

Design and Implementation of Online To-Let System

By

Tanvir Ahamed
ID :172-15-9761

**This Report Presented in Partial Fulfillment of the
Requirements for the Degree of Bachelor of Science in
Computer Science and Engineering**

Supervised by

Ms. Nazmun Nessa Moon
Associate Professor
Department of Computer Science and
Engineering Daffodil International
University

Co-Supervised by

Mr. Golam Rabbany
Lecturer
Department of Computer Science and
Engineering Daffodil International
University



**DAFFODIL INTERNATIONAL
UNIVERSITY**
Dhaka, Bangladesh

September 17, 2025

APPROVAL

This Project titled Design and Implementation of Online To-let System, submitted by Name: **Tanvir Ahmed** ID No: **172-15-9761** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. In Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **17 September, 2025**.

BOARD OF EXAMINERS



.....
Dr. S.M Aminul Haque (SMAH)
Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



.....
Ms. Nazmun Nessa Moon (NNM)
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



.....
Md. Abbas Ali Khan (AAK)
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



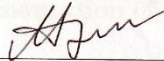
.....
Dr. Md. Zulfiker Mahmud (ZM)
Professor
Department of Computer Science and Engineering
Jagannath University

External Examiner

DECLARATION

I hereby declare that this project has been done by us under the supervision of **Ms. Nazmun Nessa Moon**, Associate Professor, Department of Computer Science and Engineering, Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Ms. Nazmun Nessa Moon

Associate Professor

Department of Computer Science and
Engineering Daffodil International
University

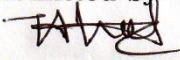
Co-Supervised by:

Mr. Golam Rabbany

Lecturer

Department of Computer Science and
Engineering Daffodil International
University

Submitted by:



Tanvir Ahamed

Student ID: 172-15-9761

Department of Computer Science and
Engineering Daffodil International
University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. I am deeply grateful to everyone who has assisted us in one way or another.

First, I express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Design and Implementation of Online To-Let system** successfully.

I am grateful and wish our profound indebtedness to **Ms. Nazmun Nessa Moon Associate Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Web development** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

I would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

I would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, I must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

The increasing demand for rental spaces such as houses, sub-let flats, offices, garages, shops, and showrooms highlights the limitations of traditional to-let systems, which often rely on physical advertisements, personal networks, or agents and thus consume significant time and effort, especially for expatriates and individuals relocating. This project is called Design and Implementation of Online To-Let System, and it introduces an online system that will make the process of finding rental houses and related services easier and more modern. The system is to be responsive and easy to use so that a user can search, view, and select properties anywhere and at any time. The research design was to examine the current methods in the manuals, interview a number of people working in international body, and determine what are the problems encountered in the current practices. Its application utilizes HTML, CSS, PHP, Laravel, SQL, MySQL, JavaScript, to create an effective, safe, and scalable system. Not only does the proposed system save time and money of searching a good rental offer, but the rental market is better accessible and transparent. The study reveals that a combination of innovative web technologies and real-life rental demands can dramatically improve the effectiveness of property searching and management processes, thus leading to the further increase in user satisfaction and service delivery

Table of Contents

Approval	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Introduction.....	1
1.2 Motivation	2
1.3 Objectives	2
1.4 Project Outcome.....	3
1.5 Organization of the Report	3
2 Background	5
2.1 Introduction.....	5
2.2 Literature Review	5
2.2.1 Similar Applications	7
2.3 Gap Analysis.....	8
2.4 Summary	9
3 Research Methodology	10
3.1 Requirement Analysis & Design Specification.....	10
3.1.1 Overview	10
3.1.2 System Design.....	10
3.1.3 Functional and Nonfunctional Requirement.....	17
3.1.4 Data Flow Diagram.....	18
3.1.5 UI Design.....	19
3.2 Detailed Methodology and Design.....	24
3.3 Project Plan.....	24

3.4	Task Allocation.....	25
3.5	Summary	26
4	Implementation and Results	28
4.1	Environment Setup	28
4.2	Testing and Evaluation.....	29
4.3	Results and Discussion	30
4.4	Summary	31
5	Engineering Standards and Design Challenges	32
5.1	Compliance with the Standards.....	32
5.1.1	Software Standards.....	32
5.1.2	Hardware Standards	33
5.1.3	Communication Standards.....	34
5.2	Impact on Society, Environment and Sustainability	35
5.2.1	Impact on Life.....	35
5.2.2	Impact on Society & Environment.....	36
5.2.3	Ethical Aspects	36
5.2.4	Sustainability Plan.....	37
5.3	Project Management and Financial Analysis.....	37
5.4	Complex Engineering Problem.....	39
5.4.1	Complex Problem Solving.....	40
5.4.2	Engineering Activities.....	40
5.5	Summary	43
6	Conclusion	44
6.1	Summary	44
6.2	Limitation	44
6.3	Future Work.....	45
	References	47

List of Figures

Figure 3.1 Database Schema diagram	11
Figure 3.2 Er diagram	14
Figure 3.3 Use Case diagram.....	16
Figure 3.4 DF Diagram	18
Figure 3.5 Ui Designm	20
Figure3.6 Admin Interface.....	21
Figure 3.7 Font-End Interface.....	23

List of Tables

Table 2.1 Summary of Literature Reviewed.....	6
Table 5.1 Mapping with complex problem solving.....	40
Table 5.2 Mapping with knowledge Profile.	40
Table 5.3 Mapping with complex engineering activities.....	43

Chapter 1

Introduction

1.1 Introduction

The Online To-Let System allows landlords to list their properties with detailed descriptions, images, rental prices, and availability status. Tenants, on the other hand, can browse properties, apply search filters based on their preferences, and contact landlords directly through the platform.

The system exists to deliver instant information alongside alert capabilities and facility tools which improve user comfort for all participants.

The system achieves accuracy and efficiency and delivers effortless access through its implementation of property listing and search filters and user authentication and booking request and notification capabilities. Controlling user information, the manufacturers control the listings of the real estates and keep the operations of the platforms running. The Online To-Let System offers easy usage to the user with the potential to access the site on a broad variety of devices, and its design permits its development to online payments and AI-aided recommendations.

The principal objective of this system is to provide an optimized rental system that improves the search and listing process as well as reducing time-consuming operations on these processes. Locating an adequate rental arrangement may be a difficult and time-consuming process and may entail archaic processes of getting it by physical advertisement, oral referrals and unreliable online listings.

The Online To-Let System is aimed at streamlining and computerizing this process with the aim of offering an efficient, easy-to-use, and transparent digital platform

1.2 Motivation

This project is inspired by the growing urge to get a more efficient and convenient means of locating and managing rental houses both in urban and rural regions. Our society is still heavily relying on manual communication, physical visits, and fragmented information to operate the traditional system of the rental business. This tends to waste time, additional expenditure and unwarranted inconvenience to the house owners as well as tenants.

All these shortcomings made us consider a piece of modernized technology that can make the process more efficient of renting and searching houses. In the current digital age, technology is expected to be incorporated to eliminate the boundaries of time, space, and accessibility.

The To-Let System web-site can guarantee that all people, no matter where they are, can easily get access to the information on renting a house, at the comfort of using their smartphone or computer that is connected to the internet. This saves time, besides rendering the exercise transparent, cost-effective and easy to both sides. Another incentive that helped us in our quest was the possibility of applying web technologies to the real world problem.

I was to create a platform that is reliable, secure, and easy to use with the help of system design, data management, and frontend-backend integration. Developing such a system posed a challenge to us to increase our technical expertise, collaboration and problem solving capability as I serve to the society. Most importantly, this project portrays our ideology that technology must be used to simplify the lives of people.

Our To-Let System aims to make the rental process a useful tool to every student, job holder, family, and house owner as it simplifies the process of rental and provides comfort, efficiency, and trust in the process, I am confident that this will make our system a valuable tool and assist many students, job holders, families, and house owners.

1.3 Objectives

Objectives aim to achieve in this phase of the project by me:

Create a defensive backend system with modern web frameworks which will properly handle user management together with property listing control and rental financial operations.

The system must include an improved search tool which enables tenants to find suitable properties by using location specifications together with price sets as well as property categories and building features.

Provide instant messaging services together with instant communication devices that provide clear communication channels between property owners and their would-be tenants.

Full testing and debugging will assist in detection of bugs thus users will be in a position to enjoy bug free systems.

1.4 Project Outcome

This project is expected to yield the following outcome: development of a functional and user-friendly Online To-Let System that can make the process of searching and managing the rental property much easier.

The system will provide the user with a secure system where he/she can locate houses, flats, offices, shops and other places where one can rent with ease courtesy of the well organized and easy to use interface. It will serve the property owners by offering them an easy way of listing and maintaining their spaces and renters with a quick and efficient means of accessing right and current information and thus saving time and energy that would be used to find the right accommodation. Besides a practical solution, the project result is also the theoretical development of a responsive and scalable framework that can be further expanded with other functions, including location-based search, notifications, or relocation service integration.

At the individual level, the result will be the improvement of technical knowledge, research capabilities, and problem-solving competencies, which will lead to academic development and subsequent career prospects.

1.5 Organization of the Report

This report is organized into several chapters to present the work in a structured and coherent manner. Chapter One introduces the project by providing background information, motivation, objectives, methodology, expected outcomes, and the overall organization of the report.

Chapter Two reviews related works and existing systems, highlighting their limitations and identifying the research gap that this project aims to address.

Chapter Three describes the system analysis and design, including requirement specifications, use case diagrams, and the architectural framework of the proposed solution.

Chapter Four focuses on the implementation details, presenting the development process, system modules, and user interface design.

Chapter Five discusses system testing, evaluation, and performance analysis to validate the effectiveness and reliability of the system. Finally, Chapter Six concludes the report by summarizing the key findings.

Additionally, the report includes references that provide the foundation of the study and appendices that contain supplementary materials such as raw data, screenshots, and code snippets.

The organization ensures a logical flow, starting from conceptualization and analysis, moving through design and implementation, and ending with validation and conclusion. This structure helps readers easily navigate through the different phases of the research and development process, making the entire study clear, comprehensive, and accessible.

Chapter 2

Background

2.1 Introduction

The demand for rental properties has grown significantly due to rapid urbanization, increased migration, and the mobility of professionals, students, and expatriates.

In the traditional way, the search of rental premises has been conducted via the advertisement in the newspapers, via personal networks or via the local agents.

Although these approaches have been popular, they are usually ineffective, time consuming and non-transparent.

In most instances, tenants find it harder to get reliable information whilst property owners find it harder to reach a larger audience. As digital technologies evolve, and the number of services provided through the network increases, a centralized and automated platform that can streamline the to-let process is in high demand. A system of this kind does not only bring convenience and accessibility to the end users but also helps in the digitalization of rental processes in accordance with the digitalization of ordinary services.

