TITLE OF THE PROJECT

BD RENT

Submitted By

Mejbah Uddin Tamim

ID: 192-16-453

Department of Computing and Information System (CIS)

Daffodil International University

Supervised By

Syed Tangim Pasha

Lecturer

Department of Computing and Information System (CIS)

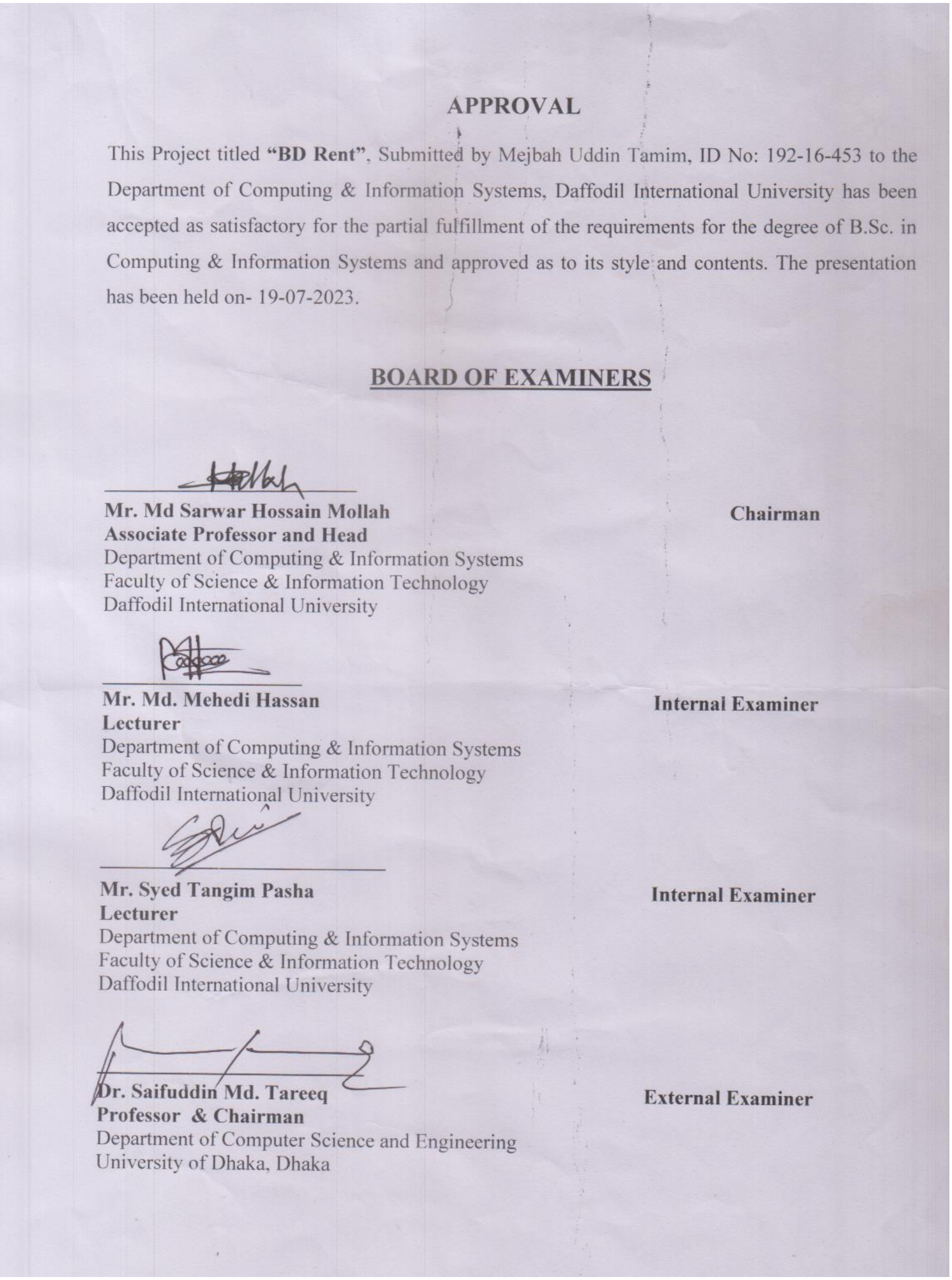
Daffodil International University

Course Code: CIS499

Department of Computing and Information System (CIS)

Submission Date: 19 July 2023

DAFFODIL INTERNATIONAL UNIVERSITY



APPROVAL

This Project titled **"BD Rent"**, Submitted by Mejbah Uddin Tamim, ID No: 192-16-453 to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on- 19-07-2023.

BOARD OF EXAMINERS

Mr. Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Chairman

Mr. Md. Mehedi Hassan
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

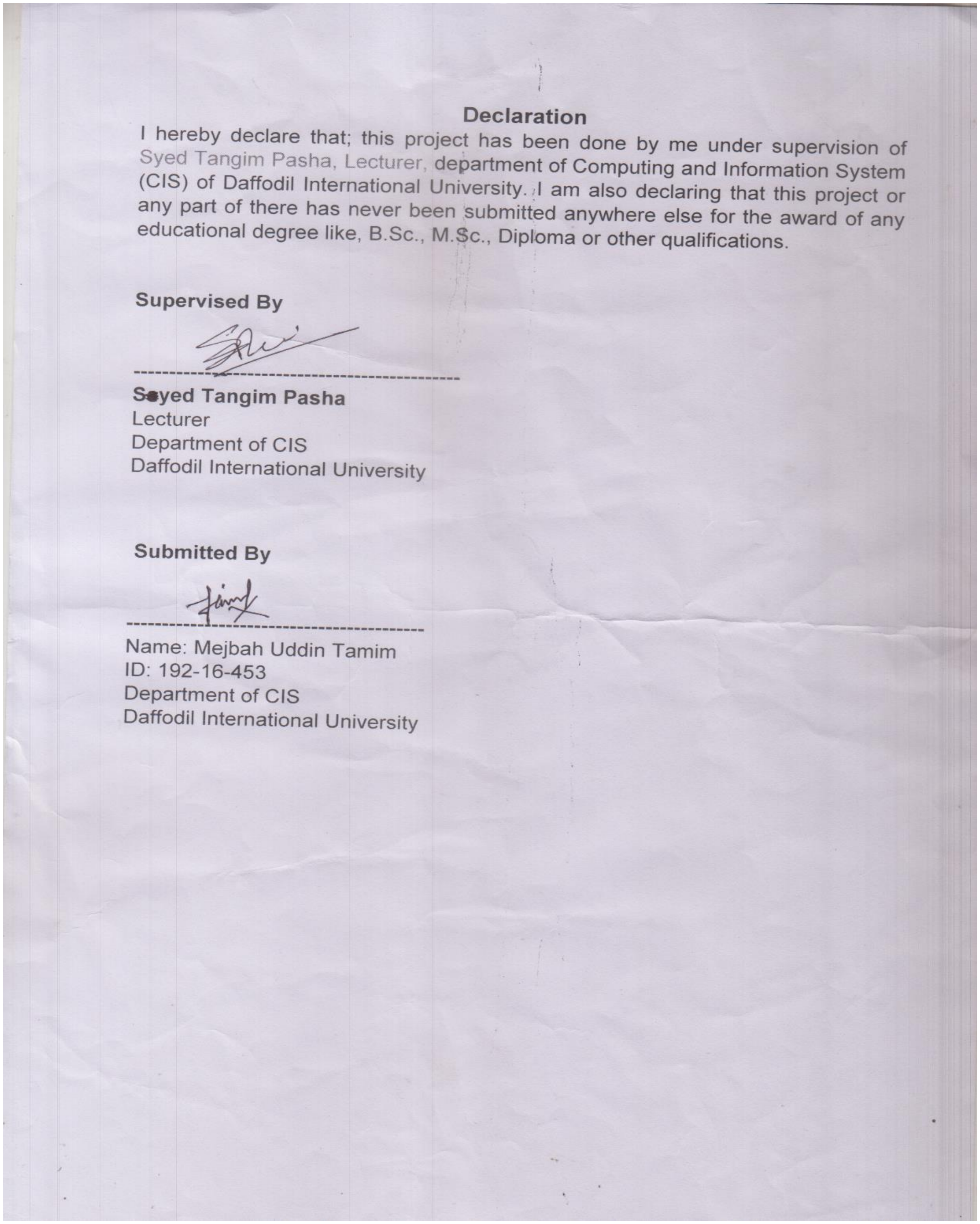
Internal Examiner

Mr. Syed Tangim Pasha
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Saifuddin Md. Tareeq
Professor & Chairman
Department of Computer Science and Engineering
University of Dhaka, Dhaka

External Examiner



Declaration

I hereby declare that; this project has been done by me under supervision of Syed Tangim Pasha, Lecturer, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By

Syed Tangim Pasha
Lecturer
Department of CIS
Daffodil International University

Submitted By

Name: Mejbah Uddin Tamim
ID: 192-16-453
Department of CIS
Daffodil International University

Acknowledgment

First, I would like to thank Allah for giving me a career in study at Daffodil International University and efficiently completing my last year's project within a specific time limit. After that, I am extremely pleased to my lovely parents and family members for supporting me throughout my university journey. Then I want to give thanks to my university for given me chance to do real world project. During the entire project development and documentation **Syed Tangim Pasha** Sir, who is my academic supervisor. Helped me and guided me to complete the project within the time frame.

Dedication

I wish to dedicate this BD Rent project to my lovely parents and my family members. Insteadof the support of my parents, it would have been very difficult for me to continue on this long journey.

Abstract

The BD Rent project is a Housing booking and management solution that is focused on people who looking for house rent in Bangladesh also people who wants to rent their house to people.
This is web-based application that let people use easily find their house based on their requirements. There are many features and batter user experience that make application more interactive application.

Table of Contents

List of Figures

List of Tables

Chapter 1 – Introduction

1.1 Introduction

The demand for house booking and rental websites has skyrocketed in recent years, as travelers seek out unique and personalized experiences in their accommodations. The convenience of browsing and booking online is a major drawcard, allowing guests to find the perfect property from virtually anywhere in the world. With endless options available at their fingertips, guests can tailor their search to fit any criteria, from location and budget to amenities and style. In Bangladesh housing is a big problem. Day by day it's getting harder for people to find a good house within their requirements.

Also, for the house owners they need a platform where they can easily list their house and manage their house listing.

In summary, the demand for house booking and rental websites is not slowing down anytime soon. These platforms offer a level of convenience, flexibility, and personalization that traditional accommodation options simply cannot match. As travelers continue to seek out unique and memorable experiences, house booking and rental websites will undoubtedly remain a popular choice for years to come.

There are many companies that launched House booking type websites in Bangladesh but most of them lacks good functionality, design, user experience. While researching existent solutions and users research, I have noted down all the things that are missing and I made sure to implement those in BD Rent project.

1.2 Document Contents

The following chapters will be described in this document to define the activities of the project.

Chapter 1: Introduction

Briefly describe the industry scope and provide some introductory important key points

Chapter 2: Initial Study

Here, we summarize the background of the proposed system, identify problem areas, and discover the possible solution by mentioning the notable reasons.

Chapter 3: Literature Review

It contains a comprehensive analysis of the problem area, and solutions, an assessment of remaining solutions, and a final endorsement.

Chapter 4: Methodology

Here, I presented a few development procedures that can be applied, describe the necessity of using them, and choose the suitable methodology for the project.

Chapter 5: Planning

Defined the project and test planning demonstrated by table format including work breakdown structure, Gantt chart, time-boxing, etc. for the proposed system.

Chapter 6: Feasibility

Covered different sorts of feasibility studies and demonstrate a cost-benefit analysis of the proposed system in this section.

Chapter 7: Foundation

Discussed the identification of the problem area, a list of overall necessities recommended solutions, and justifications will be recorded.

Chapter 8: Exploration

Demonstrated the use case and activity model of the old system as well as the new system, the requirement catalog, and also the new system prototype.

Chapter 9: Engineering

Provided UML diagrams for representing the business logic and consumer behavioral model of the project.

Chapter 10: Deployment

Discussed and display some coding illustrations of the core modules, development problem breakdown, and development priority.

Chapter 11: Testing

Demonstrated some functional and non-functional testing including unit, integration, module, security, accessibility testing, etc.

Chapter 12: Implementation

All varieties of implementation procedures are covered in this section and the training session is enclosed according to the user also.

Chapter 13: Critical Appraisal and Evaluation

Here, all appraisal and critical evaluations will be enclosed in a sequential way.

Chapter 14: Lessons Learned

In this section, mainly the difficulties that I have challenged and what I have acquired during the project development have been exposed.

Chapter 15: Conclusion

Here, the project summary is described along with the success, and the experience that I achieve throughout the project.

Chapter 2 – Initial Study

2.1 Background of the project

BD Rent is a house booking and renting websites that provides service people who are looking for house rent or people who are looking to rent their house. In a rapidly evolving world where urbanization is on the rise, the need for efficient and reliable housing solutions has become more critical than ever before. In this context, BD Rent emerges as the pioneering project that caters to the growing demand for a seamless house rental experience. With a laser-sharp focus on connecting homeowners with potential tenants, BD Rent revolutionizes the way people rent their houses, creating a paradigm shift in the rental market.

BD Rent stands at the forefront of the house rental revolution, satisfying the escalating demand for seamless, efficient, and reliable housing solutions. By harnessing the power of advanced technology, BD Rent creates an ecosystem that connects homeowners and tenants, ensuring a win-win situation for both parties. As the world embraces a more dynamic and transient lifestyle, BD Rent rises to the occasion, facilitating the evolution of the rental market and redefining the way people find and secure their ideal rental properties.

2.2 Problem Area

While searching for house rent in Bangladesh. I found different websites that are offering house rent service in Bangladesh. But from user perspective I found different types of problem and lacking which causing problem for users. I will briefly describe those problems as follows:

- Lack of mobile optimization
- Absence of proper user experience
- Lack of user and host documents

Lack of mobile optimization: Mobile optimization is paramount for a house booking website. As the world becomes increasingly mobile-centric, catering to the needs of mobile users is no longer an option but a necessity. While doing research I saw lack of mobile optimization. I make sure my project has better mobile optimization than my competitor website.

Absence of proper user experience: For users and host both need a optimized experience in website. It's important for users to find their home based on the requirement and easily book that. On the other hand, host needs to easily host their house and manage everything easily. My websites have complex user experience for both users and host.

Lack of user and host documents: In order to make any website secure, website needs to take proper authorize documents from users like NID, passport etc. While doing the research I saw many websites don't need to upload that many documents which can lead to huge security issue. Because anyone can create fake account and do any harm. In BDRENT project I made sure every users must upload documents to make user every users is real and are not fake.

2.3 Possible Solution

The foundation of a successful house booking website lies in its user interface design. A clean and intuitive layout with a visually appealing design can captivate users' attention and enhance their browsing experience. Implementing a responsive design that adapts seamlessly to various devices ensures accessibility to a wider user base. Intuitive navigation menus, interactive maps, and high-quality property images are indispensable in helping users quickly locate and evaluate potential houses.

To build trust and instill confidence in users, robust security measures must be in place throughout the booking process. Employing industry-standard encryption protocols to safeguard sensitive user information, such as payment details and personal data, ensures secure transactions. Implementing a reliable and verified payment gateway allows users to complete transactions seamlessly while protecting their financial information.

By focusing on these key areas—user interface design, advanced search functionality, secure transactions, and effective communication channels—a house booking website can provide a comprehensive solution to streamline the house booking experience. Emphasizing user-centric design and implementing robust technical features ensures a hassle-free and efficient process for property owners and tenants alike. With a commitment to continuous improvement and an unwavering dedication to customer satisfaction, the house booking website can become a trusted and preferred platform for all accommodation needs.

Chapter 3 – Literature Review

In the research process literature review is a crucial step. Providing a comprehensive summary that serves as a background and context for a primary research project. Its purpose is to objectively evaluate, describe, and clarify previous research, thereby establishing a theoretical foundation and offering an overview of current knowledge. By conducting a literature review, researchers can identify related philosophies, approaches, and gaps in existing research, ultimately aiding them in refining the description of their own investigation.

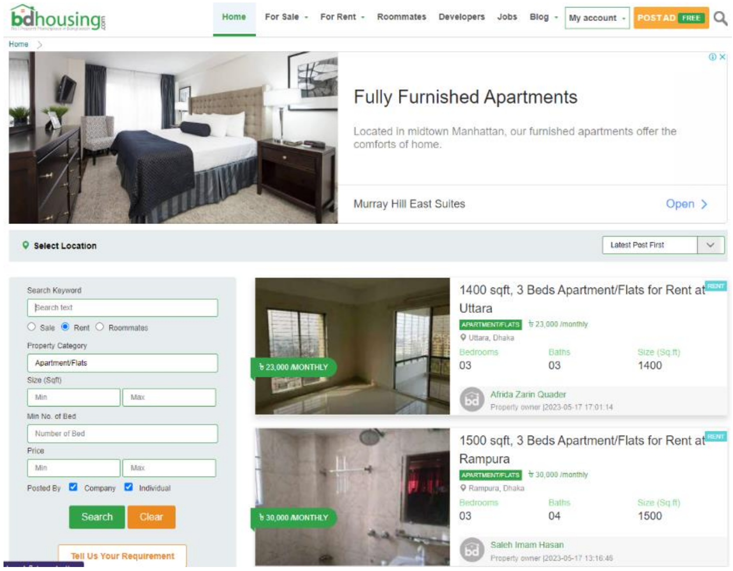
3.1 Discussion on the Problem Domain

In an era of increasing digitalization, house booking websites have revolutionized the way people search for and secure accommodations. These platforms connect homeowners and renters, streamlining the process and providing convenience and accessibility. However, behind the user-friendly interfaces and extensive property listings lie several significant challenges that house booking websites must navigate. In this article, we will explore some of the most critical problem domains faced by these platforms and discuss potential solutions to enhance user experiences and overall satisfaction.

- User Interface and user interaction
- Trust and Verification
- Quality and consistency of listing
- Pricing Transparency
- Security and Data Privacy

3.2 Comparison of the leading solutions

Bdhousing



Fully Furnished Apartments

Located in midtown Manhattan, our furnished apartments offer the comforts of home.

Murray Hill East Suites

৳ 23,000 /MONTHLY

1400 sqft, 3 Beds Apartment/Flats for Rent at Uttara

APARTMENT/FLATS ৳ 23,000 /monthly

Uttara, Dhaka

Bedrooms	Baths	Size (Sq.ft)
03	03	1400

Afrida Zarin Quader
Property owner |2023-05-17 17:01:14

৳ 30,000 /MONTHLY

1500 sqft, 3 Beds Apartment/Flats for Rent at Rampura

APARTMENT/FLATS ৳ 30,000 /monthly

Rampura, Dhaka

Bedrooms	Baths	Size (Sq.ft)
03	04	1500

Saleh Imam Hasan
Property owner |2023-05-17 13:16:46

Figure 1: Overview of the Bdhousing

Best Features

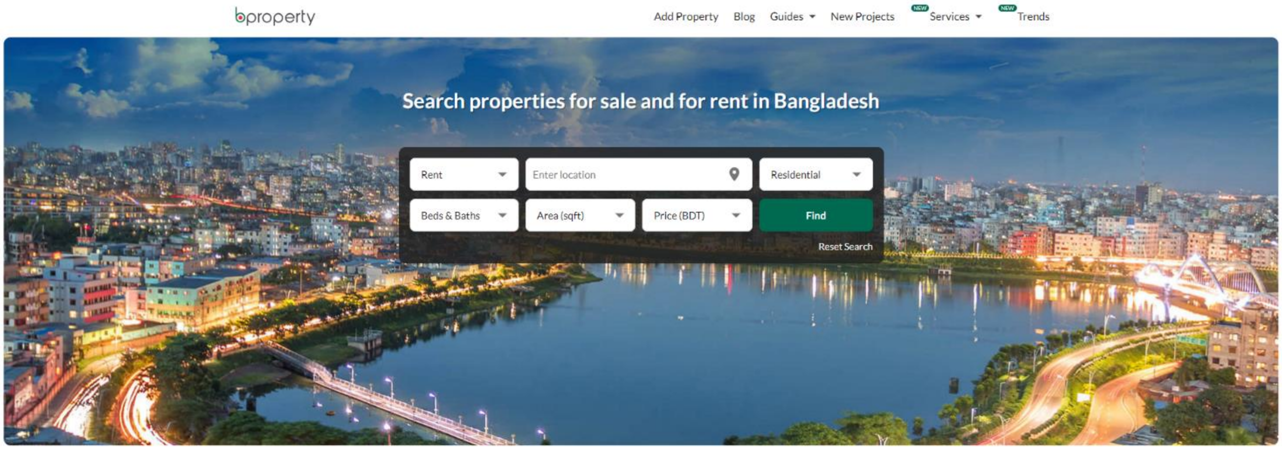
- ✓ Different types of result filter options
- ✓ Good amount of house listing
- ✓ Different types of house options for user

Limitations

- Website design is not modern
- Lacking in mobile responsiveness
- Too many Ads on the website
- Complex booking process

- Leak of support system
- Firebase authentication
- No AI features

Bproperty



Search properties for sale and for rent in Bangladesh

Figure 2: Overview of the Bproperty

Best Features

- ✓ For computer version website design is modern
- ✓ Good amount of house listing
- ✓ Different types of house options for user

Limitations

- Lack of security
- Complex house booking system
- Admin panel for house host is complex
- Firebase authentication

3.3 Recommended Approach

According to the analysis of leading solutions, the remaining systems have some defects which lose the user experience as well as public demand. There are some features and thoughts that need to the proposed system that makes it a more user-interactive application. Below I have included in the proposed system listed here:

- Well design website for PC version
- It should be mobile responsive
- Proper documentations collection for security purpose
- House booking process should be easy for users
- Easy to manage the house listing for host
- Admin can check all the activity in the website
- Proper data restore

Also, while designing the website, I will focus cognitive psychology. Cognitive psychology studies individual-level mental processes such as information processing, attention, language use, memory, perception, problem solving, decision-making, and thinking.

Attention: Attention is the most important process in BD Rent project. That's why the interface needs to be user-friendly. So that users can check all the house and vacation listing.

Learning: Users can learn through many cognitive processes, such as memory, thought, and perception. Users need to learn by browsing the website. Websites need to have proper icons and navigation so that users can learn while browsing.

Language: People can learn languages much more easily earlier in life through fluid intelligence. As the website targets mostly Bangladeshi people. That's why websites need to have simple and easy-to-read language.

Memory: Memory is also one important process which is able to store, encode and retrieve information. Our website needs to have information that is beneficial for views. Avoid having unnecessary information or function.

Cognitive Framework: The cognitive framework links perception, thinking, and behavior. As a result, cognitive therapy is a blend of psychological, cognitive, and behavioral correction.

Mental Model: A mental model is based on belief, not facts: that is, it's a model of what users know (or think they know) about a system such as your website. In BD Rent, there will be navigations and buttons so the user can interact with the system and know what to do next.

Gulfs of execution and evaluation: The gulf of evaluation is the degree to which the system/artifact provide representations that can be directly perceived and interpreted in terms of the expectations and intentions of the user.

In the interaction design of our website will provide feedback on every user action. Information Processing: Information Processing is. the change in a state of information. It allows us to split information down into steps that can be easily read and understood by the human brain.

Chapter 4 – Methodology

4.1 What to use

The role of software development methodologies in the software development process cannot be underestimated. Various well-known software development methodologies exist, including the waterfall model, prototype model, agile software development, rapid application development (RAD), dynamic systems development method (DSDM), spiral model, V-model, and incremental model (Lombardi, 2022). In this discussion, I will focus on exploring the advantages and disadvantages of three specific models: the waterfall model, dynamic systems development method (DSDM), and rapid application development (RAD) (alliancesoftware, 2017).

Waterfall Model

This methodology is a widely recognized and visionary approach to project management, which divides the project's lifecycle into a series of distinct phases. In this model, each phase must be completed in a sequential flow, producing clear execution outcomes that serve as input for the subsequent phase. This software development technique is commonly employed by numerous software companies to handle large-scale IT projects for their clients, as it allows for a disciplined and efficient allocation of budget and resources while ensuring the successful achievement of project objectives (Sherman, 2015).

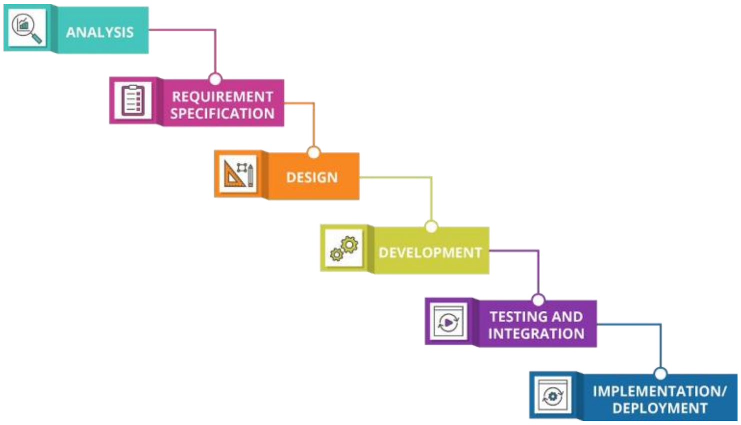


Figure 3: Waterfall Model

Advantages of the Waterfall Model

- One of major advantage is it simplicity and ease of use.
- It provides greater control over the project scope, budget and timeline.
- It provides clear structure of documentations. Each phase has it specific deliverables. documentation can be created at each stage, ensuring that there is a thorough record of the development process

Disadvantages of the Waterfall Model

- Only appropriate for minor kinds of projects but not well-matched for large-scale or maintenance-type projects.
- Quite tough to modify anything because changes are not accepted and are restricted back to the previous stage.
- Does not integrate any approach for error correction because it assumes that no error occurs by developers during any phase which does not happen.
- Avoid process overlapping, which is important for increasing efficiency and reducing cost

Rapid Application Development (RAD)

RAD seeks to reduce the time it takes to develop software by using a variety of techniques, including prototyping, iterative development, and timeboxing. This often involves the use of drag-and-drop tools that enable developers to quickly create mockups of application screens and interactions. This approach helps developers get feedback from stakeholders early in the process, which enables them to refine designs and requirements before committing significant resources to development

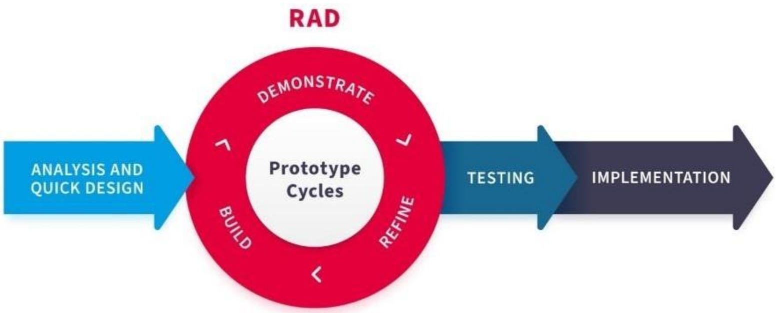


Figure 4: Rapid Application Development

Advantages of the Rapid Application Development (RAD)

- Adaptable to requirements modification at any time
- Well-structured for small projects with a fewer workforce
- Minimize overall project risk and development time
- Take half time to develop a new feature and increases the reusability of features.
- Encourage consumers to provide feedback for improvement

Disadvantages of the Rapid Application Development (RAD)

- Dependent on strong team collaboration
- Unable to maintain with large teams
- Required highly expert developers which increase the cost
- Needs user involvement throughout the product's lifespan
- Only appropriate for short-term jobs.

