RENTAL APARTMENT MONITORING SYSTEM

BY

KUIT YING QIAN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS) BUSINESS INFORMATION SYSTEMS

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: RENTAL APARTMENT MONITORING SYSTEM

Academic Session: JAN 2024

I KUIT YING QIAN

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

(Author's signature)

Verified by,

(Supervisor's signature)

Address:
12A, Lorong Permai Jaya 4.
Taman Permai Jaya,
14000, Bukit Mertajam, Penang.

Dr Abdulkarim Kanaan Jebna
Supervisor's name

Date: 26 April 2024

Date: 26 April 2024

FACULTY/INSTITUTE* OF __BUSINESS INFORMATION SYSTEMS__

UNIVERSITI TUNKU ABDUL RAHMAN

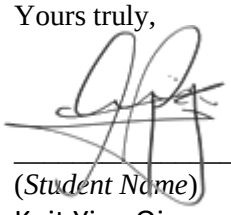
Date: 26 April 2024

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that _____***KUIT YING QIAN*** _____ (ID No: __***20ACB02533*** _) has completed this final year project entitled "*RENTAL APARTMENT MONITORING SYSTEM*" under the supervision of DR. ABDULKARIM KANAAN JEBNA (Supervisor) from the Department of Information Systems, Facultyof Information and Communication Technology , and DR. WONG PEI VOON (Co-Supervisor) from the Department of Digital Economy Technology, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,



_____

(*Student Name*)
Kuit Ying Qian

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**RENTAL APARTMENT MONITORING SYSTEM**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : KUIT YING QIAN

Date : 26 APRIL 2024

ACKNOWLEDGEMENTS

I would like to extend my sincere appreciation to all individuals who have significantly contributed to the successful realization of this project. The consistent support, advice, and contributions provided by you have been of great value throughout this project.

I would like to express my deepest appreciation to Dr. Abdulkarim Kanaan Jebna for his invaluable experience and guidance, which have played a crucial part in the accomplishment of this project. The valuable input and commitment have shown towards my academic development has had a significant influence.

I would want to express my gratitude to my family for their constant belief in my capabilities and their ongoing support. The assistance they received served as a source of inspiration to overcome obstacles and show perseverance to finish this project.

ABSTRACT

There is a lack of a complete monitoring system in the property market in Malaysia, specifically in Kuala Lumpur which is one of the busiest regions to accurately determining apartment values based on several characteristics. This lack of presence leaves users without a reliable tool for making reliable decisions regarding property investments or rents. Moreover, users frequently have difficulties in creating accurate forecasts as a result of insufficient understanding of market dynamics as well as relevant variables that impact flat prices.

Therefore, here comes the need of an interactive dashboard to illustrate and provide a comprehensive view for the peoples. The aim of this project is to create a monitoring system that can generate price predictions for apartments in Kuala Lumpur by analyzing their features. This approach seeks to fill the void in the market by offering customers a simpler visual representation of apartment rental rates, taking into account criteria such as the number of bedrooms, number of bathrooms, location, facilities, and amenities. Through the utilization of this technology, individuals can enhance their property search experience and make well-informed choices regarding property purchases.

This project applies the CRISP-DM (Cross-Industry Standard Process for Data Mining) technique to provide a path to simplify the processing of project data. This technique comprises six steps: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, and Deployment. Every individual stage plays a crucial role in guaranteeing the success of the project and reducing possible risks throughout every phase of its execution. Programming languages like Python is used for tasks such as web scraping, data processing, and visualization. The obtained data is shown using various graphs in order to comprehend the relationships among different variables that impact the apartment rental.

After finishing the project, the monitoring system effectively produces price estimates for apartments in Kuala Lumpur based on their existing attributes. Users could access to a simplified visual illustration of apartment rental, enabling them to make wise decisions regarding property investments or rentals. The technology offers useful insights into the elements that impact flat prices, enabling users to enhance their apartment search process and navigate the property market with greater efficiency.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CRISP-DM	CRoss-Industry Standard Process for Data Mining
HTML	Hyper Text Markup Language
PHP	Personal Home Page Tools (Hypertext Preprocessor)
CSS	Cascading Style Sheets
XGBoost	Extreme Gradient Boosting

Chapter 1 Introduction

The modern rental flat environment is a dynamic and changing arena that is heavily influenced by technological improvements, urbanization, and altering demographics. Based on the research by Malaysia's leading property site, PropertyGuru, the reduced external demand due to the slowdown of Chinese economy has influenced Malaysia's economic growth [1]. Even though the economic growth slowed down, but the average listing prices for properties keep rising along with the optimization of prices by the sellers. Therefore, this scenario has led to the shifting of buyer's interest towards the rental market and from the article, it is expected to persist in the future.

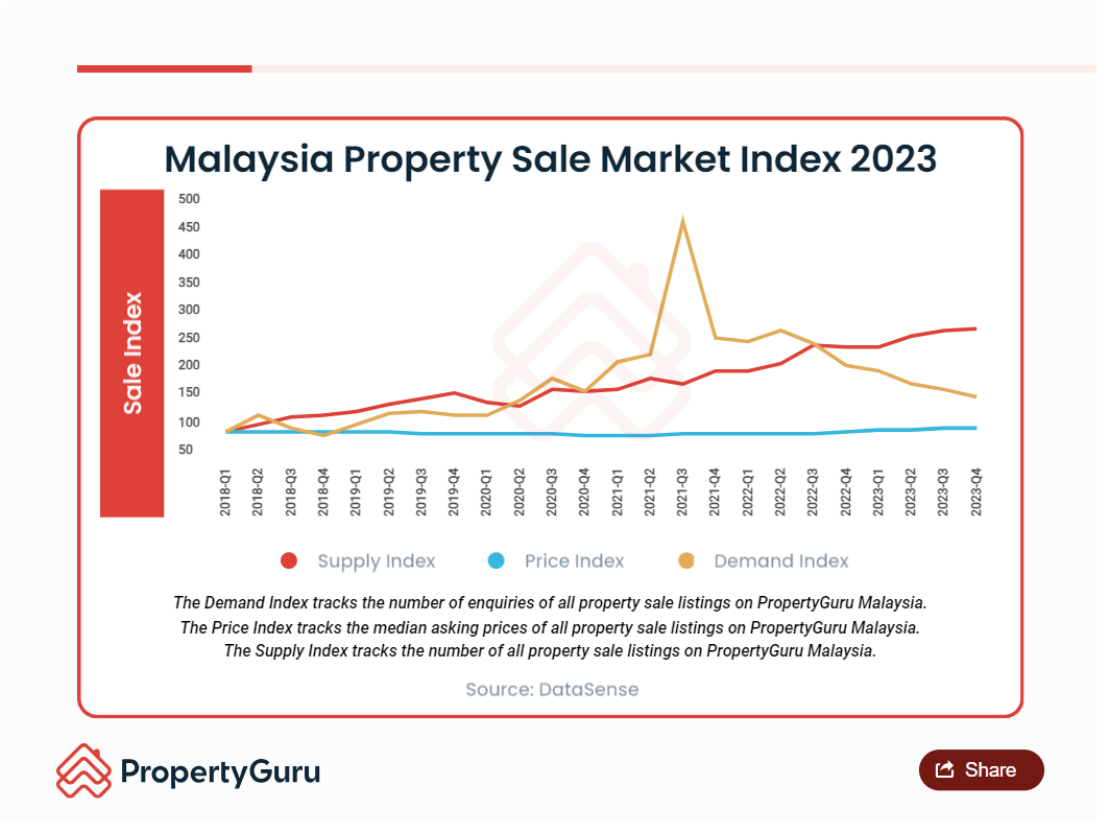


Figure 1 (a) Visualization of Supply, Price and Demand Index

Besides, Business Times News also provided strong support towards the statement where there will be more renters than buyers for the coming few years [2]. According to the survey in PropertyGuru's Consumer Sentiment Study (CSS) for H1 2024 , 29% of existing renters are considering extending their renting period for up to two years

before buying a home. They mentioned reasons such as not having enough money saved to buy a property and the increasing costs of properties [2].

In addition, 48% of existing and potential tenants view the necessity of a 2+1 deposit as challenging, indicating difficulties in achieving reasonable rental prices through negotiation. These findings highlight the necessity of legal actions to address affordability concerns and promote a rental market that is more transparent for prospective tenants[2]. This has strengthened the need of apartment monitoring system for them to analyze and visualize such that they are not getting scammed by the landlords too.

With an increasing number of individuals and families choosing rental apartments as their permanent residence, it is becoming increasingly vital to efficiently monitor and manage rental properties. The purpose of this introduction section is to provide a full background for understanding the existing rental monitoring scenario's issues, their significance, and the proposed solution, the "Rental Apartment Monitoring System" especially for Malaysia.

1.1 Problem Statement and Motivation

Nowadays, technology sectors are improvising every day, especially since the Covid-19 virus outbreak. Society is driving to a technology-focused era, and implementation has been done in countries worldwide. Based on Pew Research Centre, it has been a trend that 96% youngsters in this generation are internet users [3]. Not just to ease youngsters who just graduate to find apartment to be rent, but also to ease every society member who are finding for apartment in a more convenient way where they would not need to collect every detail of the apartment available to be rent which is a waste in time and energy. A website which monitors and categorizes the information besides price of the apartment available for rental is simply perfect for the current society.

The aim of this project is to develop a system that improves renters' experience, as there is **a lack of dashboard systems available to the public in Malaysia**. The primary objective of the dashboard is to offer a unified and user-friendly design, enabling users to conveniently compare different flats and promptly access relevant details about their desired apartment, without the necessity of engaging in extensive reading or conducting research.

Besides, the system's creation is to **efficiently communicate information to users** using concise and engaging text, complemented by visually appealing representations. This approach guarantees the correct and straightforward communication of the essential aspects. Concurrently, the system will possess the capacity to show crucial areas, such as existing amenities and offered property facilities.

As mentioned by Siva Shanker, CEO of Real Estate Agency at Rahim & Co International Sdn Bhd, the majority population end up renting as they cannot afford houses [4]. The earnings and incomes of individuals have failed to match the escalating costs of properties, resulting in an uncontrollable rise in property prices. Therefore, the increased market values of real estate have prompted a significant number of individuals to choose rental solutions instead of engaging in property ownership. In the rental market, individuals have the opportunity to acquire desirable properties at a relatively affordable cost in comparison to the outright purchase of a residential dwelling. The process of moving from one residential unit to another becomes easier when individuals come across more suitable units or when they need to relocate to a different city or state.

Based on the Figure 1.1, it visualizes that renting has taken up most percentage out of the 4 categories, which is 34.56%. The data has proven that people are more preferable to rent properties nowadays instead of purchasing a property. Other than that, it also shows that the people at Klang Valley which represents the combination of Selangor and Kuala Lumpur, that are the focus location of the project, have higher percentage which prefers to stay in rented properties compared to buying property or living with parents or relatives. As most of the renters are people who find jobs there and not from Kuala Lumpur, they would prefer to save their money by renting a house and find a

comparatively convenient location to stay.

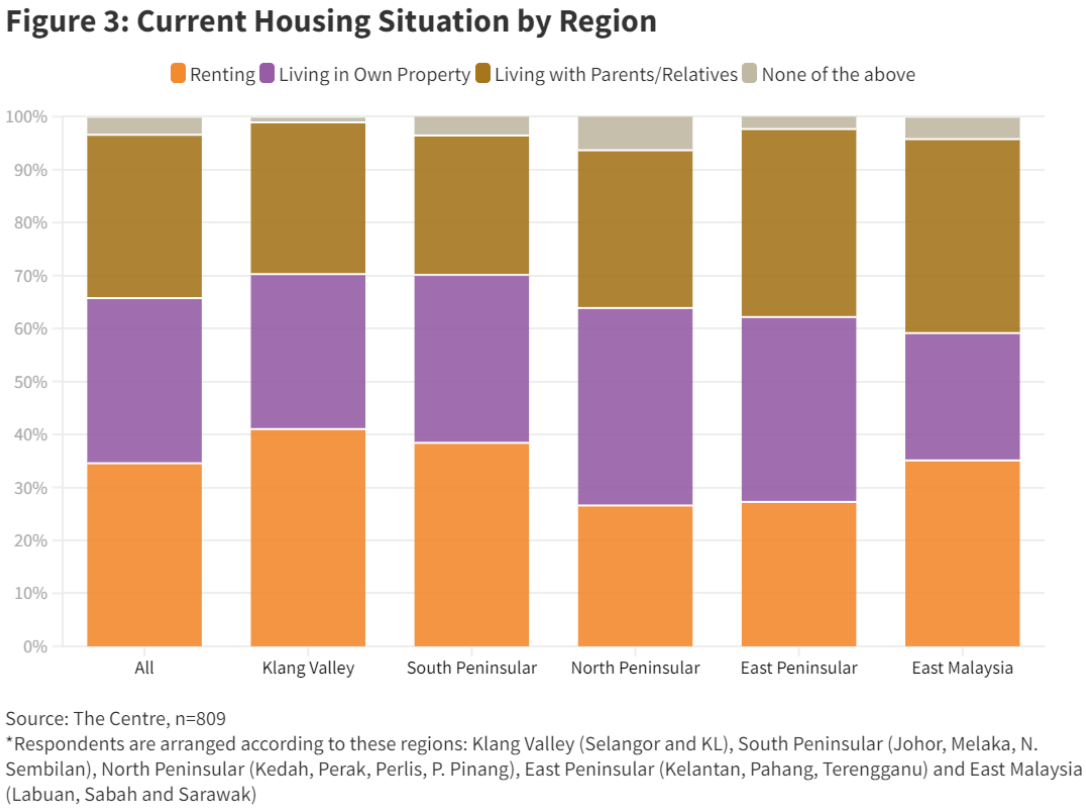


Figure 1.1 Current Housing Situation by Region from The Centre [5]

Thus, it is important to develop a system which could predict and visualize the data of the apartment rental for these populations. Landlord and property investors could also utilize this dashboard to check on the prediction if the landmark will increase in value for the coming years. Landlords could also offer promotion based on the requirement, which is highly required by the renters, or provide any other benefits for the properties which is lower required to improve their rental on properties.

1.2 Objectives

- **Provide prediction of apartment rental through monitoring system**

  A prediction monitoring system which will ease the users on analyzing the apartment functionality and worth could ease the users to obtain information and more accurate predictions of how the development and landscape of the apartment will influence the price, is it worth the price? The system will bring convenience to the users by predicting the apartment's rent such that users would not miss the best price to rent the apartment. Benefits and advantages of the apartment would be

visualized through the graph to ease the information absorption of users, thus reducing the user's time spent on highlighting out the points one-by-one.

Besides having the already existing data to check on the estimated rental of the apartment, user could also input the attributes field themselves and generate a predicted price for that property. They could have a rough overview of the price the property should be, it is rather slightly higher or lower than the estimated price if the environment affecting the apartment price is stable. This helps them to exclude the apartments which are overpriced too.

- **To observe factors influencing the rental prices the most**

As there will be information gathered about the apartments during the project, the visualization of predictions could also show which factor have the strongest bond with the rental and affects the rental prices the most. Investors could find out what is the main focus of the renters and plan their investment on the new apartment. An example for the attributes is existence of barbeque area, sauna, swimming pool, or even the most important facility these days which is the Wi-Fi or internet.

- **Deployment of Dashboard**

By developing the dashboard, the processed and organized data will be presented in a user-friendly format, making it easier for users to interpret and understand. Instead of displaying in calculations such as deviations, or certain professional terms, the dashboard is aimed to visualize in the simplest way which suits to various people with differ demographics. Users of the dashboard could save their time reading through each and every page of the apartment to draw out a conclusion on the average value for apartments in certain regions or areas. This dashboard would be able to support them on this calculation and even provide the best prediction available such that they could be informed on the future possible price if the apartment rental would be rising or not.

- **Identify the location with less renting activities**

As the dashboard would show data influenced by the location, it could also provide information about the location which have more renting activities and less renting activities. Users of the system could focus on the location which have fewer renting activities, thus aiming to further improve the location such that the renting activities

would increase or to maintain the current situation, as some location might be for the residency of the locals.

1.4 Contributions

As the user-centric accessibility dashboard is rarely found, this project aims to provide one and specially aiming on the busiest city consists of many outsiders who are urging to rent a property. Besides, data collection would be done in this project to provide the user data from Malaysia's biggest rental market, which is the Mudah.my website. This approach could provide the users with more relevant data by running the code. The project aims to enhance user accessibility to the prediction function and provide users with comprehensive insights in a clear manner through involving the data scraping to displaying the final predicted values in the dashboard. Users would have the opportunity to acquire knowledge regarding the mean value of real estate in a specific locality, so enabling landlords or property investors to make more accurate estimations of pricing that would optimize mutual benefits. The rental rates could be regulated at a moderate level due to the landlords' desire to engage in rental competition, to ensure their rental prices, remain reasonable.

The development of dashboard plays a crucial role in this project as the dashboard will help users to process information and data rapidly than just using texts. The dashboard serves as the centralized hub which contains and combine the web scraping, data processing, and the display for the data [6]. The design of dashboard aims to be user friendly and easy to function as it was supposed to ease user and quicken the time needed for user to obtain the insights needed.

In order to integrate Rental Apartment Monitoring System more effectively into the comparison of project, it would be advisable to present it in a graphical format, such as a pie chart which visualizes the comparison of factors affecting the rental of apartment easily. The system would be invented in a way that it shows dashboards which not only consist of characters, but also illustrated by charts and graphs. It is believed that humans can translate visual content quicker than any data [7].

While undergoing the data understanding there would be a graph to visualize the relation between the factors collected and the rental. A regression model will be used

to predict the price of the apartments and to produce a prediction graph on top of the Rates and Trends graph to ease users obtain authentic data values.

1.3 Project Scope and Direction

The project tends to study the apartment price around Malaysia's capital, Kuala Lumpur which also includes the connected state, Selangor, and provide predictions on the renting price of the apartments. Factors such as the development of the apartment surroundings, property market economics, and number of rooms and bathrooms will also affect the price of the apartment. This is mainly because those factors are known to be important when renters are planning to rent a property and will be specified out by the landlords as a benefit or add on to their properties such that people will prefer their over others.

Therefore, the project aims to collect the data of apartments around Kuala Lumpur and observe the current market price for the prediction. Once the data is obtained, it is able to check on which factors have the highest impact on the rental price when compared to each other.

Additionally, it would provide estimation on the future trends which is the growth of the rental price of the property. The property's rental estimation could help people to identify if they could afford to rent the property for long term or short term. If they could not afford it, they could find other options through this prediction system and have a basic mental preparation on the state of their renting activities.

There will be statistics visualized such that the users could read on the past trends and compare it with the predictions made by the system to have their own opinion on the condition of the apartment rental. Investors could plan on investing in the property at the place where it seems to be growing.

There would be a function where the system would enable the users to input the attributes data to estimate the property with those characteristics would be priced. This function aims to facilitate the potential property owner who is planning to rent their properties estimates the price they should be pricing out. Through this system, they could ensure the price they are labelled is somewhere around the average price, might as well increase or decrease slightly based on their properties' condition.

1.5 Report Organization

The following chapters will show the research in a more detailed description. Chapter 2 will address the issues and conduct a comparison of the present practices and existing systems. Chapter 3 of the document includes the proposed methodology used, as well as a brief description of the system's design. The only methodology involved is the CRISP-DM methodology which serves as the base for a data science process, and the system would be focusing on prediction where we will be data scraping it first at this stage. Chapter 4 provides an in-depth analysis of the tasks completed and the outcomes achieved throughout the stage, as well as a review of the anticipated deliverables resulting from this project. Lastly, Chapter 5 provides a comprehensive summary of this whole report and highlight the key elements of this project

Chapter 2 Literature Review

This chapter presents an analysis of various systems that contain the necessary functionality or exhibit similarities to the Rental Apartment Monitoring System. This chapter offers an opportunity to explore further into the study of functions that can be established to enhance functionality, hence facilitating a greater knowledge, and providing valuable insights. By systematically evaluating the advantages and disadvantages of the studied systems, it is possible to foster greater innovation and enhance the overall concept of the newly developed system.

2.1Previous Works on Apartment Rental Systems

Among the websites studied, there are a several pros and cons on the currently existing system. The advantages of the existing systems should be kept and implement in the system that are going to be developed. In the meantime, the limitations which appear in the systems should be identified and improvised too such that users can obtain more benefits from the new system, which is the main purpose of a new system being developed.

2.1.1 California Housing Partnership

The California Housing Partnership, also known as the California Affordable Housing Map and Benefits Calculator, is a private non-profit organization which offers a comprehensive compilation of affordable housing projects that have received funding from both state and federal sources. This resource includes an analysis of the anticipated social, economic, and environmental advantages that these developments are expected to bring to their respective local communities. They play a crucial role in the preservation and expansion of the housing supply that is accessible to households with

low incomes in the state of California [8].

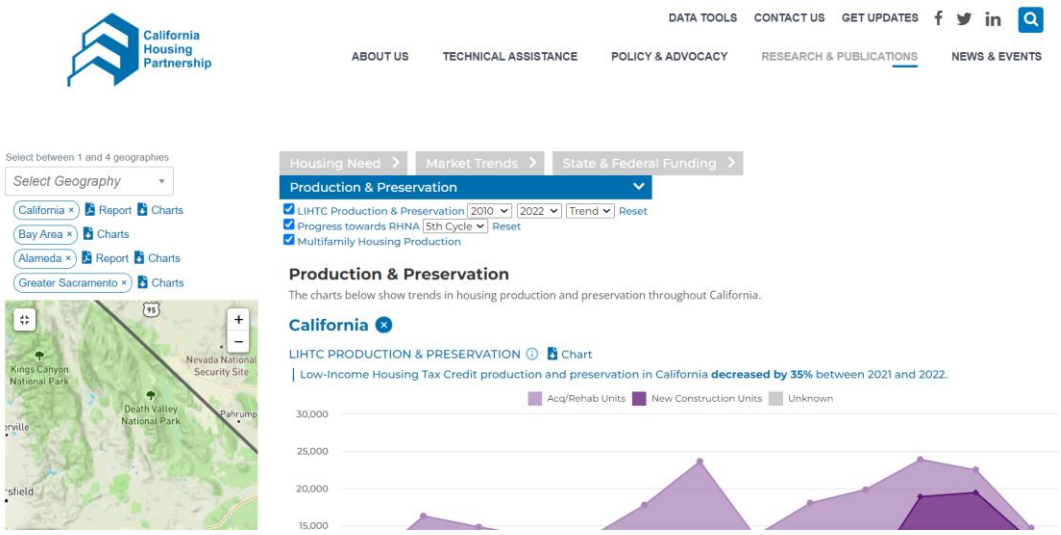


Figure 2.1.1 User Interface of California Housing Partnership Dashboard

California Housing Partnership categorizes their analysis into 4 sections which are Housing Need, Market Trends, State and Federal Funding, and Production and Preservation. Under each section, there are still checkboxes for users to check on what they want to see in the graphs.

