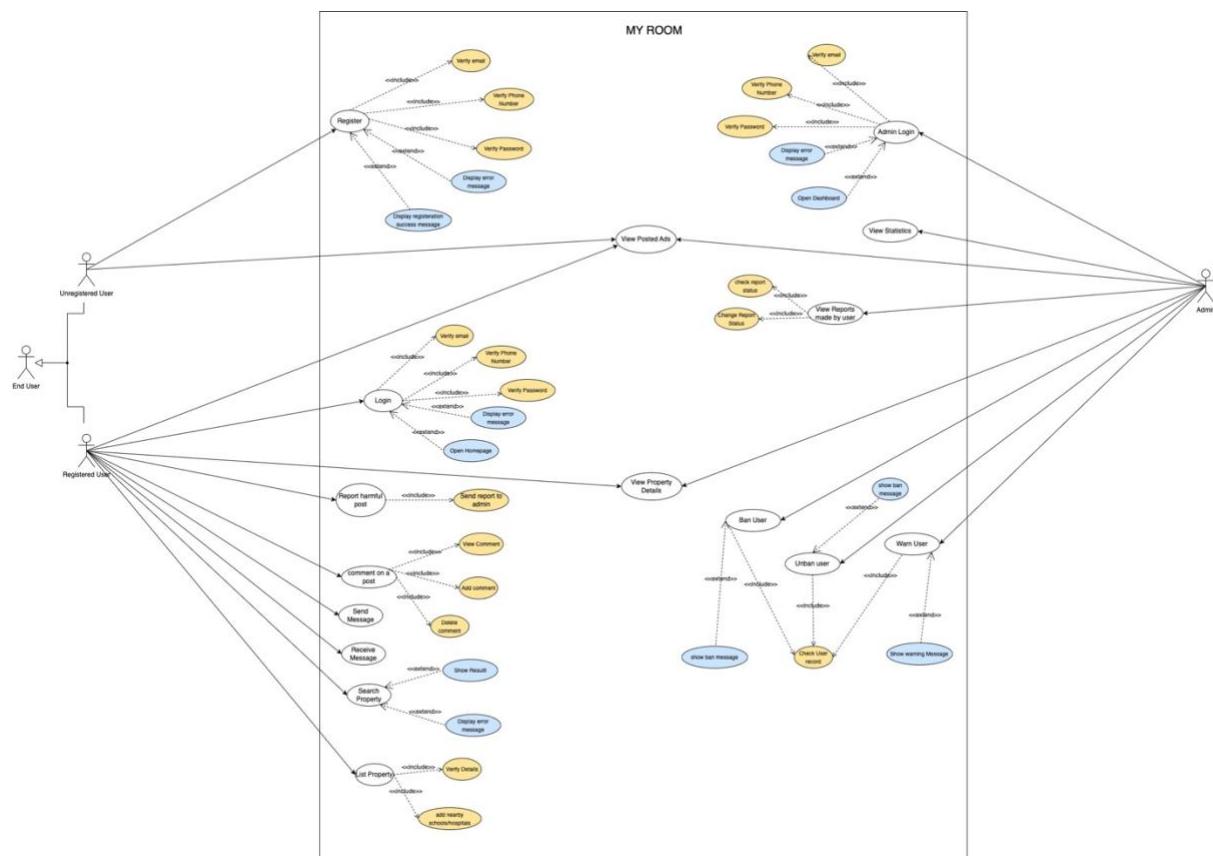


Figure 6: Use case Diagram

3.2. SRS Document

The SRS document is kept further down inside the Appendix section. Here, is the link to navigate to the whole [SRS Document](#) for this Project.

3.3. Wireframes

3.3.1. Wireframe for the whole system

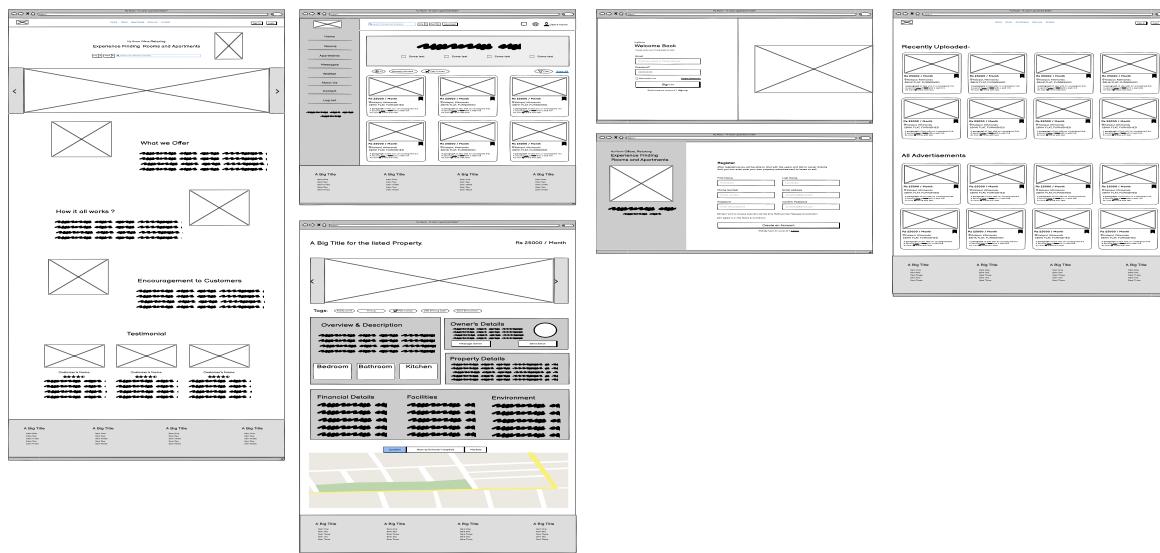


Figure 7: Wireframe for the whole system

3.3.2. Wireframe for the individual component system

The detailed images of wireframe of whole project are kept further down inside the appendix section. Here, is the link to navigate to the detailed [Wireframes](#) for each webpage.

3.4. Initial Entity Relationship Diagram

Assumptions for initial ERD

- A user can create one or more listings (1:N).
- A listing can be for a single property (1:1).
- A user can participate in one or more chats (1:N).
- A user can send one or more messages in a chat (1:N).
- A user can make one or more reports (1:N).
- A report can be about a single listing (1:1).
- An admin can review one or more reports (1:N).

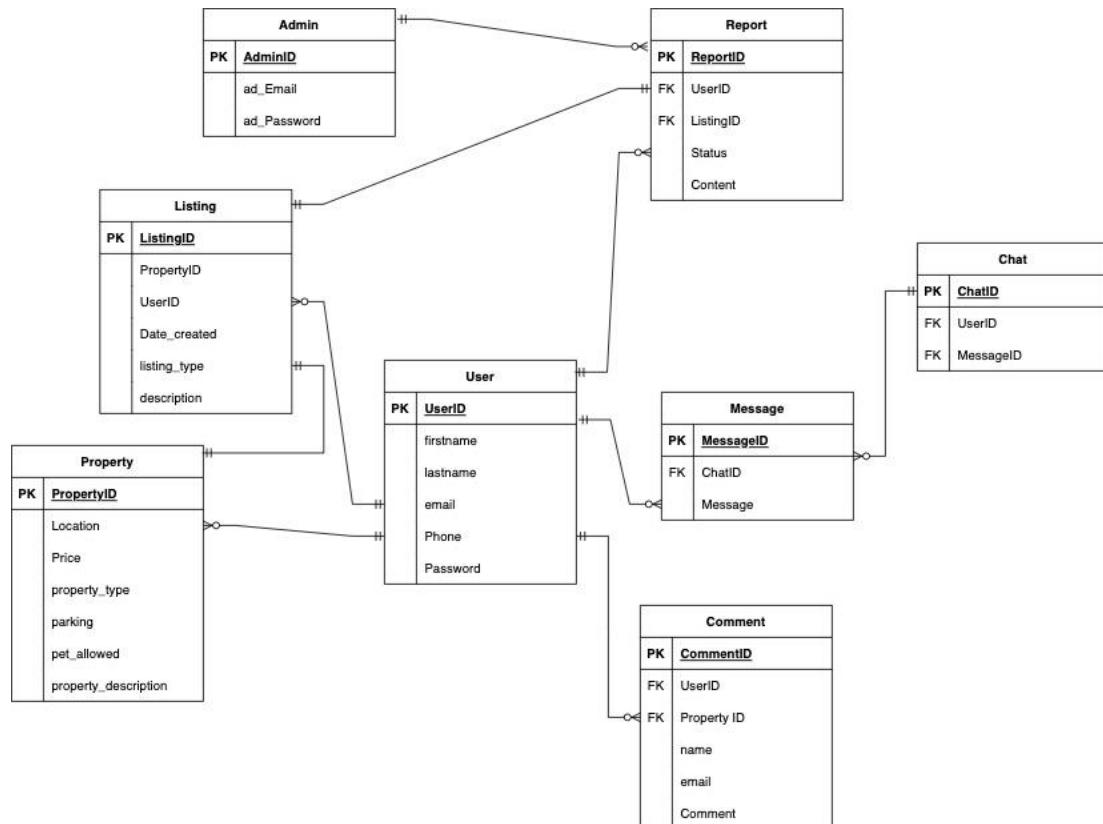


Figure 8: Initial ERD for the system

4. Analysis of Progress

4.1. Progress Table

| S.N. | Tasks | Status | Progress |
|------|---|-----------|----------|
| 1. | Topic Research | Completed | 100% |
| 2. | Topic Finalization | Completed | 100% |
| 3. | Feasibility Study | Completed | 100% |
| 4. | Technical Requirement Research | Completed | 100% |
| 5. | Similar Projects Research | Completed | 100% |
| 6. | Finalize Proposal | Completed | 100% |
| 7. | Gather Required Resources | Completed | 100% |
| 8. | Plans for Risk Management | Completed | 100% |
| 9. | Plans for unforeseen future Change | Completed | 100% |
| 10. | Design Logo | Completed | 100% |
| 11. | Develop use case diagram for overall system | Completed | 100% |
| 12. | Develop SRS Document | Completed | 100% |
| 13. | Develop High Level use case diagram | Completed | 100% |
| 14. | Develop Wireframes | Completed | 100% |
| 15. | Design Initial Entity Relationship Diagram | Completed | 100% |
| 16. | Develop required Flowchart | Completed | 100% |
| 17. | User Interface Design | Completed | 100% |
| 18. | Finalize Interim Report | Completed | 100% |
| 19. | Develop Login/Register Module | Doing | 50% |
| 20. | Develop Landing page | To-do | 0% |
| 21. | Develop Homepage | To-do | 0% |
| 22. | Develop Property Listing Page/form | To-do | 0% |
| 23. | Develop Advance Search Feature | To-do | 0% |
| 24. | Develop Property Description Page | To-do | 0% |
| 25. | Develop Chat Module | To-do | 0% |

| | | | |
|-----|--|-------|----|
| 26. | Finetune and test API used while development | To-do | 0% |
| 27. | White Box Testing of the system | To-do | 0% |
| 28. | Black Box Testing of the system | To-do | 0% |
| 29. | Final Test and Reviews From supervisors | To-do | 0% |
| 30. | Proper Documentation of the project | To-do | 0% |
| 31. | Finalization of the Project | To-do | 0% |
| 32. | Finalization of the FYP Report | To-do | 0% |
| 33. | Submission of Final Year Project | To-do | 0% |

Table 2: Progress Table according to the Gantt chart

4.2. Progress Review

The project has made significant progress so far, with all tasks up to the development of the interim report being completed. In the initial phase for this project proper research was done and based on that research topic for the final year project was selected. Then in order to determine the technical and financial feasibility for this project proper research and study was done and reached a final conclusion about how the development of the project should be undertaken moving forward. Afterwards to determine the software and hardware requirement for the project Technical Requirement Research was done, then the initial phase was completed for this project.

