# HaBHA: HANAP BOARDING HOUSE APARTMENT A MOBILE BASED APPLICATION FOR FINDING BOARDING HOUSE

KRENTZ JERALD GORDON

JAYVE JAMES JAUOD

OLIVER SUPAPO

AN UNDERGRADUATE CAPSTONE PROJECT PROPOSAL SUBMITTED TO THE FACULTY OF THE COLLEGE OF COMPUTING AND INFORMATION SCIENCES,

SOUTHERN CHRISTIAN COLLEGE, MIDSAYAP, COTABATO

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

JULY 2024

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| Title Page  Table of Contents  List of Tables  List of Figures  List of Appendices | | Page  i  ii  iii  iv  vi |
|  | | |
| CHAPTER 1 | INTRODUCTION  Project Context  Objectives  Scope and Limitation  Theoretical Framework  Conceptual Framework | 1  1  2  3  4  5 |
|  | | |
| CHAPTER 2 | REVIEW OF RELATED LITERATURE  Technical Background  Related Literature  Definition of Terms | 6  6  11  16 |
|  | | |
| CHAPTER 3 | DESIGN AND METHODOLOGY  Project Methodology  Project Design | 18  18  23 |

|  |  |
| --- | --- |
| Literature Cited | 47 |
| Curriculum Vitae | 55 |

LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| Table  1  2  3  4 | Title  Schedule Feasibility  Use-Case Specification – User’s Boarder  Use-Case Specification - House Owner  Use-Case Specification - Administrator | Page  25  30  31  33 |

LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| Figure  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | Title  Theoretical Framework  Conceptual Framework  Rapid Application Development  Data Flow  User System Flow  Owner System Flow  Admin System Flow  User’s Boarder Use Case  House Owner Use Case  Admin Use Case  User’s Boarder Sequence Diagram  Boarding House Owner Sequence Diagram  Administrator’s Sequence Diagram  Class Diagram  Entity Relationship Diagram  Login Page  Creating account Page  Registration Page  Dashboard Page  Edit Profile Page | Page  4  5  20  23  26  27  28  29  31  33  34  35  36  37  38  39  39  40  40  41 |

LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| Figure  21  22  23  24  25  26  27  28  29  30  31 | Title  Location Page  Search Page  Apartment Details  Payment Page  Owner Dashboard Page  Manage Establishment Page  Edit Profile Page  Notification Page  Manage Request Page  List Page  Manage Owner Establishment Page | Page  41  42  42  43  43  44  44  45  45  46  46 |

LIST OF APPENDICES

|  |  |  |
| --- | --- | --- |
| Appendix  A  B | Title  Statement of Work  Survey Questionnaire | Page  49  55 |

# CHAPTER 1

# INTRODUCTION

# Project Context

The exponential growth of Internet users has made e-commerce a promising business opportunity. In this market, there is little physical interaction between buyers, sellers, and payments. This unique feature has sparked the interest in mobility-supporting technology as an opportunity for business owners due to its potential to optimize selected business processes.

The use of mobility-supporting technology in e-commerce has been widely acknowledged as an interesting opportunity for business owners. It can enhance various aspects of particularly the e-commerce industry, in the areas of sales and service. The influence of social proof in online shopping has been identified as a critical factor impacting the sales of digital microproducts, making it a key area for future research in e-commerce.

The developers conducted a random interview within Poblacion 5 Midsayap North Cotabato to know what the different concern and difficulties are on finding boarding house in the area. Searching for an apartment or boarding house for rental or permanent residence has become a tough job. However, these tasks can be made easier with the help of mobile application websites that could assist people in finding a house. In the era of smartphones, mobile apps can make this job easier, allowing users to search for a house for rent or a home to buy more efficiently.

Boarders from far areas spend more time looking for accommodation. It takes more effort to go to every boarding house and apartment to check and inquire regarding the price or the monthly payment, amenities, location, their accommodation rules and regulations. So, the developers want to develop a Mobile application that called HaBHA (Hanap Boarding House Apartment) it is a mobile application that is designed to address the issues that have a friendly UI (User Interface) which could help the user navigate more efficiently and lessen the consumed time of finding an accommodation that suits the boarder as well as the boarding house owners. HaBHA is an application that could locate different boarding houses or apartments, it can also accept online booking and reservation, it can also filter the price and distance and can automatically tally the expenses.

The application will accommodate mostly boarders such as students, workers, and people with no permanent homes that are having a hard time finding a boarding house nearby it will also benefit the landowners by displaying its facilities to the application. Most of the Boarders especially when they are new in town don’t have any contact with any boarding house owner which makes it difficult to locate the available boarding houses nearby. The developers desire to address the concerns regarding the difficulties of both Boarders and Boarding house owners such as the location of the boarding house and Information about the Boarding house that will surely help both Boarders and Owners.

# Objective

The Capstone Project’s main objective is to Design and Develop an application that can address the process of finding affordable and quality boarding house or apartments within the area.

Specifically, it aimed to:

1. Allow users to precisely locate an apartment/boarding house with the assistance of Google Maps.
2. Use a filter to sort both price range and distance.
3. Accept online payment via G-Cash.
4. View all the included and excluded amenities of the selected boarding house.
5. Automatically tallies the total expenses.
6. Customize features of the boarding house.
7. USE (Usefulness, Satisfaction, and Ease of use) Survey.

# Scope and Limitations

The HaBHA will be able to locate the different Boarding houses within the Midsayap area. It can filter the price, location, and amenities of each boarding house. It can also cater online payment through G-cash and can accept online booking and reservation. In this application the owner can be able to display its facilities through this app. The application also automatically tallies all the expenses. The system can run to mobile devices, and it has an In-App notification for users to notify the user everytime they are connected to the internet. The application can generate revenue through the per transaction.

The Capstone Project is limited to only within Midsayap area. The application will require an internet connection to run. The data of the boarding house relies only on the data provided by the boarding house owner which makes some data not quite accurate because of recent changes. The online payment is only limited to G-cash and no other online payment. Approval for booking and reservation may take some time depending on the owner. Auto tally will not cater all amenities especially if the boarders have their own water and electric meter. And lastly some mobile devices, especially older version mobile phones may have trouble running this application due to modern technology.

# Theoretical Framework

The System “HAYBOL: An Android-Based Apartment Locator Application” uses Input-Process-Output (IPO) paradigm which includes Purposive Sampling and 2 sets of questionnaires in their input phase. The Process phase includes Requirement Planning, User Design, Construction and Cutover. In Requirement Planning three apartment owners in Siniloan, Laguna and 30 selected tenants were interviewed using focused- discussions. In the User Design the prototype of the mobile application was developed. The final suggestions for the improvement of the mobile application were integrated in Construction and lastly the Cutover this is where testing and users training were conducted which result to the output HAYBOL

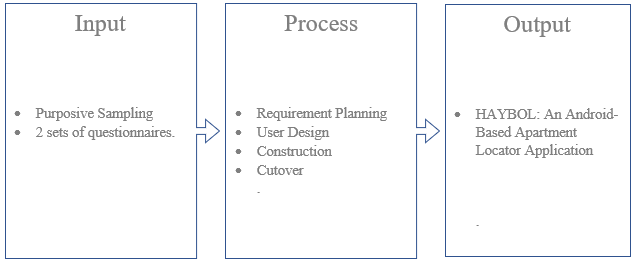


Figure 1. Theoretical Framework.

# Conceptual Framework

Developing the system HaBHA will also use Input-Process-Output (IPO) paradigm. The user will input their preferences such as location, amenities, and budget into the HaBHA interface. Additionally, boarding house and apartment owners provide details about their available spaces. The Process phase includes the application utilizes Google Maps API to display available accommodations based on user preferences. Users can make online bookings/reservations and payments through G-cash. Boarding house and apartment owners receive notifications of potential tenants and manage their listings. And then in the Output phase seekers receive a list of available accommodations meeting their criteria, along with booking and payment confirmation. Owners receive notifications of booking requests and tenant inquiries.

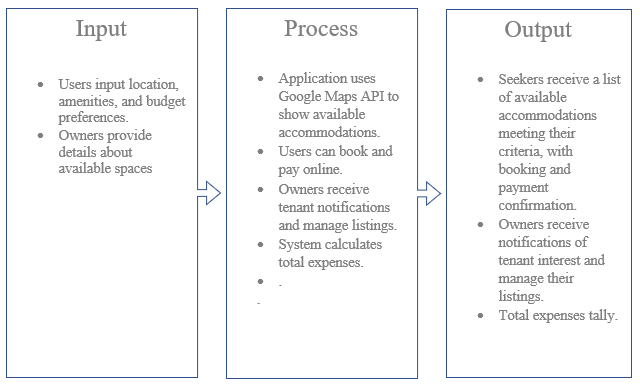


Figure 2. Conceptual Framework.

.

# CHAPTER 2

# REVIEW OF RELATED LITERATURE

# Technical Background

.NET MAUI

Britch et al. (2023) .NET MAUI or .NET Multi-platform App UI is a C#-based language and XAML framework designed to develop cross-platform systems and applications like Mobile Applications and desktops. It’s an expansion of Xamarin.forms. We can design applications across different platforms with .NET MAUI with one project folder or environment, but we can also choose one specific platform to develop. Its main goal is to empower developers to develop cross-platform application logic and user interface layouts that share only one code.

Britch et al. (2023) also state that .NET 6 or the updated version has specific-platform frameworks like .NET Android, .NET iOS, .NET macOS, and Windows UI 3 (WinUI 3) that share the same library with .NET Base Class Library (BCL). Each framework delivers a different execution or runtime environment. It can be written on a PC or Mac and compiled with its respective native application packages. Android application is compiled from C# to intermediate language (IL), if it’s launched, it will be compiled to native assembly just-in-time (JIT). In iOS application is compiled from C# to native ARM assembly code. In macOS applications use Mac catalyst with the help of UIKit, and AppKit, and APIs. Lastly, Windows applications are compiled using the Windows UI 3 library.

Britch et al. (2023) state that .NET MAUI delivers a package of controls and actions that displays data, identifies the application's activity, and retrieves and analyzes data. It has a well-detailed layout engine for designing pages, different navigation types, maintaining data binding, customizing handlers, and APIs that have features like GPS, accelerometer, detect changes in battery and network status. It also has constructed text-to-speech engines, events callback based on the registered URL like browser, and copy and paste of text in the system's clipboard. .NET MAUI and XAML have hot reload that enables developers to make changes to the code while the application is running, and the changes will automatically be applied without rebuilding the application again.

C# Language

Altexsoft (2021) discusses the good and bad of C# programming. C# is considered a great choice for Windows desktop apps, enterprise solutions, and even game development, as Unity game engine is built on C#. Object-oriented programming, or OOP, was the foundation of C# from the beginning. This concept of coding is predicated on your ability to describe the kind and structure of data and apply a set of standard functions to it. By organizing data into objects, OOP facilitates the division of an application into smaller, more manageable, and quickly combined parts. The grammar of the C# language is like human language, which makes it a high-level language with memory access capabilities. Stated differently, the hardware requires compilation of the C# code for it to comprehend its commands due to its high level of abstraction from machine code. Because their syntax is easier to comprehend and control than those of low-level languages like C, high-level languages are advantageous for developers.