Besides, users could also download reports and charts provided based on the selected geographies to understand more about what the charts represent. They provided the user to select up to 4 geographies categorization to show the charts, which included California's regions and counties.

2.1.2 Lancashire County Council

The major objective of the Lancashire County Council is to cater to the needs of all individuals residing and employed within the Lancashire region, with a focus on enhancing well-being, contentment, and overall standards of living. Additionally, they play a vital role in ensuring the well-being of vulnerable individuals within the community who need additional assistance. Although several services offered by the organization are widely recognized and acknowledged, such as their assistance to local businesses and collaboration with educational institutions, several other services may only be discernible to individuals who directly engage with them. In general, the endeavors of the council have a significant influence on the well-being and essential

assistance of all county residents, as seen by their implementation of diverse programs and services [9].

The Lancashire County Council presents all charts in a static position, hence facilitating user experience by eliminating the need for vertical scrolling to view the graphs. Additionally, the charts and graphs were appropriately presented inside the given framework. Users are required to navigate the webpage by utilizing the designated button in order to access subsequent or preceding pages. The table of contents serves as a navigational aid, offering users a concise overview of the content that will be displayed in subsequent pages within the dashboard. Additionally, a red arrow was provided to facilitate the user's navigation back to the main page.

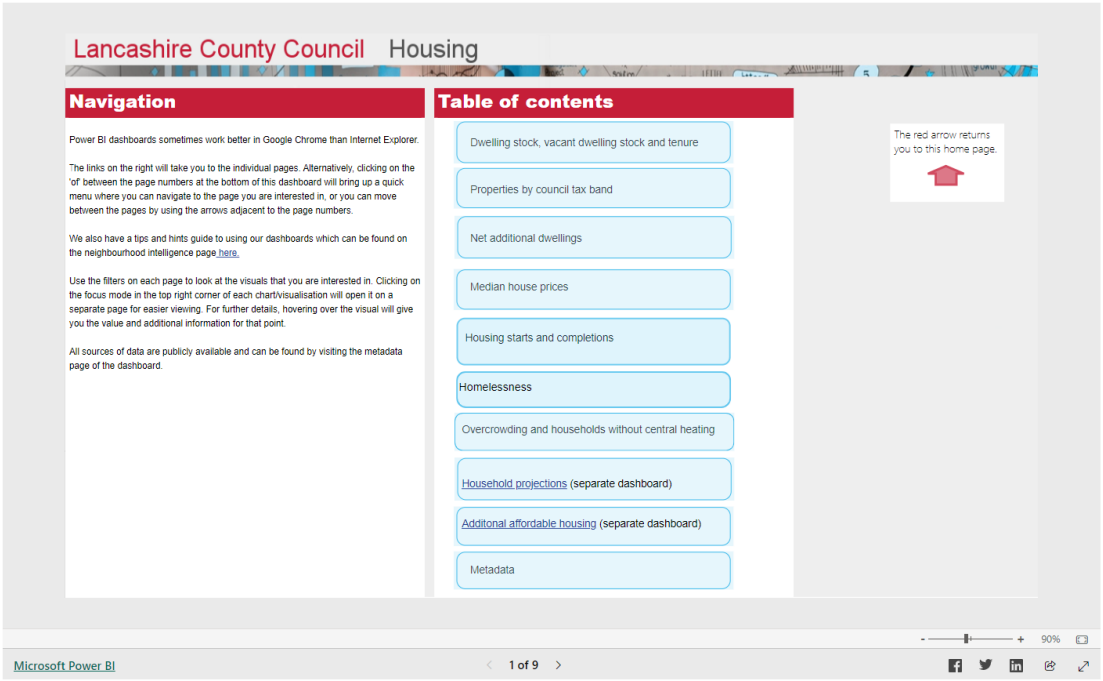


Figure 2.1.2 (A) Housing Dashboard provided by Lancashire County Council

The Lancashire County Council offers a forecast of housing trends for the year 2043, drawing comparisons to the housing landscape of 2018. The category of graphs could be altered, providing users with other options. The categories are conveniently positioned next to the corresponding graphs, enabling ease of comparison for users. Additionally, a pop-out navigation feature was implemented whereby users could access certain pages by clicking on that page's number. This design enhancement facilitated efficient page navigation, eliminating the need for users to repeatedly click in order to reach their desired page.

Figure 2.1.2 (B) Household Projections Dashboard provided by Lancashire County Council

2.1.3 99acres

99acres is an India Real Estate Property Site which provide variety of services related to the rental or purchasing of property to their users. It is a website with very mature functionalities and designs which eases their user in obtaining the information they needed. One of the strengths exist in 99acres is that they provided the "Insights" tab which direct users to another page and there they will **have clearer vision on the details of the property**. Insights help their user to have better understanding on localities, read resident reviews on the property [10].

Besides, 99acres provides also the Locality and Property Insights such that users of the website could obtain the information directly without needing to have research on the map afterwards which is also provided in 99acres. The map in 99acres is personalized by the website and the surrounding area is circle up so users could know what utility is provided near the property they are interested. It helps users to analyse the conveniency for daily and basic needs if they were to rent a property there. 99acres also provides charts to visualize the properties in the locality for their users such that they could have

a quick review. The price range, BHK (bedroom, hall, kitchen) configuration, property type, and posted by categories of the properties in the area would be illustrated.

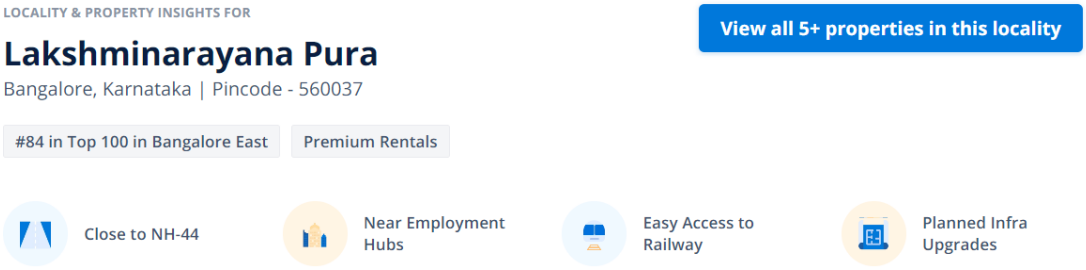


Figure 2.1.3 (A) Locality and Property Insights

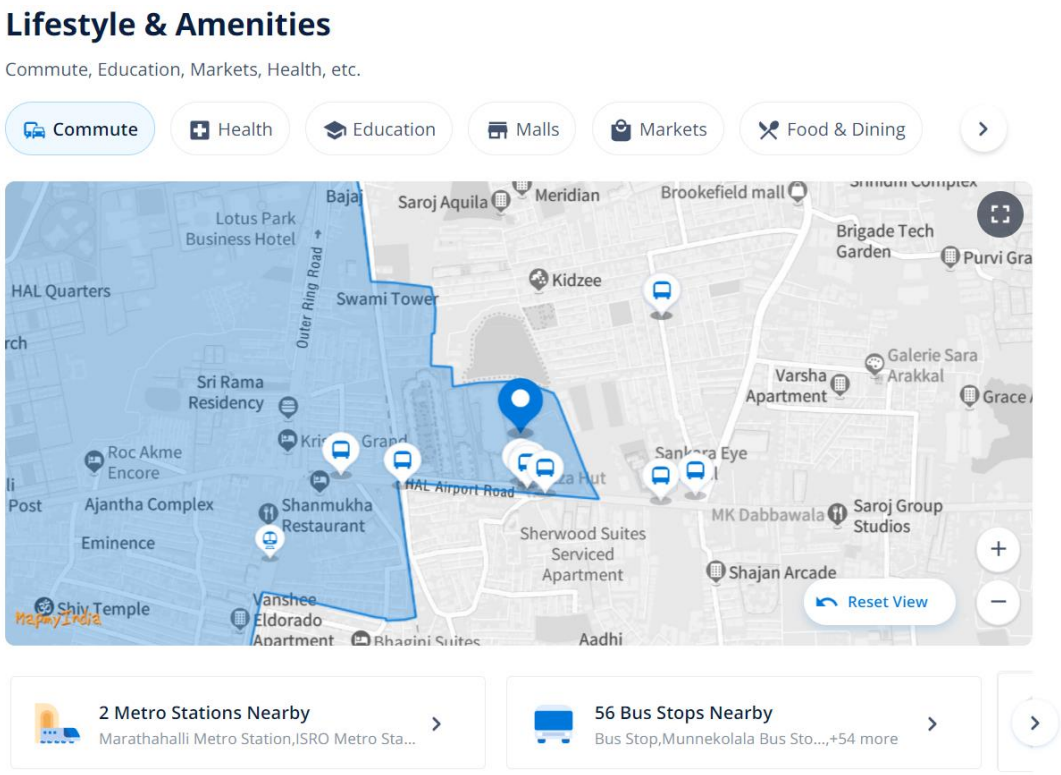


Figure 2.1.3 (B) Life and Amenities

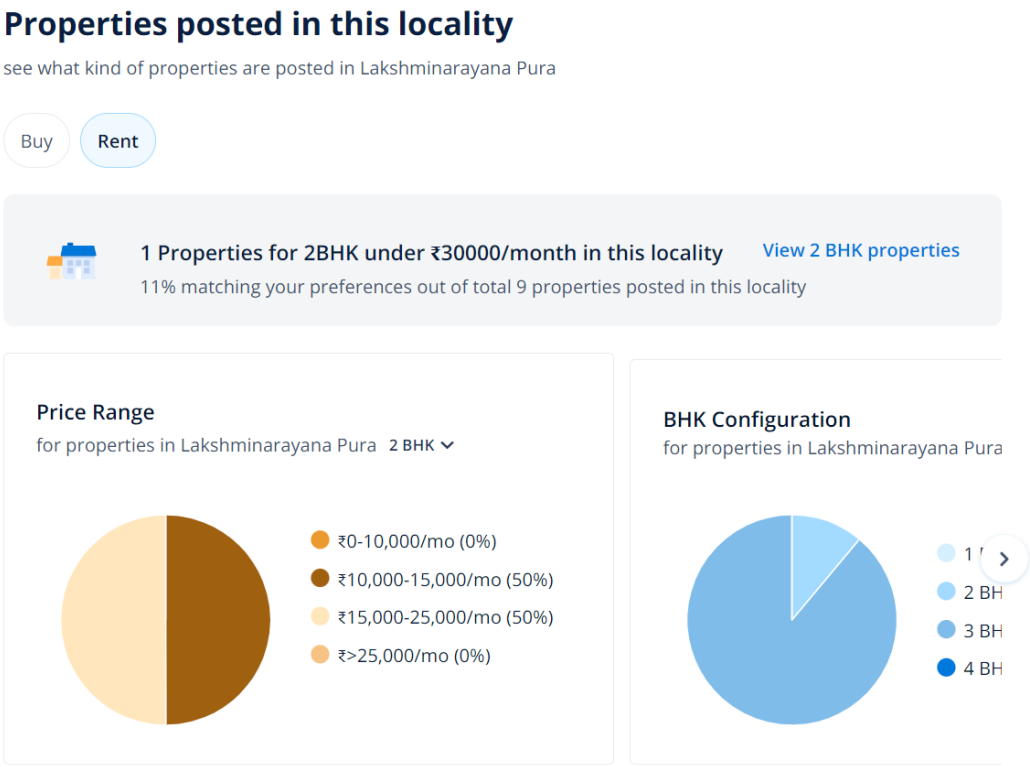


Figure 2.1.3 (C) Properties posted in the selected locality

2.1.4 Mudah.my

Mudah.my is a local Malaysia marketplace which provides platform for sellers in all categories to sell or rent their products, and people to buy and sell new or preloved goods with a simple post [11]. Mudah.my could be said as the best website at Malaysia which provides variety categorization for their users to select on based on their preference. It **provides a detailed requirement categorization at the selection pane** to filter out unwanted properties characteristics and show only the apartment which consists of the selected characteristics (refer to Figure 2.1.4 (B)). Mudah.my provides around 20 filters on the included requirements, from the general categories like location, area, property types to the detailed categories such as minimum and maximum number of bedrooms, minimum and maximum square feet size, monthly rent from and to, build year from and to, and others for the users to filter out the required type of apartment they are looking for.

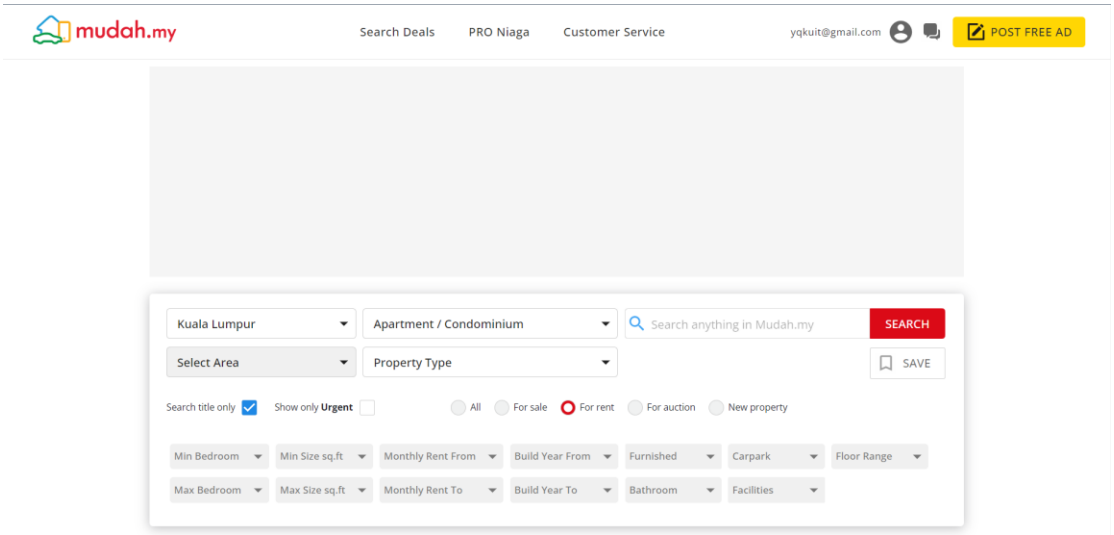


Figure 2.1.4 (A) Mudah.my selection pane

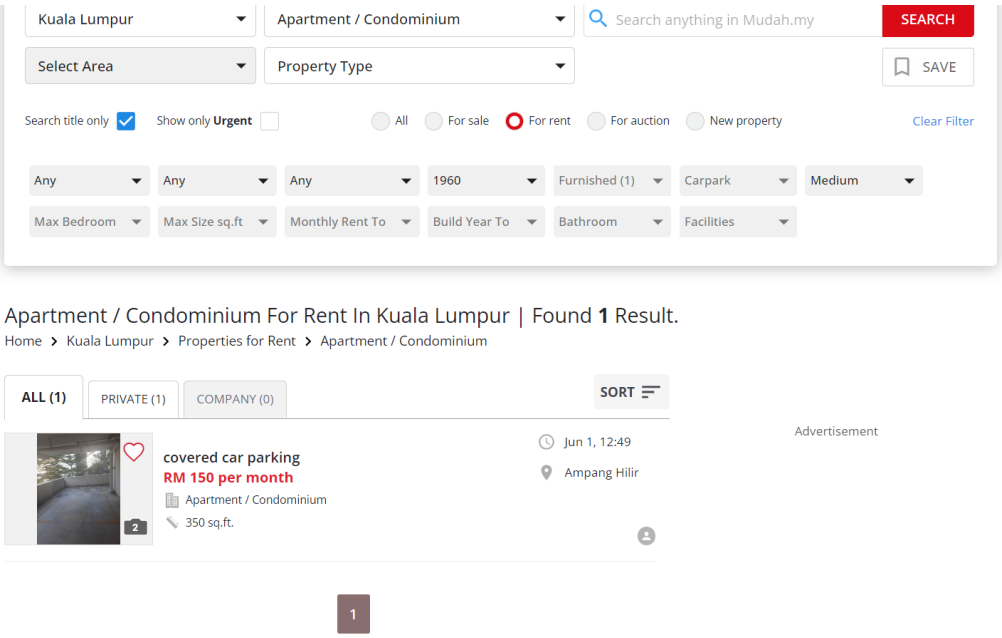


Figure 2.1.4 (B) Example of the filtered properties

The marketplace also provided dashboards for their users. There are two types of users which could be created at this website, Private Account Users as shown in Figure 2.1.4(C) and PRO Niaga Account Users in Figure 2.1.4(D). Each user would be led to a **different dashboard for the two types of users** when they selected on each **"Dashboard" tab**. Private accounts users are for the users who are not sellers, but just wish to check on the property rent or sale. While PRO Niaga Account Users are provided for the sellers to create their account and check on the status of the advertisement of product, and product they are providing.

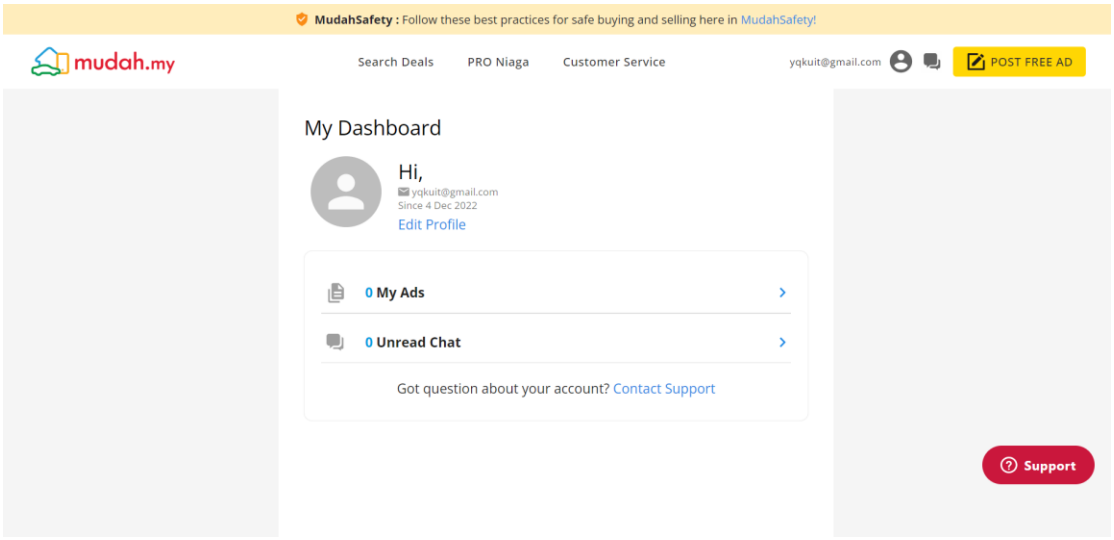


Figure 2.1.4(C) Mudah.my dashboard for Private Account users

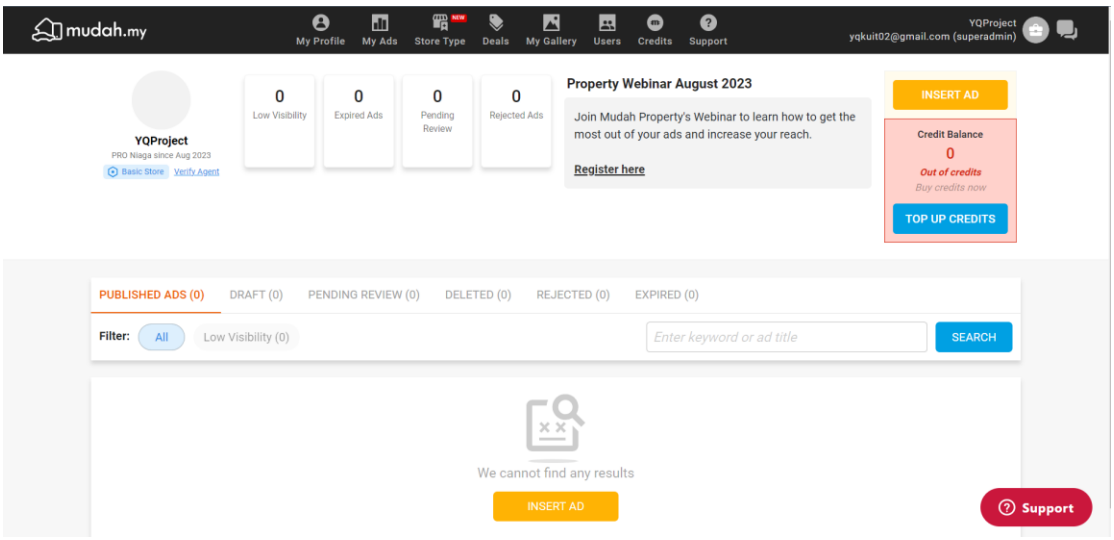


Figure 2.1.4(D) Mudah.my dashboard for PRO Niaga Account users

As shown in Figure 2.1.4(D) The PRO Niaga Account User's version of dashboard is functional and contains more information for the users to check on. Functions to check on the advertisements condition, such as low visibility, expired ads, pending review, and rejected ads are provided by Mudah.my to ease the users, who is the seller of property to follow up on the advertisements' status.

2.2 Strengths and Weakness

Shown below are the two tables comparing the strengths and weaknesses of the systems reviewed, California Housing Partnership, Lancashire County Council, 99arces and Mudah.my.

California Housing Partnership	
Strengths	**Weaknesses**
Provide a list of regions and counties for the purpose of categorization.	A maximum of 4 geographies could be selected only
Reports for all 58 counties are available for download	Charts displayed in a slightly messy view
Provide categorization of Housing Need, Market Trends, State and Federal Funding, and Production and Preservation	No navigation provided to show what data is visualize in the section
Have an information button to select and check the details of the charts	

Table 2.2.1 Strengths and Weaknesses of California Housing Partnership

Strengths

California Housing Partnership provides the list of all regions and counties existing in California such that the users could observe the visualized data based on their selected preferences.

A report in pdf format is available for all 58 counties is available such that the users could read on the details of each graph. There are some regions which they provided excel sheets for users to look at the data more detailly.

California Housing Partnership also categorizes their information properly where the data display was separated into 4 categorizes, which is Housing Need, Market Trends, State and Federal Funding, and Production and Preservation. Furthermore, when user selected the category, it will show the graphs title for user to uncheck the unwanted graphs if they are not interested to them. This could help filter the graphs and make the interface cleaner and ease the focus of users on the graph left.

Not only they provided the selections mentioned above, but they also provided an “i” button for users to click on, and the brief description of the graph would be displayed.