Then came the next phase, where research on previously existing systems and similar projects was done which helped to understand the market and the competition. The required resources, including hardware and software, were gathered for the project. Risk management plans were also developed to identify and mitigate potential risks. Then the Gantt chart was made in order to track the progress and allocate the time properly for achieving the project on time which is acting as the blue print for the project. Based on this research and findings, the final proposal for the project was prepared.

Plans were developed to deal with unforeseen future changes that may arise during the development of the project. After the required research and study was carried, logo and use case diagram of overall system was designed. The SRS (Software Requirements Specification) document was prepared to detail the requirements for the project. High level use case diagrams were also developed to outline the user interactions with the system.

Then came the phase that gave shape to the project i.e., development of wireframes and UI/UX (User Interface / User experience) design for the web application. Wireframes were developed to provide a visual representation of the project's user interface. Initial Entity Relationship Diagrams were also prepared to show the relationships between the different entities in the project. Flowcharts were developed to outline the flow of events in the project particularly flow for user and admin. And, the user interface for the project was designed to ensure that it was user-friendly and visually appealing.

The interim report, which provides an overview of the progress made so far, was finalized. The project progress is moving smoothly, strictly meeting the requirements of the Gantt chart. As of now as per the Gantt chart project has reached its development phase. Currently, work is underway on the login/register module and the progress is at 50% as per the Gantt chart. The remaining tasks, including the development of the landing page, homepage, property listing page/form, advance search feature, property description page, chat module, finetuning and testing of APIs, white box testing, black box testing, final testing and reviews, proper documentation of the project, finalization of the project, and finalization of the FYP (Final Year Project) report, are yet to be started. The progress for these tasks is at 0% as those will be carried out according to the plans laid out in the Gantt chart.

Overall, the project has made significant progress so far and is on track to be completed successfully. So, for the moment the conclusion has been reached that the remaining tasks will be completed efficiently and on schedule.

4.3. Progress Timeline

Based on the progress made so far it is clear that the project progress is moving along with the time allocated on the Gantt chart which was submitted in the proposal. And it is highly possible to achieve the project on time if any unforeseen circumstances didn't affect the projects progress. However, all the future work must be done as planned in the Gantt chart.

4.4. Action Plan

The project will be moving forward along with the schedule of the Gantt chart as all the work till date is met. And the iterations (frontend development, backend development, unit testing & reviews from supervisors) present in development phase according to the evolutionary prototyping will be carried out accordingly. However, there is some changes from the previously submitted proposal.

4.4.1. Changes made

In the proposal for this project, it was decided that MYSQL will be used as a database however there was compatibility issue of MYSQL server with M1 Mac, which means that it cannot be used for the project. On the other hand, MongoDB is a popular database software that is known for its compatibility with various operating systems and hardware, including M1 Macs. Therefore, switching to MongoDB from MySQL would be a suitable solution to ensure that the project can be carried out smoothly and without any issues related to compatibility.



Figure 9: (Anon., 2022)MongoDB Logo

MongoDB is a document-oriented NoSQL database that is used for large-scale data storage. MongoDB employs collections and documents rather than tables and rows as in traditional relational databases. Documents are made up of key-value pairs, which are the fundamental unit of data in MongoDB. Collections are analogous to relational database tables in that they include sets of documents and functions. MongoDB is a database that first appeared in the mid-2000s (Taylor, 2022).

In summary, switching to MongoDB from MySQL for this project would be a practical decision based on compatibility and performance considerations. It would ensure that the project can be carried out without any issues related to database software compatibility and can take advantage of the advanced features offered by MongoDB.

5. Future Work

The future works are the tasks that have yet to be started and are scheduled to be finished in the upcoming weeks or months. The progress for these tasks is currently at 0%, but this does not necessarily mean that no effort has been put into them. Some of these tasks have already been planned or prepared for ahead of time, but the actual work will begin at a later point in the project timeline. Following tasks are on the to do list according to the plan and Gantt chart.

5.1. Development of the login/register module

Development of this module is underway and half completed and it is moving accordingly with the Gantt chart. This module development will be completed on time as its development reached the backend development stage already.

5.2. Develop Landing page

This is the first page users see when they visit the website and provides an overview of the purpose and content of the site, has yet to be initiated. It should include calls to action, such as a sign-up or login button, to encourage user engagement. This task is crucial as the landing page is often the first impression users have of the website and it is important to make a good first impression to keep users interested in exploring the site.

5.3. Develop Homepage

The homepage is the main page of the website and it should be user-friendly and engaging to keep users on the site. It should be organized and include calls to action to encourage specific actions like signing up or making a purchase. The homepage is often the most visited page on a website, so it's important to make it as effective as possible.

5.4. Develop Property Listing page/form

The property listing page lists all available properties and allows users to filter and sort them. The property listing form allows users to list their own properties. This task is important for finding and advertising properties.

5.5. Develop Property Listing page/form

This task involves creating a feature that allows users to search for properties on the website using specific criteria, such as location, size, and price. The goal is to provide an easy-to-use tool that delivers accurate and relevant results to improve the overall user experience.

5.6. Develop Property Listing page

The property description page is a webpage that provides in-depth information about a particular property, including its location, size, and features, as well as photos and a description of the property. This task is crucial because it enables users to fully understand a specific property before making a decision on whether to rent or buy it.

5.7. Develop Chat Module

The chat module is a feature on the website that enables real-time communication between users. It allows users to send individual message to the other person and it should be user-friendly. This task is important because it enables communication between users and helps foster a sense of community on the website.

5.8. Finetuning and testing APIs

Optimizing and APIs on the website is a task that has not yet been started. This involves making sure that the APIs are functioning properly and efficiently in order to allow different software applications to communicate with each other on the website. This task is important because it ensures that the website is able to communicate effectively with other software applications, which is necessary for the proper functioning of many of the website's features.

5.9. Final Test and review

This task involves various testing methods such as black box and white box testing to ensure the website's functionality and internal structure. The system will also be tested using specific test cases and receive final reviews from supervisors. This is important to identify and fix any issues before launch and ensure a high-quality website.

5.10. Submission of Final Report to RTE

Project will be submitted to the RTE department's online portal in google classroom in the given deadline. This includes whole project folders which contains reports, plans, researches, codes, designs, etc.

6. References

Digite, 2019. *Digite*. [Online]

Available at: <https://www.digite.com/agile/scrum-methodology/>
[Accessed 20 November 2022].

Minott, C., 2022. *Study.com*. [Online]

Available at: <https://study.com/learn/lesson/rational-unified-process-methodology-examples.html>

[Accessed 20 November 2022].

Teach-itc, 2022. *Teach*. [Online]

Available at: https://www.teach-ict.com/as_a2_ict_new/ocr/A2_G063/331_systems_cycle/prototyping_RAD/miniweb_pg3.htm

[Accessed 10 November 2022].

Taylor, D., 2022. *Guru99*. [Online]

Available at: <https://www.guru99.com/what-is-mongodb.html>
[Accessed 26 dec 2022].

Anon., 2022. *MongoDB*. [Online]

Available at: <https://www.mongodb.com/what-is-mongodb>
[Accessed 25 dec 2022].

Room Finder Nepal, 2022. *Room Finder Nepal*. [Online]

Available at: <https://roomsfindernepal.com/about-us/>
[Accessed 25 dec 2022].

Dalay Dai, 2022. *Dalay Dai*. [Online]

Available at: <https://dalaydai.com/info/about/>

[Accessed 26 dec 2022].

Gharbeti, 2022. *Gharbeti*. [Online]

Available at: <https://www.gharbheti.com/aboutus>

[Accessed 26 dec 2022].

Hamro Rental, 2022. *Hamro Rental*. [Online]

Available at: <https://www.hamrorental.com/>

[Accessed 25 dec 2022].

tech-ict, 2022. *tech-ict*. [Online]

Available at: https://www.teach-ict.com/as_a2_ict_new/ocr/A2_G063/331_systems_cycle/prototyping_RAD/miniweb/pg3.htm

[Accessed 25 dec 2022].

vscode, 2022. *Visual Studio Code*. [Online]

Available at: <https://code.visualstudio.com/docs>

[Accessed 25 dec 2022].

mozilla, 2022. *mdnwebdocs*. [Online]

Available at: <https://developer.mozilla.org/en-US/docs/Web/HTML>

[Accessed 26 dec 2022].

Andrew Zola, 2022. *techtarger*. [Online]

Available at: <https://www.techtarget.com/whatis/definition/bootstrap>

[Accessed 27 dec 2022].

Bootstrap, 2022. *Bootstrap*. [Online]

Available at: <https://getbootstrap.com/docs/5.0/about/brand/>

[Accessed 27 dec 2022].

Reactjs ORg, 2022. *Reactjs*. [Online]

Available at: reactjs.org

[Accessed 27 dec 2022].

Herbert, D., 2022. *Hubspot*. [Online]

Available at: <https://blog.hubspot.com/website/react-js>

[Accessed 27 dec 2022].

Taha Sufiyan, 2022. *Simplilearn*. [Online]

Available at: <https://www.simplilearn.com/tutorials/nodejs-tutorial/what-is->

nodejs#what_is_nodejs

[Accessed 27 dec 2022].

nodejs, 2022. *nodejs*. [Online]

Available at: <https://nodejs.org/en/>

[Accessed 27 dec 2022].

