



CSE327 Project Report from Section 8, Group 1

Abashon (A House Rental Website in Bashundhara R/A)

1. K. M. Lutfor Rahman (1411162042)
2. Most. Airin Akter (1812770042)
3. Salman Sad Sakil (1711064642)
4. Sanjida Akter (1831522042)

Supervised by:
Dr. Md. Sazzad Hossain

Submission: June 2023

ECE Department

Acknowledgements

We thank our honorable supervisor, Dr. Md. Sazzad Hossain, for his immense support throughout this project. His guidance and deep knowledge of software engineering helped us understand and solve the target problem.

We would also like to thank the almighty for letting us through the difficulties we have faced during the design and implementation process of this project.

Contents

1	Introduction	1
1.1	Background	1
1.2	Statement of the problem	1
1.3	Objectives (General & Specific)	2
1.4	Proposed system	3
1.5	Benefits or Significance of the project	4
1.6	Scope of the project	5
1.7	Feasibility Assessment	5
1.7.1	Economic Feasibility	5
1.7.2	Technical Feasibility	5
1.7.3	Operational Feasibility	6
1.7.4	Schedule Feasibility	6
2	Literature Review	7
2.1	Technological Advancements in House Rental Systems	7
2.2	Market Trends and Business Models	7
2.3	User Experience and Satisfaction	8
2.4	Challenges and Limitations	8
2.5	Legal and Regulatory Considerations	8
3	Methodology	11
3.1	Date collection	11
3.2	Project plan	12
3.2.1	Architecture plan	13

3.2.2	Front-end plan	13
3.2.3	Backend development	13
3.3	Development plan	14
3.4	Work plan	14
4	Proposed system	15
4.1	Proposed system	15
4.2	Functional requirements	15
4.3	Non-functional requirements	16
4.4	System models	18
4.4.1	Use case diagram	18
4.4.2	Context diagram	19
4.4.3	ER diagram	19
5	Design contents	20
5.1	Introduction	20
5.2	Proposed system architecture	21
5.3	Sub-system decomposition	23
5.4	System layout	24
5.5	Usear interface design	26
6	Implementation	29
6.1	Introduction	29
6.2	Algorithm development	29
6.3	Coding	30
6.4	Hardware and software acquisition	30
6.5	Installation	32
6.6	Testing	32
6.6.1	Unit testing:	32
6.6.2	Integration testing:	32
6.6.3	System testing:	32
6.7	Maintenance	33
7	Conclusion and Recommendations	36

7.1 Conclusion	36
7.2 Recommendation	36
References	38

List of Figures

3.1	Software architecture	12
3.2	Gantt chart	14
4.1	Use case diagram	18
4.2	Context diagram	19
4.3	ER diagram	19

List of Tables

Listings

Chapter 1

Introduction

1.1 Background

This is a proposal for the website development of a house rental system in Bashundhara RA. This document will provide a brief idea of our project. It will contain the background and product context, user story, objectives, solution description, front end back end development, development plan, project schedule.

1.2 Statement of the problem

The house rental system in Bashundhara is currently inefficient and outdated. There is no central database of available properties, and finding a rental is often time-consuming and frustrating.

Here are some of the problems with the current system:

- **Lack of transparency:** There is no way for tenants to compare prices and amenities across different properties easily. This makes it challenging to find the best deal.
- **High fees:** Tenants often pay high fees to real estate agents and landlords. This can add up to a significant amount of money throughout a lease.
- **Security risks:** There is no way for tenants to verify the identity of landlords or the condition of properties before signing a lease. This can lead to problems down the road.

The current system is also unfair to landlords. They often have to deal with time-consuming and costly rental applications. They also have to worry about finding reliable tenants who will pay their rent on time.

A new house rental system could address these problems and make renting a house in Bashundhara more efficient, transparent, and fair.

Here are some of the features of the new system:

- **Centralized database:** A centralized database of available properties would allow tenants to compare prices and amenities across different properties easily.
- **Low fees:** The fees charged to tenants would be low or nonexistent. This would save tenants money and make renting more affordable.
- **Security features:** The system would have security features to protect tenants from fraud and scams. This would give tenants peace of mind knowing that they are dealing with a reputable landlord.
- **Automated process:** Many of the processes involved in renting a house, such as application processing and lease signing, would be automated. This would save landlords time and money.

A new house rental system would benefit both tenants and landlords. It would make renting a house more efficient, transparent, and fair.

1.3 Objectives (General & Specific)

A home is a permanent or semi-permanent housing space for a person, family, household, or multiple families in a tribe. You rent a room in your house that is always open for paying clients to stay in for a brief period. This room is not a housing unit and is only utilized as a hotel or similar establishment. A person's most fundamental requirement is a place to call home. Individuals who do not have a home can exist by paying rent to people who have more dwellings, which is one of the four needs of human beings. However, we must comprehend the relationship between the individual who rented the residence and the tenant. Several renters with various behaviors exhibit various traits in rented persons. Some renters may get along well, while others may not. Those renters who are misbehaving may act unfavorably against

the rented folks. But here's what we've been thinking how do we live with those rented folks with horrible behavior? Renting a property has both advantages and disadvantages. Renter benefits include the ability to work different types of jobs during the construction of the house, the ability not to repair if the house depreciates, and so on. On the other side, the tenant may face challenges or disadvantages due to that specific property, such as failure to pay rent on time may result in bankruptcy, and the renter may ask him to vacate his home within a specific time frame. He or she may not have complete freedom to wander around the compound of that specific residence, and he or she may not even have the authority to enter after the entry time has passed. The leased person may use match time to meet the renter; however, he or she must pay a commission for the intermediary. Protects humans from hazardous substances. Home is a necessity, and support is a need for specific individuals. People who cannot afford to acquire or pay private leased sector rents are given social homing. This problem was taken into account by our system, and it was resolved in the time allotted.

There is no adequately allocated home, and the system is challenging to organize based on the user's interests. In addition, the house rental administration system is virtually entirely manual. The administrative system cannot create an online house rental management system, and most work is done by unlicensed intermediaries who are unaware of the administrative system, making it more difficult and expensive for customers to locate a property. The difficulty with the existing system is that finding a home is difficult and time-consuming. Also, extra cash to find a place to live. The system requires more human power. When the user needs information on their residence, they are unable to obtain it. There is an excessive amount of time spent looking for a home. The payment system's complexity. When anything in the rental unit breaks and the tenant's health or safety is jeopardized, or the building or property is at risk until repairs can be done, an emergency repair is necessary. Due to problems, management has become more complex.

