

PROPERTY LISTING MANAGEMENT PLATFORM

A Thesis
Presented to the Faculty of
Computer Studies Department
College of Science
Technological University of the Philippines
Ayala Boulevard, Manila

By

Cujardo, Gwynn R.
Florendo, Cyrus Nathaniel P.
Jaime, Jewel Vaida A.
Quiambao, Aeron Justin C.
Salangsang, Arsenic M.

In Partial Fulfillment of the
Requirements for the Degree
Bachelor of Science in Information System

2024

APPROVAL SHEET

ACKNOWLEDGEMENT

ABSTRACT

TABLE OF CONTENTS

PRELIMINARIES	Page
Title Page	i
Approval Sheet	ii
Acknowledgement	iii
Abstract	iv
Table of Contents	v
List of Tables	vii
List of Figures	viii
List of Appendices	x
Chapter 1 THE PROBLEM AND ITS SETTING	1
Introduction	1
Background of the Study	2
Objectives of the Study	5
Significance of the Study	7
Scope and Limitations	8
Chapter 2 CONCEPTUAL FRAMEWORK	10
Review of Related Literature	10
Synthesis of Review of Related Literature	31
Review of Related Studies	33
Conceptual Model of the Study	36
Operational Definition of Terms	38
Chapter 3 METHODOLOGY	39
Project Design	39
Project Development	71
Operating and Testing Procedure	73
Evaluation Procedure of the Study	80
Chapter 4 RESULTS AND DISCUSSION	Error! Bookmark not defined.
Project Description	Error! Bookmark not defined.
Project Structure	Error! Bookmark not defined.
Project Test Results	Error! Bookmark not defined.

Project Capabilities and Limitations	Error! Bookmark not defined.
Project Evaluation	Error! Bookmark not defined.
Chapter 5 SUMMARY OF FINDINGS, CONCLUSIONS, AND RECOMMENDATIONS	Error! Bookmark not defined.
Summary of Findings	Error! Bookmark not defined.
Conclusions	Error! Bookmark not defined.
Recommendations	Error! Bookmark not defined.
REFERENCES	84
RESEARCHERS' PROFILE	100

List of Tables

Table	Page
1 Sample Test Case	74
2 Classification of Error Severity	75
3 Classification of Error Priority Levels	76
4 Overall Summary of Functionality Test Cases	77
5 Reliability Test Cases Summary	77
6 Testing Procedure for Functionality Suitability	78
7 Testing Execution Summary	80
8 Four-point Likert Scale	82
9 The Range of Mean Ratings and Equivalent Description Rating	82

List of Figures

Figures	Page
1 Conceptual Model of the Study	36
2 Entity Relationship Diagram	40
3 Use Case Diagram	42
4.A System Flowchart (Developer and Customer Side)	44
4.B System Flowchart (Broker Side)	45
5.A Module Hierarchy (Admin, Broker, Customer)	46
5.B Module Hierarchy (Developer)	47
6 Data Flow Diagram – Level 0 (Context Diagram)	48
7 Data Flow Diagram – Level 1	49
8 Login Page	50
9 Two-Factor Authentication	51
10 Admin Account Management	52
11 Dashboard - Developer	53
12 Sites	54
13 Units – Developers	55
14 Add Units	56
15 Adding Units	57
16 Broker Account Management	58
17 Add Broker Account	59
18 Broker’s Commission	60
19 Sales	61

20	Customers – Developers	62
21	Dashboard - Broker	63
22	Units – Broker	64
23	Available Units	65
24	Unit Information	66
25	Commissions – Broker	67
26	Customers – Broker	68
27	Document Status – Customer	69
28	Payment Schedule - Customer	70
29	Agile Scrum Methodology Framework	71

List of Appendices

Appendix	Page
A Software Evaluation Instrument of ISO 25010	92
B Sample Answered Evaluation Sheet and Result Sheets	93
C Summary of Respondent' Evaluation	94
D Test Cases for Functionality	95
E Test Cases for Reliability	96
F Test Incident Logs	97
G Gantt Chart	98
H User Manual	99
I Thesis Grammarian Certification	100
J Certificate of Similarity Index using TurnItIn	101

Chapter 1

THE PROBLEM AND ITS SETTING

Introduction

The Philippines, an archipelago of over 7,000 islands, is home to many residential dwellings. From single-family homes to high-rise condominiums, a wide variety of options are available to cater to the diverse needs and preferences of the population. It is a major milestone for most people to purchase a house and is often considered a symbol of stability and success. However, as condominiums redefine modern life, they have recently emerged as the new choice among urban Filipinos. In the Philippines' biggest cities, high-rise condominiums are starting to be a prominent feature of the skyline (Mioten, 2024).

As the field of information technology continues to grow, productivity tools have become essential in both personal and professional environments. According to Simplilearn (2024), productivity tools are programs to simplify tasks and streamline workflow. These tools transform the way people work by enabling employees, teams, and individuals to collaborate remotely, communicate efficiently, and manage tasks in real time without needing to engage in person. Productivity tools have features that are helpful in streamlining processes and increasing overall productivity.

Managing clients, tracking commissions, and keeping up with available units is a crucial challenge for brokers, as the people of today prefer to buy units such as condominiums to not only keep up with the trends, but also because of the many benefits that comes with owning one. In addition, it can also prove to be a challenging task for developers to track the available units, brokers, and clients if no proper system is put in place. However, with

the emergence of advanced technology, web applications and systems that facilitate productivity tools make these tasks simpler. However, with the emergence of advanced technology, web applications and systems that facilitate productivity tools make these tasks simpler. Additionally, listing units, whether sold or available, improves communication and streamlines tasks like tracking milestones, units, and deals between brokers and developers, leading to a smoother and more efficient process.

Fortunately, the current study can help both brokers and developers track their clients, units, and commissions. By utilizing the agile scrum methodology and a quantitative research approach, the researchers aim to achieve the desired outcomes of the study.

In conclusion, as more people move to urban areas in the Philippines, the demand for housing is evolving, creating a growing need for innovative solutions in the real estate market. This research project introduces a system that harnesses the power productivity tool to make the developers' and brokers' tasks simpler. By offering a user-friendly approach, the system helps streamline the tracking of clients, deals, and commissions, benefiting both developers and brokers.

Background of the Study

Real estate is a tangible asset made up of the property and the land on which it sits. As people began to recognize the value of real estate, there were those who saw opportunities in land and housing. Many nurtured and supported this process, ultimately shaping the industry into what it is today. Additionally, individuals can earn income from their real estate assets by turning them into businesses, such as renting through platforms like Airbnb

or leasing them as apartments. Though it is immovable, real estate, like other assets, is also subject to supply and demand. This means that property prices and rent are heavily affected by these forces. When demand increases, prices rise; when demand decreases, prices fall.

The Philippines offers a wide range of living arrangements, from single-family homes to modern high-rise condominiums, catering to diverse housing needs. In recent years, however, condominiums have surged in popularity and become the preferred choice for residential property among Filipinos. The condominium market in the Philippines is rising and is being driven by rapid urbanization. As more people migrate to urban areas such as Manila in search of employment and better opportunities, the need for space-efficient, convenient living arrangements has grown. In addition, modern condominiums are designed with the needs of people, offering a self-contained lifestyle without the need to leave the general premises. Lastly, the market is willing to adapt to changes in consumer demands and market conditions (Mioten, 2024).

With the increase in the condominium market, the presence of real estate agents has become more prominent. Whether in public spaces or on social media, it is now usual to see agents actively promoting condominium projects to potential buyers. As the demands are rising, developers and investors are capitalizing on the growing condominium market, building new projects to supply the increasing demand (Mioten, 2024). These projects often emphasize prime locations, modern amenities, and flexible payment terms to attract a wider range of potential buyers.

Mentioned are the many factors that work together to form a thriving condominium market in the Philippines. From the rise of urban migration and shifting lifestyle trends to

the active efforts of real estate agents and developers, these factors have played a key role in contributing to the steady growth and transformation of the condominium sector. In property listing, many websites and platforms have been created to handle properties, yet there are no consistent and well-known brands. There is currently no well-known productivity tool specifically designed for developers and brokers, highlighting a crucial gap in the market. A dedicated tool could streamline the process of tracking and managing clients, facilitating better communication between agents and buyers. This would provide a consistent way to stay updated on matters such as buyer payments and other relevant concerns. By leveraging technology and data analytics, these platforms can provide agents or brokers with valuable insights helping them stay and be more motivated and productive. Moreover, while technology has undoubtedly made the buying process easier in many ways, it can also introduce new complexities and challenges. Combining the concepts of listing and productivity tools can help create a platform that promotes efficiency and encourages productivity.

In conclusion, the condominium market in the Philippines presents significant opportunities for innovation and growth. With the increasing urbanization and economic development in the country, demand for efficient and high-quality properties are on the rise, presenting an opportunity for services to capitalize and develop an integrated system that focuses on the agents' productivity, from viewing milestones, tracking commissions and units, to managing clients. By providing sustainable and accessible solutions, and fostering innovation and collaborative relationships, the proper platform can unlock and

enhance the potential of the developers and their respective brokers, contributing to a better experience for everyone involved.

Objectives of the Study

General Objective

The study's main objective is to develop a productivity tool through a website application catering to both developers and brokers. The platform allows developers to list units, generate payment schedules and commissions, while simultaneously allowing brokers to view their milestones and manage their clients.

Specific Objectives

The following are the specific objectives of the study:

1. Create and design the system with the following characteristics:
 - a. A platform for developers to list their available units for sale, generate commission rates, set payment schedules, and create broker accounts.
 - b. Tool for brokers to track clients, units, commissions, view their milestones, and submit customer information including documentary requirements.
 - c. A feature for buyers to access their payment schedules and documentary requirements.

- d. The system will use automated calculations to generate reports for the brokers' milestones.
2. Create the website application using the following software development tools and scripting language:
 - a. Front-end tools
 - HTML
 - JS
 - CSS
 - Vue
 - b. Back-end tools
 - Python
 - c. Framework
 - Django
 - d. Database Management System
 - PostgreSQL
 - e. Version Control
 - GitHub
 - f. IDE
 - Visual Studio Code
 - g. User-Interface Tools
 - Figma

3. Test and improve the website application based on functional suitability and reliability
4. Evaluate the acceptability of the website application by gathering multiple users to operate the system. The tool that will be used to evaluate the application is ISO25010, which will evaluate the application's acceptability in terms of sustainability, timeliness, accuracy, and maintainability.

Significance of the Study

With urban expansion, changing lifestyles, and attractive investment prospects, the condominium market in the Philippines has been rising steadily, especially in major cities.

For developers, the web-based system will offer a platform to list and oversee available and unavailable units efficiently. For brokers, the system will be a productivity tool to view their milestones, manage clients, and track units together with possible deals. For buyers, the system will provide a practical tool to easily identify their payment schedule and required documents.

Finally, this capstone project will be valuable for future researchers. It can serve as a reference to enhance future studies with similar objectives, and future researchers can build upon the data and features introduced in this study, thereby solidifying, and expanding the work initiated by the current researchers.

Scope and Limitations

This study centers on the development of a productivity tool website dedicated to brokers and developers. It endeavors to provide developers with a robust platform to list their units, enabling them to effectively connect with brokers. Through a user-friendly interface, developers will have the flexibility to create, update, customize, and remove their listings within the system, ensuring accurate representation and efficient management of their units.

In addition to serving developers, this study strives to address the critical needs of brokers who work to connect clients with the right condominium units. By facilitating easier access to available units, this initiative aims to enhance convenience and streamline the brokers' tasks, simplifying the sales process and fostering a more seamless transition into their new living arrangements or commercial spaces.

This study's overarching goal is to foster mutual benefits for both developers and brokers alike. By leveraging property listings, commission tracking, and payment schedules, brokers can pinpoint units that align closely with their clients' preferences, budgetary constraints, and lifestyle requirements. Simultaneously, brokers stand to gain increased visibility for their transactions and successes, helping them attract more clients and generate more sales.

The property listing management platform is meticulously crafted using a diverse array of innovative web development technologies and frameworks, including but not limited to Python, HTML, JS, CSS, Vue.js, Django, PostgreSQL, GitHub, Visual Studio Code, and Figma. By harnessing the power of these tools, the platform offers unparalleled scalability,

reliability, and performance, ensuring a seamless user experience across various devices and browsers.

Moreover, the system's web-based architecture eliminates the need for cumbersome downloads or installations, providing users with instant access to its full suite of features and functionalities directly through popular web browsers such as Chrome, Safari, Firefox, Opera, and Edge. By implementing stringent security measures and encryption protocols, the platform safeguards sensitive user data and ensures uncompromising privacy and data integrity.

Developers can use their accounts to display available units to brokers, as well as generate payment schedules and commissions. They can also manage brokers' accounts by creating or updating their information, providing hands-on access to organizational details. Additionally, brokers can use their accounts to submit customer information and track ongoing or completed transactions.

Furthermore, the platform adopts a seamless user experience design, allowing users to seamlessly transition between distinct roles and functionalities with minimal friction. Whether brokers are viewing listings and client details or developers are updating unit availability, the platform offers intuitive navigation and robust backend support, empowering users to achieve their respective objectives with ease and efficiency.

Chapter 2

CONCEPTUAL FRAMEWORK

This chapter presents the related literature, studies, and systems, a synthesis of the whole literature review, the conceptual model of the study, and the operational definition of terms relevant to Property Listing Management Platform.

Review of Related Literature

This part contains literature related to the Property Listing Management Platform.

Condominium

A condominium, or condo, is a privately owned residential unit within a larger complex or building that contains multiple similar units (Chen, 2024). Similarly, McMillin (2022) defines a condo as an independently owned unit within a larger community, with shared ownership of communal spaces and facilities such as pools, garages, and gyms. According to Faelnar (2024), in the Philippines, there are different types of condominiums to cater to the varying needs of the residential market. The different types of condominiums include:

- High-rise condominiums. It is located in urban areas with easy access to public transport, shopping, and dining, and offers stunning views, especially from higher floors.
- Mid-rise condominiums. It is located near the city but not in its center and typically has 4-6 stories.

- Low-rise condominiums. Provide more outdoor space, often including garages, and offer greater privacy, similar to townhouses.
- Garden-type condominiums. It is detached, surrounded by landscaped areas or pools, and offers the most privacy and exclusive outdoor space.

Additionally, Arranz (2021) mentioned that condo units vary in size and design, each offering unique characteristics and style. The types of condo units include:

- Studio. The layout of this unit is an open-plan apartment, the smallest and most affordable type.
- Bedroom Unit. This unit has partitions to separate the bedrooms from the different areas of the unit.
- Loft. This unit offers high ceilings and an open design, with a living area on the lower level and a bedroom above.
- Bi-level Type. This unit has a second floor, with common areas on the lower level and bedrooms upstairs.
- Penthouse Type. This unit is located at the top of a building and is the largest and most exclusive unit.

The current study aims to collaborate with developers to list units, allowing their respective brokers to easily search and access comprehensive details of the available units. In addition, the current study has considered and understood the distinct types of condominium units. The current will apply this understanding in filtering the search, which will assist brokers in finding the appropriate unit that will match their potential buyer's preferences.

Payment Schedule

A payment schedule is a contract between buyer and seller outlining the timing and method of payment for a transaction. It assists businesses in monitoring the amounts their customers owe them as well as the debts they owe to their creditors. Payment schedules are applicable in a variety of situations, such as loans, mortgages, and credit agreements. (What is a payment schedule?, 2024). Supported by Kintu (2018), a payment schedule is an agreement between two parties regarding the amounts and payment dates. In addition, there are different types of payment schedules, and this includes:

- Lump sum payment. A one-time payment is made at the end of a transaction.
- Installment payment. A payment is divided into several parts and made over a specified timeframe.
- Deferred payment. A payment that is postponed to a later date than initially planned.
- Fixed payment. A payment that occurs at regular intervals, such as every two weeks or monthly.
- Variable payment. A payment that fluctuates based on factors like sales or performance metrics.
- Milestone payment. A payment triggered by the completion of specific predetermined goals.
- Revolving payment. A recurring payment is made regularly, similar to a subscription service.

The current study has considered and understood the distinct types of payment schedules. The current study will apply this understanding in the computation or quotation of payment schedules, which will assist both broker and potential buyer.

Broker – Agent

In an article written by Segal (2024), he differentiated real estate agents and real estate brokers. Both of them are licensed professionals who assist clients in buying or selling properties and earn a commission once the deal is finalized. The difference is that brokers have additional training and licensing requirements. They can hire an agent to work for them and get a cut of the agent's commissions.

The current study aims to collaborate with the developer's respective brokers or agents to manage clients, track deals, and view their milestones, allowing them to work more efficiently and productively.

Commission

According to CFI Team (2024), commission is the payment given to an employee upon completing a task, typically related to selling a specific quantity of products or services. The methods companies use to structure and distribute commissions can differ. Two approaches are:

- Flat commission. Where the employee receives a fixed rate or percentage for every sale.

- Ramped commission. Where the percentage increases as the employee achieves higher sales or meets set targets.

Folger (2024) stated that real estate agents and brokers typically earn their income through commissions, which are often based on a percentage of the property's selling price. Commissions are paid directly to the brokers, who then distribute the agent's share. This payment is deducted from the sale proceeds before the seller receives their portion.

The current study has understood the concept of commission. The current study will apply this understanding in the computation of the agent's or broker's commission, which will be seen in their milestone in the system.

Web-based System

In an article written by (Senthil, 2024), in the past, web-based applications had limited functionality, but improvements in technology, security, and internet speed have significantly expanded the possibilities for web-based systems. A web-based system is an application that you access using HTTP, typically describing applications that operate within a web browser. A web-based system is an application that you access using HTTP. Typically, it is used to describe applications that work in a web browser. Web-based systems are accessible anywhere and have lower development costs, increased efficiency, and reduced hardware costs.

The current study proposed a property listing management platform that is web-based because of its accessibility, that can be accessed if there is an internet connection and browser.

Data Analytics

Based on an article by The Investopedia Team (2024), data analytics is the science of examining raw data to draw conclusions. Shao et al. (2022) added that data analytics involves estimation, statistics, organization, user experience, and computing technologies, encompassing various theoretical and statistical methods and trends. According to Ghasemaghaei (2019), data analytics use refers to how many companies use technologies designed to efficiently get useful information from substantial amounts of diverse types of data. Moreover, data analytics tools help share and improve knowledge. Finally, using advanced data analytics tools is key to better decision-making.

The current study will employ data analytics in a web-based system to assist respective brokers and agents by providing valuable insights about their successful deals.