The adaptability of C# is one of its best features. It can be applied to a variety of projects, including desktop, web, and mobile development. They are also perfect for developing cloud-based applications, which are becoming increasingly crucial in the current tech environment. As I highlighted in my last newsletter, we anticipate that applications will become more and more reliant on distributed systems as cloud computing grows.

G Cash

According to Cudis (2019) in his article " GCash Strengthens Dominance in PH ", GCash, being the leader in the mobile wallet segment, aims to continuously innovate and improve the user experience through added-value services and features. They also stated that most people who use mobile wallets prefer GCash among others because it charges low or no service fees for digital transactions. Users feel safe to use online transactions with GCash because it also provides users with increased security and peace of mind when using the app with the recent launch of Customer Protect, a program that provides compensation for unauthorized financial transactions on their GCash app.

Gudgeon et al. (2020) Emphasize the importance of layer-two block chain protocols, which are relevant in the context of mobile wallet applications. These protocols improve the security and efficiency of digital transactions, aligning with GCash's commitment to providing increased security to its users. The implementation of layer-two block chain protocols can further enhance the security features of mobile wallet platforms, contributing to the overall user experience.

Nagy et al. (2010) discuss the organizational adoption of open-source software, highlighting the barriers and remedies associated with such adoptions. This insight is relevant to the continuous innovation and improvement efforts of GCash, as the company can benefit from understanding the challenges and potential solutions related to technological advancements in the mobile wallet industry. By leveraging open-source technologies, GCash can further enhance its service offerings and features, contributing to its market dominance.

Google Maps

Law et al. (2019) examined the use of intelligent systems and technology for location-based services. They highlighted the importance of user-friendly interfaces and accurate mapping data for ensuring the effectiveness of mapping services. This finding is particularly relevant to the use of Google Maps for finding boarding houses nearby, as it emphasizes the need for accurate and reliable information to assist users in their search.

Arain et al. (2017) focused on location privacy protection for mapping services. They discussed the concept of mixed zones, which allow users to conceal their exact location while still receiving relevant search results. This research finding has implications for the use of Google Maps in finding boarding houses, as it raises important privacy concerns that should be considered in the development and use of mapping services.

Battin and Markande (2016) developed a location-based reminder Android application using Google Maps API. This application allows users to set location-based reminders, which can be useful for remembering the location of boarding houses or other accommodations. This finding demonstrates the potential for utilizing Google Maps in innovative ways to assist users in their search for boarding houses nearby.

Visual Studio

Visual Studio is a development environment that programmers use to create websites, web applications, web services, and mobile applications. It is a powerful tool with great debugging and editing experience (Murali, 2023). Visual Studio is known for its user-friendly interface and intuitive features, making it accessible for both novice and experienced developers.

It functions by providing developers with a comprehensive set of tools and features for writing, debugging, testing, and deploying code efficiently. Visual Studio offers a user-friendly interface, code editors, project management tools, and integrated debugging capabilities, streamlining the development process.

Visual Studio is an optimal choice due to its robust support for C# and .NET development, which are ideal technologies for building scalable and secure web applications. With Visual Studio, developers can easily manage the complexities of the project, maintain a consistent codebase, and ensure seamless integration with other components of the system. Additionally, Visual Studio provides collaboration features and a rich ecosystem of extensions, enhancing productivity and enabling the creation of a sophisticated and reliable online system.

# Related Literature

Hamzah et al (2022), boarding houses have become a somewhat essential necessity, particularly for students who are studying abroad and do not want to spend a lot of time or energy traveling. Students whose homes are located far from the university also use boarding houses as temporary housing.In most college towns, there is a shortage of off-campus, student-Friendly housing puts rentals in high and constant demand. Most tenants in an apartment are students enrolled in a higher education program that operates on a term or semester system. In this regard, recognized that finding an apartment in the Philippines is difficult when unfamiliar with a place and environment. In addition, the location of an apartment plays a significant role in choosing the neighborhood because one needs to consider security and personal convenience. Note-worthy is also the fact that students'" study habits are also affected by living in an apartment away from the school. Students who perform better academically are those living in apartments compared to those living in their homes due to their struggle of travel (Consignado, 2017).

Hamzah and Findawati (2021) Based on the statistics from Pangkalan Dikti 2020 and BPS Sidoarjo, Sidoarjo has about 28,000 migrant students who are on productive age, meaning they are between the ages of 19 and 29 and about 16,000 active college students. Every migrant college student attending a Sidoarjo college leaves their home to reside in a temporary or permanent location in Sidoarjo. Additionally, some recognized that one of the advantages of living in an apartment is that when a student is in the room, all the things he or she needs, like books, notes, and other study materials, are already present.

Prasetyo and Hudallah (2018). Students and workers from outside the area reside in boarding houses, which are temporary housing consisting of rooms of varying sizes. It takes a lot of energy, money, and time to locate the boarding house outside. Therefore, it is essential to design an information system that can assist in resolving this issue. This research aims to create a geographic information system that allows users to search for boarding houses based on their preferences. The waterfall development method is applied in this study. PHP is the programming language of choice, along with MySQL and AJAX implementation for more interactive data collection. A satellite image can be obtained from the boarding house coordinates by using API 3 on Google Maps. A coordinate point is needed for the satellite imagery to be displayed on this database. The system can also analyze the coordinates' distance by using the haversine algorithm.

Hamzah and Findawati (2021) aim to assist college students in locating suitable boarding houses near their academic institutions. One approach is gathering information from Sidoarjo boarding houses and turning it into a database for boarding house owners and college students. The study produced an information system that loads data on boarding houses in Sidoarjo based on geography and includes features that make it easier for college students to find suitable boarding houses. The study will help college students find a suitable and nearby boarding house more efficiently and assist boarding house owners by providing them with marketing materials for their establishment. Users who look for the closest boarding house can make use of the boarding house route search information system's many capabilities, which include boarding house routes and thorough information browsing. Data can be updated frequently since boarding house owners find it easy to manage their data. The purpose of their project is to develop a location information system for boarding houses using Android. First, a system that has been designed for research purposes is created (Hamzah et al., 2022).

The most popular app for hunting a house is Hosing.com. With the help of this app, one can post a flat for sale, find a new apartment to rent or buy, search for a hostel, and look up new localities. This app has detailed data and pictures of every property. One can use a filter and search for every specific detail needed. The Common Floor property app can help individuals search for real estate projects and properties. This app could be used for searching, buying, renting, and selling properties like apartments, flats, houses, villas, residential projects in India. It has innovative filter options, which Search properties by using map search. Additionally, another mobile, the app 99acresis one of the housing apps that could be used for searching a place to buy, sell, or rent. It is widely used in India. Moreover, Magic Brick is another application that can assist a person interested in buying, selling, or renting properties. It gives detailed and latest information about properties in India. It has a map facility so that a user can visually explore the area near the property. It has photos and videos of the neighborhood for a customer to see. Magic Bricks, however, does not disclose details until it verifies the details of another person. Lastly, Property Plus which uses the search property to buy and rent via the mobile. This app is easy to use, and with its memory requirements, it takes less memory space to installMoreover, running on mobile. This app redirects one to other housing-related apps like 99acres, Apartment Guide, Back page, Click, Century21, eBay Classified, Flat-club, Homefinder.com, and among others (Consignado et al., 2017).

The advancement of technology has made it simpler for people to locate all available information. An application for searching for boarding houses makes it easier for students to find information about the boarding houses they are interested in. This program is compatible with Android devices and provides users easy access to boarding house profiles and facilities. It also has GPS functionality to locate boarding house locations, making it a straightforward process to locate them. Ninety percent of respondents to the study agree that there should be a search application for boarding houses. Students may more easily obtain the information they need about boarding houses and make price choices, due in part to this app, which is highly helpful for a variety of parties such as houses managers. Boarding house managers can advertise their own boarding houses (Widiastuti et al., 2021).

According to Manalu et al. (2016), a person would call the boarding house management and visit the location immediately when looking for a boarding home or other type of room rental. Time and effort are required for this task. An Android application is being created to optimize the procedure and make it easier. The application's chat function and push notifications are crucial components that let users obtain more recent information from management. But now, there isn't a smartphone application for renting a room or a house with push notification capabilities and built-in chat. This project aims to create and assess a smartphone application that provides information about boarding houses via chat and push notifications. This program has the following features: chat, broadcast information, see boarding rooms list, view detail, review, view map, and log in. According to a review, the app is helpful for connecting building managers and room seekers, as well as for providing information about boarding houses.

The Google Maps Application Programming Interface (API), Yahoo! Maps API, Microsoft Bing Maps API, Nokia Ovi Maps API, and ESRI ArcGIS API have all recently seen a rise in interest in developing online map services. Application developers use the Maps API as a platform to combine spatial data from multiple sources to create new customized services. This term is commonly used to refer to these services as "mashups." Online mapping applications have revolutionized because of Maps API's utilization. Still, the map "mashups" have two main shortcomings. First, the application developer prepares a small amount of typically non-secured spatial data using open-source techniques like XML, Fusion Tables, CSV, or KML. These techniques aren't appropriate for data sources in the format of a commercial database kept on a protected data server. Second, while map "mashups" emphasize the usage of the Maps API platform to supply customized services or data quickly, they typically lack advanced features and user-friendly interfaces that allow the user to edit the data. The purpose of their work is to provide an online mapping application that enables extensive functionalities for the user to edit data but requires access to data sources in the form of a commercial database kept on a secure data server. A case study of creating a web map service for the People's Garden Initiative of the United States Department of Agriculture (USDA) to showcase tens of thousands of gardens online. Spry, Microsoft SQL database, Microsoft aspx.NET, Google Maps API, and Google Geocoderer. This online map application is developed using the Ajax framework (Hu & Dai, 2013).

# Definition of Terms

**C#.** (C Sharp) is a powerful, object-oriented programming language developed by Microsoft for building a variety of applications, from desktop software to web and mobile apps. The developer refers C#, it is often considered the best choice for development due to its versatility, robustness, and extensive ecosystem, offering developers a wide range of tools and capabilities to efficiently build high-quality software solutions across various platforms and domains.

**XAML.** (Extensible Application Markup Language) is a declarative XML-based language used primarily for designing user interfaces in applications developed for platforms such as Windows Presentation Foundation (WPF), Universal Windows Platform (UWP), and Xamarin.Forms. Developers rely on XAML to structure and organize the HaBHA App. Developer rely on XAML for using MAUI due to its declarative syntax, familiarity, powerful data binding, styling capabilities, and robust tooling support.

**API.**  Stands for Application Programming Interface. It's a set of rules, protocols, and tools that allows different software applications to communicate with each other.

**Visual Studio.** It is an integrated development environment (IDE) created by Microsoft, used primarily for building software applications across various platforms such as Windows, web, mobile, and cloud. The developers will utilize Visual Studio to create the HaBHA App, providing resources for front-end design (including XAML and others), backend development (such as .NET MAUI), debugging, and deploying to either web servers or cloud platforms.

**OOP**. Stands for Object-Oriented Programming. It's a programming paradigm that organizes software design around objects, which encapsulate data and behavior, promoting modularity, reusability, and maintainability in software development.