1.4 Proposed system

The system's primary goal is to create an online house rental management system. The following is a list of particular objectives that must be met to achieve the overall goal: To make it easier to keep track of who wishes to come home and for an administrative management system. For

the house finders, create an internet home renting system. Allow the administrator to see home finders. To create an online system that is entirely functional and automated for house rental management. To offer a well-organized and dependable system with the fewest possible faults. Customers may easily register to rent a property anytime from the comfort of their homes.

1.5 Benefits or Significance of the project

The project House Rental System for Bashudhara has many benefits and significance. Here are some of them:

- **Increased efficiency:** The House Rental System will help increase the rental process's efficiency. This will allow landlords and tenants to find each other easily and communicate. This will save both parties time and money.
- **Improved transparency:** The House Rental System will also help improve the rental market's transparency. This will require landlords to provide accurate information about their properties. This will help tenants to make informed decisions about where to rent.
- **Reduced fraud:** The House Rental System will also help reduce rental market fraud. This is because it will require landlords to verify their identity and provide proof of ownership. This will help to protect tenants from being scammed.
- **Improved security:** The House Rental System will also help improve rental market security. This will allow landlords to track who is entering and leaving their properties. This will help to deter crime and to protect tenants from harm.
- **Increased access to housing:** The House Rental System will also help increase housing access. This is because it will make it easier for people to find rental properties that meet their needs. This will be especially beneficial for people who are looking for affordable housing.

1.6 Scope of the project

Overall, the House Rental System is a valuable project that will positively impact the rental market in Bashudhara. It will help increase efficiency, transparency, reduce fraud, improve security, and increase access to housing.

For example, it could help to:

- Attract new businesses and residents to the area
- Increase property values
- Stimulate the local economy
- Improve the quality of life for residents

The House Rental System is a promising project that has the potential to make a real difference in Bashudhara. It is a project that is worth supporting.

1.7 Feasibility Assessment

1.7.1 Economic Feasibility

The economic feasibility of a house rental system in Bashudhara can be assessed by considering the following factors:

- The demand for rental housing in Bashudhara is high, as the area is home to many young professionals and students.
- The cost of renting a house in Bashudhara is relatively affordable compared to other areas of Kathmandu.
- The rental market in Bashudhara is relatively stable, with few fluctuations in prices

Based on these factors, it is likely that a house rental system in Bashudhara would be economically feasible.

1.7.2 Technical Feasibility

The technical feasibility of a house rental system in Bashudhara can be assessed by considering the following factors:

- A reliable internet connection is essential for a house rental system.
- The development of a user-friendly website or mobile app would make it easy for users to find and book rental homes.
- Using a secure payment system would protect the interests of both landlords and tenants.

Based on these factors, it is likely that a house rental system in Bashudhara would be technically feasible.

1.7.3 Operational Feasibility

The operational feasibility of a house rental system in Bashudhara can be assessed by considering the following factors:

- The availability of qualified staff to manage the system is essential.
- The development of clear policies and procedures would ensure the system runs smoothly.
- The establishment of a customer service team would help to resolve any problems that may arise.

Based on these factors, it is likely that a house rental system in Bashudhara would be operationally feasible.

1.7.4 Schedule Feasibility

The schedule feasibility of a house rental system in Bashudhara can be assessed by considering the following factors:

- The time required to develop the system and launch it.
- The time required to market the system and attract users.
- The time required to manage the system and resolve any problems that may arise.

Based on these factors, it is likely that a house rental system in Bashudhara could be launched within a reasonable timeframe.

Overall, the feasibility of a house rental system in Bashudhara is high. The system would be economically, technically, operationally, and schedule-feasible.

Chapter 2

Literature Review

The house rental system has witnessed significant advancements in recent years with the emergence of online platforms and digital technologies. This literature review explores existing research and scholarly articles on the house rental system. It will provide insights into various aspects of house rental systems, including technological developments, market trends, user experiences, and challenges landlords and tenants face.

2.1 Technological Advancements in House Rental Systems

Numerous studies have highlighted the impact of technology on the house rental market. Research done by Voumick et al. 2021 discusses the role of mobile applications in simplifying the house rental process and improving the overall user experience. The study emphasizes the importance of user-friendly interfaces, real-time updates, and secure payment systems.

2.2 Market Trends and Business Models

Studies by Alam et al. 2022 explore the house rental industry's market trends and business models. They discuss the rise of online platforms like Airbnb and their impact on traditional rental markets. These studies analyze the factors influencing rental prices, demand and supply dynamics, and the changing preferences of renters.

2.3 User Experience and Satisfaction

Research on user experiences and satisfaction in house rental systems focuses on understanding the factors contributing to customer satisfaction and loyalty. A study by G. Liu and Chang 2021 examines the importance of trust, reliability, and transparency in online rental platforms. It highlights the significance of accurate property information, prompt customer service, and fair dispute resolution mechanisms.

2.4 Challenges and Limitations

Several studies highlight the challenges and limitations landlords and tenants face in the house rental system. Research by X. Liu et al. 2022 investigates the issues of information asymmetry and the potential for fraudulent activities in online rental platforms. It suggests implementing verification systems and user reviews to address these challenges.

2.5 Legal and Regulatory Considerations

The legal and regulatory aspects of the house rental system are another critical area of research. Studies by Lian et al. (2019) and Xu et al. (2020) examine the legal frameworks and regulations governing rental transactions. They discuss lease agreements, tenant rights, and landlord obligations, emphasizing the importance of legal compliance in the rental process.

Rental housing came into play due to the seriousness of housing problems. Not most people can afford to build houses due to several factors, such as poor income; hence there was a need for rental houses. (Komu, 2011) outlines four sources of demand for rental housing needs:

Graduates: They require rental housing since they are from different levels of schooling and hence are entering the labor market for the first time.

Transfers: For several reasons, some people may have to relocate from their previous settlements to different of urban areas. This may be due to reasons like work transfer, promotion or social reasons such as marriage, rural-Urban migration, expatriate population, etc.

There are various forms of tenancy tenure in the world which include: tenant, subtenant, thinka tenant, anticresis, Jjogbang, keymoney, and sharer tenant. In Tanzania, the most common form is a monthly tenant though most of the payment required is either on a semi-annual or annual

basis. Also, there are several forms of ownership: cooperatives, private landlords, public landlords, and Associations –NGOs (Komu, 2013).