More so, having an online system will help to decrease the reliance on intermediaries, decrease the number of errors in property information, and improve the level of trust between property owners and renters.

2.2 Literature Review

An overview of the literature and systems in the field of property rental and management can give an understanding of the recent tendencies, issues, and technologies in the field. A table provided below will summarize the most important findings of past studies that can be used to create a smart and efficient to-let system.

Table 2.1: Summary of Literature Reviewed.

Author(s)	Year	Title	Methodology	Key Findings
Ahmed et al.	2020	Online Property	Survey & Analysis	Found that centralized online platforms improve efficiency in property search.
Rahman & Chowdhury	2019	Web-Based Real Estate Management	Case Study	Highlighted the importance of user-friendly interfaces in rental systems.
Khan et al.	2021	Mobile Applications for Housing Services	Quantitative	Mobile apps significantly reduce time spent by users in finding rental properties.
Li & Zhang	2018	Smart Property Management using IoT	Experimental	IoT-enabled systems can provide real-time property monitoring for better management.
Smith & Johnson	2019	Digital Platforms for Rental Market	Comparative Study	Online platforms enhance transparency and reduce dependency on intermediaries.
Brown et al.	2020	Responsive Web Design in Real Estate Apps	Design Analysis	Responsive designs increase user engagement and accessibility on multiple devices.
Ahmed & Lee	2022	Database-Driven Property Systems	System Analysis	Proper database management improves accuracy and reliability of rental information.
Gupta et al.	2018	Location-Based Services in Real Estate	Case Study	Location-based search features improve efficiency and user satisfaction.
Wilson & Thomas	2021	E-Rental Platforms for Urban Areas	Survey	Found high demand for integrated rental platforms in densely populated cities.
Rahman & Akter	2022	Digital Transformation in Housing Services	Review	Adoption of digital platforms can modernize rental processes and reduce operational costs.

2.2.1 Similar Applications

A number of online systems in Bangladesh and all over the world are created to ensure the ease of rental and management of properties and which serve as valuable information in designing an Online To-Let System.

The-To let is one such web-based service, where its users are able to post and search rental ads, thereby helping to digitalize local property management [1].

In the same vein, BD To-Let, unites property owners and tenants with the aid of a special rental listing platform, which makes the process more user-friendly to users throughout Bangladesh [2].

Property has also contributed to the property rental ecosystem by incorporation of a massive database of properties along with professional consultancy, which assists in improved decision making among tenants and buyers [3].

Lamed Bangladesh pays attention to convenient design and mobile access that allows customers to find out information without switching on their computers but using their smartphones [4].

The other significant platform is Apnar Bari, which offers property advertisements and interactions between users and allows consolidating property-related information, which helps to centralize it among the renters and landlords [5].

Besides the rental services, there are other technological tools that aid in the formulation of effective online systems.

The Laravel documentation is a guideline to create scalable applications with high security [6], whereas the MySQL Reference Manual is the necessary database management guidelines [7].

To have integration between the backend and the frontend, the PHP official documentation can be used to guarantee dynamic functionality [8], and the W3Schools tutorials contain all the tools to build up resources on HTML5 and CSS3 to create a responsive web design [9].

Moreover, the JavaScript Guide by MDN facilitates the provision of interactive and dynamic characteristics that are necessary in enhancing user experience in property management systems [10].

Taken altogether, these platforms and technologies contribute to the relevance of responsive design, well-integrated databases, dynamism, and user-friendly interfaces, which all offer methodological insights to designing an efficient and user-friendly Online To-Let platform.

Gap Analysis

The gap analysis has revealed that the existing rental sites do not have the capabilities such as live availability, open rental information, customer reviews and relocation services.

. Most platforms lack live availability of rental properties, making it difficult for users to find up-to-date options. Open rental data is rarely provided, which restricts informed decision-making. Additionally, customer review systems are either missing or insufficient, reducing transparency and trust. Furthermore, relocation support and guidance are generally absent, creating a notable gap in user experience and overall service quality.

Features	Bikroy.com	Lamudi	MagicBricks	Airbnb	Proposed system
Search option of properties	Yes	Yes	Yes	Yes	Yes
Advanced filtering (Location, Rent)	Limited	Yes	Yes	Yes	Yes
Property availability status update	No	No	Yes	Yes	Yes
Direct owner-to-tenant connection	Limited	Yes	Yes	Yes	Yes
User reviews and ratings	Yes	Yes	Yes	Yes	Yes
Wish list	No	Yes	Yes	Yes	Yes
Notifications / Alerts for new listings	No	No	Yes	Yes	Yes
Transparent rental details (agreement info, extra costs)	No	Limited	Limited	No	Yes
Responsive design (mobile-friendly)	Yes	Yes	Yes	Yes	Yes
Relocation support services	Yes	Yes	Yes	No	Yes

2.3 Summary

This chapter presented the necessary background and related studies that form the foundation of the proposed Online *To-Let System*. It started with a background of the existing practice of renting and the role of digital platforms in solving the existing problems. To review previous research and applications, a literature review was performed and summarized with the methodologies and key results. In addition, an explanation of related applications illustrated the weaknesses and strengths of current web and mobile-based renting solutions. Gap analysis was subsequently conducted to define missing features and weaknesses of existing systems which assisted in defining the novel contributions of the proposed work. In general, this chapter creates the context of the research, outlines the current limitations and determines why creating a smart, responsive and user-oriented to-let platform is necessary.

Chapter 3

Research Methodology

3.1 Requirement Analysis & Design Specification

The need analysis stage of the study was conducted with the aim of determining the basic needs of the users and the issues involved in the current property renting practices. The fundamental needs of the proposed Online To-Let System were identified through the use of literature review, observation of the user and the responses of tenants, landlords, and property managers.

The analysis demonstrated that the users have needs to have a platform where they can easily locate and filter properties in terms of location, rent, property type, and availability and also have a good communication channel between owners and tenants. It was also determined that secure authentication and profiles management is needed to provide protection on the information of the user and property posting and management functionality are essential to enable property owners to post and update listings on real-time basis. The system should also provide the responsive interface, which is accessible to such devices as desktops, tablets, and smartphones, in order to increase the usability and convenience. The interaction among the users, property data and process of the system were modeled in a conceptual model based on these findings and formed the basis of the system architecture.

These requirements were then converted to a more planned approach to implementation in the design specification which stressed on user-friendly interface design, a secure and scalable database management and efficient property data accessibility. A cyclical development model was used whereby a system prototype was developed, tested and perfected in cycles to ensure quality, availability, and user approval. The feedbacks obtained after each of the testing stages were useful and were incorporated into the subsequent version of the system, which enhanced its reliability and effectiveness.

In conclusion, the requirement analysis and design specification made sure that the proposed system would overcome the inefficiencies of the old methods of renting and it

will provide a modernized, easy to use, and scalable system that would provide more efficiency, accessibility and reliability in the process of renting.

3.1.1 Overview

The research methodology used in the study offers a systemic plan concerning the definition of the problems of traditional to-lets and coming up with a practical solution. It also includes qualitative and quantitative approach in its bid to know perfectly the requirements of the users as well as the system requirements. It involves identification of the problem, and data collection, requirement analysis, system design, development, and evaluation. After a proper definition of the requirements, the study went to the design stage where the system architecture, data flow and interaction diagrams were prepared to offer a clear blueprint on how to implement the system. A development loop was embraced and the prototypes were constructed, tested and polished in accordance with the user feedback. The system was evaluated at the evaluation phase to ensure that it meets the intended objectives through the adoption of the testing which focuses on usability and functionality and efficiency. This systematic approach to the methodology guarantees that the study is not merely based on the needs in the real world but also that the solution is feasible, scalable, and advantageous to property owners as well as tenants.

3.1.2 System Design

The research method applied in the study provides a systemic plan in regards to definition of the problems of traditional to-lets and development of a viable solution. It also entails qualitative and quantitative approach in its quest to know best the requirements of the users and the system requirements. It is characterized by problem identification, and data collection, requirement analysis, system design, development, and evaluation. The study proceeded to the design phase after a good definition of the requirements, to present a clear blue print on how the system can be implemented, the system architecture, data flow and interaction diagrams were created. It adopted a development loop and created the prototypes which were tested and refined as per the user feedback. The system was assessed at the evaluation phase to make sure that it achieves the intended objectives by adopting the testing that emphasizes on usability and functionality and efficiency.

This logical methodology of the approach ensures that the study is not simply founded on the requirements in the real world but also that the solution is viable, scalable, and beneficial to both the owners of the property and its tenants.

Schema Diagram

The Online To-Let System database schema is structured to ensure that there are good relationships between the users and houses, profiles, images, and chats. The basic details of every registered user are then saved in the Users table, this includes the name, email, password, and authentication token. A Profile table is related to every user and includes personal information including address, city, phone number, age, NID, and profile picture. This guarantees the system maintains the login credentials together with user information that is longer.

The Houses table holds all the information regarding houses put on rent. It offers house name, price, number of rooms, balconies, area, availability of gas and parking, place of location, division and status.

Both houses are associated with the person who posted them. House Galleries table has the option of uploading numerous images of a given house so that the house listing is much more alluring and understandable. Chats table is added to facilitate communication between the house owners and the renters. It logs down messages transmitted among users and connects them to the particular house.

This opens a safe and straight forward method of negotiating rental information. All in all, the schema is a well-organized system with users being able to manage profiles, post houses, place house images and talk to each other via chats. The relationship design provides free flow of data, eliminates duplication, and provides uniformity throughout the platform.