Figma

Figma is a web-based tool for interface design that allows team-based collaborative design projects. It also has many features that will help in the design phase of the project (Bracey, 2024). According to Scott (2024), just like Adobe XD and Sketch, Figma provides similar features, but its advantage over its competitors is the ability for real-time collaboration. In addition, an article by Giglio (2024), mentioned that Figma combined easy vector editing, the convenience of a web-based platform, and the ability to collaborate in real time resulted in establishing itself as a popular design tool.

The current study will employ Figma in the user interface design phase to create and test interactive prototypes. This approach aims to streamline the design process, enhance

collaboration among team members, and ensure that the final product meets user expectations. By leveraging Figma's robust set of tools and its integration capabilities with collaboration platforms, the study seeks to demonstrate the effectiveness of this platform in producing high-quality, user-centered designs while facilitating seamless communication and cooperation within the design team.

Visual Studio Code

Visual Studio Code, often abbreviated as VS Code, is a free, lightweight, yet powerful source code editor available for Windows, macOS, Linux, and Raspberry Pi OS, running on both desktop and web platforms (Heller, 2022). According to Chris (2023), VS Code is like Sublime Text and Atom, offering a rich text editing experience like a miniature version of Visual Studio. VS Code provides developers with a customizable environment through various plugins, making it versatile and adaptable to different programming needs (Your Ultimate Guide To Visual Studio vs Visual Studio Code, n.d.). Lastly, Heller (2022) mentioned that VS Code has built-in support for JavaScript, TypeScript, and Node.js, along with a vast ecosystem of extensions for other languages, runtimes, environments, and clouds, catering to a wide range of development tasks.

In the current study, Visual Studio Code, or VS Code, will serve as the IDE (Integrated Development Environment) due to its compatibility with various programming languages required for the development of the web-based system. Additionally, its rich ecosystem of extensions can aid in debugging tasks. Lastly, the researchers' familiarity with VS Code further supports its selection as the preferred IDE.

GitHub

GitHub, an online interface, facilitates real-time collaboration among users (Coursera, 2023). According to Lutkevich and Courtemanche (2023), it serves as a web-based version control and collaboration platform primarily targeted at software developers. The platform's expansive features redefine the collaborative landscape for software development projects, extending beyond mere code storage to encompass version control, issue tracking, and code review functionalities, all crucial components in contemporary software development processes (Webb, 2024).

GitHub claims that it is utilized by over four million organizations and more than one hundred million developers (GitHub, n.d.). Webb (2024) notes that GitHub's popularity among software developers stems from its comprehensive feature set, which supports every aspect of the development process, fostering collaboration without boundaries. Additionally, its user-friendly interface simplifies code management for users of all skill levels. Furthermore, GitHub integrates seamlessly with many development tools and automates tasks through GitHub Actions, enhancing efficiency, productivity, and code security with automated fixes.

Given GitHub's robust collaboration features and widespread adoption, the current study will primarily utilize it to enhance collaboration among researchers. Leveraging its real-time collaborative capabilities, version control system, and issue tracking functionalities, the study aims to streamline project management processes and facilitate seamless collaboration on code development and review tasks.

Python

Python, as Lukaszewski (2019) notes, is a versatile programming language suitable for any modern computer operating system. Similarly, Coursera (2024) describes Python as a general-purpose language that is not specialized for any specific problems. In addition, Miller (2023) stated that Python's versatility enables it to be used in a wide range of applications across various industries. According to a study by Statista, Python ranks as the third most popular programming language worldwide among developers.

In web development, Python plays a significant role in back-end development, handling server-side tasks like interacting with databases and APIs, as outlined by Miller (2023). Coursera (2024) supports this by highlighting Python's back-end development capabilities, including data processing, database communication, URL routing, and security measures.

The current study will employ Python due to its simplicity, extensive libraries, security capabilities, and strong community support. Python also easily integrates with various databases, web services, and other programming languages, making it flexible and powerful in terms of developing a robust, scalable, and maintainable web application. Python ensures that both the development and operational aspects of the system are efficient and effective.

Django

Django, a Python-based web framework, earns the moniker 'batteries included web framework' for its abundance of built-in features, enabling the rapid development of efficient web applications (GeekforGeeks, 2024). It encompasses everything from the

Django Admin Interface to default databases like SQLite3. Moreover, an article on Django Introduction (2024) emphasizes its high-level nature, facilitating the creation of secure and easy-to-manage websites.

Additionally, Noble Desktop (2024) stated that Django is currently among the top ten (10) frameworks used by web developers as it offers many attractive attributes, including:

- Robust - it automatically comes with a wide range of extra features capable of handling common web development tasks.
- Fast - it simplifies the creation of web applications from conception to completion to reduce the amount of time and expertise required for development.
- Simple - it benefits from being built on Python with a reputation for being simple and accessible
- Secure - it includes many default features that help protect your application and its users, making it one of the most secure frameworks around
- Scalable - it comes equipped with various components that can be easily unplugged and replaced for effortless scalability
- Versatile - it can be customized to fit applications of any type, size, or scope
- Well-Supported - it is well supported both by the official governing body and the programmers who regularly utilize the framework

Django, the framework that the current study will employ, provides many benefits in terms of creating the rental property listing and management web application. Django's built-in features and modular design makes it a proper choice for this system. Django also

has full support for PostgreSQL, ensuring that the web application's queries are done in a clean and powerful manner.

Hyper Text Markup Language (HTML)

HTML, or Hyper Text Markup Language, serves as the foundational language for constructing web pages (Astari S., 2023). It enables the creation and organization of webpage elements such as sections, paragraphs, and links using tags and attributes. Lutkevich (2020) highlights HTML's role as a text-based method for defining content structure within HTML files, guiding web browsers in rendering text, images, and multimedia elements on webpages.

Additionally, Lutkevich (2020) underscores HTML's widespread adoption, accessibility across all browsers, ease of learning, clean source code, open-source nature, and compatibility with backend programming languages. Meanwhile, Adetunji (2023) emphasizes HTML's significance as a standard markup language for developing websites, applications, and highlights that it lacks aesthetics and functionality.

The current study will use HTML, justified by its fundamental role in web development, ability to structure content effectively, widespread adoption, ease of learning, open-source nature, compatibility with backend languages, and status as the standard markup language for web development.

Cascading Style Sheets (CSS)

CSS, short for Cascading Style Sheets, is a language developed in the 1990s to style web documents, which has become crucial for web developers and plays a vital role in enhancing the user experience online, working alongside various markup languages (BasuMallick, 2022). According to (Domantas G., 2023), it is used to style elements written in markup languages like HTML, which forms the foundation of websites, while CSS focuses on the visual aesthetics of the entire site. Unlike programming languages such as C++ or JavaScript, CSS is specifically designed for styling web pages and is not considered a programming language (Eygi, 2019).

The current study will use CSS to enhance user experience, separate content from presentation, and provide specific styling capabilities. By using CSS, it will enable the creation of visually appealing, user-friendly, and consistently styled web pages that are easier to maintain and perform efficiently across different platforms and devices.

JavaScript

JavaScript, a dynamic programming language, offers various functionalities such as performing math calculations, dynamically modifying HTML content in the DOM, generating dynamic style declarations, fetching content from other websites, and more (Megida, 2021). According to Jordana (2024), JavaScript, recognized as a scripting language, enhances web pages by adding interactive elements like dropdown menus and animated graphics, thus improving user engagement. In addition, GeekforGeeks (2024) further emphasizes its lightweight, cross-platform nature, known for its versatility in web

development and its application beyond web browsers. JavaScript's dynamic nature enables its utilization not only in web development but also in web applications and game development, facilitating the implementation of dynamic features not achievable with HTML and CSS alone. Lastly, developers often pair JavaScript with HTML and CSS to enhance websites with interactivity, and they may leverage third-party libraries to incorporate advanced features into their projects without coding them from scratch (Jordana, 2024).

The current study will use JavaScript because of its ability to provide dynamic functionality, enhance user engagement, offer versatility and cross-platform capabilities, enable the implementation of dynamic features, complement HTML and CSS, and allow the use of third-party libraries for advanced features. Additionally, JavaScript is essential for creating responsive and interactive user interfaces and is widely supported by modern web browsers. Its extensive ecosystem, including frameworks and tools, further extends its applicability in both front-end and back-end development. JavaScript's large developer community and extensive documentation make it easier to find resources and support, facilitating a smoother development process.

Vue.js

Vue.js offers several advantages, including its speed, simplicity, and strong support from both libraries and its community (Kugell, 2022). According to Simplilearn (2022), its features encompass a range of functionalities such as virtual DOM, data binding, components, event handling, transitions, computed properties, templates, directives, and

routing. In addition, combining Angular-influenced approaches with streamlined features, Vue.js focuses on front-end interfacing and application development. Its core library emphasizes the view layer and is designed for incremental adoption into projects (Corbo, 2022). These features make Vue.js a popular choice for building user interfaces and single-page applications, offering developers the flexibility and efficiency they need for modern web development projects.

As Vue.js is a versatile and efficient framework that simplifies the development of modern web applications, the current study will employ this technology. Vue.js provides developers with a comprehensive set of tools and features that streamline the development process and enable the creation of highly interactive and responsive user interfaces. Additionally, Vue.js has a rich ecosystem that will provide the researchers with access to a wealth of resources, enabling them to extend its functionality, solve common development challenges, and accelerate the development process.

PostgreSQL

PostgreSQL is an open-source object-relational database management system (ORDBMS) that offers scalability, high performance, and robust features for analytics and data management. It is widely utilized across various applications, businesses, and organizations (Scott, 2024). In addition, Hilbert (2023) mentioned that using PostgreSQL has many advantages, including customization, extensibility, scalability, robustness and reliability, and community support. According to Doerrfeld (2024), PostgreSQL is a hugely

popular object-relational database. The latest State of PostgreSQL report shows that it's used more today in their organizations than a year ago.

PostgreSQL, as mentioned, is an object-relational database management system and is the database that the current will employ. As noted by Scott (2024), PostgreSQL is used by the majority of e-commerce systems to produce, process, and manage product catalogs, customer information, orders, and transactions. With its scalability and dependability, e-commerce platforms can accommodate the large volume of transactions and dynamic inventory changes that improve consumer experience and operational efficiency.

ISO 25010

ISO 25010, known as "Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models," is a standard that focuses on defining and assessing quality requirements and models for both systems and software (Britton, 2021). As outlined by Obrenović (2021), this standard serves as a set of guidelines and suggestions for assessing the quality of software products. It forms a component of the ISO/IEC 25000 series, which comprises various international standards in the realm of software engineering.

According to ISO 25000, ISO/IEC 25010 comprises nine quality characteristics, and these are:

Functional Suitability pertains to the extent to which a product or system can deliver that satisfies both stated and implied requirements. This characteristic is composed of the following sub-characteristics:

- **Functional Completeness** pertains to the range of functions encompassing all specified tasks and user objectives.
- **Functional Correctness** pertains to the accuracy and precision with which a product or system delivers the required results.
- **Functional Appropriateness** pertains to how effectively functions fulfill designated tasks and objectives.

Performance Efficiency pertains to how effectively resources are utilized in relation to performance. This characteristic is composed of the following sub-characteristics:

- **Time Behavior** pertains to the speed of response, processing times, and throughput rates of a product or system during its operation.
- **Resource Utilization** pertains to the quantity and types of resources consumed by a product or system during its operation.
- **Capacity** pertains to the maximum limits of a parameter within a product or system.

Compatibility pertains to the ability of a product, system, or component to effectively exchange information and fulfill its intended functions within a shared hardware or software environment. This characteristic is composed of the following sub-characteristics:

- **Co-existence** pertains to the ability of a product to efficiently fulfill its functions within a shared environment alongside other products, without causing adverse effects on them.
- **Interoperability** pertains to the proficiency of two or more systems, products, or components to communicate and utilize exchanged information effectively.

Interaction Capability pertains to the extent to which a product or system allows designated users to exchange information through its user interface to accomplish specific tasks across various usage scenarios. This characteristic is composed of the following sub-characteristics:

- **Appropriateness recognizability** pertains to the extent to which users can determine if a product or system is suitable for their requirements.
- **Learnability** pertains to the extent to which users can learn the functions of a product or system to use it within a specified period.
- **Operability** pertains to the extent to which a product or system possesses attributes that make it simple to operate and manage.
- **User error protection** pertains to how much a system safeguards users against operational errors.
- **User engagement** pertains to the extent to which a user interface presents functions and information in an inviting manner, encouraging ongoing interaction.
- **Inclusivity** pertains to the extent to which a product or system is accessible to individuals from diverse backgrounds.

- **User assistance** pertains to the extent to which a product accommodates users with a wide range of characteristics and capabilities to achieve specific goals in a defined context of use
- **Self-descriptiveness** pertains to the extent to which a product provides necessary information to users, making its capabilities and usage immediately understandable without excessive reliance on additional resources,

Reliability pertains to the ability of a system, product, or component to execute designated functions under predefined circumstances. This characteristic is composed of the following sub-characteristics:

- **Faultlessness** pertains to the extent to which a system, product, or component performs designated functions without errors during regular operation
- **Availability** pertains to the extent to which a system, product, or component is operational and available for use when needed.
- **Fault tolerance** pertains to the extent to which a system, product, or component continues to function as intended despite the presence of hardware or software faults.
- **Recoverability** pertains to the extent to which a product or system can recover affected data and restore the desired system state in the event of an interruption or failure.

Security pertains to the level at which a product or system safeguards information and data against potential security threats or vulnerabilities. This characteristic is composed of the following sub-characteristics:

- **Confidentiality** pertains to the extent to which a product or system ensures that data is accessible only to authorized individuals.
- **Integrity** pertains to the extent to which a system, product, or component ensures that its state and data are safeguarded from unauthorized modification or deletion, whether due to malicious actions or computer errors.
- **Non-repudiation** pertains to the extent to which actions or events can be verified to have occurred, preventing denial of these events or actions afterward.
- **Accountability** pertains to the extent to which the actions of an entity can be uniquely linked back to that entity.
- **Authenticity** pertains to the extent to which the identity of a subject or resource can be verified to be as claimed.
- **Resistance** pertains to the extent to which a product or system can continue operating despite being targeted by malicious actors.

Maintainability pertains to the degree of ease with which a product or system can be altered for enhancements, corrections, or adjustments to suit evolving environmental conditions and needs. This characteristic is composed of the following sub-characteristics:

- **Modularity** pertains to the extent to which a system or computer program is constructed from separate components, allowing changes to one component with minimal impact on others.
- **Reusability** pertains to the extent to which a product can serve as an asset in multiple systems or in the construction of other assets.
- **Analysability** pertains to the effectiveness and efficiency of assessing the impact of changes on a product or system, diagnosing deficiencies or failures, or identifying parts for modification.
- **Modifiability** pertains to the ease and efficiency with which a product or system can be modified without introducing defects or diminishing existing quality.
- **Testability** pertains to the effectiveness and efficiency of establishing test criteria for a system, product, or component, and conducting tests to verify if those criteria are met.

Flexibility pertains to the extent to which a product or system can adjust to alterations in its requirements, usage contexts, or system environment. This characteristic is composed of the following sub-characteristics:

- **Adaptability** pertains to the extent to which a product or system can be efficiently adjusted for or moved to various hardware, software, or operational environments.
- **Scalability** pertains to the extent to which a product can manage increasing or decreasing workloads or adjust its capacity to handle fluctuations effectively.
- **Installability** pertains to the effectiveness and efficiency with which a product or system can be installed and/or uninstalled in a specific environment.

- **Replaceability** pertains to the extent to which a product can substitute another specified software product for the same purpose within the same environment.

Safety pertains to the extent to which a product or system can prevent endangerment to human life, health, property, or the environment under specified conditions. This characteristic is composed of the following sub-characteristics:

- **Operational Constraint** pertains to the extent to which a product or system limits its operation within safe parameters or states when faced with operational hazards.
- **Risk Identification** pertains to the extent to which a product can recognize events or operations that may pose unacceptable risks to life, property, or the environment.
- **Fail Safe** pertains to the extent to which a product can automatically switch to a safe operating mode or return to a safe condition in case of failure.
- **Hazard Warning** pertains to the extent to which a product or system provides alerts about unacceptable risks to operations or internal controls, enabling timely reactions to maintain safe operations.
- **Safe Integration** pertains to the extent to which a product can ensure safety during and after integration with other components.

The current study will employ ISO 25010 in testing and evaluating the developed system. It is an ideal choice as it provides a comprehensive framework that encompasses not only functional aspects but also non-functional characteristics. ISO 25010 offers a structured approach to assess these critical aspects, ensuring that the system meets basic

functional requirements and performs optimally and reliably under various conditions, enhancing user satisfaction and system effectiveness.

Synthesis of Review of Related Literature

Based on the information gathered by the researchers, condominiums are one of the options for living arrangements that cater to the diverse housing needs of people today. A condominium is a privately owned residential unit in a building with shared ownership of communal spaces and facilities. Additionally, condominiums have different types, ranging from garden-type condominiums to high-rise condominiums. Lastly, condominium units also encompass several types.

Additionally, real estate brokers and agents are needed to sell and buy condominiums. Depending on the company, sometimes, the agents or brokers are paid through commissions when they successfully sell a unit, and it also has different times. The researchers also gathered information about payment schedules. They learned that there are also several types of payment schedules, and it depends on the client or the developer what kind of payment they can offer and do. Recognizing this diversity in these topics, the researchers aim to develop a web-based listing management platform and productivity tool tailored for developers, brokers, and agents. By leveraging web-based technology, the system will prioritize accessibility and ease of use for developers and brokers, enhancing the overall experience.

Moreover, the system will incorporate data analytics to offer valuable insights for viewing milestones. This integration aims to improve managing clients, tracking units, provide actionable insights for brokers, and streamline property searches for prospective clients, making the process more efficient and user-friendly.