# CHAPTER 3

# DESIGN AND METHODOLOGY

# Project Methodology

Rapid Application Development (RAD) will be used in developing an application that is shown in figure 3. This methodology makes things done quickly and effectively. With RAD, teams collaborate closely and work in short bursts called iterations. It can adjust their plans easily based on feedback from users and stakeholders. By breaking big tasks into smaller, more manageable ones, RAD makes the whole process smoother. Plus, RAD focuses on user needs, the result is more likely to be something that people actually want and find useful. In today's fast-paced world, where technology and customer demands are always changing, RAD offers a flexible and efficient approach to getting projects done. This methodology has four (4) phases: Requirements Planning, User Design, Construction, and Cutover.

In the study of Daud et al. (2010), it is suggested that a system be constructed utilizing the Rapid Application Development approach to manage practical training applications, as managing the application process manually is a laborious operation. The methodology was selected because it enables quick system installation in a real-world setting. Beginning in the middle of the 2009 semester, which ran from December to April, the development process was iterated according to the requirements gathered. The rapid application development technique proved to be the appropriate approach for this project, as it satisfies the purpose of speedy implementation for a small size system. The practical training application system had been implemented for three semesters.

In the Development of Geographic Information Systems of Sasmito et al (2020). It is a computer system that can record, store, write, analyze, and display geographical data. They use the Rapid Application Development (RAD) method, GIS was created on a website platform to give the public access to data on industry profiles, production types, investment values, industry maps, and industrial sites in each town and subdistrict. In contrast to the waterfall method, the RAD method was selected since it could speed up developing websites.

In the study of Suwandi et al. (2022) which is Implementation of a school information system. The purpose of this study is to support the school and parents in handling student grades, school data about extracurricular and academic activities, and teacher-parent communication. The method used is Rapid Application Development (RAD). This aims to deliver significantly faster development and effectively complete the steps of requirements planning, workshop design, rapid application development, and implementation in 90 days.

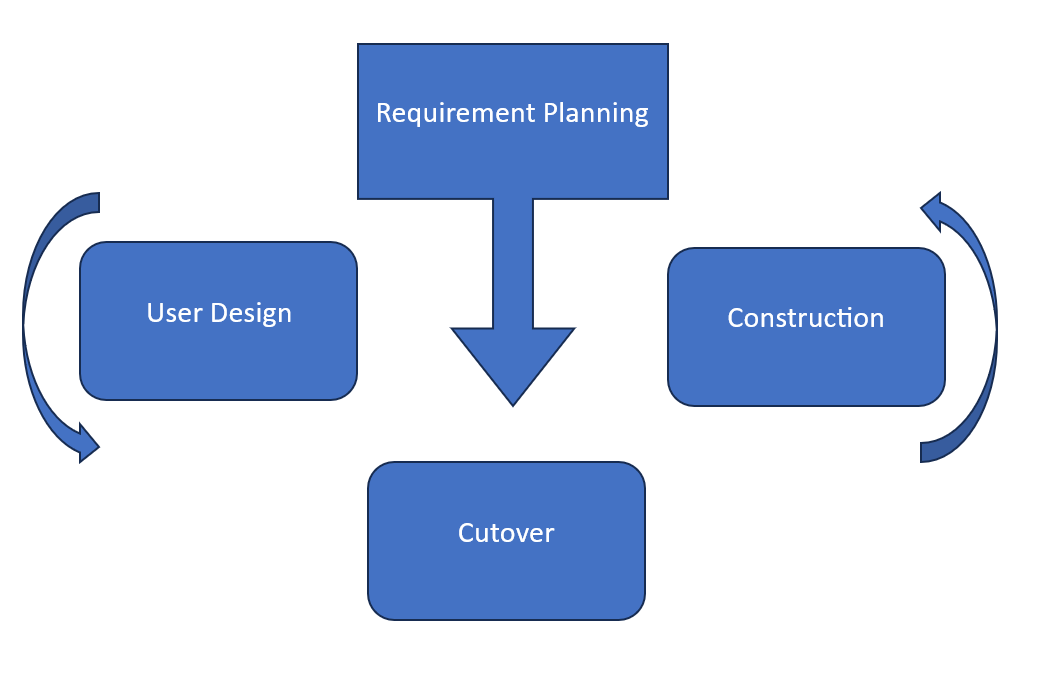


Figure 3. Rapid Application Development.

**Phase 1 – Requirements Planning**. In the Requirements Planning phase of the RAD methodology, developers and stakeholders will collaborate closely to define the project’s objectives, scope, and user requirements. Developers will engage in intensive discussions with stakeholders, including boarders, to gain a comprehensive understanding of the desired functionality and features of the application. They will employ various techniques such as interviews and brainstorming sessions to elicit requirements effectively. Simultaneously, stakeholders will actively participate by providing insights into their needs and preferences. They will communicate their expectations and desired outcomes to the development team, ensuring alignment with organizational goals and user needs. Through this iterative and collaborative process, developers and stakeholders will work together to establish a solid foundation for the subsequent phases of development, having a shared understanding of the project's objectives and laying the groundwork for successful implementation.

**Phase 2 – User Design.** In the User Design phase in RAD methodology, developers and stakeholders will collaborate to translate the gathered requirements into user-friendly designs that meet the needs and expectations of the end-users. This phase will emphasize rapid prototyping and iterative design cycles, allowing stakeholders to visualize the proposed solution early in the development process. Stakeholders will actively participate in reviewing and providing feedback on these prototypes, ensuring that the design aligns with their expectations and addresses their requirements effectively. Through continuous iteration, developers and stakeholders will work together to refine the user interface, improve usability, and optimize the overall user experience of the software application. This collaborative approach will foster a sense of ownership and engagement among stakeholders, leading to the development of a solution that truly meets the needs of the end-users.

**Phase 3 – Construction.** In the Construction phase of RAD methodology, developers will use what they have gathered from the Requirements Planning and User Design phases to rapidly build the software application. Developers will work closely with stakeholders to prioritize features and functionalities, ensuring that the most critical requirements are addressed first. They will employ various programming languages, frameworks, and tools to implement the designed solution efficiently. Continuous integration and testing practices will be adopted to ensure the quality and reliability of the software application throughout the development process. Stakeholders will actively participate in reviewing and testing each iteration, providing feedback to the development team for further refinement. Through this collaborative and iterative approach, developers and stakeholders will work together to iteratively build and refine the software application, ensuring that it meets the evolving needs of the end-users and stakeholders.

**Phase 4 – Cutover.** In the Cutover phase in RAD methodology, developers together with 3 landlords and 10 boarders will collaborate to deploy the software application into the production environment smoothly. Developers will finalize the software and ensure that it’s ready for use by conducting comprehensive testing and addressing any remaining issues or bugs. They will also prepare documentation and training materials to support end-users in using the new system effectively. Meanwhile, stakeholders, including end-users and boarders, will participate in user acceptance testing to validate that the software meets their requirements and expectations. They will also provide feedback on the deployment process and identify any additional training needs. Together, developers and stakeholders will ensure a seamless transition to the new system, providing ongoing support and maintenance to ensure its continued success in meeting organizational goals and user needs.

# Project Design

# Requirement Analysis

**Data Flow Diagram**. Figure 4 illustrates the process of the existing business process of inquiring about boarding houses. The boarders give its information to inquire Boarding House and the Boarding house Owner Receive boarders’ information, the boarding house owner give an approval and availability of boarding house, then the boarders give a payment to the boarding house owner and the boarding hose owners give the payment receipt to the boarders to prove that already paid.

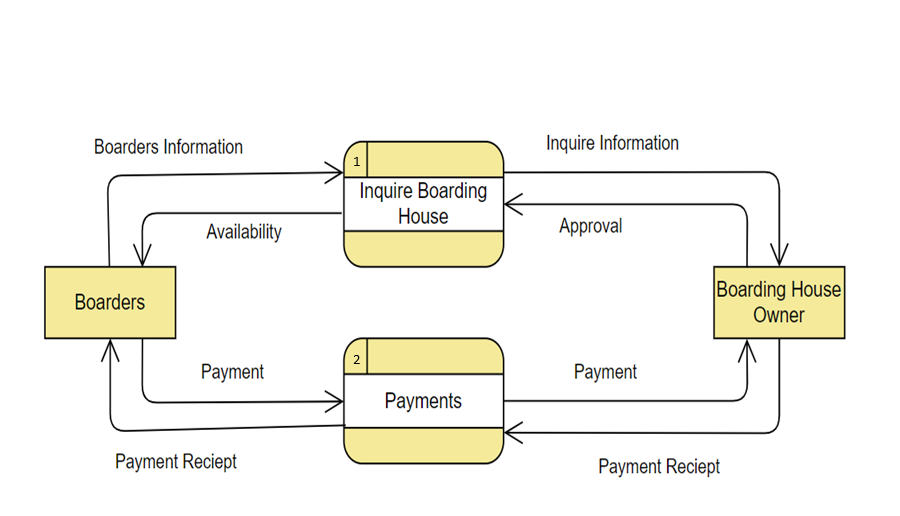


Figure 4. Data Flow Diagram.

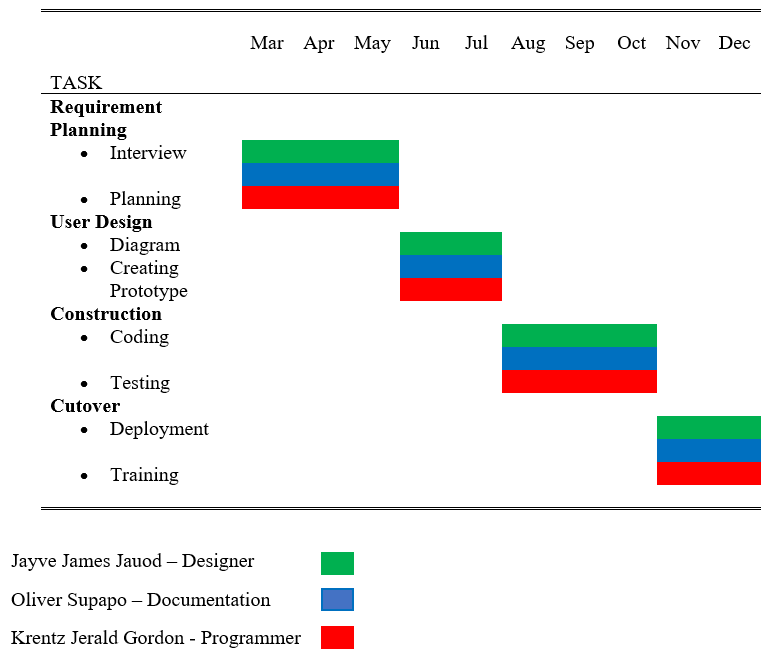
# Technical Feasibility

Upon assessing the outline design of the system requirements, the HaBHA (Hanap Boarding House Apartment) mobile application is technically feasible. The hardware requires smartphones and server for the backend of the system. Software requires a mobile app development environment, database management system, and third-party APIs for payment processing and notifications. When it comes to the network the user needs to be connected to the internet for registration, login, and searching for boarding houses. The developers themselves will develop the mobile application, database management and its server. The platform of HaBHA has sufficient needs for the future needs because of its scalable architecture, flexible infrastructure and performance optimization. A prototype will be required to gather feedback from potential users.