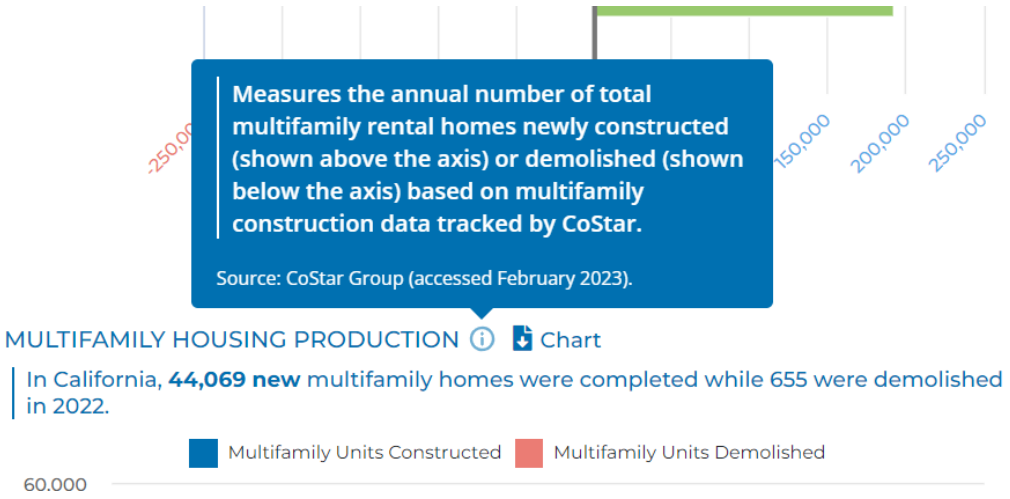


Figure 2.2.1 Information icon beside the graph title

Weaknesses

There is a maximum of 4 geographies could be selected only, which cause the users unavailable to observe the average data of California overall. The users could only get limited information about the comparison and the average value is limited.

Other than that, the charts are displayed in a slightly messy view where there are no obvious borders as a boundary between the graphs. Users have to scroll down the page to have an overview of the graphs based on the filtered information. The graphs could be displayed in a page or a fixed position which provides pages navigation, such that users could focus on a section only to observe information.

California Housing Partnership does not provide navigation to directly link the graph when user clicking the graph title at the selection under the categories. There is no link or buttons for the users to be directed back to the "main page" or the categories section. It is not convenient for the users if they would like to view the data with another filtered information.

Lancashire County Council	
Strengths	**Weaknesses**
Clean interface for user to view	Only available for Lancashire County
Provide information in detail to ease users' understanding	Does not provide comparison of one district to the others

Interactive and easy access for user to navigate to find required information	
Includes the England other than the cities in Lancashire County	

Table 2.2.2 Strengths and Weaknesses of Lancashire County Council

Strengths

Lancashire County Council provides prediction to year 2043 based on the collected data during 2018 or years before. This function has actually helped their users to see the trends and future growth that might be happening provided the environment maintain stable throughout the year.

Lancashire County Council has a clean interface for users to view on. One of the designs done by them is putting all the dashboard into a frame. Users get to understand what the module will be presenting through the brief description before the frame was shown in advance. Then, the data will be presented in graphs or charts based on their related section which then does not cause troubles as the loaded information might not be memorized by users due to human's limited memory.

The information of Lancashire County Council is provided in detail as the graphs and charts were labelled in texts which explains what they were representing clearly and comprehensively. Users could understand what the graph is presenting better through the existence of words instead of having only illustration.

Within the frame, there are several functions for the users to redirect to random pages exist in the module. Users would not need to be clicking repeatedly to check on desired page which is kill time and unnecessary. There would be a popped-up text-frame shown once the user clicks on the pages shown under every dashboard.

Although mentioned in the system's name, it was supposed to be showing data of cities in Lancashire County which is the Northwest of England. However, they included the whole England in certain areas of sections which could be shown, for a more detailed understanding for their users.

Weaknesses

The weaknesses consist in this system is that they are only available for Lancashire, but not for other places. Malaysia does not have a system which has been developed as well

as Lancashire County Council. People who are interested in the other locations could not utilize this system to facilitate their predictions.

Besides, there is a lack of function or module where the users could select a fixed location and compare the picked location to the others. As an example, the user could select Lancaster as their "fixed location", and then the dashboard displays all the other cities in Lancashire such as Blackpool, Burnley and so on to be showing the comparison of average prices.

99arces	
Strengths	**Weaknesses**
"Insights" tab provided, direct users to another page for clearer vision on details	Show property rates through zone category only
Locality and Property Insights	Does not have comparison of current property to the other in the same area
Provides charts to visualize comparison of current property to the others in the area	Does not show prediction of price trends in the future

Table 2.2.3 Strengths and Weaknesses of 99arces

Strengths

99arces provided "Insights" tab which direct the users to another page to view on more details about the properties they are interested in. Users could check on the trends and rates of certain area and understand more about the property they are about to rent.

Besides, there are also locality and property insights which visualize the present localities around the apartment such that users get to know visualize the surroundings without needing to visit the location physically.

Furthermore, 99arces includes charts which for different kinds of feature compared by current property to the others in the same area. This is a bonus for the system as it visualizes the information clearly and makes the users understand the apartment and obtain these benefits provided by just scrolling the page down.

Weaknesses

99arces provided the great function of property rates, but the property rates only categorize rates by zone category only. I believe that it would be better if 99arces also provides the rates of the bedroom number, bathroom number, or any included facilities or necessities. As so, the users would be able to see the average rental of properties based on the factors they are more focusing on. On the other words, provides more personalized system to the users.

The system also does not provide comparison function for the users to check on the rental of certain property compared to the others which have the same characteristics in the same area. By having this function, users could understand better if the property's rental is reasonable or not; if they are the landlords, they could check on how to promote their properties with the prices slightly lower from the average.

99arces does not provide the prediction function although it seems to be providing a lot other conveniency to their users. If there were a predicting dashboard provided, users could have viewed the future trends and predicted their investment or rental based on the charts. Providing an assumption to illustrate how prediction would help: If the rental were going to have a high raise by the following year, the renters could have considered if they could afford the raise in the rental, or to find a new apartment to rent for. This prevents the renters from having trouble to pay the rentals or having nowhere to live in a sudden.

Mudah.my	
Strengths	**Weaknesses**
Provides detailed requirement categorization at the selection pane	Only provides dashboard for sellers to observe their advertisements status
Provide two types of user accounts with different user interface	No estimated prediction of apartment price in future
Provides featured label to certain properties	

Table 2.2.4 Strengths and Weaknesses of Mudah.my

Strengths

As mentioned in 2.1.4, Mudah.my provides a very detailed requirement categorization at the selection pane for their users. It is obvious and easy to access as it's literally the first thing users see once the page is load. The requirement categorization almost included every single property that could be manipulated or chosen to the users.

Mudah.my is very considerate as Mudah.my provided two types of account types for their users to logon, Private Account User and PRO Niaga Account User. This action makes an impact on users who are the sellers or renters, as the dashboard functions are different for both user accounts. Dashboard plays an important role in visualizing the collected data for the users to understand the status of their advertisements.

Other than that, Mudah.my provides featured label for certain properties once the users paid for it. Featured labels would be prioritized and shown on the first row of the page before any other property.

Weaknesses

The weaknesses existed in Mudah.my is they only provides dashboard for sellers to observe their advertisements status. For PRO Niaga Account Users, Account users could not follow up on the statistic of their properties sell, which means landlords could not have a view on the requirement of the renters, what is the average price for the properties similar to them. For Private Account Users, they could not have the chance to have an overview on the average rental for certain areas.

There is also no prediction on the rental for the properties such that the renters could have a preview on the rental prices for the consequent years. Renters could plan their spendings ahead or to seek alternative housing options in advance of potential difficulties while paying monthly rentals.

Chapter 3 System Methodology/Approach

The project implements the CRISP-DM methodology, which includes phases such as business understanding, data understanding, data preparation, collection, and pre-processing. This systematic methodology guarantees an organized review and application of data to effectively achieve project goals.

3.1 Design Specifications

The project is implementing the CRISP-DM approach which consists of phases including Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment.

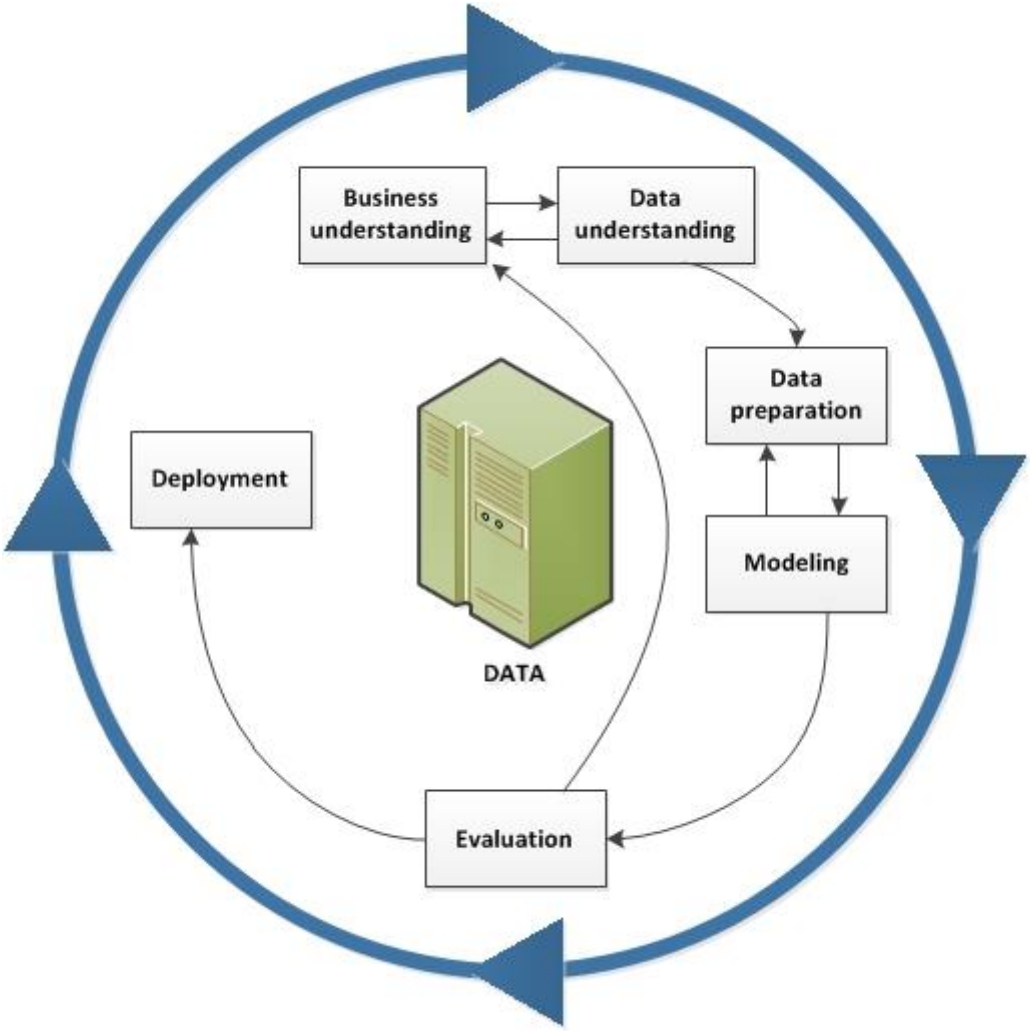


Figure 3.1 CRISP-DM data mining life cycle

Cross-Industry Standard Process for Data Mining, or so-called CRISP-DM, is a methodology which includes the describe the 6 phases, Business Understanding, Data Understanding, Data

Preparation, Modeling, Evaluation, and Deployment. CRISP-DM is also a process model that provides an overview of the data mining life cycle [12]. CRISP-DM also tends to focus on data exploration and visualization and allows users to create a personalized data mining model which suits their systems [12]. It is the most common methodology for data mining and was published to standardize data mining processes in the industries [13]. CRISP-DM was selected to be the methodology used in this project due to the monitoring system would be needing to retrieve the data from existing websites and make comparisons among those data retrieved.

Below is the progress carried out or has been made during this project.

3.1.1 Business Understanding

Business Understanding phase requires the understanding of objectives and requirements of the project [14]. The goals gained from the data mining should be identified by utilizing CRISP-DM, the background information should be collected, and the objectives should be listed down [14].

As for the whole project, the objective is to develop an analysis dashboard which can provide predictions and to show the users detailed trend of the apartment. The dashboard is very much needed to be developed as there are lacking useable dashboard in the Malaysia property marketing area , where people might not know that their rental prediction could be much more accurate with the help of this Rental Apartment Monitoring system. With the presence of the Rental Apartment Monitoring System, users could easily identify the acceptable rental price range of their desired properties. The selected technologies and tools to develop this system would be Visual Studio Code which could access several types of programming languages needed such as Python, HTML, CSS, PHP and so on.

The software used in this project are Jupyter Lab 2.7.0 to present data obtaining and data preprocessing process. Jupyter Lab is selected to carry out these data handling processes as they have a simple user interface and install needed libraries easier by just calling “pip install”. Besides, the codes in Jupyter Lab could be separated into several division, which ease us to run the needed code snippet in the particular division without needing to rerun the whole code.

As for the display of dashboard, Visual Studio Code is used as it could support Shiny app in python. Shiny app in python is used as they could easily retrieve the data or model saved

through the data processing process previously. Uniformed coding language would easy the transmission of data between each file and easier when there is error detected or to be fixed.

3.1.2 Data Understanding

Under the Data Understanding phase, scraped data gets a closer look to avoid unexpected problems during the data preparation [15]. Major data needed to show the trends and as significant information to strengthen the prediction on the trends will be collected by this stage. Web scraping would be conducted in this project to collect data for the system to carry out the following processes.

In this project, Beautiful Soup has been used to extract the data from the web source. Beautiful Soup functions to scrape the webpages and properties information easily without needing to download or save the driver of the web application. The bedroom numbers, bathroom numbers, parking lot provided, floor range and more were collected though this approach and then stored in a csv file.

After the extraction of data from the web source, the relationship of the data is visualized to check if it is necessary or not. The data would be remained mostly such that it could provide the user more insight and selection when they are viewing it from the dashboard.

Data like facilities and additional facilities of properties will be removed as they were already separated into more detailed column to check the existing of facilities provided. Only the attributes which are related and could affect the rental prices would be included in this project. The null values are removed in this process and a visualization of the data is presented to facilitate the comparison of the relationship between the attributes.

3.1.3 Data Preparation

On this phase, values that are missing, outliers or anomalies, improperly formatted data, invalid data, or difficult processed data would be formatted [16]. The data would be identified in this phase if it is needed or should be removed, such as containing null values. The invalid values in data set would cause inaccurate or misleading outcomes as some of the formulas such as mean, and mode will be reliable on and infected by the values. The time and effort used on this phase would be consumable, so it is important to end the previous phases such that the time and effort needed would be minimized. Jupyter Lab would be used in this phase to assist the filtering of the data.

Once the data is being processed and cleaned, it was divided into two sets of data, which is the train set and test set. Train set aims to train the model for the prediction, while test set is used after the fine tuning of the model. Both sets exist with the aim of training and provide a better model with the highest accuracy possible.

The action carried out could be replacing null values with mean, median, or mode for either numerical attribute or categorical attribute. Besides, the data would undergo the visualization to check the relationship between each existing attributes and define the outliers which should be removed in the pipeline before training the models.

Pipeline has been done for this project to assemble the process that will be carried out and ease the process flow. The replacement of blank or null data is done easily by using pipeline where it could be defined to be replacing by mode, mean, or median for numerical attributes, while for categorical attributes could apply the Simple Imputer to select the most frequent data as a replacement. There are new attributes trained to find new attributes which could present the data relationships better in the graphs.

3.1.4 Modeling

Modelling techniques that will be used would be selected in this phase based on the target data, and the modelling techniques would make modelling assumptions about that data set [17]. There are two types of models, Categorical Models and Regression Models which will need to be identified before selecting which models to be used and tested. The selected modelling in this project is Regression Model as the data targeted the rental price. Regression Models could be used to observe the changes and predictions more accurately compared to classification models in this project. Multiple regression models would be carried out to train for the best model which has the lowest root mean squared error, and the cross validation would be carried out. The model will be ensured to be the optimal model through cross validation, not overfitted nor underfitted which will highly impact the accuracy of the results. Fine tuning would be carried out at this stage after to ensure the final model selected was the most accurate model among all.

There are several regression models used to test out the best model which could provide the prediction most accurately. The models used are as follows, Linear Regression, Decision Tree Regressor, Random Forest Regressor, Ridge Regression, SGD Regressor, XGBoost Regression, Light GBM Regression, K-Nearest Neighbor (KNN) Regression, Support Vector Machine (SVM), Neural Network, Polynomial Regression, and Lasso Regression.

Out of these 12 models tested out, 5 better models which were defined based on their Mean Percentage Error were brought out to continue the fine-tuning. XGBoost Regression, Light GBM Regression, K-Nearest Neighbor (KNN) Regression, Neural Network, and Polynomial Regression were selected based on their Mean Percentage Error, but not Root-Mean-Squared-Error (RMSE) or Mean-Squared Error (MSE) as the gaps between the rental prices between each apartment is far and it is not accurate to be used for judgement.

3.1.5 Evaluation

In the evaluation phase, results of the previous phase should be evaluated to ensure the results obtained are useful and accurate. There are two types of results that would be obtained in this phase, which is the final models chosen from the previous phase and findings, which is conclusions or inferences drawn from the model or during the data mining process [18].

As the scraped data contains sufficient amount of data to support the training, it is then proceeded to select the best model by undergoing the cross-validation process which mitigates the risk of overfitting and provide a more accurate estimation of the unseen data. Based on the model results obtained, it shows that the model performance is acceptable as the model output is steady. It was then test with test set to ensure the availability of this model and ensure that this is the perfect model which does not overfit nor underfit the data.

3.1.6 Deployment

The deployment phase of CRISP-DM consists of two activities, planning and monitoring the deployment of results and the wrapping up of tasks [19]. A project review and overview would be done to produce a final report. A dashboard which could provide predictions and display the data values analyzed would be the result of this project. The accuracy of the predictions would be mandatory data to be ensured so it produces a reliable prediction for the user of the system. The deployment of this project to display a dashboard by using Shiny app might be facing some obstacles. Fortunately, the obstacle was then solved by realizing the correct method to call the system. The dashboard could display the predicted value and current value which ease the comparison of apartment for the user. Users are able to input their data in the dashboard for personalized predictions. After all, the deployment of the apartment is quite a success, but the accuracy of the personalized predicted data could be improved to provided more reliable insights to the users.

3.2 Project Timeline

Activity	Period (Week)													
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Coding														
Chapter 1														
Problem Statement and Motivation														
Research Objective														
Project Scope and Direction														
Contributions														
Report Organization														
Chapter 2														
Previous Works on Apartment Rental System														
Strength and weaknesses														
Chapter 3														
Design Specifications														
System Design Diagram														
Chapter 4														
System Components Specifications														
System Components Interaction Operations														
Chapter 5														
Hardware Setup														
Software Setup														
Machine Learning Models														
System Operation														
Implementation Issues and Challenges														
Chapter 6														
Objectives Evaluations														
Chapter 7														
Conclusion														
Recommendation														

Figure 3.2 Project Timeline

Chapter 4 System Design

The apartment rental prediction system combines machine learning algorithms and data analytics to accurately estimate rental pricing and occupancy patterns in a certain area. The system gathers and analyses many data points, such as past rental prices, location, property variables, and additional facilities. Using advanced algorithms, it examines this data to detect trends and correlations that impact rental costs and demand. The algorithm utilizes user inputs, including desired location, budget, and property preferences, to customize predictions according to individual requirements. Moreover, it uses real-time data inputs to consistently update its models and adjust to evolving market conditions. The system's output offers users precise forecasts of rental pricing and occupancy rates, empowering them to make well-informed choices when renting or leasing units. Moreover, the technology provides landlords and property managers with valuable data and recommendations to enhance pricing tactics and maximize rental income. The flat rental prediction system provides renters and property owners with essential information to effectively navigate the ever-changing real estate market.