The web-based system will be developed using a suite of integrated tools to ensure efficiency and effectiveness throughout its creation and operation. User interface design will be facilitated by Figma, a powerful UX/UI design tool, which will streamline collaboration and the design process by enabling the creation of interactive prototypes, ensuring user-centered designs, and seamless communication within the design team. Version control will be managed through GitHub, serving as a collaborative platform for efficient project management, facilitating version control and issue tracking to streamline collaboration among researchers and developers, and ensuring an organized and transparent development process. Visual Studio Code will serve as the Integrated Development Environment (IDE), offering versatility and compatibility with various programming languages, supported by an extensive ecosystem of extensions for efficient development and debugging tasks. Database management will be handled by PostgreSQL, an open-source ORDBMS, which will efficiently manage structured data with scalability and security, ensuring that the system's data is well-structured and manageable. Django, a Python-based framework, will simplify web application development, providing built-in features and scalability to create a robust and secure listing and management system. Both backend and frontend development will be supported by HTML, JavaScript, CSS, and Vue.js, with HTML forming the foundation of web pages, JavaScript adding interactivity

and dynamic functionality, CSS enhancing user experience through styling capabilities, and Vue.js simplifying frontend development by enabling the creation of highly interactive and responsive user interfaces. Python will power the system's backend, known for its simplicity and versatility, ensuring efficient development and operation through its compatibility with databases and web services. Lastly, after the system is developed, it will undergo testing and evaluation using ISO 25010 as an assessment tool.

Review of Related Studies

This part contains related studies and works that have already been conducted on the topic of rental property listing systems and management systems.

A capstone project by Setty (2022) entitled "Management System for an Apartment" successfully aimed to create a standard web-based online platform that will benefit both renters and homeowners. The author developed a system wherein the homeowners can list their properties that are for rent, and the renters will be able to see the listed properties.

A similar work was conducted by Monteverde et al. (2023) entitled "A Web-Based Rental House Smart Finder using Rapid Application Development Basis for Evaluation of ISO 20501". The authors developed a system that provides necessary information on specific rental properties and aids prospective tenants in finding affordable variations of rental houses. Its goals are to simplify the process of looking for a rental home, lessen the time consumed in finding rental homes, and quickly access the information about it.

In addition, several other works have developed systems with similar features aimed at improving the rental process for both renters and homeowners. These include works by Voumick et al. (2021), Paul (2022), Rathore et al. (2021), and Rastogi et al. (2023). Each of these studies focused on creating platforms that list rental properties, provide essential information, simplify the search process, reduce the time required to find rental homes, and offer quick access to detailed property information.

Moreover, some studies have integrated an additional payment feature into their system. For example, Ikuomola and Asefon (2022) and Misyam and Selamat (2021) developed a system that not only lists rental properties and provides essential information but also integrates an online payment system. This feature allows renters to make rental payments directly through the platform, thereby streamlining the rental process even further and providing a comprehensive solution for both property listings and financial transactions.

Focusing on a different aspect of the rental market, Hamzah et al. (2022) addressed the specific needs of boarding house tenants, particularly students. According to their study, there is a significant lack of information about the availability of boarding houses, which makes it challenging for prospective tenants to find suitable accommodations. To address this issue, the authors developed an Android-based Boarding House Rental Location Search System Application. This application helps tenants easily find the location and details of boarding houses, while also enabling boarding house owners to provide updated information about their properties.

While the studies have primarily focused on listing and searching for rental properties, there are also works that delve into systems that will help brokers and agents in their work, offering solutions beyond mere property discovery.

In a study entitled “The Real Estate Transaction Trace System Model Based on Ethereum Blockchain Platform” VO and Nguyen (2022) developed a Real Estate Transaction Trace designed using blockchain network architecture. The RETT system can manage and track the entire process of real estate transactions involving all participants.

Similarly, Pankratov et al. (2020) studied blockchain technology and real estate transaction and their management. This study revealed that the use of blockchain technology in real estate transactions makes it faster, more secure, transparent, and cost-effective, as it will minimize human errors.

Lastly, Song et al. (2022) devised a model-agnostic, milestone-based task tracker emphasizing tracking task progress. Their tracker helps users identify the subtasks needed to complete the overall task and monitor progress.

These studies collectively highlight the evolution of residential management systems and technology, introducing innovative features to streamline property searching and real estate transactions.

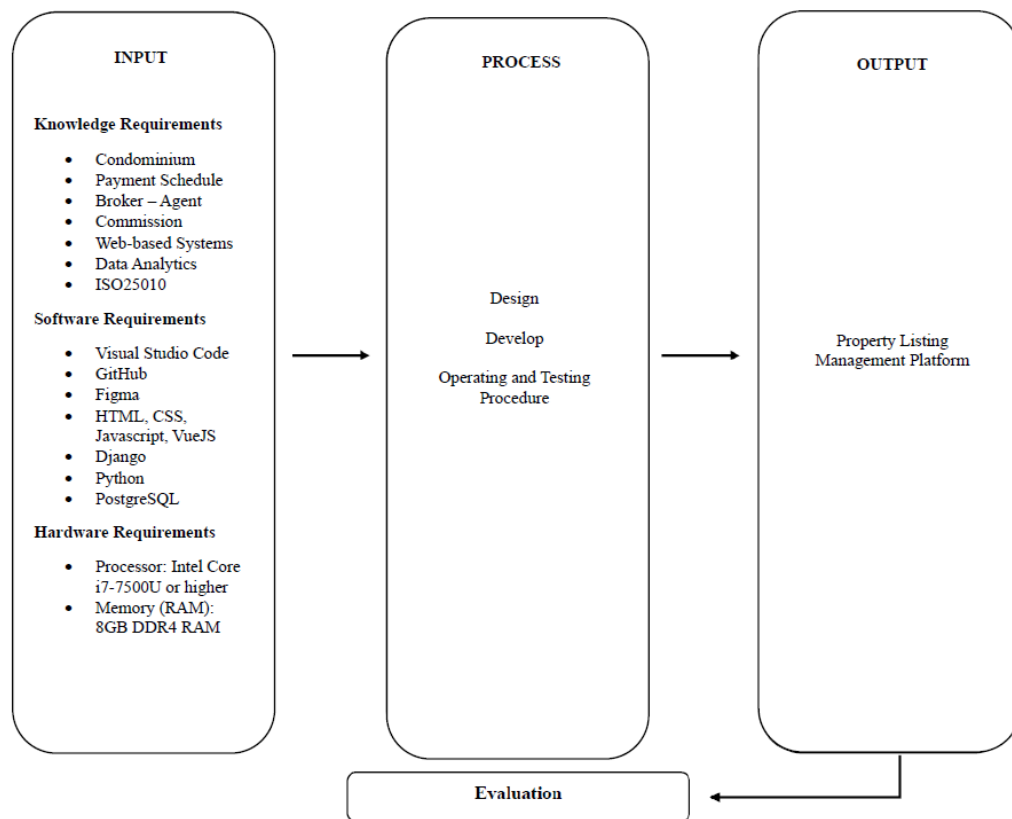
The current study will develop a platform that caters to property listing and productivity tool for the agents and brokers. By integrating features from previous research, it will offer a comprehensive solution for developers, brokers, and agents alike. The property listing management platform aims to aid brokers and agents. Leveraging insights from the authors,

the current study aspires to address existing gaps in the market and provide an all-encompassing, user-friendly platform for the real estate ecosystem.

Conceptual Model of the Study

Figure 1

Conceptual Model of the Study



The figure above shows the study's conceptual model, which depicts the overview of the flow and the whole concept of the study.

Input

The content of the input phase is the three (3) major requirements of the system. First are the knowledge requirements under this requirement, the knowledge the researchers should understand to establish a foundation for the study. It consists of condominium, payment schedule, broker-agent, commission, web-based system, data analytics, Google Maps API, and ISO25010. Second, the software requirements needed to develop the system. It consists of Visual Studio Code, GitHub, Figma, HTML, CSS, JavaScript, Vue.js, Django, Python, and PostgreSQL. Lastly, the hardware requirements to access the system. It consists of a Processor: Intel Core i7-7500U or higher, and Memory (RAM): 8GB DDR4 RAM.

Process

The content of the process phase outlines how the system will be developed. This phase involves tasks such as designing, developing, operating, and testing procedures.

Design: In this phase, the researchers should create an entity relationship diagram, use case diagram, system flowchart, module hierarchy, data flow diagram, and architecture diagram. These diagrams help visualize and summarize the specifics of the system.

Develop: In this phase, the researchers will develop the property listing management platform using front-end tools, back-end tools, frameworks, a database management system, version control, and an IDE.

Operating and Testing Procedures: After development, the system will undergo testing to assess its functionality.

Output

The developed system, Property Listing Management Platform.

Evaluation

The developed system will be evaluated using ISO 25010, an assessment tool to examine the system's applicability, efficiency, compatibility, usability, dependability, security, maintainability, and portability.

Operational Definition of Terms

For better understanding of the study, the researchers gave several terms that were defined operationally used in the study.

Agent refers to a person who works for the developers to sell units. May be a broker or a person that is under brokers.

Broker refers to a person who works for developers to sell units.

Developer refers to the property developer of condominium projects.

Property Listing Management Platform refers to the listing and productivity tool for developers and respective brokers.

Chapter 3

METHODOLOGY

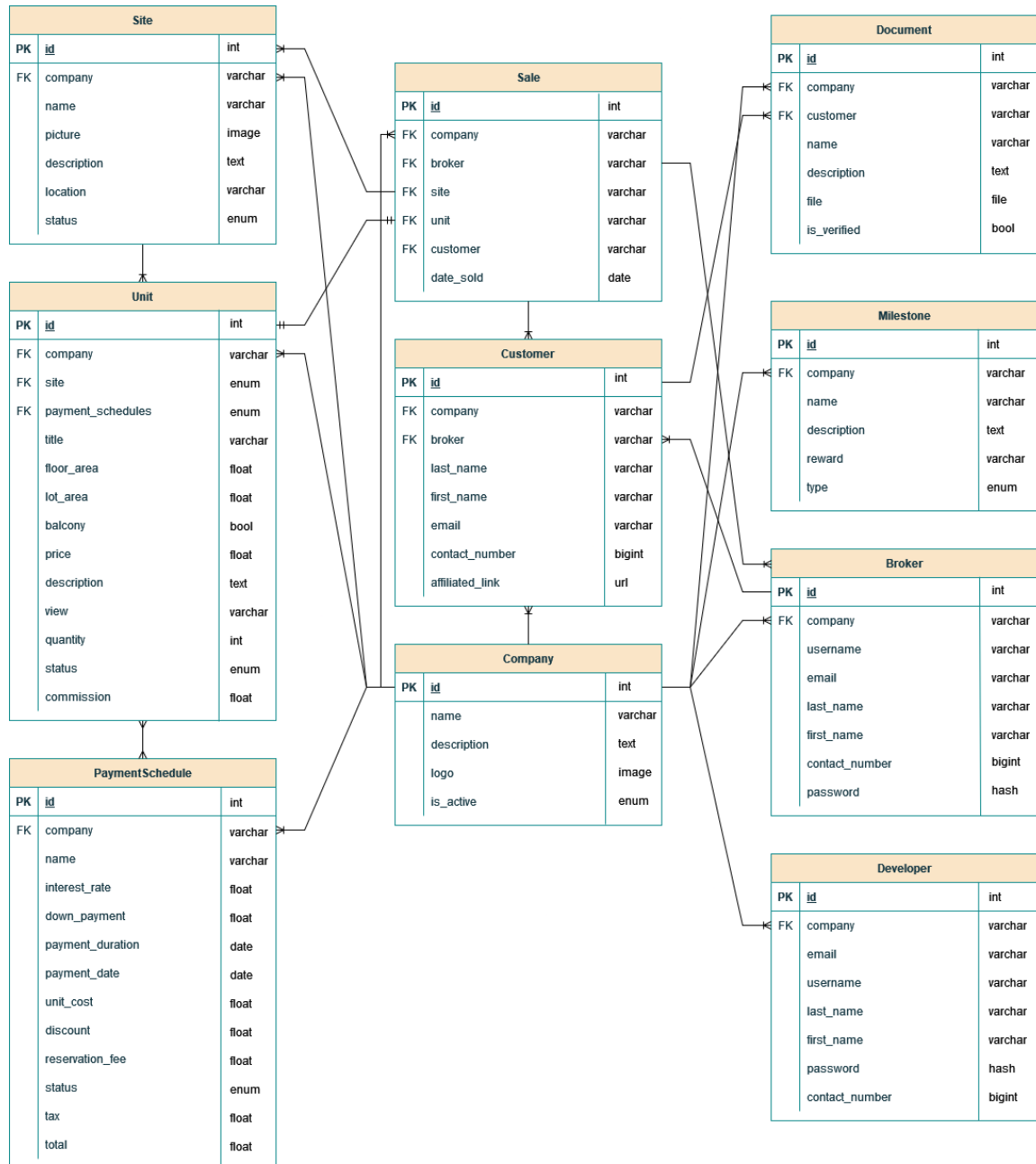
This chapter contains the research methodology of the study with the following sections: project design, project development, operation and testing procedure, and evaluation procedure.

Project Design

The study will develop a web-based system that will serve as a rental property listing and management system. The web-based system is designed to assist tenants in finding listed properties for rent and help property owners list their properties and manage them. The project design of the study is explained below using the Entity Relationship Diagram, Use Case Diagram, System Flowchart, Module Hierarchy, Data Flow Diagram, and Wireframe.

Entity Relationship Diagram

The entity-relationship diagram (ERD) is a visual representation of the data model for a system. It depicts the system's entities, the attributes of those entities, and the relationships between them. In the context of a property listing and management system, an ERD would illustrate entities such as Users, Listings, Properties, Tenants, Payments, and Notifications, showing how they interact and are related to each other. By using an ERD, people of interest can understand the data structure, identify potential issues, and ensure that the database is designed efficiently to support the system's functionality.

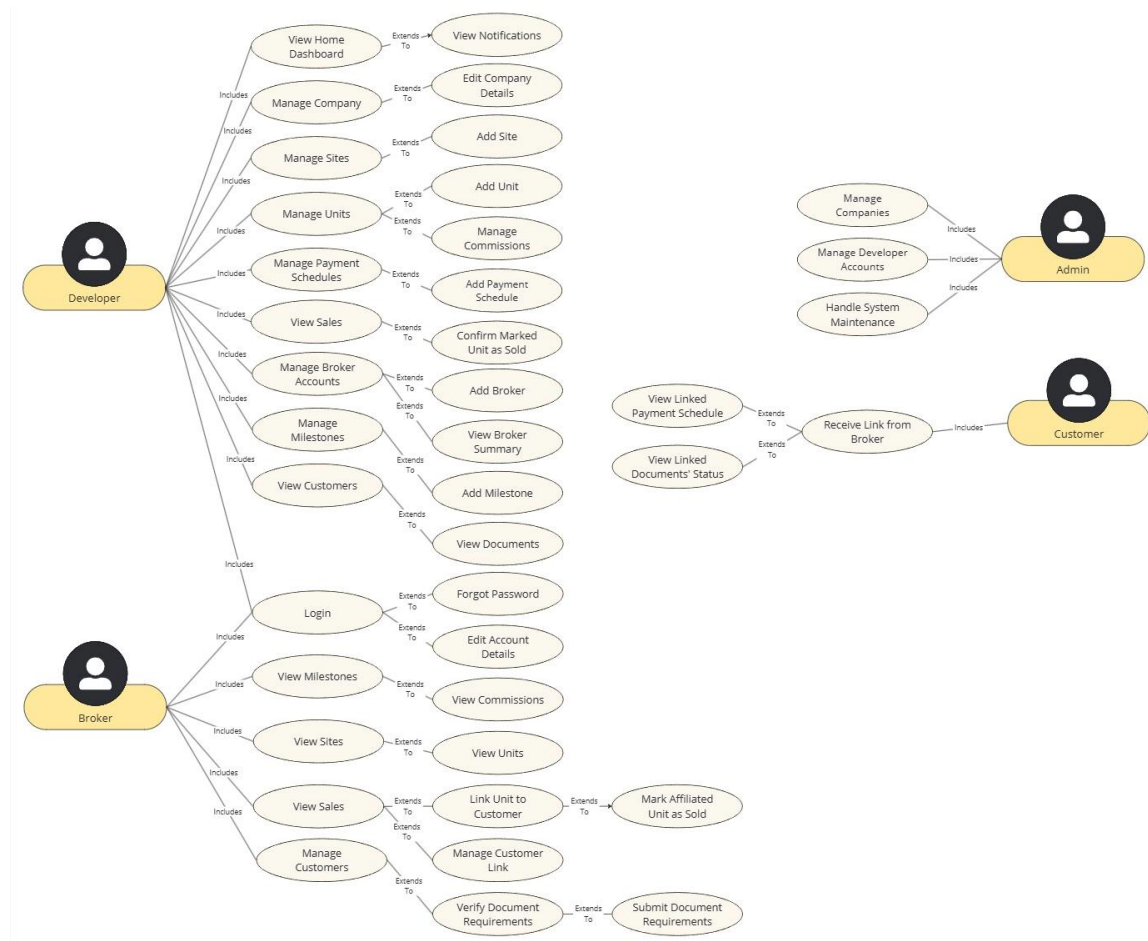
Figure 2*Entity Relationship Diagram*

As the system revolves around the management of property sales and customer interactions, the main entities include companies, sites, units, brokers, developers, customers, and sales, along with related documentation, milestones, and payment schedules. The system allows developers to manage companies and associated sites, each with its affiliated units. Units have detailed attributes such as floor area, balcony availability, price, and commission rate. Brokers and developers handle customer relationships, including tracking customer information, managing affiliated links, and overseeing the sales process. Sales transactions are recorded with a one-to-many relationship between customers and units, as each customer may be linked to multiple unit purchases over time.

Supporting entities, such as documents, milestones, and payment schedules, provide essential functionality to ensure comprehensive record-keeping and workflow tracking. Through these interconnected entities, the system facilitates the seamless operation of property sales, customer management, and administrative oversight for real estate projects.

Use Case Diagram

The use case diagram is a visual representation of the functional requirements of a system, illustrating the interactions between users and the system itself. It displays the various ways a user can interact with the system to achieve specific goals. The purpose of a use case diagram is to provide a clear and organized depiction of the system's functional scope and the several ways users can utilize its features.