Dynamic Systems Development Method (DSDM)

Dynamic Systems Development Method (DSDM) is used for building a high-quality project according to business needs with limited time and budget for small software developments up to large-scale business process change by following agility style development. (Agile Business Consortium) This methodology follows a rule of thumb that 80% of the solution can be delivered for 20% of the total effort so that it provides the working software step-by-step portion of the total solution within a short time and then develop iteratively in a manner of user feedback.

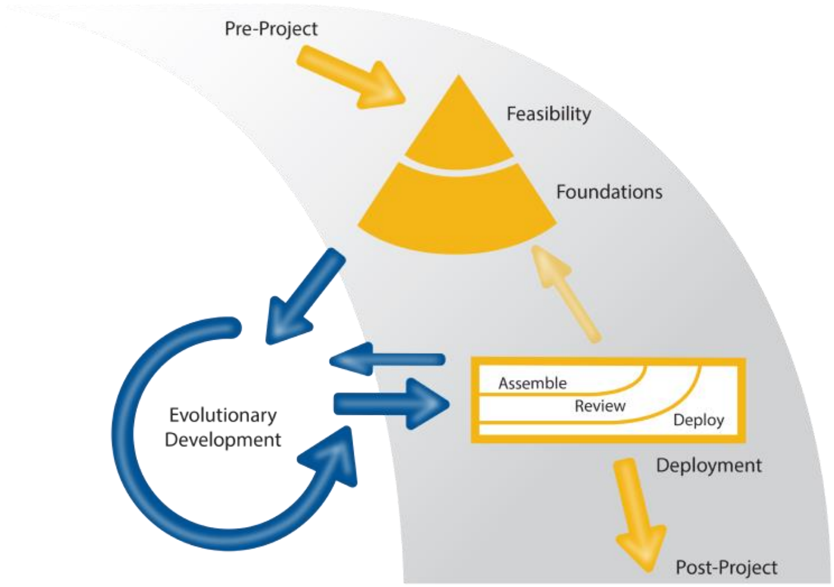


Figure 5: Dynamic Systems Development Method

Advantages of the Dynamic Systems Development Method (DSDM)

- Provides a high level of customer satisfaction by shorter development time
- Maintain product quality within a more predictable schedule and accurate project estimation
- Stakeholders can comprehend the latest developments that occur within the project because of user involvement
- Promote frequent feedback and suggestions for critical evaluation of the solution
- The risk of constructing the wrong solution is greatly controlled

Disadvantages of the Dynamic Systems Development Method (DSDM)

- Small businesses may be facing challenges due to the high cost of the installation process
- Only senior programmers play a role at the time of decision-making process
- Need a well-organized approach to project management, or it will collapse
- Strong involvement and commitment of the consumers and stakeholders
- Must allocate the schedule of the project delivery for determining how much time each part

Choosing Methodology

Which project management approach is appropriate will depend on several factors.I have made a decision to use the DSDM approach for the BD Rent project after taking into consideration the abovementioned factors because this methodology consistently produces high- quality products and delivers them on schedule. Any firm or business organization can adopt the DSDM technique due to its flexibility and if a client or customer decides that a modification is required for the project's development, they are free to change their requirements. Enhancing stakeholder and user involvement throughout the lifecycle ensures that high-quality product is accomplished on time.

4.2 Why to use

In the realm of project management, the Waterfall model stands as a beacon of standardized, structured, and organized work methods, significantly enhancing the likelihood of project success within the confines of limited time and budget. By embracing this model, organizations can harness its repeated successful aspects and leverage a continuous improvement process, resulting in reduced time requirements and elevated final product quality. While selecting the most suitable methodology for a project can be daunting, the adoption of the Waterfall model greatly bolsters project accomplishment and ensures the delivery of desired outcomes.

- ✓ Standardized and Structured Work Methods
- ✓ Organization and Time Management
- ✓ Achievement of Project Objectives
- ✓ Enhanced Product Quality

4.3 Sections of Methodology

The DSDM approach focus to deliver the most important business needs instead of addressing the less important features by following the iterative and incremental process. Throughout the development of the solution, DSDM enables the business representatives to provide feedback and request changes from its iterative nature. Six stages of the DSDM Lifecycle are proposed to guide the BD Rent project from conception to execution. The stages of DSDM are as follows:

Feasibility: The DSDM journey begins with a comprehensive assessment of project feasibility. During this stage, project teams analyze the viability of the proposed solution, considering factors such as time, budget, available resources, and the project's alignment with organizational goals. This critical evaluation sets the foundation for informed decision-making, ensuring that projects move forward with a clear understanding of their potential benefits and challenges.

Business Study: Once the project's feasibility is established, the DSDM approach shifts focus to understanding the business needs and requirements. Through close collaboration with stakeholders, project teams dive deep into gathering, analyzing, and documenting the project's scope, objectives, and desired outcomes. This stage emphasizes the importance of aligning the project with the organization's strategic goals and identifying key success factors that will shape the project's direction.

Functional Model Iteration: With a solid understanding of the business needs, the DSDM model enters the functional model iteration stage. Here, iterative development cycles take place, where prototypes and working models are created to validate requirements and functionalities. This iterative approach allows for continuous feedback and refinement, ensuring that the evolving solution aligns closely with stakeholder expectations and remains adaptable to changing requirements.

Design and Build Iteration: Building upon the functional model iteration, the DSDM methodology proceeds to the design and build iteration stage. During this phase, detailed design specifications are established, and development activities take place to transform the validated prototypes into a tangible product or solution. Collaboration between developers, designers, and stakeholders is paramount, ensuring a smooth flow of communication and rapid progress towards the project's completion.

Implementation: The implementation stage marks the culmination of the DSDM journey, as the developed solution is deployed, integrated, and tested in a controlled environment. Rigorous testing ensures that the solution meets quality standards and performs as expected. This stage also encompasses the training of end-users and the preparation of necessary documentation, setting the stage for a seamless transition to full-scale implementation.

4.4 Implementation Plan

This is the final stage of the project effort where the finished system is made available for the consumer. This section establishes the release principles, project settings, and schedule program. The proposed system should be made available for practice as soon as a fault with it is found and addressed. Most were offered in order to meet user demand; no excess work is included and the system will be ready for launch following this process.

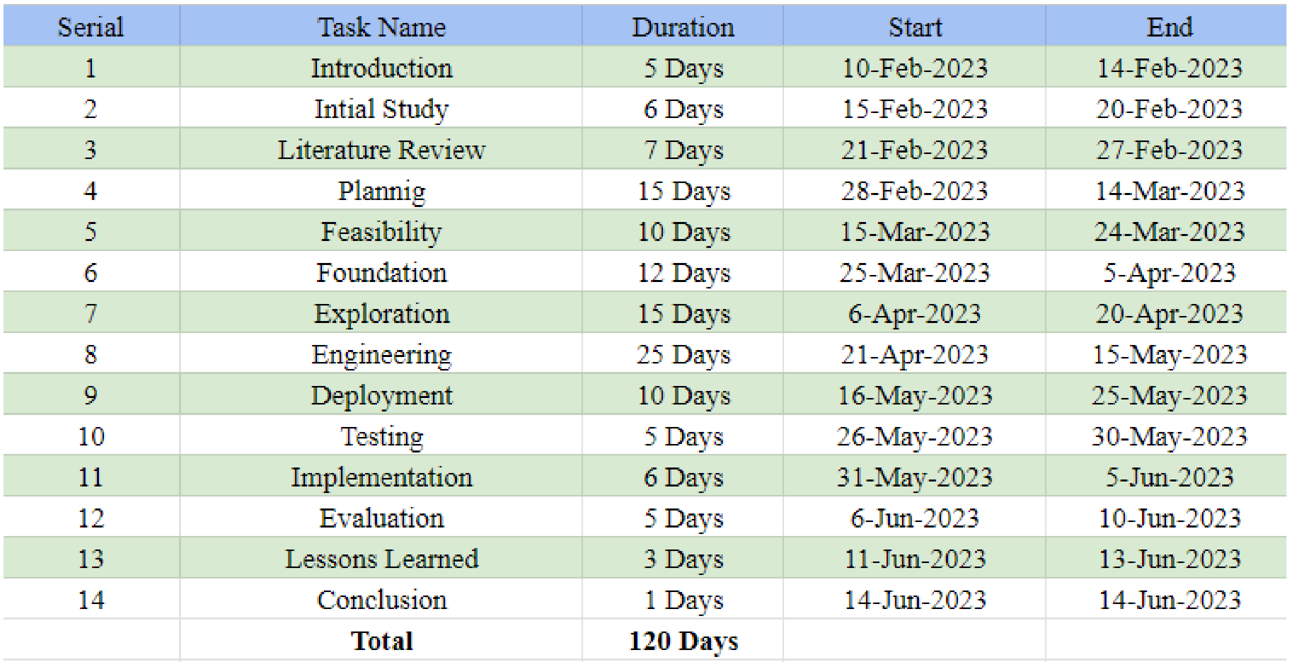
Chapter 5 – Planning

5.1 Project Planning

The initial phase of project planning involves the creation of a set of carefully crafted documents that are firmly based on the project's scope. This scope acts as a definitive definition of how the project will be executed and controlled, taking into consideration both cost estimation and schedule baseline. Prior to commencing the actual implementation of the project, it is crucial to follow a well-structured project planning process that incorporates various elements like the work breakdown structure, timeboxing, and Gantt Chart. These measures play a vital role in ensuring the project's sustainability while minimizing potential risks and challenges. This section serves as a comprehensive source of information, encompassing every aspect of the project and facilitating the transformation of abstract concepts into tangible reality. Such thorough planning is imperative at every stage of the project lifecycle. (Lutkevich, Project planning: What is it and 5 steps to create a plan, 2021) Within this phase, decisions are made regarding the division of the project into distinct components, each of which needs to be accomplished within specific timeframes. Screenshot Regenerate response Screenshot.

5.1.1 Work Breakdown Structure

A WBS is a hierarchical decomposition of the project into smaller, more manageable components. These components can be broken down into smaller tasks and sub-tasks until they are small enough to be executed by an individual or team. The main purpose of a WBS is to provide clarity and understanding of the project scope, deliverables, and timelines. By breaking down the project into smaller components, it becomes easier to estimate time and cost, assign resources, and monitor progress. A well-constructed WBS also helps to minimize project risk, as it identifies potential areas where problems may arise. To create a WBS, I typically start with the project objective and break it down into major deliverables. Then, I divide each deliverable into smaller components until the project is broken down into its most basic elements. Each element should be clear, concise, and well-defined so that everyone involved in the project understands what needs to be done. It is essential to ensure that the WBS is comprehensive and covers all aspects of the project. This includes not only the technical requirements but also other important factors such as project management, quality control, and safety.



Serial	Task Name	Duration	Start	End
1	Introduction	5 Days	10-Feb-2023	14-Feb-2023
2	Intial Study	6 Days	15-Feb-2023	20-Feb-2023
3	Literature Review	7 Days	21-Feb-2023	27-Feb-2023
4	Plannig	15 Days	28-Feb-2023	14-Mar-2023
5	Feasibility	10 Days	15-Mar-2023	24-Mar-2023
6	Foundation	12 Days	25-Mar-2023	5-Apr-2023
7	Exploration	15 Days	6-Apr-2023	20-Apr-2023
8	Engineering	25 Days	21-Apr-2023	15-May-2023
9	Deployment	10 Days	16-May-2023	25-May-2023
10	Testing	5 Days	26-May-2023	30-May-2023
11	Implementation	6 Days	31-May-2023	5-Jun-2023
12	Evaluation	5 Days	6-Jun-2023	10-Jun-2023
13	Lessons Learned	3 Days	11-Jun-2023	13-Jun-2023
14	Conclusion	1 Days	14-Jun-2023	14-Jun-2023
	Total	**120 Days**		

Table 1*: Work Breakdown Structure of BD Rent*

5.1.2 Time Boxing

According to the Project Management Institute's Guide to the Project Management Body of Knowledge (PMBOK), time boxing is "a technique in which a fixed time period is allocated for completion of a specific task or set of tasks" (Project Management Institute, 2017). Time boxing is a project management technique that involves setting a fixed amount of time for a specific task or activity. This approach helps to increase productivity and efficiency by limiting the scope of work and preventing unnecessary delays or distractions. The first step in implementing time boxing is to identify the key tasks or activities that need to be completed within a project. Once these are identified, each task should be assigned a specific time frame according to its level of complexity and priority.

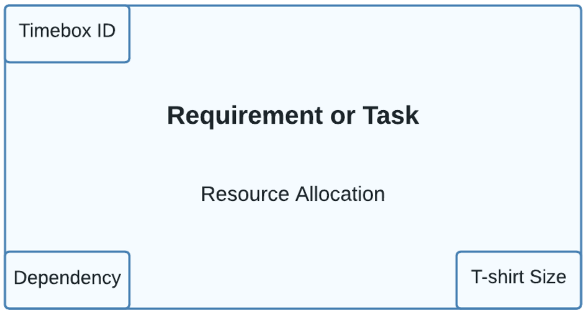


Figure 6*:* Sample of Timeboxing

Time-Boxing List

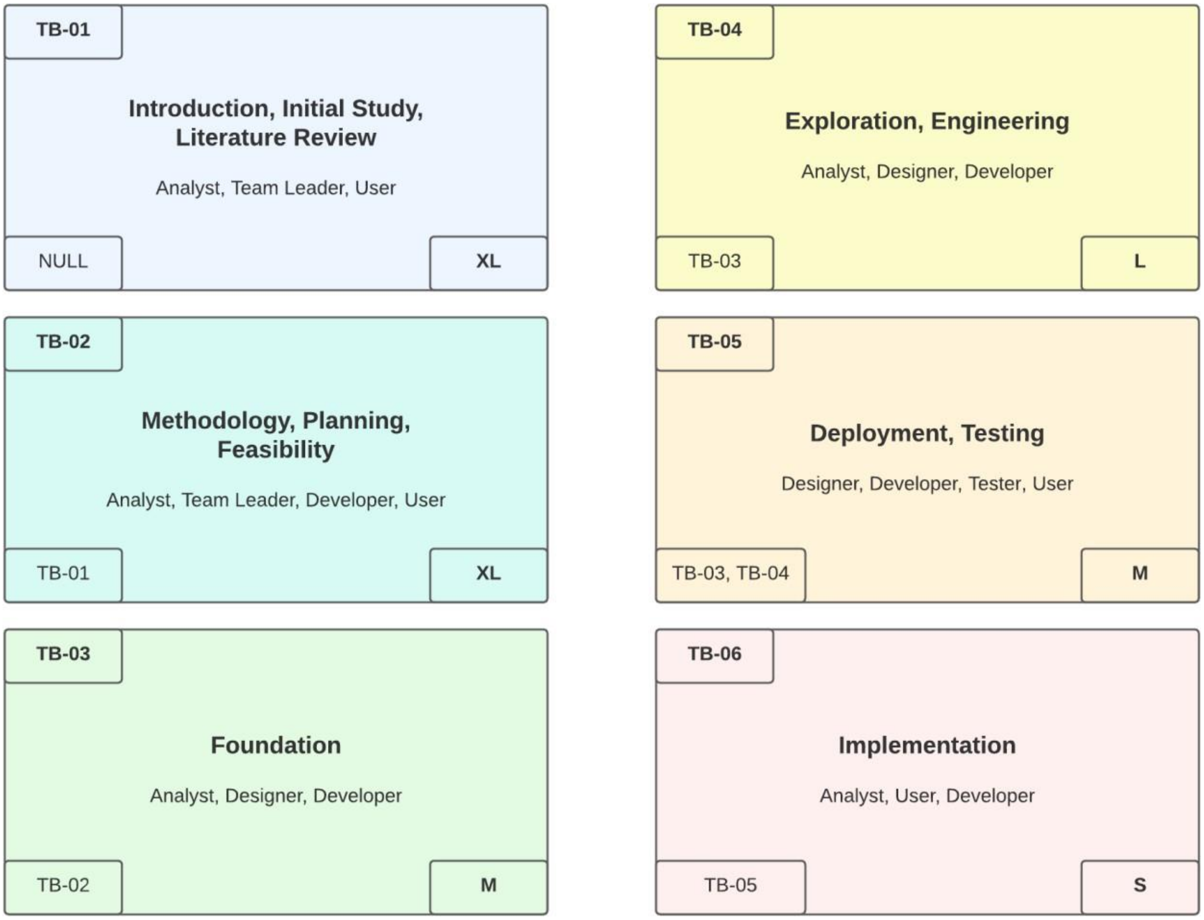

TB-01
Introduction, Initial Study, Literature Review
Analyst, Team Leader, User
NULL
XL
TB-04
Exploration, Engineering
Analyst, Designer, Developer
TB-03
L
TB-02
Methodology, Planning, Feasibility
Analyst, Team Leader, Developer, User
TB-01
XL
TB-05
Deployment, Testing
Designer, Developer, Tester, User
TB-03, TB-04
M
TB-03
Foundation
Analyst, Designer, Developer
TB-02
M
TB-06
Implementation
Analyst, User, Developer
TB-05
S

Figure 7*:* Time-Boxing List

5.1.3 Gantt Chart

Gantt chart is a visual representation of a project schedule that shows the start and end dates of each task within a project. The Gantt chart provides an overview of a project's timeline, as well as its various components and dependencies. This makes it an invaluable tool for project managers who need to keep track of multiple tasks and ensure that they are completed on time. One of the key benefits of a Gantt chart is that it can easily be shared with team members and stakeholders. This helps to ensure that everyone involved in the project is aware of its progress and can identify any potential roadblocks or issues that may arise.

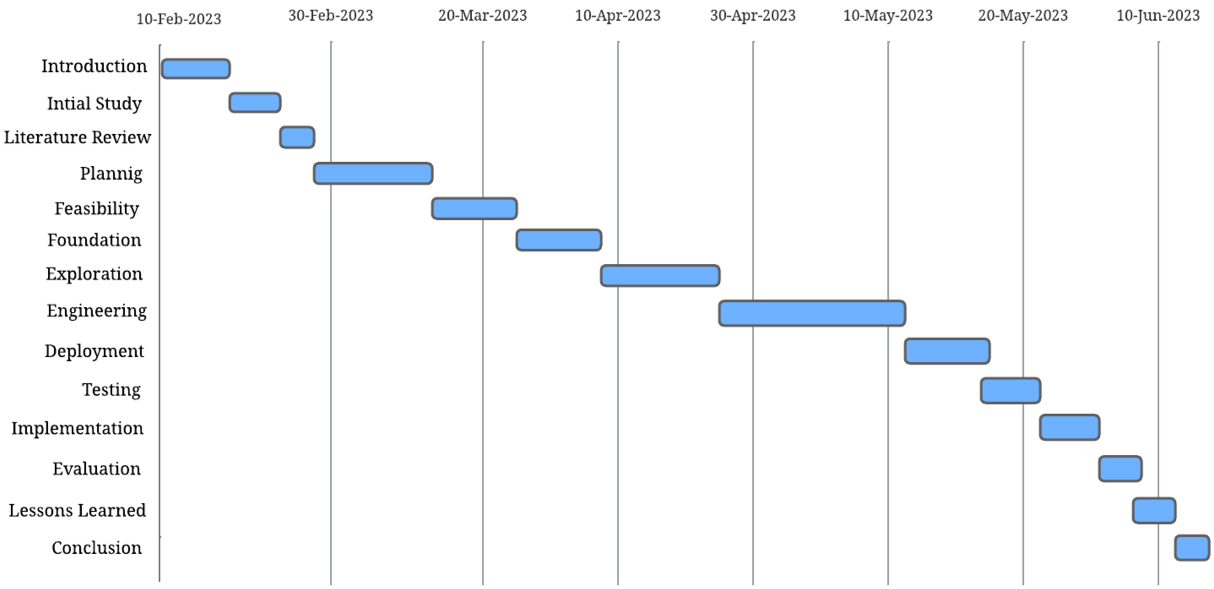


Figure 8*: Gantt Chart of BD Rent*

Chapter 6 – Feasibility

6.1 All Possible Types of Feasibility

Operational Feasibility:

The identification of the project's human resources, together with forecasts based on operating feasibility are essential for a system which will be applied once it is developed and implemented. It shall also give guidance for control by focusing on how a new project will have an effect on daily activities, what procedures should be followed and the strengths that must be taken into consideration to maintain this system in its entirety. An automatic system makes it very easy for the administrator to administer their systems., which also saves time by automatically calculating the mathematical operation. Therefore, there is no need to perform manual calculations for operating the entire project. There is no need to register for admin since the BD Rent admin panel will only be used by the admin. For his first use, he will receive a default username and password at the time of the installation process and after that, he can changehis credentials by going to the profile settings tab. The admin can efficiently perform all the operations including checking all the users documents, booking, house listing.

Technical Feasibility:

A large part of discovering resources is establishing technical feasibility which evaluates the industrial requirements of the intended project. If the current technological competency is capable of satisfying the project criteria, the proposed system is regarded as technically feasible. The technical feasibility test confirms that BD Rent users can easily manage their activities as opposed to the prior manual approach. Web-based applications are platform-independent, and the development platform and resources employed are appropriate and cost-effective, assuring their technological feasibility.

Therefore, in the context of this BD Rent system, the technological aspects are as follows:

- **Hardware**
  - ✓ Desktop PC
    - ○ Intel Core i5 3th gen
    - ○ 500 GB HDD
    - ○ 8 GB DDR3 RAM
  - ✓ Wi-fi Router
- **Software**
  - ✓ Microsoft Word, Excel, and PowerPoint
  - ✓ Chrome
  - ✓ Google Firebase
  - ✓ Windows 10 (Operating System)
  - ✓ Visual Studio Code Editor
- **Database**
  - ✓ MongoDB
- **Technology**
  - ✓ Client-Side
    - ○ HTML 5
    - ○ CSS 3
    - ○ Tailwind
    - ○ JavaScript
    - ○ React
    - ○ React Router
    - ○ Express JS
    - ○ Stripe

6.2 Cost Benefit Analysis

Market Research Analysis Based on the Feasibility Factors:

House booking websites offer convenience. With just a few clicks, users can book a room or an entire apartment from the comfort of their homes. Gone are the days when people had to physically go from one hotel to another searching for suitable accommodation. Moreover, these websites provide detailed information about the property, including pictures, amenities available, and prices, making it easier for users to make informed decisions. Secondly, affordability is another factor driving the demand for house booking websites.

In Bangladesh, hotels can be quite expensive, especially in urban areas. House booking websites, on the other hand, offer a variety of properties at different price points, allowing users to choose options that fit within their budgets. Additionally, some websites offer discounts and promotions, making accommodation even more affordable. Thirdly, house booking websites offer a wide range of options. Whether you're looking for a budget-friendly option or a luxury apartment, there's something for everyone on these websites.

Users can filter their searches based on location, price, amenities, and other preferences, ensuring that they find exactly what they're looking for. Finally, safety is another reason why more and more people are turning to house booking websites. With the ongoing pandemic, people are hesitant to stay in crowded hotels and prefer the privacy and safety of apartments. House booking websites often have strict hygiene and safety protocols in place, giving users peace of mind when booking accommodation.

Cost in total:

Serial No.	**Equipment Cost**	**1st Year**	**2nd Year**	**3rd Year**	**4th Year**	**5th Year**	**Cost in Total**
1	Web-based Application	৳1,20,000					৳1,20,000
2	Email, Domain & Hosting	৳15,000	৳15,000	৳15,000	৳15,000	৳15,000	৳75,000
4	Desktop Application	৳1,25,000					৳1,25,000
5	Maintenance	৳55,000	৳55,000	৳55,000	৳55,000	৳55,000	৳2,75,000
Total		৳4,50,000	৳95,000	৳95,000	৳95,000	৳95,000	**৳5,95,000**

Table 2 Total Cost Estimation for project

Chapter 7 – Foundation

7.1 The Problem Area Identification

The main focus of this application is to make easier for the people to find their home based on the requirements and house owner to rent their house. My main focus is to make the users experience as much smooth as possible. Users play an important role in identifying the problem area, given that we build all types of software for user centricity and their content is a fundamental basis on which our solution can be built. There's so much choice when it comes to requirement elicitation techniques. Below is an overview of some of these techniques:

7.1.1 Interview

One on one interview means one of the primary sources of requirements which, in essence, involves face to face interviews between a system analyst and stakeholders or any other person who would be participating in an ongoing system gathering process. We must keep certain criteria to ensure that we will be able to conduct a successful interview. For example, selecting well thought and trained subjects who are able to interact with the present or new system as well as prepare an open and closed ended series of questions for a potential interviewee. As a consequence, we are categorizing a question set for the users of the proposedsystem, and that is given below:

- What are the problems you face when you are looking for the house?
- How complex is the booking process?
- How is the UI/UX of the website?

7.1.2 Survey methodology

It is a data gathering technique widely employed in various fields, including research, market analysis, and social sciences. It is a systematic approach to collecting information from a targeted population or sample through the use of structured questionnaires or interviews. The survey methodology begins with careful planning and design. Researchers define the objectives of the survey, determine the target population, and select appropriate sampling methods to ensure representativeness. Sampling techniques such as random sampling or stratified sampling may be employed to obtain a diverse and unbiased sample.

7.1.3 Observation

Observing people interacting with your website or product can be useful for data collection because of the candor it offers. By using a third-party tool to record users' journeys through your site or observe a user's interaction with a beta version. The website will provide information about matches, players, ranking, etc. So by gathering information using this technique, we can get an idea of how users interact with the system and what they like and dislike.

7.1.4 Observations

To obtain a more accurate understanding of the user's situation in their current work, an system analyst uses observation as another technique for gathering information. A system analyst can easily see how the user behaves and their environment affects his or her interaction with the system using this approach. Validation of the data which had been previously collected from other sources is also very helpful with this method. As a result, the following important observations are made. Find out the lack of features of the traditional process:

- See how is the users experience of existing system.
- Understand how much users are comfortable while using the existing system.
- To check is the any frustration while using the existing system.

7.2 Specific Problem Area Identification

After doing some researches there are some key problems I have identified:

- ➢ Lack of user's experience
- ➢ No booking processes
- ➢ Complex system to upload house listing for host
- ➢ Complex system for host to manage their house listing
- ➢ Complex system for users to edit their booking

7.3 Possible solutions

Based on my research here are some key possible solutions that I think should implanted on new system.