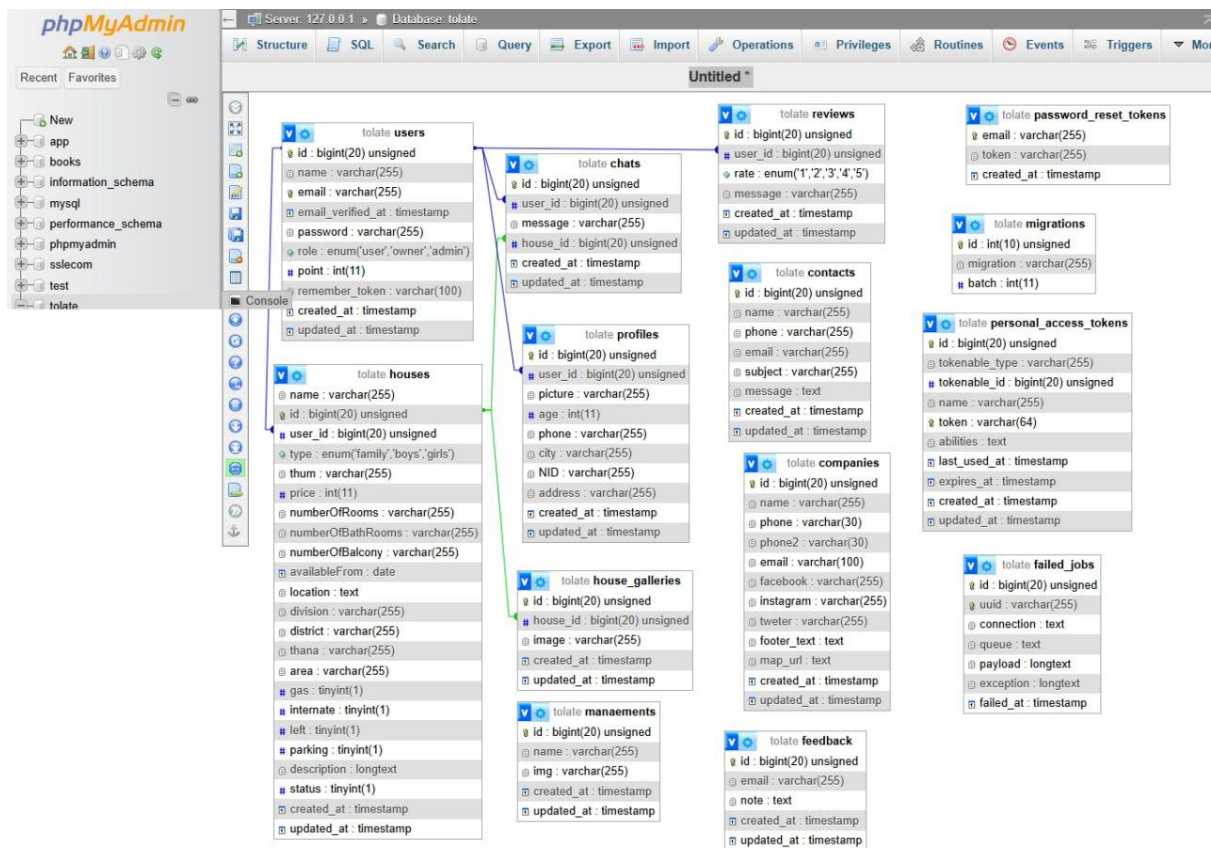


Figure 3.1: Database Schema diagram

ER Diagram

This is an entity-relationship (ER) Model of a Online To-Let (House Rental) Management System. It shows the interaction between the various entities in the database structure. The User entity is used to store vital data, including ID, Name, Email, Password, and a Remember Token that is used to provide a secure authentication procedure. Every user will be allowed to have a Profile with such information as Address, Phone, City, Age, NID, and Profile Picture, which will verify the identity and be trusted within the system. The Houses table is where the house related information is handled such as the Name, Price, Number of Rooms, Number of Balcony, Gas availability, Parking, Location, Division, and Status (available or booked). The houses are associated with the user who created them.

Moreover, the House Galleries organization enables one to store several images of a house, and descriptions can be made to visualize it more efficiently, allowing the

renters to make the right decision. In order to communicate, Chats entity is integrated, which enables people to communicate with each other. The chat records include User ID, House ID, and Messages, and these enable the tenants and landlords to negotiate the details of the rental in the platform safely.

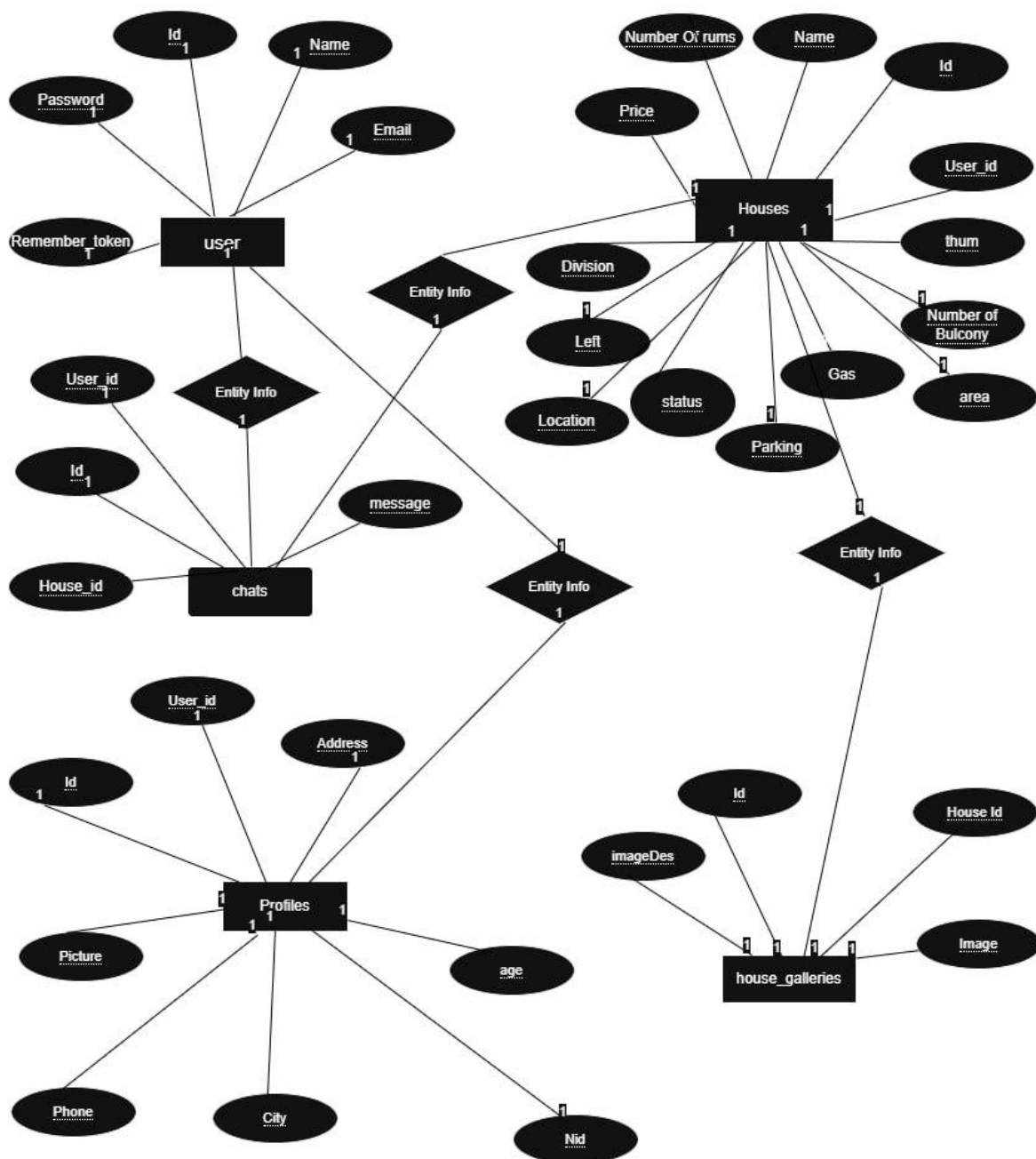


Figure 3.2: ER diagram

Use Case Diagram

The use case diagram of the Online To-Let System illustrates how different users interact with the platform to fulfill their needs. The main actor of the system is the user, who can either be a house owner or a house seeker.

At the beginning, the user registers and logs into the system, which allows them to access various services securely. Once logged in, the user can manage their personal profile by updating information such as address, phone number, city, and identification details.

For house owners, the system provides the option to post houses for rent by entering details like price, number of rooms, location, gas and parking facilities, as well as uploading multiple images with descriptions. On the other hand, house seekers can search and view available houses by applying filters such as price, division, or location, and they can check all the necessary details before making a decision.

To enhance interaction, the system also includes a chat feature where users can communicate directly regarding rental inquiries. This makes the process of negotiation and clarification easier and more transparent. Additionally, the system allows seekers to send inquiries or booking requests, which owners can respond to through the platform.

Altogether, the use case diagram highlights how the Online To-Let System effectively connects house owners and seekers, enabling registration, profile management, house posting, house searching, image viewing, and direct communication in a smooth and integrated manner.

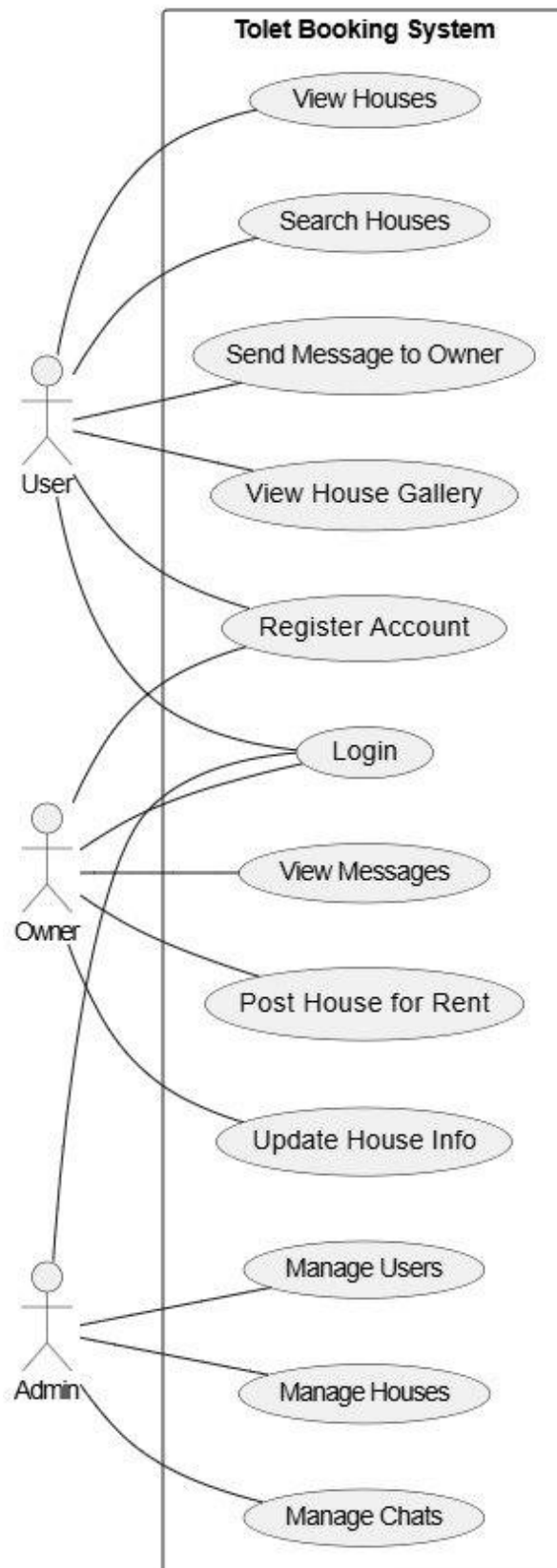


Figure 3.3: Use Case diagram

3.1.3 Functional and Nonfunctional Requirements

Functional Requirements

The functional requirements specify the essential features and services that the Online To-Let System must provide to users. These include:

1. **User Registration and Login** – The system should allow new users to register and existing users to log in securely.
2. **Profile Management** – Users should be able to update personal details such as contact information, address, and profile picture.
3. **Property Listing** – Owners should be able to add details of available houses, flats, offices, or shops, including price, location, and facilities.
4. **Search and Filtering** – Users should be able to search rental properties by price range, location, number of rooms, or type (family, bachelor, office, etc.).
5. **Image and Gallery Upload** – Property owners should upload multiple images of their property to give better visualization to potential renters.
6. **Chat/Communication** – The system should provide a messaging feature to connect tenants with owners directly.
7. **Reviews and Ratings** – Users should be able to leave feedback, ratings, and reviews about a property or owner.
8. **Admin Management** – The admin should have the ability to verify users, manage listings, and monitor system activity.