Figure 3*Use Case Diagram*

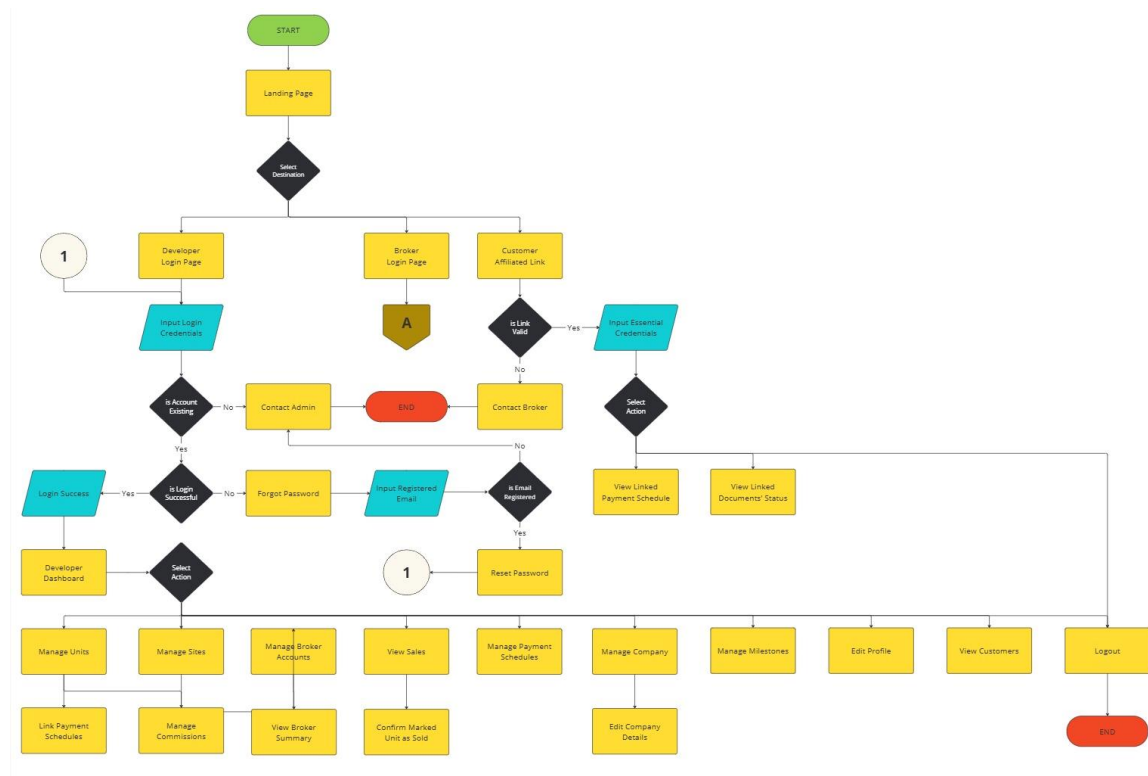
Each use case represents a distinct functionality or service provided by the system, highlighting the relationship between the actors and the use cases. In the context of a property listing management system, different functionalities can be done depending on the section in the system where the user is present.

The users have two main use cases: the developer and broker use cases. The broker side of the system allows for viewing milestones, sites and sales, and managing customers.

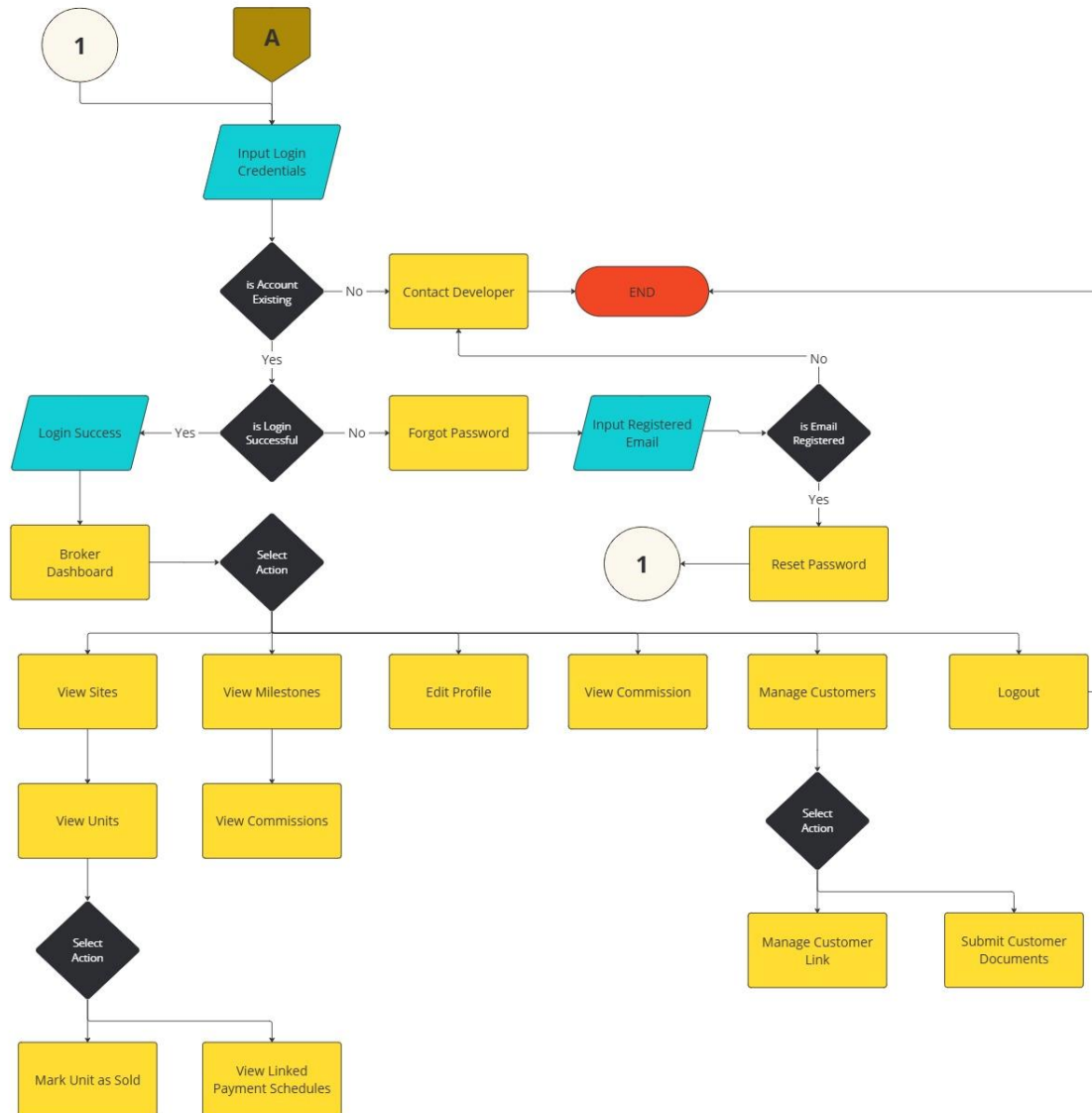
On the other hand, the developer side of the system has many functionalities, from managing company, sites, units, payment schedules, broker accounts, milestones, and document requirements. The options of viewing customer information and sales are also provided to allow for better transparency, allowing to have up-to-date and consistent information about the customer and sold units.

System Flowchart

A system flowchart is a diagram that depicts the flow of data through a system and how different processes interact with each other. It is a visual representation that outlines the system's steps, decision points, and the flow of control within. System flowcharts are used to understand, analyze, and communicate the structure and behavior of a system, making it easier to identify potential issues and optimize processes.

Figure 4.A*System Flowchart (Developer and Customer Side)*

Shown in Figure 4.A is the flow of the system from the landing page to the developer and customer side of functionalities, ranging from viewing linked payment schedules and documents' status, viewing customers' information, and managing units, broker accounts, payment schedules, company, and sites. Shown also is a connector, connector A, which takes an action from Figure 4.B.

Figure 4.B*System Flowchart (Broker Side)*

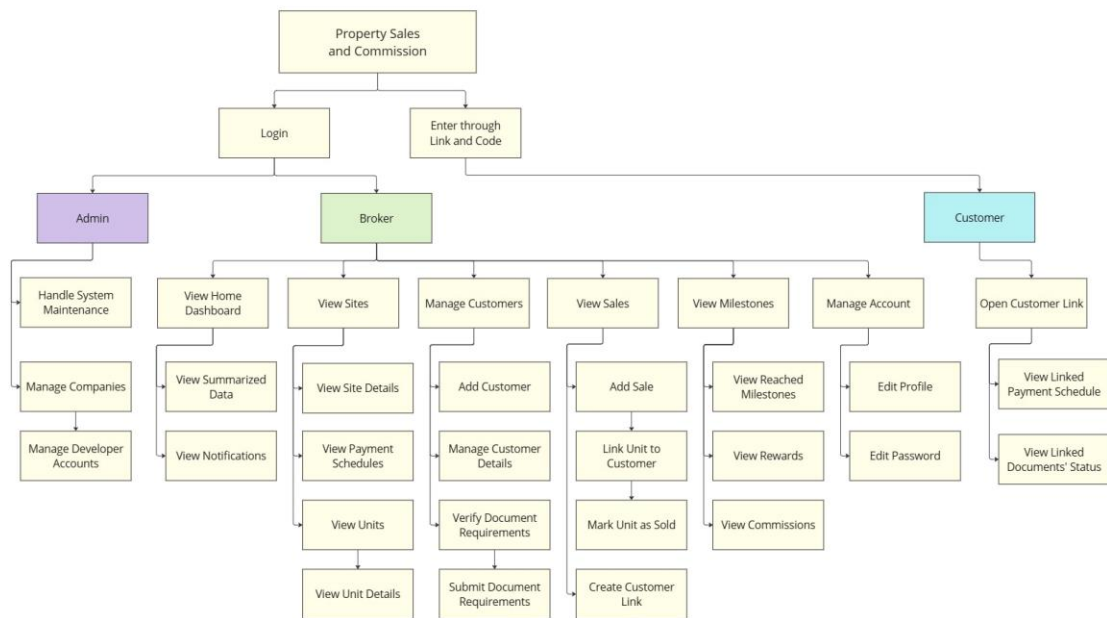
Shown in Figure 4.B is the flow of the system from logging in of the broker to its respective side of functionalities. Specific objectives, such as viewing their milestones and units and managing customers.

Module Hierarchy

A module hierarchy outlines the organization of different functional components in a software system, breaking down the application into manageable sections, each responsible for specific features or tasks.

Figure 5.A

Module Hierarchy (Admin, Broker, Customer)

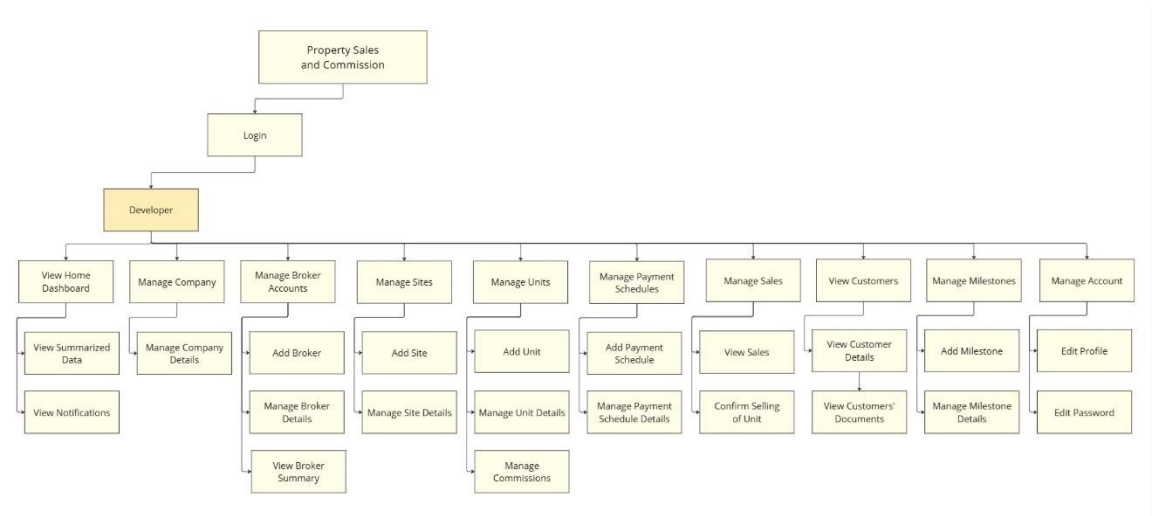


For the system's module hierarchy, one can easily understand the functions of what each module can provide. The admin module allows for overall system management, including handling system maintenance and managing developer accounts. Additionally, the broker module focuses on managing customers and viewing units, commissions and milestones. Furthermore, the customer module allows customers to access linked payment

schedules and document statutes through affiliated links. Lastly, the developer module can be seen in Figure 5.B.

Figure 5.B

Module Hierarchy (Developer)



Shown in Figure 5.B is the sub-modules under the developer module. The developer module focuses on managing companies, broker accounts, sites, units, payment schedules, sales and milestones, and view customers.

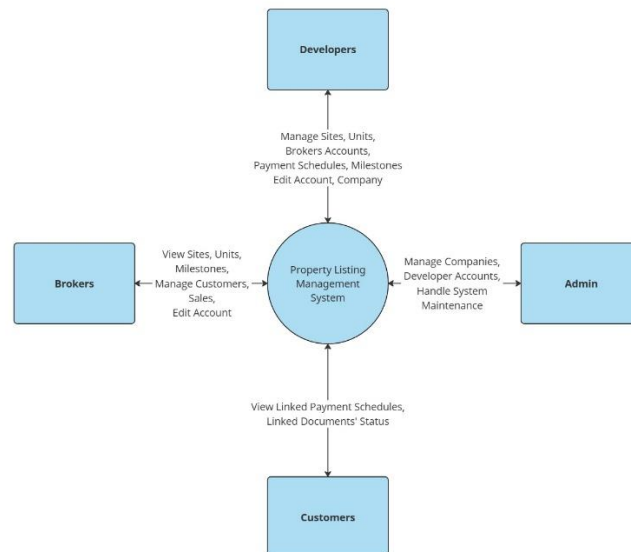
Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the flow of data within a system. It illustrates how data is processed by a system in terms of inputs and outputs, and how it moves between different entities, processes, and data stores. DFDs help in understanding the overall functionality of the system and the interactions between its components.

A Context Diagram is the highest level in a Data Flow Diagram and provides a broad overview of the system. It depicts the system as a single process and shows its interaction with external entities. The context diagram helps in understanding the system boundaries and the main data exchanges with the external environment.

Figure 6

Data Flow Diagram – Level 0 (Context Diagram)



The context diagram provides a high-level view of the Property Listing Management Platform, illustrating the interactions between the system and external entities. The key external entities include developers, brokers, admins, and customers. Developers can interact with the system to manage units, brokers accounts, payment schedules, units and company, and edit accounts. Brokers can view sites, units and milestones, manage customers, and edit account. Customers can view linked payment schedules and

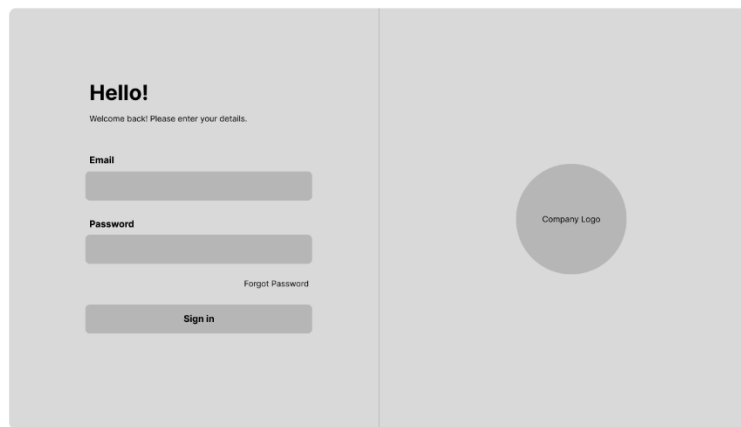
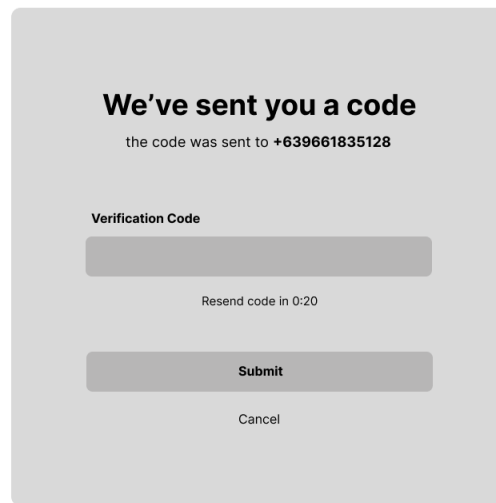
Wireframe**Figure 8*****Login Page***

Figure 8 shows the login page, which serves as a secure entry point for users to access their accounts by entering their credentials. Upon signing in, users must complete two-factor authentication to access the PLMP online platform.

Figure 9*Two-Factor Authentication*

The image shows a mobile application screen for two-factor authentication. At the top, it says "We've sent you a code" in bold, followed by "the code was sent to +639661835128". Below this is a label "Verification Code" above a text input field. Under the input field, it says "Resend code in 0:20". At the bottom, there are two buttons: "Submit" and "Cancel".

Figure 9 displays the page for two-factor authentication, which serves as an additional security measure for users. After submitting the correct code, users will gain access to the PLMP online platform.

Figure 10*Admin Account Management*

COMPANY NAME
Property Listing Management Platform

Welcome Back, Admin!

ACCOUNT MANAGEMENT

Sort: Date Created ▾ All Status ▾ Search Account Add +

<input type="checkbox"/>	Account Name	Date Created	Email Address	Status	Actions
<input type="checkbox"/>	Gwynn Rafer Cujardo	01/10/2024	gwynn.cujardo@tup....	Active ●	✎ 🗑
<input type="checkbox"/>	Gwynn Rafer Cujardo	01/10/2024	gwynn.cujardo@tup....	Active ●	✎ 🗑
<input type="checkbox"/>	Gwynn Rafer Cujardo	01/10/2024	gwynn.cujardo@tup....	Active ●	✎ 🗑

Sign Out

< 1 2 3 4 ... 10 >

Figure 10 shows the admin account management page, where admins can create accounts for the developers. Admins can also view the list of accounts that have access to the PLMP online platform.