Public rental housing in Tanzania is mostly confined to urban areas and accounts for almost 10% of the total stock; this extends to the employer's housing. Most of these public rental housing are owned by the National Housing Corporation (NHC). Public rental housing has faced setbacks such as policy changes like privatization of the public sector, adoption of home ownership strategies, inherent housing sector problems, and national economic problems (Komu, 2011).

Therefore, private housing has taken up many rental houses, especially in rural areas. Many people have resorted to private rental housing as a legal means of accessing housing (Alananga Sanga, 2017). The census conducted in 2012 in Kinondoni municipality, Dar es Salaam - Tanzania, shows private rental housing rights are held by around 50 percent of the 446,504 families (Alananga Sanga, 2015).

(Golland, 1996) defined Private sector housing management as any process not connected with the state's actions, neither directly constructed by the state nor financially sponsored by the state.

The three factors that are important in influencing the level of new house building are direct capital investments by the state for public housing, state support for production and consumption and changes in the profitability of house builders in the private sector (Golland, 1996).

Profit as a motivation underlying the private sector as argued by (Hancock, 1998), private housing landlords have aimed to gain income through the rental housing system hence the increase of rental houses due to tenants' demands which brings about the problem of managing them.

Currently, most of property managers manage their properties and the tenant details on paper and some via spreadsheet programs. Once a customer finds a vacant house, he can call or email the house manager indicating the size of the house he would like to be rented to him. Therefore, the Rental House Management System tends to minimize the problem brought about by the current system of managing the houses hence helping the landlords and managers to manage their rental houses easily.

The literature review reveals the evolving nature of house rental systems, driven by technological advancements, changing market dynamics, and user expectations. It highlights the significance of user experience, trust, and transparency in ensuring successful rental transactions. The

review also identifies information reliability, security, and legal compliance challenges, calling for further research and innovative solutions in the house rental domain.

Chapter 3

Methodology

3.1 Date collection

To effectively collect data for a house rental system, we consider the following approaches:

1. **Online Surveys:** Design and distribute online surveys to collect information from landlords and tenants. The surveys can cover various aspects such as rental preferences, satisfaction levels, desired features, pricing expectations, and feedback on the rental process. Platforms like Google Forms or Survey Monkey can be used to create and administer surveys.
2. **Interviews:** Conduct interviews with landlords and tenants to gather in-depth insights. These interviews can be structured or semi-structured, focusing on rental experiences, challenges faced, preferences, and suggestions for improvement. Interviews can be conducted in person, over the phone, or through video calls.
3. **Data Analysis:** Analyze existing data from rental platforms and websites. Many online rental platforms provide anonymized data that can be utilized to understand market trends, rental prices, demand patterns, and property features that are in high demand. This data can be obtained through partnerships or collaborations with rental platform operators.
4. **Focus Groups:** Organize focus groups consisting of landlords and tenants to facilitate group discussions on specific topics related to house rentals. Focus groups allow participants to share their experiences, opinions, and suggestions in a collaborative setting,

enabling researchers to gain multiple perspectives simultaneously.

5. **Data Mining:** Utilize web scraping techniques to collect data from rental websites and platforms. Web scraping can help gather information on property listings, rental prices, property features, and user reviews. However, ensuring compliance with legal and ethical guidelines is essential when collecting data through web scraping.
6. **Observational Studies:** Conduct observational studies by observing rental transactions and interactions between landlords and tenants. This approach can provide valuable insights into the rental process, including negotiation strategies, communication patterns, and challenges encountered during the rental journey.
7. **Collaboration with Rental Agencies:** Collaborate with rental agencies and property management companies to access their data on rental listings, tenant profiles, and rental agreements. This collaboration can provide comprehensive and reliable data for analysis. Remember to ensure privacy and confidentiality when collecting data and obtaining permission from participants or providers. Ethical considerations should be prioritized throughout the data collection process.

3.2 Project plan

The user will request data using a browser, and the requested info will be sent from the database.

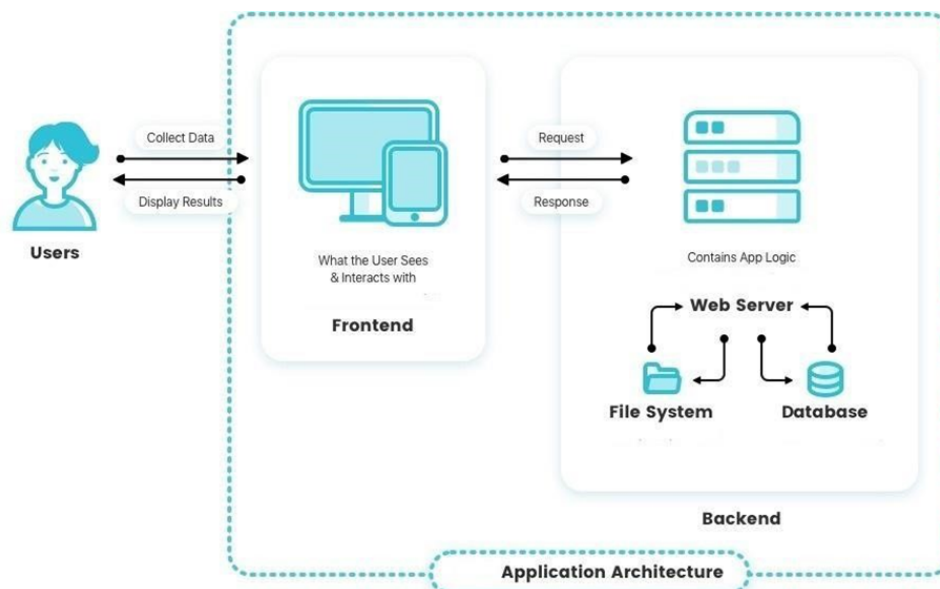


Figure 3.1: Software architecture

3.2.1 Architecture plan

Firstly, we'll build a basic UI layout and add basic functionalities to the website using HTML, JS, and CSS. Credential information will use a MySQL database, and the rest of the info, like house details and other relevant info.

3.2.2 Front-end plan

A total of 8-page templates in the plan-

1. Home page
2. User page
3. Search result page
4. House details page
5. Register/login page
6. Payment page

Map layout on search results will be visible using a map API.

3.2.3 Backend development

1. Account, Password, Bill:
 - (a) Sign up form
 - (b) Login
 - (c) Forgot Password
 - (d) Resend verification code
 - (e) PDF bill generator
 - (f) Communication via email
2. Searching facility:
 - (a) Facility based
 - (b) Block based

(c) Rent based

3.3 Development plan

The development methodology will adopt the agile methodology. Each cycle will complete in around two weeks. At the end of each cycle, we will launch a working website. We will run it, test the functionalities and review our progress. In the next cycle, we'll add/remove relevant features.