# Schedule Feasibility

Table 1 shows the outline of the Gantt chart of the project. The Gantt chart visualizes the project’s timeline, including start and completion dates, as well as the contributions of each team member. It aims to outline the project’s development tasks and their scheduled completion within a specified timeframe. The chart serves as a visual representation of the project’s progress and helps stakeholders understand the timeline and individual responsibilities.

Table 1. Schedule Feasibility.



# User’s Flow Diagram

Figure 5 shows the System flow of the user. First the user will login then if the user has the valid credentials, then he can proceed to the dashboard. In the dashboard the user can be able to search, filter, edit profile and log out. After searching and filtering the user can be able to book and pay for it online then the system will end.

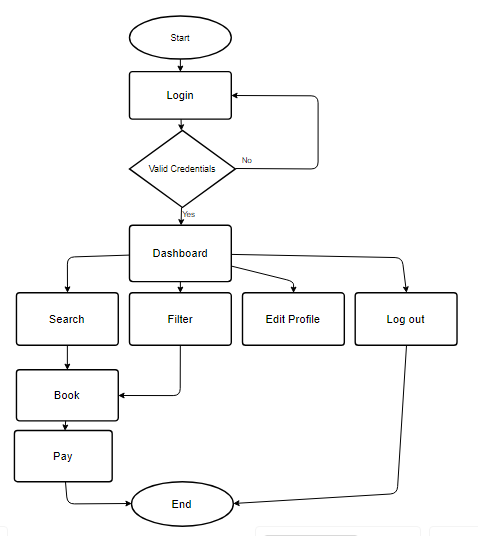


Figure 5. System Flow User.

# Boarding House Owner’s System Flow

Figure 6 shows the Data Flow of the Boarding House owner. The owner will login to the system if the credentials are valid and the owner will proceed to the dashboard. In the dashboard the owner can be able to see the booking notifications, edit its establishments such as its amenities, edit its profile and log out. After checking the booking notification, the owner can accept or decline the bookings of the users. If the owner accepts the user can pay it and the user shall receive the payment and then the owner will update the availability of its boarding house.

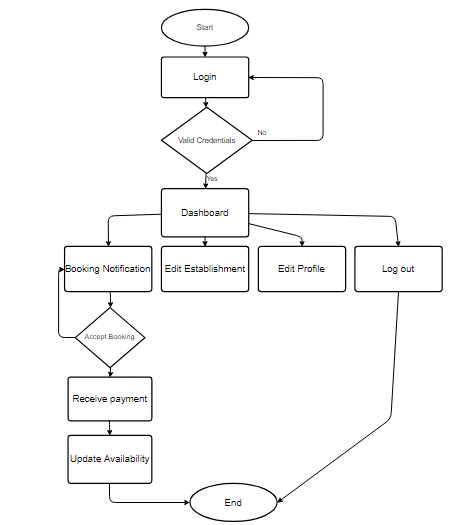


Figure 6. Owner System Flow.

# Administrator’s System Flow

Figure 7 shows the Data Flow of the Admin. The admin will login to the system and check if its credential is valid. If the credentials are valid the user can proceed to the dashboard. The admin can be able to See and manage the Boarding House Application of the boarding house owner, Book user if there are problems when it comes to booking the admin can be able to book users, edit owner establishment just like in the booking, manage users and log out. After checking the credibility of the boarding house, the admin can be able to approve or decline the application of the owners. If the booking credentials are valid the admin will approve the boarding house application. In the booking feature of the admin the admin can be able to book users and update the availability of the room.

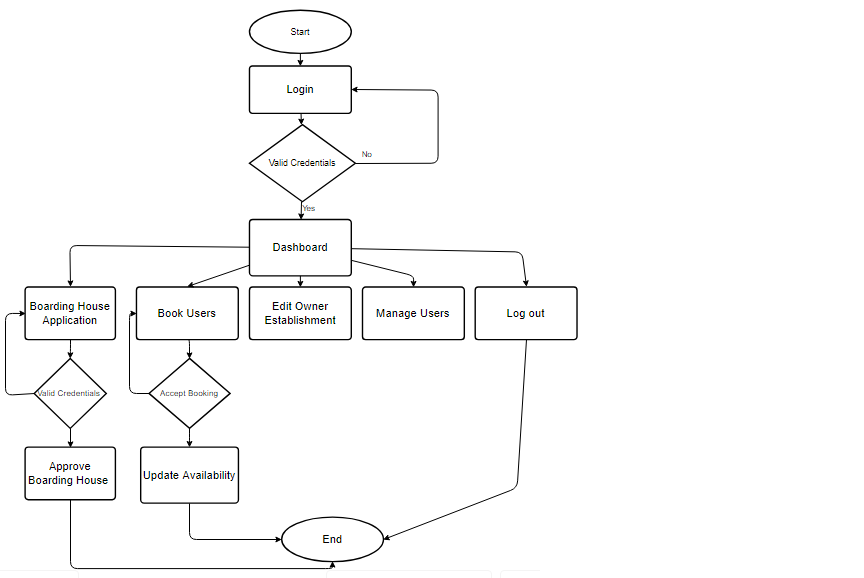


Figure 7. Admin System Flow.

# User’s Use Case Diagram

Figure 8 shows the use case diagram of a user or a boarder. It illustrates how the boarder interacts in the system.

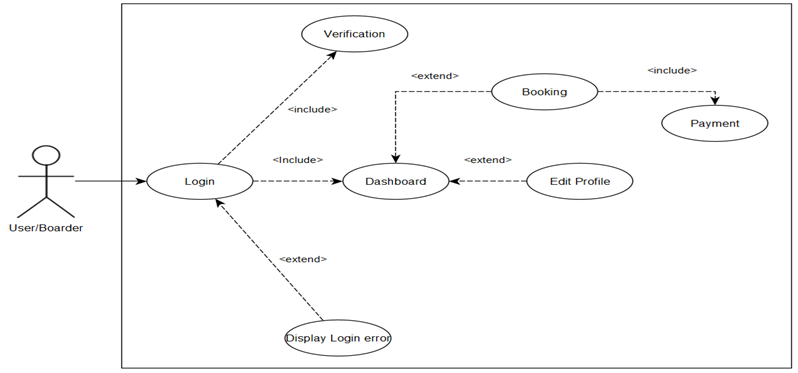


Figure 8. Boarder Use Case User.

# User’s Boarder’s Use Case Specification

The table provides a detailed description of the User/Boarder use case, including its brief description, pre-condition, flow of event, and post-condition transactions for accessing system features.

Table 2. User’s Use Case Specification.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Case** | **Brief Description** | **Pre-Condition** | **Flow of Event** | **Post-Condition** |
| Login | This use case outlines the process by which a user login his/her account in the system. | User must have an account | User accesses login page, inputs username and password, system verify, directs user to Dashboard if valid. | User is logged in and directed to Dashboard. |
| Booking | This use case outlines the process by which the user has a choice and booked it prefer establishment. | User logged in and on dashboard. | The user clicks the Booking tab on dashboard and book it prefers establishment and make a payment. | After booking the user make a payment for its booking. |
| Edit Profile | This is where the user can edit his/her profile. | User logged in and on dashboard. | User click the Edit Profile tab on dashboard and edit his/her profile. | User edit his/her profile. |
| Payment | The is where the user can pay through online | User booked his/her prefer establishment | User logged in and on dashboard the user clicks the Booking tab and booked his/her prefer establishment and make payment. | User pay and get receipt after payment |

# House Owner Use Case Diagram

Figure 9 shows the use case diagram of a House owner. It illustrates how the House owner interacts in the system.

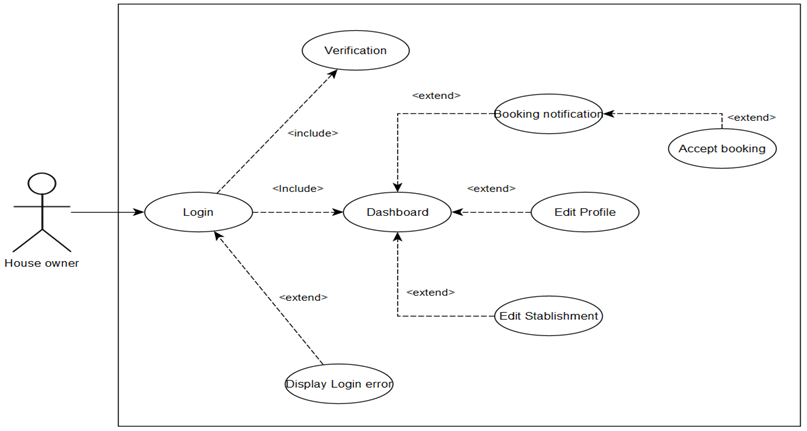


Figure 9. House Owner Use Case.

# House Owner’s Use Case Specification

The table provides a detailed description of the House owner’s use case, including its brief description, pre-condition, flow of event, and post-condition transactions for accessing system features.

Table 3. House Owner Use Case Specification.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case** | **Brief Description** | **Pre-Condition** | **Flow of Event** | **Post-Condition** |
| Login | This use case outlines the process by which a user login his/her account in the system. | User must have an account. | User accesses login page, inputs username and password, system verify, directs user to Dashboard if valid. | User is logged in and directed to Dashboard. |

Table 3 Continued

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case** | **Brief Description** | **Pre-Condition** | **Flow of Event** | **Post-Condition** |
| Booking Notification | This use case is notification for the booked establishment. | User logged in and on dashboard. | Users click the Booking Notification tab on dashboard then user have a choice whether to accept or not. | Users choose whether to accept or not the booked establishment. |
| Edit profile | This is where the user can edit his/her profile. | User logged in and on dashboard. | User click the Edit Profile tab on dashboard and edit his/her profile. | User edit his/her profile. |
| Edit Establishment | The is where the user can edit his/her establishments. | User logged in and on dashboard. | User click the Edit Establishment tab on dashboard and edit his/her establishments. | User can edit his/her establishments. |

# House Owner’s Use Case Diagram

Figure 10 shows the administrator’s use case diagram. It illustrates how the admin interacts in the system.

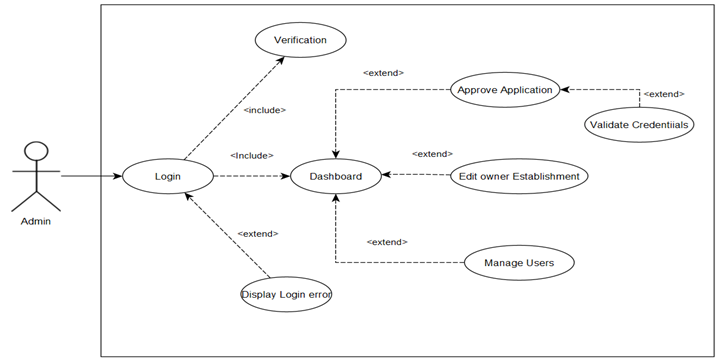


Figure 10. Admin Use Case.

Administrator’s Use Case Specification.

The table provides a detailed description of the Administrator's use case, including its brief description, pre-condition, flow of event, and post-condition transactions for accessing system features.