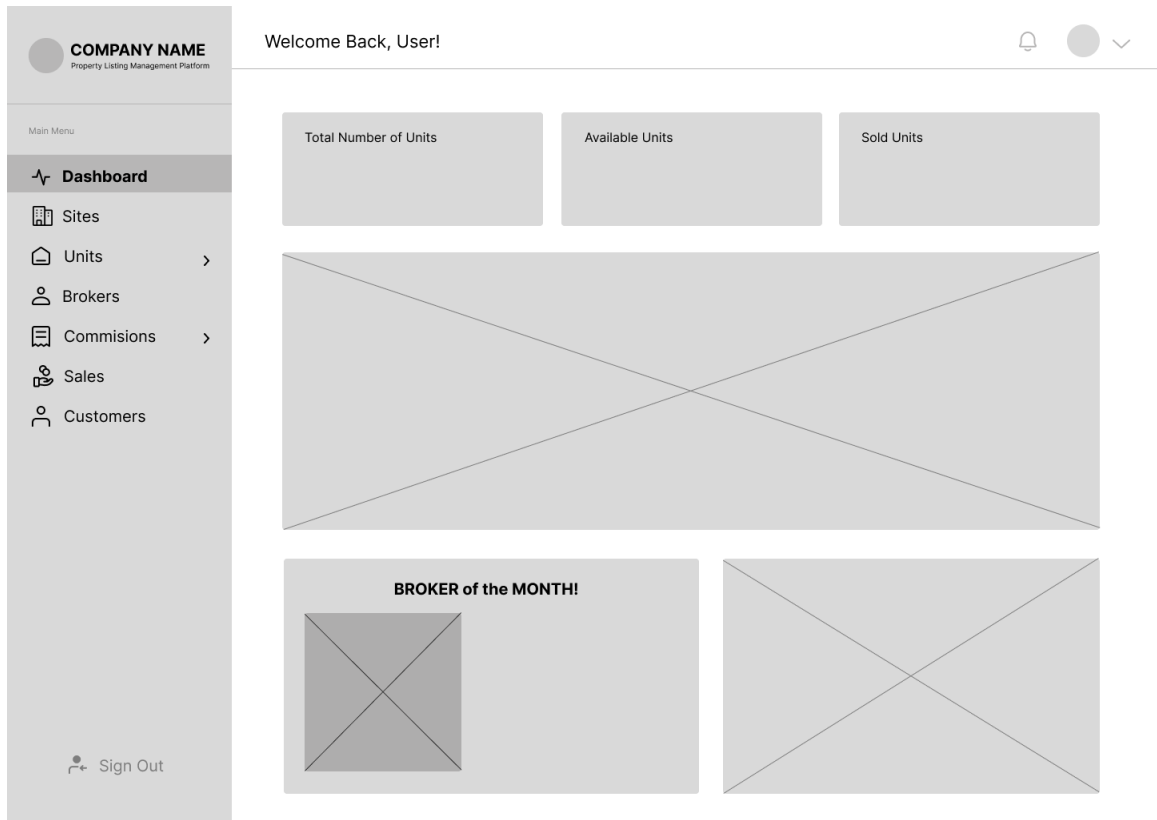
Figure 11*Dashboard - Developer*

Figure 11 displays the developer's dashboard, which is the first screen they see after logging into the system. It showcases the total number of units, available units, sold units, and the broker of the month.

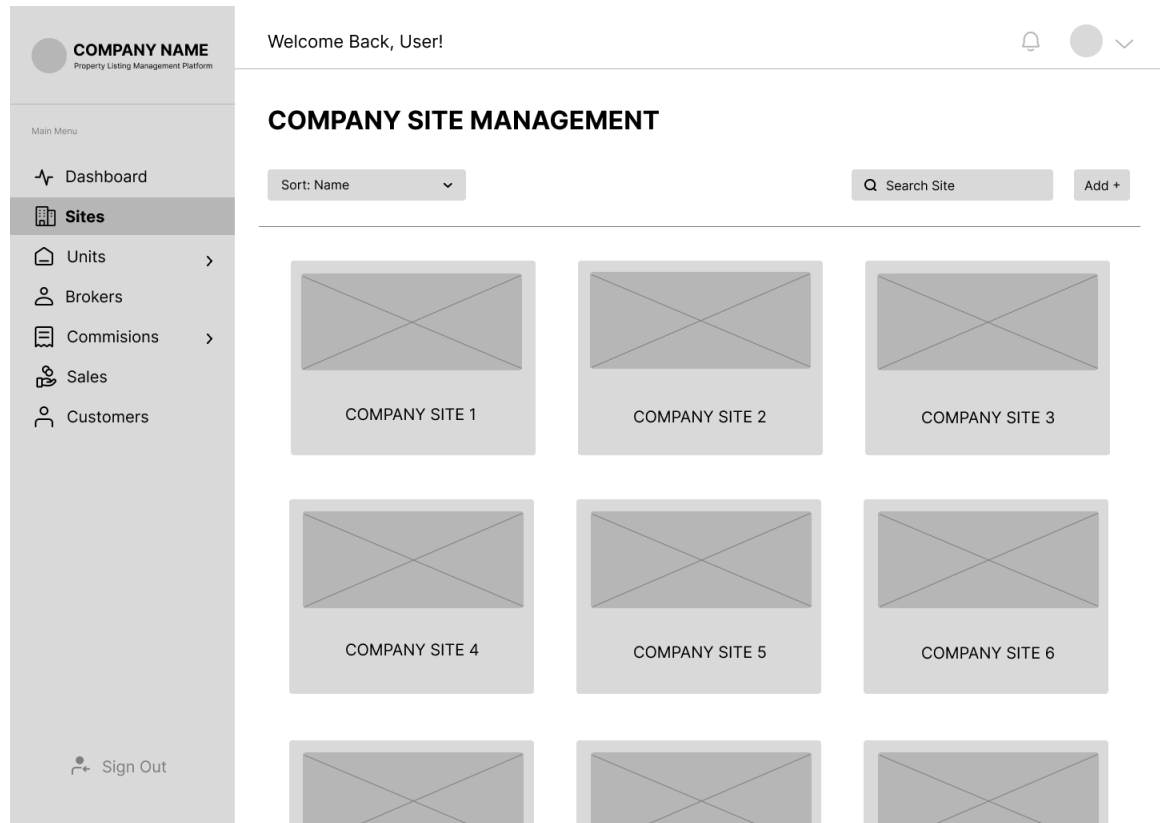
Figure 12*Sites*

Figure 12 shows the sites managed by the developers. On this page, developers can view sites that are available, preselling, and sold out, as well as add new sites to the platform.

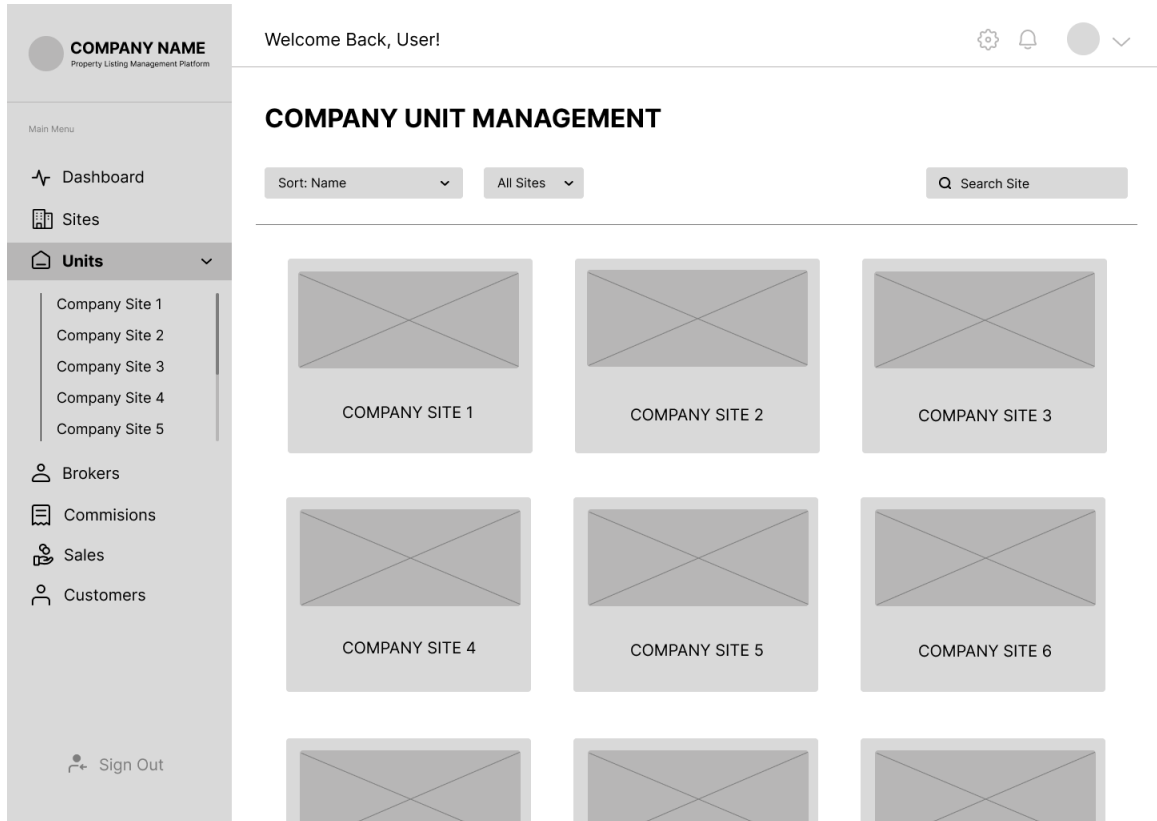
Figure 13*Units – Developers*

Figure 13 displays the first screen that the user sees after clicking ‘Units’ on the right side (Main Menu). On this page, developers select the site where they want to add a unit.

Figure 14

Add Units

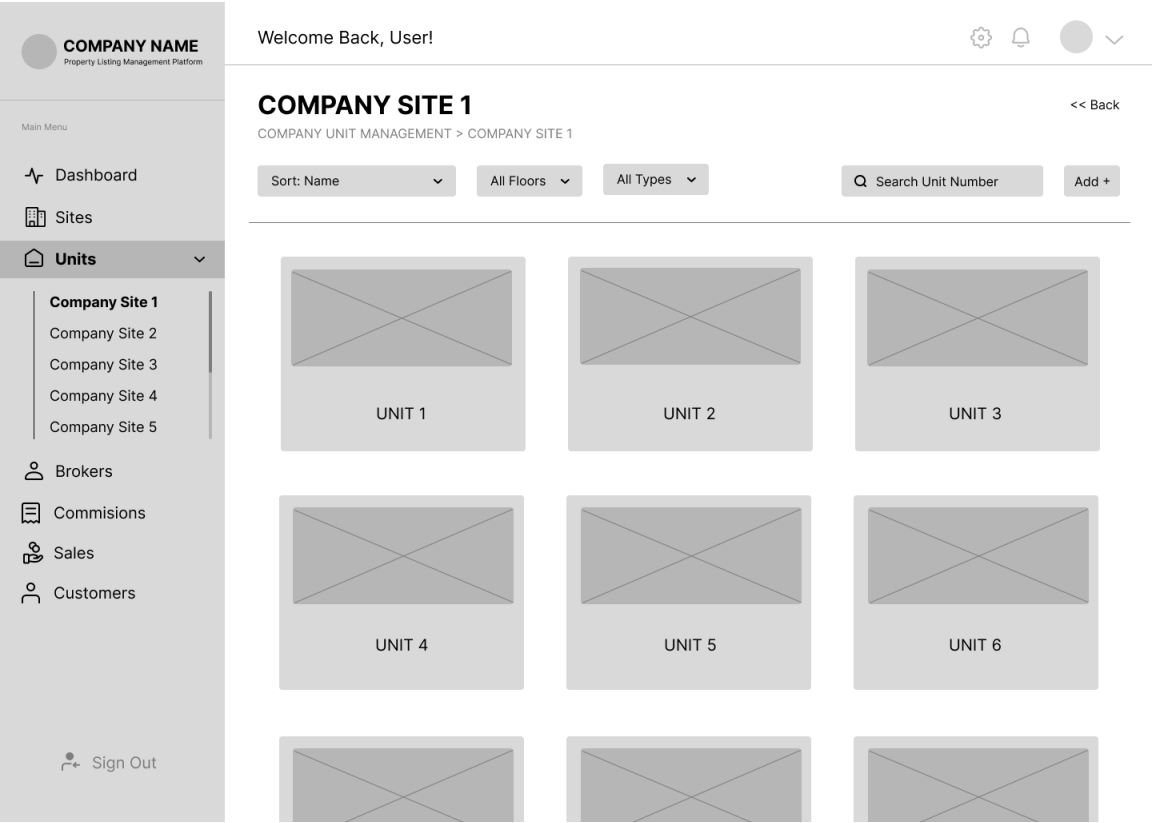


Figure 14 shows the screen that appears after clicking a site in Figure 13. On this page, developers can add units under the selected site.

Figure 15*Adding Units*

The screenshot displays the 'Add UNIT NAME' form within the 'COMPANY NAME Property Listing Management Platform'. The interface includes a sidebar menu with options like Dashboard, Sites, Units, Brokers, Commissions, Sales, and Customers. The main content area shows the form title 'Add UNIT NAME' and a breadcrumb trail 'COMPANY SITE MANAGEMENT > COMPANY SITE 1 > ADD UNIT'. The form is divided into three main sections: 'DETAILS', 'AMMENITIES', and 'PRICING DETAILS'. The 'DETAILS' section contains fields for Unit Number, Floor Number, Build (Year), Fully Furnished, Classification, Bedrooms, Baths, Floor Area (m²), Balcony, View, and Type. The 'AMMENITIES' section features a 'Select Amenities' dropdown and a list of amenity checkboxes. The 'PRICING DETAILS' section includes fields for Total Price, Total Broker Commission, and Interest Rate. A 'Sign Out' button is located at the bottom left of the sidebar.

COMPANY NAME
Property Listing Management Platform

Welcome Back, User!

Add UNIT NAME
COMPANY SITE MANAGEMENT > COMPANY SITE 1 > ADD UNIT

Save Cancel

DETAILS

Unit Number: Floor Number:

Build (Year): Fully Furnished:

Classification:

Bedrooms: Baths:

Floor Area (m²): Balcony:

View: Type:

AMMENITIES

Select Amenities

☐ ☐ ☐ ☐

Add Amenities.....

PRICING DETAILS

Total Price:

Total Broker Commission:

INSTALLMENT DETAILS

Interest Rate:

Sign Out

Figure 15 displays the screen that appears after clicking the ‘Add’ button in the upper right corner of Figure 14. On this page, the developers can fill in the information about the unit.

Figure 16*Broker Account Management*

COMPANY NAME
Property Listing Management Platform

Welcome Back, User!

Main Menu

- Dashboard
- Sites
- Units
- Brokers**
- Commissions
- Sales
- Customers

Sign Out

BROKER ACCOUNT MANAGEMENT

Sort: Name All Affiliation Search Broker Add +

Broker Name	Email Address	Contact No.	Affiliation	Actions
Gwynn Rafer Cujardo	gwynn.cujardo@tup...	+63 999 999 9999	Team A	
Gwynn Rafer Cujardo	gwynn.cujardo@tup...	+63 999 999 9999	Team A	
Gwynn Rafer Cujardo	gwynn.cujardo@tup...	+63 999 999 9999	Team A	

< 1 2 3 4 ... 10 >

Figure 16 shows the ‘Brokers’ page, where developers can view a list of all brokers working under them and create accounts for new brokers.

Figure 17

Add Broker Account

COMPANY NAME

Property Listing Management Platform

Main Menu

Dashboard

Sites

Units

Brokers

Commissions

Sales

Customers

Sign Out

Welcome Back, User!

Sort: Name

Broker Name

Gwynn Rafer Cujardo

Gwynn Rafer Cujardo

Gwynn Rafer Cujardo

First Name

Last Name

Username

Password

Affiliation

Create Account

Cancel

Broker

Add +

Actions

< 1 2 3 4 ... 10 >

Figure 17 displays the form where the developer fills in the broker’s information to create an account.

Figure 18*Broker's Commission*

COMPANY NAME
Property Listing Management Platform

Welcome Back, User!

Main Menu

- Dashboard
- Sites
- Units
- Brokers
- Commissions**
- Sales
- Customers

Sign Out

BROKER COMMISSION

Sort: Name All Affiliation Search Broker

Broker Name	Total Commission
Gwynn Rafer Cujardo	₱ 0.00
Gwynn Rafer Cujardo	₱ 0.00
Gwynn Rafer Cujardo	₱ 0.00

Figure 18 shows the ‘Commissions’ page, where the developers can view the list of the brokers along with their earned commissions.

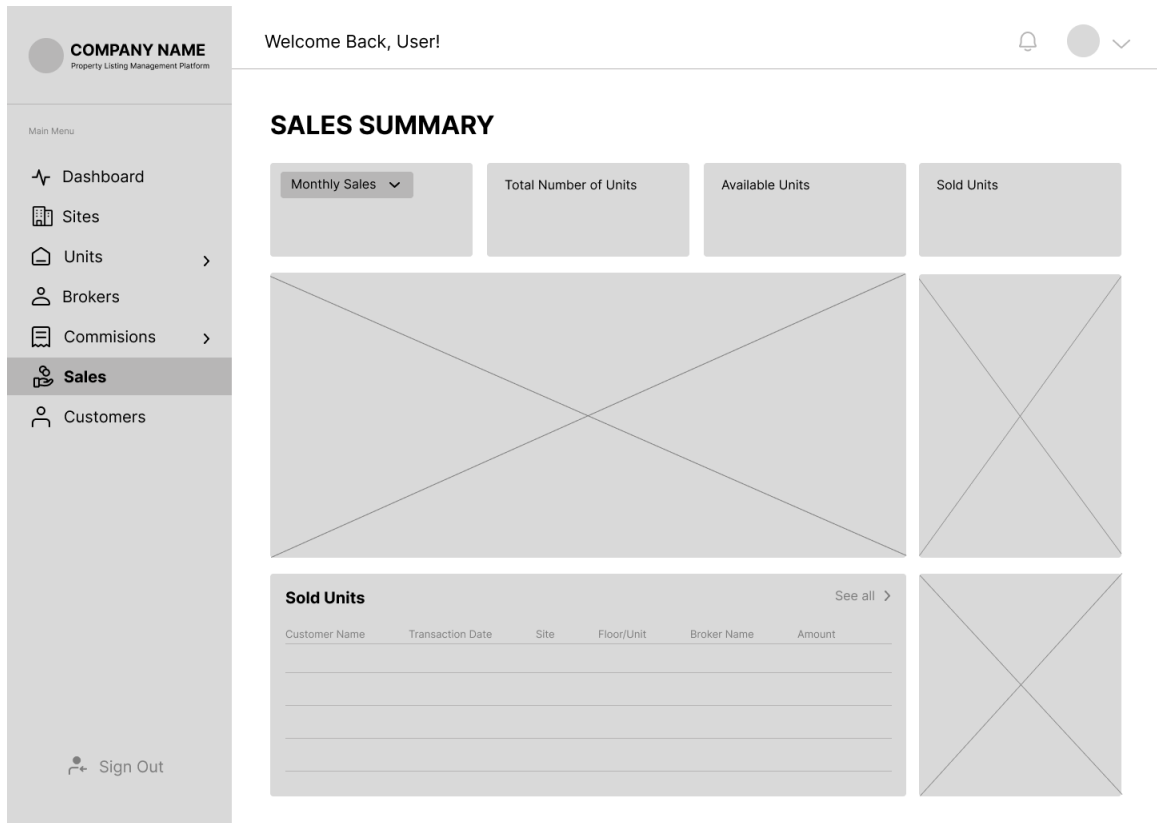
Figure 19*Sales*

Figure 19 displays the 'Sales' page, where the developers can view a summary of sales, including the total number of units, available units, and sold units.