3.4 Work plan

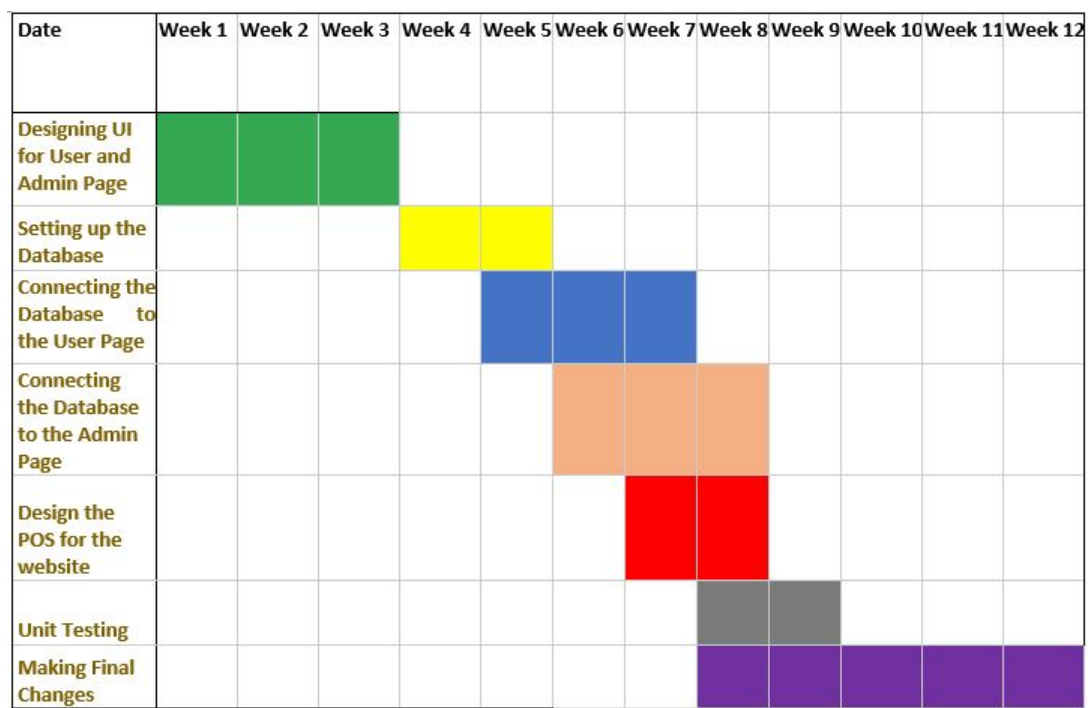


Figure 3.2: Gantt chart

Chapter 4

Proposed system

4.1 Proposed system

The user initially wants to sign up and create the account, and the user logs in, the system automatically will show the number of rented houses in particular places. With this information like the owner's name, house rent, address, and mobile number will, the user avoid the house broker, rent payment form, and registration form. Each form has command buttons like new, search, cancel, Back, and exit. With the command buttons, you can manipulate the database

4.2 Functional requirements

It required user participation and assertions of facts and assumptions that describe the system's expectations in terms of mission objectives, environment, restrictions, effectiveness measurements, and appropriateness.

- A system that improves the efficiency of information storage and retrieval.
- A system that is easy to learn and use
- A system that is fast in processing transactions
- A system that is flexible, safe, and convenient

This is a necessary task, action, or activity that was accomplished. The proposed system can:

- Allow the administrator to add houses, tenants, and defaulters' details
- Allow the administrator to delete houses, tenants, and defaulters' details
- Allow the administrator to search data in the database
- Allow the administrator to edit data in the database

4.3 Non-functional requirements

Accessibility, Usability, Documentation, Hardware and software considerations, Quality issues, security issues, User interface, human factors, Performance characteristics, Error handling, Extreme conditions, System modification, and Feasibility study...

- **Usability:** The system provides a help and support menu in all interfaces for users to interact with the system. The user can use the system by reading help and support.
- **Security:** The system provides a username and password to prevent the system from unauthorized access. The staff's password is greater than eight characters. The subsystem should provide a high level of security and integrity of the data held by the system, only authorized personnel of the company can give access to the company's secured page on the system
- **Performance Requirements:** The application should be scalable and should perform without any interruption for all the users. The system should have a high-performance rate when executing the user's input and should be able to respond within a short period.
- **Safety Requirements:** Login and visit the website should be safe for the users. That malware can't get into the device by browsing it.
- **Security Requirements:** All personal data should be protected, and no one can access further information without the administrator's permission. The system provides a username and password to prevent the system from unauthorized access. The user password must be greater than eight characters. Only users with valid passwords and usernames can view the users' page.
- **Hardware and software considerations:** The system provides a help and support

menu in all interfaces for users to interact with the system. The user can use the system by reading help and support. The system should always be available 24 hours, seven days a week. Also, in the occurrence of any significant system malfunctioning, the system should be available in 1 to 2 working days so that the business process is not severely affected. Considering the level of knowledge possessed by the users of this system, a simple but quality user interface should be developed to make it easy to understand and require less training.

- **Availability:** The system should always be available 24 hours a week. Also, in the occurrence of any significant system malfunctioning, the system should be available in 1 to 2 working days so that the business process is not severely affected.
- **Business Rules:** The application administrator has full permission to control the system
- **Error handling:** Error should be considerably minimized, and an appropriate error message that guides the user to recover from an error should be provided. Validation of users' input is highly essential. Also, the standard time to recover from an error should be 15 to 20 seconds.
- **Ease of use:** Considering the level of knowledge possessed by the users of this system, a simple but quality user interface should be developed to make it easy to understand and require less training.