Table 4. Administrator’s Use Case Specification.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case** | **Brief Description** | | **Pre-Condition** | | **Flow of Events** | **Post-Condition** |
| Log in | | This use case outlines the process by which an administrator logs his/her account in the system. | | User must have an account. | User accesses login page, inputs username and password, system verify, directs Admin to Dashboard if valid. | User is logged in and directed to Dashboard. |
| Approve Application | | The user makes an approval for the application of the house owner. | | User logged in and on dashboard. | Users click the Boarding House Application tab on dashboard. | Users choose Whether to approve or not the application |

Table 4 Continued

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case** | **Brief Description** | | **Pre-Condition** | | **Flow of Events** | | **Post-Condition** | |
| Edit Owner’s Establishment | | The admin may edit on each establishment of the house owner | | User logged in and on dashboard | | Users click the Edit Owner Establishment tab on dashboard | | User can edit the establishment of the house owner | |
| Manage users | | The user managed the user of the system. | | User logged in and on dashboard. | | User click the Manage User tab on dashboard. | | Admin can manage the users. | |
| Validate Credentials | | The admin validates the credentials of the application of the House Owners. | | Admin checked if it has an applicant in Boarding house Application tab. | | Admin clicked the Boarding house Application tab and checked if it has an applicant. | | Admins validate the Applicant credentials. | |

User’s Boarder Sequence Diagram

Figure 11 shows the sequence diagram of the User/Boarder. It illustrates the step-by-step interactions between the User, System, User Database, Establishment Database, Payment Database, and Google Maps API.

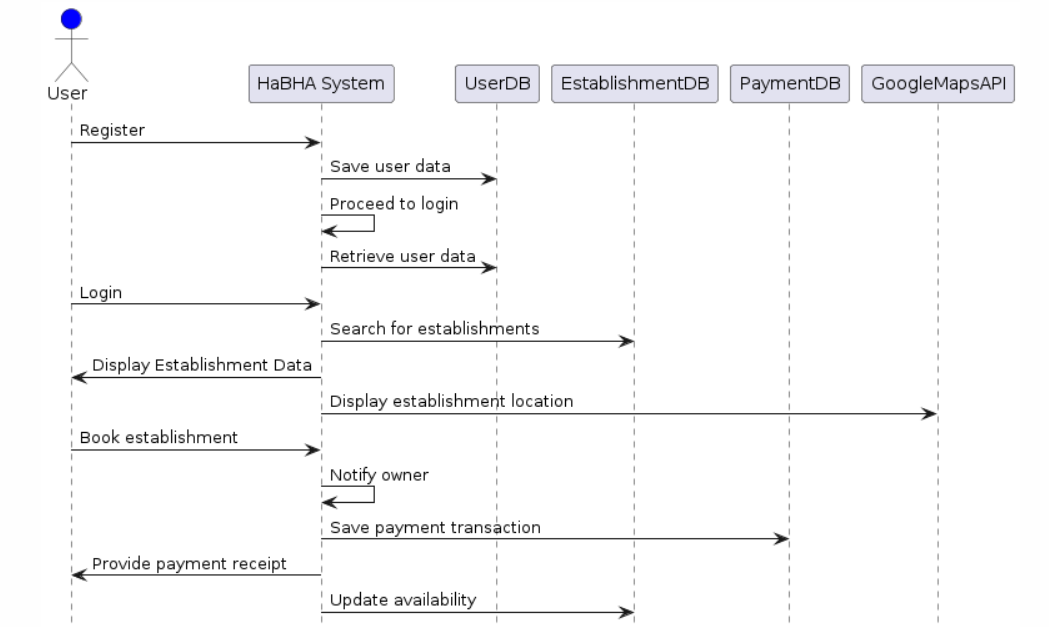


Figure 11. User’s Boarder Sequence Diagram.

# Boarding House Owner’s Sequence Diagram

Figure 12 shows the sequence diagram of the House Owner. It illustrates the step-by-step interactions between the House Owner, System, Owner Database, Establishment Database, Payment Database, and Google Maps API.

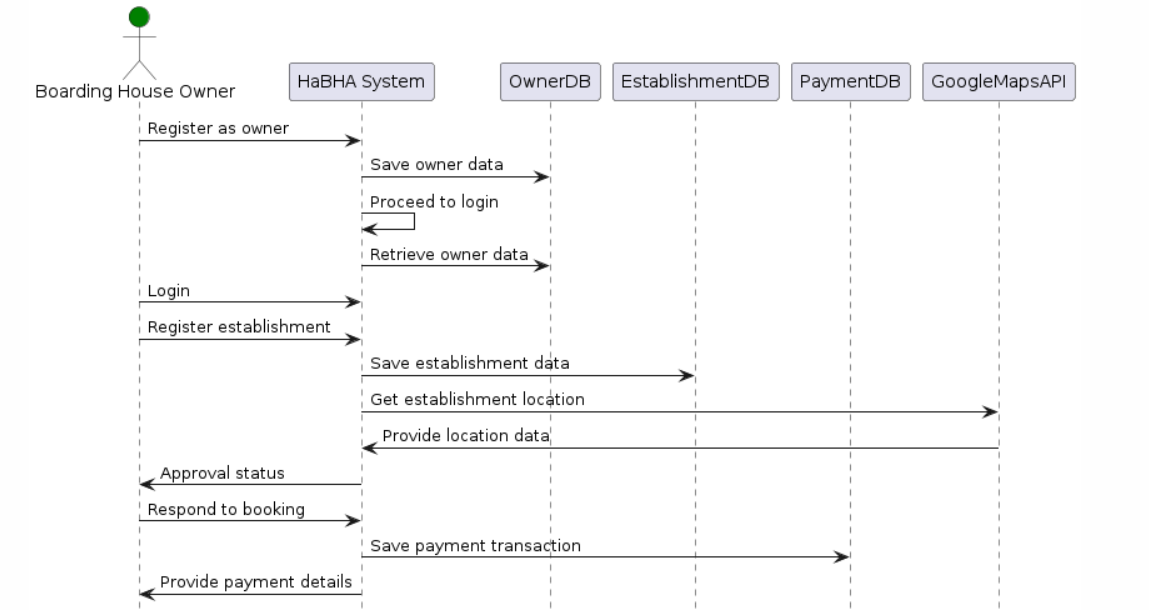


Figure 12. Boarding House Owner’s Sequence Diagram.

# Administrator’s Sequence Diagram

Figure 13 shows the sequence diagram of the Administrator. It illustrates the step-by-step interactions between the Administrator, System, Admin Database, and Establishment Database.

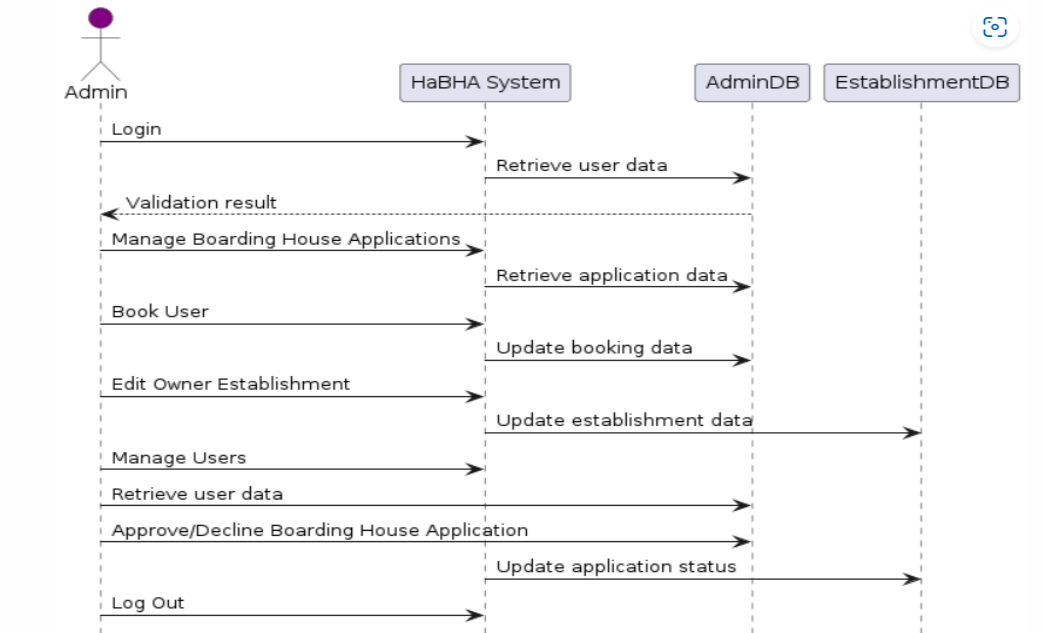


Figure 13. Administrator’s Sequence Diagram.

# Design Class Diagram

Figure 14 shows the class diagram of HaBHA. In the class diagram, the accounts table is the main foundation of the system. The Admin, Boarder and Boarding House Owner table rely on Accounts table. The Booking relies on Boarding House table and Payment relies on Booking table. The table Boarding House has a CRUD (CREATE, READ, UPDATE, DELETE operation same at Establishment Application.

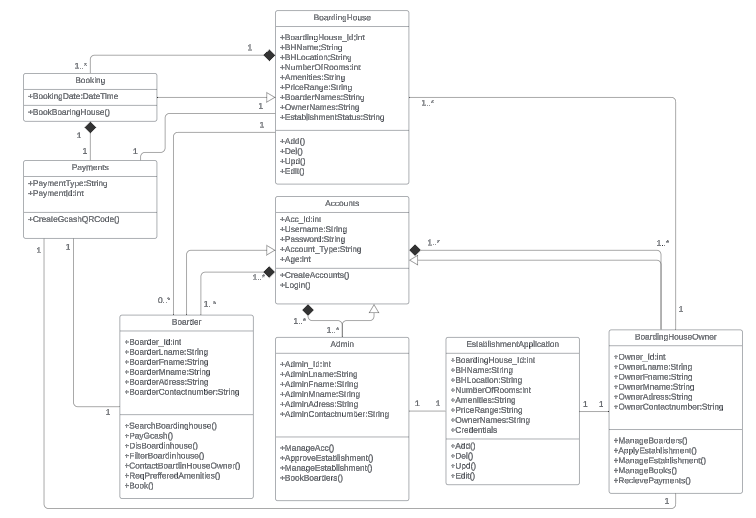


Figure 14. Design Class Diagram.

# Entity Relationship Diagram

Figure 15 shows the entity relationship of HaBHA database. It highlights the connections between different tables, such as Accounts, Admin, Boarding House, Boarding House Owner, Boarders and Payment.

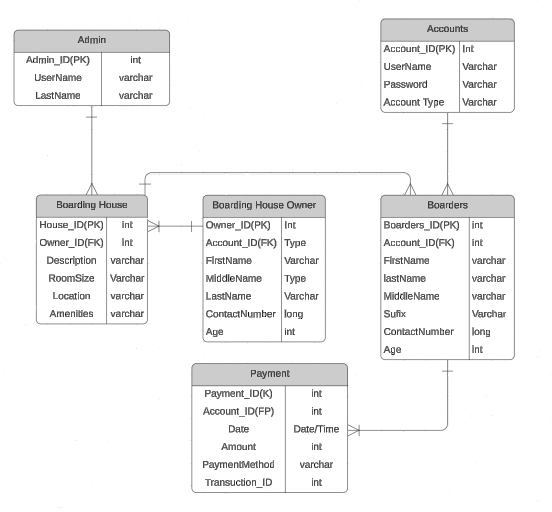


Figure 15. Entity Relationship Diagram.