Nonfunctional Requirements

The non-functional requirements ensure the system's performance, security, and usability. These include:

1. **Performance** – Normal load should see the system respond in 3-5 seconds to user requests.
2. **Scalability** – Due to the system, the number of users and property listings should be able to increase without the performance getting worse.
3. **Security** – The encryption and secure authentication should be used to protect user data, which includes password, personal information, and communication.
4. **Reliability** – The system should ensure high availability with minimal downtime.

5. **Usability** The interface must be easy to use, friendly, and responsive on desktops, tablets and mobile devices..
6. **Maintainability** – The system must be user friendly to update, debug and add in new features.
7. **Compatibility** – The system should work across different browsers and devices without errors.

3.1.4 Data Flow Diagram

The Data Flow Diagram (DFD) demonstrates the way the information moves in the Online To-Let System. It demonstrates the process of interaction between external users (house owners, tenants, and administration) and internal processes of the system, which include authentication, property management, communication and storage of data in the database. On the context level (Level 0 DFD), users (e.g., searching or posting rental listings) send their requests to the system, where they are processed and responses are sent (e.g., displaying available houses, sending messages).

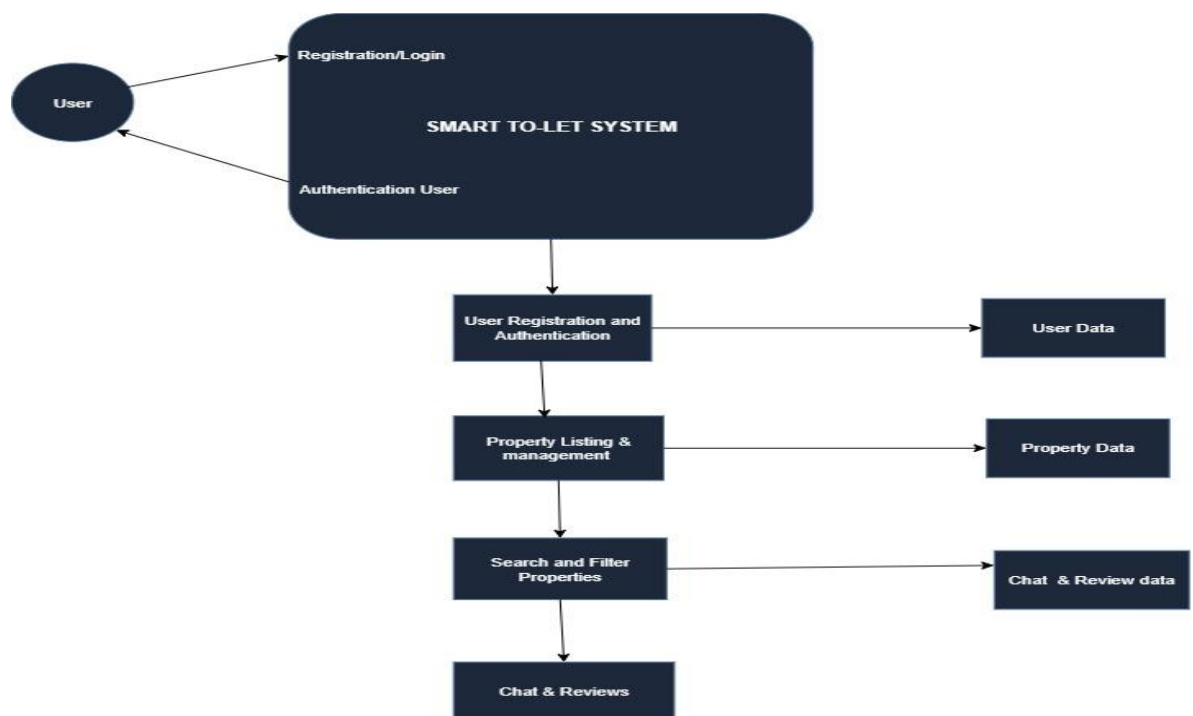
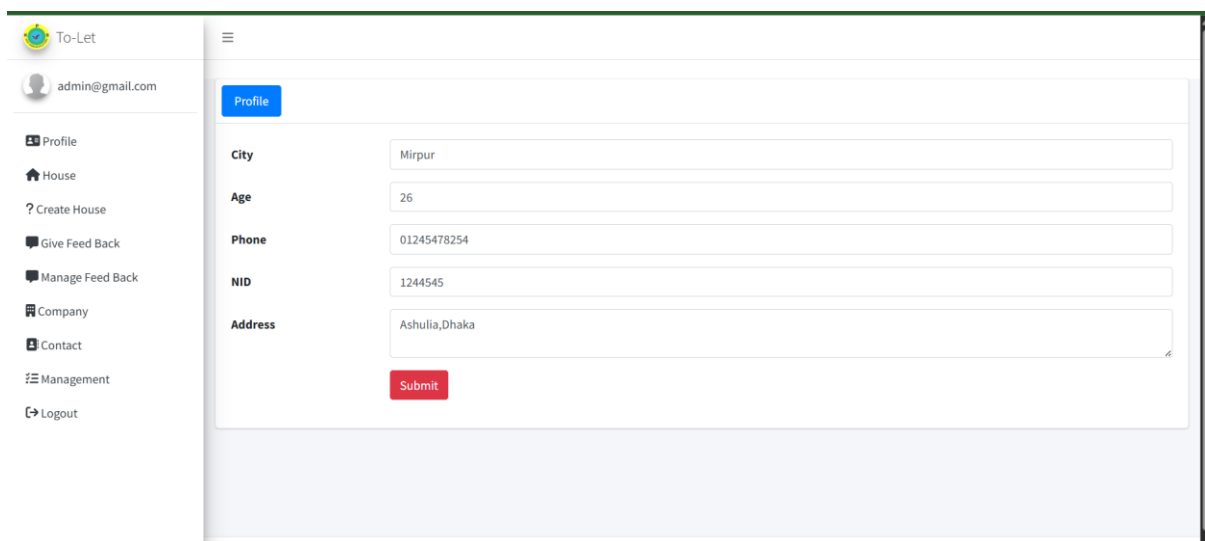


Figure 3.4: DF diagram

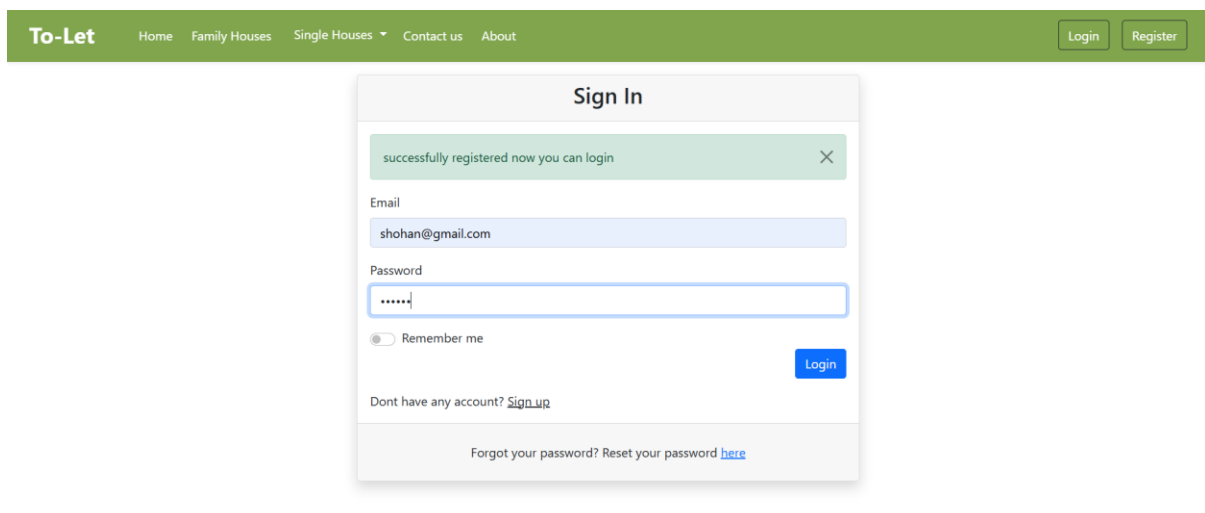
3.1.5 UI Design

Online To-Let System has a user interface (UI) to be easy, clean and user-friendly. The home page contains a search bar and filtering possibilities where the user can narrow his/her search of the house according to the location of the house, the price of the house, the number of rooms in a house, and the facilities. A list/grid of houses in the market is shown with images, price and key information to browse. The UI offers simplicity to house owners who have a conveniently placed “Add House-button through which they can submit photos, rent price and descriptions. The page of each house details contains the gallery of photos, the facilities list, the date of availability and contact/booking section.



The screenshot shows the 'To-Let' user profile page. On the left is a sidebar with a user profile icon and email 'admin@gmail.com', and a list of menu items: Profile, House, Create House, Give Feed Back, Manage Feed Back, Company, Contact, Management, and Logout. The main content area has a 'Profile' button at the top. Below it are input fields for City (Mirpur), Age (26), Phone (01245478254), NID (1244545), and Address (Ashulia, Dhaka). A red 'Submit' button is at the bottom of the form.

+



The screenshot shows the 'To-Let' Sign In page. The top navigation bar includes 'To-Let', 'Home', 'Family Houses', 'Single Houses', 'Contact us', and 'About', along with 'Login' and 'Register' buttons. The main content area features a 'Sign In' modal. Inside the modal, there is a green success message: 'successfully registered now you can login'. Below this are input fields for Email (shohan@gmail.com) and Password (masked with dots). A 'Remember me' checkbox is present. A blue 'Login' button is at the bottom right. At the bottom of the modal, there is a link for 'Sign up' and a link for 'Reset your password here'.