7. Bibliography

- <https://www.merorealestate.com/properties-details?&id=142>
- <https://roomsfindernepal.com/>
- <https://www.gharbheti.com/>
- <https://www.rentalnepal.com/property/studio-room-for-rent-in-bhaisepati-12463>
- <http://zillow.com/>
- <http://realtor.com/>
- <http://rightmove.co.uk/>
- <http://redfin.com/>
- <http://realestate.com.au/>
- <http://99.co/>
- <http://housing.com/>
- <http://property24.com/>
- <http://99acres.com/>
- <https://nazrul.me/>
- <https://www.geeksforgeeks.org/>
- <https://www.w3schools.com/>

8. Appendix

8.1. SRS Document

8.1.1. Project Title: My Room

8.1.2. Category: Web Application

8.1.3. Introduction

8.1.3.1. Purpose

When developing software, the requirements stage involves the creation of Software Requirements Specifications (SRS) (also called a requirements document). This report is created after all requirements have been collected, assessed, and laid out, serving as a foundation for software engineering initiatives. In order to help clients determine whether the report (SRS) meets their needs, it is a formal report that also acts as a software representation. The prerequisites for both users and the system are carefully described as well.

The purpose of this SRS is to specify the requirements for the room/apartment finder web application, which will allow users to search for and list properties for lease or sale and make the process of finding rooms/apartments seamless.

8.1.3.2. Project Scope

The proposed room/apartment finder web application is a platform that allows users to search for and find rooms or apartments to rent or buy. The

room/apartment finder web application will include a login and registration system, a chat feature, a comment feature, a property listing feature, a map showing the location of properties, a search and filtering function, a feature for reporting inappropriate listings, etc. The application will also allow users to upload images and videos of their properties. The application will be developed making it proper responsive and optimized for the use in the desktop websites and for mobile devices as well. It allows users to list their own properties for rent or sale, and to interact with other users through a chat feature.

Some potential advantages of this system for people are:

- **Convenience:** The room/apartment finder web application provides a centralized platform for users to search for and find rooms or apartments, saving them the time and effort of searching through multiple sources.
- **Ease of use:** The application has a user-friendly interface and various features, such as an advanced search option and easy filtration options, to make it easy for users to find properties that meet their specific needs and preferences.
- **Increased visibility:** The application allows users to list their own properties, providing them with increased visibility and the opportunity to reach a wider audience of potential renters or buyers.
- **Communication:** The chat feature allows users to communicate with each other and ask questions about properties, facilitating the process of finding and securing a room or apartment.

8.1.4. Overall Description

8.1.4.1. Product Description

The room/apartment finder web application will be developed as a web-based application that can be accessed from any device with a web browser. It will be developed using React for the frontend and Node.js for the backend, and will use a Mongo DB database to store user and property information.

8.1.4.2. User Characteristics

The users of the room/apartment finder web application will be individuals looking for properties to lease or buy, or individuals looking to list their own properties for lease or sale. They may be interested in finding properties in specific locations, with certain amenities, or that meet other criteria.

8.1.4.3. Constraints

There may be constraints on the development of the room/apartment finder web application, such as budget constraints, time constraints, or technical constraints. Some of the constraints are listed below:

- i. Product may not be completed on time as the project might take longer than expected
- ii. There could be the problem with budget as some APIs may not be free and should be purchased
- iii. Some frameworks, API, technology may not work on the device that the project is being built on as
- iv. There might occur some technical difficulty like some software, frameworks not working properly on the device this project is being developed.

8.1.5. System Perspective

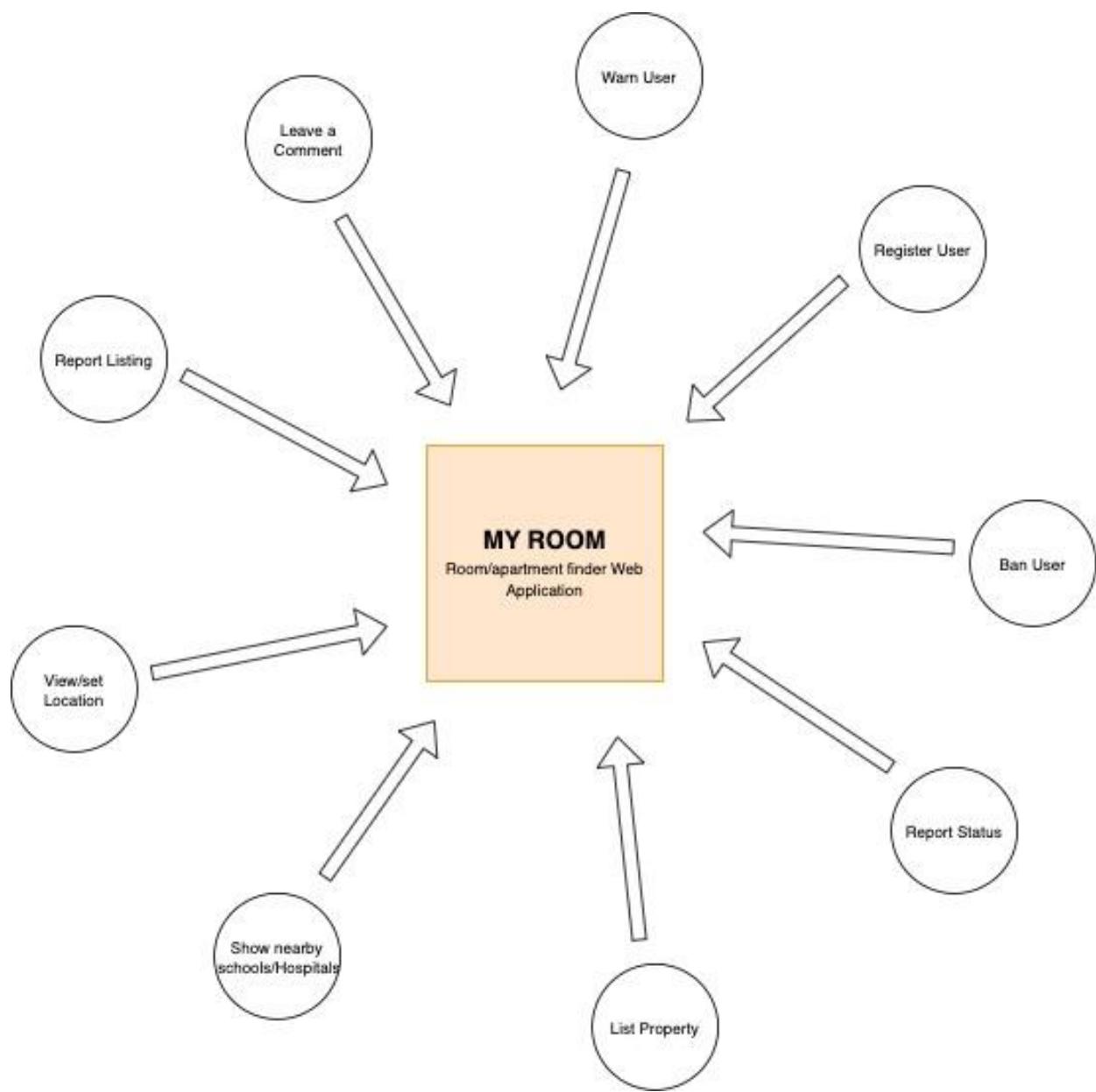


Figure 10: System Perspective

8.1.6. System Features

Primary Features

- i. Online Ad Posting of the property to rent.
- ii. Seller can Specify either to lease or to sell.
- iii. The user is divided into buyer and seller however they both can sell or buy the property listed.
- iv. There will be the expiry date of 30 days on every ad posted.
- v. Time passed after the ad is posted will be visible to the user
- vi. The ad posted by the seller will disappear after the expiry time passes and the seller will be notified about the removal of ad.
- vii. The seller could further decide to repost the ad if the property is not yet sold out or rented.
- viii. Embedded map will be shown in the details of the property added by the seller which is a must and seller cannot post any property without the map showing property's location.
- ix. There will be a section for the images of the property including the surrounding of the property, seller should upload the image of the property before making the ad publicly available.
- x. Property filter option will be available to the users which will help the user filter the property listed for "Rent Only", "Buy Only" & "Rent or Buy Both"
- xi. User can search up for the property they want, further this function enables user to search the property based on nature of the property like location, price range, recent ads, property size, etc.
- xii. Buyer will be able to choose the property they want and add it to their Wishlist.
- xiii. Seller can specify what luxury or facilities are available in the property they are selling like, furnished or non-furnished , parking space, Water, Electricity, bathroom, etc.

Secondary Features

- i. Near-by schools/ markets and hospitals can be seen easily in the map for property
- ii. Property filter with pets allowed or not allowed
- iii. Seller could upload videos of the property through which buyer can have virtual tour of the property
- iv. Buyers could directly ask question to the sellers about the additional information of property on the comments section present in every ad.
- v. Will be report feature on each ad so that user can report the ads that are not good for the site
- vi. Admin will receive the report notification and can remove that ad or block the user if they violate the terms more than once.

8.1.7. External Interface requirement

- **User interfaces:** The room/apartment finder web application will have a user interface that allows users to search for properties, list their own properties, chat with other users, and leave comments on property descriptions. The interface will be responsive, meaning it will adjust to the size of the device it is being viewed on. The interface will also include navigation links to allow users to access different parts of the application, and will have a search bar for users to enter their search criteria. This whole interface will be as modern minimal and effective as possible.
- **Hardware interfaces:** The room/apartment finder web application will not require any specific hardware interfaces.
- **Software interfaces:** The room/apartment finder web application will interface with the Mongo DB database to store and retrieve user and property information. It will also interface with a mapping library like Google Maps to display the location of properties and nearby schools and hospitals.

- Communication interfaces:** The room/apartment finder web application will use a real-time communication library like Socket.io to implement the chat feature.