- ✓ Build a web application that will provide good user experience
- ✓ Users can book houses directly from the website

- ✓ Easy booking system
- ✓ Host can easily upload his house listing and manage that
- ✓ Users can easily edit their booking

7.4 Rich Picture

A rich picture is a type of diagram that is used in systems thinking and business analysis to represent complex situations and relationships. It is a visual tool that helps to identify and understand the key elements, stakeholders, and interactions within a system or organization.

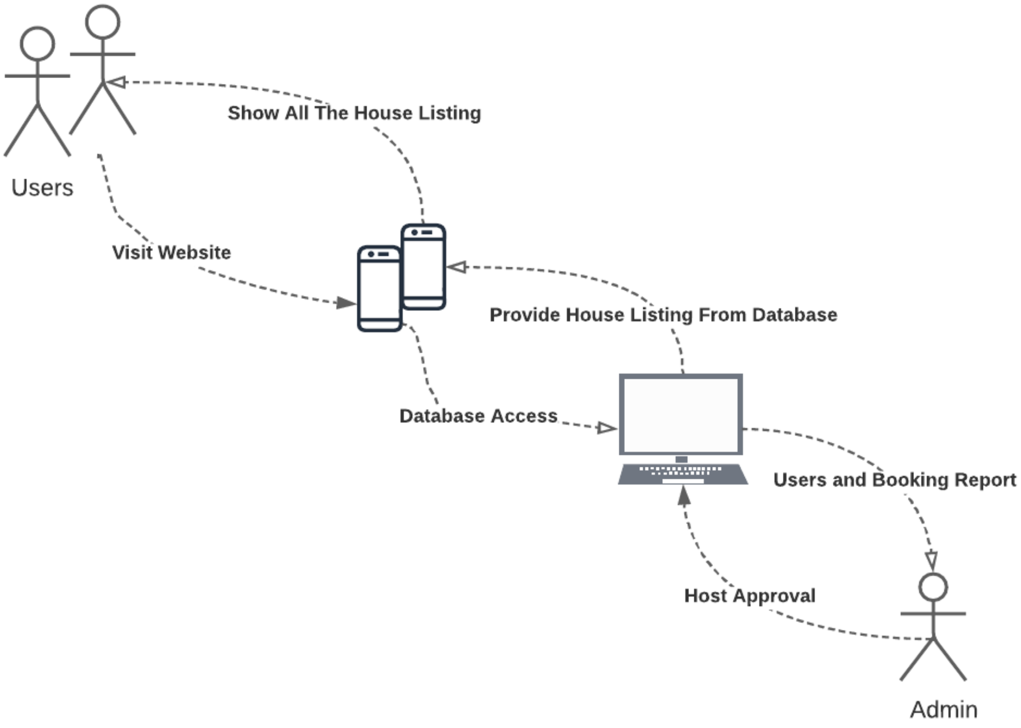


Figure 9: Rich Picture of BD Rent

7.5 Overall Requirement List

The requirement list is mainly separated into two segments:

- ✓ Functional Requirement
- ✓ Non-Functional Requirement

Functional Requirements

- ✓ Admin must have user authentication.
- ✓ Users can manage their profile
- ✓ User can see all the house listing
- ✓ Users can book house
- ✓ Users can make payment
- ✓ Users can request to become host
- ✓ Admin can approve user to become host request
- ✓ Users can have upload their listing
- ✓ Users should edit their house listing
- ✓ Admin must see the all the users details
- ✓ Admin must see all the bookings

Non-Functional Requirements

- ➢ Implement a User-friendly interface design with well-color contrast.
- ➢ The system should be available with 99.9% uptime.
- ➢ All data and information must be protected.
- ➢ The loading time of each page should be less than three seconds.
- ➢ Control access through proper authentication and authorization process

7.6 Which Technology to be implemented

Web based application will be best solution for this type of project. For the frontend development, I will use React and React Router. To enhance the UI design, I will utilize different components from the Tailwind CSS framework. On the backend, I will rely on Node.js as the server runtime and utilized its framework, Express.js, to build a robust backend server. For user authentication, I integrated Firebase. Also, I implemented JSON Web Tokens (JWT) for authorization, which allowed me to validate and manage user access permissions. To store and retrieve data, I used MongoDB, a NoSQL database. For handling payments, I integrated Stripe, a popular payment processing platform.

7.6.1 Client Server Application

Client-server application is a type of software architecture that involves two distinct components: a client and a server.

The client is a piece of software that runs on an end-user device, such as a desktop computer or mobile phone. The server, on the other hand, is a piece of software that runs on a remote machine and provides services to the client.

The client-server model is commonly used in networked systems, where multiple users need to access shared resources or data. In this model, clients send requests to the server, which processes these requests and returns a response back to the client.

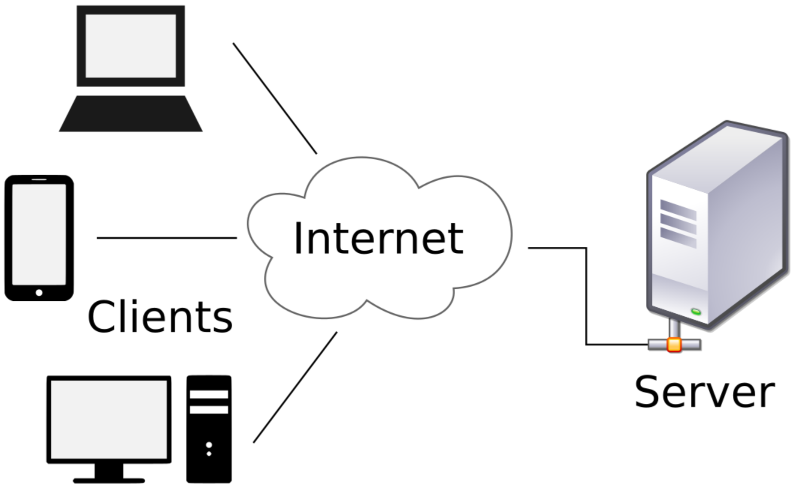


Figure 10: Client Server Application Model

Features of the client-server application

- ✓ Centralized data management
- ✓ Scalability
- ✓ Improved security
- ✓ Enhanced performance

7.6.2 Web Application

A web browser is a piece of software that enables users of web browsers to perform specific tasks which are stored on the server.

Features of the web application

- ✓ **Accessibility:** Web applications are accessible from any device with an internet connection, including desktop computers, laptops, smartphones, and tablets.
- ✓ **User-centric design:** A well-designed web application puts the user at the center, providing an intuitive interface and easy-to-use navigation.
- ✓ **Interactivity:** Web applications offer interactive features such as forms, chatbots, and real-time updates that keep users engaged.
- ✓ Cloud-based hosting: Web applications can be hosted in the cloud, which offers numerous benefits such as scalability, cost-efficiency, and accessibility.

7.7 Recommendation and Justification

The purpose of BD Rent is too accessible from anywhere in the world. Whenever people need booking, they can easily search for house and easily book them. On the other hand, a host can easily list their house and make money. For those type application it suggested client service application is the best and only solution. Without client service application it cannot be achieved.

Chapter 8 – Exploration

8.1 Old System Use Case Diagram

A Use Case Diagram is a type of diagram used in software development to describe the interactions between users and a system. It represents the functional requirements of a system in a graphical form, illustrating the user's goals and system functions that enable them to achieve those goals. The diagram presents actors, use cases, and their relationships. Actors are individuals, organizations, or systems that interact with the system, while use cases represent the specific tasks or actions that a user performs with the system.

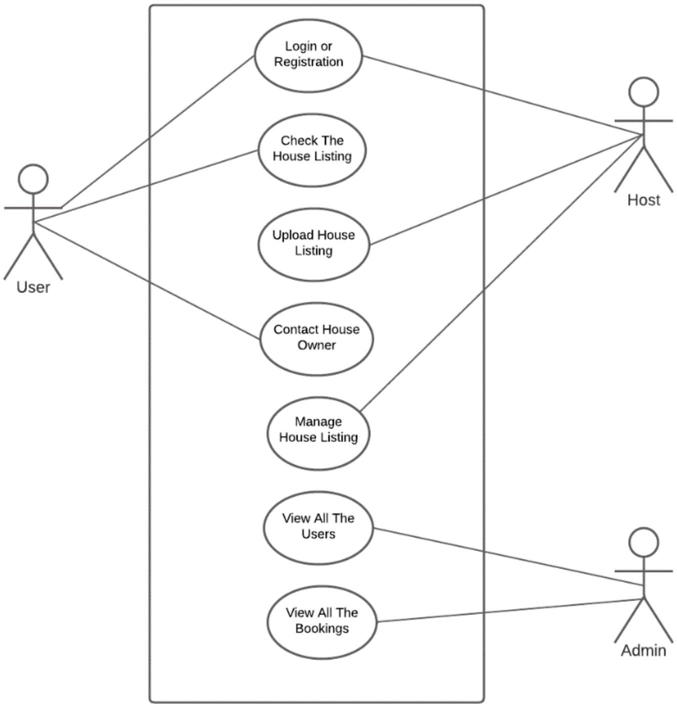


Figure 11*:* Use Case Diagram of Bproperty

8.2 Old System Activity Diagram

A System Activity Diagram is a type of diagram used in software development to visualize the flow of activities within a system. It provides a high-level view of the system's behavior by illustrating the sequence of actions that occur in response to an event. The diagram presents activities, decisions, and branching conditions.

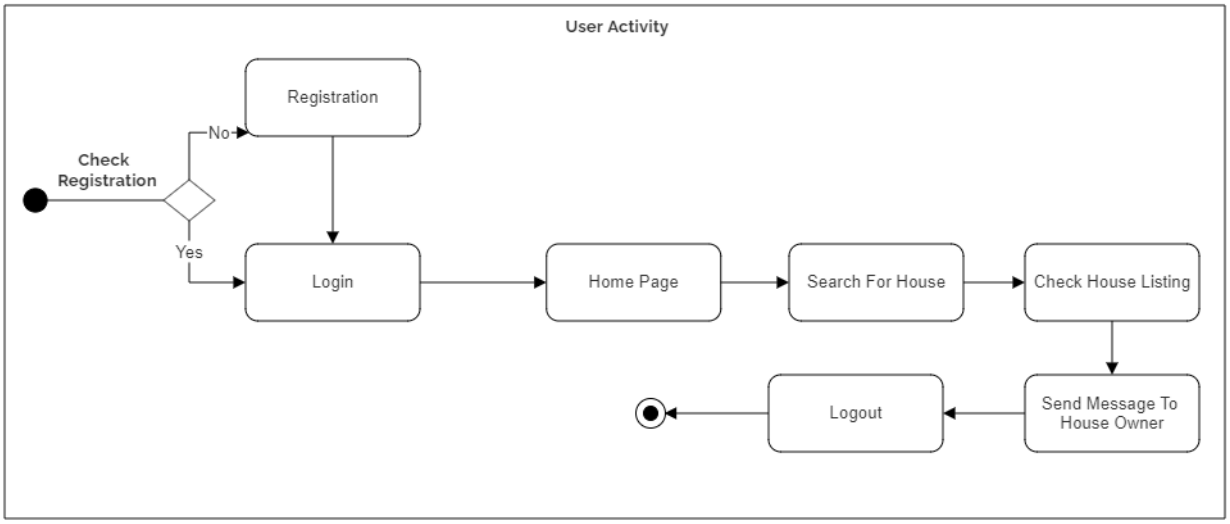


Figure 12*: Old System Activity Diagram*

8.3 Full System Activity Diagram

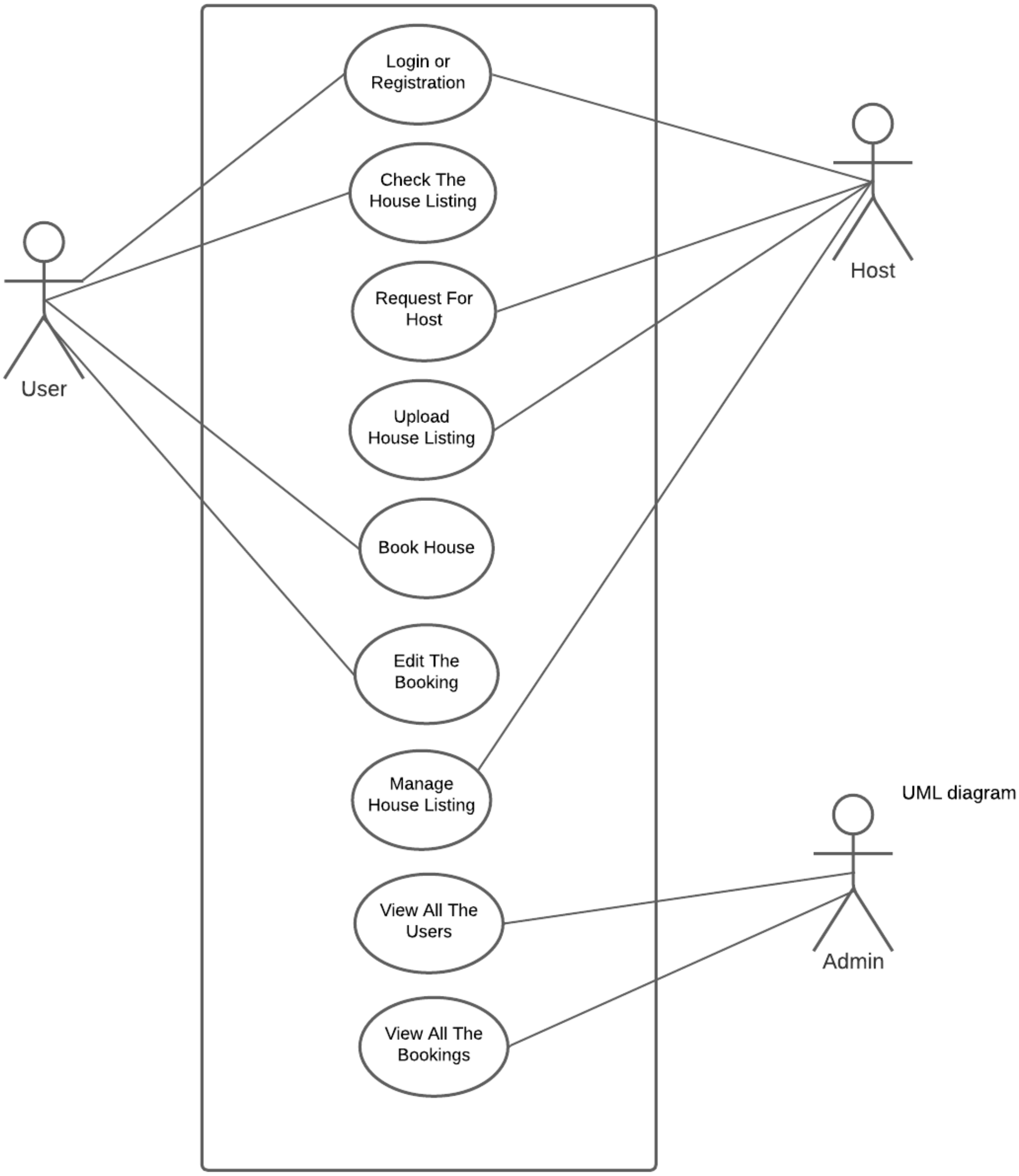


Figure 13: Use Case of BD Rent System

8.4 Full System Activity Diagram

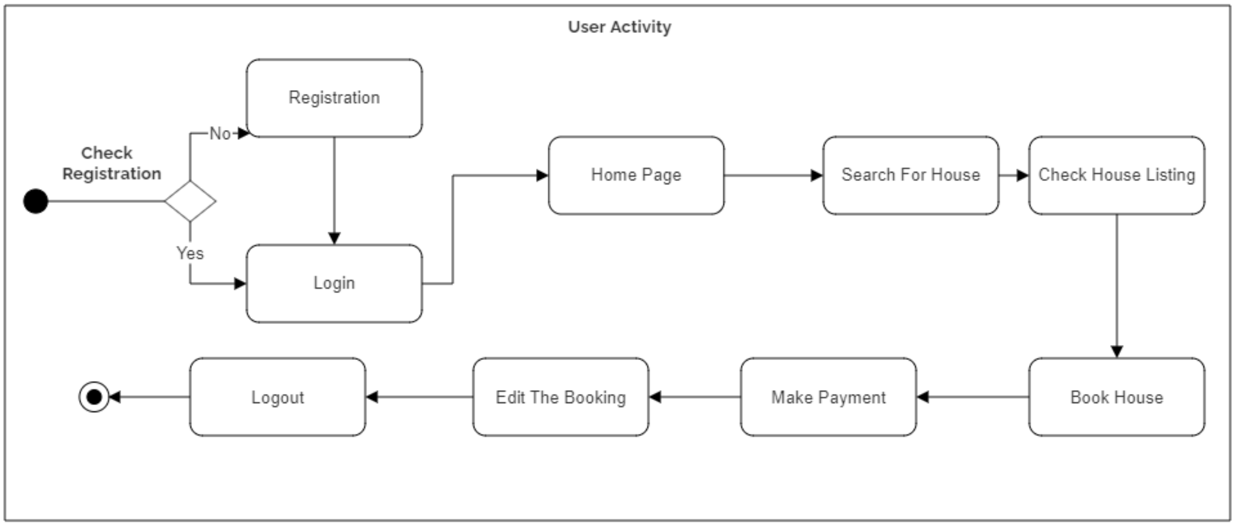
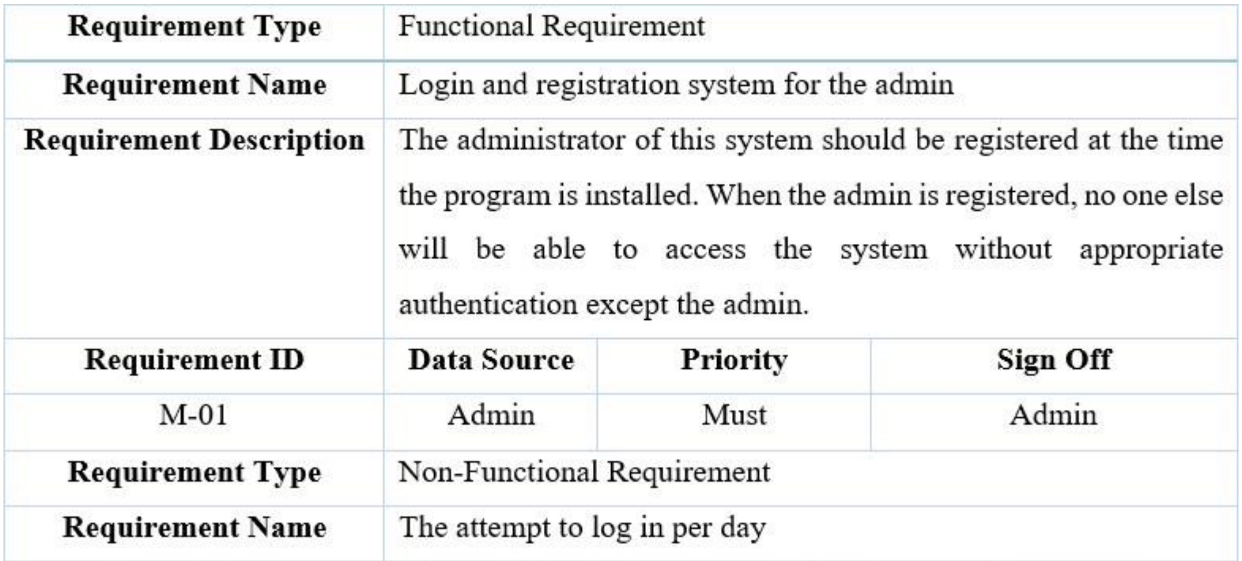


Figure 14*:* Admin’s Activity Diagram

8.5 Requirements Catalogue

Login system



Requirement Type	Functional Requirement		
Requirement Name	Login and registration system for the admin		
Requirement Description	The administrator of this system should be registered at the time the program is installed. When the admin is registered, no one else will be able to access the system without appropriate authentication except the admin.		
Requirement ID	**Data Source**	**Priority**	**Sign Off**
M-01	Admin	Must	Admin
Requirement Type	Non-Functional Requirement		
Requirement Name	The attempt to log in per day		

Table 3 Requirement catalog for the login system

8.6 Prioritized Requirements List (PRL)

This strategy is the most effective method for developing a project's priority list.The following is a list:

Priority Type	Must Have
Serial No.	**Requirement List**
1	Admin will be able to register an account during the project installation process and also login into the system
2	Manage application credentials and email settings by admin
3	Admin will able to see all the users' details
4	Admin will be able to see all the bookings happed in the website
5	Admin will be able to manage manage all the payment details

Table 4: Must-have requirement list

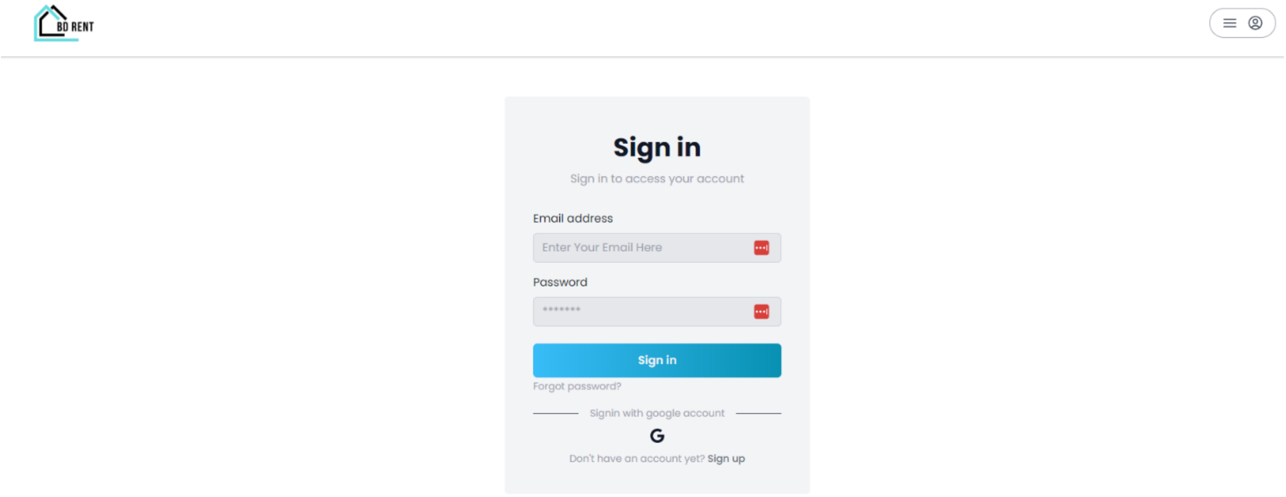
Priority Type	Should Have
Serial No.	**Requirement List**
1	Admin will be able to manage roles and staff
2	Update Admin Profile

Table 5: Should-have requirement list

Priority Type	Could Have
Serial No.	**Requirement List**
1	Manage website general settings
2	Application credentials database management

Table 6: Could have requirement list

8.8 Prototype of the new system



BD RENT

Sign in

Sign in to access your account

Email address

Enter Your Email Here

Password

****

Sign in

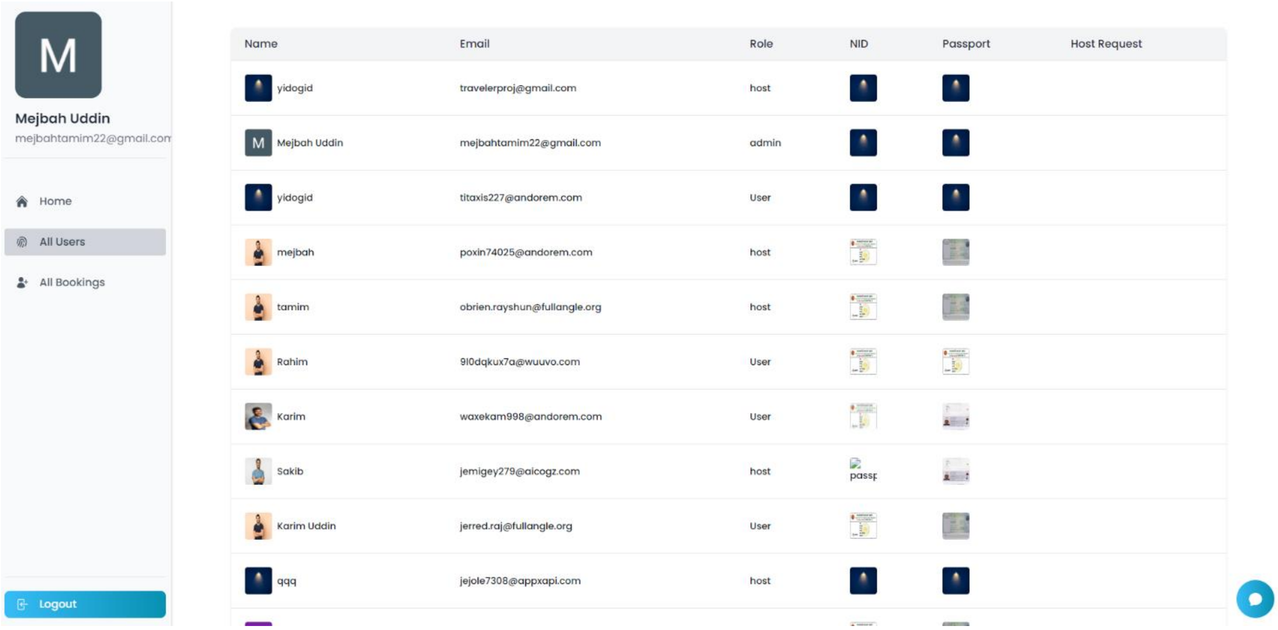
Forgot password?