4.1 System Components Specifications

Data Ingestion:

The data used in this project is collected from Malaysia's biggest marketplace, Mudah.my. As they limited the access towards the page navigation in the scraped data, the final page which is needed to lead the code such that there is a maximum limit of pages needed to be scraped were then found through alternating the "current_page" in the webpage's link. Besides, it is also found that the maximum limit page could be shown for the website is 250 pages, which ease the limitations of pages to access in the code. The "headers" is included which define the headers used in the HTTP request such that the user-agent string is included to mimic a web browser. Thus, prevent the request sent being blocked by the Mudah.my website.

```
# Start Loop to visit each page
current_page = 1
linkRegions = ["kuala-lumpur", "selangor"]
# linkRegions = ["kuala-lumpur"]



while current_page <=250:
    print("Current Page: ", current_page)

    for linkRegion in linkRegions:
        print(linkRegion)
        headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.
        mudahUrl = f"https://www.mudah.my/{linkRegion}/apartment-condominium-for-rent?o={current_page}"
#         print(mudahUrl)
```

Figure 4.1.1 (a) Code to access webpage's link

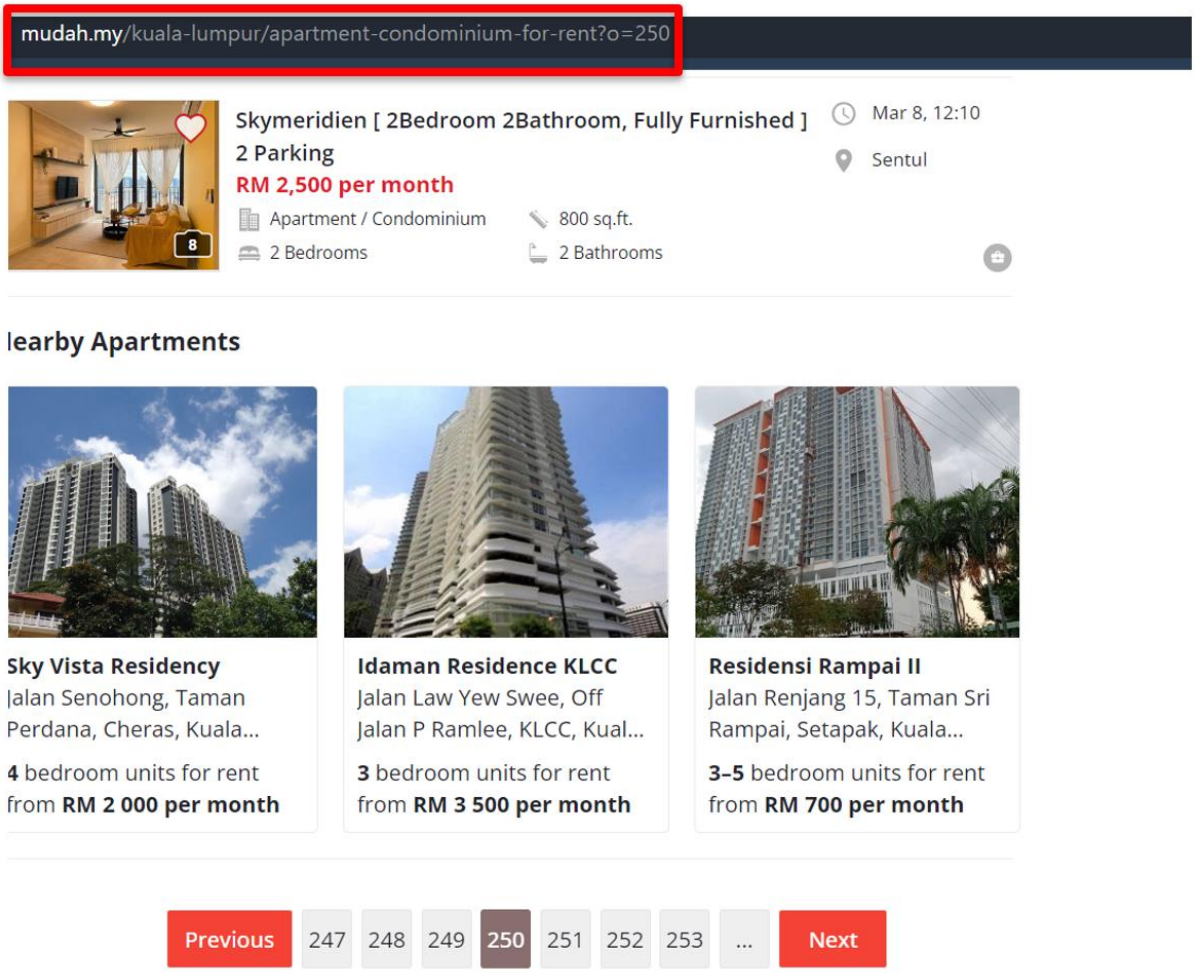


Figure 4.1.1 (b)(i) During page is 250

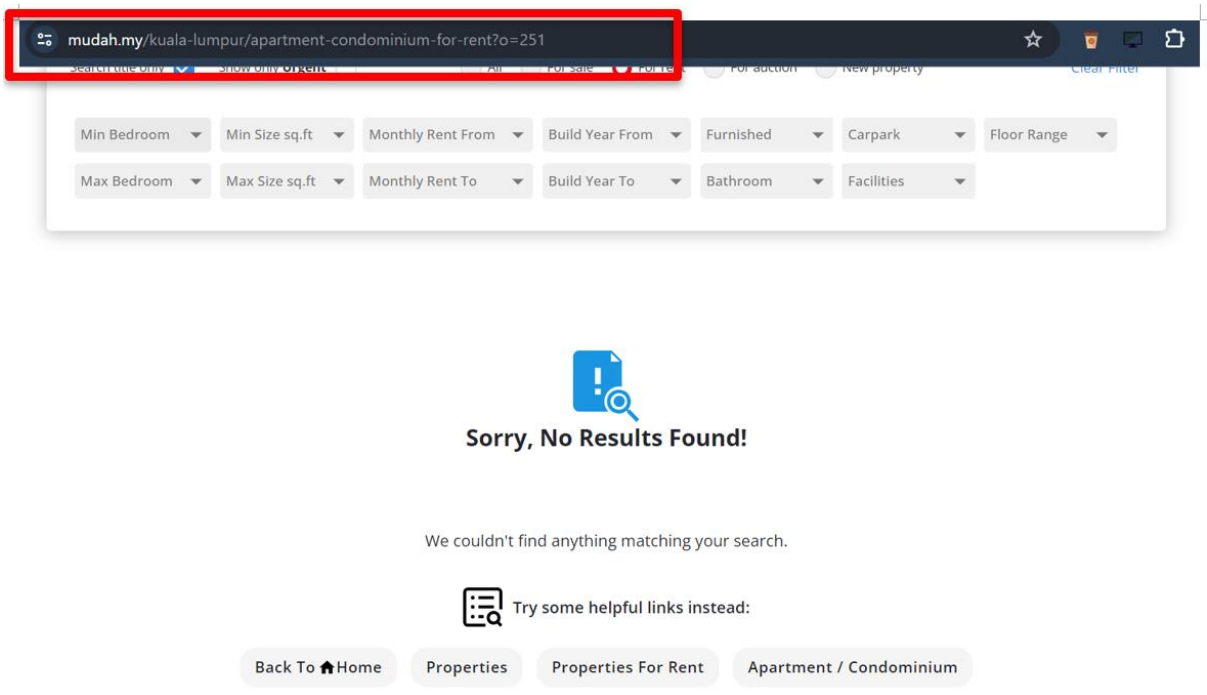


Figure 4.1.1 (b)(ii) During page is 251

Next the “mudahRequest” is to get into the link specified above, and “soup” is to occur the contents in the link, which is also the main page we are login in to view available apartments under the filter we set, Kuala Lumpur and Selangor region apartments from page 1 to page 250.

```
mudahRequest = requests.get(mudahUrl,headers=headers)
  print("mR:",mudahRequest)

soup = BeautifulSoup(mudahRequest.content, "html.parser")
```

Figure 4.1.1 (c)(i) Access to main page

As shown in Figure 4.1 (c)(ii) the links to access each apartment are obtained through calling all the anchored sections as this website’s division ID is different from the inspection on the website and not fixed after the web scraping. Through this action, we could prevent the changes in the ID or naming of sections which store the links. Based on observation, the links are mostly stored under the “href” type data and ended with a “.htm” format. By carrying out this filtering, it could reduce the unnecessary links which are not related and minimize the scope of links obtained.

```
# Find all <a> tags
links = soup.find_all("a")

# Set to store unique links
unique_links = set()

# Extract href attribute from each <a> tag, filter and remove duplicates
for link in links:
    href = link.get("href")
    if href and href.endswith("htm") and href not in unique_links:
        unique_links.add(href)
```

Figure 4.1.1 (c)(ii) Obtain link to apartments page

The apartment links are then filtered out by using unique links as there were duplications of each links in the page. Therefore, unique links are filtered and saved into a list named “apartmentLinks”. Next, the apartment webpage is access through sending request as how we previously managed to access the main page. Through this process, it is realized that time sleep

is needed for each link to pass through and request access. 5 seconds is the best time after trying several times, as if there are lesser time in between the next request, the website would block the request and return error code 429. Although it is checked that 5 seconds is the minimal time required in between the access to mimic human clicking speed, but to prevent any error occurs during the scraping of large data for 500 pages in total, a if loop is created. The if loop is to limit such that only apartment request (aparReq) equals to 200 would proceed to the next stage, else it would be ignored.

If the apartment request returns status code 200, it would proceed to collect the contents of the page in text. The required attributes are stored in script with id "__NEXT_DATA__", so the code would find the section and continue the web scraping of the required attribute.

```
apartmentLinks= []

for hLink in unique_links:
    apartmentLinks.append(hLink)

  print(apartmentLinks)

for apartment_link in apartmentLinks:
    apartReq = requests.get(apartment_link, headers=headers)
    time.sleep(5)
    print(apartReq)

    if apartReq.status_code == 200:
        #elememts in page
        apartPage = BeautifulSoup(apartReq.text, "html.parser")

        apartEle = apartPage.find('script', id = '__NEXT_DATA__')
```

Figure 4.1.1 (c)(iii) Access to each apartment page to view detail of particular apartment

The apartEle which contains the data of the whole <script id= '__NEXT_DATA__'> section is the processed by extracting the text content from the HTML tag containing JSON-formatted data representing information about the apartment details. The extracted JSON format content will then be parses into a Python dictionary by using the "json.loads()" function. The following step is then related to the extraction of the data that would be filled in the attributes.

```
# # Extract text content from the Tag
apart_text = apartEle.text
# category_params = apartEle.get("categoryParams")

scriptIDjson = json.loads(apart_text)
# category_params_ = category_params.find("categoryParams")
```

Figure 4.1.1 (d)(i) Extract JSON format content into Python Dictionary

The "getID" is to obtain the ID for each apartment to access the attributes which consists of the value of needed data for our excel. As the monthly rent and other required attributes are saved under different keys, so the attributes which saved all the data are assigned as "getAttributes" to obtain quicker and easier access.

The monthly rent is obtained by calling categoryParams key in "get Attributes". The [0] indicates that it was in the first place of the list of categoryParams and ['value'] calls out the key which contains the monthly rent. It was nested under several dictionary and list, therefore the process of obtaining is slightly lengthy and requires detailed calling of values step-by-step instead of obtaining the data through calling divisions.

The other attributes are observed to be store under another dictionary key called the propertyParams. Thus, to obtain the attributes exist, the getPropertyDetails is called and load every existing dictionary in it to acquire their value in the "params" key. The "value" and "label" key are called such that the data stored in this both key would be selected and append to the "data" list and "label" list assigned previously. Through this, we could obtain every value and their label stored in the same index. When we call the "label", the "value" would be obtained without needing to count the index or which place it is stored if any problem existed. It could be called and track back easily in a human understandable way instead of computer indexes.

```
getID = scriptIDjson["props"]["initialState"]["adDetails"]["byID"]

for i in getID:
    IDequals= i

getAttributes = scriptIDjson["props"]["initialState"]["adDetails"]["byID"][IDequals]["attributes"]


# Monthly Rent
getMonthlyRent = getAttributes["categoryParams"][0]["value"]

#Others
getPropertyDetails = getAttributes["propertyParams"]

data=[]
labels=[]
for x in getPropertyDetails:
    for y in x["params"]:
        data.append(y["value"])
        labels.append(y["label"])
```

Figure 4.1.1 (d)(ii) Extract required attributes data

Figure 4.1.1 (d)(iii) Example of how attributes are stored in propertyParams

The obtained values would then be stored in assigned variables. As an example, “Building name” which is the label for the building name as saved previously would be called and the value stored by the same index as Building Name would be called out and saved into “prop_name” variables for easier implementation afterwards. The “address” data requires more processes as they were in a line and not all of it is useful.

Thus, the address was split by the comma symbol, and it is realized that some addresses would have been filled in with wrong format. The for-loop is to capture the value as it is known that the only two regions related to this project so if the first “x” obtained is neither “Kuala Lumpur”

or “Selangor” region, then it would be assumed as area where the next data stored in the split_address would be region.

```
address =data[labels.index("Address")]
split_address = address.split(", ")[-2:]

prop_name = data[labels.index("Building Name")]
completion_year = data[labels.index("Completion Year")]
monthly_rent = getMonthlyRent
property_type = data[labels.index("Property Type")]
rooms = data[labels.index("Bedroom")]
parking = data[labels.index("Parking Lot")]
bathroom = data[labels.index("Bathroom")]
size = data[labels.index("Property Size")]
furnished = data[labels.index("Furnishing")]
facilities = data[labels.index("Facilities")]
additional_facilities = data[labels.index("Amenities")]

for x in split_address:
        print(x)
    if x == "Kuala Lumpur" or x == "Selangor":
        region = x
    else:
        area = x
```

Figure 4.1.1 (d)(iv) Assign variables for data to be stored

The variables will then be called to be append into dictionary assigned before the start of the code and after the whole loop which extract the 250 pages each for Kuala Lumpur and Selangor region, they would all then be saved into the csv file. After each page ends the loop by storing all the data acquired into the dictionary, the current_page will add on one such that the loop could proceed to scrape the next page.

```
            mudahDataDict["apartment_link"].append(apartment_link)
            mudahDataDict["prop_name"].append(prop_name)
            mudahDataDict["completion_year"].append(completion_year)
            mudahDataDict["monthly_rent"].append(monthly_rent)
            mudahDataDict["area"].append(area)
            mudahDataDict["property_type"].append(property_type)
            mudahDataDict["rooms"].append(rooms)
            mudahDataDict["parking"].append(parking)
            mudahDataDict["bathroom"].append(bathroom)
            mudahDataDict["size"].append(size)
            mudahDataDict["furnished"].append(furnished)
            mudahDataDict["facilities"].append(facilities)
            mudahDataDict["additional_facilities"].append(additional_facilities)
            mudahDataDict["region"].append(region)

        else:
            print("Unable to load")

current_page+=1
print(current_page)
```

Figure 4.1.1 (e) Append the extracted details to relative attributes

```
mudahDataDict = {
    "prop_name":[],
    "completion_year":[],
    "monthly_rent":[],
    "area":[],
    "property_type":[],
    "rooms":[],
    "parking":[],
    "bathroom":[],
    "size":[],
    "furnished":[],
    "facilities":[],
    "additional_facilities":[],
    "region":[],
    "apartment_link":[]
}
```

Figure 4.1.1 (f) Dictionary to set and store the extracted attributes

```
df = pd.DataFrame(mudahDataDict)
df.to_csv(r"C:\Users\yqkui\Universiti Tunku Abdul Rahman\UTAR - FYP - Oct22-IIPSPW-Rental Apartment Mon Sys-(YING QIAN) - Oct
```

Figure 4.1.1 (g) Define the location to store the dictionary into required csv file

Data Preparation:

For the data preparation it could be divided into two parts where the first part is regarding to cleaning and extracting necessary data information from the scrapped data in excel file, and the second part is related to data processing and model training.

First Part

After analyzing the data which has been scrapped from the website, it is realized that the main problem here is the monthly rent which is the main objective we are focusing on contains string and is not in required format.

	ads_id	prop_name	completion_year	monthly_rent	location	property_type	rooms	parking	bathroom	size	furnished	facilities	additional_facil
0	100323185	The Hipster @ Taman Desa	2022.0	RM 4 200 per month	Kuala Lumpur - Taman Desa	Condominium	5	2.0	6.0	1842 sq.ft.	Fully Furnished	Minimart, Gymnasium, Security, Playground, Swi...	Air-Cond, Coc Allowed, Was Mac
1	100203973	Segar Courts	NaN	RM 2 300 per month	Kuala Lumpur - Cheras	Condominium	3	1.0	2.0	1170 sq.ft.	Partially Furnished	Playground, Parking, Barbeque area, Security, ...	Air-Cond, Coc Allowed, KTM.
2	100323128	Pangsapuri Teratak Muhibbah 2	NaN	RM 1 000 per month	Kuala Lumpur - Taman Desa	Apartment	3	NaN	2.0	650 sq.ft.	Fully Furnished	Minimart, Jogging Track, Lift, Swimming Pool	
3	100191767	Sentul Point Suite Apartment	2020.0	RM 1 700 per month	Kuala Lumpur - Sentul	Apartment	2	1.0	2.0	743 sq.ft.	Partially Furnished	Parking, Playground, Swimming Pool, Squash Cou...	Cooking Allo Near KTM/ Washing Mac
4	97022692	Arte Mont Kiara	NaN	RM 1 299 per month	Kuala Lumpur - Mont Kiara	Service Residence	1	1.0	1.0	494 sq.ft.	Not Furnished	Parking, Security, Lift, Swimming Pool, Playgr...	Air-(

Figure 4.1.2 (a) (i) Head of csv

Therefore, the monthly_rent column under the code in Figure 4.1.2 (a) (ii) to remove the string value in it and alter the data type of monthly_rent into float for easier use in the future. This adjustment ensures the integration into the prediction model easier. Besides, to assure the value in columns like completion year, rooms, parking, size and so on is in integer or float such that the data could be processed in the prediction model without any problem. By undergoing this step, we could also ensure that the data types match the expected attributes category based on their usage in this project and guarantee the compatibility with future analysis and interpretation. The location for the property is also split into region and area respectively such that the data is able to be process by the model better when grouping or predicting based on area or region.

```
# Change data type to required data type
#Monthy rent is infront to make monthly rent a number

# MONTHLYRENT
# Filter all RM amounts
def filter_rm_amount(amount_str):
    if isinstance(amount_str, str):
        numeric_values = re.findall(r'\d+', amount_str.replace(' ', ''))
        return [int(value) for value in numeric_values] if numeric_values else None
    elif isinstance(amount_str, float):
        return [int(value) for value in str(amount_str).replace(' ', '').split() if value.isdigit()]
    else:
        return None

# Apply filter_rm_amount to 'monthly_rent'
rentData['monthly_rent'] = rentData['monthly_rent'].apply(filter_rm_amount)

# Explode the lists into separate rows
rentData = rentData.explode('monthly_rent')

# Change the data type to int64
rentData['monthly_rent'] = rentData['monthly_rent'].astype('float64')
```

Figure 4.1.2 (b) (i) Filtering Monthly Rent

	prop_name	completion_year	monthly_rent	property_type	rooms	parking	bathroom	size	furnished	facilities	additional_facilities	region	area
0	The Hipster @ Taman Desa	2022	4200	Condominium	5	2	6	1842	Fully Furnished	Minimart, Gymnasium, Security, Playground, Swi...	Air-Cond, Cooking Allowed, Washing Machine	Kuala Lumpur	Taman Desa
1	Segar Courts	0	2300	Condominium	3	1	2	1170	Partially Furnished	Playground, Parking, Barbeque area, Security, ...	Air-Cond, Cooking Allowed, Near KTM/LRT	Kuala Lumpur	Cheras
2	Pangsapuri Teratak Muhibbah 2	0	1000	Apartment	3	0	2	650	Fully Furnished	Minimart, Jogging Track, Lift, Swimming Pool	NaN	Kuala Lumpur	Taman Desa
										Parking,			

Figure 4.1.2 (b) (ii) After filtering

As there will be several facilities listed in a row, therefore it is better to split it into their own dedicated column which allows a clearer representation and analysis of individual facilities associated with each property. A new binary encoding scheme is implemented as 1 implementing the existence of the corresponding facility, while 0 represent its absence. Through this way the diverse range of facilities offered by each property could be captured effectively and improve the data accessibility during analysis and modeling, ultimately contributing to improved insights and decision-making processes.

Second Part

For this part it was related to the data processing process to find the best model which could provide the most accurate predictions to the data set scrapped. After loading the data from the excel file by using Pandas, we first replace the values for 0 to NaN such that the system could capture and filter the invalid values easier, but there is an exception for the data in the columns

derived of facilities and additional facilities. The columns derived from facilities will keep the zeros in their data as it represented the existence of the corresponding facility. The set of data were than split into train and test set with test set size 20% from the data provided.

```
import pandas as pd
from sklearn.model_selection import train_test_split

# Load the dataset, replacing 0 and "0" with NaN
rentData = pd.read_csv("seperated_mudah-apartment-kl-selangor_filtered_with_facilities.csv", na_values=[0, "0"])

# Identify columns or conditions where "0" values are meaningful and should be retained
columns_to_keep_zeros = ['barbeque area_exists', 'club house_exists',
       'gymnasium_exists', 'jogging track_exists', 'lift_exists',
       'minimart_exists', 'multipurpose hall_exists', 'parking_exists',
       'playground_exists', 'sauna_exists', 'security_exists',
       'squash court_exists', 'swimming pool_exists', 'tennis court_exists',
       'aircond_exists', 'cooking allowed_exists', 'internet_exists',
       'washing machine_exists', 'near ktm/lrt_exists']

# Replace NaNs with zeros only in columns specified in columns_to_keep_zeros
rentData[columns_to_keep_zeros] = rentData[columns_to_keep_zeros].fillna(0)

# Split the dataset into train set and test set by 80% and 20%
train, test = train_test_split(rentData, test_size=0.2, random_state=42)

# Examine and verify the dataset quantities
print(train.shape)
print(test.shape)
```

Figure 4.1.2 (c) Replace 0 with NaN

Next, as we now have null values in the data set, thus it is important to replace the null values with mean, median, or mode such that the disposal of the data could be reduced. Then we will proceed to check on the coefficients to determine the relationships between each attribute with the data we are going to predict, which is Monthly Rental in this project.

```
train['completion_year'] = train['completion_year'].fillna(train['completion_year'].median())
train['rooms'] = train['rooms'].fillna(train['rooms'].median())
train['parking'] = train['parking'].fillna(train['parking'].median())
train['bathroom'] = train['bathroom'].fillna(train['bathroom'].median())
```

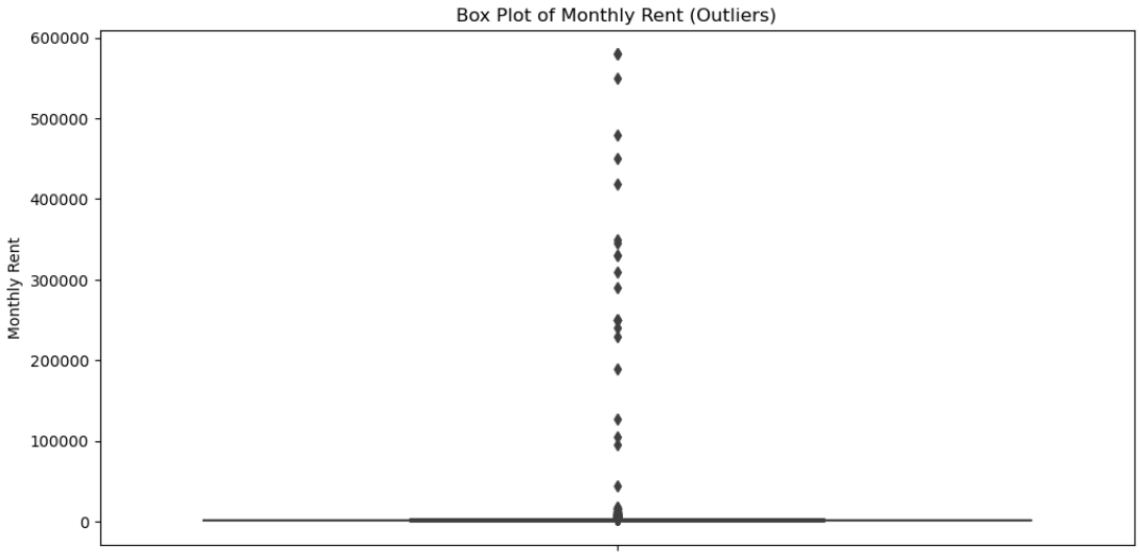
```
train['furnished'] = train['furnished'].fillna(train['furnished'].mode()[0])
train['facilities'] = train['facilities'].fillna(train['facilities'].mode()[0])
train['additional_facilities'] = train['additional_facilities'].fillna(train['additional_facilities'].mode()[0])
```

Figure 4.1.2 (d) Fill in null values with median and mode

It was then proceeded to the visualization of the data in graph to observe the relationship of attributes and to define the outliers in the data set. The outliers of the data set would be recorded and check if the data is cleaned with inside range. After visualizing all the data, the range of cleaned data would be moved to the data preprocessing stage.

Monthly Rent

```
plt.figure(figsize=(12, 6))
boxplot = sns.boxplot(data=train, y=train["monthly_rent"],showfliers=True)
plt.title("Box Plot of Monthly Rent (Outliers)")
plt.ylabel("Monthly Rent")
plt.show()
```




```
plt.figure(figsize=(12, 6))
boxplot = sns.boxplot(data=train, y=train["monthly_rent"],showfliers=False)
plt.title("Box Plot of Monthly Rent (Outliers)")
plt.ylabel("Monthly Rent")
plt.show()
```

Figure 4.1.2 (e) Example of visualizing and identifying outliers

During the Data Preprocessing stage, the numerical attributes and categorical attributes would be defined. NaN rows and invalid data would be dropped out from the data set to provide better accuracy in the model prediction. A custom transformer would be created to preprocess data for the drop invalid and feature processing function, and load in a pipeline.