8.1.8. Functional Requirements

8.1.8.1. Register Feature

| ID | Requirement Description | |
|-----|-------------------------|---|
| FR2 | S.Req.ID | System Requirement |
| | R1 | User can should be able to view the register form. |
| | R2 | Register form should validate the info given by the user |
| | R3 | System should check is the entered phone number and email is unique to the application or not |
| | R4 | Registration successful message should be shown to the user |
| | R5 | If any error is found system shall notify the user with proper message |

Table 3: Register Function Functional Requirements

8.1.8.2. Login Feature

| ID | Requirement Description | |
|------------|--------------------------------|---|
| FR1 | S.Req.ID | System Requirement |
| | R1 | The login form must be accessible from the home page of the application. |
| | R2 | The login form must validate the email and password entered by the user. |
| | R3 | The login form must allow the user to reset their password if they have forgotten it. |
| | R4 | The login form must display an error message if the login attempt is unsuccessful. |
| | R5 | User should be redirected to home page if the login is successful. |

Table 4: Login Feature Functional Requirement

8.1.8.3. Property Listing Feature

| ID | Requirement Description | |
|----------|---|--|
| FR3 | People can list their property for sell/lease after filling the form properly. | |
| S.Req.ID | System Requirement | |
| R1 | user interface is needed to allow users to navigate to the listing form from where it is most accessible like navbar. | |
| R2 | User should be able to specify either to lease or to sell. | |
| R3 | Listing form should contain almost all the description that is required before the listing. | |
| R4 | User should be able to select the location of the property. | |
| R5 | User should be able to upload image/video of the property. | |
| R6 | Listing should disappear after the expiry time set by the user or default expiry time. | |
| R7 | Form should have a submit button which validates the form and add the property to the database and listing page. | |
| R8 | Proper Listing should be displayed to the user. | |
| R9 | In case error is detected user should be notified with proper message. | |

Table 5: Property Listing feature functional requirement

8.1.8.4. Map Location Feature

| ID | Requirement Description | |
|-----|--|---|
| FR4 | User can view the location of property in map view on property description page. | |
| | S.Req.ID | System Requirement |
| | R1 | User interface is needed to display the map on the property description page. |
| | R2 | Map should show the location with a proper visible marker |
| | R3 | System is required to enable interactivity on the map, such as the ability to zoom in or out and pan around |

Table 6: Map Location Feature Functional Requirement

8.1.8.5. Comment Feature

| ID | Requirement Description | |
|-----|-------------------------|--|
| FR5 | S.Req.ID | System Requirement |
| | R1 | Comment section is needed comment on the property description page, and a system is required to store the comment in the database. |
| | R2 | People should be able to view the comments made previously on that listing. |
| | R3 | User should be able to delete their own comments, and a system must update the database accordingly. |
| | R4 | Username and email of the person who left the comment should be displayed properly in comment section |

Table 7: Comment Feature Functional Requirement

8.1.8.6. Show Nearby Schools and Hospitals Feature

| ID | Requirement Description | |
|----------|---|--|
| FR6 | User can view the nearby schools and hospitals around the property listed. | |
| S.Req.ID | System Requirement | |
| R1 | User interface is needed to display the map that shows nearby schools and hospitals on the property description page. | |
| R2 | System is required to enable interactivity on the map, such as the ability to zoom in or out and pan around | |
| R3 | Multiple markers should be used to show the schools/hospitals | |

Table 8: Nearby schools/hospitals Functional Requirement

8.1.8.7. Advance Search Feature

| ID | Requirement Description | |
|-----|-------------------------|---|
| FR7 | S.Req.ID | System Requirement |
| | R1 | A user interface is needed to search the property and an advance search form should be available to the user. |
| | R2 | Search button should be present to search after selecting different search options |
| | R3 | System should have multiple advance search options like price, rent, buy, etc. |
| | R4 | System should properly show the results of search to the user in appealing way. |

Table 9: Advance Search Feature Functional Requirement

8.1.8.8. Chat Feature

| ID | Requirement Description | |
|-----|-------------------------|--|
| FR8 | S.Req.ID | System Requirement |
| | R1 | A user interface is needed to direct the user to the chat feature like an icon in navbar and chat button in property description page. |
| | R2 | Users should be able to send and receive message. |
| | R3 | System should have a proper user interface so show the list of previous chats with other user and to start a new conversation |
| | R4 | System is required to support real-time messaging. |

Table 10: Chat feature Functional requirement

8.1.8.9. Report Feature

| ID | Requirement Description | |
|-----|-------------------------|--|
| | S.Req.ID | System Requirement |
| FR9 | R1 | A user interface is needed to allow users to report an inappropriate listing from the property description page, and a system is required to store the report in the database. |
| | R2 | Users should be select a reason for the report from a predefined list of options. |
| | R3 | System should send the reports made to the admin for the review of the report |

Table 11: Report feature functional requirement

8.1.8.10. Admin Review

| ID | Requirement Description | |
|------|-------------------------|---|
| FR10 | S.Req.ID | System Requirement |
| | R1 | Admins should be able to view a list of reports, and a system is required to retrieve the list of reports from the database and display it on the page. |
| | R2 | Admin Should be able to view the contents of reports. |
| | R3 | Admin should be able to take necessary actions to the reports like blacklisting the user, warning the user, etc. |
| | R4 | System should allow the admin to view the status of the report and change the status after reviewing. |

Table 12: Admin review functional requirement

8.1.8.11. Expiry date of Ad

| ID | Requirement Description | |
|------|-------------------------|---|
| | S.Req.ID | System Requirement |
| FR11 | R1 | System is required to set the maximum expiry date for each listing to 30 days from the time it was created. |
| | R2 | System is required to check for listings that have reached their expiry date and remove them from the application. |
| | R3 | A user interface is needed to allow users to repost their listings after the expiry date, and a system is required to verify the reposted listing before making it live on the application again. |
| | R4 | User interface is needed to display the expiry date of a listing to the user who created it. |

Table 13: Expiry time feature functional requirement

8.1.8.12. Ban User Feature

| ID | Requirement Description | |
|------|-------------------------|---|
| | S.Req.ID | System Requirement |
| FR12 | R1 | A user interface is needed to allow admins to initiate the banning process for admin. |
| | R2 | System is required to prevent banned users from accessing the application. |
| | R3 | Interface is needed to allow admins to view a list of banned users for admin. |
| | R4 | Admins should be able to unban a user. |

Table 14: Ban user functional requirement

8.1.8.13. Warn User Feature

| ID | Requirement Description | |
|----------|--|--|
| FR13 | Admin can warn the ban if user violates some conditions for the first time. | |
| S.Req.ID | System Requirement | |
| R1 | A user interface is needed to allow admins to initiate the warning process for admin. | |
| R2 | System is required to keep track of warning and show the list of warned users to admin. | |
| R3 | System is required to notify the user about the warning with appropriate message. | |
| R4 | System should automatically ban the user if the user receives the warning for the second time. | |

Table 15: Warn user functional requirement

8.1.9. Non-Functional Requirements

- **Performance requirements:** The room/apartment finder web application should have a fast response time and be able to handle a large number of users simultaneously. It should also be able to handle large amounts of data, such as property listings and user information.
- **Testing requirements:** The room/apartment finder web application should be thoroughly tested using black box and white box testing method before it is released to ensure that it is functioning as intended. Testing should include functional testing, performance testing, accuracy testing, etc
- **Free from Malware:** The room/apartment finder web application should be free from malware and viruses there should not be any hidden malware or viruses that will cause harm to the user.
- **Proper responsive:** The application should be fully web responsive so that the user using any mobile or desktop device can use the application with same friendliness in all state of window size.

8.1.10. Flow Chart

8.1.10.1. Flowchart for user as seller

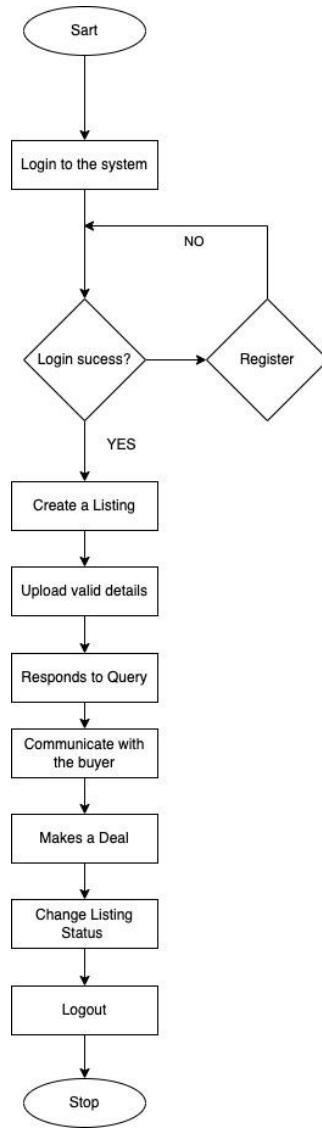


Figure 11: Flowchart for the user as seller

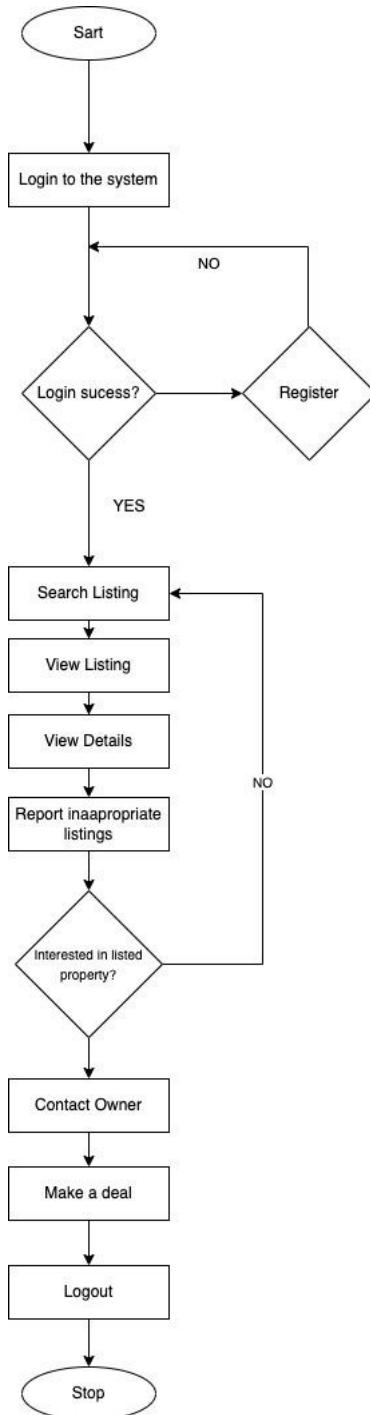
8.1.10.2. Flowchart for user as Buyer

Figure 12: Flowchart of user as buyer

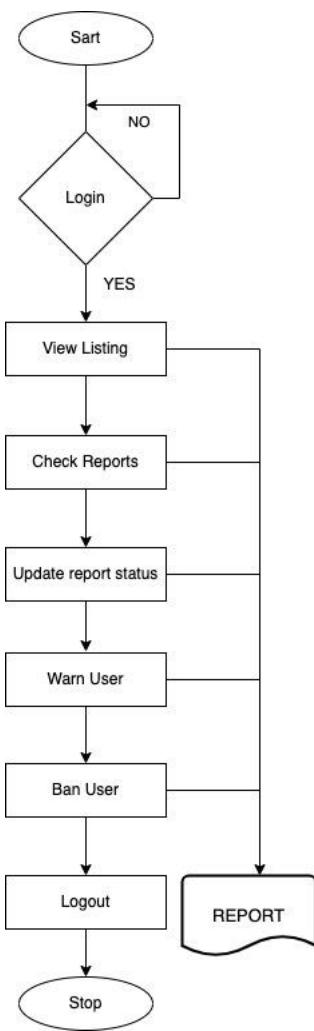
8.1.10.3. Flowchart for Admin

Figure 13: Flowchart for admin

8.1.11. User Interface Design

Some of the user interface designs are attached below.