Signin with google account

G

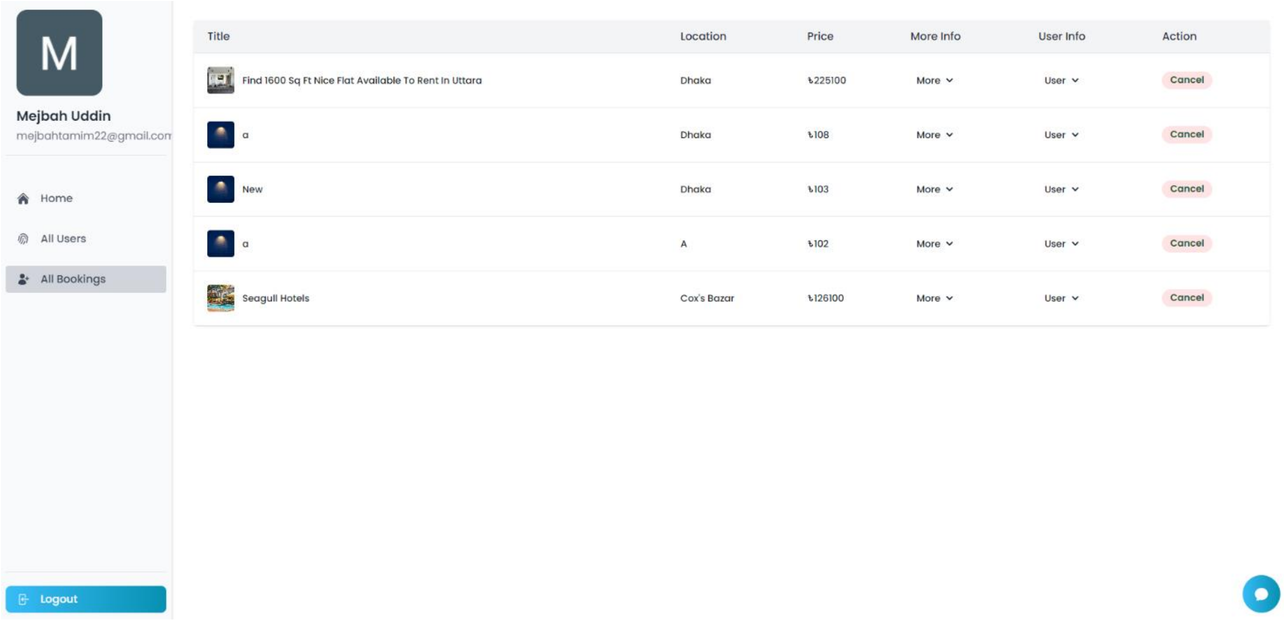
Don't have an account yet? Sign up

Figure 15: Login page



Name	Email	Role	NID	Passport	Host Request
yidogid	travelerproj@gmail.com	host			
Mejbah Uddin	mejbahtamim22@gmail.com	admin			
yidogid	titaxis227@andorem.com	User			
mejbah	poxin74025@andorem.com	host			
tamim	obrien.rayshun@fullangle.org	host			
Rahim	9l0dqkux7a@wuuvo.com	User			
Karim	waxekam998@andorem.com	User			
Sakib	jemigey279@aicogz.com	host	passp		
Karim Uddin	jerred.raj@fullangle.org	User			
qqq	jejole7308@appxapi.com	host			

Figure 16: All users list



Title	Location	Price	More Info	User Info	Action
Find 1600 Sq Ft Nice Flat Available To Rent In Uttara	Dhaka	৳225100	More	User	Cancel
a	Dhaka	৳108	More	User	Cancel
New	Dhaka	৳103	More	User	Cancel
a	A	৳102	More	User	Cancel
Seagull Hotels	Cox's Bazar	৳126100	More	User	Cancel

Figure 17: All booking list

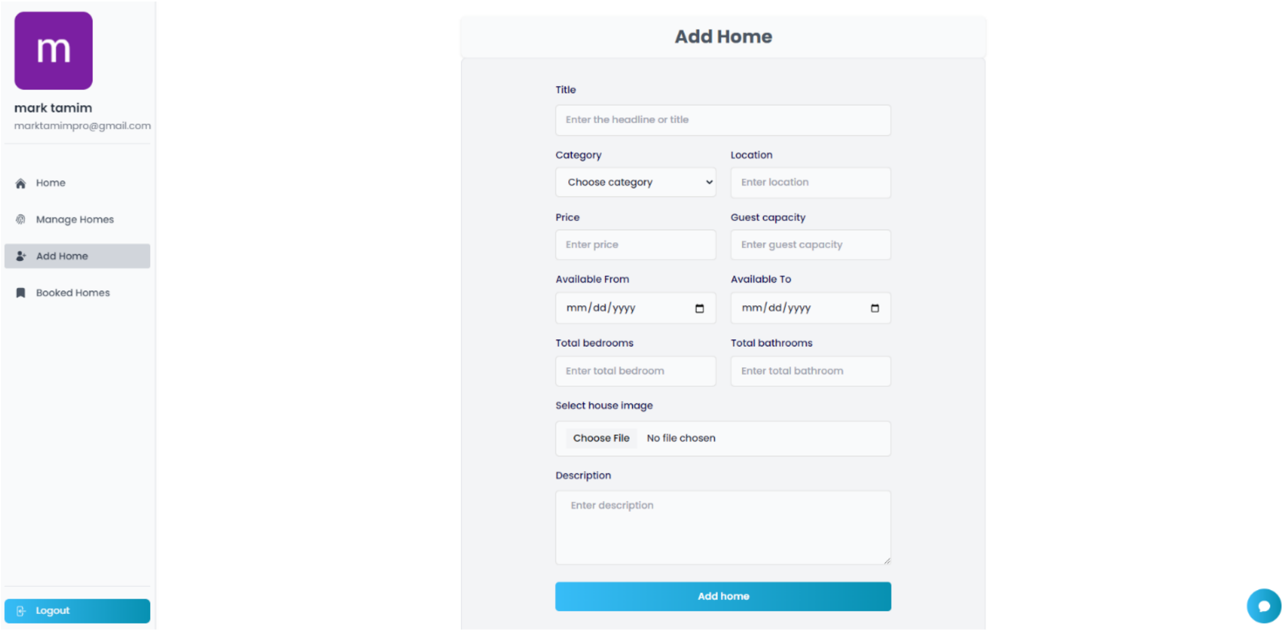
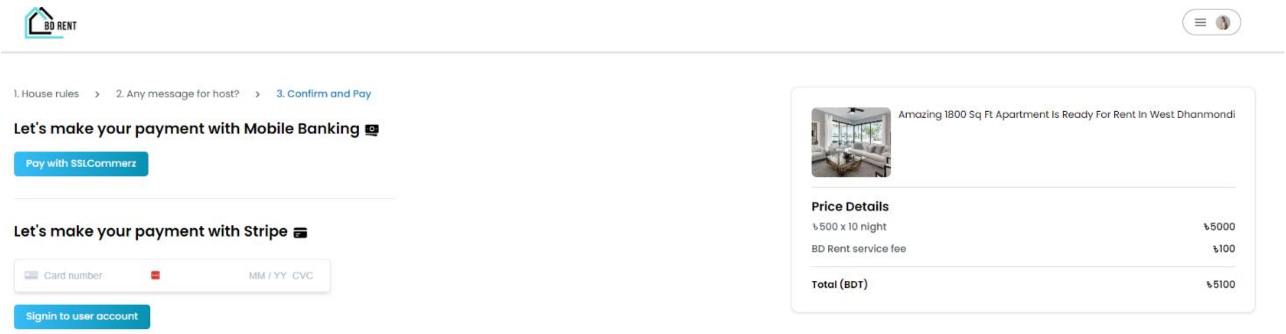


Figure 18: Add new home



Let's make your payment with Mobile Banking

Let's make your payment with Stripe

Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi

Price Details

৳500 x 10 night	৳5000
BD Rent service fee	৳100
Total (BDT)	৳5100

Figure 19: Payment page

Chapter 9 – Engineering

9.1 New System Modules

In BD Rent project, I developed various types of modules based on the business requirements. Below I will show some of the major modules that that I have developed with explanation.

Module for the Admin system

Serial No.	Action	Interaction
1	Admin needs to provide login details once the login form is accessible.	After login details given system will check if everything okay or not.
2	The admin will enter the sign-in or login button.	After collecting the admin's credentials, the system will match the credentials with database data. If the credentials are correct, the admin will be redirected to the admin dashboard, else an error notice will be displayed.

Table 7: Module for the admin system

Module for Users

Serial No.	Action	Interaction
1	The user will enter sign-in button.	After clicking on the sign-in button. User will go create account page. In that page user will give all his necessary data. Then a user account will be created in database.
2	The user will enter login button.	After entering the button system will try to manage the login info with the database login credential. If it matches user will be redirected to the home page.
3	The user will search for home and book home.	When user search for home system will check all the houses based on the user requirements and show on the interface. In booking process user needs to give the details and make payment. A notification will go to house owner and the admin.

Table 8: Module for user

Module for Host

Serial No.	Action	Interaction
2	The administrator will choose the function that best meets his needs.	The system will lead the visitor to a certain page based on the request.
3	The administrator will request that information be added, updated, or deleted based on his specifications.	The system will respond to the admin's request and proceed according to the admin's instructions.
3	Host clicked on upload house listing	After clicking on upload house listing user will go to upload house listing page. There user needs to give the house details to upload his house listing.

Table 9: Module for the host

9.2 Use Case of BD Rent

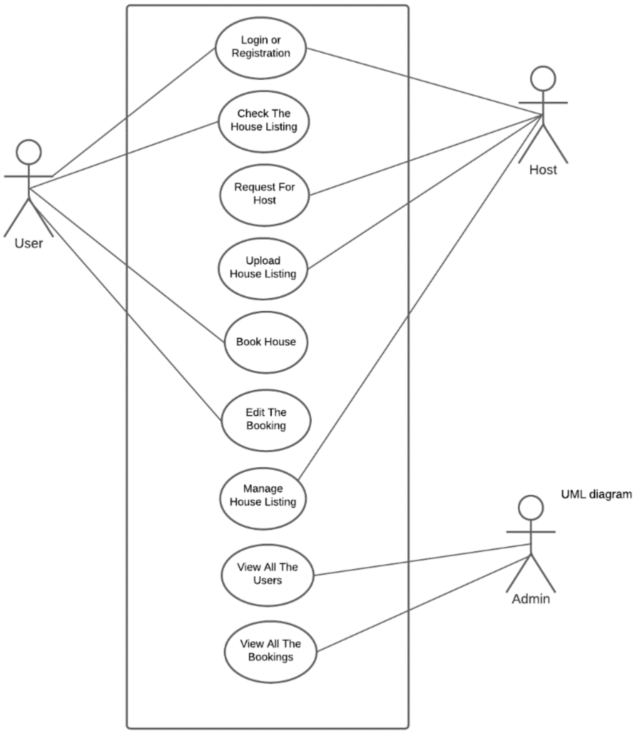


Figure 20*: Use case of the BD Rent*

9.3 Sequence Diagram of BD Rent

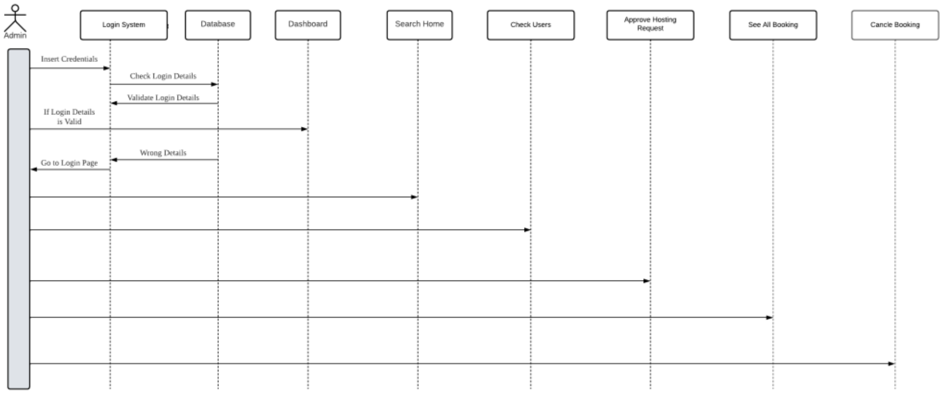


Figure 21*: Sequence diagram of the BD Rent*

9.4 Component Diagram of BD Rent

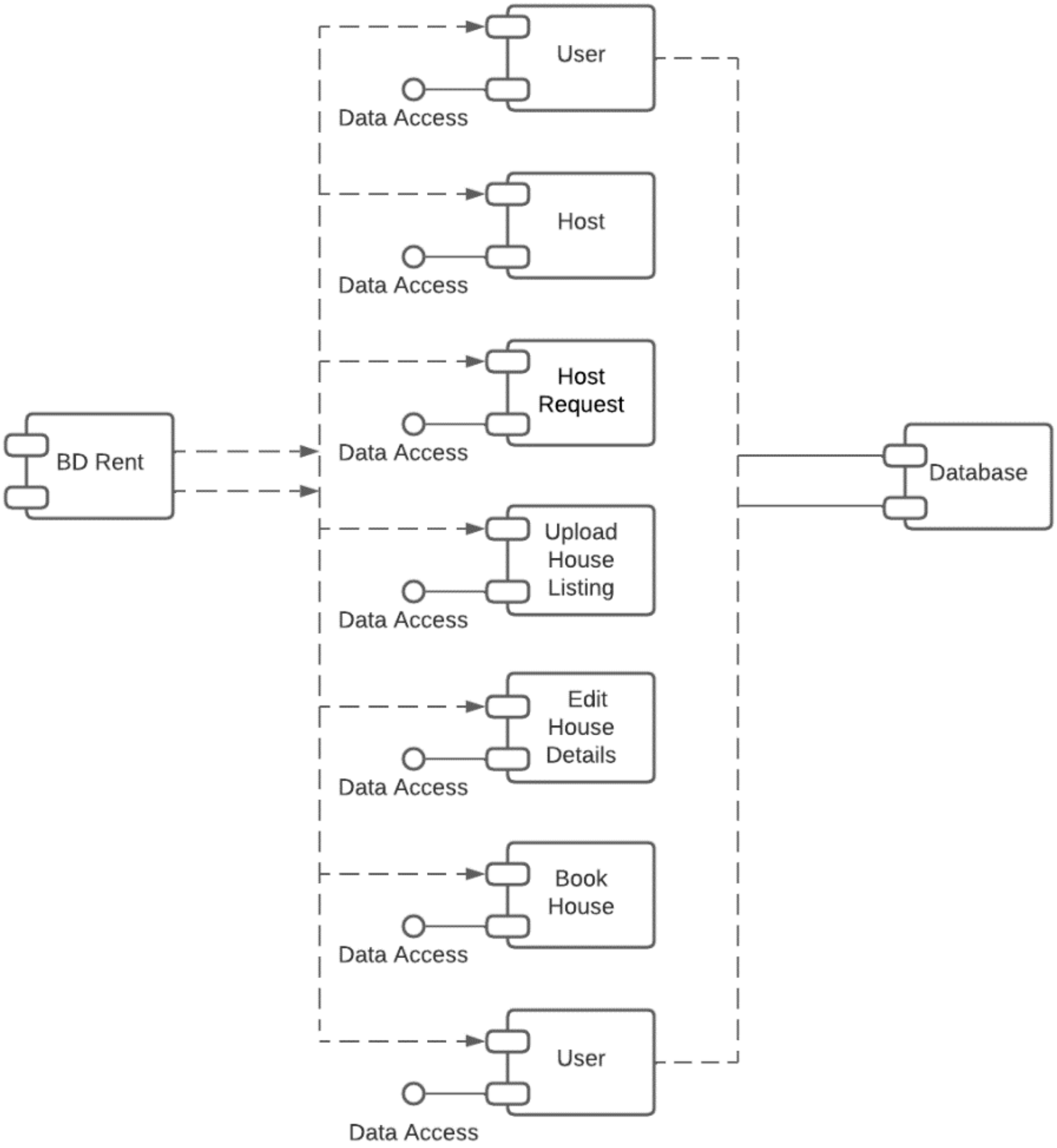
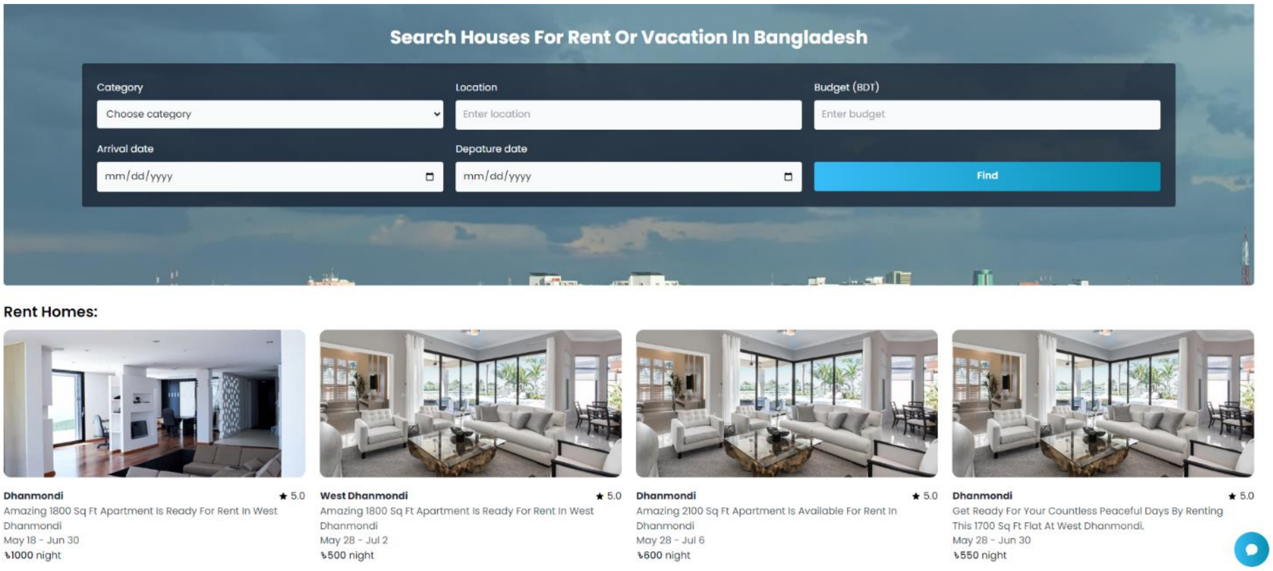


Figure 22*: Component diagram of the BD Rent*

9.5 System Interface Prototype



Search Houses For Rent Or Vacation In Bangladesh

Category: Choose category

Location: Enter location

Budget (BDT): Enter budget

Arrival date: mm/dd/yyyy

Depature date: mm/dd/yyyy

Find

Rent Homes:

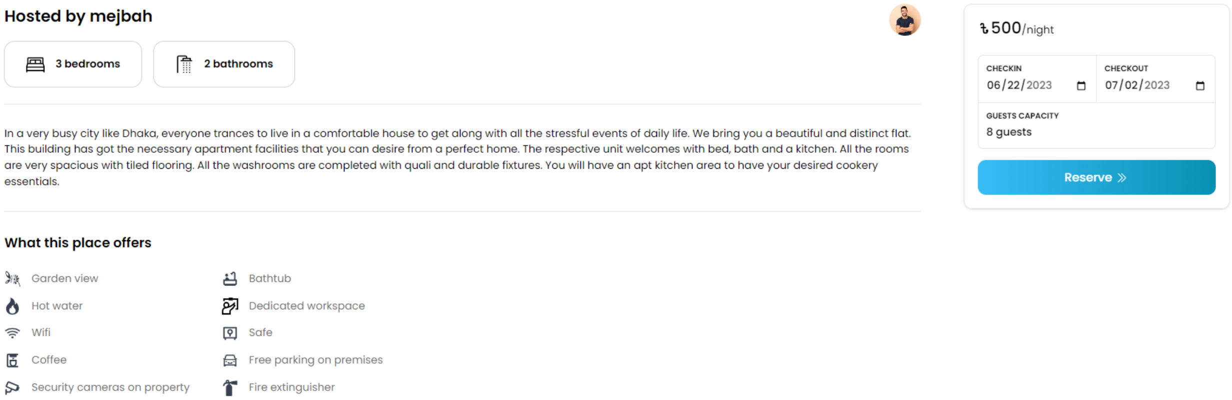
Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 18 - Jun 30
৳1000 night

West Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 28 - Jul 2
৳500 night

Dhanmondi ★ 5.0
Amazing 2100 Sq Ft Apartment Is Available For Rent In Dhanmondi
May 28 - Jul 6
৳600 night

Dhanmondi ★ 5.0
Get Ready For Your Countless Peaceful Days By Renting This 1700 Sq Ft Flat At West Dhanmondi.
May 28 - Jun 30
৳550 night

Figure 23*:* Home page interface prototype



Hosted by mejbah

3 bedrooms

2 bathrooms

In a very busy city like Dhaka, everyone trances to live in a comfortable house to get along with all the stressful events of daily life. We bring you a beautiful and distinct flat. This building has got the necessary apartment facilities that you can desire from a perfect home. The respective unit welcomes with bed, bath and a kitchen. All the rooms are very spacious with tiled flooring. All the washrooms are completed with quali and durable fixtures. You will have an apt kitchen area to have your desired cookery essentials.

What this place offers

- Garden view
- Hot water
- Wifi
- Coffee
- Security cameras on property
- Bathtub
- Dedicated workspace
- Safe
- Free parking on premises
- Fire extinguisher

৳500/night

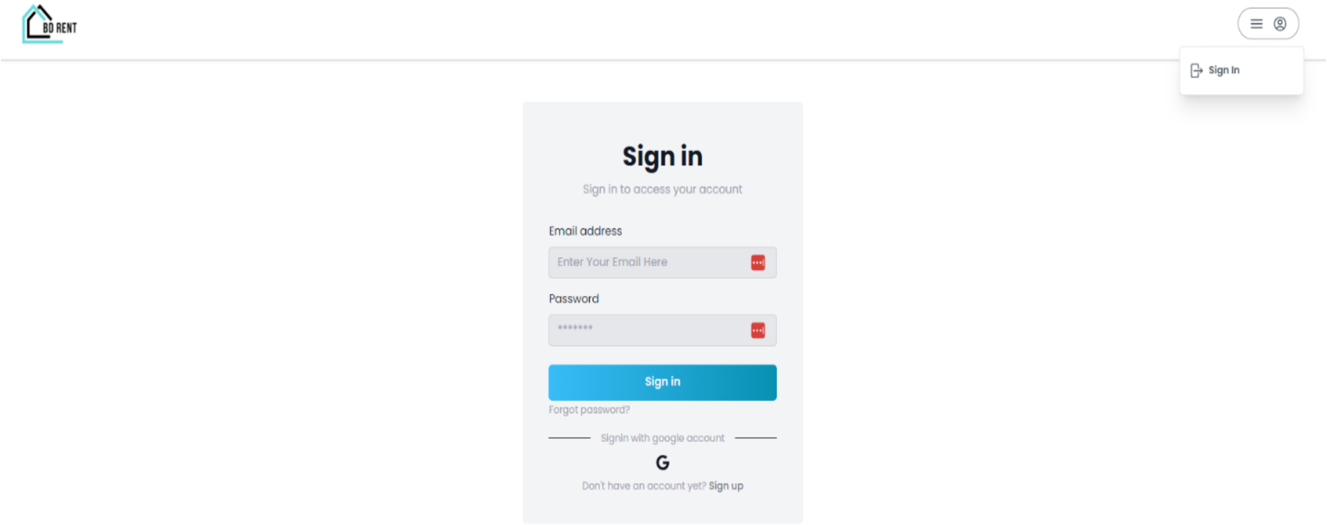
CHECKIN: 06/22/2023

CHECKOUT: 07/02/2023

GUESTS CAPACITY: 8 guests

Reserve »

Figure 24: Home details prototype



BD RENT

Sign In

Sign in

Sign in to access your account

Email address

Enter Your Email Here

Password

****

Sign in

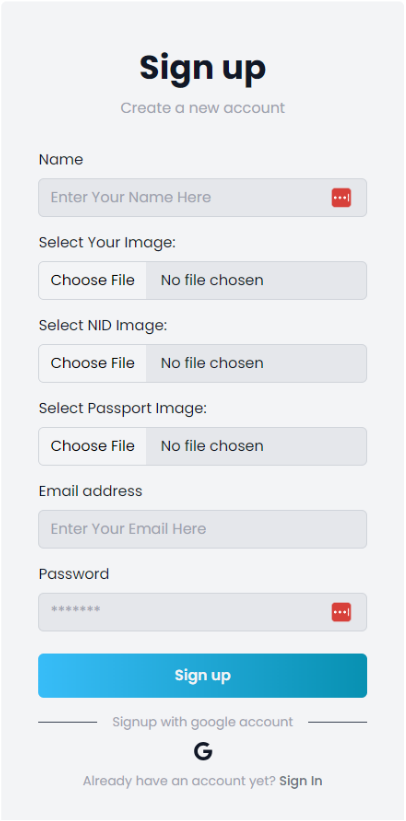
Forgot password?