# Description of Prototypes

Figure 16 shows the log in page of HaBHA where the Boarder, House Owner and the Admin can login so that they can enter the username and password that you registered.

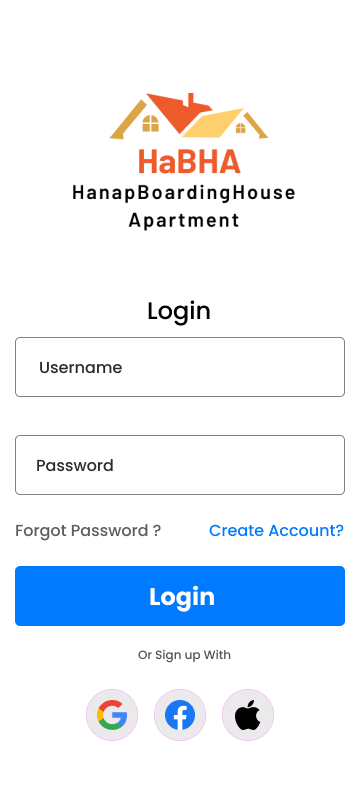


Figure 16. Login Page.

Figure 17 shows the options available to users when creating an account, including whether to be boarder or boarding house owners.



Figure 17. Creating account Page.

Figure 18 shows the registration form that you can fill up the name, Email Add, password and they have a radio button that is for Agreeing with our Terms of use and policy privacy.

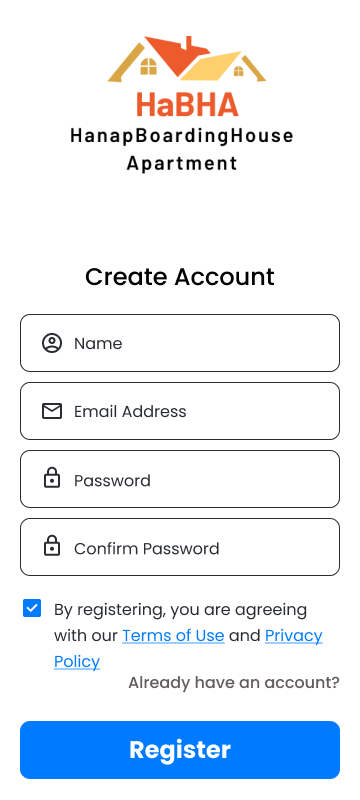


Figure 18. Registration Form Page.

Figure 19 shows the dashboard of the application that they have a search bar, and they have some room shows in the dashboard and the new listing of room.

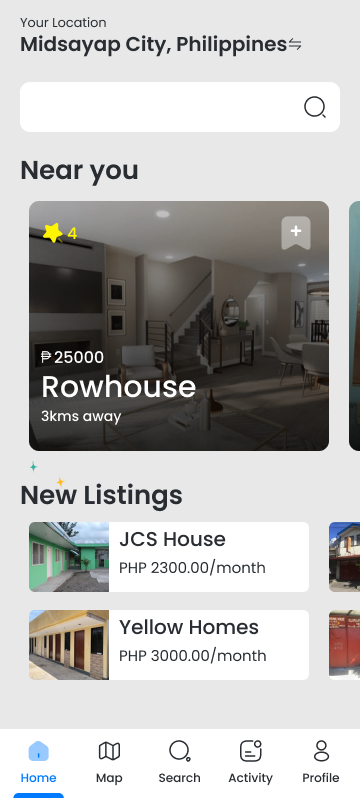


Figure 19. Dashboard Page.

Figure 20 shows the profile of the user that can be able to edit their personal information like first name, last name, mobile number, email address and address.

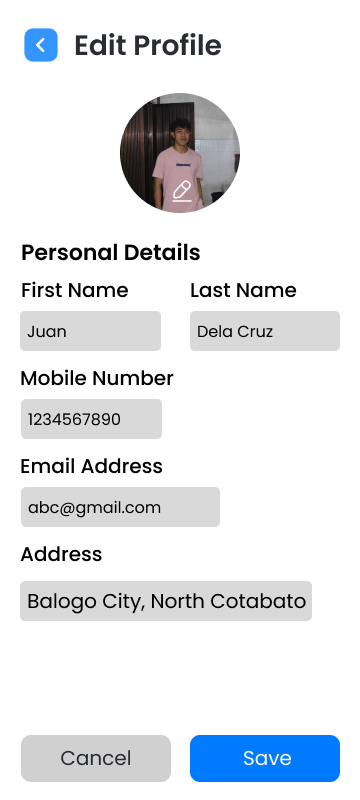


Figure 20. Edit Profile Page.

Figure 21 shows the maps of the boarding house, and they have a search bar to search for the perfect room for you, it shows the distance of the boarding house or apartment.

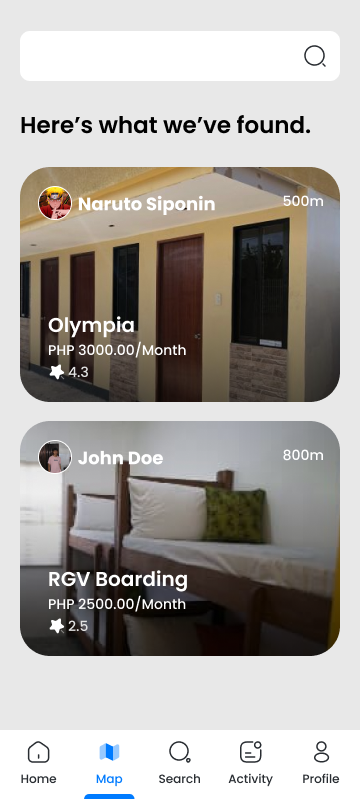


Figure 21. Location Page.

Figure 22 shows the search, in this dashboard they have a search bar, filter button, show the name of the establishment owner, price in different type of rooms and monthly fee.

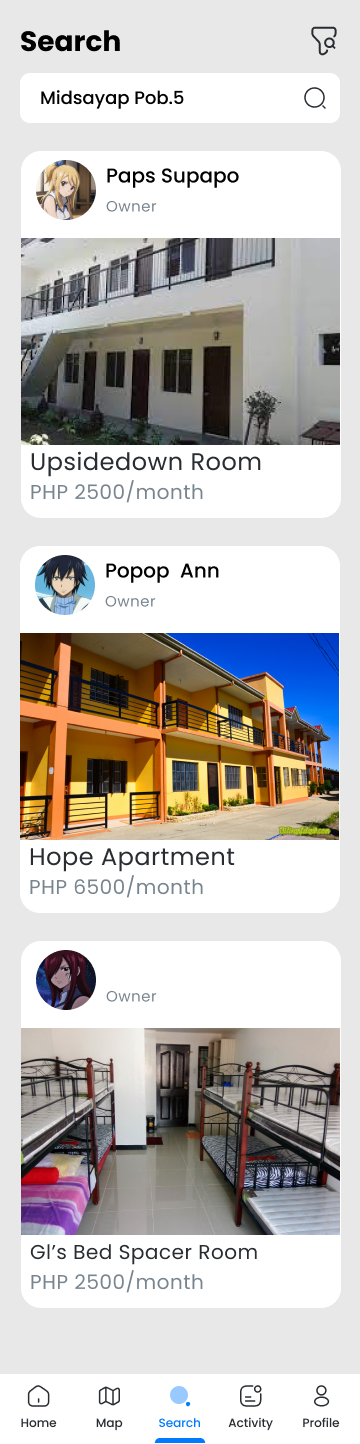


Figure 22. Search Page.

Figure 23 shows the details of the apartment. They have a button for virtual tour and name of the establishment, Reviews, they have an owner name, and you can be able to contact the owner via clicking the chat button beside the owner’s name.

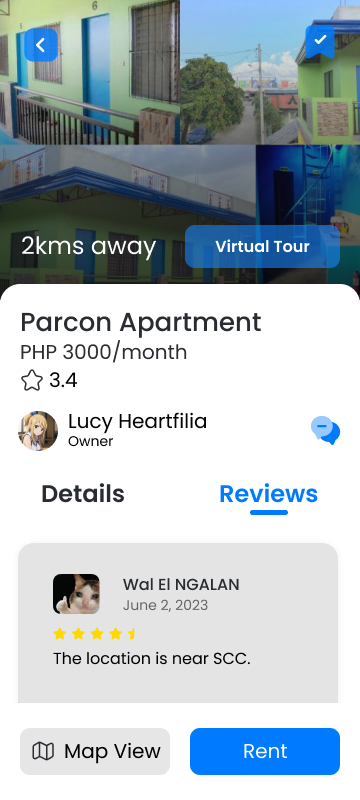


Figure 23. Apartment Details Page.

Figure 24 shows the payment method. They can see all the amount to pay, and you can see the name of the merchant, discount and the total amount.

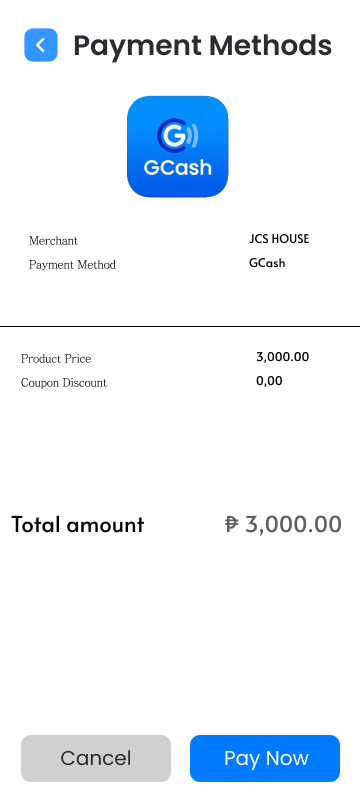


Figure 24. Payment Page.

Figure 25 shows the dashboard of the owner that can see their list of rooms, the user looks their establishment and the popular listing.

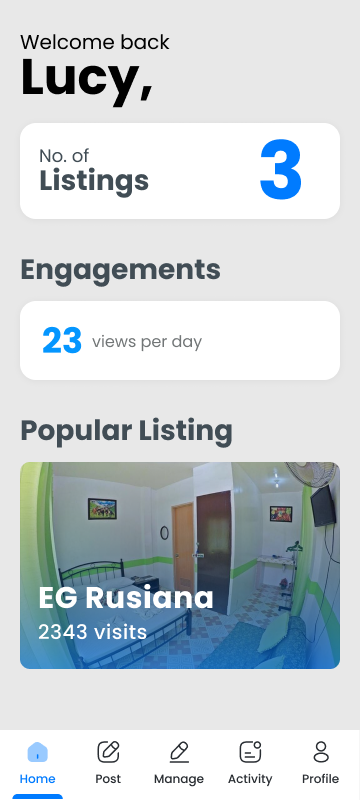


Figure 25. Owner Dashboard Page.

Figure 26 shows the Managing of the establishment they can be able to edit their own establishment, delete.