8.1.11.1. Register Page

MY Room offers you Relaxing
Experience Finding
Rooms and Apartments

Rent Buy Home Loans Calculator Agencies Agents Properties

Register

After Registering you will be able to chat with other users and owners directly.
You can even list your property to lease or to sell.

First Name Last Name
Email address Phone Number
something@gmail.com 984997278323
Password Confirm Password

 Yes i want to receive important emails fro My Room like message notifications.
 I agree to all the terms and conditions.

Create Account

Already have an account with My Room? [Log in](#)

Figure 14: Register page UI

8.1.11.2. Login Page

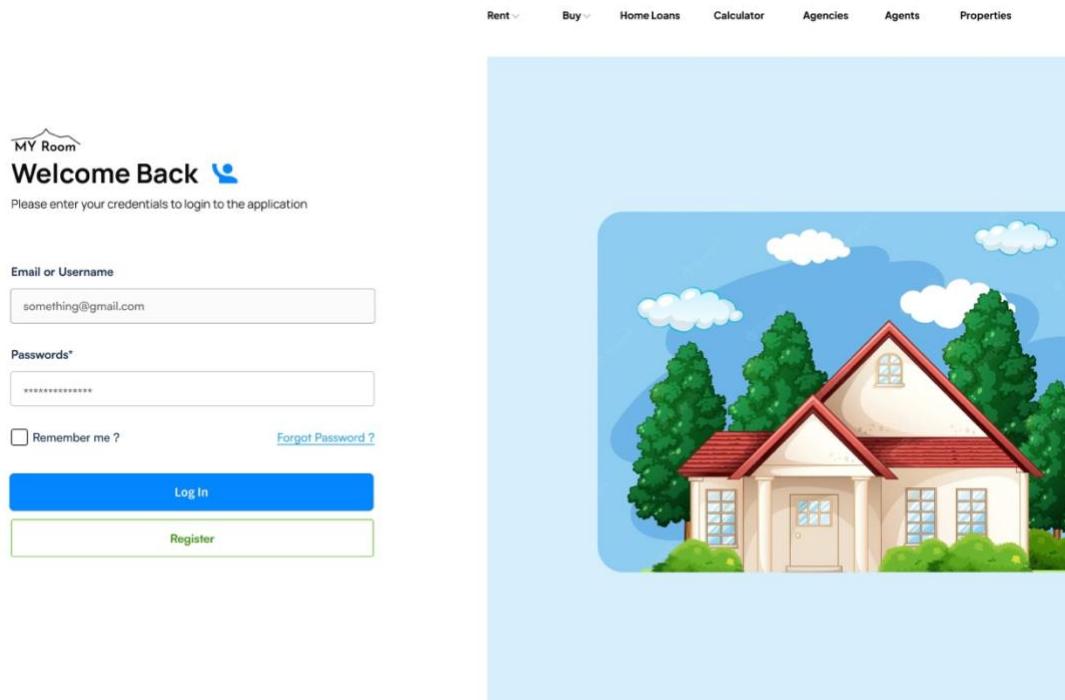


Figure 15: Login Page UI

8.1.11.3. Landing Page UI

WHAT WE OFFER

HOW IT WORKS ?

ENCOURAGEMENT TO CUSTOMERS

What Our Customers Say

Figure 16: Landing page UI

8.2. Resource Requirement

Software Requirements

- **VS Code**



Figure 17: Vs code Logo

(vscode, 2022)Visual Studio Code is a lightweight yet capable source code editor that runs on your desktop for Windows, macOS, and Linux. It comes with built-in JavaScript, TypeScript, and Node.js support, as well as a powerful ecosystem of extensions for other languages and runtimes (including C++, C#, Java, Python, PHP, Go, and.NET).

- **HTML**

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content (mozilla, 2022). Since this project is a web-based application HTML is a must.

- **CSS**

Css will be used in this project to give styling to the project. And to make the application responsive and as user friendly as possible.

- **Bootstrap**



Figure 18: (Bootstrap, 2022) Bootstrap Logo

Bootstrap is a free, open-source toolkit for developing websites and web applications. It was created to make it easier to build responsive, mobile-first designs using a set of pre-designed templates and styles. (Andrew Zola, 2022). Bootstrap will be used as the CSS framework to complete the project smoothly.

- **React JS**

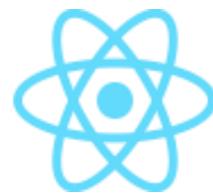


Figure 19: (Reactjs ORg, 2022)React Logo

"React.js is a tool developed by Facebook that allows developers to create interactive user interfaces and web applications in a fast and efficient manner. It uses JavaScript and reduces the amount of code needed compared to using vanilla JavaScript. It is open-source, which means it is available for anyone to use and modify. (Herbert, 2022) As this project will be a single page application react is going to be used

- **Node js**



Figure 20: (nodejs, 2022) Nodejs Logo

Node.js is a tool used by developers to build web applications that run on the server side, rather than in a user's browser. It is designed to handle large amounts of data and utilizes an event-driven, asynchronous model. Node.js was created in 2009 and has regularly updated versions, with the latest release being version 15.14 in April 2021. (Taha Sufiyan, 2022)

8.3. High Level Use Case Diagram

8.3.1. Actors of My room web application

| |
|---|
| Actors: User(General People) and Admin |
|---|

Table 16: Actors of the web application

8.3.2. Register

Preconditions: The user does not have an account with My room.

| |
|-----------------------|
| Name: Register |
| Actor: User |

Description: The user opens the registration page and fills the form then, they click the "register" button, and the application checks the phone and email address to ensure it is unique. If it is unique, the application creates a new user record with the provided information and logs the user in. If the email address is already in use, the application displays an error message.

Table 17: High level use case for register

8.3.3. Login

Preconditions: The user has an account with My Room.

| |
|---|
| Name: Login |
| Actor: User |
| Description: The user needs to go to the login page and enter their email address and password. Then, clicks the "login" button and the system will check if their email and password are correct. If they are, the system will let the user log in and take them to the homepage of My Room. If the email and password are not correct, the website will show an error message. |
| Exceptions: If the user does not have an account, user can click the "register" button to create a new account. |

Table 18: High Level Use case diagram for Login

8.3.4. Property Listing

| |
|--|
| Name: Property Listing |
| Actor: User |
| Description: The user needs to be logged in and click the "list property" button. They can then enter information about their property, like type, rent amount, etc. and further upload images or videos, and click "submit". new property listing with the provided information will be created and displayed on the website for other users to see. |
| Exceptions: If the user is not logged in, they will be redirected to login page upon clicking on list property button. |

Table 19: High Level Use case Diagram for property Listing

8.3.5. Map Location

| |
|--|
| Name: Map Location |
| Actor: User |
| Description: User needs to click on a property listing to view its details. The website will show the property description page, which includes all the details of the property. The user can see a map with the location of the property marked on it in location section. |

Table 20: High Level Use case diagram for map Location

8.3.6. Comment

| |
|--|
| Name: Comment |
| Actor: User |
| Description: User need to open the property description page of particular listing. They can then enter their comment in the provided text field and click "submit". The website will add the comment to the list of comments for the property and display it on the property description page. |
| Exceptions: If the user is not logged in, they will be redirected to login page after clicking the submit button. |

Table 21: High Level use case diagram for comment

8.3.7. Show Nearby schools/Hospitals

| |
|---|
| Name: Show Nearby Schools/Hospitals |
| Actor: User |
| Description: the user needs to click on a property listing. They can then click the "view nearby schools/hospitals" toggle button in the map view section, and the website will show a map with the location of the property and nearby schools and hospitals. |

Table 22: High Level Use case diagram for show nearby schools/hospitals

8.3.8. Advance Search

| |
|--|
| Name: Advance Search |
| Actor: User |
| Description: The user needs to click the "advance search" button then a form will be shown. They can then enter their search criteria and click the "search" button. The web application will look for properties that match the search criteria entered by the user and display a list of them on the search results page. If the user does not fill out the search form correctly or the criteria, they entered are not valid, the website will display an error message. |

Table 23: High Level Use case diagram for Advance search

8.3.9. Chat

| |
|--|
| Name: Chat |
| Actor: User |
| Description: The user clicks on the chat button of navbar or property description page. Then the application displays the chat window accordingly either list of people or specific person's chat box. The user can then send and receive the messages. |
| Exceptions: If the user is not logged in, they will be redirected to login page after clicking the chat button. |

Table 24: High Level Use Case Diagram for Chat Feature

8.3.10. Report

| |
|---|
| Name: Report |
| Actor: User |
| Description: The user must be logged in and visit the Property Description page. They can then click the "report" button, which will open a form for them to type in a reason for the report. After the user submits the form, the application will notify the admin about the report. |
| Exceptions: If the user is not logged in, they will be redirected to login page after clicking the chat button. |

Table 25: High Level Use Case Diagram for Report Feature

8.3.11. Report Review

| |
|---|
| Name: Report Review |
| Actor: Admin |
| Description: Admin must first log in and click the "review reports" button on the homepage. They will see a list of pending reports and can select one to review. The admin can then review the report and the associated property listing. They can decide to dismiss the report or take action on the listing (e.g., warn the user, ban the user, remove the listing). They must click "submit" to save their decision." |

Table 26: High Level Use Case Diagram for Report Review Feature

8.3.12. Ban User

| |
|--|
| Name: Ban User |
| Actor: Admin |
| Description: Admin must click the "ban user" button on the report review page. A confirmation message will appear and the admin must confirm to ban the user. The application will then ban the user and show a success message." |

Table 27: High Level Use Case Diagram for Ban user feature

8.3.13. Warn User

| |
|---|
| Name: Warn User |
| Actor: Admin |
| Description: Admin must click the "Warn user" button on the report review page. A confirmation message will appear and the admin must confirm to warn the user. The application will then warn the user and show a success message." |