The image shows a web application interface for a user profile. On the left is a sidebar with the logo 'To-Let' and a user profile icon with the email 'shohan@gmail.com'. The sidebar menu includes 'Profile', 'House', '? Create House', 'Feed Back', and 'Logout'. The main content area has a blue 'Profile' button at the top. Below it are input fields for 'City' (Baipail), 'Age' (32), 'Phone' (01456784512), 'NID' (19951245781546456), and 'Address' (Baipail,savar,Dhaka). A red 'Submit' button is at the bottom of the form.

Figure 3.5 : UI Design

Admin Interface

Admin interface is the secure and centralized interface through which the administrators can manage the whole system effectively. With the help of this interface, the admin will be able to track every single activity of the user, to approve or deny the listing of houses, to be sure that only the verified and authentic information will be posted on the platform.

It gives a summary of the performance of the system and provides the most important statistics of the system including the number of users, active listings and recent bookings on the dashboard.

The admin is also able to monitor the communications such as chats, and reviews and avert abuse to ensure that the environment is safe to all users. In addition, the interface allows to create reports and analytics that can be used to monitor system development, user activity, and booking trends.

The layout of the panel is easy to navigate and displayed in a user-friendly manner, and it is not complicated to update the system settings, user management, and house postings control all in one location.

The image displays two screenshots of the 'To-Let' Admin Interface. Both screenshots feature a sidebar on the left with the following menu items: Profile, House, Create House, Give Feed Back, Manage Feed Back, Company, Contact, Management, and Logout. The user is logged in as 'admin@gmail.com'.

Top Screenshot: Profile Form

City	Mirpur
Age	26
Phone	01245478254
NID	1244545
Address	Ashulia,Dhaka

Submit

Bottom Screenshot: Org. Name Form

Org. Name :	To-Let	Twitter :	https://web.facebook.com/?_rdc=1&_rdr#
Email :	demo@gmail.com	Instagram :	https://www.instagram.com/#reactivated
Phone :	01866580248	Footer Text :	Find houses, flats, bachelor or family rooms easily. Our trusted to-let platform connects tenants and landlords quickly, safely, anytime.
Phone 2 :	Enter Phone 2		
Facebook :	https://web.facebook.com/?_rdc=1&_rdr#		

map

```
<iframe src="https://www.google.com/maps/embed?pb=1m18!1m12!1m3!1d6136.89734019276712d90.31468563483486!3d23.878708420638787!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3755c1007ceb31bb%3A0x3d3b64e3171492f9!2sDaffodil%20International%20University%20Lake%20Park!5e1!3m2!1sen!2sbd!4v1757136613073!5m2!1sen!2sbd" width="600" height="450" style="border:0;" allowfullscreen="" loading="lazy" referrerpolicy="no-referrer-when-downgrade"></iframe>
```

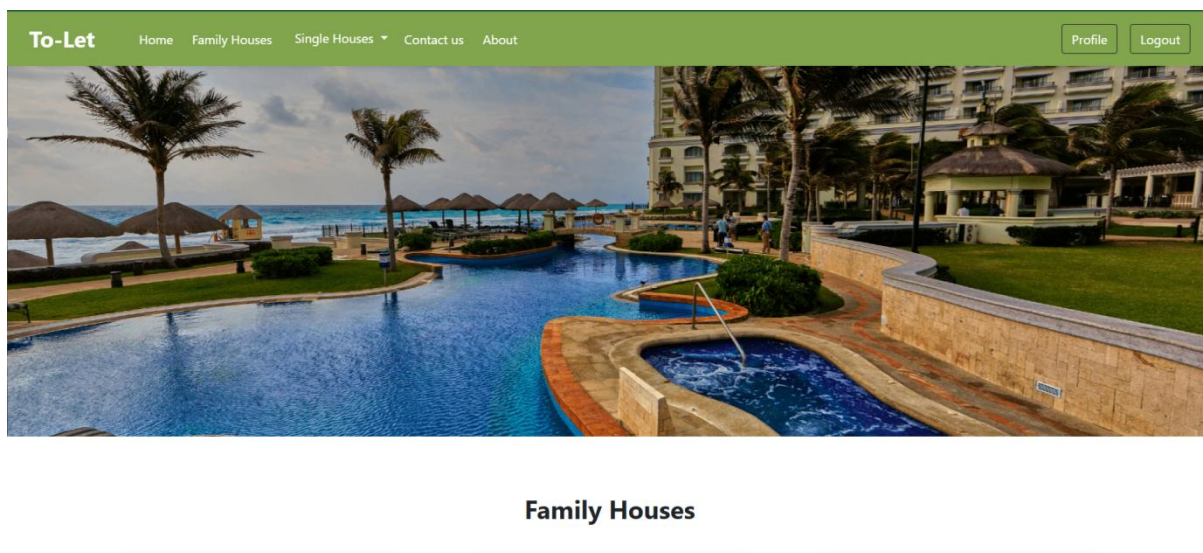
Submit

Figure 3.6 : Admin Interface

Font-end Interface

The Front-End interface of the Online To-Let System is based on simplicity, responsiveness, and ease of use to facilitate the interaction of the tenant and house owners. The home page also has a search engine and a filtering tool that enable users to filter and find the rental houses fast depending on their location, cost, number of rooms and facilities available. All properties have pictures, rental information, and main characteristics to enable the user to compare the properties simply and settle on a decision. To the owners of houses, the interface has an available feature of posting new

listings where photos are uploaded, pricing the rentals and specifying the property characteristics in an organized manner. In the navigation bar, there are links to the most vital sections including Home, Search, Add House, Chats, and Profile, making sure that there is a smooth flow of features on the system. Every detail page involving a specific property offers a gallery view, descriptions, reviews, as well as direct communication, which provide a smooth experience to tenants. The design is very simple and clear and the layouts are organized in a very good way and the use of colors has been kept the same to make the interface look visually attractive. The responsiveness of the front end is another key characteristic of the front end since it is designed to respond to desktops, tablets, and mobile devices without compromising its functionality. The interface is created based on the conventional web technologies (HTML, CSS, JavaScript), as well as such frameworks as Bootstrap that will provide better responsiveness and styling. Interactive features such as dynamic search, image sliders and chat windows are incorporated to ensure that the user can get a contemporary effect. All in all, the front-end interface will provide both tenants and house owners with an opportunity to perform their activities efficiently but with a simple but professional look and feel.



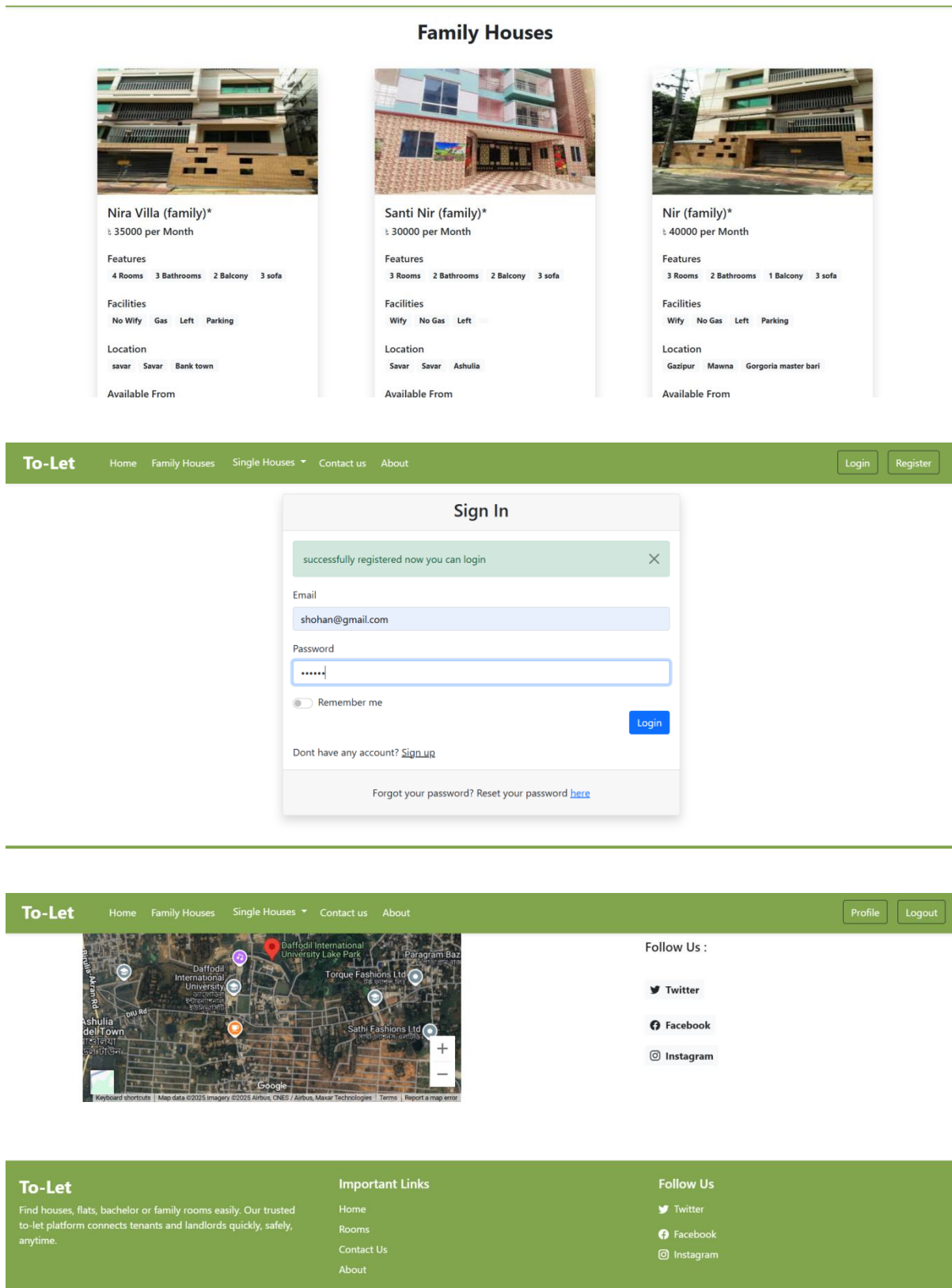


Figure 3.7 : Front-End interface

3.2 Detailed Methodology and Design

When developing the Online To-Let System, we intended to develop a flexible system that will be able to accommodate the different needs of the tenants and landlords. After evaluating various technologies and approaches, I have chosen our design keeping in mind the ease of use, scalability and ease of use of the integration.

I had originally thought about developing a monolithic application that was all in PHP or all in Java. Although this might be the solution in smaller projects, it was difficult to maintain and scale up as the system expanded, and a failure of one component could impact the whole platform.