4.4 System models

4.4.1 Use case diagram

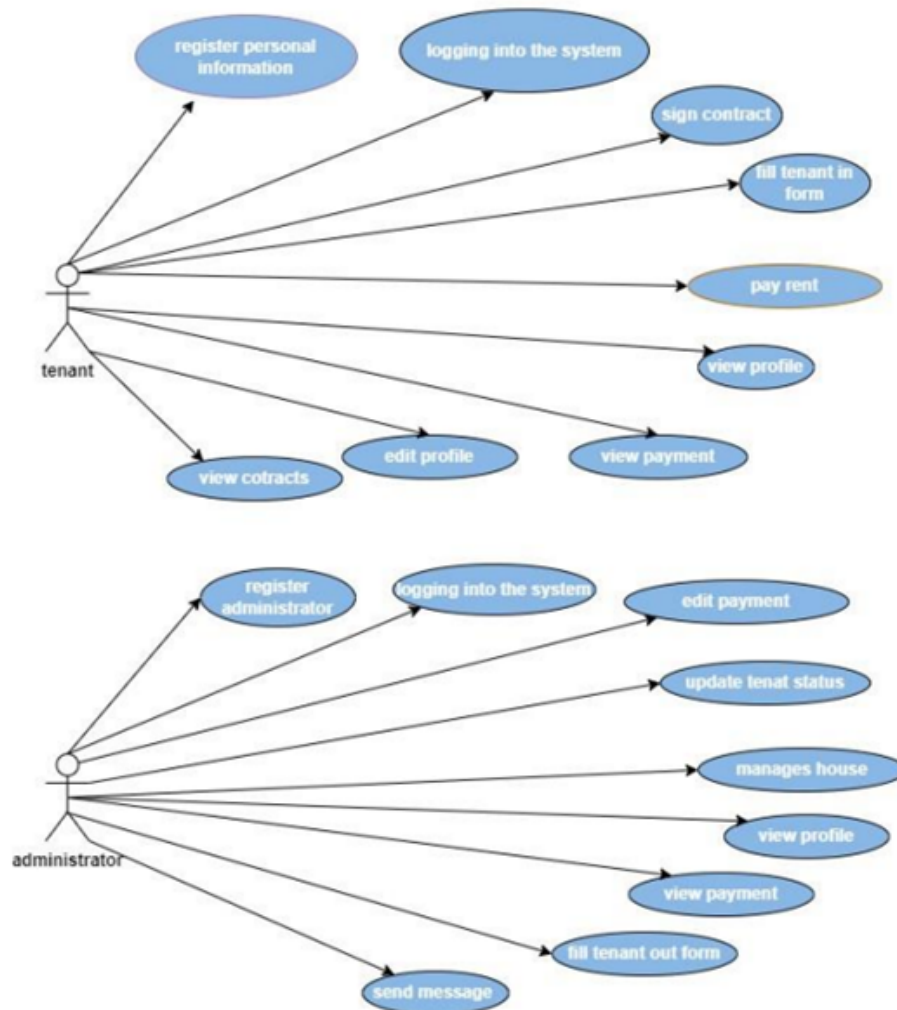


Figure 4.1: Use case diagram

4.4.2 Context diagram

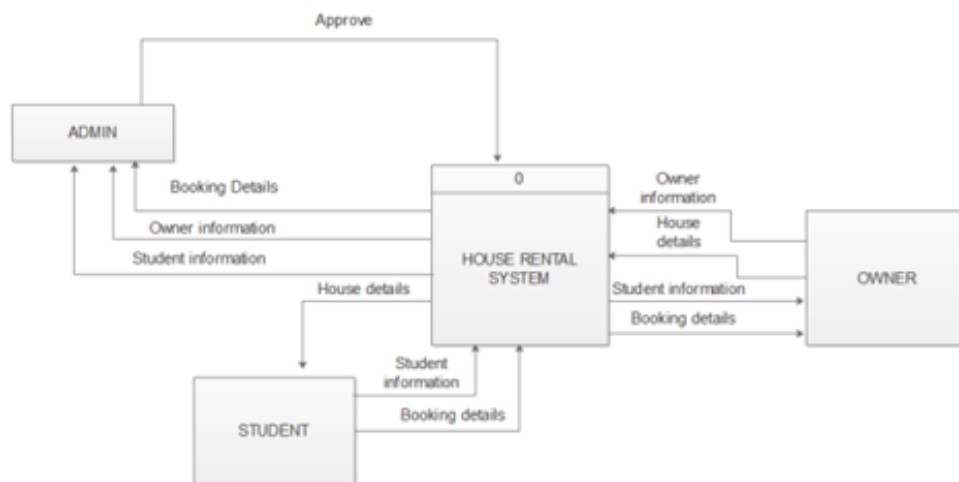


Figure 4.2: Context diagram

4.4.3 ER diagram



Figure 4.3: ER diagram

Chapter 5

Design contents

5.1 Introduction

Abashon is built to revolutionize the way people find and rent homes. Whether you are a tenant searching for the perfect place to live or a property owner looking to connect with potential tenants, Abashon provides a streamlined solution. Through careful attention to design, we aim to create an intuitive and enjoyable user experience that simplifies the process of finding, renting, and managing rental properties.

The success of any software project relies heavily on well-thought-out design decisions. Abashon's design content encompasses many considerations, all intended to enhance usability, security, performance, and scalability. By investing in user interface design, we strive to provide a visually appealing and intuitive interface that guides users seamlessly through the rental process. Additionally, robust system architecture ensures that the application can handle increasing traffic and user demands while maintaining a high level of performance.

Security is paramount when dealing with sensitive user information and financial transactions. Therefore, Abashon incorporates rigorous security design to safeguard user data and protect against potential threats. We adhere to industry best practices, implementing authentication, encryption, and secure communication protocols to maintain the privacy and integrity of our users' information.

Moreover, a well-designed database structure enables efficient storage and retrieval of property listings, user profiles, and transactional data. By carefully considering the database design, we ensure optimal performance and scalability as the platform grows and accommodates an ever-expanding user base.

5.2 Proposed system architecture

1. **Client-Side:** The client-side architecture of Abashon focuses on delivering a responsive and interactive user interface. It consists of the following components:

- **Web Browser:** The primary client-side component where users interact with Abashon through a web interface. It renders the user interface elements, handles user input, and communicates with the server-side components.
- **User Interface (UI):** The UI component encompasses the visual design, layout, and interactive elements of Abashon. It is responsible for presenting property listings, search functionality, user profiles, and other features to the users.
- **Authentication and Authorization:** This component handles user authentication and authorization processes, ensuring secure access to user-specific information and actions.
- **API Integration:** Abashon may integrate with external APIs, such as payment gateways or map services, to provide additional functionality, such as online payments or property location mapping.

2. **Server-Side:** The server-side architecture of Abashon is responsible for processing client requests, managing data, and orchestrating the application's core functionality. It comprises the following components:

- **Web Server:** A web server, such as Apache or Nginx, serves as the entry point for client requests. It handles HTTP requests, forwards them to the appropriate components, and returns responses to the client.
- **Application Layer:** The application layer contains the core business logic and acts as an intermediary between the web server and the data layer. It processes

client requests, validates input, performs operations on the data layer, and generates responses.