Table 28: High Level Use Case Diagram for Warn user feature

8.4. Wireframes

8.4.1. Landing Page Wireframe

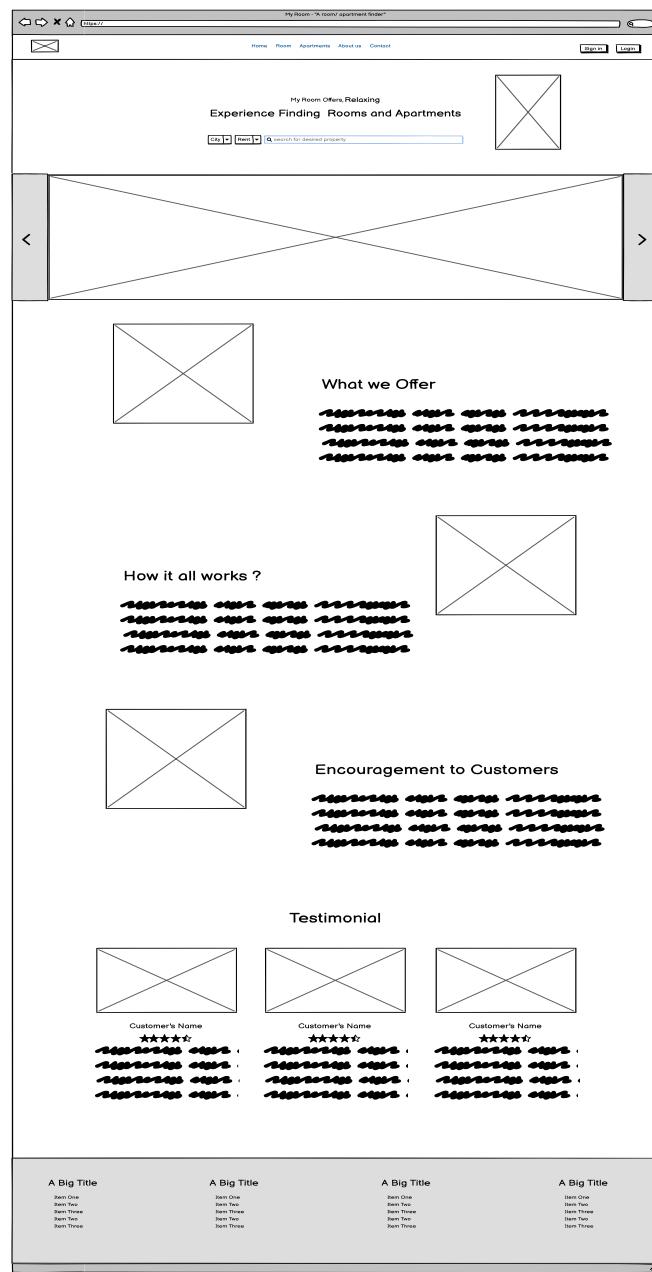


Figure 21: Wireframe for Landing Page

8.4.2. Login Page

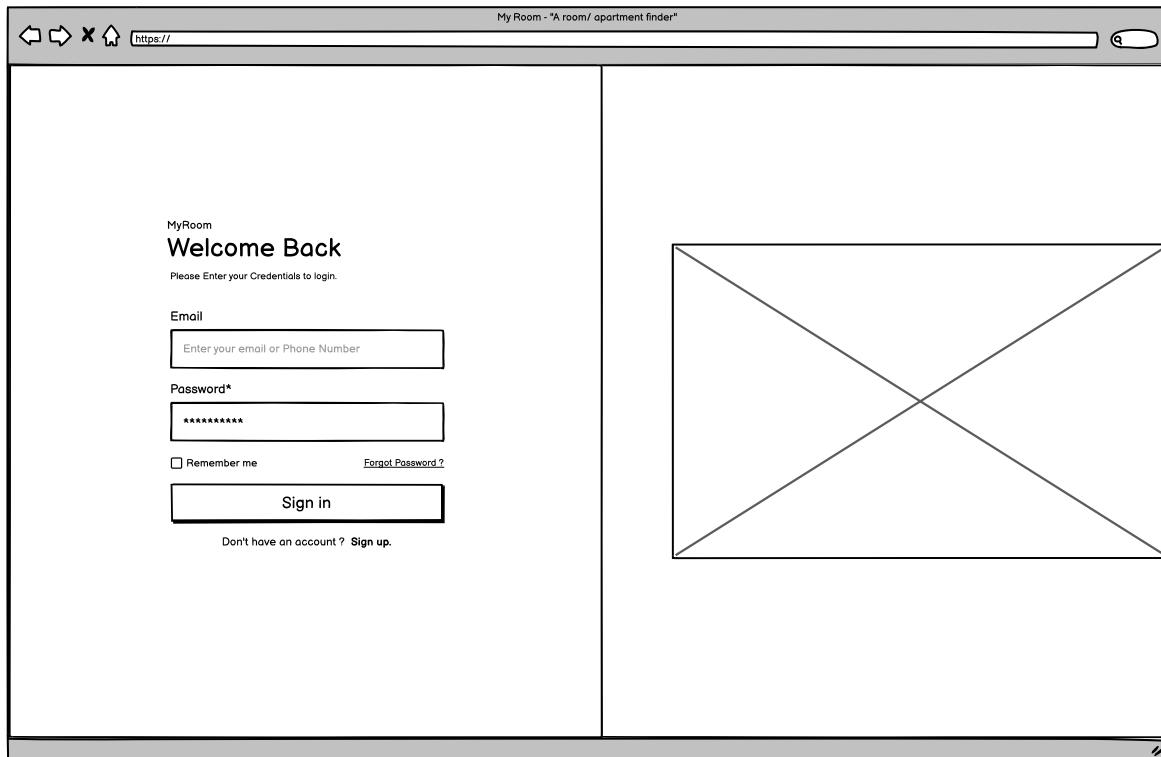


Figure 22: Wireframe for Login Page

8.4.3. Register Page

The wireframe shows a web browser window titled "My Room - 'A room/ apartment finder'". The left sidebar contains the text "My Room Offers, Relaxing Experience Finding Rooms and Apartments" and features a large square placeholder with a diagonal cross and some hand-drawn scribbles at the bottom.

The main content area is titled "Register". It includes a brief description: "After registering you will be able to chat with the users and talk to owner directly. And, you can even post your own property advertisement to lease or sell."

The registration form consists of several input fields:

- First Name (text input)
- Last Name (text input)
- Phone Number (text input)
- Email address (text input, containing "something@gmail.com")
- Password (text input)
- Confirm Password (text input, containing "something@gmail.com")

Below the form are two checkboxes:

- Yes I want to receive important emails from MyRoom like Message Notification.
- I agree to all the Terms & Conditions.

A large "Create an Account" button is centered below the checkboxes. At the bottom of the page, there is a link "Already have an account? [Log in](#)".