Figure 20*Customers – Developers*

COMPANY NAME
Property Listing Management Platform

Welcome Back, User!

Main Menu

- Dashboard
- Sites
- Units
- Brokers
- Commissions
- Sales
- Customers**

Sign Out

CUSTOMER DETAILS

Sort: Name | All Sites | All Brokers | Search Customer Name

Customer Name	Site	Units Owned	Broker Name
Gwynn Rafer Cujardo	Site 1	2	Arsenic Salangsang
Gwynn Rafer Cujardo	Site 1	1	Arsenic Salangsang
Gwynn Rafer Cujardo	Site 1	1	Arsenic Salangsang
Gwynn Rafer Cujardo	Site 1	1	Arsenic Salangsang

< 1 2 3 4 ... 10 >

Figure 20 shows the ‘Customers’ page, where the developers can view the list of customers who have purchased a unit.

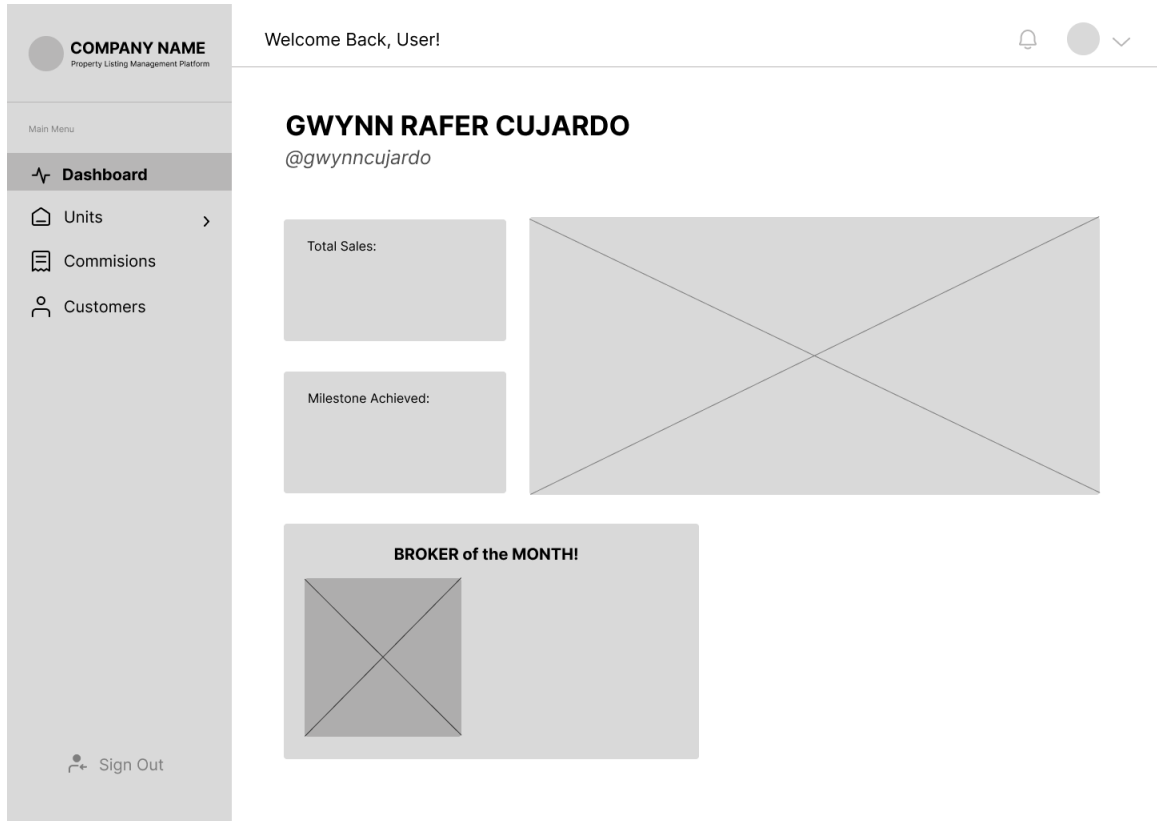
Figure 21*Dashboard – Broker*

Figure 21 displays the broker's dashboard, which is the first screen they see after logging into the system. It showcases the total sales, milestone achieved, and the broker of the month.

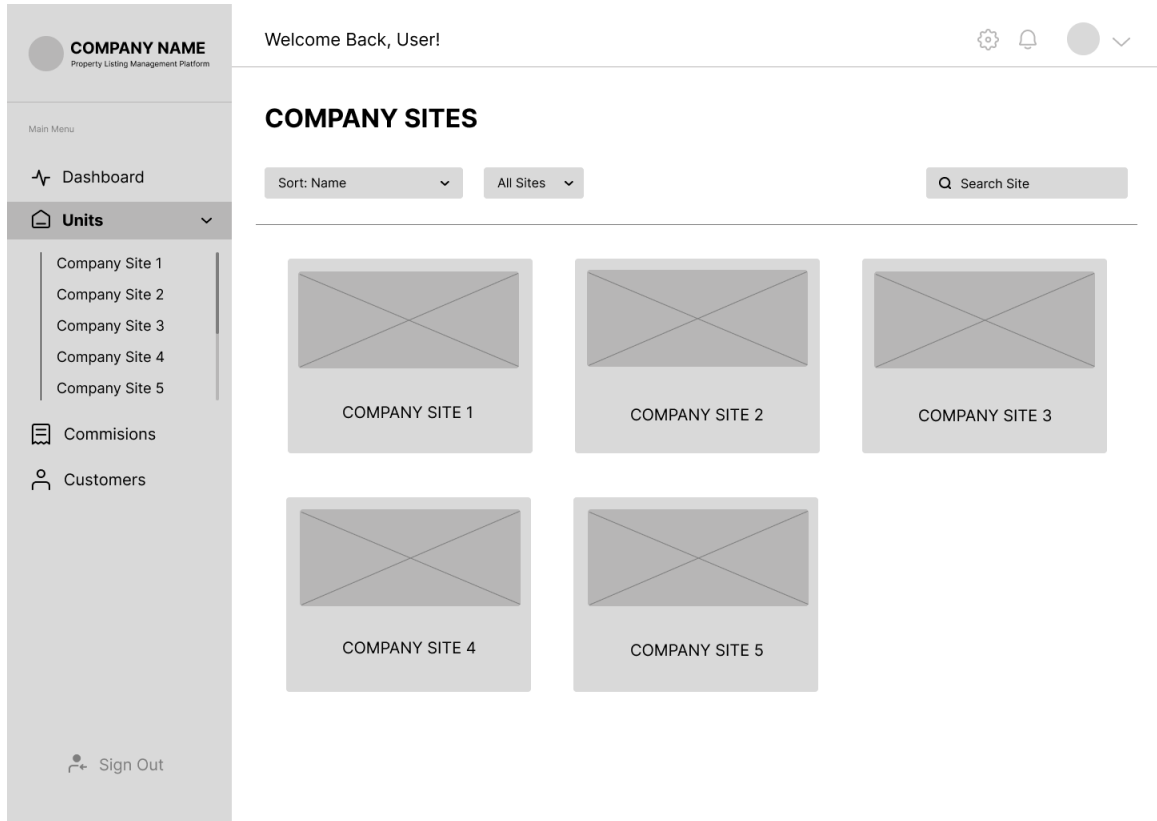
Figure 22*Units – Broker*

Figure 22 shows the ‘Units’ page, where the brokers can view the sites with units available for sale.

Figure 23

Available Units

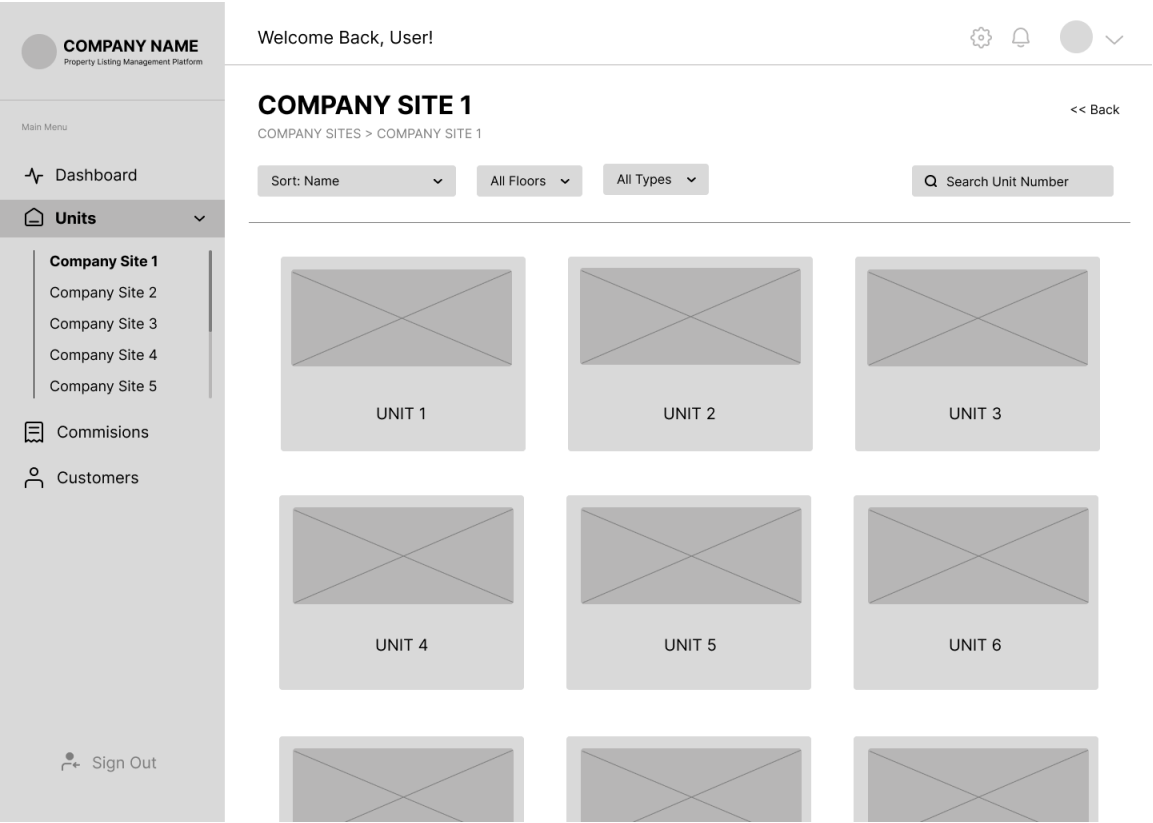


Figure 23 displays the screen that appears after clicking a site in Figure 22. On this page, brokers can view the available units for sale under the selected site.

Figure 24

Unit Information

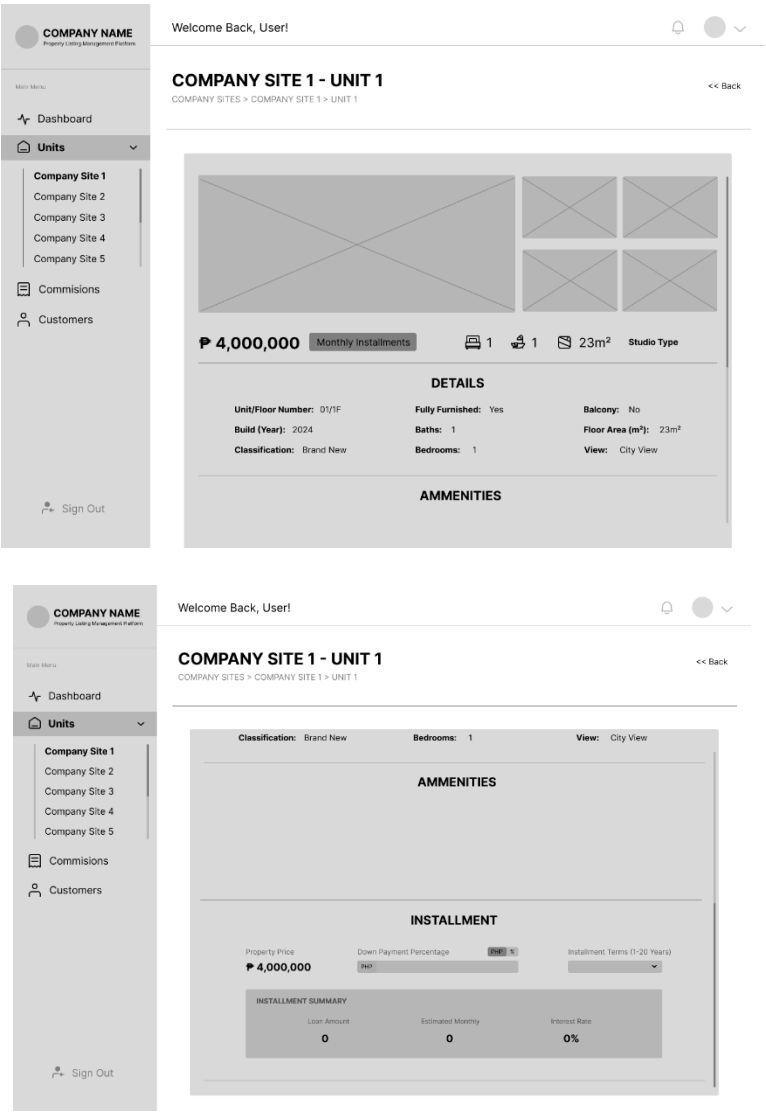


Figure 24 shows the screen after selecting an available unit in Figure 23. This page showcases the information for the selected unit.

Figure 25

Commissions – Broker

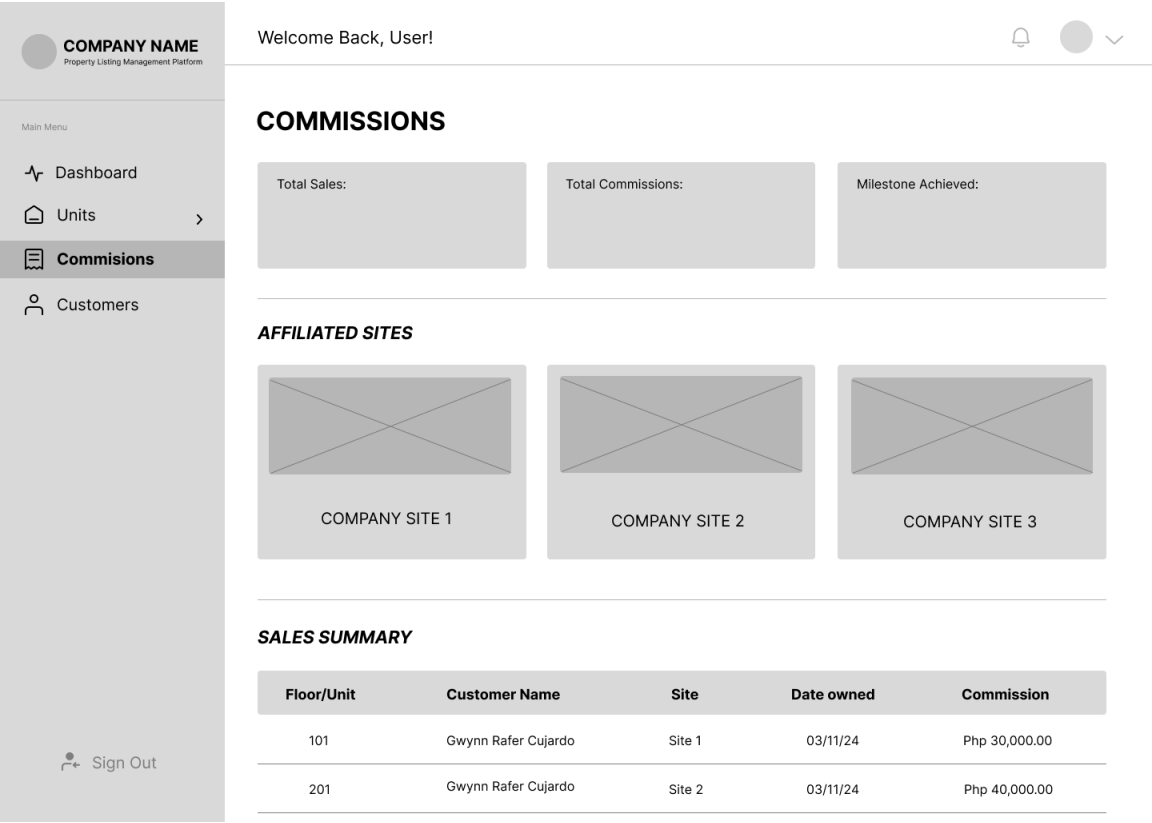


Figure 25 displays the ‘Commissions’ page, where the broker can view their total sales, total commissions, milestones achieved, and their sales summary.