Signin with google account

G

Don't have an account yet? Sign up

Figure 25*:* Login page prototype



Sign up

Create a new account

Name

Enter Your Name Here

Select Your Image:

Choose File No file chosen

Select NID Image:

Choose File No file chosen

Select Passport Image:

Choose File No file chosen

Email address

Enter Your Email Here

Password

****

Sign up

Signup with google account

G

Already have an account yet? Sign In

Figure 26*:* Sign up page prototype

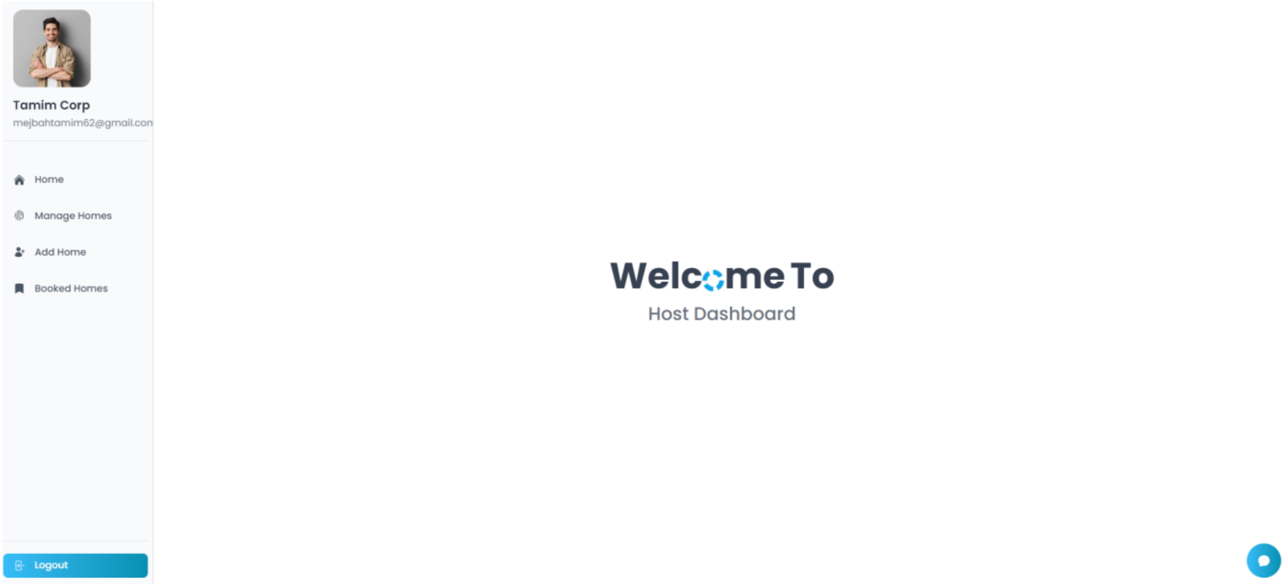
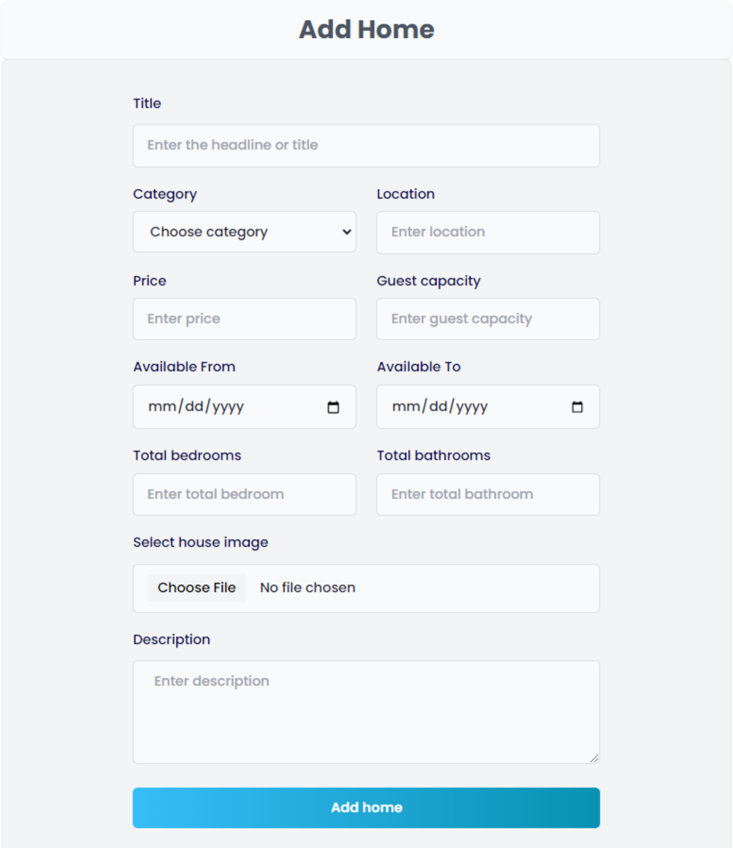


Figure 27: User dashboard prototype



Add Home

Title

Enter the headline or title

Category

Choose category

Location

Enter location

Price

Enter price

Guest capacity

Enter guest capacity

Available From

mm/dd/yyyy

Available To

mm/dd/yyyy

Total bedrooms

Enter total bedroom

Total bathrooms

Enter total bathroom

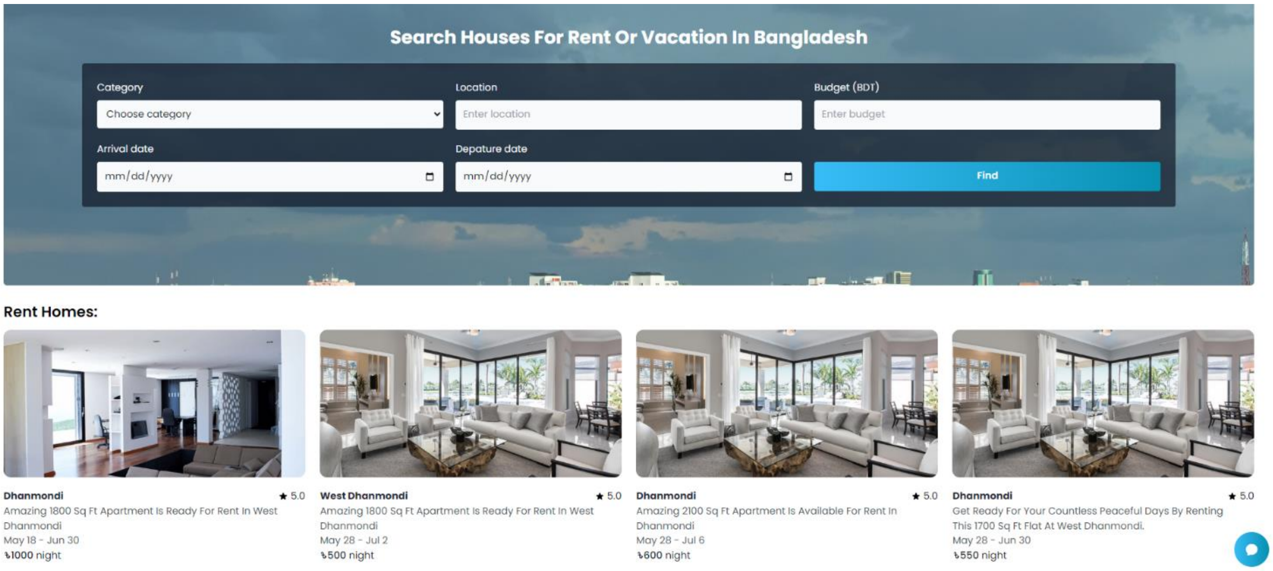
Select house image

Choose File No file chosen

Description

Enter description

Figure 28: New home add page prototype



Rent Homes:

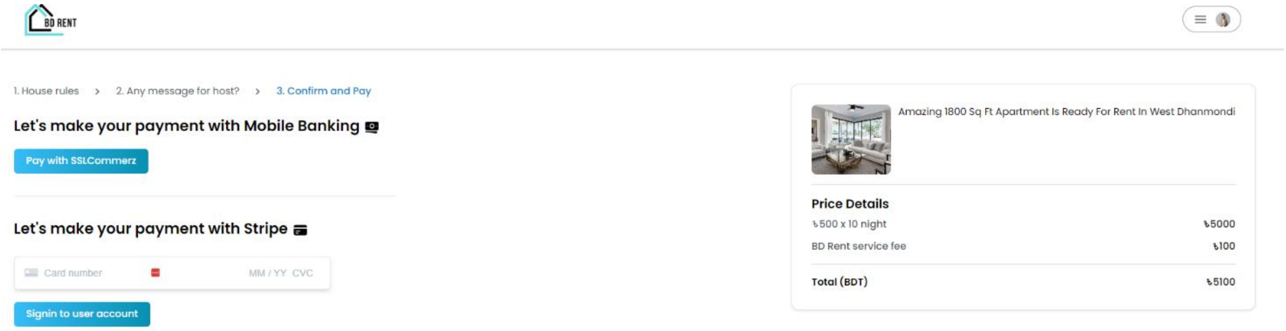
Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 18 - Jun 30
৳1000 night

West Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 28 - Jul 2
৳500 night

Dhanmondi ★ 5.0
Amazing 2100 Sq Ft Apartment Is Available For Rent In Dhanmondi
May 28 - Jul 6
৳600 night

Dhanmondi ★ 5.0
Get Ready For Your Countless Peaceful Days By Renting This 1700 Sq Ft Flat At West Dhanmondi.
May 28 - Jun 30
৳550 night

Figure 29: Home page search bar prototype



Let's make your payment with Mobile Banking

Let's make your payment with Stripe

Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi

Price Details

৳500 x 10 night	৳5000
BD Rent service fee	৳100
Total (BDT)	৳5100

Figure 30: Payment page prototype

Chapter 10 – Deployment

10.1 Possible problem breakdown

The medium and big projects contain a lot of features and functionalities. As a consequence, in order to facilitate and more effective development we need to split the process into separate tasks so that it is not really difficult to properly complete large projects within specified timeframes and budgets. To the proposed BD Rent project, some details are as follows:

- ✓ Normal Error handling
- ✓ Having responsive database design and development
- ✓ Lack of support from users

To avoid those possible problems, I have taken those steps:

- ✓ First clearly identify business requirements and finalizing them.
- ✓ Use different data gathering technique.
- ✓ Real users’ feedback
- ✓ Include every type of error handling

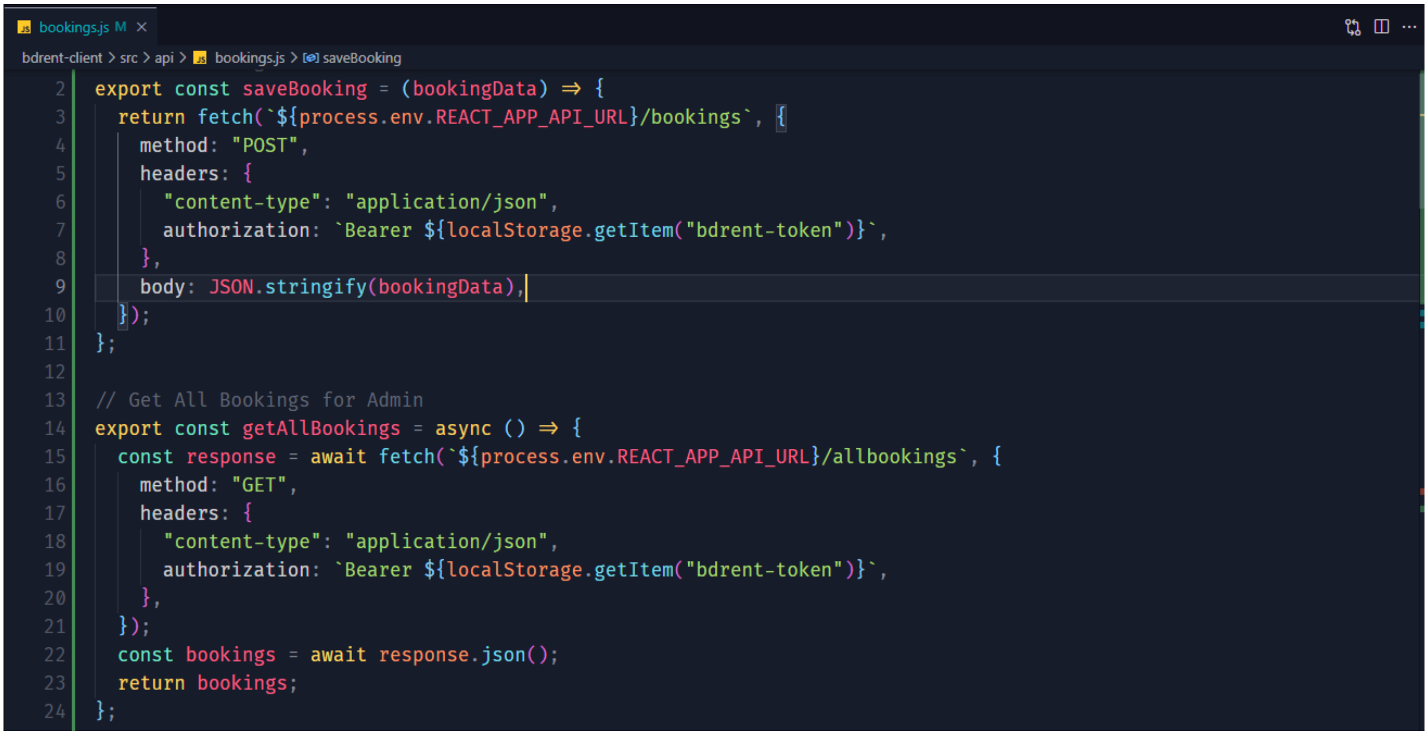
10.2 Prioritization while developing

While doing the project there are many steps and elements included. In order to build the project efficient way, I need complete the task in a prioritize way. Below I have created prioritization list while developing the project.

Priority Serial	Requirement
01	Create database architecture
02	Build a responsive database

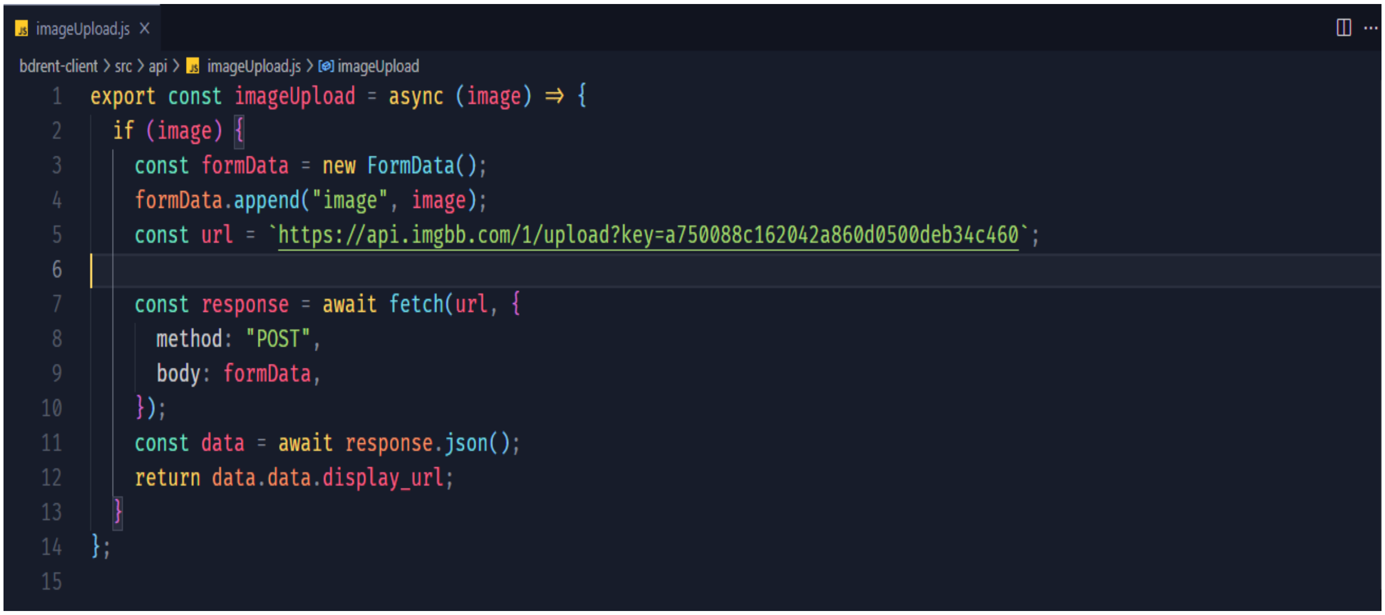
03	For users and admin there must be authentication and authorization to the system during the installation process.
04	Admin will be able to manage user’s role
05	User’s and admin can update their profile
06	Users can upload house list and do the management
07	Admin can see all the users and booking details

Table 10: Prioritization while developing



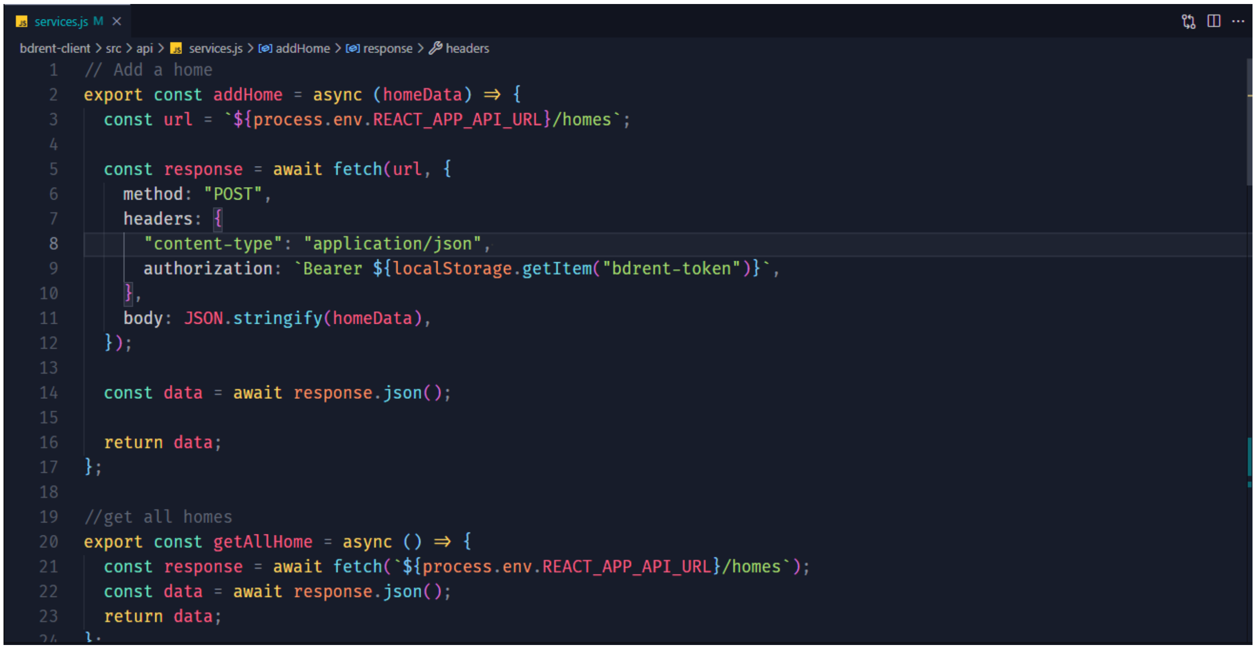
```
export const saveBooking = (bookingData) => {
  return fetch(`${process.env.REACT_APP_API_URL}/bookings`, {
    method: "POST",
    headers: {
      "content-type": "application/json",
      authorization: `Bearer ${localStorage.getItem("bdrent-token")}`,
    },
    body: JSON.stringify(bookingData),
  });
};

// Get All Bookings for Admin
export const getAllBookings = async () => {
  const response = await fetch(`${process.env.REACT_APP_API_URL}/allbookings`, {
    method: "GET",
    headers: {
      "content-type": "application/json",
      authorization: `Bearer ${localStorage.getItem("bdrent-token")}`,
    },
  });
  const bookings = await response.json();
  return bookings;
};
```

Figure 31: Add booking data to database



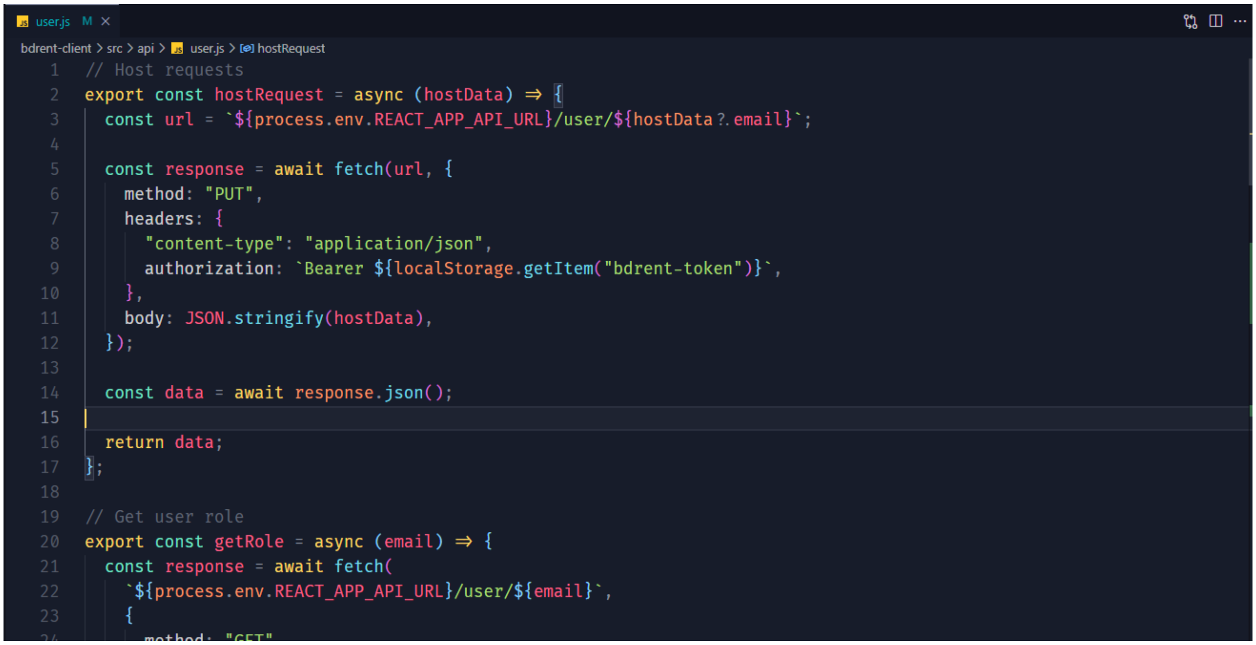
```javascript
export const imageUpload = async (image) => {
  if (image) {
    const formData = new FormData();
    formData.append("image", image);
    const url = `https://api.imgbb.com/1/upload?key=a750088c162042a860d0500deb34c460`;

    const response = await fetch(url, {
      method: "POST",
      body: formData,
    });
    const data = await response.json();
    return data.data.display_url;
  }
};
```

Figure 32*: Image upload to imgbb*


services.js M ×
bdrent-client > src > api > services.js > addHome > response > headers

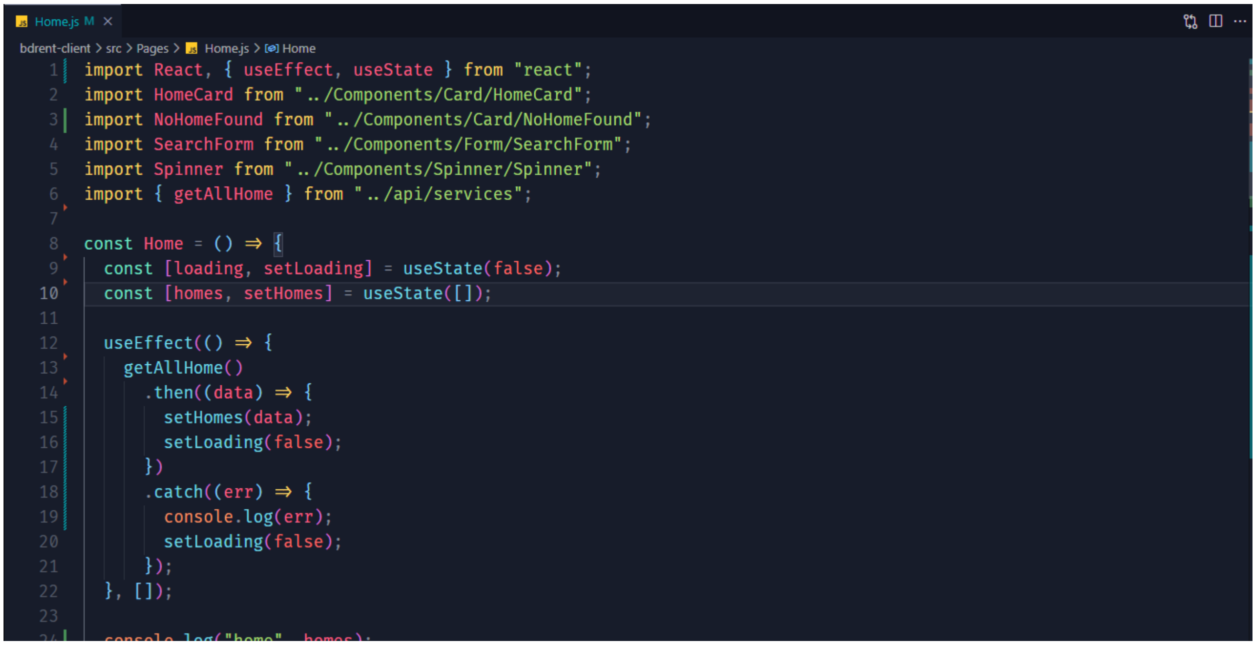
```javascript
// Add a home
export const addHome = async (homeData) => {
  const url = `${process.env.REACT_APP_API_URL}/homes`;

  const response = await fetch(url, {
    method: "POST",
    headers: {
      "content-type": "application/json",
      authorization: `Bearer ${localStorage.getItem("bdrent-token")}`,
    },
    body: JSON.stringify(homeData),
  });

  const data = await response.json();

  return data;
};

//get all homes
export const getAllHome = async () => {
  const response = await fetch(`${process.env.REACT_APP_API_URL}/homes`);
  const data = await response.json();
  return data;
};
```

Figure 33*: Add home to database*



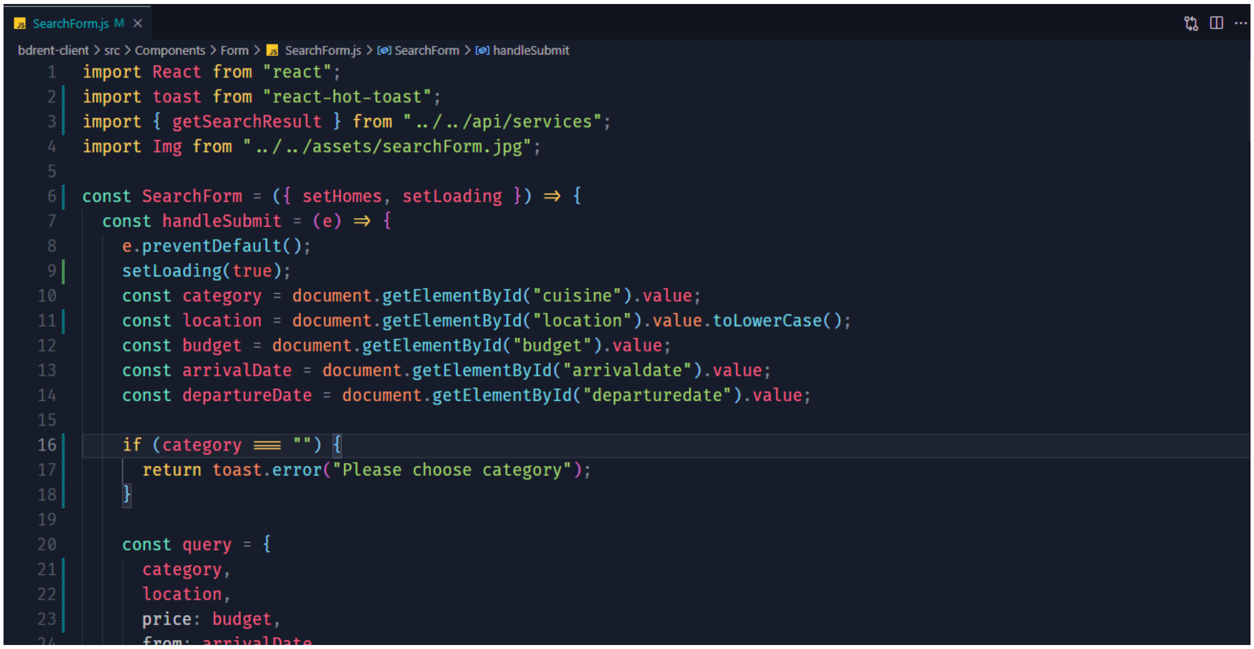
```javascript
// Host requests
export const hostRequest = async (hostData) => {
  const url = `${process.env.REACT_APP_API_URL}/user/${hostData?.email}`;

  const response = await fetch(url, {
    method: "PUT",
    headers: {
      "content-type": "application/json",
      authorization: `Bearer ${localStorage.getItem("bdrent-token")}`,
    },
    body: JSON.stringify(hostData),
  });

  const data = await response.json();

  return data;
};

// Get user role
export const getRole = async (email) => {
  const response = await fetch(
    `${process.env.REACT_APP_API_URL}/user/${email}`,
    {
      method: "GET"
```