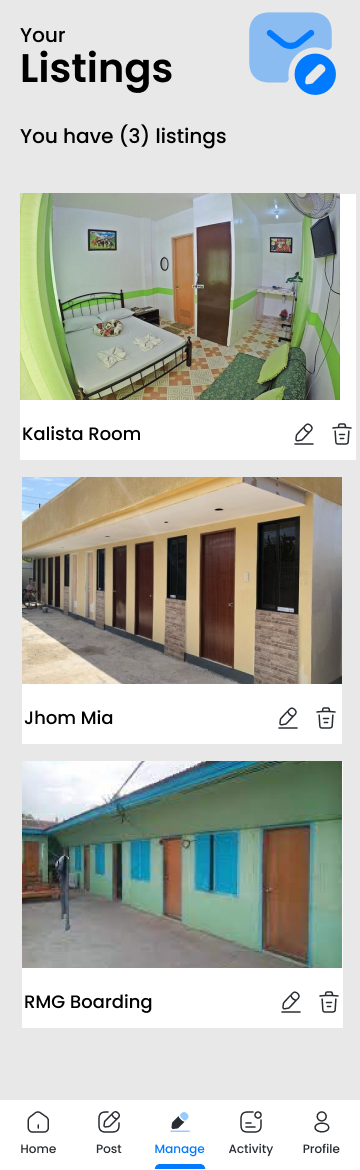


Figure 26. Manage Establishment Page.

Figure 27 shows the form of edit profile the owner can edit their personal information like first name, last name, mobile number, email add and address.

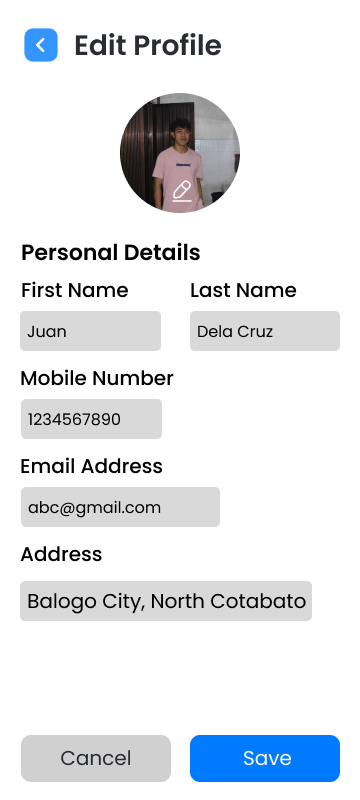


Figure 27. Edit Profile Page.

Figure 28 shows this notification that the owner notifies if there is a rental request, and they can see the viewers of their establishment.

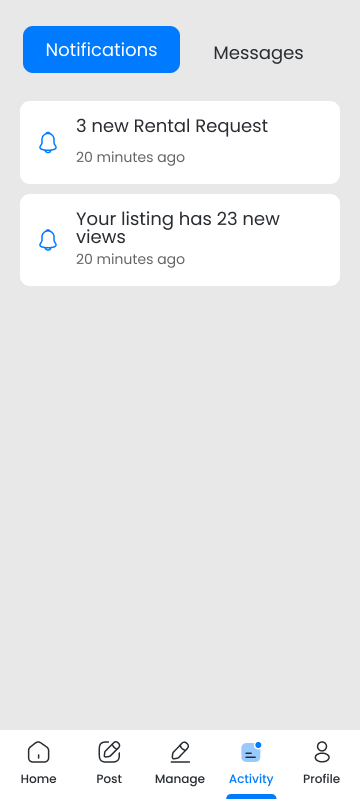


Figure 28. Notification Page.

Figure 29 shows the Approve application the admin can be able to approve or reject the owner and user.

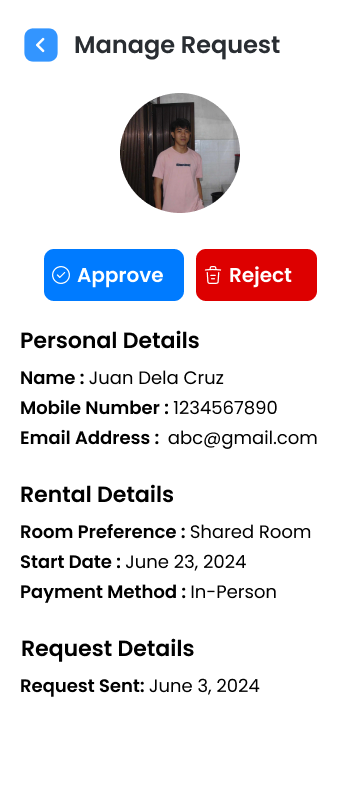


Figure 29. Manage Request Page.

Figure 30 shows a list of users and owners. The admin can edit or remove any user or owner. This provides flexibility in managing user roles.

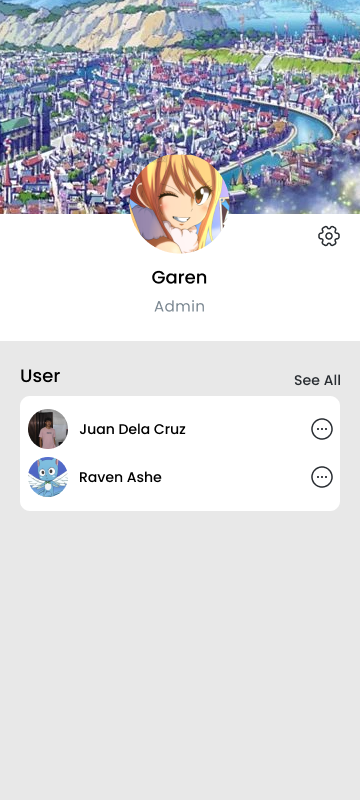


Figure 30. List Page.

Figure 31 shows the managing owner establishment the admin can be able to edit or delete the establishment of the owner or edit the establishment details.

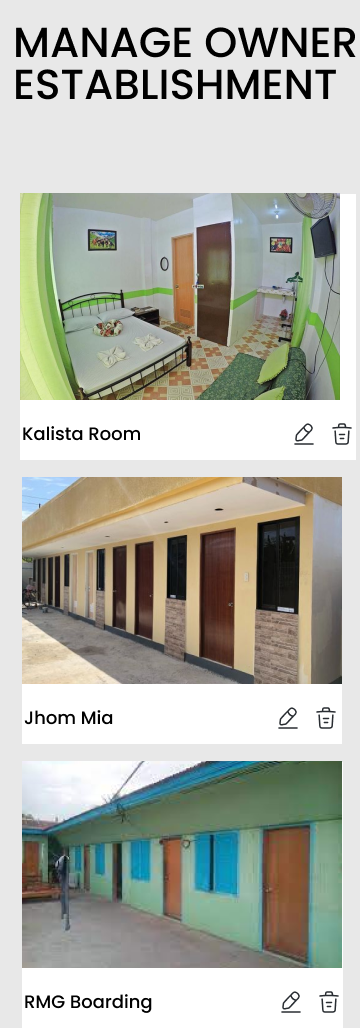


Figure 31. Manage Owner Establishment Page.

LITERATURE CITED

AlexSoft. (2021, October 29). The good and the bad of C# programming. *AltexSoft*. <https://www.altexsoft.com/blog/c-sharp-pros-and-cons/>

Arain, Q. A., Deng, Z., Memon, I., Zubedi, A., & Mangi, F. A. (2017). Map services based on multiple mix-zones with location privacy protection over road network. *Wireless Personal Communications*, *97*, 2617-2632. https://doi.org/10.1007/s11277-017-4626-0

Battin, P., & Markande, S. D. (2016, September). Location based reminder Android application using Google Maps API. In *2016 International conference on automatic control and dynamic optimization techniques (ICACDOT)* (pp. 649-652). IEEE. https://doi.org/10.1109/ICACDOT.2016.7877666

Britch, D., mhrastegari., jonpryor., Banovv., jconry. (2023). *What is .NET MAUI?* .NET MAUI. Microsoft Learn. https://learn.microsoft.com/en-us/dotnet/maui/what-is-maui?view=net-maui-8.

Consignado, M. L. L. S., Velasco, M. L. A., Sanvictores, A. P. A., Jain, A. M., & Balahadia, F. F. (2017). HAYBOL: An Android-Based Apartment Locator Application. *International Journal of Computing Sciences Research*, *1*(2), 1-9. 10.25147/ijcsr.2017.001.1.06

Cudis, C. (2019, October 3). GCash strengthens dominance in PH. *Philippine News Agency.* https://www.pna.gov.ph/articles/1082215

Daud, N. M. N., Bakar, N. A., & Rusli, H. M. (2010, June). Implementing rapid application development (RAD) methodology in developing practical training application system. In 2010 International Symposium on Information Technology (Vol. 3, pp. 1664-1667). IEEE. https://doi.org/10.1109/ITSIM.2010.5561634

Gudgeon, L., Moreno-Sanchez, P., Roos, S., McCorry, P., & Gervais, A. (2020). Sok: Layer-two blockchain protocols. In Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24 (pp. 201-226). Springer International Publishing. https://doi.org/10.1007/978-3-030-51280-4\_12

Hamzah, A., & Findawati, Y. (2021). Boarding House Information System Based on Geographic Location in Sidoarjo. *Procedia of Engineering and Life Science*, *1*(1). https://doi.org/10.21070/pels.v1i1.866

Hamzah, M. A., Mustika, N., Mustapa, M., & Rahmah, U. (2022). Application of Boarding House Rental Location Search System Based on Android. *Ceddi Journal of Information System and Technology (JST)*, *1*(2), 30-36. https://doi.org/ 10.56134/jst. v1i2.23

Hu, S., & Dai, T. (2013). Online map application development using Google Maps API, SQL database, and ASP .NET. *International Journal of Information and Communication Technology Research*, *3*(3). https://doi.org/10.1007/978-3-642-27485-5\_17

Law, S., Paige, B., & Russell, C. (2019). Take a look around. *ACM Transactions on Intelligent Systems and Technology, 10*(5), 1–19. https://doi.org/10.1145/3342240

Manalu, S. R., Wibisurya, A., Chandra, N., & Oedijanto, A. P. (2016, November). Development and evaluation of mobile application for room rental information with chat and push notification. In *2016 International Conference on Information Management and Technology (ICIMTech)* (pp.7-11). IEEE. https://doi.org/10.11 09/ICIMTech.2016.7930293

Murali, M. (2023, November 17). What is Visual Studio? All You Need to Know. https://blog.hubspot.com/website/what-is-visual studio.