Another option I considered that could save development time involved having content management systems with rental plug-ins, but these systems did not have the flexibility to add custom features, including real-time updates to the booking system, landlord dashboards, and secure user authentication.

Having considered various frameworks, such as Django and Laravel, I decided to use Larval as the backend as our team is familiar with PHP, and Larval offers a structured MVC framework, embedded authentication, database operations with the help of Eloquent ORM, and high-quality security.

To develop the frontend, I chose to work with HTML, CSS, and JavaScript because these technologies allow creating responsive interfaces, loading them quickly, and interactive dashboard interfaces. MySQL was selected because it is efficient in managing relational data of users, the property, bookings and reviews.

It was built to handle multiple users, property searches, booking updates, and reviews in real time, and the modular architecture of Laravel enables the implementation of additional features without impacting current operations.

To secure user data, security and role-based access controls were incorporated into the platform and the overall design is to ensure scalability, maintainability, and the ability to support future growth.

Making sure that the technologies are well selected and that the building of the Online To-Let System should be modular, both tenants and landlords will be provided with a confident, secure and user friendly experience.

3.3 Project Plan

In order to develop and implement the Online To-Let System, I possessed a properly

designed project plan. The project was divided into phases having some objectives, timeframes, and results.

This technique helped us to assess the progress, focalization, and distribution of resources in the development process. The first two to one week was used on the research and preparation of the project.

In this phase I did research through the web rental sites available, reviews on the sites through user surveys and interviews with tenants and landlords, and study of literature. This enabled us to develop the functional requirements of the system and the critical issues, such as the degree of effectiveness of the search system, property management, and effective booking processes. The design and prototyping were in a period of one week. Figma and Draw.io assisted us in creating wireframes and UI mockups as well as the system architecture on a large scale. The first design solution was the selection of PHP with Laravel as a backend code and HTML, CSS, and JavaScript as a frontend code. The development stage was divided into front end and back end.

The frontend development lasted three weeks in which it focused on responsive interfaces, property listing, search options, dashboards, and booking forms with the application of HTML, CSS, and JavaScript. Back end development was completed in three weeks, and it included the development of RESTful APIs through the assistance of Laravel, management of database structure, user authentication, property management, and booking and review systems.

The quality assurance and the testing were completed in one week following development. Manual and automated tests were performed in such a manner that the functionality under different situations was healthy.

The functional testing, integration testing and user acceptance tests served to perfect the system and to get rid of any issue. The final phases were the deployment and the post-launch monitoring and this took about one week. The system was already hosted in a live server and its performance including the security and its usability was tested. The user instructions and specifications have been prepared and minimal analytic added to monitor the usage and performance. The project was planned to hold weekly progress meetings to organize them and address the challenges in time.

The efficient and gradual process allowed us to complete the Online To-Let System on time and offer us the safe and convenient to use, as well as the fully functioning online system.

3.4 Task Allocation

This table depicts the timeline of the principal activities in each period of the project, from week 12 to week 48.

Tasks	Weeks																		
	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48
Data collection phase																			
Preprocess all the data																			
Model training																			
Create a demo application.																			

3.5 Summary

This chapter covered research methodology and design approach used in the development of the Online To-Let System.

I described our data collection process, the process of system development and the process of the selection of certain technologies.

There are several other options which have been looked at such as monolithic applications, CMS based applications and frameworks and Laravel has been selected in regard to its well structured format, security provisions and convenience.

Its front-end is responsive in HTML, CSS and JavaScript and its backend is a powerful MySQL framework, the system has been made to be modular, scalable and easy to use.

The project plan provided the development, testing, deployment and documentation in a systematic manner in terms of the existence of specific phases, schedules and operations.

In general, the design options and strategy will make the Online To-Let System credible, secure, and able to expand in the future and offer an efficient system to tenants and

landlords. I outlined the research approach and design strategy that was followed in the development of the Online To-Let System.

This involved gathering of data, analysis of the requirements, and a meticulous analysis of the various technological alternatives.

The different options were discussed, and a well-organized, safe, and convenient system was adopted to be implemented. The system has been made modular, scalable and simple to maintain, responsive, and with strong back-end support.

The development, testing, deployment, and documentation stages were well planned in the project, and the execution of the project went smoothly.

The strategy, in general, ensures a secure and reliable as well as efficient platform between tenants and landlords, and the flexibility to implement improvements in the future.

Chapter 4

Implementation and Results

4.1 Environment Setup

I have established a well-arrangement development environment so that I could easily develop our Online To-Let System using the latest tools and technologies. The architecture was created to support development of both front and backends, testing, deployment and overall collaboration as well as be flexible to accommodate later update and improvement. The development environment has been made to be scalable, secure, and reliable to ensure that the system is able to operate in the development and production environment. This method was applied in ensuring that the work process was sustainable at the project life cycle. The reason that I selected PHP is that it is a backend language that has in-built authentication, a structured design and good security based on MVC.

The decoupling of services provided in Laravel modular architecture allowed to simplify the development, testing and maintenance of such services as property management, booking and user administration. I installed MySQL, the reliable relational database system to control the database, as this is an efficient database to operate to store organized information such as the users, properties, booking, and reviews. The Laravel Eloquent ORM provided a simple and secure way of talking to the database through the use of PHP objects rather than raw SQL queries.

To ensure uniformity across different environments, I deployed the system using Kubernetes. Kubernetes provided a robust container orchestration platform that enabled the backend, database, and supporting services to run as separate pods. Its orchestration and management features allowed us to seamlessly launch, scale, and monitor all services in a coordinated manner, ensuring reliability and consistency in system performance.

User authentication and role management was done using larval authentication and middleware authentication and role management. This made the safe access, registration and management of tenants, landlords and administrators possible. The frontend was created in Visual Studio Code, using HTML, CSS and JavaScript, which

has good interactive UI facilities, but is easy to work with. In total, I set up the following environment: PHP as the back-end development, the Larval framework, MySQL as the relational database, Docker and Docker Compose as the containerization and orchestration service, user management via Larval authentication, the frontend via Visual Studio Code, API testing using Postman, and version control using Git and GitHub. This was a firmly established, stable and adaptable environment where a scalable and maintainable System of Online To-Let could be installed.

In summary, our development environment included:

- **PHP 8** – for backend programming
- **Laravel Framework** – main backend framework
- **MySQL** – relational database management
- **HTML5, CSS3, JavaScript** – frontend technologies
- **Blade Template Engine** – frontend-backend integration
- **Visual Studio Code** – development environment
- **Postman** – API testing tool
- **Git & GitHub** – version control and collaboration
- **Shared Hosting Server** – deployment environment

By setting up this environment, I was able to create a strong and adaptable foundation that ensured the Online To-Let System remained secure, scalable, and maintainable. As a result, both tenants and landlords could rely on the platform for efficient and uninterrupted use.

4.2 Testing and Evaluation

Following the development of the Online To-Let System, intensive testing and evaluation was to be done to verify that the system met the functional, performance and usability requirements. The testing process had several steps, which involved unit testing, integration testing, system testing, and user acceptance testing.

Integration testing was conducted to ensure that the front end and the backend were cooperating correctly and that the flow of information between the user interface, APIs and the database was normal and appropriate. General performance, reliability and security testing of the application were checked by system testing in controlled conditions. In order to measure the responsiveness and efficiency of the system,

performance evaluation was done to test the system in different conditions. The Key Performance Indicators included performance indicators like page load time, search response time, booking processing speed and concurrent user handling. Security testing was carried out as well so that sensitive information such as user credentials, reservation details, etc. I am not leaked to unauthorized-users.

The Online To-Let System was compared and contrasted with the existing systems of rental management.

Unlike the generic solutions based on CMS, our system offers a custom and scaled solution, a modular architecture, real-time booking notifications and enhanced user dashboards. Modular design is simpler to maintain, faster to adopt new features, and more fault-tolerant than monolithic applications since the failure of a single component within a system does not affect the entire system.

It has been determined in the analysis that our platform is more practical, reliable and functions better, and that it considers the needs of tenants, landlords and administrators specifically.

Lastly, the testing and evaluation procedure proved that the Online To-Let System is reliable, durable and efficient. The performance criteria and the comparative analysis demonstrate that the system can accommodate realistic rental operations and offer significant benefits compared with the other solutions.

4.3 Results and Discussion

The developed platform achieved the intended goals by simplifying the entire house-renting process for both property owners and tenants.

The application allowed users to browse available houses using advanced filters, view detailed descriptions, and confirm bookings in real time.

On the other side, owners were able to add, update, and manage property information easily through a secure dashboard. All these characteristics made the environment transparent and effective as compared to the previous manual rental process.

Performance testing on the system showed that it responded to user queries despite the number of requests being made at any particular time. The search activities, confirmation of booking and data update used only a few seconds and were smooth and trustworthy.

Trial users expressed that they were more than satisfied with the interface (easy to

use and responsive) and how easy it was to handle their property online without necessarily having to use intermediaries. In the context of the results, it was clear that the project architecture was flexible and could be expanded in the future. The modular architecture also makes it easier to add new functionalities such as mobile applications, online payment gateway, complete with security controls, and property recommendation using artificial intelligence. These deliberations demonstrate the areas of enhancement in the subsequent phase of development.

4.4 Summary

In this chapter, the testing, evaluation, and the results of the developed platform were shown. The system was tested in different phases, such as functional, performance, and security testing, which ensured that the system was reliable and that it fulfilled the design specifications. The performance monitor revealed quick response rates, efficient database trafficking and free-flowing user responses. The comparative analysis indicated the benefits of using the platform in contrast with the traditional rental practices and available solutions, particularly regarding transparency, scalability, and convenience of the user. Some limitations and possible areas of improvement were also identified in the results and discussions which give guidance on how it can be improved in the future. In general, the results supported the idea that the system is effective and proved its possibility of practical application.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with the Standards

The chapter is dedicated to the engineering and software standards, which were considered in the creation of the platform. The criteria were chosen in order to offer quality, security, usability and maintainability of the system. The best alternatives were selected based on the objectives and technical specifications of the project by considering a number of alternatives.

5.1.1 Software Standards

To achieve quality, security and maintainability, the development process was done using the generally accepted software standards. These standards helped shape the coding practices, system architecture, and the way the data is handled as the platform became more reliable and able to withstand the future. The principles, based on which one of the primary standards has been developed, may be regarded as the ISO/IEC 25010 (System and Software Quality Requirements and Evaluation), which contains the description of the way within which software can be tested with references to its functionality, reliability, usability, performance, security, and maintainability.