Figure 34: Host request add to database



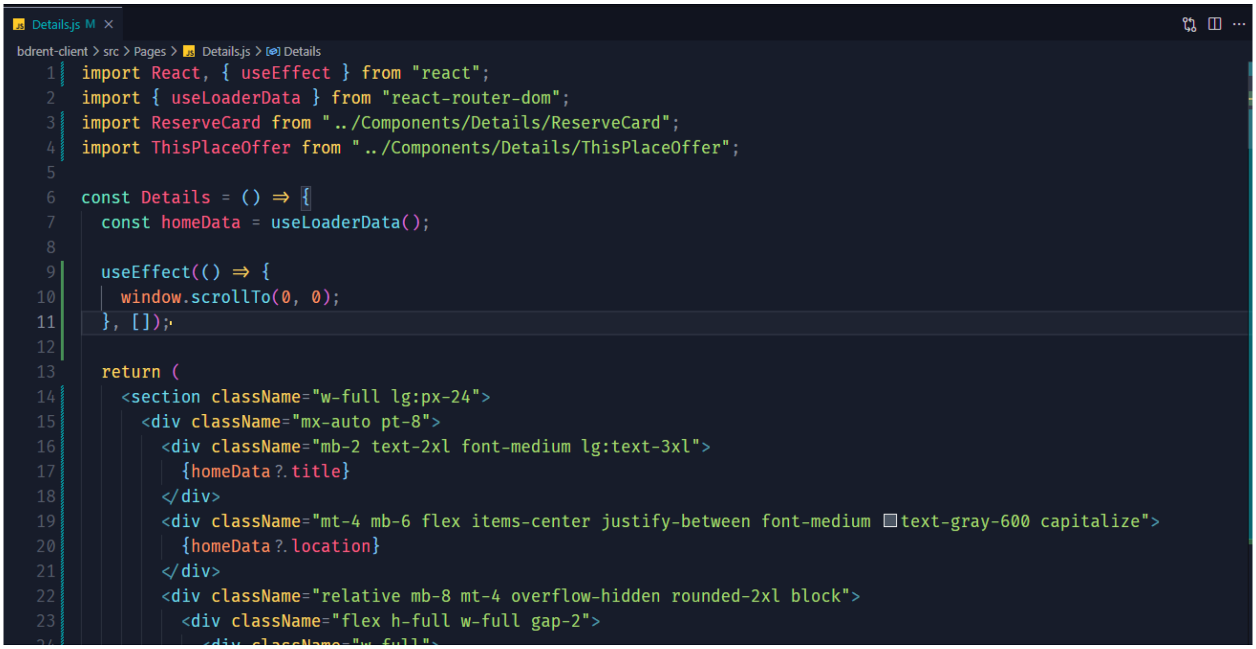
```javascript
import React, { useEffect, useState } from "react";
import HomeCard from "../Components/Card/HomeCard";
import NoHomeFound from "../Components/Card/NoHomeFound";
import SearchForm from "../Components/Form/SearchForm";
import Spinner from "../Components/Spinner/Spinner";
import { getAllHome } from "../api/services";

const Home = () => {
  const [loading, setLoading] = useState(false);
  const [homes, setHomes] = useState([]);

  useEffect(() => {
    getAllHome()
      .then((data) => {
        setHomes(data);
        setLoading(false);
      })
      .catch((err) => {
        console.log(err);
        setLoading(false);
      });
  }, []);

  console.log("home", homes);
```

Figure 35: Search form, homes cards


SearchForm.js M ×
bdrent-client > src > Components > Form > SearchForm.js > SearchForm > handleSubmit

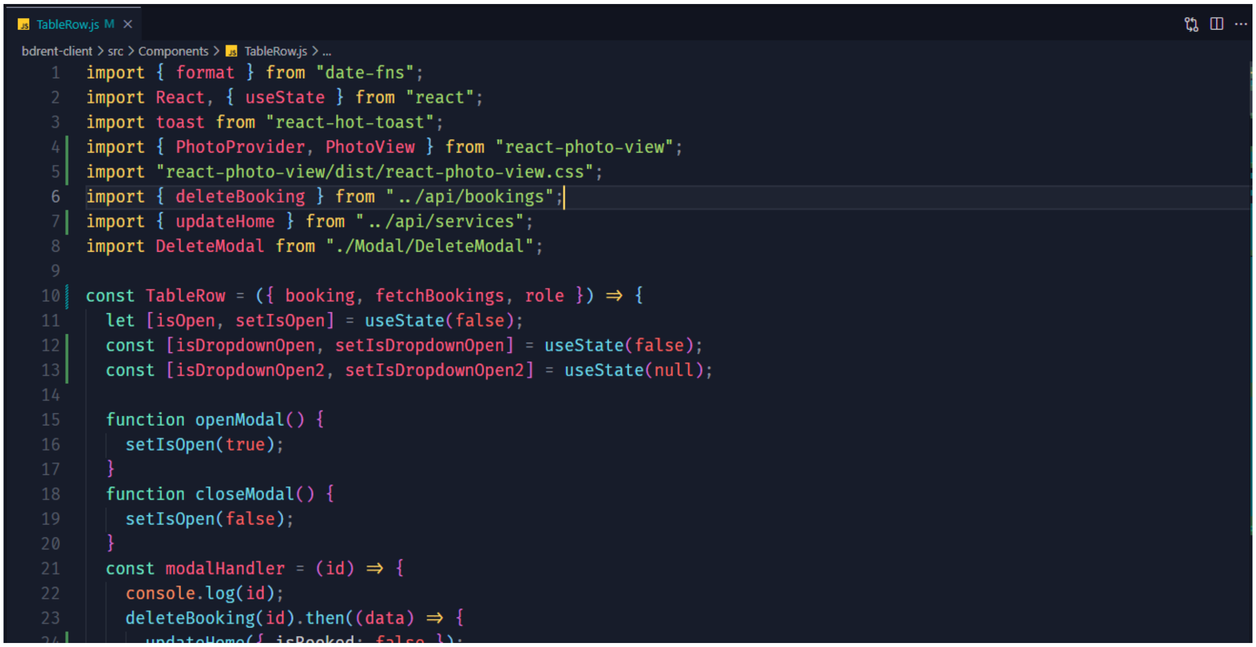
```
import React from "react";
import toast from "react-hot-toast";
import { getSearchResult } from "../../api/services";
import Img from "../../assets/searchForm.jpg";

const SearchForm = ({ setHomes, setLoading }) => {
  const handleSubmit = (e) => {
    e.preventDefault();
    setLoading(true);
    const category = document.getElementById("cuisine").value;
    const location = document.getElementById("location").value.toLowerCase();
    const budget = document.getElementById("budget").value;
    const arrivalDate = document.getElementById("arrivaldate").value;
    const departureDate = document.getElementById("departuredate").value;

    if (category === "") {
      return toast.error("Please choose category");
    }

    const query = {
      category,
      location,
      price: budget,
      from: arrivalDate,
```

Figure 36*: Search house by filter*



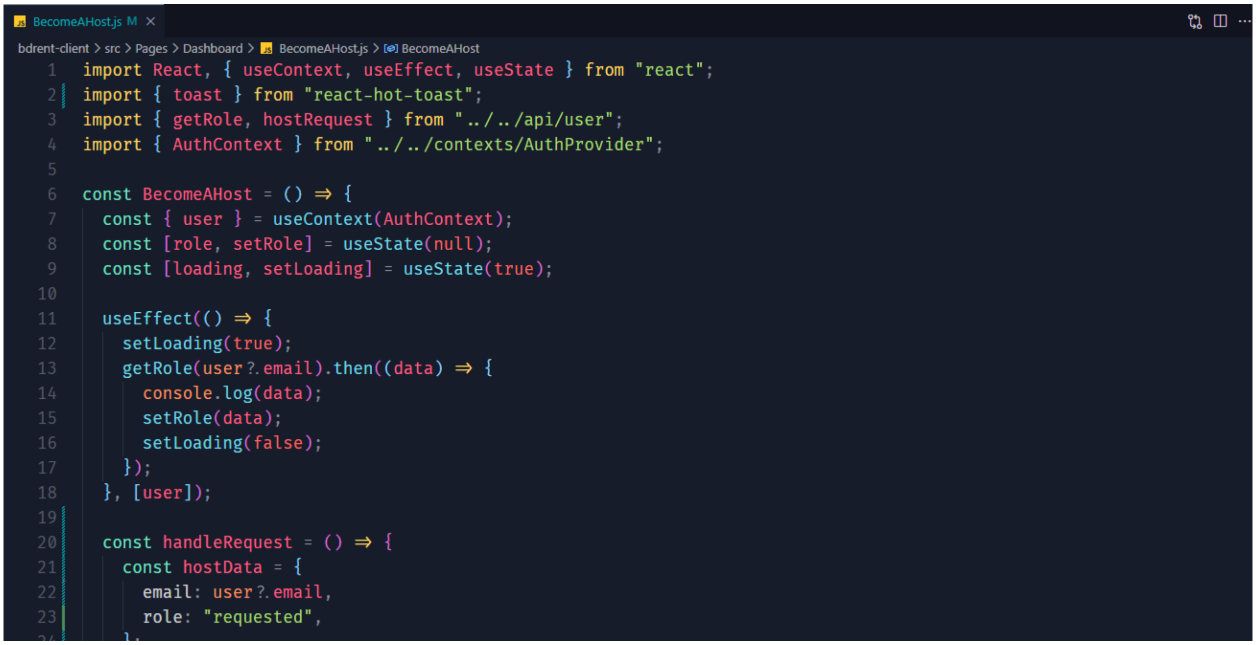
```
import React, { useEffect } from "react";
import { useLoaderData } from "react-router-dom";
import ReserveCard from "../Components/Details/ReserveCard";
import ThisPlaceOffer from "../Components/Details/ThisPlaceOffer";

const Details = () => {
  const homeData = useLoaderData();

  useEffect(() => {
    window.scrollTo(0, 0);
  }, []);

  return (
    <section className="w-full lg:px-24">
      <div className="mx-auto pt-8">
        <div className="mb-2 text-2xl font-medium lg:text-3xl">
          {homeData?.title}
        </div>
        <div className="mt-4 mb-6 flex items-center justify-between font-medium text-gray-600 capitalize">
          {homeData?.location}
        </div>
        <div className="relative mb-8 mt-4 overflow-hidden rounded-2xl block">
          <div className="flex h-full w-full gap-2">
            <div className="w-full">
```

Figure 37*: Specific home details*



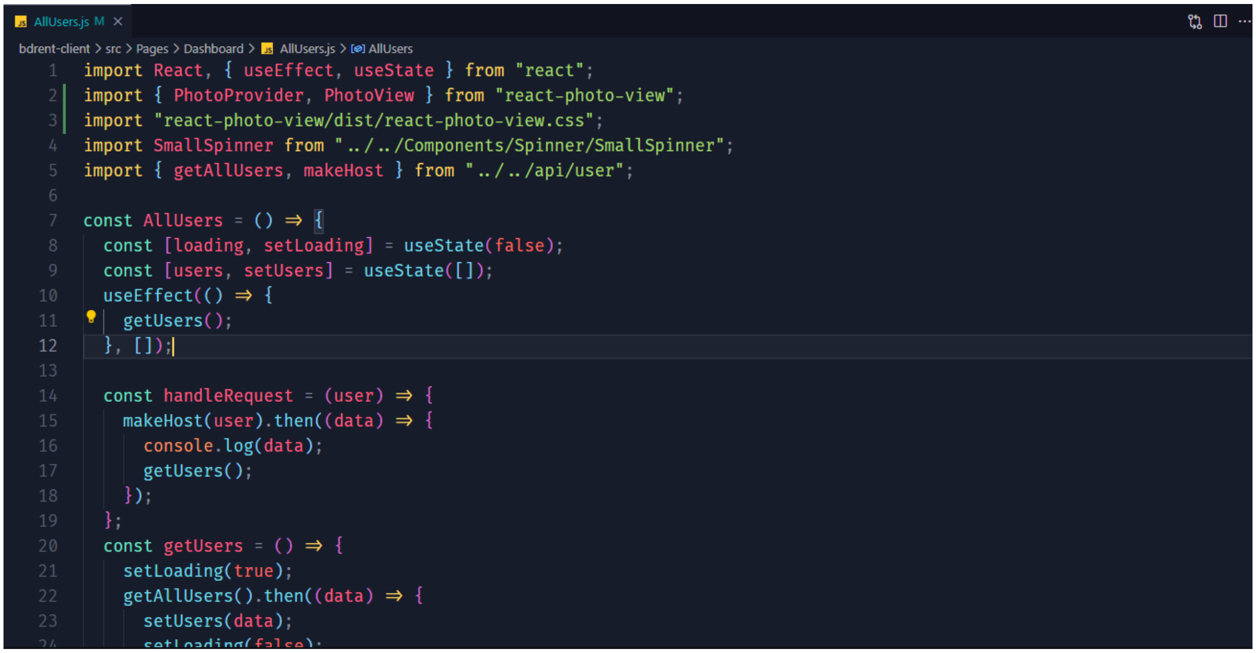
```
import { format } from "date-fns";
import React, { useState } from "react";
import toast from "react-hot-toast";
import { PhotoProvider, PhotoView } from "react-photo-view";
import "react-photo-view/dist/react-photo-view.css";
import { deleteBooking } from "../api/bookings";
import { updateHome } from "../api/services";
import DeleteModal from "./Modal/DeleteModal";

const TableRow = ({ booking, fetchBookings, role }) => {
  let [isOpen, setIsOpen] = useState(false);
  const [isDropdownOpen, setIsDropdownOpen] = useState(false);
  const [isDropdownOpen2, setIsDropdownOpen2] = useState(null);

  function openModal() {
    setIsOpen(true);
  }
  function closeModal() {
    setIsOpen(false);
  }
  const modalHandler = (id) => {
    console.log(id);
    deleteBooking(id).then((data) => {
      updateHome({ isBooked: false });
```

Figure 38: Table row to show all bookings


BecomeAHost.js M ×
bdrent-client > src > Pages > Dashboard > BecomeAHost.js > BecomeAHost

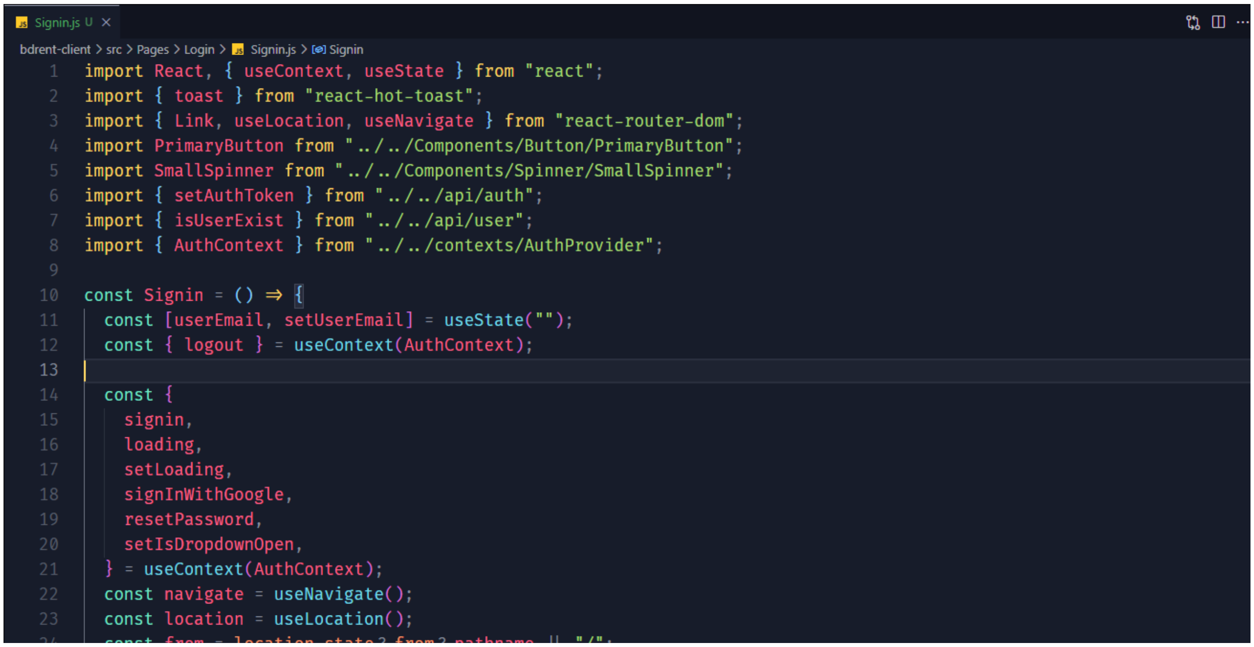
```
import React, { useContext, useEffect, useState } from "react";
import { toast } from "react-hot-toast";
import { getRole, hostRequest } from "../../api/user";
import { AuthContext } from "../../contexts/AuthProvider";

const BecomeAHost = () => {
  const { user } = useContext(AuthContext);
  const [role, setRole] = useState(null);
  const [loading, setLoading] = useState(true);

  useEffect(() => {
    setLoading(true);
    getRole(user?.email).then((data) => {
      console.log(data);
      setRole(data);
      setLoading(false);
    });
  }, [user]);

  const handleRequest = () => {
    const hostData = {
      email: user?.email,
      role: "requested",
    };
```

Figure 39: Request to become a host


AllUsers.js M ×
bdrent-client > src > Pages > Dashboard > AllUsers.js > AllUsers

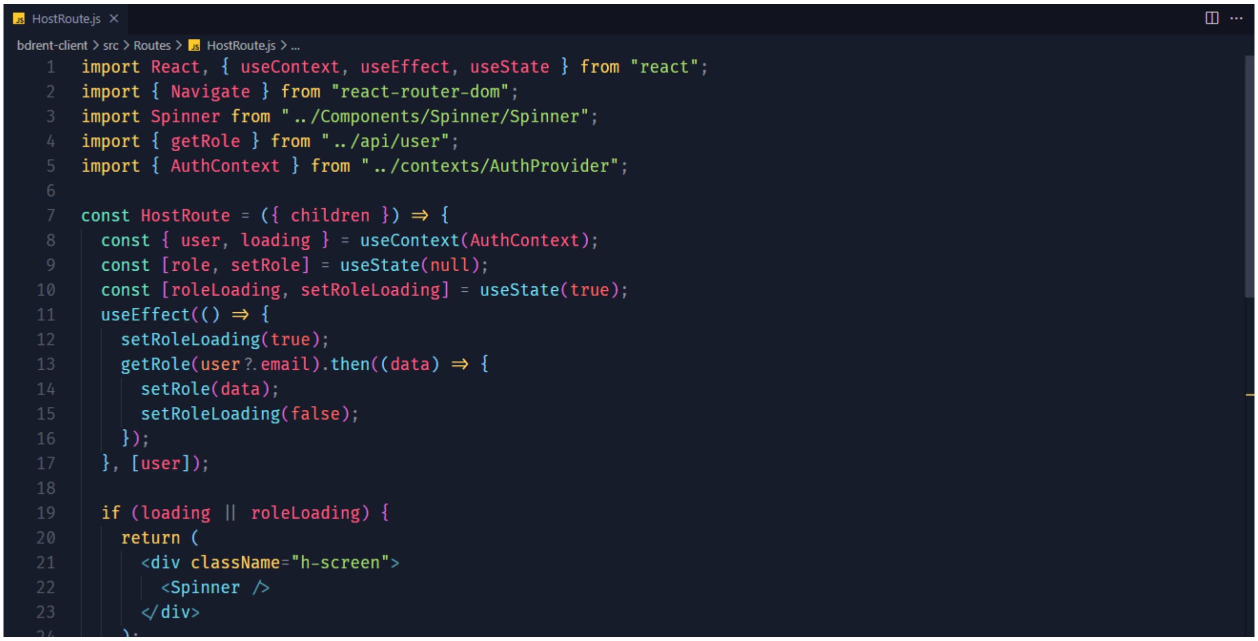
```
import React, { useEffect, useState } from "react";
import { PhotoProvider, PhotoView } from "react-photo-view";
import "react-photo-view/dist/react-photo-view.css";
import SmallSpinner from "../../Components/Spinner/SmallSpinner";
import { getAllUsers, makeHost } from "../../api/user";

const AllUsers = () => {
  const [loading, setLoading] = useState(false);
  const [users, setUsers] = useState([]);
  useEffect(() => {
    getUsers();
  }, []);

  const handleRequest = (user) => {
    makeHost(user).then((data) => {
      console.log(data);
      getUsers();
    });
  };
  const getUsers = () => {
    setLoading(true);
    getAllUsers().then((data) => {
      setUsers(data);
      setLoading(false);
```

Figure 40*:* Admin can see all users every information



```
import React, { useContext, useState } from "react";
import { toast } from "react-hot-toast";
import { Link, useLocation, useNavigate } from "react-router-dom";
import PrimaryButton from "../../Components/Button/PrimaryButton";
import SmallSpinner from "../../Components/Spinner/SmallSpinner";
import { setAuthToken } from "../../api/auth";
import { isUserExist } from "../../api/user";
import { AuthContext } from "../../contexts/AuthProvider";

const Signin = () => {
  const [userEmail, setUserEmail] = useState("");
  const { logout } = useContext(AuthContext);

  const {
    signin,
    loading,
    setLoading,
    signInWithGoogle,
    resetPassword,
    setIsDropdownOpen,
  } = useContext(AuthContext);
  const navigate = useNavigate();
  const location = useLocation();
  const from = location.state?.from?.pathname || "/";
```

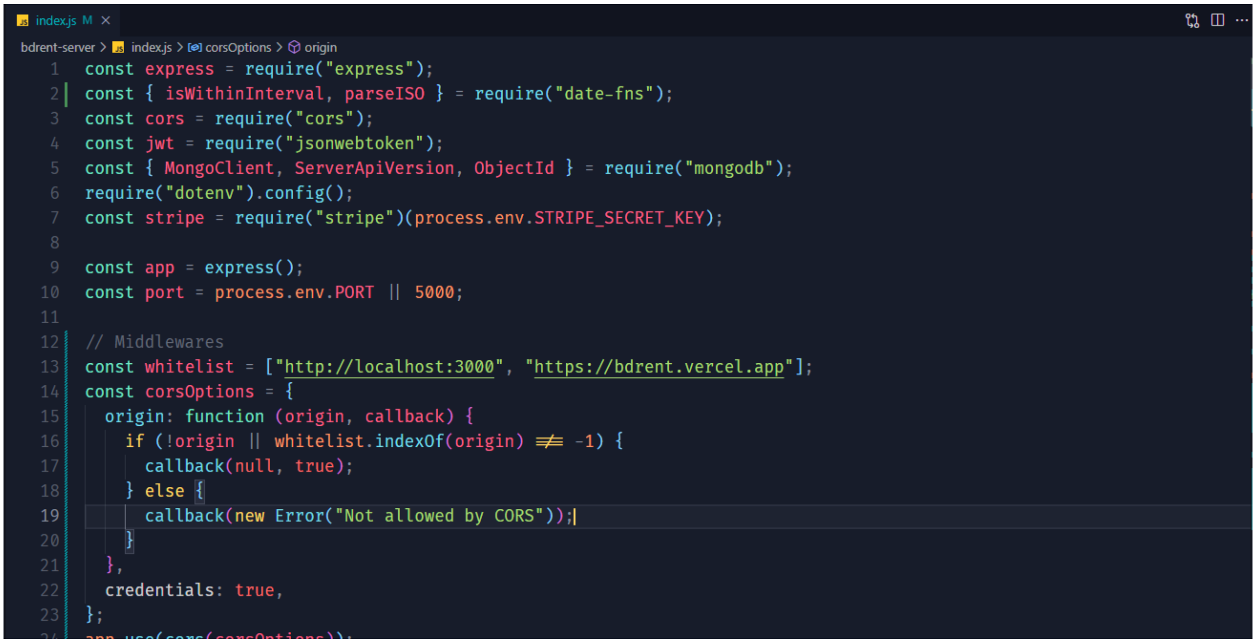
Figure 41: Email sign-in and Google sign-in



```javascript
import React, { useContext, useEffect, useState } from "react";
import { Navigate } from "react-router-dom";
import Spinner from "../Components/Spinner/Spinner";
import { getRole } from "../api/user";
import { AuthContext } from "../contexts/AuthProvider";

const HostRoute = ({ children }) => {
  const { user, loading } = useContext(AuthContext);
  const [role, setRole] = useState(null);
  const [roleLoading, setRoleLoading] = useState(true);
  useEffect(() => {
    setRoleLoading(true);
    getRole(user?.email).then((data) => {
      setRole(data);
      setRoleLoading(false);
    });
  }, [user]);

  if (loading || roleLoading) {
    return (
      <div className="h-screen">
        <Spinner />
      </div>
    );
```

Figure 42*: Host route for an authenticated user with role host*

Server code:



```javascript
const express = require("express");
const { isWithinInterval, parseISO } = require("date-fns");
const cors = require("cors");
const jwt = require("jsonwebtoken");
const { MongoClient, ServerApiVersion, ObjectId } = require("mongodb");
require("dotenv").config();
const stripe = require("stripe")(process.env.STRIPE_SECRET_KEY);

const app = express();
const port = process.env.PORT || 5000;

// Middlewares
const whitelist = ["http://localhost:3000", "https://bdrent.vercel.app"];
const corsOptions = {
  origin: function (origin, callback) {
    if (!origin || whitelist.indexOf(origin) !== -1) {
      callback(null, true);
    } else {
      callback(new Error("Not allowed by CORS"));
    }
  },
  credentials: true,
};
app.use(cors(corsOptions));
```

Figure 43*: From database connection to jwt token setup and all other things*

Chapter 11 – Testing

BD Rent is a web-based house booking website. While developing the project, for frontend development, I used React and React Router. For UI design I have used different components from the Tailwind CSS framework. On the backend, I used Node.js as the server runtime and utilized its framework, Express.js, to build a robust backend server. For user authentication, I integrated Firebase. Also, I implemented JSON Web Tokens (JWT) for authorization, which allowed me to validate and manage user access permissions. To store and retrieve data, I used MongoDB, a NoSQL database. For handling payments, I integrated Stripe, sslcommerz.

Project Name	BD Rent
Name of Product	https://bdrent.vercel.app/
Product Description	House booking website and management system
Project Technology	Web based react framework and MongoDB, NoSQL used.
Project Duration	10 Feb 2023 – 14 Jun 2023

11.1 Test Plan Acceptance

Like any other test plan, the Acceptance test plan also includes some components like Scope, Approach, Test Environment, Resources, Responsibilities,

There are two types of testing, I have followed. I will discuss about them in below:

- Functional Testing

- Non-functional Testing

Functional Testing

Functional testing is an essential technique in software development that enables Quality Assurance (QA) professionals to evaluate the behavior of software applications according to predefined requirements. Black-box testing approaches are used in functional testing to ensure that the tester is unaware of the internal system logic. The primary objective of functional testing is to determine whether the system behaves as expected. In this essay, I will demonstrate why functional testing is critical in software development and discuss various approaches that can be used to ensure that applications meet pre-defined requirements.

- **Unit Testing**

✓ Validation of form fields for input fields
✓ Show all the house listing
✓ Check whether search filter working perfectly

- **Module Testing**

✓ Create a new account
✓ Log in with the invalid data.

- **Integration Testing**

✓ System login with appropriate credentials
✓ Admin details add and edit the information.

Non-functional Testing

Functional testing ensures that the product’s features and capabilities perform as intended, non-functional testing plays a crucial role in optimizing system performance, usability, reliability, and other aspects vital to a product's success.

- Security Testing
- Accessibility Testing
- Usability Testing

11.2 Test Case

A test case is a fundamental aspect of software testing. It provides a systematic approach to ensuring that a specific software testing purpose is met, such as evaluating a certain program path or ensuring a given requirement is fulfilled. Test cases include input restrictions, plans, processes, and expected outcomes.