- **Database Management System (DBMS):** Abashon employs a DBMS, such as MySQL, to store and manage structured or unstructured data. The DBMS handles data retrieval, storage, and querying operations efficiently.
- **Data Access Layer:** The data access layer interfaces with the DBMS and provides an abstraction for accessing and manipulating data. It handles tasks such as connecting to the database, executing queries, and returning results to the application layer.
- **Authentication and Authorization:** This component manages user authentication and authorization processes on the server side. It validates user credentials, enforces access control rules, and ensures secure data interactions.
- **Caching:** To enhance performance, Abashon may incorporate a caching layer, such as Redis or Memcached, to store frequently accessed data, reducing the load on the database and improving response times.
- **External Services Integration:** Abashon may integrate with external services, such as email providers or SMS gateways, to enable features like notifications or user communication.

3. **Infrastructure and Deployment:** The infrastructure and deployment aspects of Abashon encompass the physical and cloud-based components required to host and run the application reliably:

- **Hosting Environment:** Abashon can be deployed on cloud platforms like Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP). This ensures scalability, high availability, and efficient resource management.
- **Load Balancer:** A load balancer distributes incoming client requests across multiple servers, improving performance and ensuring high availability by evenly distributing the load.
- **Scalable Server Cluster:** The application can be deployed on a cluster of servers to handle the increasing traffic and user demand. A scalable infrastructure allows

adding or removing servers dynamically based on usage patterns.

5.3 Sub-system decomposition

The Abashon home rental web app can be decomposed into several interconnected subsystems, each responsible for specific functionality. The following subsystem decomposition provides an overview of the major components and their relationships within the Abashon system:

1. **User Management Subsystem:** This subsystem handles user registration, authentication, and profile management.
 - **User Registration:** Manages user sign-up and account creation.
 - **User Authentication:** Handles user login and session management.
 - **User Profile Management:** Allows users to update their profiles, including personal information and preferences.
2. **Property Management Subsystem:** Description: This subsystem deals with property listing and management for property owners.
 - **Property Listing:** Enables property owners to create and manage property listings, including details, images, and availability.
 - **Property Search:** Provides search functionality for tenants to find suitable rental properties based on location, price, and other criteria.
 - **Property Booking:** Facilitates the booking process for tenants, allowing them to request property reservations and communicate with property owners.
3. **Payment Subsystem:** This subsystem handles payment processing and financial transactions related to property rentals.
 - **Payment Gateway Integration:** Integrates with third-party payment gateways to process rental payments securely.
 - **Payment Management:** Manages payment transactions, records, and associated financial data.
 - **Billing and Invoicing:** Generates invoices and manages billing information for property rentals.

4. **E-mailing Subsystem:** This subsystem enables the communication between tenants and property owners.
 - **E-mailing Interface:** Provides an interface for users to send and receive e-mails related to property inquiries, bookings, and negotiations.
 - **Notification Management:** Handles the delivery of system notifications and alerts to users.
5. **Integration Subsystem:** This subsystem integrates with external services and APIs to enhance functionality and provide additional features.
 - **Map Integration:** Integrates with map services to display property locations and provide navigation features.
 - **External APIs:** Integrates with third-party APIs for services like weather information, transportation, or local amenities.

5.4 System layout

The system layout for the Abashon home rental web app describes the physical and logical arrangement of components within the system. Here is a generalized system layout for Abashon:

1. **User Interfaces:**
 - **Web Browsers:** Users access Abashon through web browsers on various devices, such as desktop computers, laptops, tablets, and smartphones.
2. **Client-Side Components:**
 - **User Interface Layer:** Implements the presentation layer of Abashon, including the UI components responsible for rendering property listings, search functionality, user profiles, and other features. It is developed using front-end technologies such as HTML, CSS, and JavaScript.
 - **Front-end Framework:** Utilizes a framework like React, Angular, or Vue.js to facilitate component-based development, state management, and efficient rendering of UI elements.

- **Authentication and Authorization:** Handles user authentication and authorization processes on the client-side, ensuring secure access to user-specific information and actions.
- **API Integration:** Integrates with external APIs, such as payment gateways or map services, to provide additional functionality like online payments or property location mapping.

3. Server-Side Components:

- **Web Server:** Serves as the entry point for client requests and forwards them to the appropriate components. It handles HTTP requests and responses.
- **Application Layer:** Implements the core business logic of Abashon, processing client requests, validating input, and orchestrating interactions between components.
- **Database Management System (DBMS):** Stores and manages structured or unstructured data related to property listings, user profiles, transactions, and other system information. Popular options include MySQL, PostgreSQL, or MongoDB.
- **Data Access Layer:** Provides an abstraction for accessing and manipulating data within the DBMS. It handles tasks such as connecting to the database, executing queries, and returning results to the application layer.
- **Authentication and Authorization:** Manages user authentication and authorization processes on the server side, validating user credentials and enforcing access control rules.
- **Caching:** Incorporates a caching layer, such as Redis or Memcached, to store frequently accessed data, reducing the load on the database and improving response times.
- **External Services Integration:** Integrates with external services, such as email providers, to enable features like notifications or user communication.

4. Infrastructure and Deployment:

- **Cloud Hosting:** Deploys Abashon on cloud platforms like Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP). This provides scalability,

high availability, and efficient resource management.

- **Load Balancer:** Distributes incoming client requests across multiple servers, improving performance and ensuring high availability by evenly distributing the load.
- **Scalable Server Cluster:** Deploys the application on a cluster of servers to handle increasing traffic and user demand. The infrastructure dynamically scales by adding or removing servers based on usage patterns.

5. External Services:

- **Payment Gateways:** Integrates with third-party payment gateways to process rental payments securely.
- **Map Services:** Integrates with map services like Google Maps or Mapbox to display property locations and provide navigation features.
- **External APIs:** Integrates with third-party APIs for services like weather information, transportation, or local amenities.

5.5 User interface design

The user interface design for the Abashon home rental web app aims to provide a visually appealing, intuitive, and user-friendly experience for tenants and property owners. Here are key design considerations and elements for the Abashon user interface:

1. Branding and Visual Design:

- **Logo and Branding:** Create a distinctive logo and establish a cohesive branding identity for Abashon.
- **Color Scheme:** Use a visually pleasing color palette that aligns with the brand and creates a sense of trust and professionalism.
- **Typography:** Choose fonts that are easy to read and convey the desired aesthetic and tone.
- **Visual Elements:** Incorporate visually engaging icons, buttons, and other UI elements that enhance usability and reinforce the overall design style.