It is a standard that helped us to identify explicit quality objectives and to evaluate the system performance appraisal in terms of quality. The other alternative was the same model predecessor developed as ISO/IEC 9126. Unlike ISO/IEC 9126, which is less complex and demands less complex usage in small projects, ISO/IEC 25010 was selected because of its more detailed characteristics and the scope of coverage of modern web-based systems. In order to develop safely the OWASP (Open Web Application Security Project) standards were used to prevent the vulnerability of common misuse, SQL, cross-site scripting (XSS) and broken authentication. The alternative would have been to fall back to structure-specific default security (Laravel or Spring Boot defaults). These built-in features provide some basic level of security but may not be all-encompassing. OWASP

standards have been selected because they are more inclusive and almost universally used in the industry, which means that they are more secure in their practices. In terms of code style and maintainability, PSR (PHP Standards Recommendation) was considered when considering a PHP based module, and PEP 8 was considered when considering scripts with Python. By implementing international standards like PSR, the project became easier to maintain in the long run, more collaborative, and better read. Finally, as part of API development, the RESTful API standards have been chosen to ensure that the communication between the frontend and backend services is unified. The other choice was GraphQL, which offers greater flexibility when making data queries but is more difficult to debug and maintain. REST was selected as it is simpler and has a better ability to scale and can be more aligned with the project requirements. The project met the quality and security and scalability requirements with the best practice in the industry by following these software standards.

5.1.2 Hardware Standards

In order to ensure reliable performance, compatibility, and scalability, specific hardware standards were considered during the development and deployment of the platform. Adequate hardware support is required to host the application itself, to serve databases and be able to support several users simultaneously.

One of the wired network communication standards was IEEE 802.3 (Ethernet standard). The standard ensures that the transfer of data is stable, the latency and reliability is low, and must support a simple interaction between the user and the system. A second possibility would have been IEEE 802.11 (Wi-Fi standard), flexible and mobile, but not always as stable as large-scale hosting due to signal jamming and bandwidth limitations. Ethernet is chosen as the preferred standard because it is faster and has a more reliable connection needed when using the backend server.

In server architecture, the system was designed to address the x86-64 architecture specifications that are now supported by the available processors and operating systems. ARM-based architecture would have been another solution; it is power-saving and is gaining popularity in the clouds. The x86 64 was however selected because it can provide more processing power, can be compatible with the existing web hosting environment and can integrate with php, laravel and database server easily.

The alternative was single disk storage that was less expensive; however, when it fails, all the information is bound to be lost. The rationale behind the selection of RAID is that

it is more redundant, faster in read/write and more reliable due to its database-based applications. Finally, IEC hardware safety criteria was followed to ensure that physical servers and networking equipment could remain online securely. Substitutes would be non-certified local hardware configurations that would cost less initially, but would be at risk in the long term. The best equipment was employed and the equipment was also desired to be IEC compliant and meet international safety standards.

5.1.3 Communication Standards

Efficient and secure communication was a critical requirement for the platform, as users depend on real-time data exchange between the client interface, application server, and database.

To achieve this, recognized communication standards were adopted to ensure compatibility, reliability, and scalability. The client to server protocol that was the most widely used was the HTTP/1.1 and HTTPS (Hypertext Transfer Protocol Secure) client to server communication protocol.

I have opted to use HTTPS instead as this provides extra protection, privacy of data, and encryption based on the SSL/TLS. The reason why I use HTTPS is because it is a secure mode of making transactions and the user details the transactions are being made is sensitive like the login details and the booking details. They interacted with one another via the parameters of the RESTful API through what the communication theory of JSON (JavaScript Object Notation) termed parameters. REST is simple and agnostic to the platform. Rather it was SOAP (Simple Object Access Protocol) that supports better built-in error management and formal standards but which is more unwieldy and challenging to write.

The rationale of using REST has been that it exploits simpler, speedy and more up-to-date Web and mobile applications. On the database side, ODBC/JDBC standards were looked at in relation to the communication between the application and database. Others like direct database connection via proprietary drivers have to be considered too since they might (or might not) be faster but less portable. JDBC/ODBC has been chosen to ensure interoperability, flexibility and long term maintainability.

One of the measures that were implemented to ensure that internal information was secure was ensuring that the traffic between the database and the backend services was encrypted using the encryption standard referred to as the SSL/TLS. Otherwise plaintext

connections would have been a possibility but would have had less overhead at the expense of sensitive information being compromised. The use of SSL/TLS was taken into consideration as it is the most encrypted and has the least impact on speed.

5.2 Impact on Society, Environment and Sustainability

The developed platform has the potential to create a positive impact on society by transforming the traditional rental process into a more transparent, efficient, and accessible system. Tenants can easily search for properties without depending on middlemen, while landlords can manage their listings and transactions digitally. This improves trust, reduces fraud, and saves time for both parties.

The solution also leads to social inclusion through digitization of the process since individuals in various locations can gain access to property information at any distance. Environmentally, the system is green in that it minimizes physical visits, paperwork and printed advertisements. There is a reduction in the number of physical visits to do property searches that directly leads to the reduction of fuel consumption and carbon emissions.

Moreover, digital database usage reduces the dependence on paper-based records, contributing to eco-friendly approaches to managing the property. The platform was developed considering scalability and adaptability as far as sustainability is concerned. The solution is economically viable as it uses open-source technologies and software standards that have become common in most systems, increasing the lifespan of the system and thereby making it economically viable.

In addition, cloud-based deployment services will help to efficiently utilize computing resources to reduce energy use as opposed to standard standalone server models. Altogether, the platform fulfills not only a burning social need but also supports the vision of environmental sustainability and guarantees the long-term sustainability of the platform due to scalable design and resource use.

5.2.1 Impact on Life

The introduction of this digital platform has a significant impact on daily life by simplifying the process of searching and managing rental properties.

This is more convenient to the tenants since they can search the houses available, compare, and even make their reservations on the internet without the hassle of having to visit the same location a number of times. This saves time and money and also helps

to minimize the possibility of fraud or misinformation. This enhances confidence and promotes good relations between owners and renters.

The convenience of locating accommodation within a short time, in a safe manner, is also valuable to students, job applicants, and families who are moving to new locations. The platform is also inclusive and offers fair access to property information without any discrimination based on geographical location.

Rural or remote people can learn about the opportunities to rent in the city without having to use the services of local agents. By doing so, the solution will not only increase the efficiency but also the general quality of life as the housing will be more accessible, reliable, and transparent.

5.2.2 Impact on Society & Environment

The platform positively impacts society by creating a more transparent and efficient rental system that reduces dependence on brokers and middlemen.

This establishes trust between owners of property and tenants and it also avoids fraud. It also saves mankind precious time and resources and makes the entire housing experience beautiful. The solution leads to social good and reduces the rate of visiting the places in search of the appropriate accommodation amongst persons of varied background including job seekers and students. As far as the environment is concerned, the system will minimize the number of visits to physical properties, paper advertisements, and manual record keeping.

The net effect is reduced motor traffic and reduced fuel use and carbon emission. The use of online communication and digital databases prevents the use of print materials, which are environmentally friendly.

The operational model promotes sustainable livelihoods in the long run by integrating technology with sustainable operations, whereby the current housing benefits are in support of the sustainability objectives of the wider scope.

5.2.3 Ethical Aspects

The development and use of the platform adhere to ethical principles to ensure fairness, transparency, and respect for user privacy.

The personal and contact information of the user is safely stored and processed without being accessed and misused by a third party. It will also be easy to use by all and I will

not need to bother thinking about their location, social status or technical capability anymore. It is neither discriminatory nor provides different opportunity to access housing information to tenants and potential renters to locate landlords. The communication and payment procedure were also assessed ethically and by doing so, the interaction is not undermined in terms of security, transparency and accountability.

Overall, the site is morally accountable because it safeguards the information of its customers, promotes transparency, and offers equal access, making it credible and socially responsible as an online rental platform.

5.2.4 Sustainability Plan

The platform was developed with long-term sustainability in mind, ensuring that it remains reliable, scalable, and adaptable to future needs.

This enables it to introduce a new functionality, change and improvements without the need to restructure significantly to reduce the cost of development and extends the life of a system. It is a system that uses less resource consumption by adopting open-source technologies, popular software's, and effective database management, but it does not impact the performance negatively.

The platform will be more environmentally-friendly, in the context of the environmental dimension, since it will reduce the number of physical visits to the property, paperwork and printed materials and, therefore, energy consumption and carbon footprint. Cloud services also use less energy since cloud hosting services consume resources efficiently. The other positive economic impact of this platform on landlords and tenants is that it makes renting more transparent and easier since there is no need to visit mediators.

Generally, the sustainability plan will render the system sustainable, cost-effective and technologically adaptable to come up with a solution to the problem of managing digital property in the long run.

5.3 Project Management and Financial Analysis

The project was managed using a structured approach, dividing tasks into planning, design, development, testing, and deployment phases to ensure timely completion and efficient resource allocation.

Performance was monitored, challenges identified, and coordination maintained by

weekly progress meetings and tracking of milestones. Tasks were tracked, deadlines managed, and communication facilitated via project management tools to ensure that the technical and operating sides were addressed appropriately.

The financial consideration adopted both the development costs and possible sources of revenue. The project budget comprised software tools, software license, web hosting cost, server cost, domain registration, development environment, testing tools and manpower. The cost of the project was estimated to be around 2,500 dollars, including the following items: tools, cloud hosting, and work of the team. There was also an alternate budget case of 1,500, which would include the free or open-source development tools, less cloud hosting, and a smaller development team.

The financial analysis shows that the project is economically viable and there is a clear cost management plan, revenue generation plan and possibility of future growth. You can also have me rewrite this in a short concise format with only budget, alternate budget, and revenue model to easily insert into your report.

SL No.	Expense Description	Amount (BDT)	Description
01	Domain & SSL	5,000	Custom domain registration and SSL certificate (1 year)
02	Hosting & Server Infrastructure	15,000	Shared/VPS hosting with MySQL & PHP support (1 year)
03	Backend Development (Laravel + PHP)	20,000	Development of system modules, API setup, and integration
04	Frontend Development (HTML, CSS, JS)	20,000	UI/UX design, responsive layouts, and interactive features
05	Database Setup &	10,000	Database configuration, optimization,
06	QA & Testing	15,000	Manual and automated testing, feedback collection
07	Developer Salaries	30,000	Backend & frontend freelance/part-time developers

SL No.	Expense Description	Amount (BDT)	Description
08	UI/UX & Branding	10,000	Logo design, wireframes, and design assets
09	Maintenance & Support (1st Year)	10,000	Server monitoring, bug fixes, and updates
10	Marketing & Outreach	10,000	Social media promotion, flyers, and demo sessions
11	Miscellaneous	10,000	Unexpected minor expenses (tools, plugins, etc.)
12	Contingency (10% of total)	20,000	Reserved buffer for unforeseen costs

Total cost : 175000

5.4 Complex Engineering Problem

During the development of the platform, several complex engineering challenges were encountered that required careful analysis and innovative solutions.