Pipeline would be created to improve and automate the process of integrating data, enhancing features, training models. As pipelines enables the smooth transfer from preprocessing stages to model training and even evaluation automatically which enhance the efficiency. This automated process could minimize the amount of time taken and decrease the risk of errors.

```
# Specify numeric and categorical features
num_attribs =["completion_year", "rooms", "parking", "bathroom", "size"]
cat_attribs = ["property_type", "furnished", "region", "area",'barbeque area_exists', 'club house_exists', 'gymnasium_exists'
       'jogging track_exists', 'lift_exists', 'minimart_exists',
       'multipurpose hall_exists', 'parking_exists', 'playground_exists',
       'sauna_exists', 'security_exists', 'squash court_exists',
       'swimming pool_exists', 'tennis court_exists', 'aircond_exists',
       'cooking allowed_exists', 'internet_exists', 'near ktm/lrt_exists',
       'washing machine_exists']

# Custom function to drop rows containing NaN values
def drop_nan_rows(df):
    return df.dropna()

def drop_invalid(df):
    # Drop invalid data and outliers data
    mask_rent = (df['monthly_rent'] < 500) | (df['monthly_rent'] >2499)
    mask_size = (df["bathroom"] < 2) | (df["bathroom"] > 2) | (df['size'] < 200) | (df['size'] > 1600) | (df["parking"] < 0)


    df = df[~(mask_rent | mask_size)]  # Use ~ to negate the mask and keep rows where both conditions are False
    return df

def featurePreprocessing(df):
    return df.drop(columns=["prop_name","facilities","additional_facilities"])

#Create a custom transformer to preprocess the data
custom_transformer = Pipeline([
    ('drop_invalid',FunctionTransformer(drop_invalid)),
    ('featurePreprocessing', FunctionTransformer(featurePreprocessing))
])

train_set_tr = custom_transformer.fit_transform(train)
```

Figure 4.1.2 (f) Data Preprocessing

For the modeling process which there are 12 models being carried out which are and left with 5 models which undergo the fine-tuning process.

During the modeling process, Decision Tree Regressor and Random Forest Regressor has occurred overfitting as the returned root-mean-squared (RMSE) value is 36.0211 with a 2.5705% of error rate, and 79.9879 RMSE with 5.7079% of error rate respectively. After carrying out the cross validation, the RMSE for Decision Tree Regressor has gone back to have 274.9220 with 19.6185% error rate, similar to the Random Forest Regressor which have a percentage of error rate around 14.8259%. These two regressions were the first to be excluded from the best model as there were overfitting occurrences.

Whilst Linear Regression, Ridge Regression, SGD Regressor, Support Vector Machine (SVM), and Lasso Regression were excluded due to the RMSE obtained by these models exceeded or are near 250 which is slightly higher than the remaining models. The final picked models which

are the XGBoost Regression, Light GBM Regression, Neural Network, and K-Nearest Neighbor (KNN) Regression which offers better RMSE and lower percentage of error.

After carrying out the fine-tuning for the 4 picked models, the Mean Percentage Error estimated for each model is around 14 to 17 percentage. At last, the XGB Regression is picked due to its low percentage error of 14.43%. It was then fit in the test case to check again if it returns the similar low percentage error output. As a result, the XGBoost Regression obtained a RMSE of 199.2095 and a Mean percentage error of 14.2156%. The model is then export as a pickle file to ease the use in other sections.

Dashboard:

The dashboard is then generated through loading the pickled model such that the data could be used for graph generation. The line graph is selected to display the comparison among current obtained monthly rental and the predicted monthly rental. It is selected as it shows clearer and does not need to worry if the data would be covered by either colored line. They could be in parallel and provides user clearer view on the raise and fall of the prices.

Whilst the histogram is used to display on attributes which could be displayed in a limited range which were close to each other. The histogram is selected to visualize the count of monthly rent to find the median monthly rent. User could roughly estimate the monthly rent would be around RM1200 to RM1500 for average rental.

The map was included to show the average monthly rent for each area. Based on the legend providing guidance towards the colors indicated in the map, users could identify which area would have higher monthly rent and which area would have lower monthly rent. This could help them to decide which area they should rent and also identify which area is more popular.

The value box is used to display the most important information which consists of one value such as the current average monthly rent which gather all the mean of the monthly rental from each apartment and the predicted average monthly rent. It was also used in the identification of area with least and most rental, as compared to using other charts, it would be hard for them to display all the areas exist and user would be hard to identify the value of it as the gap for the highest and lowest count is huge.

4.2 System Components Interaction Operations

Data Flow

The data flow in the apartment rental prediction system begins with the web scraping module, which is collecting data related to the apartments from internet source, Mudah.my. After gathering, this data undergoes thorough processing and preprocessing to remove any duplicate entries, anomalies, and irrelevant data, guaranteeing the accuracy and efficiency of the dataset. Afterwards, the data that has been cleansed is inputted into the predictive modeling module, where machine learning algorithms analyze it in order accurately forecast rental pricing based on the related attributes. The dashboard module serves as the intermediate between the system and users, collecting predictions from the database and displaying them in an engaging and visually appealing manner. By utilizing the dashboard, users can acquire useful insights into rental market trends and make well-informed decisions on apartment rents.

Communication Protocols

- Web Scraping Module

  In this module, it is responsible to fetch the apartment related data from the real estate website, which in this case is Mudah.my. The HTTP protocols is utilized to establish communication with the websites. To ease and unify the programming language used to access the data throughout this project, python is selected to be the main language for web scraping and other modules. Besides, the python language is easy to read compared to other languages and supports libraries such as BeautifulSoup and Selenium which is popular and powerful for parsing HTML and XML documents. Python's compatibility with widely used data analysis and visualization tools like as pandas, NumPy, and Matplotlib enables effortless analysis and visualization of web-scraped data. This integration simplifies the process, enabling you to extract useful insights or generate visualizations from the scraped data with no difficulty.

- Data Mining Module

  The purpose of this module is to obtain relevant insights and patterns from the acquired data connected to apartments. The flexibility and multiple libraries of Python make it the chosen language for this module, ensuring a smooth transition from web scraping to data mining. The module utilizes Python's native libraries and widely used data mining frameworks like scikit-learn or TensorFlow. It applies diverse data mining

techniques such as clustering, classification, and regression to show underlying patterns and relationships within the dataset. Python's compatibility with data analysis tools enables users to explore and comprehend mining findings, allowing them to obtain profound insights on apartment rental trends and patterns to provide better predictions. The data mining module in Python smoothly connects with other components of the system, enabling efficient data flow and analysis from web scraping to data mining.

- Dashboard Module

  The dashboard is mostly supported by Shiny for Python, with a mix of matplotlib or sklearn which is a widely used dashboard components in python language. By having the combinations of different components, the data could be displayed in a more interactive and informative visualizations. Besides the components available in the Shiny, Matplotlib which provides a wide set of plotting functions to allow the creation of various charts, graphs, and visualization are widely used in this project to help illustrate the data obtained better. The dashboard module enables users to easily analyze rental market trends, analyze predictive model outcomes, and make well-informed judgements about flat rentals by utilizing this collection of tools and frameworks.

4.3 System Design Diagram

4.3.1 Use Case Diagram

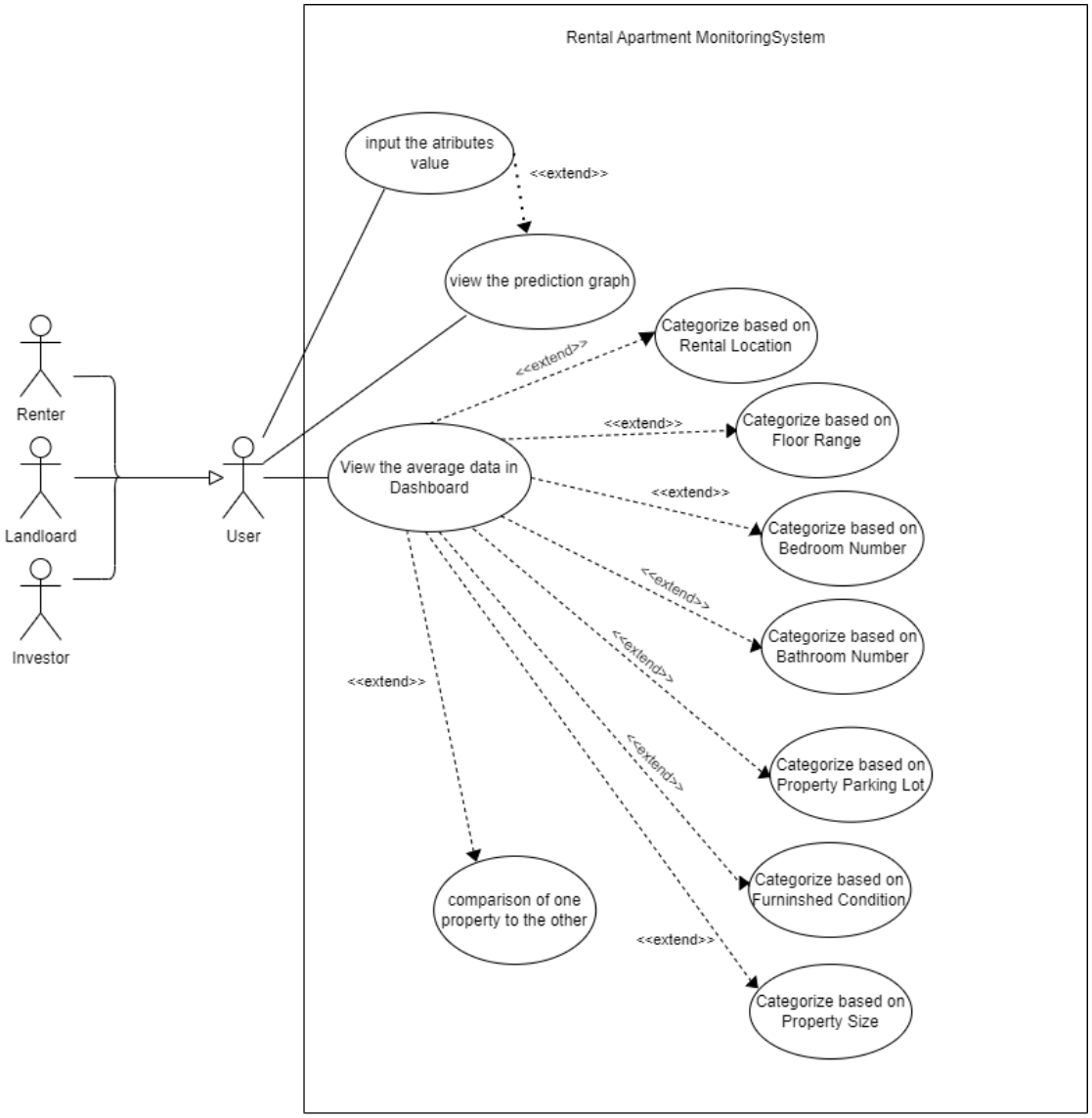


Figure 4.3.1 Use Case Diagram

Figure 3.2.1 shows the Use Case Diagram which illustrates the expected flow of the system would be working. The user would be able to view the average data collected in a dashboard view including the prediction graph. The displayed graph could be categorized based on Rental Location, Floor Range, Bedroom Number, Bedroom Number, Property Parking Lot, or Furnished Conditions. Besides, assume the user is a landlord or Investor, they could access to input the value of the attributes they would like to predict for or compared with the other existing properties. Other than that, the users could also select a fixed property to be compared with the other properties to have a clearer view on the differences.

4.3.2 Activity Diagram

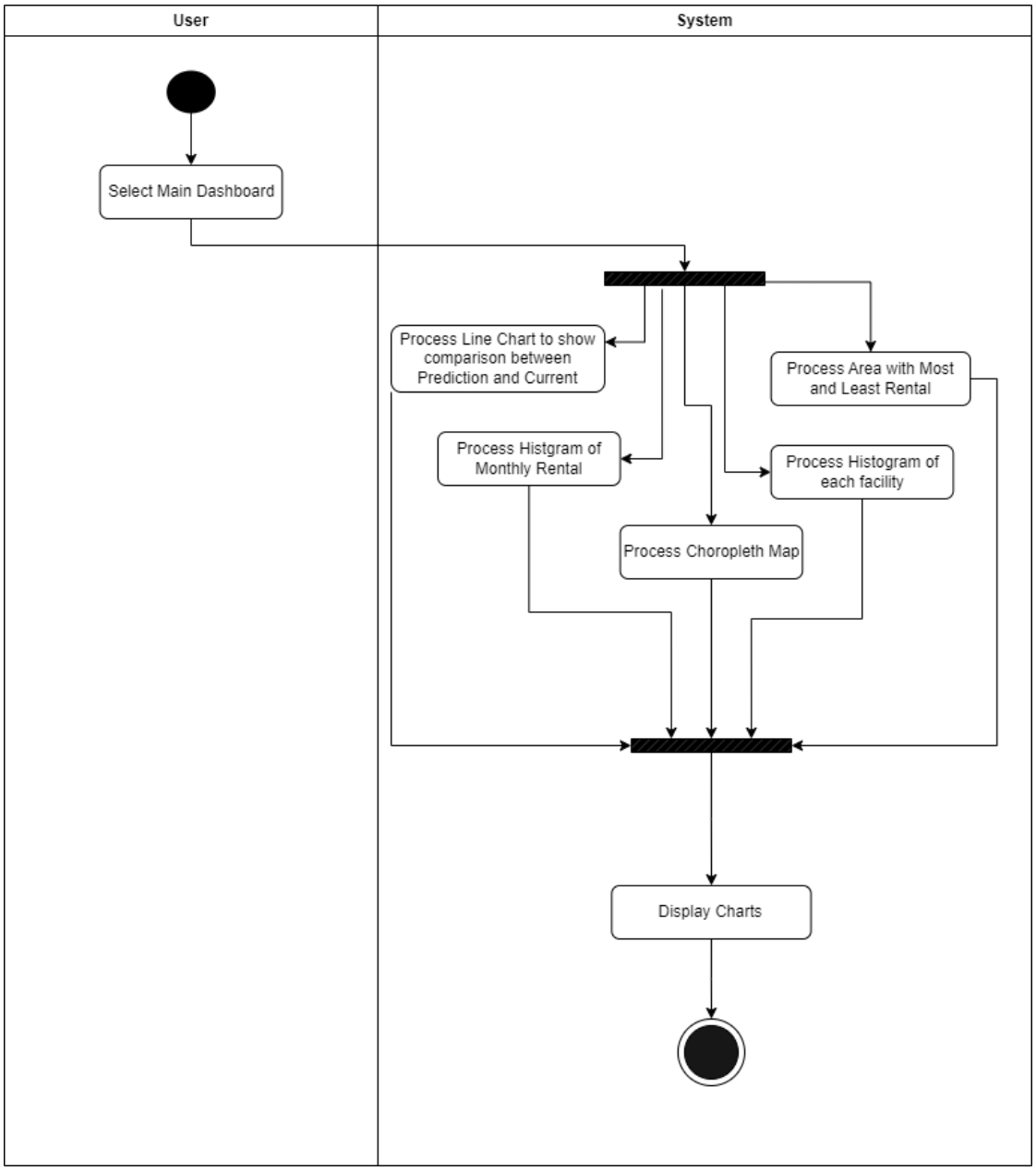


Figure 4.3.2 (a) Main Dashboard Activity Diagram

The main dashboard activity diagram displays how the system actually produce the chart behind the user interface. Once the user selects the tab which as an example here, the main dashboard, the system would be loading the data and process the charts based on the requirements set in the code. Once all the process is done, the charts would be displayed all together. For the display of the graph like above mentioned, the system done most of the works automatically without needing users to input anything.

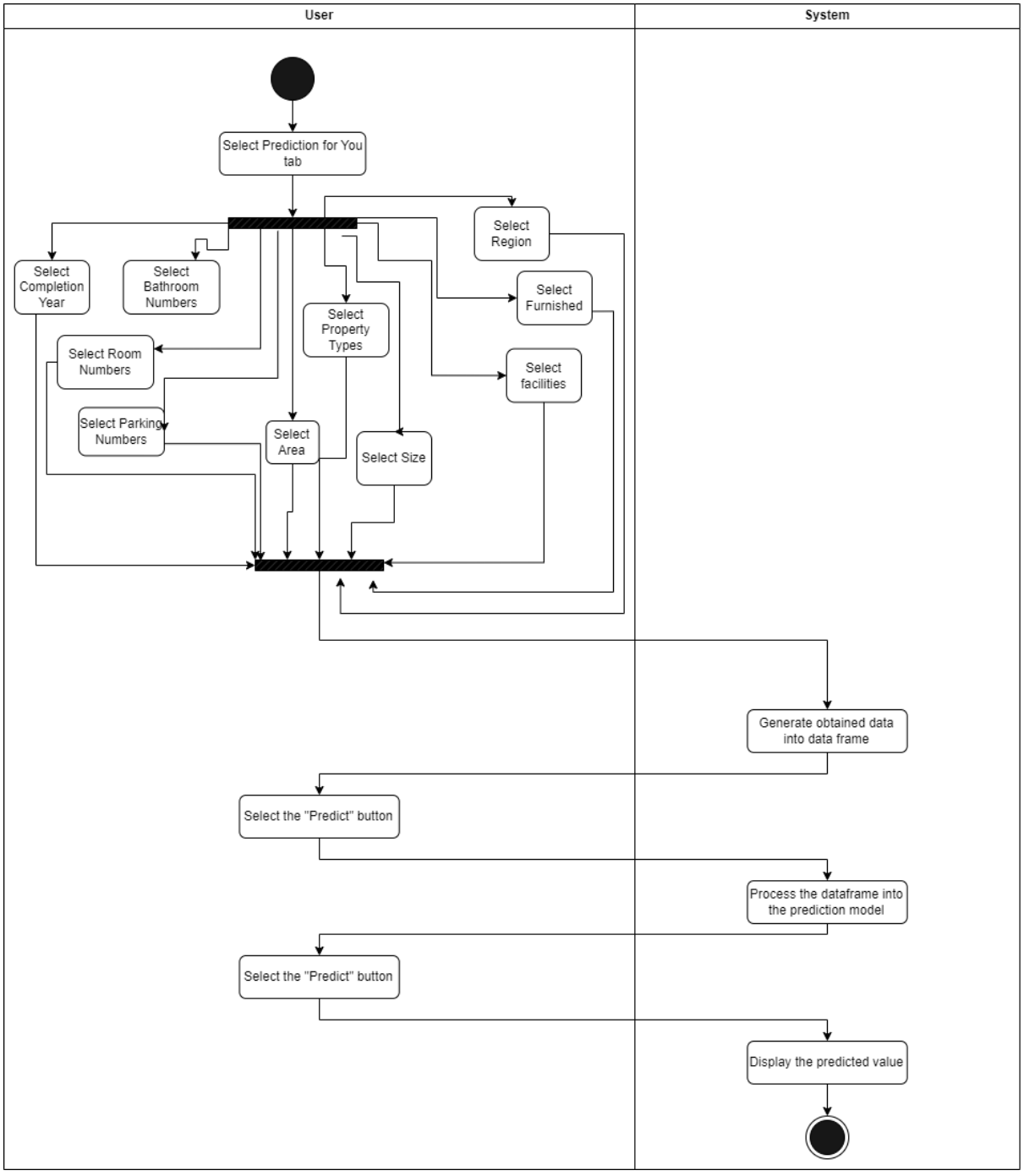


Figure 4.3.2 (b) Prediction for user Activity Diagram

"Prediction for You" which is the tab which provides user to input the attributes based on their needs is the part which consists of most user input in the project. For this screen, the user is required to input values based on the selection give or inside the value box. Then the system would generate the values collected into a data frame with similar format as the csv file used in the training of prediction model to be processed. Once the user selects or click on the "Predict" button, the system will immediately generate the predicted value in the

value box provided. This function ease user which has to estimate for only few apartments which attributes is unique or provide a more accurate prediction based on their own property attributes.

Chapter 5 System Implementation

5.1 Hardware Setup

Hardware used in this project is a computer which could access to the internet and programming tools. Web scaping could be supported by this computer and saved into a csv file.

Description	Specifications
Model	Inspiron 14 5410 2-in-1
Processor	11th Gen Intel(R) Core (TM) i7-1195G7
Operating System	Windows 11 Home Single Language
Graphic	NVIDIA GeForce GT 930MX 2GB DDR3
Memory	16.0 GB
Storage	457GB

Table 5.1 Specifications of laptop

5.2 Software Setup

Jupyter Lab

Jupyter Lab, launched by Windows PowerShell, provides a simplified environment for executing code in a cell-by-cell manner, which significantly improves time and workflow efficiency. This configuration allows users to execute those cells that are relevant to the desired outcomes, eliminating the need to repeatedly rerun the full codebase. Jupyter Lab is used in a wide range of activities, including data scraping, preprocessing, and analyzing geographical data such as obtaining longitude and latitude for various areas. Jupyter Lab improves productivity, encourages experimentation, and simplifies the sequential development process in data-related projects by enabling periodic code execution.

Visual Studio Code 1.88.1

Visual Studio Code (VS Code) is a flexible and powerful code editor that offers wide assistance for several programming languages and frameworks, making it a great choice for dashboard development.

- VS Code provides full assistance for the Python programming language. Python is extensively utilized for doing data analysis and visualization activities, which makes it a perfect selection for constructing dashboards. VS Code offers advanced functionalities such as syntax highlighting, code completion, and debugging for Python, enhancing development efficiency and productivity. In addition, VS Code provides support for widely used Python libraries, such as matplotlib and plotly, which are used for developing dashboards.
- VS Code features a dynamic ecosystem of extensions created by the community, which enhance its capability for different purposes. There is a wide range of extensions available for Python development, including dedicated extensions for Shiny and Python-based dashboard frameworks.

Shiny App

Shiny is a multifunctional Python functionality that simplifies the process of creating interactive visualizations, dashboards, and workflow applications without requiring expertise in web development. Its ability to integrate well with widely used Python data science libraries like Pandas, NumPy, and Plotly, together with its support for different data sources that can be accessed through Python, guarantees smooth integration into current workflows. Shiny's user-friendly interface, pre-configured elements, and adaptable structure enable developers to create advanced applications that can adapt to changing requirements. Moreover, its varied deployment alternatives, such as cloud hosting and self-hosted servers, makes it a solution suitable for a wide variety of data visualization and analysis jobs [20].

API Ninjas

API Ninjas is an online source which provides users to login and obtain their own API key to access the API provided by them .There are various API provided, and for this project it is used to generate the longitude and latitude of the areas such that it could be plotted on the map for better visualization.