2. Homepage and Navigation:

- **Clear and Concise Homepage:** Design a clean and organized homepage that highlights the key features and benefits of Abashon.
- **Intuitive Navigation:** Implement a user-friendly navigation menu that allows easy access to property listings, search functionality, user profiles, and other important sections of the web app.

3. Property Listing and Search:

- **Property Showcase:** Display property listings with high-quality images, essential details, and an intuitive layout to attract and engage users.
- **Advanced Search:** Provide a robust search functionality that allows users to filter properties based on location, price range, amenities, and other relevant criteria.
- **Map Integration:** Incorporate a map view to visualize property locations and provide users with a spatial understanding of available rentals.

4. User Registration and Authentication:

- **Seamless Registration:** Streamline the user registration process with a clear and intuitive form, minimizing the steps required to create an account.
- **Secure Authentication:** Implement a secure and user-friendly login mechanism with options for password-based login or social media authentication.

5. User Profiles and Dashboard:

- **User Profile Management:** Provide users with the ability to create and manage their profiles, including personal information, saved searches, favorite properties, and booking history.
- **Dashboard:** Design a user-friendly dashboard that offers a comprehensive view of relevant information, such as ongoing or upcoming bookings, notifications, and account settings.

6. Booking Process:

- **Property Details:** Present detailed information about rental properties, including descriptions, amenities, pricing, availability, and owner contact details.

- **Booking Workflow:** Create a seamless and intuitive booking process that guides users through property selection, date selection, payment options, and confirmation.
- **Communication Channels:** Enable direct communication between tenants and property owners within the app to facilitate inquiries, negotiation, and clarifications.

Chapter 6

Implementation

6.1 Introduction

The chapter of this report focuses on detailing the process and outcomes of developing the Abashon home rental web app. This chapter provides insights into the technical aspects, challenges encountered, and solutions implemented during the development phase. It showcases the transformation of the design concepts and user interface into a fully functional and robust web application.

6.2 Algorithm development

Developing a home rental web app like Abashon involves implementing various algorithms to enable core functionality and provide a seamless user experience. Here are some key algorithmic aspects that could be considered during the development process:

1. **Property Search Algorithm:** This algorithm enables users to search for rental properties based on location, price range, amenities, and other criteria.
 - **Approach:** Implement an algorithm that efficiently filters and ranks property listings based on user input. Utilize indexing techniques, such as spatial indexing, to optimize search performance. Consider factors like property availability and user preferences to provide relevant search results.
2. **Booking and Reservation Algorithm:** This algorithm manages the booking and reservation process for users, facilitating property selection, availability checking, and confir-

mation.

- **Approach:** Implement an algorithm that handles the complex logic of managing property availability, handling overlapping bookings, and ensuring proper synchronization between the user interface and the database. Consider concurrency control mechanisms and transaction handling to maintain data consistency during the booking process.

3. **Payment Processing Algorithm:** This algorithm facilitates secure and efficient payment processing for rental transactions.

- **Approach:** Integrate with a trusted third-party payment gateway and implement a payment processing algorithm that ensures the secure transfer of financial data, verifies transaction details, and generates payment receipts or invoices.

4. **Notification Algorithm:** This algorithm enables real-time communication between users, including property owners and tenants, facilitating inquiries, negotiation, and updates.

- **Approach:** We've implemented an e-mailing algorithm that allows users to get instant e-mails on various updates.

These algorithmic aspects provide a starting point for developing the Abashon home rental web app. However, the specific algorithms and their complexity may vary based on the unique requirements and features of your application. It's essential to conduct thorough analysis, consider scalability and performance, and leverage existing algorithms or libraries to optimize development efforts.

6.3 Coding

Here is the link to our code: <https://drive.google.com/file/d/1xGV11MXorucnEfJh4mVFpwvDIxWZCGvw/view?usp=sharing>

6.4 Hardware and software acquisition

To successfully acquire the hardware and software components needed for the Abashon home rental web app, consider the following considerations:

1. Hardware Acquisition:

- **Web Servers:** Acquire servers or cloud-based hosting services capable of handling the expected traffic and scalability requirements of the web app. Consider factors like processing power, memory, storage, and network connectivity.
- **Networking Infrastructure:** Ensure a reliable and high-speed network infrastructure to support the web app's connectivity requirements. This includes routers, switches, firewalls, load balancers, and other networking equipment.
- **Storage Systems:** Assess storage needs for the application, including database storage, media storage for property images, and backup systems. Acquire suitable storage solutions like hard drives, solid-state drives (SSDs), or cloud storage services.

2. Software Acquisition:

- **Operating System:** Choose an appropriate server operating system, such as Linux distributions (e.g., Ubuntu, CentOS) or Windows Server, based on your team's expertise and compatibility with the web app's technology stack.
- **Web Server Software:** Acquire a web server software like Apache HTTP Server, Nginx, or Microsoft IIS to handle HTTP requests and serve the web application.
- **Database Management System (DBMS):** Select a DBMS that suits the requirements of Abashon, such as MySQL, PostgreSQL, or MongoDB. Acquire the necessary licenses or consider open-source options based on your needs and budget.
- **Programming Languages and Frameworks:** Identify the programming languages and frameworks required for the web app's development, such as JavaScript (for front-end development) and backend frameworks like Node.js, Ruby on Rails, or Django. Acquire the necessary licenses or ensure the availability of open-source libraries and frameworks.
- **Development Tools and IDEs:** Procure development tools and integrated development environments (IDEs) suitable for the chosen technology stack, such as code editors (Visual Studio Code, Sublime Text), version control systems (Git), and collaboration tools (Slack, JIRA).
- **External Services and APIs:** If Abashon integrates with external services or

APIs for features like payment processing or map services, ensure proper acquisition or subscription to those services.

3. Licensing and Legal Considerations:

- Ensure compliance with software licensing terms and any legal requirements associated with the acquisition and usage of the software components.
- Confirm that the acquired hardware and software components align with relevant regulatory and legal obligations, such as data privacy and security regulations.

6.5 Installation

To install this web app, the client has to load the SQL database, put the app folder in the host folder, and use the server computer to serve it to the client.

6.6 Testing

6.6.1 Unit testing:

First, we tested all the modules differently to check if they worked properly. Such as, we unit-tested the e-mail system, the registration system, the house management system, the user management system, the payment and map APIs, and the bill generator.

6.6.2 Integration testing:

For integration testing, we put multiple modules together to check if they can interact with each other and do not hamper their integrity. As all the modules depend on each other, this test is a must to check if everything is working properly.