One of the main challenges was to design a scalable and modular system architecture which could support number of users, search of property in real time, booking and updating at any time without degradation of performance.

Integration of the front and the backend is something that had to be planned out and abide by industry standards so that data transfer between the front and the back is smooth and also to ensure data protection in the database.

The other important challenge was to implement safe user authentication and role-based access control. It needed a robust security infrastructure to maintain privacy of confidential user data and ensure that only authorized functions were available to tenants, landlords and administrators. This has been achieved by integrating the Laravel authentication system and the encryption and secure communication protocols that are industry standards.

The second critical issue was performance optimization as the platform needed to provide high response time even at its peak. They were able to achieve reliable performance through database indexing, query optimization, caching, and efficient frontend rendering techniques. In addition, usability and responsiveness were also issues in creating a user interface to be applied to a multitude of devices and screen sizes.

Overall, these are the engineering problems that required technical abilities as well as software and hardware standards alongside innovative problem solving. They did this in their way, and the platform is now reliable, scalable, secure and capable of fulfilling the needs of its users which gives it a strong base to expand on in the future.

5.4.1 Complex Problem Solving

In this section, provide a mapping with problem solving categories. For each mapping add subsections to put rationale (Use Table 5.1). For P1, you need to put another mapping with Knowledge profile and rational thereof.

Table 5.1: Mapping with Complex Engineering Problem.

Problem ID	EP1 Dept of Knowledge	EP2 Range Of Conflict ing Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stakeholder Involvement
<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>

Mapping with Knowledge Profile

This section is designed to map the overall problem and EP1 (*multiple between K3, K4, K5, K6, K8 for attaining EP1*) to the Knowledge Profile.

Table 5.2: Mapping with knowledge Profile.

K1 Natural Science	K2 Mathematics	K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K7 Comprehension	K8 Research Literature
<i>No</i>	<i>No</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>No</i>

5.4.2 Engineering Activities

Problem Identification

Yes – The project started with recognizing the challenges of the traditional house rental process in Bangladesh, such as lack of transparency, time consumption, and communication gaps.

System Analysis

Yes – A detailed requirement analysis was conducted to define user needs, including house owner registration, tenant search, and secure communication.

System Design

Yes – Both frontend (HTML, CSS, JS) and backend (PHP, Laravel, MySQL) designs were prepared, covering database structure, UI mockups, and architectural flow.

Implementation/Development

Yes – The system was developed using Laravel for backend and JavaScript-based frameworks for frontend to ensure scalability and responsiveness.

Testing & Validation

Yes – Manual and automated testing ensured that functionalities such as login, property posting, and search filters worked as expected.

Deployment

Yes – The system was deployed on a VPS hosting environment with a custom domain and SSL for real-world accessibility.

Maintenance & Support

Yes – Continuous monitoring, bug fixing, and updates were planned to ensure system reliability and long-term usability.

Research & Innovation

Yes – The project explored different design alternatives (CMS, frameworks like Django, Laravel) and integrated unique features like smart search and secure communication.

Project Management

Yes – Agile-based project management was applied, with proper resource allocation, financial planning, and weekly progress tracking.

Mapping with Complex Engineering Activities

This section is designed to map the overall problem and EA's (*multiple*).

Table 5.3: Mapping with Complex Engineering Activities.

EA1 Range of re- sources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
<i>Yes</i>	<i>No</i>	Yes	No	Yes

5.5 Summary

This chapter mapped the Online To-Let System to relevant knowledge profiles, engineering activities, and complex engineering activities. It has been established that the system has a close relationship with engineering principles, expert knowledge, design, practice and research literature. It also discusses all the major engineering activities, including problem identification, system analysis, design, implementation, testing, deploring, and maintenance. Furthermore, the project is characteristic of

complicated engineering processes, such as the involvement of various sources, contact with users, creativity, societal and environmental impacts, and a trade-off between familiarity and novel solutions. Overall, the mappings demonstrate that the Online To-Let System is not only technically possible, but also academic, and it contributes not only to engineering practice, but to the overall betterment of society.

Chapter 6

Conclusion

6.1 Summary

The Online To-Let System was designed and developed to address the inefficiencies of the traditional house rental process by offering a secure, transparent, and user-friendly digital platform. The research methodology guided the project from requirement analysis through design, development, and testing, ensuring that both functional and non-functional requirements were met. The system integrates modern web technologies such as Laravel, PHP, MySQL, HTML, CSS, and JavaScript to provide scalable solutions for tenants and house owners.

In Defence, the primary focus was on requirement gathering, literature review, system design, and the development of core modules. The outcomes demonstrated the feasibility of implementing a web-based solution that reduces time, enhances accessibility, and ensures reliability.

These accomplishments are the premise of the additional enhancements, which are the advanced features and optimization that will be considered in the later stage.

6.2 Limitation

Despite the fact that Online To-Let System is an effective solution that can enhance the traditional house rental process, there are some weaknesses that must be considered in the future development.

The existing version is primarily web-based, but does not have a mobile application, and may restrict accessibility by some users. Such highly developed services as AI-based recommendations, automatic fraud detection, and in-built payment systems are not in use yet.

Also, the system is reliant on internet connectivity that is stable, and safe hosting, therefore, performance problems might emerge in regions where the network coverage is low. The project is also limited by resources and budget and this may be limiting its implementation at a large scale and with a large number of users in the initial phases.

6.3 Future Work

Although the current implementation of the Online To-Let System has been effective in resolving the inherent issues involved in the house rental procedure, it still has much room to grow and enhance the processes in the upcoming stages. The major objectives are to create a complete working Android and iOS mobile application to enhance its ease of use and offer on-the-go services. This would make it more adoptable and convenient as a significant part of the Bangladesh user population uses smartphones. The other field of the further work is the inclusion of advanced Artificial Intelligence (AI) capabilities. The recommendation engines powered by AI can examine the preferences of users and offering the most appropriate houses depending on their location, budget and facilities. In the same manner, the use of Natural Language Processing (NLP) applications like chatbots can be used to give immediate answers to user queries making the system more interactive and user-friendly. Machine learning methods may also be applied to detect fraud detection systems in order to detect fake postings or suspicious activities. Moreover, by incorporation of secure online payment systems, tenants and landlords will be able to use the platform to control rent, deposits, etc. Automation, which will be implemented to give reminders, track payment history, and issue electronic receipts, will increase financial transparency. In order to provide security to such transactions, further developments will be aimed at the integration of more secure encryption tools and multi-factor authentication. Another factor of future development is scalability. It must be migrated to an effective cloud environment to support massive user traffic. The cloud-based scaling will allow the system to deliver a high-performance during the peak usage and enable users in other geographical locations to be served without downtime. Moreover, the feedback of the users will be also important in the future improvements. The platform will be able to develop by gathering and examining user information and review in order to accommodate the needs of tenants and house owners. This involves provision of advanced filtering, location based services on the basis of GIS integration and multilingual services to attract wider audience. Lastly, it can be considered cooperating with real estate agencies, housing authorities, and financial institutions in order to expand the ecosystem of the system. Such partnerships will elevate the credibility of the system, offer the verified listings, and also might offer the value-added services, including rent insurance or mortgage loans.

References

- [1] “the-toilet,” *agoda.com* [Online]. Available <https://www.thetoilet.com/> [Accessed: Jan. 26, 2025]
- [2] “bdtoilet,” *its holiday ltd* [Online]. Available [https:// https://www.bdtoilet.com/?date=2024-12-26](https://www.bdtoilet.com/?date=2024-12-26) [Accessed: Dec. 26, 2024]
- [3] “Bproperty,” *Bproperty.com* [Online]. Available: <https://www.bproperty.com/> [Accessed: Feb. 15, 2025]
- [4] “Lamudi Bangladesh,” *Lamudi.com.bd* [Online]. Available: <https://www.lamudi.com.bd/> [Accessed: Mar. 2, 2025]
- [5] “Apnar Bari,” *Apnarbari.com* [Online]. Available: <https://www.apnarbari.com/> [Accessed: Jan. 30, 2025]
- [6] “Laravel Documentation,” *Laravel.com* [Online]. Available: <https://laravel.com/docs/> [Accessed: Feb. 10, 2025]
- [7] “MySQL Reference Manual,” *Oracle* [Online]. Available: <https://dev.mysql.com/doc/> [Accessed: Feb. 12, 2025]
- [8] “PHP Official Documentation,” *PHP.net* [Online]. Available: <https://www.php.net/manual/en/> [Accessed: Jan. 20, 2025]
- [9] “HTML5 and CSS3 Tutorial,” *W3Schools* [Online]. Available: <https://www.w3schools.com/> [Accessed: Feb. 5, 2025]
- [10] “JavaScript Guide,” *Mozilla Developer Network (MDN)* [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide> [Accessed: Feb. 8, 2025]

172-15-9761

ORIGINALITY REPORT

11%

SIMILARITY INDEX

9%

INTERNET SOURCES

2%

PUBLICATIONS

9%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	6%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	www.coursehero.com Internet Source	1%
4	Submitted to United International University Student Paper	1%
5	Bui Thanh Hung, M. Sekar, Ayhan Esi, R. Senthil Kumar. "Applications of Mathematics in Science and Technology - International Conference on Mathematical Applications in Science and Technology", CRC Press, 2025 Publication	<1%
6	Submitted to Elizabeth School of London Student Paper	<1%
7	Submitted to Sheffield Hallam University Student Paper	<1%
8	Submitted to Kingston University Student Paper	<1%
9	Submitted to National College of Ireland Student Paper	<1%
10	Submitted to Karachi Institute of Economics & Technology, Karachi Student Paper	<1%

11	jtec.utem.edu.my Internet Source	<1 %
12	su-plus.strathmore.edu Internet Source	<1 %
13	www.utupub.fi Internet Source	<1 %
14	dl.lib.uom.lk Internet Source	<1 %
15	journal.amikmahaputra.ac.id Internet Source	<1 %
16	jurnal.pnj.ac.id Internet Source	<1 %
17	www.diva-portal.org Internet Source	<1 %
18	Larry E. Wood. "User Interface Design - Bridging the Gap from User Requirements to Design", CRC Press, 2018 Publication	<1 %
19	ir.juit.ac.in:8080 Internet Source	<1 %
20	www.medrxiv.org Internet Source	<1 %
21	Ramesh Chandra Poonia, Kamal Upreti, Mohammad S. Khan. "Smart Cyber-Physical Systems - Innovations and Real-World Implications", CRC Press, 2025 Publication	<1 %

Exclude quotes

Off

Exclude matches

Off