Figure 26

Customers – Broker

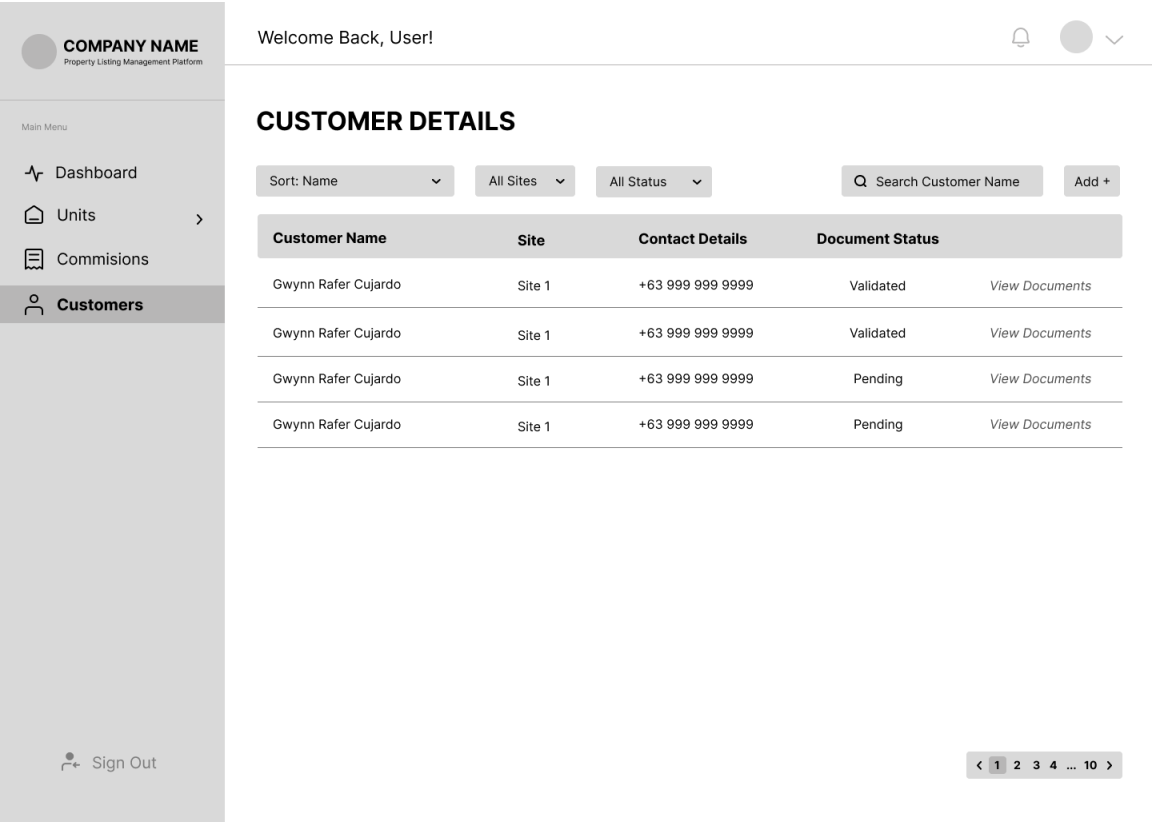


Figure 26 shows the list of customers the broker has handled.

Figure 27*Document Status – Customer*

COMPANY NAME
Property Listing Management Platform

Welcome, Customer Name!

Main Menu

Document Status

Payment Schedule

GWYNN RAFER CUJARDO

Assigned Broker: **ARSENIC SALANGSANG**
Selected Unit: **UNIT 101 / SITE 1**

DOCUMENTS NEEDED: Please submit the following documents to complete the process.

Document Name	Document Status	Date of Submission
VALID GOVERNMENT ID	Submitted	03/11/24
RESERVATION AGREEMENT FORM	Submitted	03/11/24
RESERVATION FEE RECEIPT	Pending	Not yet submitted
CUSTOMER'S INFORMATION SHEET	Pending	Not yet submitted

Ensure that all documents are up-to-date and complete.
If you have any questions or need further assistance, please contact **ARSENIC SALANGSANG**.

CONTACT INFORMATION:
Email: **arsenic.salangsang@tup.edu.ph**
Phone Number: **(+63) 966 183 5128**

Sign Out

Figure 27 displays the first page the customer sees after accessing the link provided by the broker. This page showcases the documents that the customers have submitted and those that still need to be submitted.

Figure 28

Payment Schedule – Customer

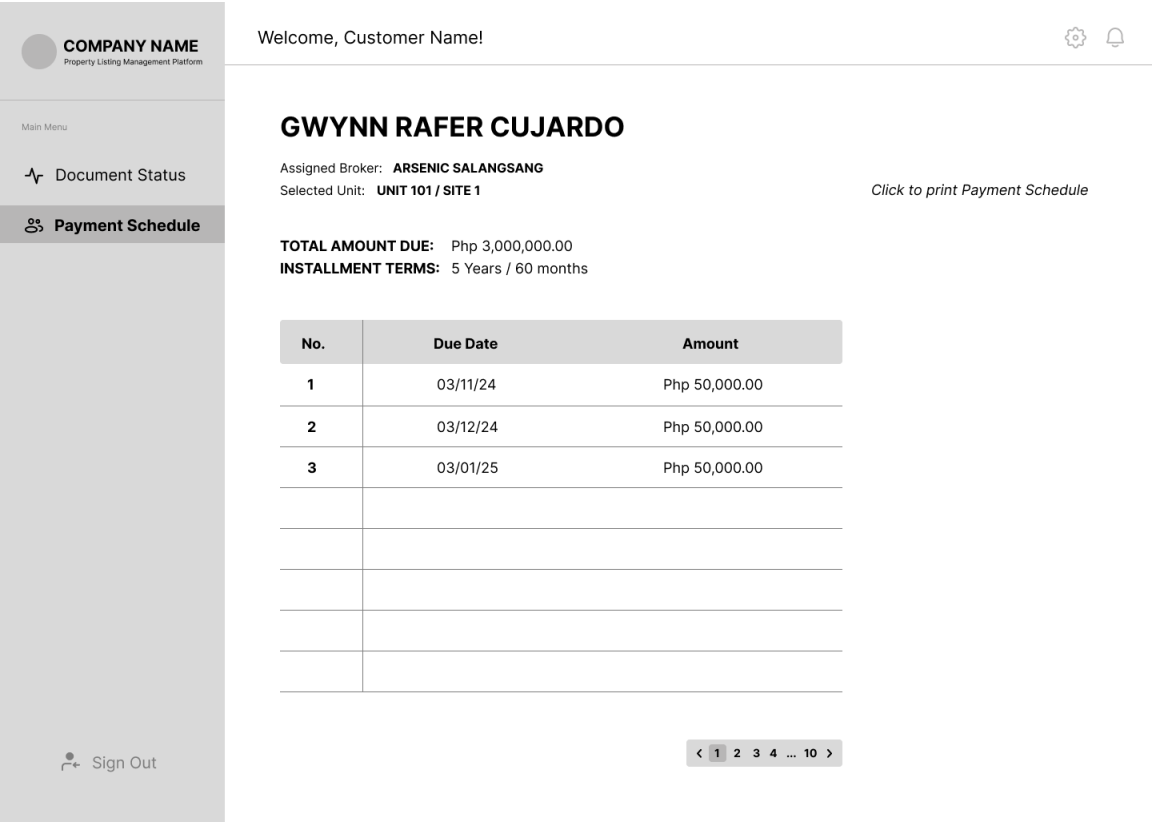
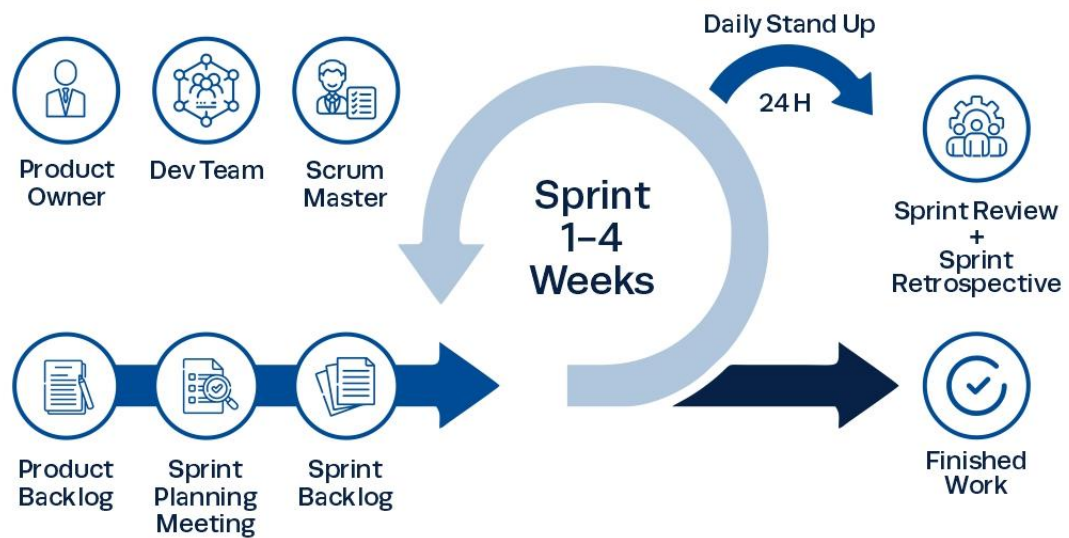


Figure 28 shows the ‘Payment Schedule’ page, which the customer can also view after accessing the provided link. This page displays the payment due dates and the corresponding amounts.

Project Development

Figure 29

Agile Scrum Methodology Framework



The methodology the researchers will use in developing the Listing and Management Platform is Agile Scrum Methodology. According to Peek (2023), agile scrum methodology is a project management system that relies on incremental development. The researchers decided to use this methodology as it is the most befitting methodology considering the timeline and general ability of the researchers. Additionally, it will allow the researchers to work on projects in small increments, which results in easily identifying the problems the researcher will encounter.

Phases:

- **Sprint Planning Meeting.** In this phase, the researchers will meet to discuss the module from the product backlog that they will work on in the next sprint and the timeline for doing so.
- **Daily Scrum Meeting.** In this phase, the researchers will hold a meeting to monitor and evaluate the project's status and ensure that everyone is working on their assigned tasks.
- **Sprint Review.** This phase is conducted at the end of every sprint. In this phase, the researchers will present their finished work to their capstone adviser. The adviser will then review the work and decide whether any changes are needed or if the work is acceptable.
- **Sprint Retrospective.** In this phase, the researchers will hold a meeting to discuss what went well during the last sprint and what can be improved in the next one. This meeting will result in continuous improvements, making each sprint more successful than the last.

Roles:

- **Product Owner.** An individual or business who will utilize the software. They are the ones who determine which features and modules will be included in the product, what changes will be made, and whether to accept or reject the completed product.
- **Development Team.** The researchers who are responsible for delivering the committed product increments.

- **Scrum Master.** An individual responsible for the productivity and effectiveness of the development team. They ensure that the team comprehends scrum principles, theoretically and practically, providing aid and guidance as the team advances through their sprints.

Tools:

- **Product Backlog.** This tool comprises a comprehensive list of features and modules derived from the needs and requirements to develop LivWell: A Rental Property Listing and Management Platform.
- **Sprint Backlog.** This tool is the tasks selected from the product backlog during the Sprint Planning Meeting phase that need to be accomplished in the upcoming sprint.

Operating and Testing Procedure

To guarantee that the system works properly, its features and capabilities were thoroughly tested. The testing method validates each function in the system if it is working and performing as planned.

Functional Suitability Testing will be performed to examine the functionality of the web application's features to ensure that it is functioning based on the intended results. The test cases contain a set of inputs, execution precondition, and expected outcomes. The form for test cases will be utilized, as shown in Table 1.

Table 1*Sample Test Case*

Test Case ID		UC Reference	
Objective			
Assumptions/ Preconditions			
Actions		Expected Result	Actual Result
Status		Severity	MAJOR
		Priority	HIGH

The table is a template of the test case form used by the researchers. Each row represents a test case with details such as the test case ID, objective, expected result, action procedure, test data, expected result, and actual result. The status column indicates whether the test case passed, failed, was not executed, or was blocked. This structured format ensures thorough testing and validation of the web application's functionality.

Table 2

Classification of Error Severity

Severity	Description
Critical	The problem signifies that the process has been completely stopped and cannot continue until it is resolved.
Major	The problem causes the system to crash. Nevertheless, some system components are still functional.
Minor	The problem does not result in any significant system failure.

The table offers a detailed classification of error severity levels, including Critical, Major, and Minor categories. This structured approach serves as a quick reference guide for assessing the impact and severity of faults within a system. By categorizing errors into these severity levels, stakeholders can efficiently prioritize and allocate resources to address issues effectively, ensuring timely resolution of critical problems and efficient management of less severe issues.

Table 3

Classification of Error Priority Levels

Priority	Description
High	The problem needs to be resolved as soon as possible since it significantly affects the application.
Medium	The problem should be fixed throughout the normal course of development.
Low	The problem must be resolved when a more crucial feature is taken care of.

The table categorizes error priority levels into High, Medium, and Low, along with detailed descriptions for each. This structured approach helps in prioritizing and allocating resources to address and resolve issues based on their impact and urgency. By assigning the appropriate priority level to each error, development teams can efficiently manage their workflow, ensuring that critical issues are fixed promptly while less critical problems are addressed at the right time during the development cycle.

Table 4
Overall Summary of Functionality Test Cases

Use Case	No. Of Test Cases
Unregistered User	
Verified User	
Total	

The table categorizes test cases based on user roles, including "Unregistered User," and "Verified User,". The "Total" row aggregates the total number of test cases across all user roles. This structured format serves as a template for capturing and organizing test case information, ensuring comprehensive coverage of functionality for each user type.

Table 5
Reliability Test Cases Summary

Test Case ID	Objectives

The test case ID is a unique identifier assigned to each test case, allowing for easy reference and tracking. Each test case's objectives include a detailed description of the goals and purposes, specifically outlining the aspects of reliability being tested or verified.

Table 6*Testing Procedure for Functionality Suitability*

Module	Steps to be undertaken	Expected Result
1. Login	<ol style="list-style-type: none"> 1. From the home page, select your role (Broker or Developer) 2. Click the corresponding login button 3. Enter user details. 4. Click “Login”. 	<ol style="list-style-type: none"> 1. The user will be redirected to the two-factor authentication with correct details. 2. Error message will be displayed if the user entered wrong details.
2. Two-Factor Authentication	<ol style="list-style-type: none"> 1. Enter the verification sent to your cellphone number. 2. Click “Submit”. 	<ol style="list-style-type: none"> 1. The user will be redirected to the selected role’s dashboard with correct verification code. 2. Error message will be displayed if the user entered wrong details.
3. Manage Sites	<ol style="list-style-type: none"> 1. From the side bar, select “Sites”. 2. Click “Add” at the upper right. 3. Input details about the site. 4. Click the “Save” button. 	<ol style="list-style-type: none"> 1. The site will be added to the company site.
4. Manage Units	<ol style="list-style-type: none"> 1. From the side bar, select “Units”. 2. Select the site you want to add the units. 3. Click “Add” at the upper right. 4. Input details about the unit. 5. Click the “Save” button. 	<ol style="list-style-type: none"> 1. The unit will be added as a unit under a company site. 2. The payment schedule and commission rate of the broker are also set.
5. Manage Broker Accounts	<ol style="list-style-type: none"> 1. From the side bar, select “Brokers”. 2. Click “Add” at the upper right. 3. Input required details for the broker’s account. 4. Click the “Save” button. 	<ol style="list-style-type: none"> 1. The broker’s account will be created and can be used to log in to the system.

6. View Milestones	1. From the side bar, select “Dashboard”.	1. The broker will see the total sales, milestones achieved, and the broker of the month.
7. View Affiliated Units	1. From the side bar, select “Units”. 2. Choose a site where the customer wants to view the available units. 3. Select a unit.	1. The broker will see the information for the selected unit within the chosen site.
8. Manage Customers	1. From the side bar, select “Customers”. 2. Click the “Add” button in the upper right side. 3. Fill in the customer details and upload the document files.	1. The broker will see a list of customers that he or she has handled. 2. Customer information is added to the list, and the documentary requirements are also uploaded.
9. View Commissions	1. From the side bar, select “Commissions”.	1. The broker will see the total sales, total commissions, milestone achieved, and sales summary.
10. View Linked Payment Schedule	1. From the side bar, select “Payment Schedule”	1. The customer will see the due date and the amount they should pay.
2. View Linked Document Status	1. From the side bar, select “Document Status”	1. The customer will see the list of documents they have submitted and the documents they still need to submit.

Reliability Testing will be conducted to evaluate the system's performance under specific environmental conditions and over a specified period. The table above will be used to document and organize test cases for various system modules and functionalities, including sign up, log in, property listings management, property management, tenant management, report, and payment. This structured approach ensures that each capability is thoroughly tested and validated against expected outcomes.

Table 7*Testing Execution Summary*

Test Execution	Expected Result	Active Result Cycle 1	Cycle 2
No. of Test Cases Executed	100%		
Results of Test Cases			
Passed	100%		
Failed	0%		
No. of Test Cases Not Executed	0%		

The table provides a summary of the test execution, indicating that all test cases have been completed with no remaining unexecuted tests. The expected results column is currently blank and requires filling in with the anticipated outcomes for each test case. The results of the tests conducted show that all test cases passed without any failures. This table serves as a status overview, confirming that the testing was successful and complete.

Evaluation Procedure of the Study

In assessing the system, ISO 25010 will be utilized to evaluate software product quality. The prototype will be reviewed by a diverse group of evaluators, including students, faculty members, professional workers, and brokers. Participants, including both faculty and students, will be drawn from the College of Science as well as other programs from various colleges related to technology for the evaluation. The evaluation will include at least 30 respondents to ensure a comprehensive assessment of the prototype's performance and

quality. During the system demonstration, audience members will be provided with surveys to complete. The evaluation procedure is presented as follows:

General Evaluation Process

1. Each respondent will receive a software assessment form to evaluate the effectiveness of the system.
2. The web-based application's functions will be demonstrated, and a discussion of its objectives will be held to demonstrate how to use it.
3. The system will be made available to all respondents.
4. According to Table eight (8), which uses a four-point Likert scale, where four (4) is the highest rating and one (1) is the lowest rating, the respondents will be asked to evaluate the system considering the ISO 25010 standard evaluation criteria.
5. This project will be evaluated using the following characteristics: functional suitability and reliability.
6. The overall weighted mean rating for each criterion and the grand weighted mean will be determined based on the evaluation data to be collected.
7. The evaluation outcome will be made use of the weighted mean value range shown in table nine (9) and the qualitative analysis relating to that range.

Table 8*Four-point Likert Scale*

Scale	Descriptive Rating
4	Highly Acceptable
3	Very Acceptable
2	Fairly Acceptable
1	Not Acceptable

The table above presents a scale with numerical values and descriptive ratings ranging from 1 to 4. On this scale, 4 indicates "Highly Acceptable," 3 indicates "Very Acceptable," 2 indicates "Fairly Acceptable," and 1 indicates "Not Acceptable." This scale is used to evaluate the level of acceptability of various items or criteria, with higher numerical values representing more favorable ratings. The descriptive ratings provide qualitative insight into the corresponding numerical values, aiding in the identification and analysis of acceptability.

Table 9*The Range of Mean Ratings and the Equivalent Descriptive Rating*

Scale	Descriptive Rating
3.26 - 4.00	Highly Acceptable
2.51 - 3.25	Very Acceptable
1.76 - 2.50	Fairly Acceptable

1.00 - 1.75

Not Acceptable

Each row in the table above represents a numerical range, and the corresponding column provides a descriptive rating. This scale is used to assess or grade something based on its level of acceptability, with each rating associated with a specific numerical range. For example, a value between 3.26 and 4.0 is classified as "Highly Acceptable." Similarly, values within other ranges are labeled as "Very Acceptable," "Fairly Acceptable," or "Not Acceptable," according to the specified descriptors.