6.6.3 System testing:

After integration testing, we put our web application together as a system and tested the entire system to find bugs and inconsistent outputs. We found several bugs and then fixed them.

6.7 Maintenance

Maintenance is crucial to keeping a home rental web app like Abashon running smoothly and ensuring its continued functionality and performance. Here are some key maintenance details to consider for Abashon:

1. Bug Fixes and Issue Resolution:

- Regularly monitor the application for bugs, errors, and user-reported issues.
- Prioritize and address identified issues promptly to maintain the app's stability and user satisfaction.
- Implement a bug tracking system or issue management process to track and resolve reported issues efficiently.

2. Performance Optimization:

- Monitor the performance of the web app, including response times, page load times, and server resource utilization.
- Identify performance bottlenecks and optimize code, database queries, and server configurations to enhance overall performance.
- Regularly test the application under different load conditions to ensure it can handle expected traffic levels.

3. Security Updates:

- Stay updated with the latest security vulnerabilities and patches related to the technologies and frameworks used in Abashon.
- Apply security updates and patches promptly to protect the application against known vulnerabilities.
- Conduct regular security audits, penetration testing, and code reviews to identify and address potential security risks.

4. Data Backup and Disaster Recovery:

- Regularly backup the application's data, including user information, property listings, and transaction history.

- Implement a robust backup strategy and schedule regular backups to ensure data integrity.
- Establish a disaster recovery plan to recover data and restore the application in the event of any unforeseen incidents or data loss.

5. Content Management:

- Regularly review and update the content of the web app, including property listings, images, descriptions, and terms of service.
- Ensure accurate and up-to-date information is provided to users, reflecting the current availability and details of rental properties.

6. Platform and Dependency Updates:

- Keep the underlying software platforms, frameworks, libraries, and dependencies used in Abashon up to date.
- Regularly update the web server, database management system, programming languages, and other supporting software to benefit from performance improvements, bug fixes, and security patches.

7. User Feedback and Feature Requests:

- Encourage users to provide feedback, report issues, and suggest new features or improvements.
- Evaluate user feedback and prioritize feature requests based on their value and feasibility.
- Communicate with users about the implementation of new features, improvements, and bug fixes.

8. Monitoring and Analytics:

- Set up monitoring tools and analytics systems to track the application's performance, usage patterns, and user behavior.
- Monitor server health, uptime, and error rates to proactively identify and address potential issues.

- Leverage analytics data to gain insights into user engagement, conversion rates, and areas for improvement.

Chapter 7

Conclusion and Recommendations

7.1 Conclusion

Effectively resolving the apartment issues is important to the buyer's long-term future, THE ABASHON aka Home Rental Project will be an important tool for creating rental housing stability by helping tenants/students of Bashundhara RA with greater credibility through initiating and documenting communications and building productive relationships with sellers. Home Rental project provides buyers of specific housing associations. Finally, the goal of the project is to create a better relationship between tenants and sellers which can be achieved through this project.

7.2 Recommendation

Some improvements can be recommended for this project, and they are-

- **Enhanced Search Filters:** Implementing advanced search filters would greatly enhance the user experience by allowing renters to narrow down their search results based on specific preferences such as pet-friendly properties, accessibility features, proximity to public transportation, and more. The inclusion of detailed filters will streamline the property selection process and help users find their ideal rental more efficiently.
- **Verified Property Listings:** Establishing a rigorous verification process for property listings would instill trust and confidence among users. Implementing measures such as property ownership verification, background checks, and validation of listing details would

help ensure the accuracy and authenticity of the properties listed on Abashon. This step would mitigate the risk of fraudulent listings and further enhance the platform's credibility.

- **Responsive Mobile Design:** In today's mobile-centric world, optimizing the Abashon House Rental website for mobile devices is crucial. A responsive design that adapts seamlessly to different screen sizes and resolutions would enable users to access and navigate the platform easily using their smartphones or tablets. Providing a mobile-friendly experience would cater to the needs of a broader user base and enhance engagement.
- **Reviews and Ratings System:** Implementing a transparent reviews and ratings system would facilitate the sharing of user experiences and foster a sense of community among renters. Allowing users to leave feedback and rate their rental experiences would assist future renters in making informed decisions. It would also motivate property owners to provide exceptional service and maintain the quality of their listings.
- **Personalized Recommendation Engine:** Utilizing machine learning algorithms and user data, developing a personalized recommendation engine would offer tailored rental suggestions based on a user's search history, saved properties, and preferences. This feature would save users time by presenting them with relevant listings that align with their specific needs and preferences.
- **Streamlined Rental Application Process:** Simplifying and streamlining the rental application process would enhance user satisfaction. Providing a standardized digital application form, enabling online document submission, and facilitating communication between renters and property owners within the platform would make the application process more efficient and convenient for all parties involved.
- **Proactive Customer Support:** Enhancing the customer support system by implementing proactive measures would further elevate the level of service provided. Proactively reaching out to users to address any concerns, offering assistance during key stages of the rental process, and promptly resolving inquiries would demonstrate a commitment to customer satisfaction and enhance the overall user experience.

References

- Alam, Mohammed, Zahid, Abid, Islam, Mohammed, and Khan, Mohammad (Mar. 2022). “Homies—An Online Web-based Platform to Find Cheap Accommodation for Travelers”. In: pp. 1540–1545. DOI: 10.1109/ICCMC53470.2022.9753715.
- Liu, Guangping and Chang, Xiayuan (2021). “The Impact of Rising Housing Rent on Residents’ Consumption and Its Underlying Mechanism: Empirical Evidence from China”. In: *SAGE Open* 11.2, p. 21582440211015709. DOI: 10.1177/21582440211015709. eprint: <https://doi.org/10.1177/21582440211015709>. URL: <https://doi.org/10.1177/21582440211015709>.
- Liu, Xiaojun, Yu, Mingqi, Cheng, Baoquan, Fu, Hanliang, and Guo, Xiaotong (2022). “Renting vs. Owning: Public Stereotypes of Housing Consumption Decision From the Perspective of Confucian Culture: Evidence From Event-Related Potentials”. In: *Frontiers in Psychology* 13. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2022.816004. URL: <https://www.frontiersin.org/articles/10.3389/fpsyg.2022.816004>.
- Voumick, Dipta, Deb, Prince, Sutradhar, Sourav, and Khan, Mohammad (July 2021). “Development of Online Based Smart House Renting Web Application”. In: *Journal of Software Engineering and Applications*. DOI: 10.4236/jsea.2021.147019.