Figure 23: Wireframe for Register Page

8.4.4. Home Page

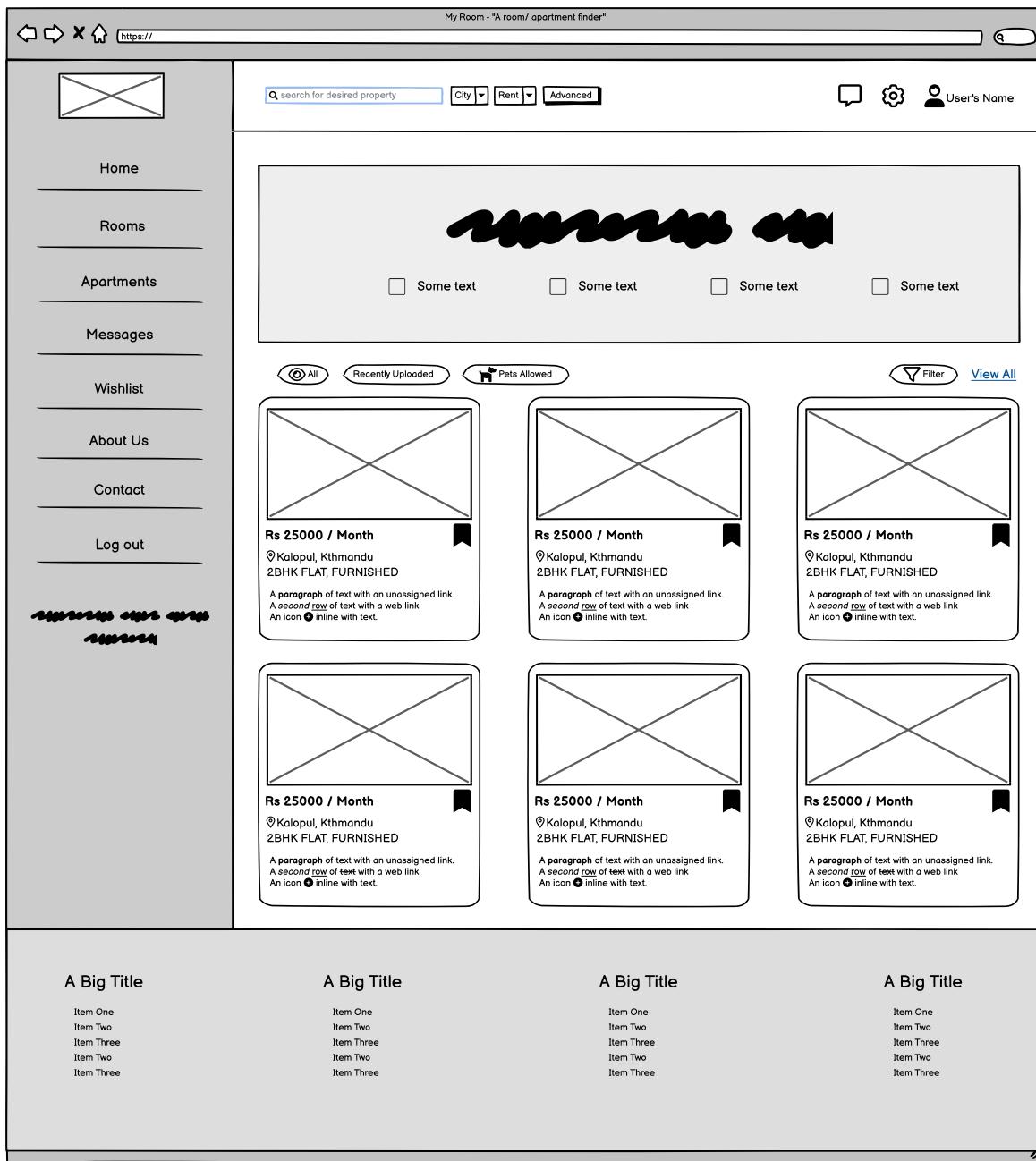


Figure 24: Wireframe for Homepage

8.4.5. Property Description Page

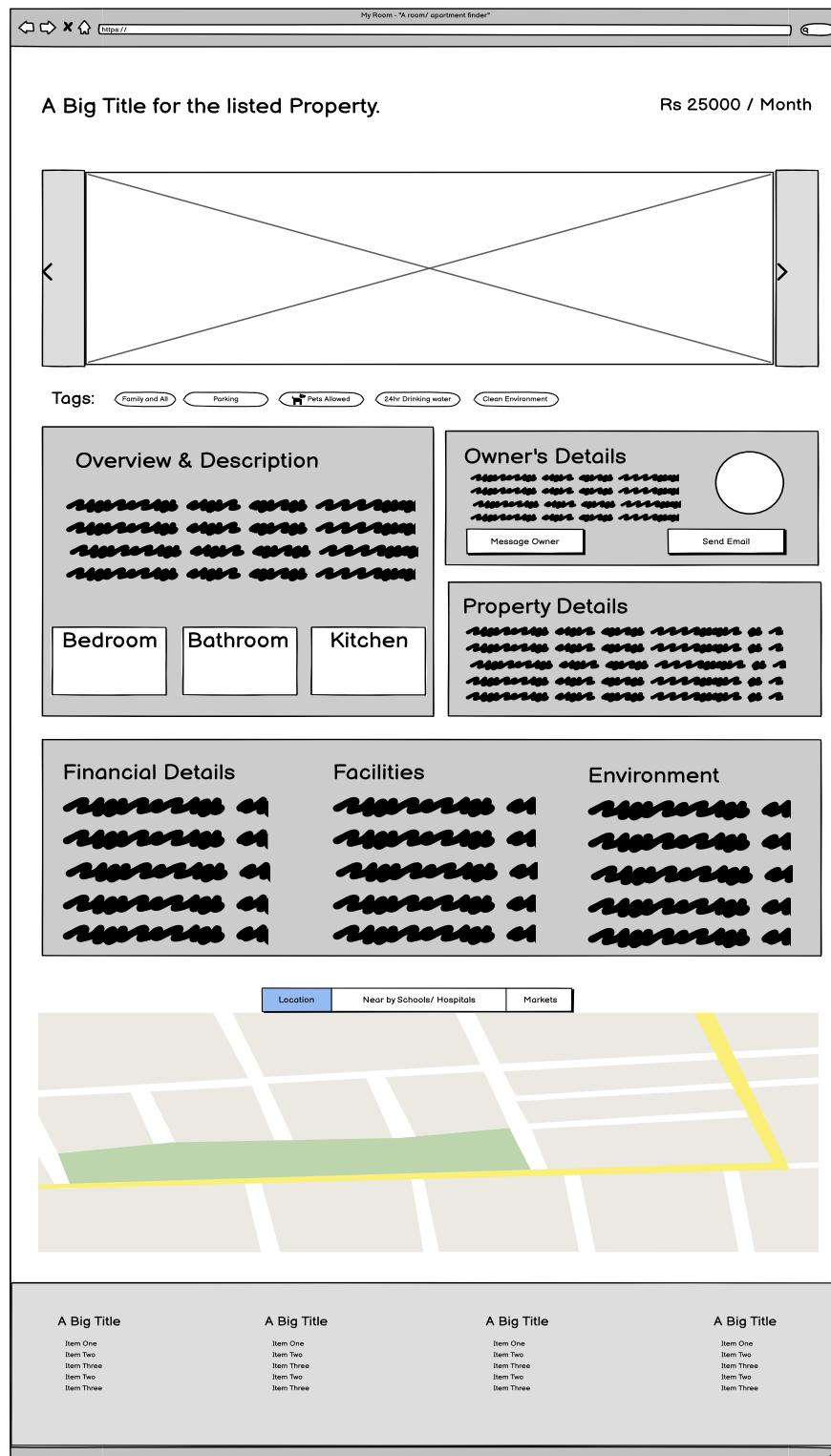


Figure 25: Wireframe for Property Description Page

8.4.6. Nav-link Room Page

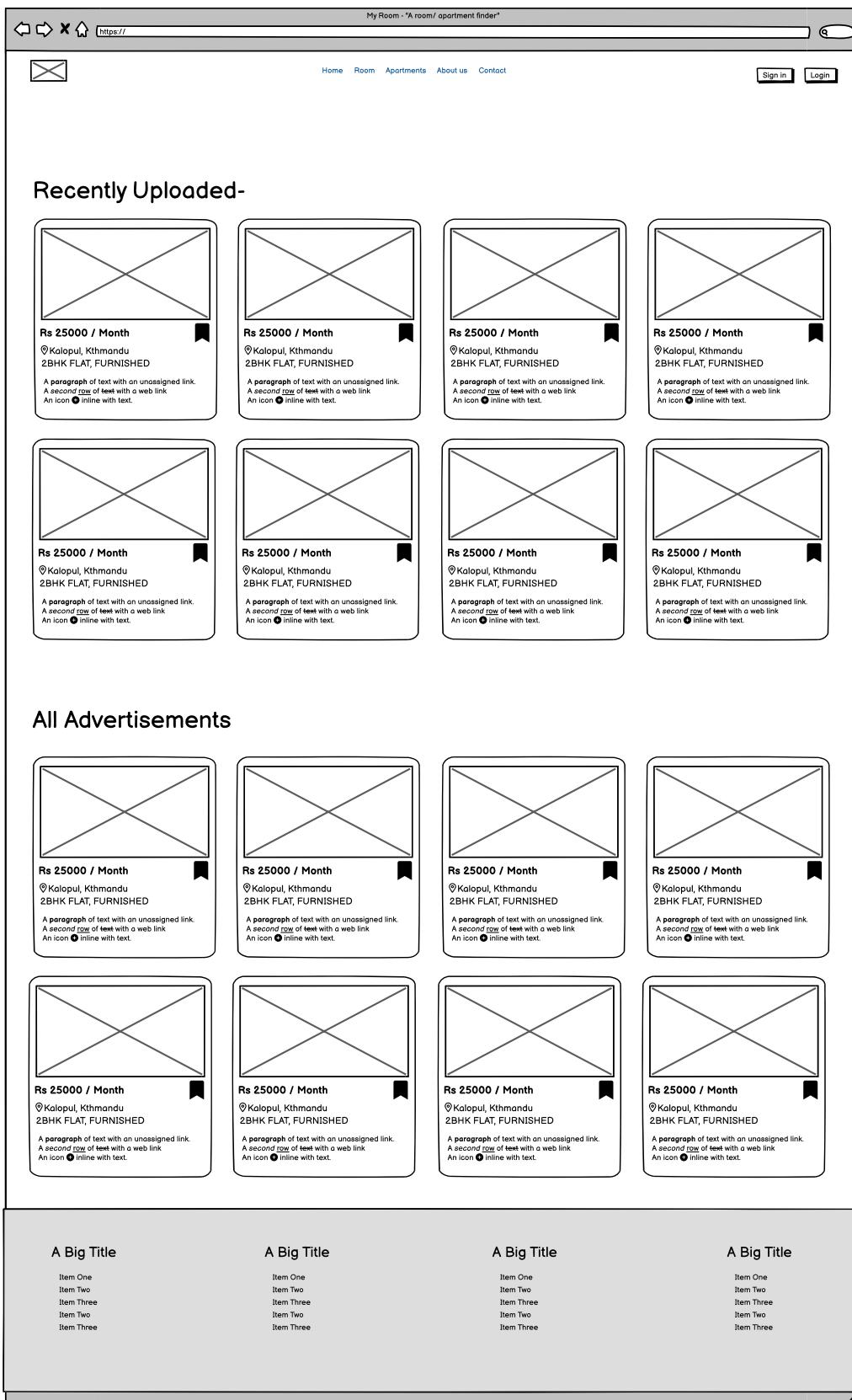


Figure 26: Wireframe for sidebar's Room menu item

8.4.7. Nav-link Room Page

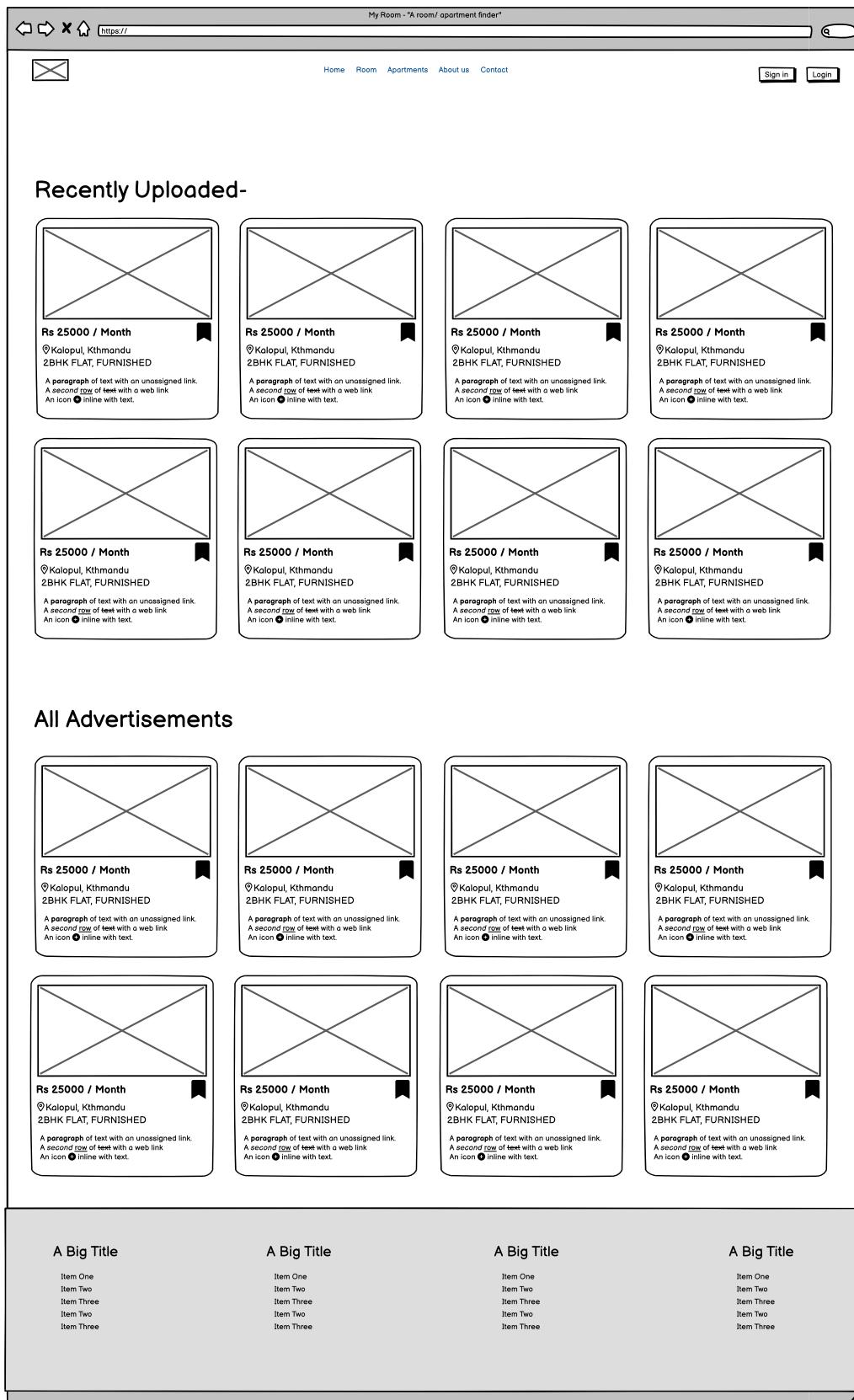