Unit Test – test case

Test Case Name	Unit Test		
Test Class			
Test Description			
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**

Table 11: Unit Test – test case

Module Test – test case

Test Case Name	Unit Test		
Test Class			
Test Description			
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**

Table 12: Module Test – test case

Integration Test – test case

Test Case Name	Unit Test		
Test Class			
Test Description			
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**

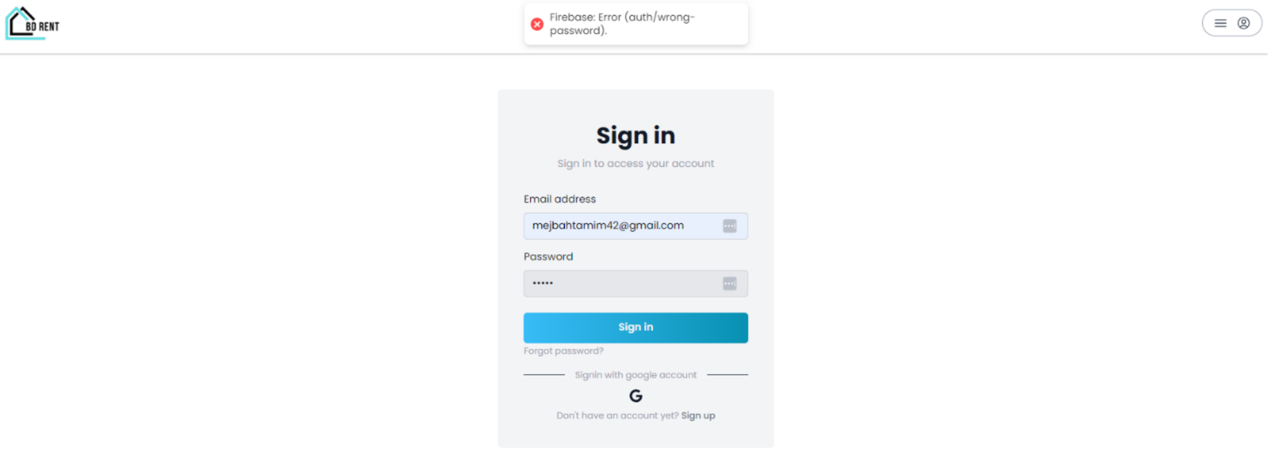
Table 13*:* Integration Test – test case

11.3 Unit Testing

Unit Test 1

Test Case Name	Unit Test		
Test Class	Admin login controller		
Test Description	Show wrong password notification for wrong credential		
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**
User Entry	1. Enter email address in the email field. 2. Enter password in the password field	A notification will be seen if the login credential is wrong	A notification will be given to give right login details

Table 14: Unit Test 1 – test case


BD RENT
Sign in
Sign in
Forgot password?
Signin with google account
G
Don't have an account yet? Sign up

Firebase: Error (auth/wrong-password).

Sign in to access your account

Email address

mejbahtamim42@gmail.com

Password

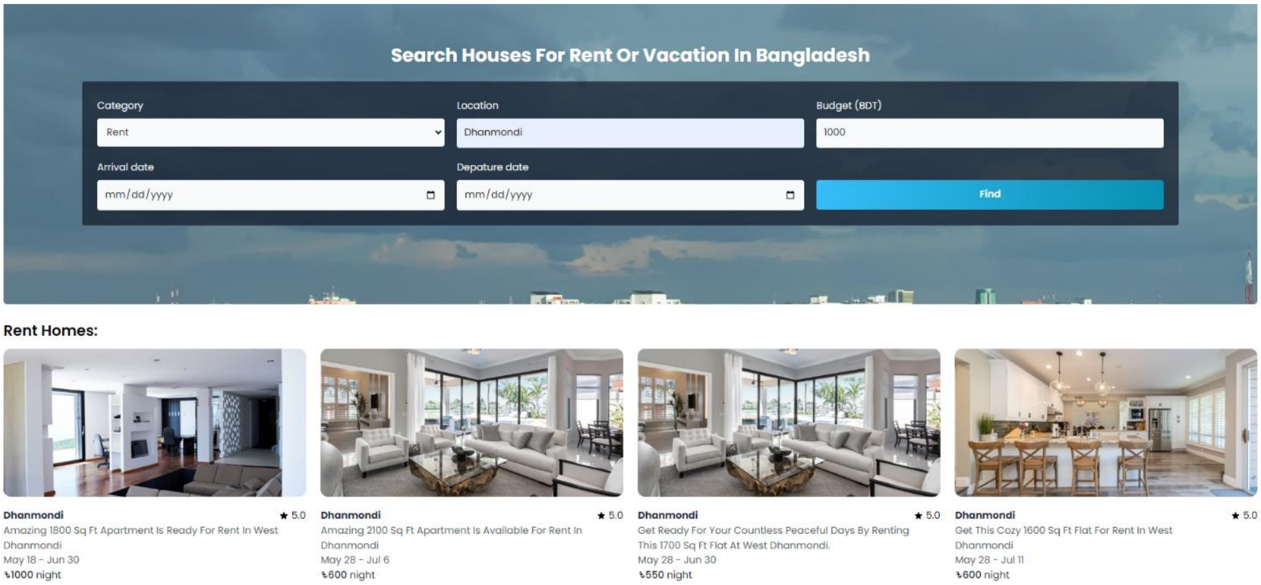
•••••

Figure 44: Unit Test Result - 1

Unit Test 2

Test Case Name	Unit Test		
Test Class	House listing		
Test Description	House listing based on the search		
Source of Data	Test Steps	Expected Result	Actual Result
User Entry	Users will give requirement on search bar house listing will be show based on that	All the house listing will be shown based on the search requirements	All the house listing will be shown based on the search requirements

Table 15: Unit Test 2 – test case


Search Houses For Rent Or Vacation In Bangladesh

Category: Rent

Location: Dhanmondi

Budget (BDT): 1000

Arrival date: mm/dd/yyyy

Depature date: mm/dd/yyyy

Rent Homes:

Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 18 - Jun 30
৳1000 night

Dhanmondi ★ 5.0
Amazing 2100 Sq Ft Apartment Is Available For Rent In Dhanmondi
May 28 - Jul 6
৳600 night

Dhanmondi ★ 5.0
Get Ready For Your Countless Peaceful Days By Renting This 1700 Sq Ft Flat At West Dhanmondi.
May 28 - Jun 30
৳550 night

Dhanmondi ★ 5.0
Get This Cozy 1600 Sq Ft Flat For Rent In West Dhanmondi
May 28 - Jul 11
৳600 night

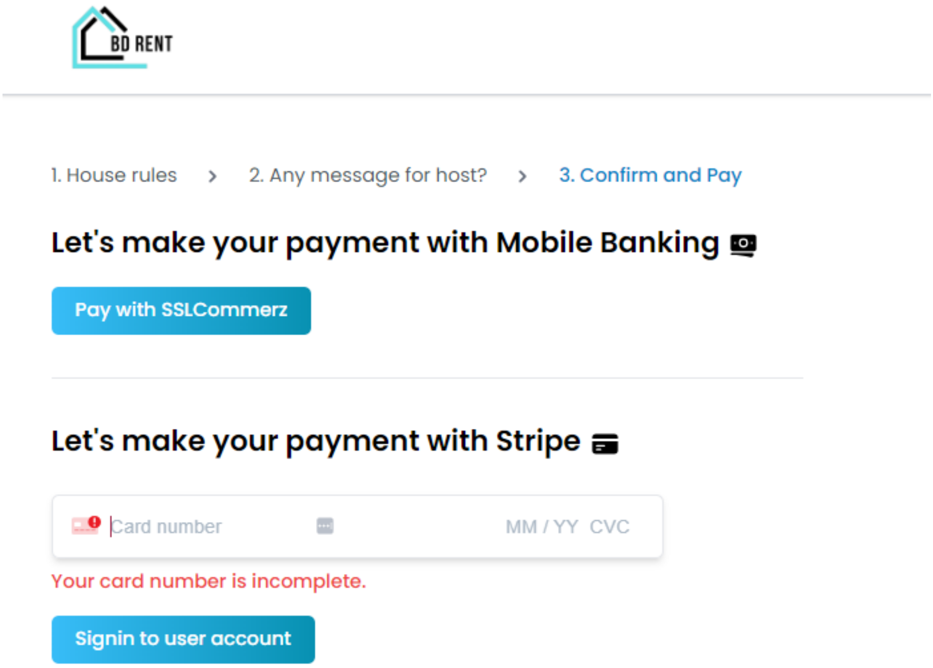
Figure 45: Unit Test Result - 2

11.4 Module Testing

Module Test 1

Test Case Name	Unit Test		
Test Class	The controller of Payment creation.		
Test Description	Make a payment attempt without entering any information.		
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**
User Entry	Go to the payment => fees head or if fees head inserted then go to Head Structure and click add record.	This field is required alert will be appeared for the required field.	Alert appears that the field is required for creating on the Fees Head or others option.

Table 16: Module Test 1 – test case



Let's make your payment with Mobile Banking

Let's make your payment with Stripe

Your card number is incomplete.

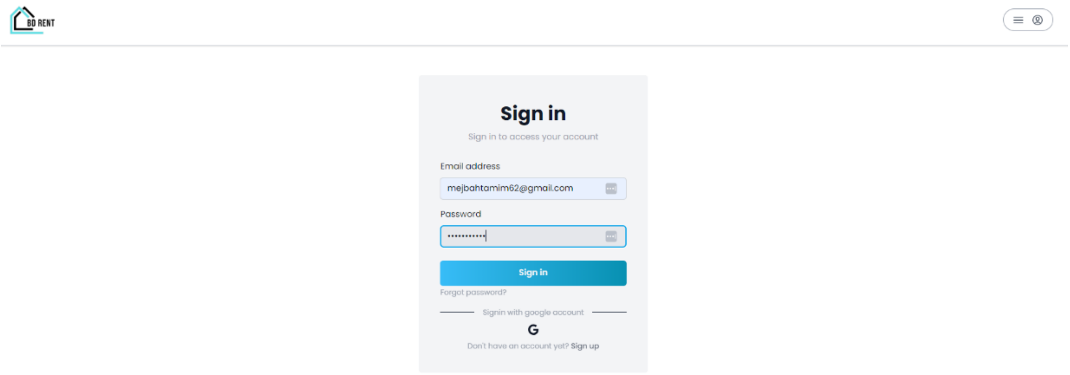
Figure 46: Model test 1 result

11.5 Integration Testing

Integration Test 1

Test Case Name	Unit Test		
Test Class	Login Authentication		
Test Description	Test Description Login with valid data and dashboard redirect.		
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**
User Entry	1. Go to the login page. 2. After entering correct login details. 3. Click on the sign in page.	If the credential is correct it will redirect to the dashboard.	If the credential is correct it will redirect to the dashboard.

Table 17: Integration Test 1 – test case



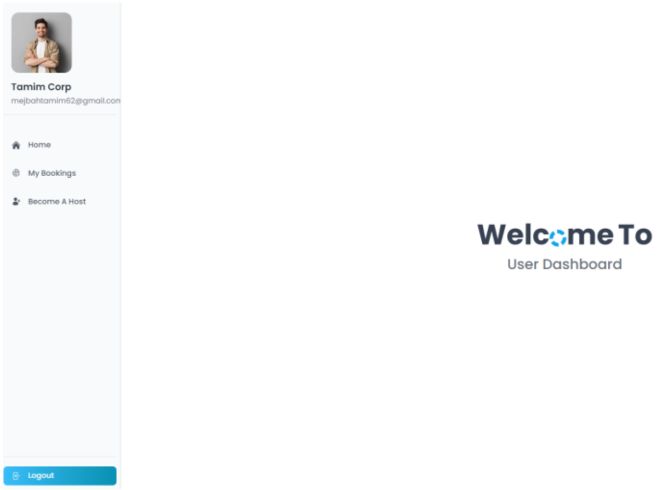
Sign in

Sign in to access your account

Email address: mejbahtamim62@gmail.com

Password: •••••••••••

Forgot password?
Signin with google account
G
Don't have an account yet? Sign up



Tamim Corp

mejbahtamim62@gmail.con

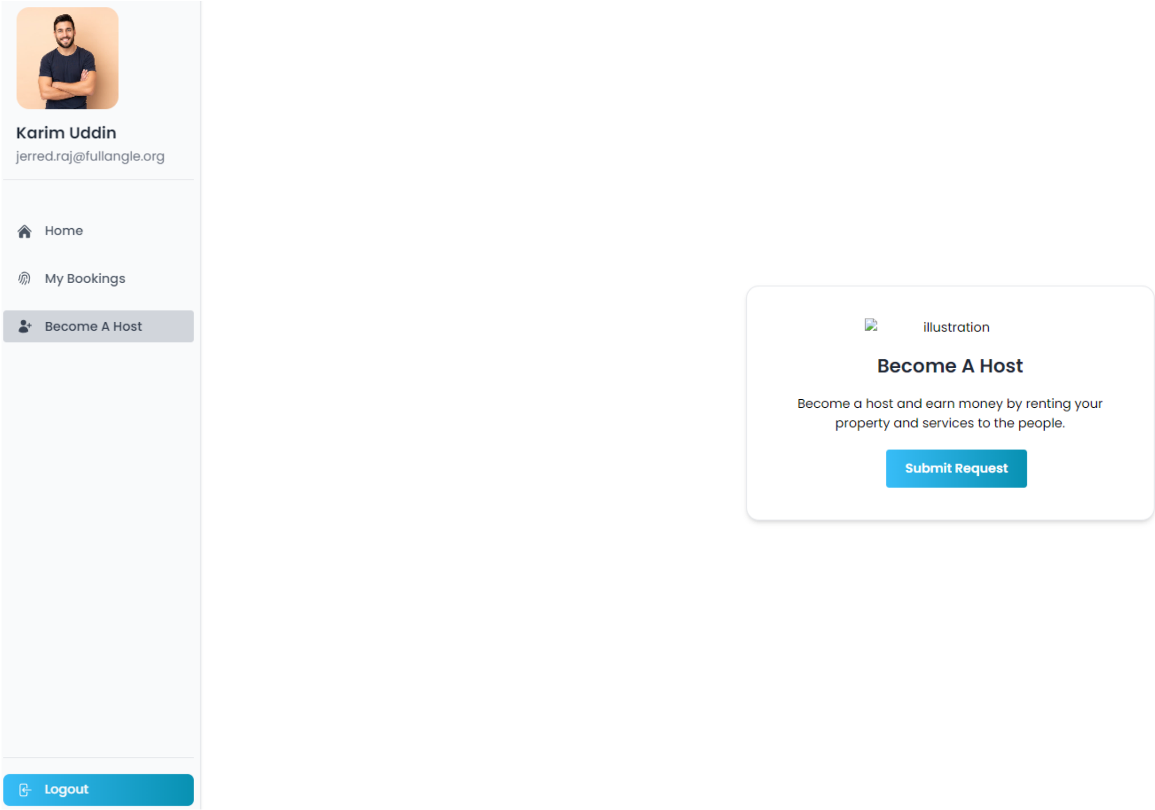
Welcome To

User Dashboard

Integration Test 2

Test Case Name	Unit Test		
Test Class	Users Host Request Control		
Test Description	User host request check and accept		
Source of Data	**Test Steps**	**Expected Result**	**Actual Result**
User Entry	1. Go to dashboard. 2. Click on Become a host	A host request will go to admin	If host request accepted, user can upload his house listing

Table 18: Integration Test 2 – test case



Become a host and earn money by renting your property and services to the people.

Chapter 12 – Implementation

12.1 Training

Upon the introduction of a new software product in the market, it is crucial to guide its users on proper usage. As such, thorough training becomes essential after software implementation to ensure optimal productivity, minimize potential issues, and enhance user satisfaction and morale from the outset. Once a software solution is deployed, there are several approaches to delivering training resources to computer system users, along with additional methods for its ongoing maintenance. It is vital to consider key aspects of the organization when formulating a plan for effective execution. The subsequent table outlines the training procedures for the BD Rent system:

Sr. No.	User	Training Scope	Time Allocation	Comment
1	Admin	• Check all the user's details • Check all the booking • Check all the house listing • Accept user's becoming host request. • Cancel any booking	4 hours	Admin is familiar with all of the functions and operators.
2	User	• Create user account • Make payment • Book house • Edit booking details • Request to become host • Upload house listing • Edit house listing • Manage bookings	5 hours	User is familiar with all of the functions and operators.

12.2 Big Bang

The primary benefit of big bang adoption is that the suggested software doesn't have to be compatible with or integrated into any existing systems. This greatly simplifies the overall design of the new system, which is particularly advantageous in an organization with several disparate systems. In terms of acquisition approaches, the big bang is the costliest, despite its limited learning opportunities and increased planning requirements. The planning process will be explained through the systematic approach of such extensive bang adoption. Big bang methodology works by swiftly decommissioning the previous system and implementing the new one. This method is significantly quicker than other tactics since it utilizes the updated system immediately after testing is finished. However, transferring data from the old system can create risks and uncertainties, as data may be compromised and the intended software could potentially fail.

12.3 Scaling

There can be lots of scaling scope for BD Rent. Like adding more categories, adding apartment buy sell option.

12.4 Load Balancing

Load balancing, also known as load equalization, counteracts the impact of a system's users and effectively distributes incoming network traffic across a collection of backend servers. This process involves monitoring the number of users accessing the server and measuring overall system usage. If one server becomes non-operational, the load balancer reroutes traffic to the remaining functional servers. As new servers join the group, the load balancer automatically delegates requests to them, ensuring seamless system performance.

Chapter 13 – Critical Appraisal and Evaluation

13.1 Objective Could be Met

No system is flawless; they all have room for improvement. Continuous upgrades are always an option. I have exerted myself to achieve all the crucial objectives that allow a school to operate this system effectively. Nonetheless, some specific needs may still need to be met. These include:

- Registration for Admin
- Fees payment management
- More categories

13.1.2 How much better could have been done

The system consistently achieves its objectives with stellar success rates. In order to enhance user-friendliness, we've meticulously collected and organized all pertinent information in a more effective manner. All users and hosts incorporate comprehensive information about themselves and adeptly address any issues that arise. Up until the demonstration, the payment processing has been deemed sufficient for this project's requirements.

13.2 Why it could not be done

All the key features have been added on the project. The reason I didn’t focus on other feature like house buying selling option because I want to make sure focus on the main feature which is house booking and managing.

13.3 Which features could not be touched

All of the features required for a house booking and management system are present in the system. its have online payment recode store. But with this system, the online payment transaction by bank or others methods is unaffected.

Chapter 14 – Lessons Learned

14.1 Pre-Project-Review-closing

BD Rent is a web-based project. To build this system, I followed a well-defined approach, which involved understanding the user’s needs, devising a plan for the system's creation, selecting a suitable system architecture, and deciding on a project title. I used React framework for Front end and for backend I have used MongoDB and NoSQL to build the system. I have created project-specific documentation. All the house booking is features have been used in BD Rent project.

14.2 What I have learned

Throughout the process of developing "BD Rent," my web application for house searching and booking, I have gained invaluable insights and learned numerous lessons. Building a project of this magnitude has presented its fair share of challenges and opportunities for growth. Here are some key takeaways from my experience:

- User Experience is Paramount: Creating an intuitive and user-friendly interface is essential to ensuring a seamless experience for both renters and homeowners. I have learned the significance of thorough user testing and continuous feedback loops to refine the application's usability.

- Efficient Search Algorithms: Developing an efficient search algorithm to help users find their ideal homes quickly has been a significant challenge. I have learned to leverage data structures and optimize search queries to deliver accurate and speedy results while considering factors such as location, price range, and amenities.

- Streamlined House Listing Management: For homeowners, managing their property listings efficiently is vital. I have incorporated features that allow homeowners to update property details, availability, and pricing easily, ensuring their listings remain up-to-date and accurate.

14.3 The problems I have faced

Developing the "BD Rent" project, an all-inclusive online platform for searching and booking houses, has been a thrilling and demanding experience. During the construction of this project, I've faced numerous hurdles along the path. Here are some issues I confronted throughout the development phase:

Scalability: As the number of users escalated rapidly, the application found it difficult to cope with the growing server load. It became imperative to expand the infrastructure and refine the codebase to maintain efficiency and responsiveness. By adopting caching systems, optimizing databases, and incorporating load balancing strategies, these problems were addressed effectively, guaranteeing a smooth user experience even during high-traffic periods.

Security: Safeguarding user data and preventing unauthorized entry presented a significant challenge. It was vital to apply strong authentication and authorization methods, encrypt sensitive information, and follow top-notch security practices to safeguard user details and uphold the confidence of both property owners and renters.

Support and Maintenance: To offer swift and dependable assistance to users and tackle their concerns promptly, it was necessary to create effective communication channels. Establishing a dedicated customer support crew and integrating a ticketing scheme aided in streamlining problem-solving processes and guaranteeing client satisfaction. Periodic updates, bug resolutions, and enhancement of performance were essential for preserving the application's stability and functionality over time.

14.4 What Solution Occurred

Addressing issues that arise during the development process is crucial, as it not only resolves problems but also enhances problem-solving skills. I successfully developed solutions for each of the challenges mentioned earlier and completed the project within the given timebox, adhering to the DSDM methodology deadline.

In developing the project, I made sure to first analyze the needs of the web developer thoroughly. My goal was to create an project with a simple format for easy parsing and efficient use in mobile development. Creating such a large-scale project was a significant challenge for me because my priority was to write clean, comprehensible code, making future development seamless for other developers.

I implemented JSON Web Tokens (JWT) for authorization, which allowed me to validate and manage user access permissions.

Chapter 15 – Conclusion

15.1 Summary of the project

BD Rent is a house booking and management system which aim to provide users to find home according to their requirement and on the other hand homeowners can upload their house listing and manage that. For the frontend development, I used React and React Router. To enhance the UI design, I utilized different components from the Tailwind CSS framework. On the backend, I relied on Node.js as the server runtime and utilized its framework, Express.js, to build a robust backend server. For user authentication, I integrated Firebase. Also, I implemented JSON Web Tokens (JWT) for authorization, which allowed me to validate and manage user access permissions. To store and retrieve data, I used MongoDB, a NoSQL database. For handling payments, I integrated Stripe, a popular payment processing platform.

15.2 Goal of the project

The main target of the BD Rent project is to connect user's and houseowner in an organized way. The main goals of the project are:

- Computerize all of the systems.
- Reduce the amount of time spent and the potential for inaccuracy.
- The management of every system is automated.
- Centralized database administration.
- Make easier for the users to find and book their next home.

15.3 Success of the project

I have developed a system that has all the features that home booking website required to have. It may be utilized effectively in the different ways. By using this project anyone can find their next home and for house owner to list their house for rent and earn money by renting their house.

15.4 What I have done in the documentation

In order to complete this documentation, I adhered to the required academic formatting and successfully accomplished all necessary tasks from the outset. Within this document, various diagrams have been incorporated to showcase the progression of distinct features. All essential information, as well as tables, were compiled utilizing Microsoft Word, while figures and diagrams were designed with the aid of external libraries for effective time management, numerous illustrations, and data analysis.

15.5 Value of the Project

Completing tasks manually can be quite challenging, requiring significant time and effort. However, advancements in technology have revolutionized our ability to automate processes, leading to time and effort savings. One such example is the BD Rent system, which enables users to find their home and book that home efficiently. On the other hand, home owner can easily become host by submitting necessary documents. This enables home owners to list this house and make money by easily renting their house.

15.6 My Experience

BD Rent project is a unique for me because I never this type of full stack project. While completing the project I have faced different types of challenges and fixed those in a efficient ways. While building the project I learned about my things like how to improve UI/UX in the project, how to build the project according to the requirements. While building the project it gave me lots of experience throughout the development.

Appendix

User Guide

To log in to the system, enter your login credentials.

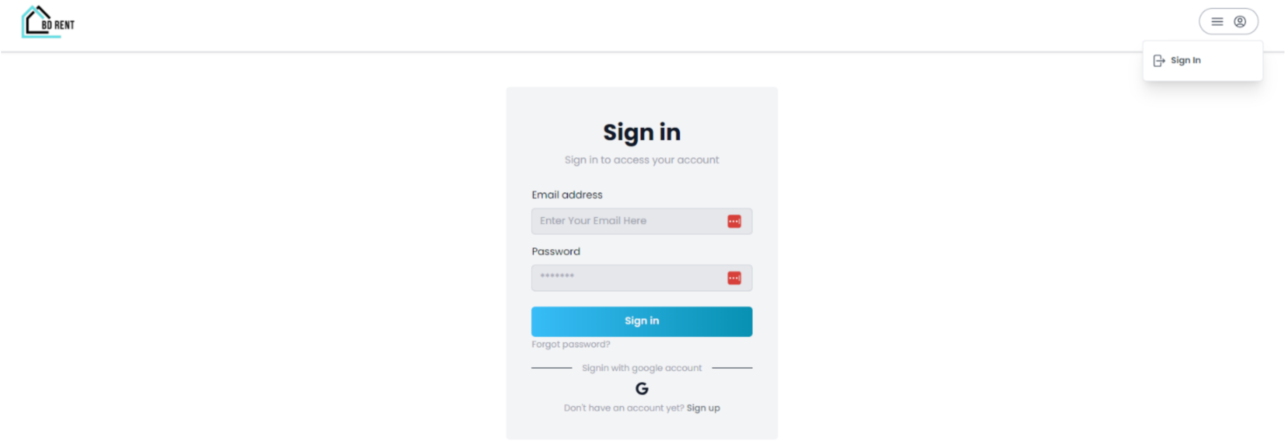


Figure 47: User guide for login

After providing the correct credentials to log in, the dashboard will be apparent to you.