Pandas

Pandas is widely known when it comes to excel related or data frame related project. Pandas could handle manipulation of DataFrame object with integrated indexing and provide tools for reading and writing data between the data structures, making data handling seamless [21]. It

also supports label-based slicing, grouping, and merging operations which help ease and simplify the data analysis workflow.

Seaborn

Seaborn is a library in Python for visualization of data based on matplotlib. It offers a more complicated interface for creating better visualization for complex data or graph [22]. In the project, Seaborn is used to define the theme used by the dashboard, and to return the plots defined in the code. Without the Seaborn library, the plots which are generated could not be visualized.

Joblib

Joblib is a Python library that provides various important features for efficient computing and parallel processing. The tool offers a transparent and efficient method of caching output values on disc. This allows users to speed up Python functions and save the results of their computations on disc. It is particularly beneficial for handling big NumPy arrays. Joblib allows for the separation of storage and flow-execution logic from domain or algorithmic code. This permits the design of operations as a collection of clearly defined phases, where inputs and outputs are represented as Python functions. These functions can be stored in a cache and executed again only when needed. Joblib also provides an incredibly parallel helper, which simplifies the process of building parallel code that is both legible and debug able. Moreover, it offers rapid compressed persistence as an alternative to pickle, allowing for efficient management of sizable Python objects [23].

Plotly

Plotly supports Dash which is not usen in this project as we have selected Shiny app to be the main app to develop our dashboard. However, plotly express which provides access to plotly related components is installed and imported to the code such that more types of graphs could be illustrated. Plotly is used to visualize the region and area on the map based on generated longitude and latitude. Choropleth and scatter plot map is plotted out using this library to show which area have the most rental prices.

Matplotlib

Matplotlib is a powerful Python library widely used for generating a broad variety of graphs and plots, such as histograms, scatter plots, line plots, bar plots, and others. The tool provides a wide range of customization features that enable users to precisely adjust the visual look and

arrangement of plots, enabling the creation of visualizations that meet the standards required for publishing. Matplotlib is implemented in the project to showcase histograms, which are valuable for visualizing the distribution of data and investigating its properties. Matplotlib has a wide range of features that allow you to tailor histograms to your unique needs. This includes the ability to modify bin sizes, colors, labels, and annotations.

5.3 Machine Learning Models

Linear Regression

Linear regression is a simple and commonly applied approach for forecasting a continuous target variable using one or more input features. It shows the correlation between the characteristics and the desired outcome through using a linear equation [24].

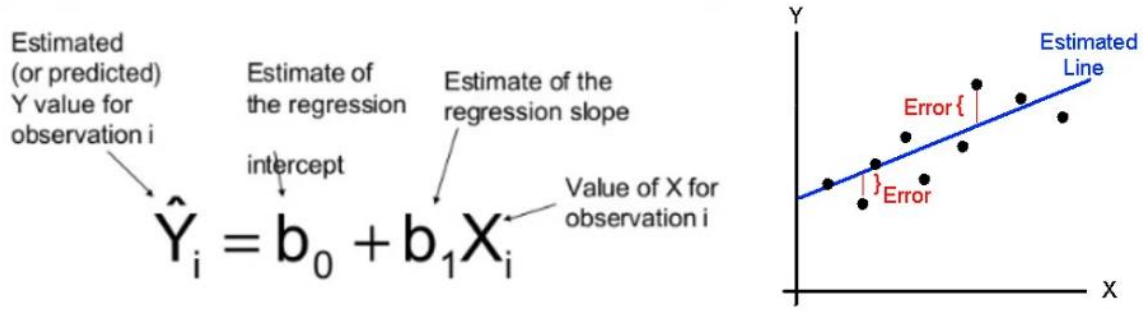


Figure 5.3(a) Linear Regression Formula [25]

Decision Tree Regressor

Decision trees are a flexible approach that may be applied to both classification and regression situations. The data is divided into subsets based on the input feature values in a recursive manner, resulting in a tree-like structure. Each internal node in the tree reflects a choice made using a feature, while each leaf node represents either a class label or a continuous value [26].

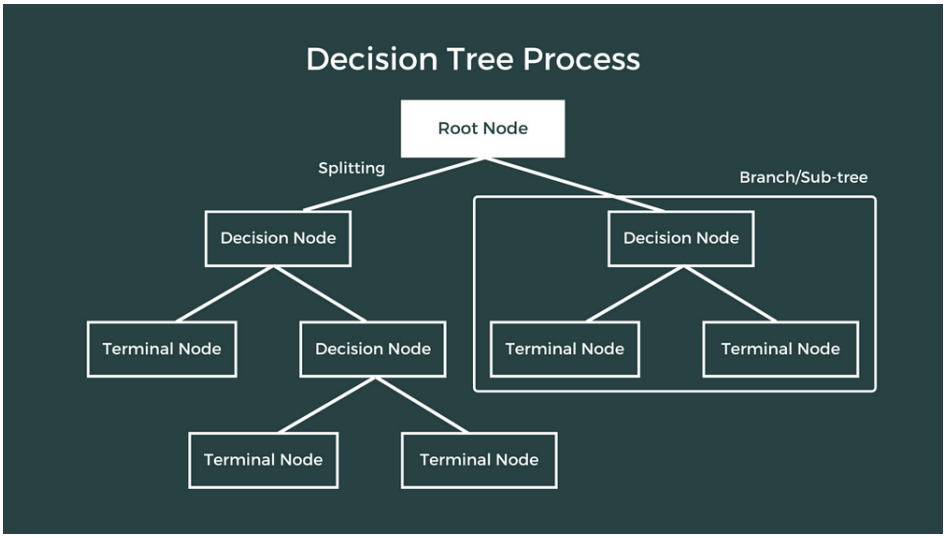


Figure 5.3(b) Decision Tree Regression Illustration [27]

Random Forest Regressor

Random forests are a type of ensemble training approach that combines many decision trees in order to enhance the accuracy and stability of the model. Their operation involves constructing decision trees on random subsets of the data and subsequently combining the forecasts from each tree to provide a final prediction [28].

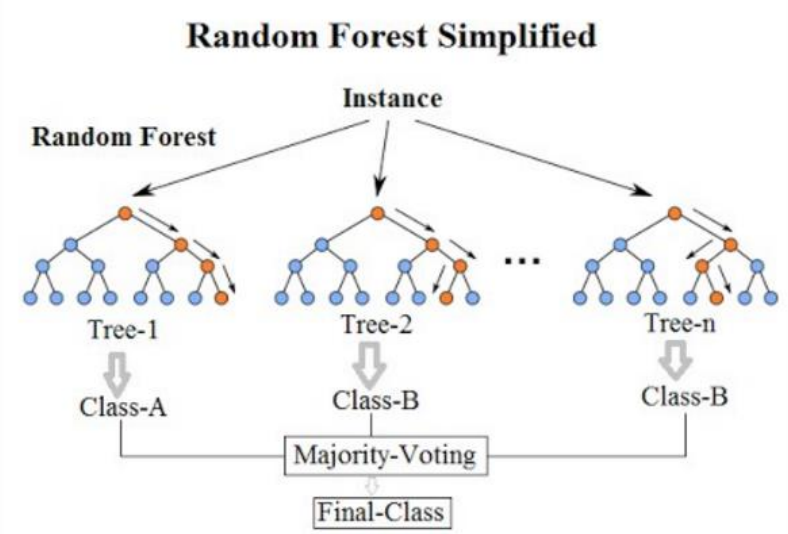


Figure 5.3(c) Random Forest Illustration [29]

Ridge Regression

Ridge regression, or L2 regularization, targets large coefficient values to reduce linear regression overfitting. Ridge regression promotes model simplicity and reduces multicollinearity issues by incorporating a normalization factor into the traditional linear regression cost function. This is especially beneficial for datasets with a large number of parameters or features. This regularization strategy is advantageous for machine learning models that handle huge parameter sets, as excessive weights on parameters might result in overfitting [30].

$$RSS_{L2} = \sum_{i=1}^{n} (Y_i - \hat{Y}_l)^2 + \lambda \sum_{j=1}^{P} B_j^2$$

Figure 5.3(d) Ridge Regression Formula [30]

SGD Regressor

The SGD Regressor is a linear model that applies the Stochastic Gradient Descent (SGD) optimization technique. The algorithm reduces a regularized empirical loss function, enabling live feature selection and learning of lightweight models. The available loss functions are squared error, Huber loss, epsilon-insensitive loss, and squared epsilon-insensitive loss. Regularization can be implemented by utilizing L2, L1, or Elastic Net penalties, which can be adjusted with parameters that govern the degree of regularization [31]. Other settings control factors such as the scheduling of the learning rate, early ending, and warm starting. The model offers techniques for adjusting, forecasting, evaluating, and transforming coefficient matrices between compact and scattered formats. Furthermore, it provides support for partial fitting, allowing for progressive learning [31].

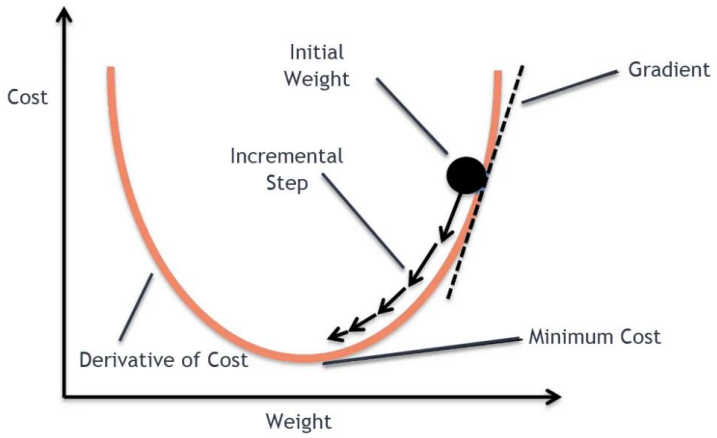


Figure 5.3(e) Stochastic Gradient Descent Visualization [32]

Extreme Gradient Boosting (XGBoost) Regression

XGBoost is a proficient open-source version of the gradient boosting technique, designed for both classification and regression tasks. It is known for its speed and efficiency, frequently exceeding previous executions [33]. It utilizes decision trees in a collective manner, progressively rectifying mistakes committed by previous models. The objective function of XGBoost comprises a loss function and a regularization term, which aid in minimizing the gap between observed and predicted values [33]. It is highly efficient when applied to organized datasets and is commonly utilized by the victors of Kaggle tournaments [34]. The key advantage of XGBoost is its capacity to combine base learners that may be weak on their own but together enhance prediction accuracy [34].

$$O_v = \frac{\text{Sum of residuals}}{\text{Number of residuals} + \lambda}$$

Figure 5.3(f)(i) Simple Formula of XGBoost Regression [33]

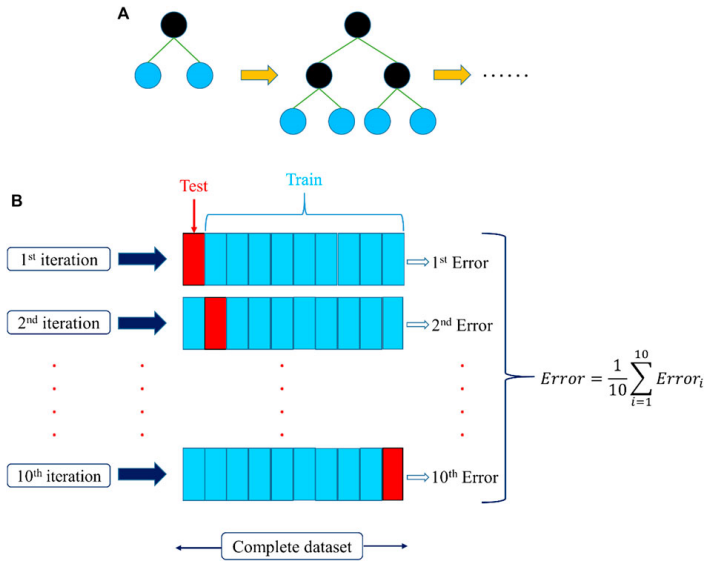


Figure 5.3(f)(ii) Visualization on how XGBoost functions [35]

Light GBM Regression

Similar to XGBoost Regression, Light GBM Regression is a gradient boosting framework which uses tree-based learning algorithms. Instead of using pre-sort-based algorithms like other decision tree learning boosting tools, Light GBM applied histogram-based algorithms [36]. Histogram-based algorithms speeds up the training by applying continuous feature values into discrete bins and also reduces memory usage. By carrying out this model, it could reduce the memory usage as they do not require additional information to be stored for pre-sorting feature values [36] . The disadvantage for this model is that it may case over-fitting due to the tree learning algorithms applied, as so max_depth parameter plays a crucial role to limit the tree depth [36].

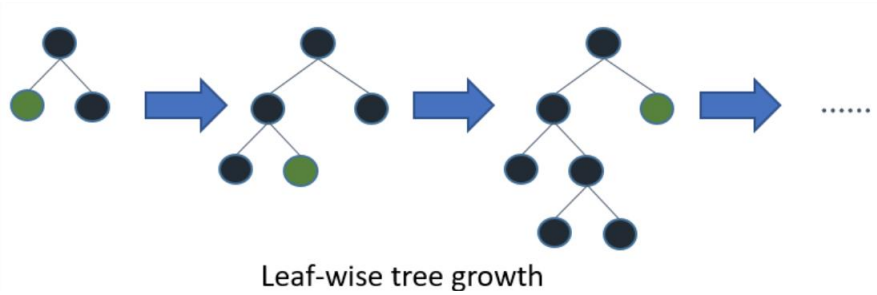


Figure 5.3 (g) Light GBM trees leaf-wise (best-first) [36]

K-Nearest Neighbor (KNN) Regression

K- Nearest Neighbor (KNN) algorithm is a non-parametric, supervised learning classifier that applies proximity to generate the predictions about grouped individual data point [37]. KNN select the average to make predictions about a classification by the defining the distance in between. The most used approach for KNN is the Euclidean distance as it is limited to real-valued vectors. Besides, there are also Manhattan distance which measures the absolute distance between two points, and also Minkowski Distance which is a generalized form of Euclidean and Manhattan distance metrics [37].

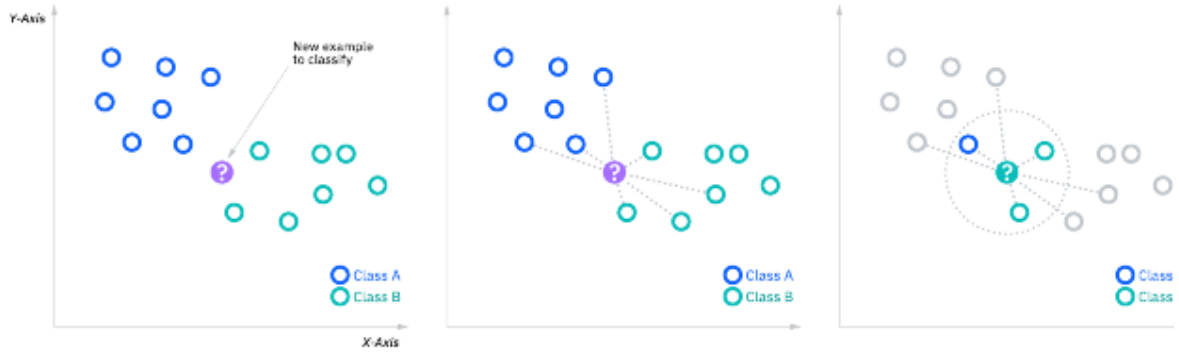


Figure 5.3 (h) K-Nearest Neighbor (KNN) Regression [37]

Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm which supports classification and regression challenges, each data item would be plotted as a pint in n-dimensional space, where n stands for the number of features exist [38]. SVM functions by identifying the hyperplane to classify the data and maximizing the distance between the nearest data point or class [38].

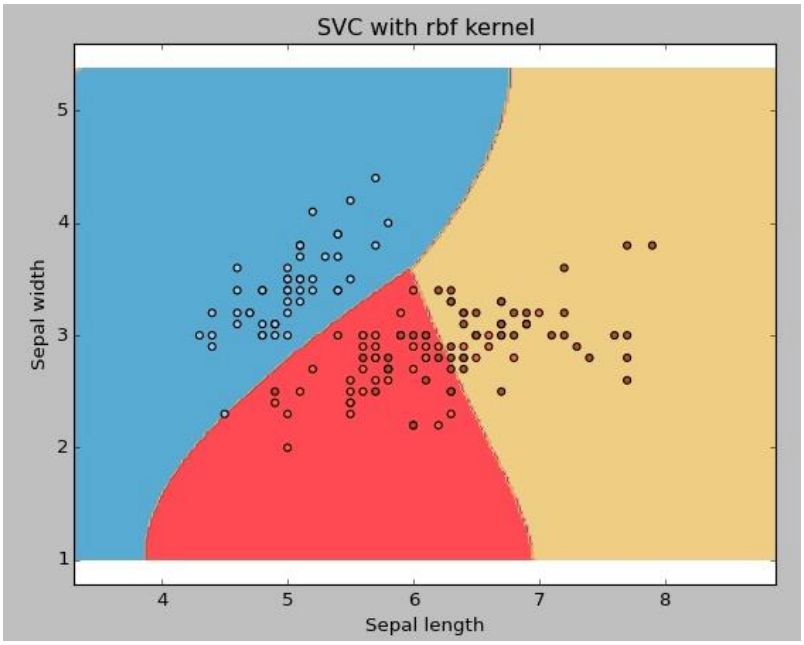


Figure 5.3 (i) Visualization of SVM with sample data [38]

Neural Network

Neural Network is a series of algorithms which attempt to analyze underlying relationships in a data set by mimicking the operation of a human brain. In short, neural networks works like human brain's neurons, or any other organic or artificial neuron system in the nature [39]. This neuron type network shows a strong resemblance to statistical techniques like curve fitting and regression analysis. It contains layers of interconnected nodes, known as perception which feeds produced signals by the multiple linear regression into a function that might be nonlinear [39].

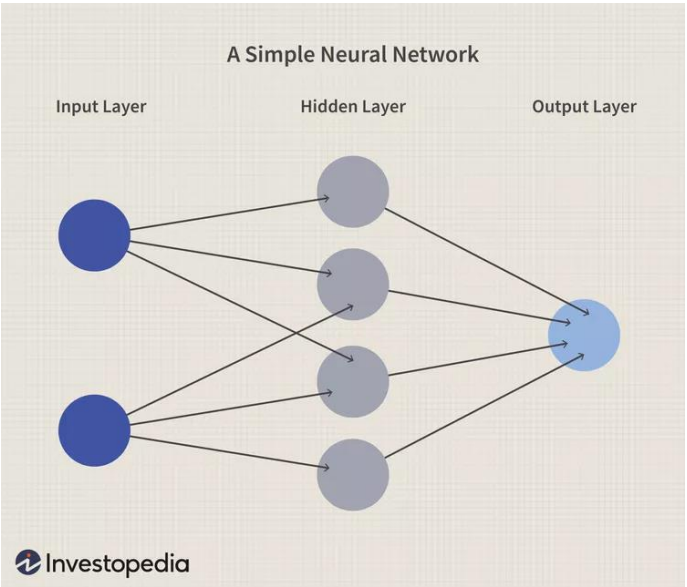


Figure 5.3 (j) Simple Neural Network illustration [39]

Lasso Regression

Least Absolute Shrinkage and Selection Operator (Lasso) Regression or L1 Regularization reduce errors through statistical method by reducing errors due to overfit in training data as it is a form of regularization for the linear regression models [40]. Machine learning commonly utilizes it to manage data with many dimensions, as it enables automatic feature selection. Through incorporating a penalty term into the residual sum of squares (RSS), which is then scaled by the normalization parameter (lambda or λ). The normalization parameter determines the level of regularization that is used. Higher lambda values increase the penalty, causing a greater shrinkage of the coefficients towards zero. Consequently, this reduces the importance of some characteristics or completely removes them from the model, leading to automatic decision-making [40].

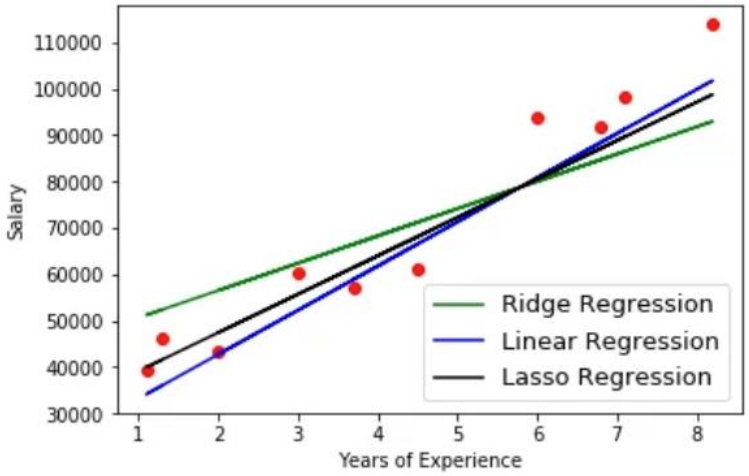


Figure 5.3 (k) Visualization of comparison between regressions for better understanding [41]

5.4 System Operation

Once the dashboard loads, the first page would display the graphs picked out which are expected to explain the data better to help the user understands the trend. Users could also check if their requirements are high or low, and set an expectation towards their requirements. Besides, this could also help them plan their future better as they could estimate the likelihood of the rental price in certain areas or apartments which offers certain facilities. They could check on the highest probability of price expected in the region too.

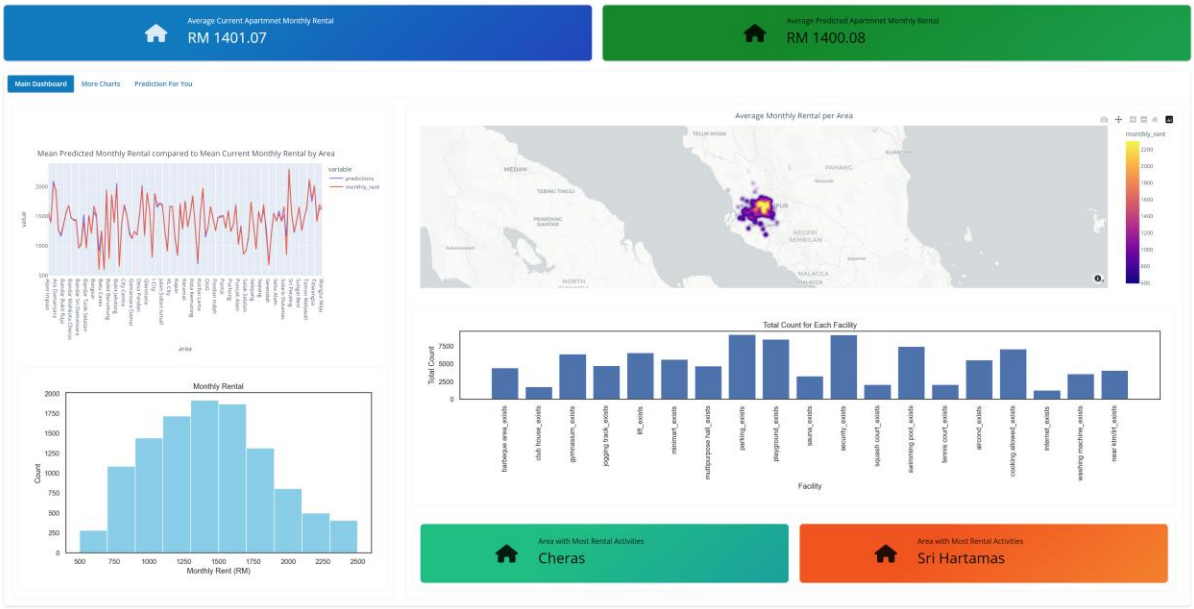


Figure 5.4 (a) Main Dashboard Page

Under the "More Charts" tab, user could filter the required data based on their preferences to show all the statistics of the current available data to provide them clearer view of existing apartments. Each column provides a minimum value and maximum value such that users could set the data for viewing in a certain range.