REFERENCES

- Adetunji, H. (2023, December 28). What is HTML: A detailed explanation of the programming language. *Medium*. <https://medium.com/@hikmohadetunji/what-is-html-a-detailed-explanation-of-the-programming-language-1d0ab69f475d>
- Arranz, M. M. G. (2021, December 26) 5 Condo unit types: Which one is for you? Federal Land, Inc. <https://federalland.ph/articles/condo-unit-types/>
- BasuMallick, C. (2022, November 7). *CSS Guide: How it Works and 20 Key Properties - Spiceworks Inc.* Spiceworks Inc. <https://www.spiceworks.com/tech/tech-general/articles/what-are-css/>
- Bracey, K. (2024, January 2). *What is Figma?* Web Design Envato Tuts+. <https://webdesign.tutsplus.com/what-is-figma--cms-32272a>
- Britton, J. (2021, May 6). *What is ISO 25010?* Perforce Software. <https://www.perforce.com/blog/qac/what-is-iso-25010>
- CFI Team. (2024, July 10). Commission. Corporate Finance Institute. <https://corporatefinanceinstitute.com/resources/career/commission/>
- Chen, J. (2024, July 25). What is a condominium? How condos work compared to apartments. Investopedia. <https://www.investopedia.com/terms/c/condominium.asp>
- Chris, K. (2023, February 3). *Visual Studio vs Visual Studio Code – What's The Difference Between These IDE Code Editors?* freeCodeCamp.org. <https://www.freecodecamp.org/news/visual-studio-vs-visual-studio-code/>
- Corbo, A. (2022, December 29). *What is vue JS?* Built In. <https://builtin.com/software-engineering-perspectives/vue-js>
- Coursera. (2023, November 29). *What is GitHub and why should you use it?* Coursera. <https://www.coursera.org/articles/what-is-git>
- Coursera. (2024, April 3). *What is Python used for? A beginner's guide.* Coursera. <https://www.coursera.org/articles/what-is-python-used-for-a-beginners-guide-to-using-python>
- Customer stories.* (2024). GitHub. <https://github.com/customer-stories>
- Django introduction - Learn web development | MDN.* (2024, February 28). MDN Web Docs. <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction>

Doerrfeld, B. (n.d.). *PostgreSQL: The database that quietly ate the world*. LeadDev.

<https://leaddev.com/tech/postgresql-database-quietly-ate-world>

Eygi, C. (2021, December 9). CSS for Beginners: What is CSS and How to Use it in Web Development? *Medium*. <https://medium.com/swlh/css-for-beginners-what-is-css-and-how-to-use-it-in-web-development-5985afe53096>

Faelnar, I. C. (2024, February 24). Different types of condominiums in the Philippines - now realty brokerage. Now Realty Brokerage. <https://www.nowrealtybrokerage.com/different-types-condominiums-philippines/>

Folger, J. (2024, August 27). How do real estate agents get paid? Investopedia. <https://www.investopedia.com/articles/personal-finance/080714/how-do-real-estate-agents-get-paid.asp>

G, D. (2023, May 15). *What is CSS and how does it work?* Hostinger Tutorials. <https://www.hostinger.ph/tutorials/what-is-css>

GeeksforGeeks. (2024, February 2). *Introduction to JavaScript*. GeeksforGeeks. <https://www.geeksforgeeks.org/introduction-to-javascript/>

GeeksforGeeks. (2024, April 18). *Django Tutorial Learn Django Framework*. GeeksforGeeks. <https://www.geeksforgeeks.org/django-tutorial/>

Ghasemaghaei, M. (2019). Does data analytics use improve firm decision making quality? The role of knowledge sharing and data analytics competency. *Decision Support Systems*, 120, 14–24. <https://doi.org/10.1016/j.dss.2019.03.004>

Giglio, P. (2024, June 21). What is Figma? - Pierluigi Giglio - Medium. *Medium*. https://medium.com/@pierluigi.giglio.01_23202/what-is-figma-f52cb75199be

Hamzah, Muhammad Akram, Mustika, Nur , Mustapa, Mahmud, Rahmah, Ummiati (2022). Application of Boarding House Rental Location Search System Based on Android from http://eprints.unm.ac.id/32270/1/Hamzah_2022_Ceddi_Application%20of%20Barding%20House%20Rental%20Location%20Search%20System%20Based%20on%20Android%20.pdf

Heller, M. (2022, July 8). *What is Visual Studio Code? Microsoft's extensible code editor*. InfoWorld. <https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html>

- Hilbert, M. (2023, May 4). *What is PostgreSQL, and why do businesses need to know more about it?* Redgate. <https://www.red-gate.com/blog/database-development/what-is-postgresql-and-why-do-businesses-need-to-know-more-about-it>
- Ikuomola, A. J., & Asefon, M. P. (2020). A secured mobile cloud-based house rental management system. In *Proceedings of 3rd International Conference on Applied Information Technology (AIT)* (pp. 1-17). https://www.researchgate.net/publication/340926278_A_Secured_Mobile_Cloud-Based_House_Rental_Management_System
- ISO 25010. (n.d.). <https://iso25000.com/index.php/en/iso-25000-standards/iso-25010?start=5>
- Jordana, A. (2024, May 13). *What Is JavaScript: A Beginner's Guide to the Basics of JS*. Hostinger Tutorials. <https://www.hostinger.ph/tutorials/what-is-javascript>
- Kintu, V. (2018, December 6). Payment schedule. Capital Com SV Investments Limited. <https://capital.com/payment-schedule-definition>
- Kugell, A. (2022, April 24). In-Depth look at VueJs: pros, cons, and real-life applications in 2024 - Trio. *Trio*. <https://trio.dev/why-use-vue-js/>
- Lukaszewski, A. (2019, July 3). *What is Python programming Language?* ThoughtCo. <https://www.thoughtco.com/what-is-python-2813564>
- Lutkevich, B. (2020, February 17). *HTML (Hypertext Markup Language)*. TheServerSide.com. <https://www.theserverside.com/definition/HTML-Hypertext-Markup-Language>
- Lutkevich, B., & Courtemanche, M. (2023, February 21). *GitHub*. IT Operations. <https://www.techtarget.com/searchitoperations/definition/GitHub>
- Mansa, Julius. (2023, July 16). How Does Supply and Demand Affect the Housing Market? *Investopedia*. <https://www.investopedia.com/ask/answers/040215/how-does-law-supply-and-demand-affect-housing-market.asp>
- McMillin, D. (2022, September 22). What is a condo? Bankrate. <https://www.bankrate.com/real-estate/what-is-a-condo/#why-buy>
- Megida, D. (2021, April 28). *What is JavaScript? A Definition of the JS Programming Language*. freeCodeCamp.org. <https://www.freecodecamp.org/news/what-is-javascript-definition-of-js/>
- Miller, S. (2023, May 4). *What is Python Used For?* Codecademy Blog. <https://www.codecademy.com/resources/blog/what-is-python-used-for/>

- Mioten, E. (2024, June 27). Condominium living further on the rise. BusinessWorld Online. <https://www.bworldonline.com/special-features/2024/06/28/604807/condominium-living-further-on-the-rise/>
- Mioten, E. (2024, June 27). The Philippine condominium market in review. BusinessWorld Online. <https://www.bworldonline.com/special-features/2024/06/28/604831/the-philippine-condominium-market-in-review/#:~:text=Although%20condominium%20prices%20also%20declined,for%20condominiums%20as%20residential%20property.>
- Misyam, M. R., & Selamat, N. (2021, November 30). House Rental Management System. Applied Information Technology And Computer Science, 2(2), 1745P-1754. <https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/2488>
- Monteverde, A. L., Maderazo, J. J. S., Cruz, K. C. M., & Magnaye, N. A. (2023). A web-based rental house smart finder using rapid application development basis for evaluation of ISO 205010. *International Journal of Metaverse (IJM)*, 1(1), 1-12. <https://doi.org/10.54536/ijm.v1i1.1464>
- Noble Desktop. (2024, May 15). *What is Django and How is it used?*. <https://www.nobledesktop.com/classes-near-me/blog/what-is-django>
- Obrenović, B. Ž. (2021, October 21). ISO 25010 Standard. *Grounded Architecture: Redefining IT Architecture Practice*. <https://grounded-architecture.io/concepts-iso25010>
- Pankratov, E., Grigoryev, V., & Pankratov, O. (2020). The blockchain technology in real estate sector: Experience and prospects. IOP Conference Series Materials Science and Engineering, 869(6), 062010. <https://doi.org/10.1088/1757-899x/869/6/062010>
- Paul, J. (2022). The Rental Zone (House Renting Website). *International Research Journal of Modernization in Engineering Technology and Science*, 4(8), 2243-2248. [https://www.irjmets.com/uploadedfiles/paper/issue 8 august 2022/29592/final/fin irjme ts1662125360.pdf](https://www.irjmets.com/uploadedfiles/paper/issue%208%20august%202022/29592/final/fin_irjmets1662125360.pdf)
- Peek, S. (2023, November 13). What is Agile scrum methodology? <https://www.businessnewsdaily.com/4987-what-is-agile-scrum-methodology.html>
- Rastogi, R., Jain, R., Mishra, P., & Prateek. (2023). Rental House Management System: An Empirical Approach with Simulation. *Acta Scientific COMPUTER SCIENCES*, 5(4). <https://actascientific.com/ASCS/pdf/ASCS-05-0434.pdf>
- Rathore, K., Syed, A., & Patel, R. (2021). Rental house management system. *International Research Journal of Modernization in Engineering Technology and Science*, 3(5), [1098].

- https://www.irjmets.com/uploadedfiles/paper/volume3/issue_5_may_2021/10054/1628083403.pdf
- S, A. (2023, August 25). *What is HTML? Hypertext Markup Language Basics explained*. Hostinger Tutorials. <https://www.hostinger.ph/tutorials/what-is-html>
- Scott, D. W. (2024, August 21). *What is Figma?* <https://bringyourownlaptop.com/blog/what-is-figma>
- Scott, P. (2024, February 28). *What is PostgreSQL? Everything You Need to Know*. Percona Database Performance Blog. <https://www.percona.com/blog/what-is-postgresql-used-for/>
- Segal, T. (2024, March 25). The differences between a real estate agent, a broker, and a realtor. Investopedia. <https://www.investopedia.com/ask/answers/101314/what-are-differences-among-real-estate-agent-broker-and-realtor.asp>
- Senthil.k-Wp. (2024, March 13). *The Benefits of Web-Based Systems for Business - Aezion*. Aezion. <https://www.aezion.com/blogs/the-benefits-of-web-based-systems-for-business/>
- Setty, Y. (2020) Management System for an Apartment. Opus Open Portal to University Scholarship. <https://opus.govst.edu/cgi/viewcontent.cgi?article=1542&context=capstones>
- Shao, C., Yang, Y., Juneja, S., & GSeetharam, T. (2022). IoT data visualization for business intelligence in corporate finance. *Information Processing and Management*, 59(1), 102736. from <https://doi.org/10.1016/j.ipm.2021.102736>
- Simplilearn. (2022, September 22). *The Best Guide to Know What is Vue JS*. Simplilearn.com. <https://www.simplilearn.com/tutorials/java-tutorial/what-is-vue-js>
- Simplilearn. (2024, May 28). Top 25 Productivity tools to get you More time in 2024. Simplilearn.com. <https://www.simplilearn.com/tutorials/productivity-tutorial/best-productivity-tools-to-maximize-your-time>
- Song, C. H., Kil, J., Pan, T.-Y., Sadler, B. M., Chao, W.-L., & Su, Y. (2022). One step at a time: Long-horizon vision-and-language navigation with milestones. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 15461–15470). IEEE. <https://doi.org/10.1109/CVPR52688.2022.01504>
- Statista. (n.d.). Residential Real estate Leases - Philippines | Forecast. <https://www.statista.com/outlook/fmo/real-estate/residential-real-estate/residential-real-estate-leases/philippines#analyst-opinion>
- Tan, V. K., & Nguyen, T. (2022). The real estate transaction trace system model based on

- Ethereum blockchain platform. In 2022 14th International Conference on Computer and Automation Engineering (ICCAE) (pp. 173–177). IEEE. <https://doi.org/10.1109/ICCAE55086.2022.9762429>
- The Investopedia Team. (2024, April 23). Data Analytics: What It Is, How It's Used, and 4 Basic Techniques. Investopedia. <https://www.investopedia.com/terms/d/data-analytics.asp>
- Visual Studio vs Visual Studio Code - What's Best In 2024?* (n.d.). <https://www.turing.com/kb/ultimate-guide-visual-studio-vs-visual-studio-code>
- Voumick, D. , Deb, P. , Sutradhar, S. and Khan, M. (2021) Development of Online Based Smart House Renting Web Application. *Journal of Software Engineering and Applications*, **14**, 312-328. doi: [10.4236/jsea.2021.147019](https://doi.org/10.4236/jsea.2021.147019)
- Webb, M. (2024, February 19). *GitHub*. Techopedia. <https://www.techopedia.com/definition/github>
- What is a payment schedule? (2024, July 17). Corporate. https://www.allianz-trade.com/en_global/news-insights/business-tips-and-trade-advice/what-is-a-payment-schedule.html

APPENDIX A

SOFTWARE EVALUATION INSTRUMENT OF ISO 25010

APPENDIX B

SAMPLED ANSWERED EVALUATION SHEET AND RESTULT SHEETS

APPENDIX C

SUMMARY OF RESPONDENT' EVALUATION

APPENDIX D
TEST CASES FOR FUNCTIONALITY

APPENDIX E
TEST CASES FOR RELIABILITY

APPENDIX F
TEST INCIDENTS LOGS

APPENDIX G
GANTT CHART

APPENDIX H
USER MANUAL

APPENDIX I

THESIS GRAMMARIAN CERTIFICATION

APPENDIX J

CERTIFICATE OF SIMILARITY INDEX USING TURNITIN

RESEARCHERS' PROFILE

GWYNN R. CUJARDO

Apt. 1, Santan St. Georgieville Subd. Tarcán, Baliuag City, Bulacan
(+63) 966 183 5129 | gwynn.cujardo@tup.edu.ph



PERSONAL INFORMATION

Date of Birth: March 12, 2003

Place of Birth: Muntinlupa City

Citizenship: Filipino

Gender: Male

Civil Status: Single

Interest: Basketball, Online Games

EDUCATIONAL BACKGROUND

Primary:

Tarcán Elementary School

Secondary:

Escuela de Angela Marici Inc.

Tertiary:

Technological University of the Philippines

Bachelor of Science in Information System

PERSONAL QUALIFICATION

- **Soft Skills:** Responsible, Trustworthy, Leadership
- **Hard Skills:** Adobe Premiere; Python; C++; HTML; CSS; JS; Photoshop; GitHub

CYRUS NATHANIEL P. FLORENDO

Doña Cristeta Subdivision, Pamplona Uno, Las Piñas City
0916 296 1558 | cyrusnathaniel.florendo@tup.edu.ph

**PERSONAL INFORMATION**

Date of Birth: April 27, 2003

Place of Birth: Las Piñas City

Citizenship: Filipino

Gender: Male

Civil Status: Single

Interest: Sports, Video Games, Internet, Food

EDUCATIONAL BACKGROUND

Secondary:

Camella School Inc.

Junior High School

University of Perpetual Help System DALTA

Senior High School

Tertiary:

Technological University of the Philippines

Bachelor of Science in Information System

PERSONAL QUALIFICATION

- **Soft Skills:** Time Management; Willingness to Learn; Leadership Skills
- **Hard Skills:** Microsoft Office; Adobe Photoshop; Vegas Pro; C Language; C++ Language; HTML; CSS; Python; MySQL; GitHub

JEWEL VAIDA A. JAIME

4-4B Camara Avenue, Happyville Subdivision, Las Piñas City
0927 025 8355 | jewelvaidaajaime@gmail.com

**PERSONAL INFORMATION**

Date of Birth: October 2, 2002

Place of Birth: Taguig City

Citizenship: Filipino

Gender: Female

Civil Status: Single

Interest: Reading; video games; puzzle; traveling; exercising

EDUCATIONAL BACKGROUND

Primary:

Mayflower Academy of Las Piñas City

Secondary:

Mary Immaculate Parish Special School

Junior High School

University of Perpetual Help System DALTA

Senior High School

Tertiary:

Technological University of the Philippines

Bachelor of Science in Information System

PERSONAL QUALIFICATION

- **Soft Skills:** Time management; flexibility; punctual; reliable; problem-solving; organizational skills
- **Hard Skills:** Microsoft Office; Adobe Photoshop; Adobe Premiere Pro; C Language; C++ Language; HTML; CSS; Python Language; GitHub

AERON JUSTIN QUIAMBAO

3761 Bagong Sikat, Tabang, Plaridel, Bulacan
0933 827 9829 | aeronjustin.quiambao@tup.edu.ph

**PERSONAL INFORMATION**

Date of Birth: September 9, 2002
Place of Birth: Malolos, Bulacan
Citizenship: Filipino
Gender: Male
Civil Status: Single
Interest: Reading; exercising; learning

EDUCATIONAL BACKGROUND

Primary:
 Montessori School Of Malolos
Secondary:
 Immaculate Conception School of Baliuag
Tertiary:
 Technological University of the Philippines
 Bachelor of Science in Information System

PERSONAL QUALIFICATION

-
- **Soft Skills:** Conscientious, Articulate, Time Management; Responsible
 - **Hard Skills:** Microsoft Office; C Language; C++ Language; HTML; CSS; Python Language; Computer Literate

ARSENIC SALANGSANG

1716-C Antonio Rivera St. Tondo, Manila
0921 836 2948 | salangsang99@gmail.com

**PERSONAL INFORMATION**

Date of Birth: October 23, 2003

Place of Birth: Manila

Citizenship: Filipino

Gender: Male

Civil Status: Single

Interest: Reading, Editing

EDUCATIONAL BACKGROUND

Primary:

General Maximino Hizon Elementary School

Secondary:

Cayetano Arellano High School

Tertiary:

Technological University of the Philippines

Bachelor of Science in Information System

PERSONAL QUALIFICATION

-
- **Soft Skills:** Responsible
 - **Hard Skills:** C++ language; C language; Python; HTML; CSS; React JS; Javascript; Github; Adobe Photoshop; Adobe Premiere Pro; Capcut; Adobe Audition; Adobe XD; Figma