Figure 27: Wireframe for Sidebar's Apartments menu item

8.5. Work Break Down Structure

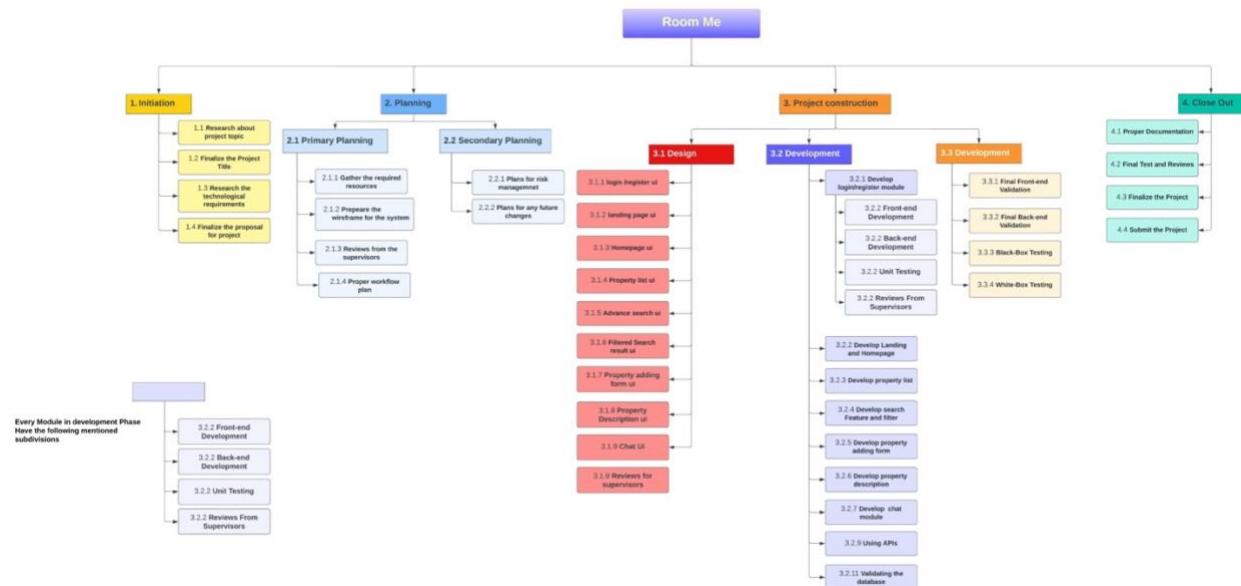


Figure 28: Work Breakdown Structure

8.6. Gantt Chart

There are no changes to the Gantt chart so it is same as in proposal for now.

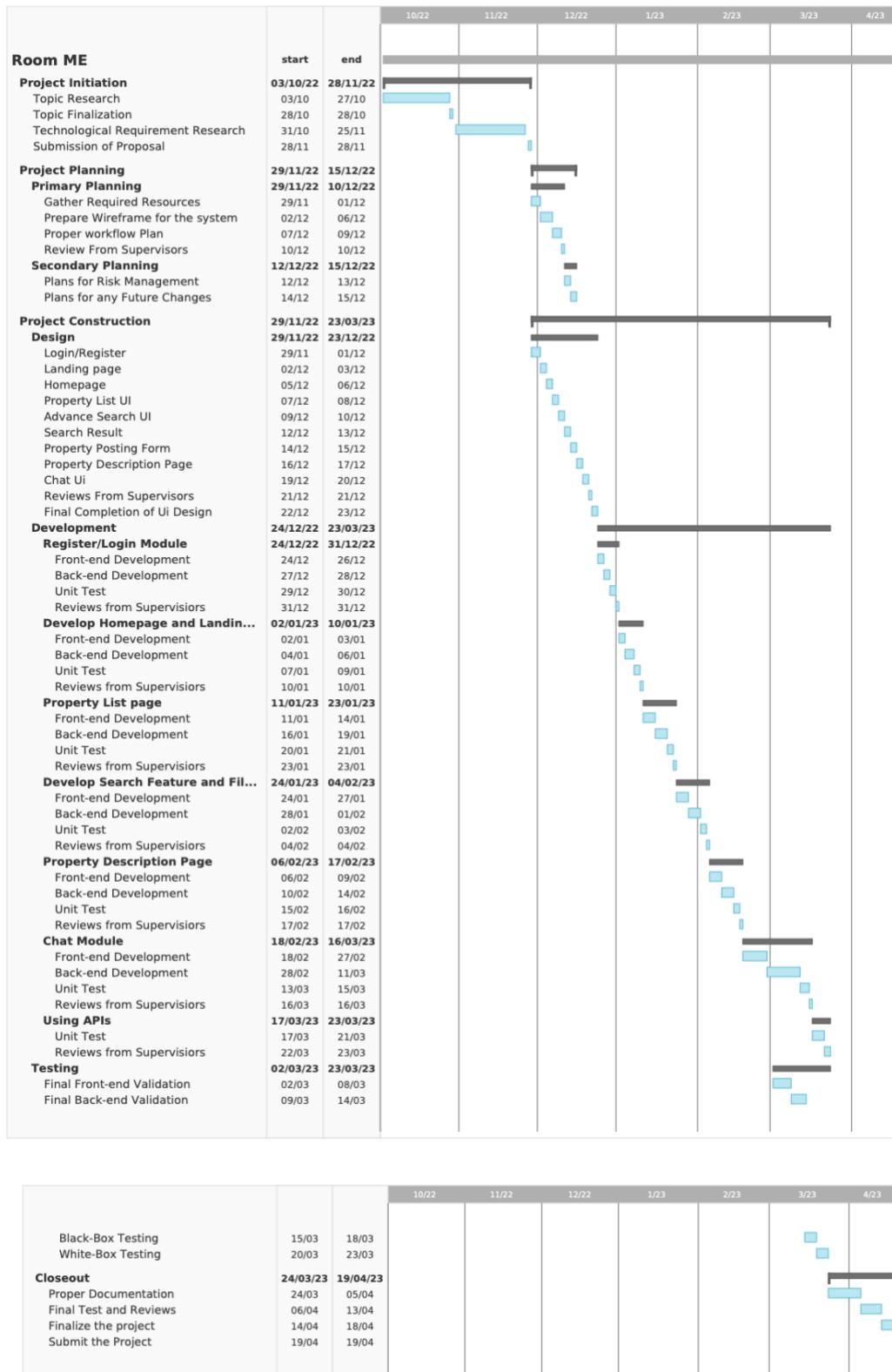


Figure 29: Gantt Chart

8.7. Milestones

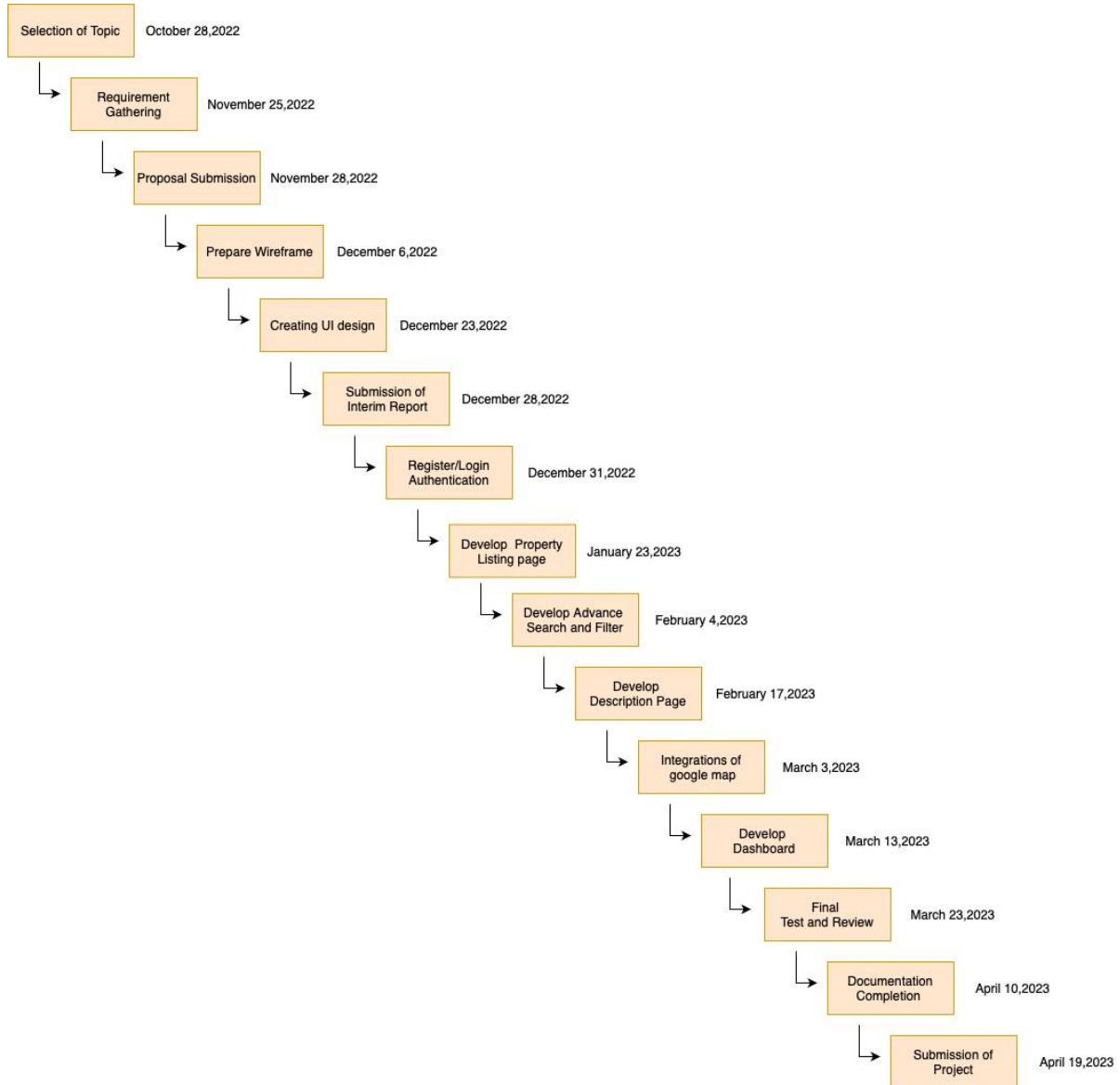


Figure 30: Milestones