Prasetyo, S. E., & Hudallah, N. (2018). Implementation of Google Maps API 3 with Haversine Algorithm in the Development of Geographic Information System Boarding House Finder. In *Proceedings of the 7th Engineering International Conference on Education, Concept and Application on Green Technology (EIC 2018)* (pp. 227-233). <http://surl.li/tjdaee>

Sasmito, G. W., Wibowo, D. S., & Dairoh, D. (2020). Implementation of rapid application development method in the development of geographic information systems of industrial centers. *Journal of information and communication convergence engineering*, *18*(3), 194-200. https://doi.org/10.6109/jicce.2020.18.3.194

Suwandi, H., Harlinda, H., & Mansyur, S. H. (2022). Implementation of a School Information System Using Rapid Application Development Method. *Jurnal Teknik Informatika (Jutif)*, *3*(6), 1501-1512. https://doi.org/10.20884/1.jutif.2022.3.6.332

Widiastuti, A., Nugroho, E. W., & Widiantoro, A. D. Y. (2021). The Information Systems of Boarding House Search Application in Soegijapranata Catholic University Semarang Based on Android. *Journal Business Technology*, *1*(1), 16. https://doi.org/10.24167/jbt.v1i1.3223

Appendix A. Statement of Work

**Statement of Work (SOW)**

**SOUTHERN CHRISTIAN COLLEGE**

**Quezon Avenue, Pob. 5**

**Midsayap, North Cotabato, 9410**

**May 2024**

HaBHA**Statement of Work**

**Table of Contents**

[Introduction/Background 51](#_Toc332473318)

[Scope of Work 51](#_Toc332473319)

[Period of Performance 51](#_Toc332473320)

[Work Requirements 52](#_Toc332473322)

[Schedule/Milestones 53](#_Toc332473323)

[Acceptance Criteria 53](#_Toc332473324)

HaBHA**Statement of Work**

**Introduction/Background**

THE HaBHA (Hanap Boarding House Apartment) is a Mobile application to accommodate mostly Boarders such as students, workers, and people with no permanent homes that are having a hard time to finding a boarding house nearby it will also benefit the landowners by displaying its facilities to the application. Most of the boarders, especially when they are new in town, don’t have any contact with any boarding house owners which makes it difficult to locate the available boarding house nearby. The developers desire to address the concerns regarding the difficulties of both Boarders and Boarding House owner such as the location of the boarding house and information about the boarding house that will surely help both boarders and boarding house owners.

**Scope of Work**

The HaBHA will be able to locate the different Boarding houses within Midsayap Area. It can filter the price, location, and amenities of each boarding house. It can Also cater online payment through G-cash and accept online booking and reservation. In this Application also automatically tallies all the expenses. The system can run on mobile devices.

**Period of Performance**

The period of the developers for HaBHA is eight (8) months, beginning in May 2024 through December 2024 tasks are scheduled to be completed within this timeframe. After the duration of the project development, the developers are no longer obliged to make changes; therefore, the client would need to pay for any upgrade or new version release of the system.

HaBHA**Statement of Work**

**Work Requirements**

As a part of the development of the HaBHA project the developers are responsible for performing tasks throughout various stages of the project. The following is a list of tasks which resulted in the successful completion of this project.

Kickoff:

* The Developers create and present detailed project plan including schedule, requirements plan, implementation plan, training plan, and transition plan.
* The developers present project plans to the boarding house owners for review and approval.

Design Phase:

* The developers gather requirements of the system through interviews and meetings.
* They will create the system and its design based on the collected data.

Build Phase:

* The developers complete all coding of approval software design.
* They provide the Boarding house owners with a detailed testing plan.
* They resolve any coding and system issues identified during usability test.

Implementation Phase:

* The developers implement the system on the boarding house owners

Training Phase:

* The developers provide training in accordance with the approved training plan provided in the project kickoff phase.

Project Handoff/Closure:

* The developers present project closure reports to the Boarding House Owners for review and approval.

HaBHA**Statement of Work**

**Schedule/Milestones**

The following schedule outlines the key milestones and deliverables for the Development of the Application:

RFP/SOW Release May 8, 2024

Developer Selection Review August 5,2024

Developer Selection August 12,2024

Period of Performance Begins September 9,2024

Software Design Review October 14, 2024

Software Implementation Review November 9,2024

Implementation Complete December 9,2024

Training Complete February 3,2025

Project Completion Review February 17,2025

Project Closure/Archives Complete March 10,2025

**Acceptance Criteria**

The acceptance of the resides with the Boarding House owners. Once the project build phase is completed, the developers will provide their report/presentation for review and approval; the Boarding House Owner either sign on the approval for the next phase to begin or reply to the developers in writing, advising what tasks must still be accomplished.

Once all project tasks are completed, the project enters the handoff/closure stage.

The developers provide their project closure report to the Boarding house owner during this stage. The acceptance of this document by the boarding house owner acknowledges acceptance of all project deliverable which signifies the successful completion of the development team’s assigned tasks.

HaBHA**Statement of Work**

**Acceptance**

Approved by:

Date:

**RHEA GARBOSA YU**

Boarding House Owner

Appendix B. Survey Questionnaire

**USE Questionnaire: Usefulness, Satisfaction, and Ease of use**

**(Kabatiran, Kasiyahan, at Kagalakan ng Paggamit)**

Pangalan (opsyonal): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**NILAYON**: Pagtaas ng Sistema ng BUSINESS WEB INVENTORY. Paki Lagyan ng tsek (✓) ang item ng iyong pinili.

**BILANG PAG-INTERPRETA**

7 Lubos na Sang-ayon

6 Sang-ayon

5 Medyo Hindi Sang-ayon/Medyo Sang-ayon

4 Walang Katiyakan/Neutral

3 Medyo Hindi Sang-ayon/Medyo Hindi Sang-ayon

2 Hindi Sang-ayon

1 Lubos na Hindi Sang-ayon

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **KAHALAGAHAN** |  |  |  |  |  |  |  |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 1. Nakakatulong sa akin na maging mas epektibo. |  |  |  |  |  |  |  |
| 2. Nakakatulong sa akin na maging mas produktibo. |  |  |  |  |  |  |  |
| 3. Napakahalaga nito. |  |  |  |  |  |  |  |
| 4. Binigyan ako ng mas maraming kontrol sa mga gawain sa buhay ko. |  |  |  |  |  |  |  |
| 5. Nagpapadali ito ng mga bagay na nais kong matapos. |  |  |  |  |  |  |  |
| 6. Nakakatipid ako ng oras kapag ginagamit ko ito. |  |  |  |  |  |  |  |
| 7. Nakakakumpleto ito ng aking mga pangangailangan. |  |  |  |  |  |  |  |
| 8. Ginagawa nito ang lahat ng inaasahan ko. |  |  |  |  |  |  |  |
| **KAGINHAWAAN PAGGAMIT** |  |  |  |  |  |  |  |
| 9. Madaling gamitin. |  |  |  |  |  |  |  |
| 10. Simple gamitin. |  |  |  |  |  |  |  |
| 11. User-friendly. |  |  |  |  |  |  |  |
| 12. Nangangailangan ito ng pinaka unang hakbang upang maisagawa ang nais kong gawin dito. |  |  |  |  |  |  |  |
| 13. Maari itong baguhin. |  |  |  |  |  |  |  |
| 14. Madali gamitin. |  |  |  |  |  |  |  |
| 15. Maari kong gamitin ito nang walang nakasulat ng mga tagubilin. |  |  |  |  |  |  |  |
| 16.Hindi ko napapansin ang mga hindi pagkakatugma habang ginagamit ko ito. |  |  |  |  |  |  |  |
| 17. Magugustuhan ito ng parehong paminsan-minsan at regular na mga gumagamit. |  |  |  |  |  |  |  |
| 18. Mabilis at madaling makakabawi mula sa mga pagkakamali. |  |  |  |  |  |  |  |
| 19. Tagumpay kong magagamit ito sa bawat pagkakataon. |  |  |  |  |  |  |  |
| **KAAYUSAN NG PAG-AARAL** |  |  |  |  |  |  |  |
| 20. Nahanap ko agad ang paraan upang gamitin ito. |  |  |  |  |  |  |  |
| 21. Madali akong naalala kung paano ito gamitin. |  |  |  |  |  |  |  |
| 22. Madali itong matutunan at gamitin. |  |  |  |  |  |  |  |
| 23. Ako ay mabilis na marunong dito. |  |  |  |  |  |  |  |
| **KASIYAHAN** |  |  |  |  |  |  |  |
| 24. Ako ay nasisiyahan dito. |  |  |  |  |  |  |  |
| 25. Ibibigay ko ito na isang rekomendasyon sa kaibigan. |  |  |  |  |  |  |  |
| 26. Mabilis itong gamitin. |  |  |  |  |  |  |  |
| 27. Ito ay gumagana sa paraang nais ko. |  |  |  |  |  |  |  |
| 28. Napakaganda nito. |  |  |  |  |  |  |  |
| 29. Nararamdaman ko na kailangan ko ito. |  |  |  |  |  |  |  |
| 30. Maginhawa gamitin. |  |  |  |  |  |  |  |

Curriculum Vitae

Krentz Jerald S. Gordon

Pigcawayan, North Cotabato

gordon.kreheu21@southernchristiancollege.edu.ph

**Personal Data**

|  |  |
| --- | --- |
| Date of Birth: | April 24, 2002 |
| Place of Birth: | Regional of Davao City |
| Age: | 22 |
| Religion: | Roman Catholic |
| Father’s Name: | Lotenio R. Gordon Jr. |
| Occupation: | Security Guard |
| Mother’s Name: | Grace S. Gordon |
| Occupation: | Housewife |

**Educational Attainment Year**

|  |  |  |
| --- | --- | --- |
| Tertiary: | Southern Christian College | 2021-2025 |
| High School | Pigcawayan National High School | 2018-2021 |
| Elementary | Pigcawayan Central Elementary School | 2008-2015 |

**Student Involvement/Work Experience:**

* N/A

**Training/Seminar Attended:**

* N/A

Curriculum Vitae

Oliver C. Supapo

Abaga, Libungan, North Cotabato

Supapo.olheu21@southernchristiancollege.edu.ph

**Personal Data**

|  |  |
| --- | --- |
| Date of Birth: | October 29, 2002 |
| Place of Birth: | Cotabato City |
| Age: | 21 |
| Religion: | Missionary Baptist |
| Father’s Name: | Raul M. Supapo |
| Occupation: | Police |
| Mother’s Name: | Gamaliel Supapo |
| Occupation: | Housewife |

**Educational Attainment Year**

|  |  |  |
| --- | --- | --- |
| Tertiary: | Southern Christian College | 2020-2025 |
| High School | St. Jude College of Science & Technology | 2014-2018 |
| Elementary | Lope T. Quial Sr. Central Elementary School | 2008-2014 |

**Student Involvement/Work Experience:**

* N/A

**Training/Seminar Attended:**

* N/A

Curriculum Vitae

Jayve James S. Jauod

Presbitero, Pigcawayan,, North Cotabato

Jauod.jaheu21@southernchristiancollege.edu.ph

**Personal Data**

|  |  |
| --- | --- |
| Date of Birth: | March 19, 2003 |
| Place of Birth: | Cotabato Regional Hospital |
| Age: | 21 |
| Religion: | Roman Catholic |
| Father’s Name: | Jimmy A. Jauod |
| Occupation: | Farmer |
| Mother’s Name: | Julieta s. Jauod |
| Occupation: | BHS |

**Educational Attainment Year**

|  |  |  |
| --- | --- | --- |
| Tertiary: | Southern Christian College | 2020-2025 |
| High School | Presbitero National High School | 2014-2018 |
| Elementary | Presbitero Central Elementary School | 2008-2014 |

**Student Involvement/Work Experience:**

* N/A

**Training/Seminar Attended:**

* **N/A**