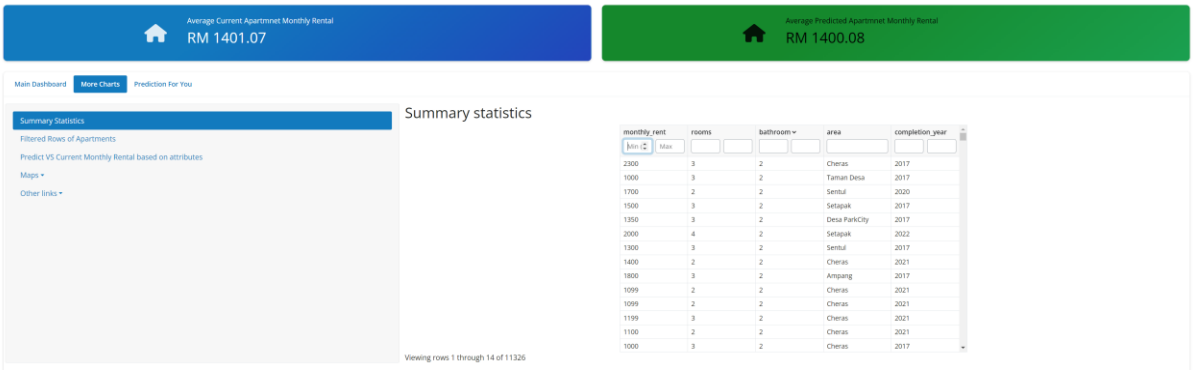


Figure 5.4 (b)(i) Summary Statistics

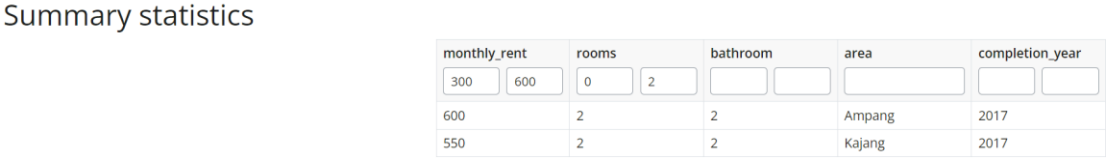


Figure 5.4 (b) (ii) Example output in Summary Statistics

While for Filtered Rows of Apartments, it provides users to filter based on ascending or descending flow of each column when user select on the column name such that user could easily filter based on what they are interested in. It could reduce the time used by user to scroll all the way down if users are not sure what they are looking for.

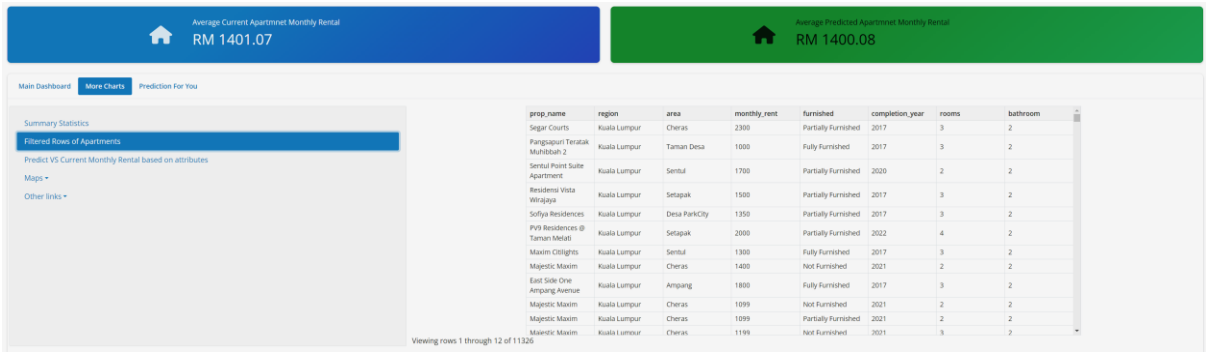


Figure 5.4 (c) (i) Filtered Rows of Apartments

For an example in Figure 5.4 (c)(ii) where the monthly rent is selected to be organized in an ascending format to show the lowest amount first. This function is especially useful for prices as the users could easily check on the highest expected amount through sorting this Monthly Rental column in a descending flow.

prop_name	region	area	monthly_rent ^	furnished	completion_year	rooms	bathroom
Lestari Apartment	Selangor	Petaling Jaya	500	Not Furnished	2017	3	2
Alam Perdana Flat	Selangor	Puncak Alam	500	Not Furnished	2017	3	2
Apartment jasmin	Selangor	Puncak Alam	500	Not Furnished	2017	3	2
	Selangor	Bandar Sunway	500	Fully Furnished	2017	3	2
Vista Perdana (Kajang)	Selangor	Kajang	500	Not Furnished	2017	3	2
Pangsapuri Prima Cempaka	Selangor	Kajang	500	Not Furnished	2017	3	2
Apartment Harmoni	Selangor	Damansara Damai	500	Not Furnished	2017	3	2
Pangsapuri Sri Meranti	Selangor	Bandar Sri Damansara	500	Not Furnished	2017	3	2
Apartment Bayu	Selangor	Petaling Jaya	500	Not Furnished	2017	3	2
Pangsapuri Sri Meranti	Selangor	Bandar Sri Damansara	500	Partially Furnished	2017	3	2
Harmoni Apartment	Selangor	Damansara Damai	500	Partially Furnished	2017	3	2

ıg rows 1 through 11 of 11326

Figure 5.4 (c) (ii) Sort by Monthly Rent ascending flow

This function shows line charts as comparison for the current monthly rental price and the predicted monthly rental price. Users could select to show which attribute to visualize the average comparison. There is also a legend provided to ease the understanding of user on the line represented in the graph.

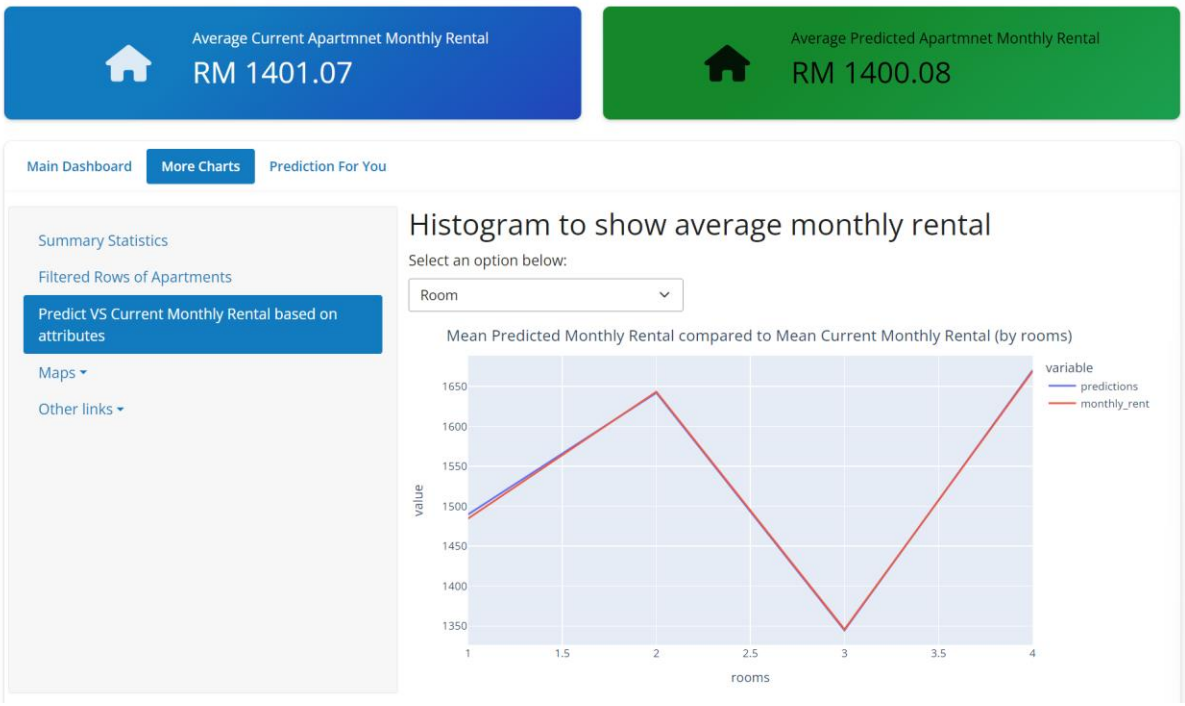


Figure 5.4 (d) Predict VS Current Monthly Rental based on Attributes

It is necessary for this project to have at least a simple map visualization since it is related to the geographical locations. As so, a map function which provides 4 alternatives, which is Choropleth Map which shows a rough visualization on how the monthly rental trend evolves in the region, and a Scatter Plot Map which shows the user a clearer picture of which area has the most and least rental price in average, each for current monthly rental and predicted monthly rental.

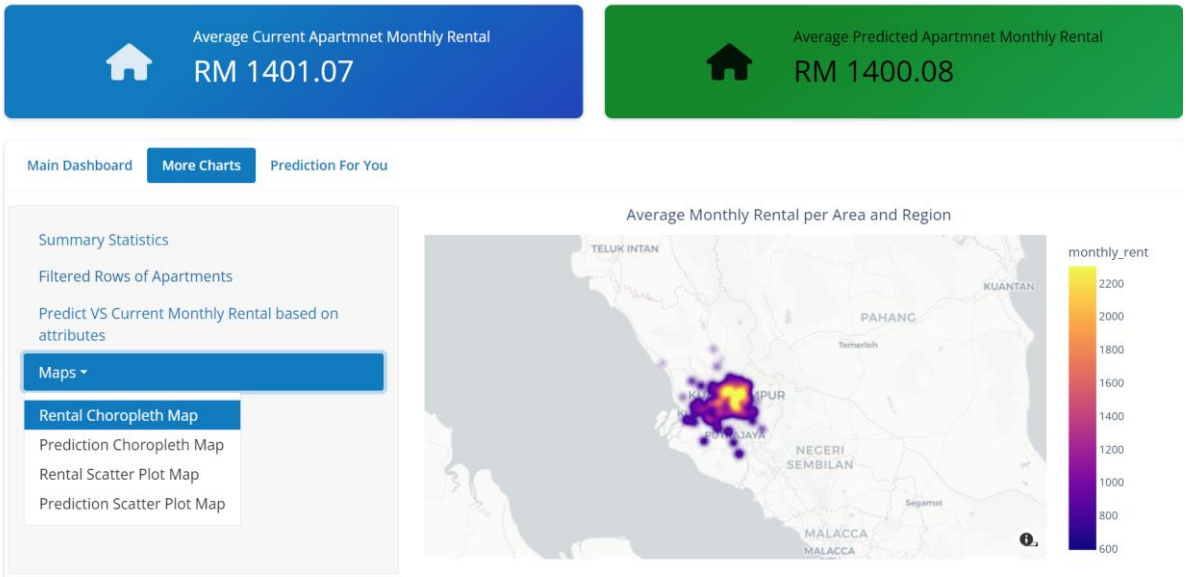


Figure 5.4 (e)(i) Current Rental Choropleth Map

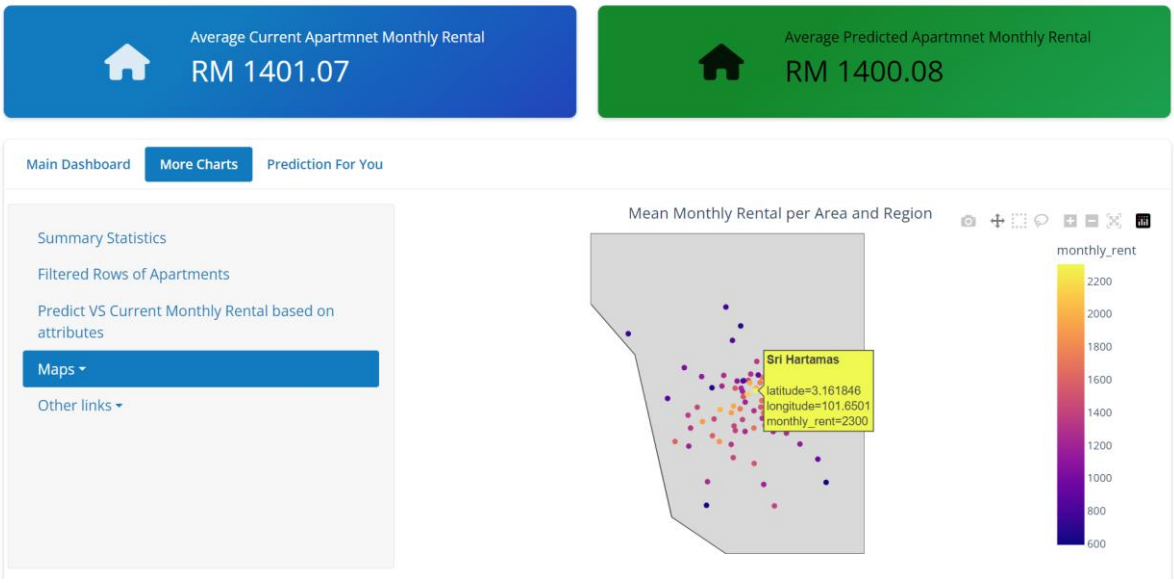


Figure 5.4 (e) (ii) Current Scatter Plot Map

Besides, as mentioned in the objective there is a prediction screen specially for the user to input their properties characteristic and provide them the predicted amount for the property. As shown in Figure 5.4 (f) are the necessary attributes which is needed by the prediction model such that it could produce a prediction value based on what is being input. After user selects the Predict button, then the attributes input would be generated by the prediction model and then display the amount predicted in the value box at the bottom right corner.

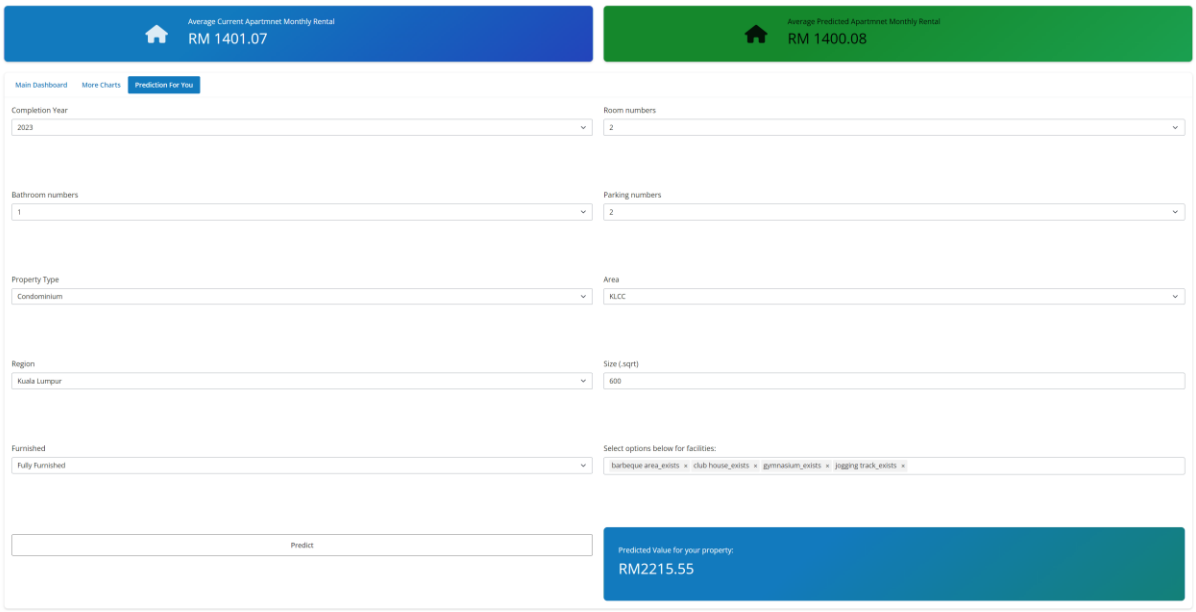


Figure 5.4 (f) Prediction For You screen

5.5 Implementation Issues and Challenges

There were several challenges faced when carrying out this project. Initially, the process of acquiring relevant information through online scraping was complex. The desired information was deeply embedded inside the website's code structure, requiring critical inspection of the scraping techniques. In addition, the existence of dynamic content and anti-scraping techniques hindered the extraction process, resulting in a longer duration to obtain useable data.

Enhancing the accuracy of the model brought another challenge. While the removal of outliers was a crucial phase to obtain better accuracy, as indicated by measures like as RMSE and MPE, it required much testing and analyzing on the data obtained to train the model. This involved conducting tests on several models, such as XGBoost, neural network, ridge regression, and fine-tuning parameters to enhance the performance. These tedious steps were lengthy but necessary for determining the best efficient model for the given dataset. It was very much time consuming as when you could not identify which attributes is interfering your prediction accuracy and waiting for the model to load. Once you decided to end the day testing certain model or make some changes with your dataset, you are required to rerun and wait for the model to load all data such that it could start on again.

The process of mapping location data, especially for choropleth maps was not easy. Although choropleth maps were successful in displaying regional data, they frequently lacked the capability to support localized maps. Mapping regions or territories on the map by hand was challenging, especially when dealing with complex geographical borders. The lack of resources and techniques for customizing maps further intensified the challenge of producing precise and useful visualizations. Even if there are resources, due to lack of time and the limitations of familiarity with the languages used, it is not a very good approach to try out the maps without knowing if you would success.

Finally, adding interactive functionalities through the use of Shiny app, particularly in Python, was a challenge for people who were not familiar with the technology. Although Shiny was extensively utilized for constructing interactive web apps in R, there was a limited availability of resources and tutorials for applying it in Python. To overcome this challenge, it was necessary to demonstrate a strong commitment to acquiring knowledge about the framework and modifying its features to align with the specific requirements of the project. Therefore, add an additional level of complexity to the execution procedure.

Chapter 6 System Evaluation and Discussion

6.1System Testing

Fixed display for Dashboard			
Purpose: To check on the layout and required output which should be displayed functions properly			
No.	Review Checklist	Success/ Not Success	Comments
1	Display value box for average monthly rental	Success	-
2	Text is displayed	Success	-
3	Navigation pane displayed correct navigation panel	Success	-
4	Navigation pane could be selected	Success	-
5	Select navigation pane shows different color	Success	-
6	The value box shows value in two decimal places	Success	-
7	Text in value box is relevant	Success	-
8	RM is included	Success	-
9	House icon is displayed	Success	-
10	Size of value box is fixed	Success	-
11	Display will fit the page	Success	The charts will be shrunk to fit the screen which will cause the graph to alter their size
12	Navigation Pane is in a box	Success	

Table 6.1 (a) Fixed display for Dashboard

Information displayed in Dashboard			
Purpose: To ensure the graphs displayed required values and functions implemented is functionable			
No.	Review Checklist	Success/ Not Success	Comments
1	Graphs displayed as required	Success	-
2	Legend included	Success	-
3	x- and y-axis display label	Success	-
4	Tab in “More Charts” is functionable	Success	-
5	Summary Statistic could insert minimum value and maximum value	Success	-
6	There are no null values nor 0 values in the data	Success	-
7	The filter functions properly and display values based on user requirement	Success	-
8	Values are sort according user’s selection either ascending or descending	Success	-
9	The number of rows scrolled are relevant	Success	-
10	The chart display based on attributes selected by user in the predict versus current monthly rental	Success	-
11	There is difference between predicted and current monthly rental, not the same data used	Success	-
12	The area is plotted based on the location on the map	Success	-
13	Maps are focusing on Kuala Lumpur, Malaysia	Success	-

14	Area information would be displayed when the mouse is hover over	Success	-
15	The map could be zoomed	Success	-
16	The completion year selection is sorted in descending	Success	-
17	Room, bathroom, and parking numbers start from 0 to10	Success	-
18	Property Type could be displayed	Success	-
19	Area existed is shown in an ascending sorting	Success	-
20	The region only has Kuala Lumpur and Selangor	Success	-
21	The size field could be filled in manually through typing or pressing the increase or decrease icon button	Success	-
22	User could select only Furnished, Not furnished, or partially furnished	Success	-
23	Drop down panel works perfectly	Success	-
24	Drop down could be scrolled	Success	-
25	All facilities are included	Success	-
26	User could select multiple facilities	Success	-
27	The select value is recorded in the backend	Success	-
28	The selected value could be transformed into a data frame	Success	-
29	The data frame could pass through the prediction model	Success	-
30	Predicted value is displayed	Success	-
31	“Predict” button would change the color when hovered	Success	-

32	Value is displayed in the value box	Success	-

Table 6.1 (b) Information displayed for Dashboards

6.2Objectives Evaluations

- ***Provide prediction of apartment rental through monitoring system***

  This objective is done by carrying out several models for training during the Data Mining Module. A finalized model is selected an export such that it could be called in the visualization module which displays the dashboard. The model is exported in pickle file format as this facilitates the easier integration and usage of the model in other Python scripts, specifically in the visualization module that is responsible for presenting the dashboard. Exporting the model allows users of dashboard to easily access it and use it to forecast flat rental costs by just loading the dashboard. This helps users make informed decisions when exploring the rental market.

  The subobjectives which is to develop a form such that users could fill in their own data in each field to generate a personalized prediction. This function was successfully implemented to provide users with the flexibility to assess precise requirements, especially in cases where there are no similar data examples as reference.

- ***To observe factors influencing the rental prices the most***

  This objective was accomplished through careful analysis and modelling conducted during the machine learning stages and shown in the dashboard. The elements that influence rental pricing were carefully discovered and analyzed using advanced approaches such as feature importance analysis and predictive modelling.

  The investigation indicated that in addition to geographical location, which had a major impact on rental pricing, other factors such as security and parking facilities were also shown to be highly important. This conclusion was derived from a combination of statistical analysis, and machine learning methods. Moreover, the significance of security and parking characteristics was confirmed by diverse model evaluation methodologies, ensuring the reliability and validity of the results.