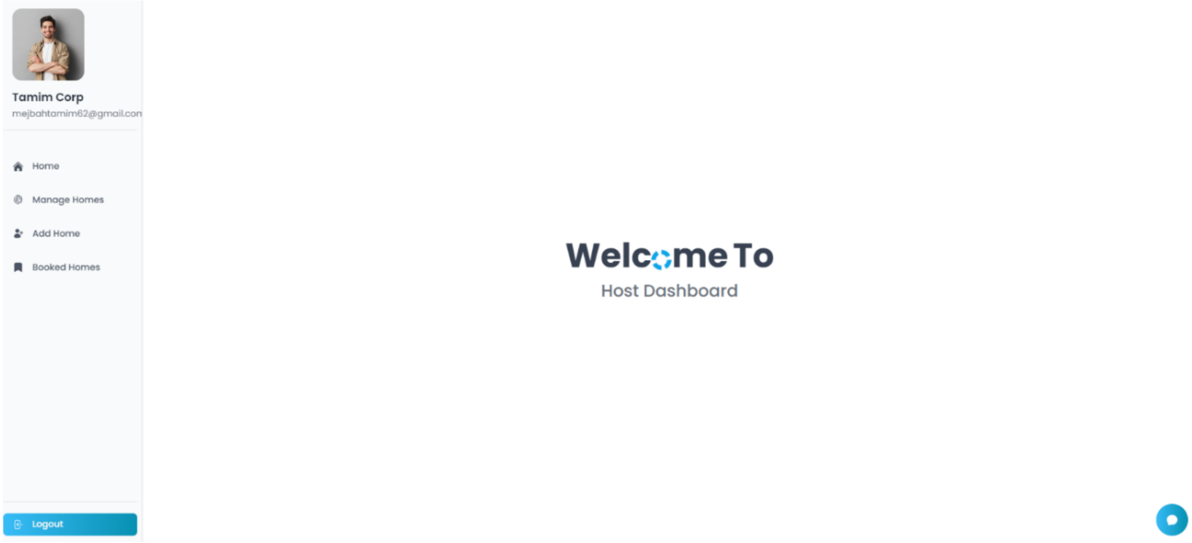
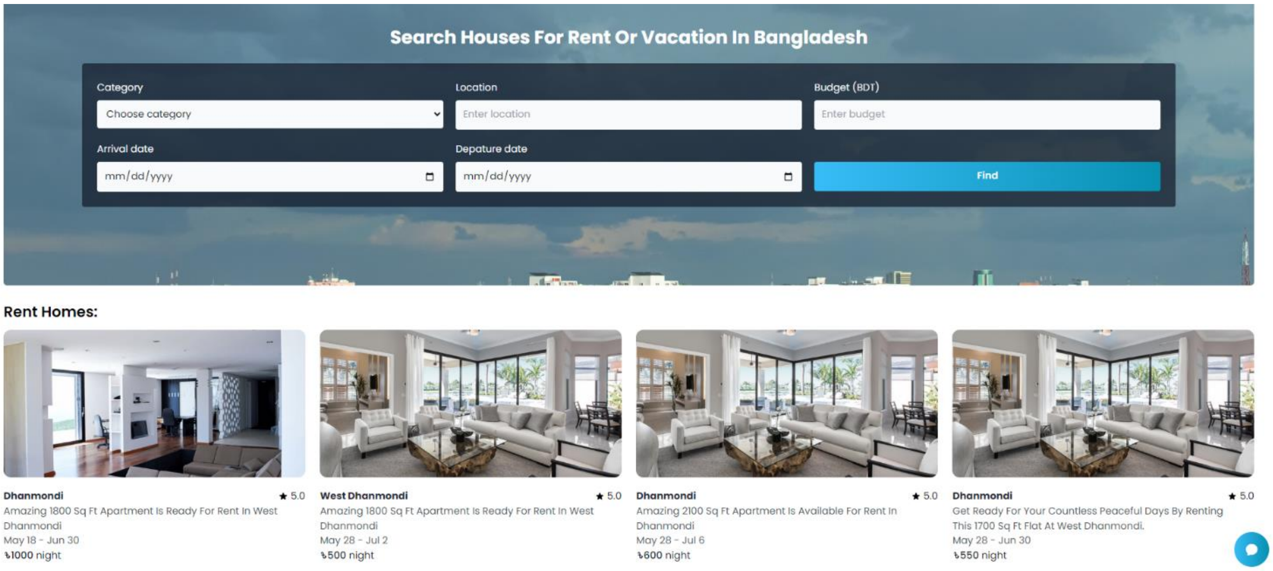


Figure 48*: User guide for user dashboard*

Users can search for house listing based on his requirements.



Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 18 - Jun 30
৳1000 night

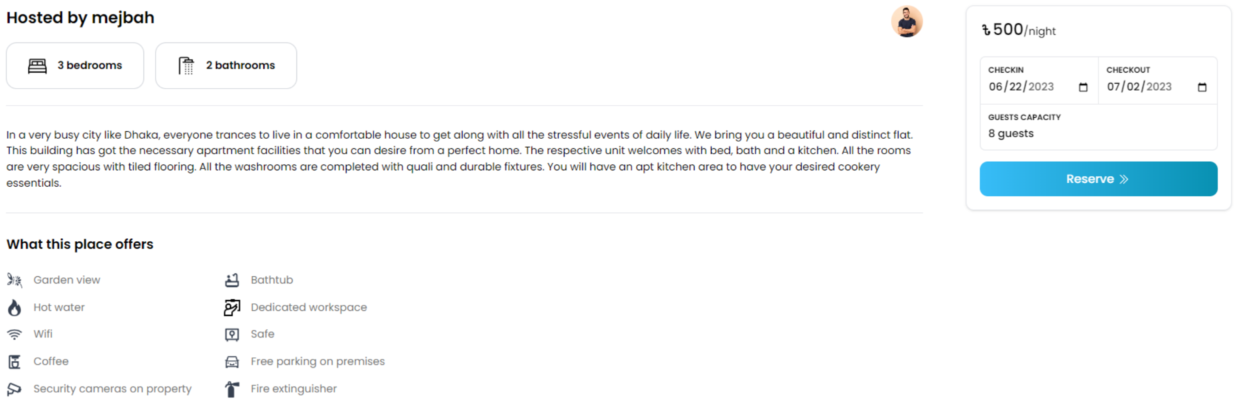
West Dhanmondi ★ 5.0
Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi
May 28 - Jul 2
৳500 night

Dhanmondi ★ 5.0
Amazing 2100 Sq Ft Apartment Is Available For Rent In Dhanmondi
May 28 - Jul 6
৳600 night

Dhanmondi ★ 5.0
Get Ready For Your Countless Peaceful Days By Renting This 1700 Sq Ft Flat At West Dhanmondi.
May 28 - Jun 30
৳550 night

Figure 49*: User guide for search filter*

If user like any house listing, he can check the house details by clicking on the listing



Hosted by mejbah

3 bedrooms

2 bathrooms

In a very busy city like Dhaka, everyone trances to live in a comfortable house to get along with all the stressful events of daily life. We bring you a beautiful and distinct flat. This building has got the necessary apartment facilities that you can desire from a perfect home. The respective unit welcomes with bed, bath and a kitchen. All the rooms are very spacious with tiled flooring. All the washrooms are completed with quali and durable fixtures. You will have an apt kitchen area to have your desired cookery essentials.

What this place offers

- Garden view
- Hot water
- Wifi
- Coffee
- Security cameras on property
- Bathtub
- Dedicated workspace
- Safe
- Free parking on premises
- Fire extinguisher

৳500/night

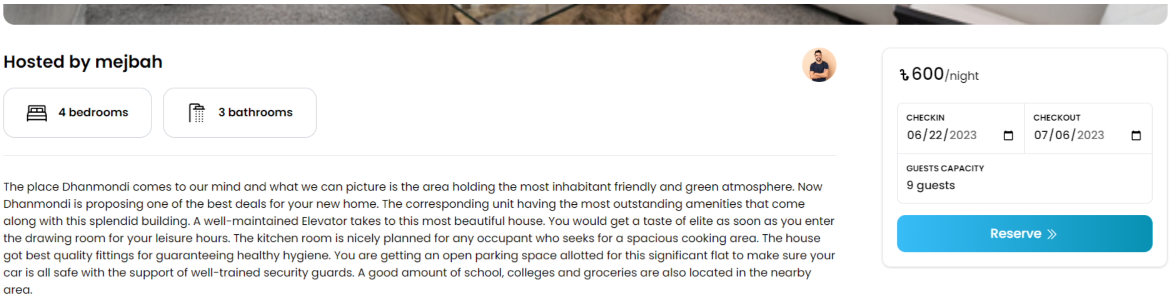
CHECKIN 06/22/2023

CHECKOUT 07/02/2023

GUESTS CAPACITY 8 guests

Figure 50*:* User guide for house details

User can select what will be his check in and check out time. Based on the house availability.



Hosted by mejbah

4 bedrooms

3 bathrooms

The place Dhanmondi comes to our mind and what we can picture is the area holding the most inhabitant friendly and green atmosphere. Now Dhanmondi is proposing one of the best deals for your new home. The corresponding unit having the most outstanding amenities that come along with this splendid building. A well-maintained Elevator takes to this most beautiful house. You would get a taste of elite as soon as you enter the drawing room for your leisure hours. The kitchen room is nicely planned for any occupant who seeks for a spacious cooking area. The house got best quality fittings for guaranteeing healthy hygiene. You are getting an open parking space allotted for this significant flat to make sure your car is all safe with the support of well-trained security guards. A good amount of school, colleges and groceries are also located in the nearby area.

৳600/night

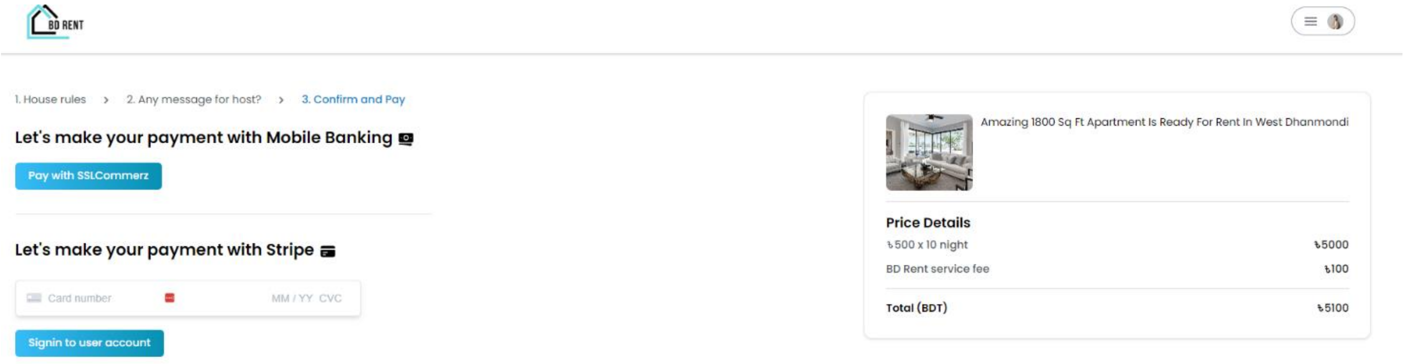
CHECKIN 06/22/2023

CHECKOUT 07/06/2023

GUESTS CAPACITY 9 guests

Figure 51*:* User guide for booking check in out selector

After clicking on the Reserve button user will be redirected to a payment page. Where he can pay using different payment method. User can pay and book the house.



1. House rules › 2. Any message for host? › 3. Confirm and Pay

Let's make your payment with Mobile Banking

Pay with SSLCommerz

Let's make your payment with Stripe

Card number MM / YY CVC

Signin to user account

Amazing 1800 Sq Ft Apartment Is Ready For Rent In West Dhanmondi

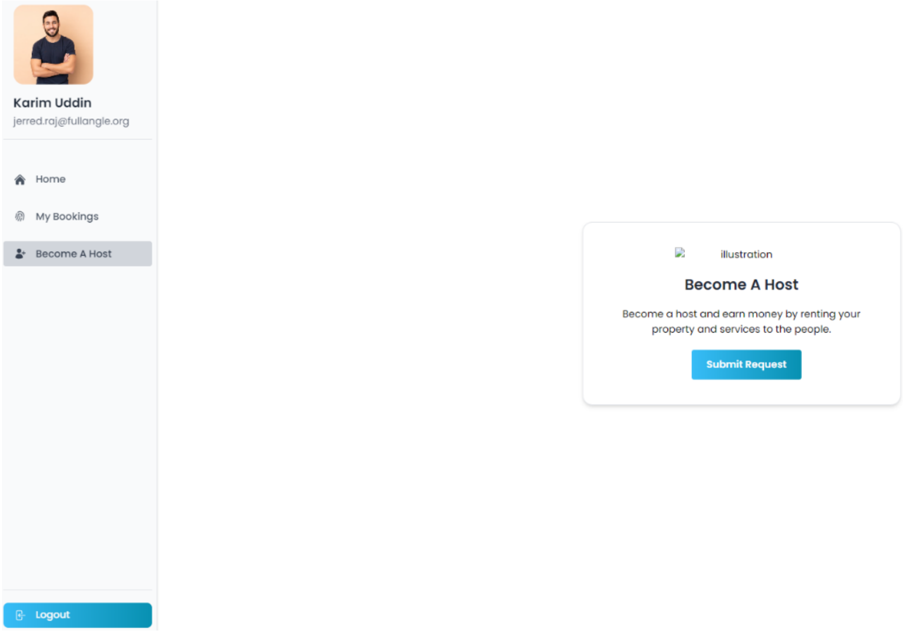
Price Details

৳500 x 10 night	৳5000
BD Rent service fee	৳100
Total (BDT)	৳5100

Figure 52*: User guide for make payment*

Host Guide

If any user wants to be host and upload house listing, he can click on "Become A Host" and submit request to become host.



Karim Uddin

jerred.raj@fullangle.org

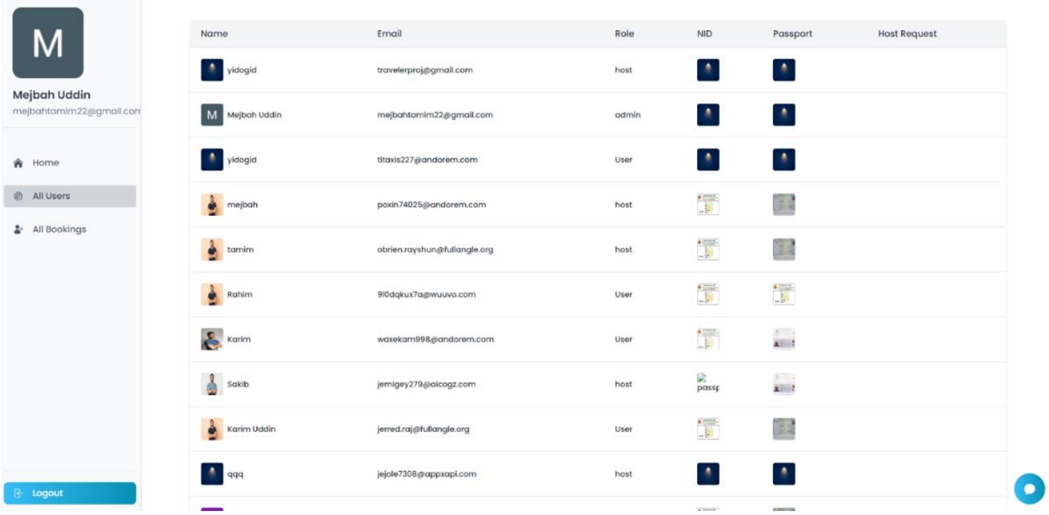
Become A Host

Become a host and earn money by renting your property and services to the people.

Submit Request

Figure 53*: User guide for apply to become host*

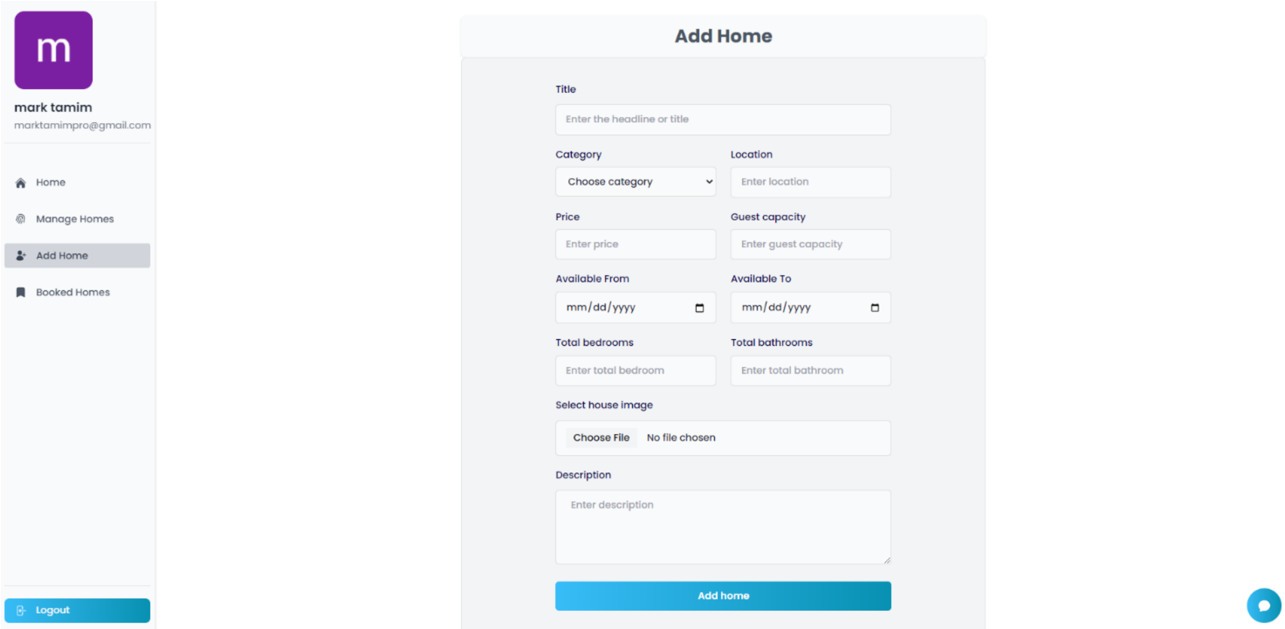
After applying to become host, a host request will be go admin. Here admin can decide whether user can become host or not



Name	Email	Role	NID	Passport	Host Request
yidogid	travelerproj@gmail.com	host			
Mejbah Uddin	mejbahtamim22@gmail.com	admin			
yidogid	titaxis227@andorem.com	User			
mejbah	poxin74025@andorem.com	host			
tamim	obrien.rayshun@fullangle.org	host			
Rahim	9l0dqkux7a@wuuvo.com	User			
Karim	waxekam998@andorem.com	User			
Sakib	jemigey279@aicogz.com	host	passp		
Karim Uddin	jerred.raj@fullangle.org	User			
qqq	jejole7308@appxapi.com	host			

Figure 54: User guide for admin approval for host

After becoming host, user can upload house listing. To upload house listing host, need to click on "Add Home" button. And a form will be show where users can put all the details of his and create a house listing.



Add Home

Title
Enter the headline or title

Category: Choose category
Location: Enter location

Price: Enter price
Guest capacity: Enter guest capacity

Available From: mm/dd/yyyy
Available To: mm/dd/yyyy

Total bedrooms: Enter total bedroom
Total bathrooms: Enter total bathroom

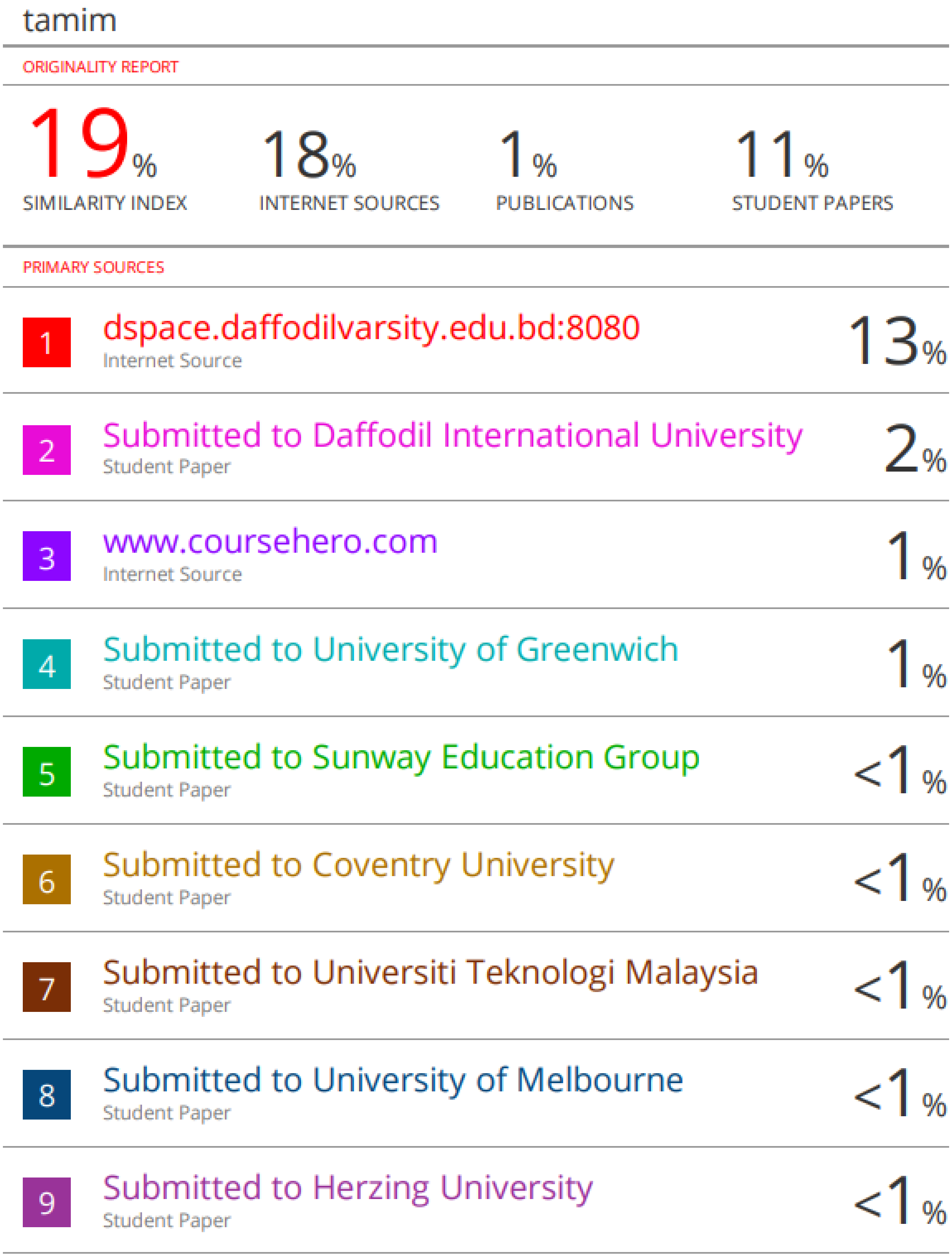
Select house image
Choose File No file chosen

Description
Enter description

Add home

Figure 55: User guide for adding new

Plagiarism Report



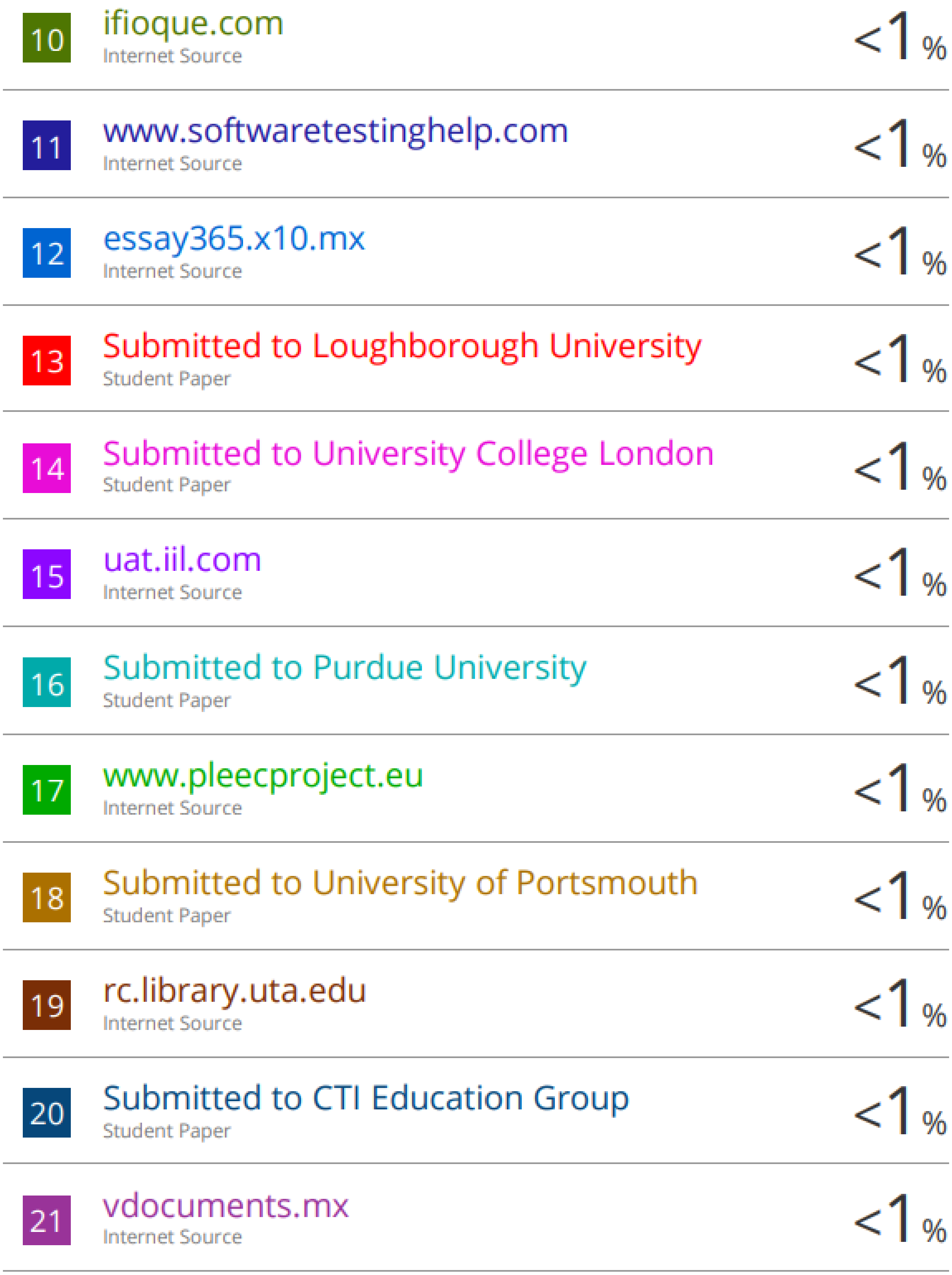
tamim

ORIGINALITY REPORT

19%	18%	1%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	13%
2	Submitted to Daffodil International University Student Paper	2%
3	www.coursehero.com Internet Source	1%
4	Submitted to University of Greenwich Student Paper	1%
5	Submitted to Sunway Education Group Student Paper	<1%
6	Submitted to Coventry University Student Paper	<1%
7	Submitted to Universiti Teknologi Malaysia Student Paper	<1%
8	Submitted to University of Melbourne Student Paper	<1%
9	Submitted to Herzing University Student Paper	<1%



10	ifioque.com Internet Source	<1 %
11	www.softwaretestinghelp.com Internet Source	<1 %
12	essay365.x10.mx Internet Source	<1 %
13	Submitted to Loughborough University Student Paper	<1 %
14	Submitted to University College London Student Paper	<1 %
15	uat.iil.com Internet Source	<1 %
16	Submitted to Purdue University Student Paper	<1 %
17	www.pleecproject.eu Internet Source	<1 %
18	Submitted to University of Portsmouth Student Paper	<1 %
19	rc.library.uta.edu Internet Source	<1 %
20	Submitted to CTI Education Group Student Paper	<1 %
21	vdocuments.mx Internet Source	<1 %