- ***Deployment of Dashboard***

  The deployment of dashboard was done which allows user to access and engage with the visualizations easily. As the development phase was done, the dashboard will be

implemented using an appropriate platform or framework, ensuring the audience could access it and gain understanding easily. The excel file was called and processed in the backend of dashboard which then by using possible data which could generate any relative charts for the users, such as Histogram and Line Charts which is widely used in the dashboard. Legend is included in the graph which have several lines, or the display function which show the details of hovered area in the maps. the implementation of the dashboard was a major achievement in the project, allowing users to utilize the knowledge gained from data analysis and visualization to make well-informed decisions.

- ***Identify the location with less renting activities***

  The identification of areas with lower levels of rental activity has been accomplished by analyzing the histogram display on the dashboard. The histogram illustrates the distribution of rental activities across various areas. By analyzing the histogram, we were able to identify places where rental operations had lower frequency. These areas were indicated by shorter bar heights or lesser counts. This study enabled the identification of particular areas where rental activities were relatively less common or where there was lesser demand for rentals. As from the graph we could see those areas such as Banting which is located far away from the city center, and Bandar Damai Perdana which have the lowest count of rental activities happening.

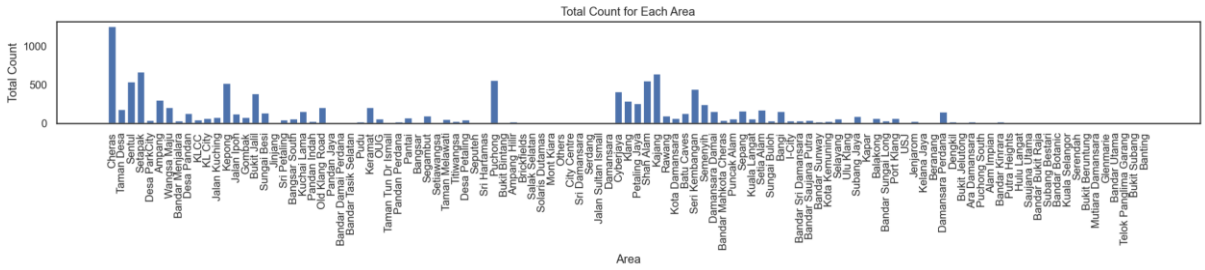


Figure 6.1 (a) Area based on rental activities count

Besides, to illustrate this in a more detailed format, there is a value box which displays the most and least rental activities area in the main dashboard such that the users could obtain the information easier in a simpler format than reviewing the histogram.

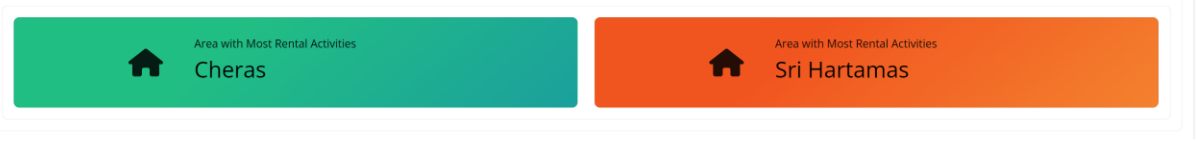


Figure 6.1 (b) Area with most and least rental activities

Chapter 7 Conclusion and Recommendation

7.1 Conclusion

The purpose of this project is to develop a rental apartment monitoring system which provides a prediction system and displays the data collected through a dashboard illustration. This system does not just provide users statistics illustrate by graphs and charts but could also provide the users apartment rental prices prediction, which could be categorized in several attributes such as Number of rooms, Location, Floor Range, Number of Parking Lots and so on. The development of this system could also provide the Malaysians a rental apartment monitoring system which could provide predictions to facilitate them estimating the prices of the apartments.

There are 4 systems being reviewed in this project, which are California Housing Partnership, Lancashire County Council, 99arces, and Mudah.my. Lancashire County Council is by far the system which has the most mature developed functions and module compared to the others. This provides ideas on what should be included in the project and future improvements on the project. CRISP-DM is used as a methodology for this project as it is suitable for the project scope. CRISP-DM 6 steps provided guidance such that the project goes on its path. The main processes that are carried out in this project are Business Understanding, Data Understanding, and Data Preparation. There are challenges faced where preparation on the knowledge is much needed before the implementation of the project such that the project could proceed smoothly.

Besides, the dashboard which is the main output for the project with included prediction system is successfully developed. Illustrating graphs to understand which areas have higher average apartment rental and shows the comparison of predicted and current value for rental prices. These insights will provide the users a clearer vision and to make future planning on their financial aspects related to accommodations.

In conclusion, the development of the rental apartment monitoring system represents a significant important step in delivering Malaysians useful knowledge regarding flat rental pricing and patterns. The development of an all-encompassing dashboard that not only displays rental data visually but also provides predictive functionalities, will be able to facilitate the users to make well-informed choices regarding their housing requirements.

7.2 Recommendation

In order to improve the apartment rental monitoring system dashboard, several ideas could be adopted to boost its efficiency and precision in offering insights into the rental market dynamics in Malaysia, specifically in Kuala Lumpur and Selangor.

Firstly, there is the potential to enhance the visualization of the maps displayed in the dashboard using matplotlib to clearly outline the boundaries of places inside Kuala Lumpur and Selangor. By distinguishing these geographic areas on the map, users can acquire a greater understanding of the rental market patterns across various neighborhoods and districts. This update will not only give customers with a more precise geographical context, but also offer a more comprehensive perspective on rental costs and the distribution of demand.

Furthermore, the data gathering efforts and improve the initial data processing processes could be improvised to enhance the precision and accuracy of the information presented on the dashboard. Action such as increasing the quantity and variety of data gathered, including specific property characteristics, past rental transactions, and market trends could result in a more complete dataset for research. The increased size of this dataset allows for the detection of minor trends and patterns, hence improving the ability of the dashboard to make accurate predictions.

Moreover, improving the preprocessing processes could uncover hidden relationships and connections within the data which offer users with more detailed understandings of rental market dynamics. The dashboard can reveal small yet major elements that impact rental prices, such as seasonal trends, property features based on the new data column in the webpage such as nearest KTM/MRT location or mini market which is nearby through the use of advanced feature engineering and more data collection effort done. By doing a more in-depth inspection, users are able to enhance their ability to make precise predictions and well-informed decisions when evaluating rental properties.

Other than that, this project could be improvised by showing more models predicted value and the possible difference of the predicted value with reality such that user could obtain more information. Users of the dashboard could forecast the value they could bear on and decide to investigate more about available apartments in the market with known requirements. This could help the users minimize the area they were looking for.

By following these ideas, the apartment rental monitoring system dashboard can become a strong and important tool for consumers traversing the Malaysian rental market. The improved mapping capabilities and advanced data processing methods enable users to acquire deeper understanding of rental market dynamics, make well-informed choices, and eventually optimize their investments in rental properties or leasing strategies.

REFERENCES

[1] A. Tamezi, “Malaysia Property Market Outlook 2024”, PropertyGuru.com.my, 22-Jan-2024 [Online]. Available: https://www.propertyguru.com.my/property-guides/malaysia-property-market-outlook-2024-72338. [Accessed: 23-Apr-2024].

[2] K. B., “More renters than buyers in the next two years due to rising living expenses,” New Straits Times, 22-Feb-2024. [Online]. Available: https://www.nst.com.my/property/2024/02/1016375/more-renters-buyers-next-two-years-due-rising-living-expenses. [Accessed: 23-Apr-2024].

[3] J. Anderson and L. Rainie, “Main findings: Teens, technology, and human potential in 2020,” Pew Research Center: Internet, Science & Tech, 29-Feb-2012. [Online]. Available: https://www.pewresearch.org/internet/2012/02/29/main-findings-teens-technology-and-human-potential-in-2020/. [Accessed: 25-Nov-2022].

[4] “More people renting properties rather than buying,” The Edge Malaysia. [Online]. Available: https://theedgemalaysia.com/article/more-people-renting-properties-rather-buying. [Accessed: 06-Sep-2023].

[5] “Room for rent, part 1,” The Centre. [Online]. Available: https://www.centre.my/post/room-for-rent-part-1. [Accessed: 06-Sep-2023].

[6] “What is a dashboard? A complete overview”, Tableau.com. [Online]. Available: https://www.tableau.com/learn/articles/dashboards/what-is#:~:text=The%20main%20use%20of%20a,relevant%20data%20in%20key%20areas. [Accessed: 25-Apr-2024].

[7] B. Caplan, “3 reasons to liven up your marketing with visual content,” Score.org, 20-Mar-2019. [Online]. Available: https://www.score.org/blog/3-reasons-liven-your-marketing-visual-content. [Accessed: 27-Nov-2022].

[8] “California housing partnership,” California Housing Partnership, 30-Sep-2015. [Online]. Available: https://chpc.net/. [Accessed: 06-Sep-2023].

[9] Lancashire County Council, “About us,” Lancashire.gov.uk. [Online]. Available: https://www.lancashire.gov.uk/council/about-us/. [Accessed: 06-Sep-2023].

[10] 99acres.com. [Online]. Available: https://www.99acres.com/property-rates-and-price-trends-in-bangalore-prffid. [Accessed: 28-Nov-2022].

[11] "Mudah.my - Malaysia's largest marketplace," Mudah.my - Malaysia's largest marketplace. [Online]. Available: https://www.mudah.my/about/index.htm?ca=9_s. [Accessed: 27-Aug-2023].

[12] "IBM documentation," Ibm.com, 17-Aug-2021. [Online]. Available: https://www.ibm.com/docs/en/spss-modeler/saas?topic=dm-crisp-help-overview. [Accessed: 04-Dec-2022].

[13] N. Hotz, "What is CRISP DM?," Data Science Process Alliance, 10-Sep-2018. [Online]. Available: https://www.datascience-pm.com/crisp-dm-2/. [Accessed: 04-Dec-2022].

[14] "Business Understanding Overview," Ibm.com, 17-Aug-2021. [Online]. Available: https://www.ibm.com/docs/en/spss-modeler/saas?topic=understanding-determining-business-objectives. [Accessed: 04-Dec-2022].

[15] "Data Understanding Overview," Ibm.com, 17-Aug-2021. [Online]. Available: https://www.ibm.com/docs/en/spss-modeler/saas?topic=understanding-data-overview. [Accessed: 04-Dec-2022].

[16] "Data preparation," DataRobot AI Cloud, 21-Mar-2018. [Online]. Available: https://www.datarobot.com/wiki/data-preparation/. [Accessed: 04-Dec-2022].

[17] "Crisp DM methodology," Smart Vision Europe, 17-Jun-2020. [Online]. Available: https://www.sv-europe.com/crisp-dm-methodology/. [Accessed: 04-Dec-2022].

[18] "Evaluation Overview," Ibm.com, 17-Aug-2021. [Online]. Available: https://www.ibm.com/docs/en/spss-modeler/saas?topic=evaluation-overview. [Accessed: 04-Dec-2022].

[19] "IBM documentation," Ibm.com, 17-Aug-2021. [Online]. Available: https://www.ibm.com/docs/en/spss-modeler/saas?topic=deployment-overview. [Accessed: 04-Dec-2022].

[20] "Shiny for Python," Shiny for Python. [Online]. Available: https://shiny.posit.co/py/. [Accessed: 22-Apr-2024].

[21] "pandas - Python Data Analysis Library," Pydata.org. [Online]. Available: https://pandas.pydata.org/about/. [Accessed: 22-Apr-2024].

[22] "Seaborn: Statistical data visualization — seaborn 0.13.2 documentation," Pydata.org. [Online]. Available: https://seaborn.pydata.org/. [Accessed: 22-Apr-2024].

[23] "Joblib: running Python functions as pipeline jobs — joblib 1.4.0 documentation," Readthedocs.io. [Online]. Available: https://joblib.readthedocs.io/en/stable/. [Accessed: 22-Apr-2024].

[24] "What is linear regression?," Ibm.com. [Online]. Available: https://www.ibm.com/topics/linear-regression. [Accessed: 22-Apr-2024].

[25] D. Raghunath, "Linear regression - raghunath D," Medium, 24-Jan-2019. [Online]. Available: https://medium.com/@rndayala/linear-regression-a00514bc45b0. [Accessed: 22-Apr-2024].

[26] "Sklearn.tree.decisionTreeRegressor," scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html. [Accessed: 22-Apr-2024].

[27] The Click Reader, "Decision tree regression explained with implementation in python," Medium, 19-Oct-2021. [Online]. Available: https://medium.com/@theclickreader/decision-tree-regression-explained-with-implementation-in-python-1e6e48aa7a47. [Accessed: 22-Apr-2024].

[28] "Sklearn.Ensemble.RandomForestRegressor," scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html. [Accessed: 22-Apr-2024].

[29] W. Koehrsen, "Random forest simple explanation - will koehrsen," Medium, 27-Dec-2017. [Online]. Available: https://williamkoehrsen.medium.com/random-forest-simple-explanation-377895a60d2d. [Accessed: 22-Apr-2024].

[30] “What is ridge regression?,” Ibm.com. [Online]. Available: https://www.ibm.com/topics/ridge-regression. [Accessed: 22-Apr-2024].

[31] “Sklearn.Linear_model.SGDRegressor,” scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDRegressor.html. [Accessed: 23-Apr-2024].

[32] H. Dawar, “Stochastic gradient descent,” Analytics Vidhya, 14-May-2020. [Online]. Available: https://medium.com/analytics-vidhya/stochastic-gradient-descent-1ab661fabf89. [Accessed: 23-Apr-2024].

[33] nikki2398, “XGBoost for regression,” GeeksforGeeks, 29-Aug-2020. [Online]. Available: https://www.geeksforgeeks.org/xgboost-for-regression/. [Accessed: 23-Apr-2024].

[34] J.Brownlee, “XGBoost for Regression”, Machinelearningmastery.com. [Online]. Available: https://machinelearningmastery.com/xgboost-for-regression/. [Accessed: 23-Apr-2024].

[35] N. M. Shahani, X. Zheng, C. Liu, F. U. Hassan, and P. Li, “Developing an XGBoost regression model for predicting Young’s modulus of intact sedimentary rocks for the stability of surface and subsurface structures,” Front. Earth Sci., vol. 9, 2021.

[36] “Features — LightGBM 4.0.0 documentation,” Readthedocs.io. [Online]. Available: https://lightgbm.readthedocs.io/en/stable/Features.html. [Accessed: 23-Apr-2024].

[37] “What is the k-nearest neighbors algorithm?,” Ibm.com. [Online]. Available: https://www.ibm.com/topics/knn. [Accessed: 23-Apr-2024].

[38] S. Ray, “Learn how to use Support Vector Machines (SVM) for data science,” Analytics Vidhya, 19-Mar-2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/. [Accessed: 23-Apr-2024].

[39] J. Chen, “What is a neural network?,” Investopedia, 07-Feb-2024. [Online]. Available: https://www.investopedia.com/terms/n/neuralnetwork.asp. [Accessed: 23-Apr-2024].

[40] “What is lasso regression?,” Ibm.com. [Online]. Available: https://www.ibm.com/topics/lasso-regression. [Accessed: 23-Apr-2024].

[41] M.Bajwa, “Ridge and Lasso a Simple Overview”,Medium.com, 25-Jun-2019. [Online]. Available: https://medium.com/@mubarakb/ridge-and-lasso-a-simple-overview-6c3cc72fe7d. [Accessed: 23-Apr-2024].

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3, Y3	**Study week no.:7**
Student Name & ID: Kuit Ying Qian 20ACB02533	
Supervisor: Dr. Abdulkarim Kanaan Jebna	
Project Title: Rental Apartment Monitoring System	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Preprocessing the data with several different models to see the pattern for collected data and identify which model could be used.

2. WORK TO BE DONE

Improvise the RMSE and accuracy as the amount obtained is extremely large

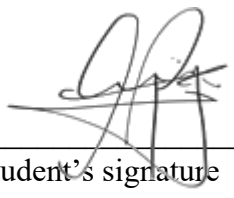
3. PROBLEMS ENCOUNTERED

- Not sure how to clean the data more thoroughly

4. SELF EVALUATION OF THE PROGRESS

Should obtain more insights previously, and understand the data and required output

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3, Y3	**Study week no.:8**
Student Name & ID: Kuit Ying Qian 20ACB02533	
Supervisor: Dr. Abdulkarim Kanaan Jebna	
Project Title: Rental Apartment Monitoring System	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Solve the data preprocessing challenge and successfully clean the data to obtain more logical insights

2. WORK TO BE DONE

Export the best model and start deploying the dashboard

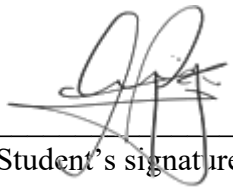
3. PROBLEMS ENCOUNTERED

- Not familiar on the prediction or could not expect what should be in the exported file which cause undecidable decision on which type of approach to carry out the deployment of dashboard

4. SELF EVALUATION OF THE PROGRESS

This research is missed out during the previous understanding and thought it could be done easily with HTML or Bootstrap approach

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3, Y3	**Study week no.:9**
Student Name & ID: Kuit Ying Qian 20ACB02533	
Supervisor: Dr. Abdulkarim Kanaan Jebna	
Project Title: Rental Apartment Monitoring System	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Roughly found the way to do the dashboard and written some report

2. WORK TO BE DONE

Deploy dashboard and understand what is in the pickle file

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS

Settle down with one method and not trying to rush the project

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3, Y3	**Study week no.:10**
Student Name & ID: Kuit Ying Qian 20ACB02533	
Supervisor: Dr. Abdulkarim Kanaan Jebna	
Project Title: Rental Apartment Monitoring System	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Explore the web scrapping as there are some patterns found for the code

2. WORK TO BE DONE

Scrap the whole data out

3. PROBLEMS ENCOUNTERED

- Not sure on how the required data should be obtained from the nested loop of dictionaries and list with JSON format

4. SELF EVALUATION OF THE PROGRESS

Settle down and not to be over rushing this project

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3, Y3	**Study week no.:12**
Student Name & ID: Kuit Ying Qian 20ACB02533	
Supervisor: Dr. Abdulkarim Kanaan Jebna	
Project Title: Rental Apartment Monitoring System	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Figure out how to solve the dashboard problem and show the dashboard

2. WORK TO BE DONE

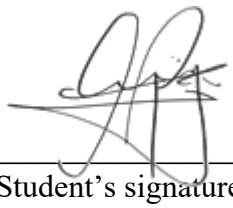
Insert more types of graphs available to visualize the max out of data obtained

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS

The graphs require thorough understanding and try to figure it out in a python method

Supervisor's signature



Student's signature

POSTER

APARTMENT RENTAL MONITORING SYSTEM

OBJETCIVES

- Provide prediction of apartment rental through monitoring system
  Monitoring is not enough? Let's provide some predictions to ease the process on analyzing the trend
- To observe factors influencing the rental prices the most
  Is the factor very much related to what you need?
- Deployment of Dashboard
  A bunch of text is bland and not interesting? No worries!
- Identify the location with less renting activities
  Check which location is best for investment

INTRODUCTION

The rental market in Malaysia is experiencing a surge in demand for apartments due to economic shifts and changing demographics. To address challenges like rising property prices and negotiation difficulties, the Rental Apartment Monitoring System offers data analysis and visualization tools for informed decision-making and transparency in the rental market.

MOTIVATION

Lack of dashboard system available for the public in MALAYSIA

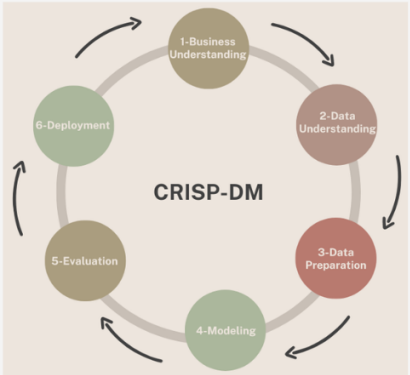
The increasing of majority population ended up renting as unable to afford houses

Scope and Direction

- Collect data of apartments around Kuala Lumpur
- Provide estimation on future trends
- Visualized statistics to read on current trend and predicted trend
- Enable input to personalized the prediction for user

CRISP-DM

Methodology used as a guide which ensures the organized review and application of data to achieve project goals



Machine Learning Models

- XGBoost Regression
- Light GBM Regression
- K-Nearest Neighbor (KNN) Regression
- Neural Network
- Polynomial Regression

PLAGIARISM CHECK RESULT

FYP_check_Turintin.pdf

ORIGINALITY REPORT

3%	2%	1%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.papercamp.com Internet Source	1%
2	dl.ucsc.cmb.ac.lk Internet Source	<1%
3	Submitted to Napier University Student Paper	<1%
4	Submitted to Westcliff University Student Paper	<1%
5	towardsdatascience.com Internet Source	<1%
6	www.coursehero.com Internet Source	<1%
7	Submitted to CSU, San Jose State University Student Paper	<1%
8	Submitted to Liverpool John Moores University Student Paper	<1%
9	www.theedgemarkets.com Internet Source	<1%

10	www.jazindia.com Internet Source	<1 %
11	Submitted to Washington University of Science and Technology Student Paper	<1 %
12	polibits.cidetec.ipn.mx Internet Source	<1 %
13	Submitted to Bocconi University Student Paper	<1 %
14	en.wikipedia.org Internet Source	<1 %
15	www.mdpi.com Internet Source	<1 %
16	Submitted to Universidade do Porto Student Paper	<1 %
17	eprints.utar.edu.my Internet Source	<1 %
18	epub.lib.aalto.fi Internet Source	<1 %
19	arxiv.org Internet Source	<1 %
20	www.agrifoodscience.com Internet Source	<1 %
21	www.yotec.net Internet Source	

PLAGIARISM CHECK RESULT

		<1 %
22	online-journals.org Internet Source	<1 %
23	research.usq.edu.au Internet Source	<1 %
24	www.lume.ufrgs.br Internet Source	<1 %
25	Allen, Phillip E.. "CMOS Analog Circuit Design", Oxford University Press Publication	<1 %
26	discombobul8d.com Internet Source	<1 %
27	publikationen.bibliothek.kit.edu Internet Source	<1 %
28	www.researchgate.net Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

PLAGIARISM CHECK RESULT

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	KUIT YING QIAN
ID Number(s)	20ACB02533
Programme / Course	BUSINESS INFORMATION SYSTEMS
Title of Final Year Project	RENTAL APARTMENT MONITORING SYSTEM

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 3 % **Similarity by source** Internet Sources: 2% Publications: 1% Student Papers: 2 %	
Number of individual sources listed of more than 3% similarity:	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Dr. Abdulkarim M. Jamal Kanan

Date: 26/04/2024

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

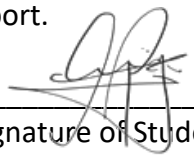
CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB02533
Student Name	KUIT YING QIAN
Supervisor Name	DR. ABDULKARIM KANAAN JEBNA

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
√	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.



_____

(Signature of Student)

Date: