

**PERSONALISED A.I. CHATBOT FOR KAMPAR TOURISM
MOBILE APPLICATION**

By
Low Zhi Yuan

A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF COMPUTER SCIENCE (HONOURS)
Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2021

REPORT STATUS DECLARATION FORM

Title: Personalised A.I. Chatbot for Kampar Tourism Mobile Application

Academic Session: JAN 2021

I LOW ZHI YUAN
(CAPITAL LETTER)

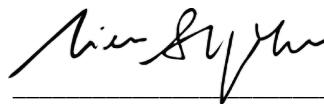
declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

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:

125, Lorong Merbau,
Taman Simpang,
34700, Simpang Perak.

Date: 15/4/2021

Liew Soung Yue

Supervisor's name

Date: 16/4/2021

**PERSONALISED A.I. CHATBOT FOR KAMPAR TOURISM
MOBILE APPLICATION**

By
Low Zhi Yuan

A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF COMPUTER SCIENCE (HONOURS)
Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2021

DECLARATION OF ORIGINALITY

I declare that this report entitled “**PERSONALISED A.I. CHATBOT FOR KAMPAR TOURISM MOBILE APPLICATION**” 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 : LOW ZHI YUAN

Date : 15/4/2021

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my FYP supervisor Ts Dr Liew Soung Yue for guiding me on my project topic. Without his proper idea and direction, I would not be able to accomplish this project successfully. He has provided me with a clear idea on my project topic and led me in every software development phases. It was my pleasure to carry out my project under his guidance.

ABSTRACT

In this modern and digitalized era, Artificial Intelligence (AI) has become popular in almost all sectors. One of the most common applications of AI is chatbot. In the market today, there are lots of chatbots applications available, such as Amazon's Alexa, Apple's Siri and Google's Google Assistant. Most of the businesses have implemented chatbots due to its high availability and efficiency, especially for customer service purpose. However, most of the chatbots are rule-based. The chatbots will return the respective results to the users only if the predefined questions are asked. Everyone gets the same answer for similar inputs. For general Question and Answer (QnA), rule-based chatbots are more than enough. Yet, for the tourism sector, chatbots which can learn the users' preferences and recommend them with their interested things are more likely to satisfy the users. Different tourists may have different personalities, and thus feel interested in different tourist attractions. If they manage to get the recommendations which match their needs, the user experience will be better.

To solve the problem stated above, an enhanced version of KamparBot, which is also one of the features of GoKampar travel guide application is proposed. The previous version of KamparBot allows the users to search for restaurant, lodging and tourist attractions when travelling in Kampar. Compared with the original application, the new KamparBot will be able to collect users' favourites and personalities for personalising their recommendations, instead of just showing same responses to all users who ask similar questions. Hence, this project will focus on developing an AI-based chatbot which can understand the user's needs and preferences and therefore recommend suitable places based on machine learning techniques.

TABLE OF CONTENTS

TITLE PAGE	i
DECLARATION OF ORIGINALITY	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
TABLE OF CONTENTS	v
LIST OF FIGURES	vii
LIST OF TABLES	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement and Motivation	1
1.2 Project Scope	1
1.3 Objectives	2
1.4 Background Information	2
1.5 Proposed Approach	4
1.6 Highlight of Achievement	5
1.7 Report Organization	6
CHAPTER 2 LITERATURE REVIEW	7
2.1 Artificial Intelligence Marketing: Chatbots	7
2.2 Analytical study on use of AI techniques in tourism sector for smarter customer experience management	8
2.3 Chatbot-based Tourist Recommendations Using Model-based Reasoning	10
2.4 HR Based Interactive Chatbot (PowerBot)	11
2.5 AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)	12
CHAPTER 3 SYSTEM DESIGN	13
3.1 System Architecture	13

3.2	System Flowchart	15
3.3	Use Case Diagram	17
3.4	System Functionalities	18
3.4.1	Navigation Drawer	18
3.4.2	“My Activity” Page	19
3.4.3	“My Favourite” Page	20
3.4.4	Recommender System	23
CHAPTER 4	METHODOLOGY AND TOOLS	24
4.1	Methodology	24
4.2	Tools / Technologies Involved	25
4.3	User Requirements	28
4.4	Timeline	29
CHAPTER 5	SYSTEM SPECIFICATION	30
5.1	IBM Watson Assistant	30
5.2	Content-Based Filtering Recommender System	35
CHAPTER 6	IMPLEMENTATION AND TESTING	37
6.1	Implementation	37
6.2	Implementation Issues and Challenges	40
6.3	Testing	41
CHAPTER 7	CONCLUSION	48
7.1	Project Review	48
7.2	Novelty and Contribution	48
7.3	Future Work	49
BIBLIOGRAPHY		50
WEEKLY REPORTS		52
POSTER		56
PLAGIARISM CHECK RESULT		57
FYP 2 CHECKLIST		59

LIST OF FIGURES

Figure Number	Title	Page
Figure 1.5.1	Overview of System Flow	4
Figure 2.2.1	Recommendation Engine for Tourism	8
Figure 3.1.1	System Architecture Diagram	13
Figure 3.2.1	System Flowchart	15
Figure 3.3.1	Use Case Diagram	17
Figure 3.4.1	Main Page	18
Figure 3.4.2	Navigation Drawer	18
Figure 3.4.3	“My Activity” Page	19
Figure 3.4.4	User Input Details	19
Figure 3.4.5	“My Favourite” Page	20
Figure 3.4.6	Game Page 1 - Lodging	20
Figure 3.4.7	Game Page 2 - Foods	21
Figure 3.4.8	Game Page 3 - Attractions	21
Figure 3.4.9	The End of Game	22
Figure 3.4.10	Result 1	23
Figure 3.4.11	Result 2	23
Figure 4.1.1	Prototyping Approach	24
Figure 4.4.1	FYP1 Timeline	29
Figure 4.4.2	FYP2 Timeline	29
Figure 5.1.1	Subscription Plan of IBM Watson Assistant	30
Figure 5.1.2	Create Assistant Page	30
Figure 5.1.3	Create Skill Page	31
Figure 5.1.4	Intent Page	31
Figure 5.1.5	Example of Intent #Yes with User Examples	32
Figure 5.1.6	Example of Intent #Foods with User Examples	33
Figure 5.1.7	Dialog Tree (Conversational Flow)	34
Figure 5.1.8	Details of Node in Dialog Tree	35
Figure 5.1.9	Content-Based Filtering	36
Figure 6.1.1	Restaurants with One-Hot Encoded Labels in Kampar	37

Figure 6.1.2	User Rating on Restaurant in Kampar	38
Figure 6.1.3	User Input with Ratings	38
Figure 6.1.4	Dot Product of User Ratings and Labels of Each Rated Restaurant	39
Figure 6.1.5	Dot Product of User Profile and Labels of Each Restaurant	39
Figure 6.1.6	Formula to Compute Final Weighted User Preference	39
Figure 6.1.7	Top 10 Recommended Restaurants	40

LIST OF TABLES

Table Number	Title	Page
Table 6.3.1	Test Cases for Sign In / Register Module	41
Table 6.3.2	Test Cases for Chatbot Module	43
Table 6.3.3	Test Cases for Recommender System Module	45
Table 6.3.4	Test Cases for Navigation Module	45
Table 6.3.5	Test Cases for “My Activity” Module	46
Table 6.3.6	Test Cases for “My Favourite” Module	47

LIST OF ABBREVIATIONS

<i>AI</i>	Artificial Intelligence
<i>QnA</i>	Question and Answer
<i>GDP</i>	Gross Domestic Product
<i>UTAR</i>	Universiti Tunku Abdul Rahman
<i>TARC</i>	Tunku Abdul Rahman University College
<i>Covid-19</i>	Coronavirus Disease 2019
<i>IBM</i>	International Business Machines Corporation
<i>VR</i>	Virtual Reality
<i>HR</i>	Human Resources
<i>OS</i>	Operating System
<i>FAQs</i>	Frequently Asked Questions
<i>PHP</i>	Hypertext Preprocessor
<i>HTML</i>	Hypertext Markup Language
<i>CSS</i>	Cascading Style Sheets
<i>API</i>	Application Programming Interface
<i>NoSQL</i>	No-Structured Query Language
<i>JVM</i>	Java Virtual Machine
<i>XML</i>	Extensible Markup Language
<i>UI</i>	User Interface
<i>IDE</i>	Integrated Development Environment
<i>UML</i>	Unified Modeling Language
<i>ERD</i>	Entity Relationship Diagram
<i>JSON</i>	JavaScript Object Notation
<i>FYP</i>	Final Year Project

Chapter 1 Introduction

1.1 Problem Statement and Motivation

Nowadays, there are a lot of chatbot applications can be found in the market. However, most of them are command-based, rather than machine learning-based. With predefined questions and answers in its database, the chatbot will be able to serve as a customer service for replying the users. However, this kind of chatbot does not have the capability to learn the users' interests through their conversations. No matter how many times the users ask the same question, the chatbot will only respond with the same answer predefined in its database. Although command-based chatbot is good enough for offering full time customer service (24-hours availability), there are still lots of rooms for improvements. Basically, all people in the world will have their own thinking, interest, and personality. Hence, it is very crucial to understand their basic needs, and make use of it to provide them with their desired responses.

To overcome these limitations, an Artificial Intelligence (AI) based chatbot with a proper recommender system feature can be implemented to serve the customers. With the help of machine learning, the chatbot will have the ability of self-learning based on the users' questions and feedbacks. After that, the chatbot will be able to customize the response for different users according to their respective preferences extracted. With this, the users might have the impression that they are having a two-way communication with the computer machine. Therefore, the main motivation for this project is to enhance user satisfaction with the system. Besides that, it is also possible for the chatbot to react in a right way to the users who speak with Malaysia slang by training and localising the model well with user data.

1.2 Project Scope

The proposed solution to the problem stated above will be a mobile application of personalised chatbot developed under the Android platform. It will be an improvement for the previous senior's project on the development of AI chatbot for Kampar Tourism mobile application, KamparBot. This project intends to enhance the user experience in using chatbot for tourism in Kampar by customizing the answers

returned by chatbot based on the users' interests. Machine learning will be used to develop a recommender system for learning the preferences of users from time to time through their asked questions and feedbacks to the response given. Eventually, the chatbot will achieve the user customisation.

1.3 Objectives

The main objectives of the personalised chatbot mobile application for tourism are as below:

- To allow users to get information regarding the tourist attractions in small town like Kampar more easily and efficiently, such as operation time, address, images and rating.
- To allow users to receive personalised response from the chatbot based on their needs.
- To allow users to communicate with the chatbot with slang without affecting the results.

1.4 Background information

According to (Hirschmann, 2020), tourism is an important sector in Malaysia as it contributes a lot to Malaysia's Gross Domestic Product (GDP), just after manufacturing and commodities sectors. Based on the research, about 5.9% of the total GDP of Malaysia is contributed by tourism industry, and Kuala Lumpur is the most popular city visited. Usually, when talking about tourism, people tend to visit those large cities with better media exposure. However, sometimes there are even more interesting places and beautiful sceneries for the travellers to explore in small towns like Kampar.

According to (Kampar District Council, 2020), "Kampar" refers to a small town developed on tin mining in earlier days. Nowadays, people's first impression on Kampar will be definitely an education city, due to the development of education institutions such as Universiti Tunku Abdul Rahman (UTAR) and Tunku Abdul

Rahman University College (TARC) there. It cannot be denied that these educational agents had successfully brought a new life to Kampar and boosted its economy for these few years. However, this may create a stereotype for the people from outside that Kampar is just a university town with students living there. Most of them might think that other than a university town, there will be nothing worth to explore in Kampar, and such a point of view is not true. Furthermore, the outbreak of Covid-19 pandemic has caused the outlook of Malaysia tourism sector to become quite bad and the economy of the country to drop significantly compared to previous years.

Chatbot can be defined as an intelligent agent which can interact with the users in natural language. In this modern era, chatbot has been widely applied in most of the businesses as part of their customer services. With the implementation of chatbot, the organisations can reduce their manpower as the chatbot is available for all time to help answering and solving their customers' problems. In general, there are two categories of chatbot, which are command-based and AI-based. For command-based chatbot, it will only reply the users with its predefined answer. Meanwhile, AI-based chatbot is more flexible as it can learn and improve itself from time to time based on the user feedback.

Recommender system is very essential if the chatbot wishes to predict the user behaviour and recommend with appropriate items. There are several machine learning techniques can be conducted to build a recommender system, such as content-based filtering and collaborative filtering. With the help of recommender system, the AI-based chatbot will be able to achieve user personalisation based on its similarity measure.

1.5 Proposed Approach

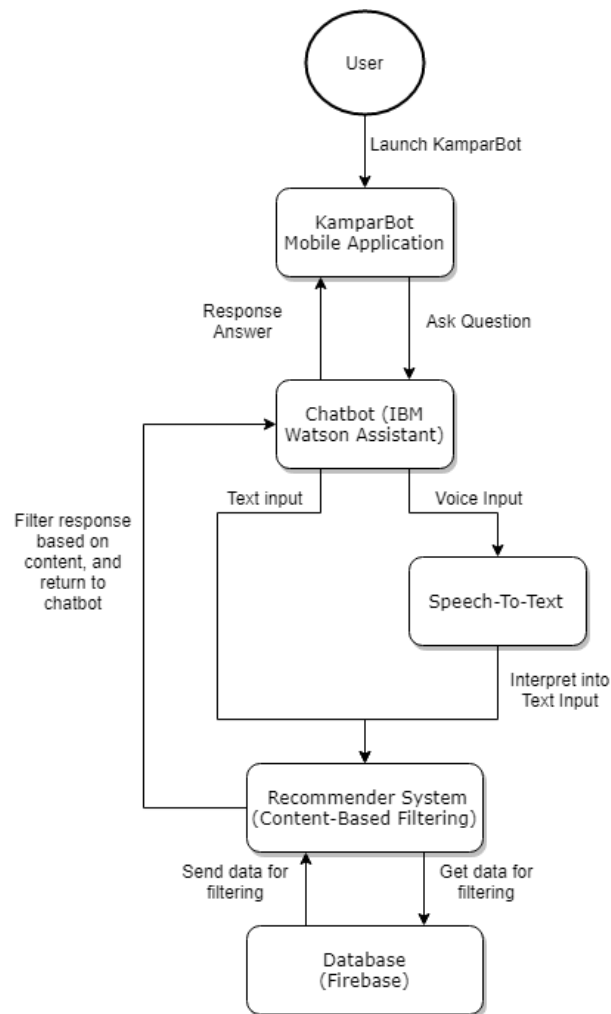


Figure 1.5.1: Overview of System Flow.

Figure 1.5.1 above illustrates an overview of the system flow in this project. There are two main modules to be focused, which are the chatbot module and the recommender system module.

For chatbot module, it is basically handled by the IBM (International Business Machines Corporation) Cloud Services such as IBM Watson Assistant. Several intents such as searching for nearby restaurant, accommodation or tourist attraction places are configured inside the Watson Assistant. In order to manage the conversational flow between user and chatbot, a dialog skill is developed for the assistant using the created intents and entities so that it can response based on defined patterns.

For recommender system module, it is required so that the chatbot able to customise the user response based on their interests and thus recommend them with their preferred places. To do so, a machine learning technique called content-based

filtering is applied to develop a model of the recommender system. Content-based filtering, also known as Item-Item filtering is a way of recommendation based on what the user likes previously. It will study the user behaviour from time to time and forecast the items which are more likely to be accepted also by the same user.

A more detailed explanation can be found in Chapter 3 later.

1.6 Highlight of Achievement

- Able to get tourism information about Kampar easily.

The user manages to know the operation time, address, images and rating of the nearby restaurant, accommodation, and tourist attraction.

- Able to receive personalised response from the KamparBot based on different user interests.

The users with different preferences manage to receive different answers which are customising personally for themselves from the same chatbot.

- Able to communicate with the chatbot with local slang (localisation) without affecting the results.

Localisation of the chatbot have been done in order to allow the chatbot to understand the user query correctly even with local slang.

1.7 Report Organization

For this report, there are 7 chapters in total. First of all, chapter 1 provides a brief explanation on the problem statement and motivation of this project, its scope, objective, and highlight of achievement, as well as the proposed approach used to achieve the objective given.

In Chapter 2, five articles were reviewed in total. A study was conducted on these previous works to find out the importance of chatbot in the market, the strengths and weaknesses of previous works, and also the improvement to be made.

Chapter 3 covers all the system design of this project. Several diagrams such as system architecture, system flowchart, and use case diagram were illustrated inside this chapter together with the explanation. All the system functionalities developed are also discussed in this chapter.

Chapter 4 provides an overview about the methodology and the tools used in this project. Project timeline and the user requirements are also included in this chapter.

Chapter 5 discusses the system specifications of this project. In this chapter, there is a brief explanation on how the design works to achieve the objectives proposed.

Chapter 6 explains the implementation and testing of each system functionality in this project. The challenges encountered during the implementation phase are also discussed in this section.

Lastly, chapter 7 wraps up the report by providing a conclusion to summarize the project with review, contributions, and the future work.

Chapter 2 Literature Review

2.1 Artificial Intelligence Marketing: Chatbots

According to the work done by (Arsenijevic and Jovic, 2019), artificial intelligence has become a popular topic either in the scientific community, or the public. It can be seen as a marketing tool which allows the marketers to generate highly personalised customers' experiences and solve their problems efficiently. In this paper, a survey of respondents' online habits and behaviors when using various communication platform was conducted with a focus on chatbots to explore the potential of artificial intelligence in marketing. In the market, chatbots can be separated into different types, which are rule-based and AI-based. However, it is interesting to find that rule-based chatbot is actually a subset of AI-based chatbot. Based on the analysis of the survey conducted, this paper claimed that most of the problems faced by users on the Internet are insufficient features of search option, difficulty to reach the desired information on the website and the organization's non-responsiveness to user queries. Nevertheless, it is believed that the chatbots manage to address this kind of problems and meet the needs of the users. The result showed that implementing chatbots in marketing brings lots of benefits to the community, such as ease of use, fast retrieval of information and 24/7 availability. With the development of technology today, the availability of artificial intelligence and its application is expected to be increased rapidly in every sectors of the economy. Thus, the organizations should consider the advantages as well as the market value of artificial intelligence and chatbots, especially in facing the difficulty to communicate with customers.

Weaknesses / Limitations

For simple questions, chatbots are able to feedback the users with quick response. However, when it comes to more complex questions or professional answers, chatbots might not be the first choice of the communication channel used. In addition, some of the users scared of getting wrong information from the chatbots, and this may lead to the low acceptance of artificial intelligence in the community currently. Besides that, in order to discover the potential of artificial intelligence in marketing, a huge amount of data is required, which is impractical in most of the circumstances.

2.2 Analytical study on use of AI techniques in tourism sector for smarter customer experience management

According to the work done by (Dey and Shukla, 2020), the tourism sector nowadays has experienced a significant change in which the AI technologies, such as virtual reality (VR), automation, machine learning and image processing algorithms are rapidly adapted to this field in order to make tourism smarter and dynamic.

Recent years, the tourism sector in India is emphasizing on the AI applications, such as chatbot, recommendation engine and automation of services. The main problem specified in this paper is that in order to get the competitive advantages from other competitors in this sector, customer experience and satisfaction is very crucial. Thus, customization based on customers' preferences has been applied by the tourism factory nowadays to attract more tourists. To achieve personalization purpose, several algorithms such as support vector machine and decision trees are used to train the model so that the recommendation engine provides customized services to the customers according to their past data, such as their recent visited places.

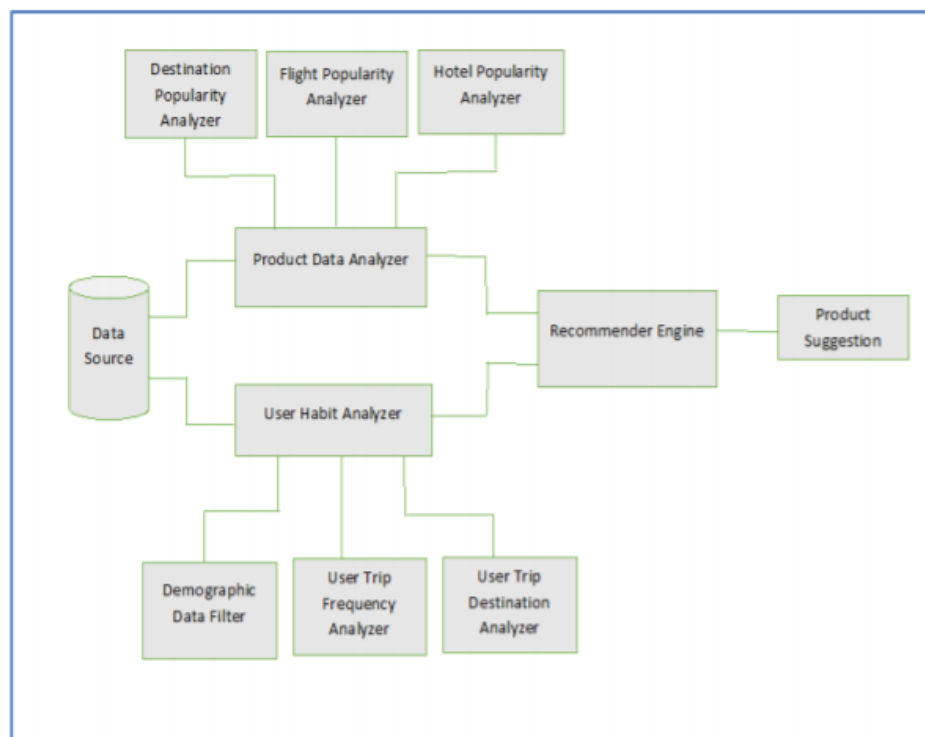


Figure 2.2.1: Recommendation Engine for Tourism.

Weaknesses / Limitations

For support vector machine, it has several limitations in training model. Firstly, it does not perform well when the datasets of the recommender system is large. Besides that, if there are many noises in the datasets, the performance of support vector machine will be greatly affected and declined. In addition, decision trees algorithm used in this paper is also not suitable as they are relatively unstable and inaccurate compared to other predictors such as s. A small change made in the data might lead to a huge change in the structure of decision trees after that. Hence, it is impractical to be implemented in the real-world application.

2.3 Chatbot-based Tourist Recommendations Using Model-based Reasoning

The work done by (Nica, Tazl and Wotawa, n.d.) is a study that emphasizes on enhancing the adaptivity of chatbots in the context of recommender systems. This article claimed that in order to recommend the customers with relevant tourism sights, the chatbots has to interact with the users frequently until their preferences are discovered. However, sometimes it is difficult for the chatbots to get enough preference information from the users. In this case, the chatbots may not be able to generate appropriate recommendations to the users. Besides that, that chatbots may not be able to control the number of recommendations to be displayed to the users due to inconsistent insufficient information on the users' preferences. Thus, it is necessary to narrow down the scope by asking the users to remove irrelevant preferences or rank the preferences according to their importance. In order to solve these issues, Shannon's information entropy was proposed for selecting new preferences from users. Meanwhile, model-based reasoning was proposed also in order to identify the main reason of inconsistencies of the preferences information. At the end, both of them will be combined together in a single recommendation algorithm, which is able to further improve user experience during the human computer interaction.

Weaknesses / Limitations

The limitation of this recommendation algorithm is that there is lack of descriptive knowledge in the model-based reasoning domain currently. Hence, this method is not mature enough to build a recommender system for tourists. Besides that, the Shannon's information entropy does not take into consideration of both semantic (meaning) and pragmatic (context) information. Therefore, the recommender systems to be developed may not be accurate.

2.4 HR Based Interactive Chatbot (PowerBot)

According to the work done by (Tadvi, Rangari and Rohe, 2020), when the employees wish to ask some questions regarding the Human Resources (HR), such as salary information, reimbursement and leave information, they have to make an appointment in order to meet the HR personnel physically. This is very tedious for the employees as they might waste their time in working. Hence, a HR based chatbot is proposed in this paper in order to create a human-like conversation to assist the company employees with their HR requests 24/7 in real time, reduce their waiting time as well as to provide personal experience for them. Thanks to the advancement in machine learning and artificial intelligence techniques, the practicality of integrating chatbot into any application have increased significantly. In this proposed system, any questions related to HR can be asked to the chatbot in natural language which is more convenient to the users. Chatbot will understand what the employee is asking and respond appropriately based on the conversational context. If there is a need to call or visit HR department for assistance, quick response will be given by the chatbot. In this article, the operating system (OS) selected is Windows due to its user-friendly and robustness. For developing the HR chatbot, Python is chosen in order to perform chat pattern comparison. If the data is matched with stored inputs, respective response will be returned to the users.

Weaknesses / Limitations

The limitation of this system is that the chatbot proposed is linguistic based (rule-based), which practice if/then logic to create the conversational flows only. Unlike AI chatbot which practicing the machine learning technique to learn and improve itself over time, rule-based chatbot is less capable, sophisticated and flexible. The order and synonyms of the keywords are predefined and only pattern matching is needed for reacting to users' question. Furthermore, the requests with similar meaning will always receive same answer from the chatbot. Hence, it is more like an interactive FAQs for the users.

2.5 AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)

The work of (Patel et al., 2019) is a study about the development of a chatbot called Unibot, which specifically designed to deal with students' questions regarding the university information, such as tuition fees, admission fees or academic calendar. Traditionally, students have to visit the university themselves in order to query this information, which is very tedious for them. Thus, this system aims to provide quick and appropriate reply to the students based on what they asked without the needs to visit university. For this system, the back end is written in Hypertext Preprocessor (PHP) language, whereas the front end part is developed using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and jQuery, whereby Ajax is responsible for calling and receiving response from the PHP file. Meanwhile, jQuery is mainly used to show the corresponding answers returned by chatbot to the users. Unlike other static chatbot, this project proposed a totally new algorithm for chatbot, which is more efficient and require lesser memory as well as minimum database hits. In short, this algorithm will pre-process the user query, by validating its spelling and eliminating irrelevant words. If the user misspells words in message, then the chatbot will provide suggestions to the user and perform the operation based on selected option. After that, a connection will be established to the database and the keywords extracted will be matched with the words stored previously using Regular Expression method. When there are multiple results found, users have to make their choices again and select the most precise question. Instead of providing default answer to the users, Unibot displays related options so that it can get to know more accurate question from users.

Weaknesses / Limitations

When the students query their questions to the chatbot, data pre-processing will be carried out to check the words' spelling and remove unwanted words. However, this process may take longer computation time and cost more. In addition, this proposed system only offered its users with text input method, rather than a combination of text and voice. The students only can interact with the chatbot by typing their questions manually. Hence, it is not convenient to the users in this scenario. Yet, these limitations can be improved later on in the project proposed, in which both text and voice input methods are provided for the users to query the chatbot.

Chapter 3 System Design

3.1 System Architecture

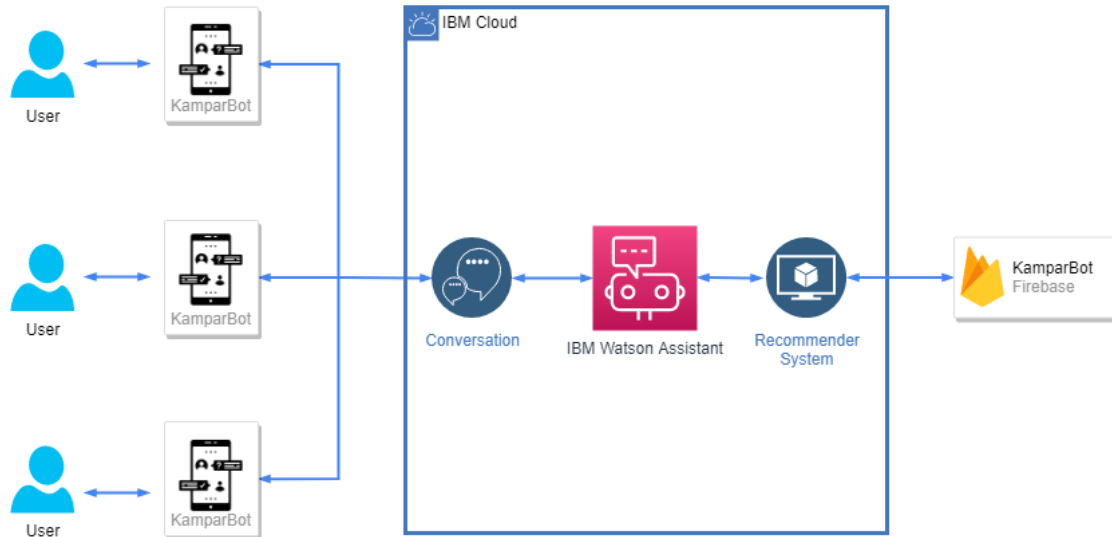


Figure 3.1.1: System Architecture Diagram.

Figure above illustrates the system architecture of this project. As stated before, KamparBot is a mobile application of chatbot for Kampar tourism. In this project, IBM cloud services were implemented to create the chatbot. IBM Watson powered by IBM contains a lot of cloud services such as IBM Watson Assistant, Watson Discovery, Watson Machine Learning and Watson Studio for developing AI solutions in the business. To handle the conversational service between users and the chatbot, Watson Assistant was chosen. Watson Assistant is a pretrained AI chatbot cloud service which provides the clients with efficient, consistent, and accurate answer. By simply calling the IBM Watson API (Application Programming Interface) using the service credentials provided, the clients manage to integrate the Watson Assistant across multiple platforms, such as Messenger, Slack and web or mobile application. With Watson Assistant, the chatbot can easily learn from conversations, enhancing its ability gradually to resolve problems and returning response to the customers without waiting for long time.

To achieve the personalisation of customer response, a recommender system is required in this project. Thus, a content-based filtering recommender model was implemented. Basically, content-based filtering will recommend the items based on its

features. For example, for a movie recommender system, the item features can be thought as the genres of the movie. Given the users' previous feedback and rating on these items, the content-based filtering would further recommend other similar items which is close to what the user likes.

Lastly, Firebase Realtime database is used to store all the information of the chatbot, such as the tourism places, user personal information and the user rating on respective places. It is a cloud-hosted No-Structured Query Language (NoSQL) database which is free and easy to use. All the data stored will be synced in real time across all the clients access to the application even if the application goes offline.

3.2. System Flowchart

Visual Paradigm Online Express Edition

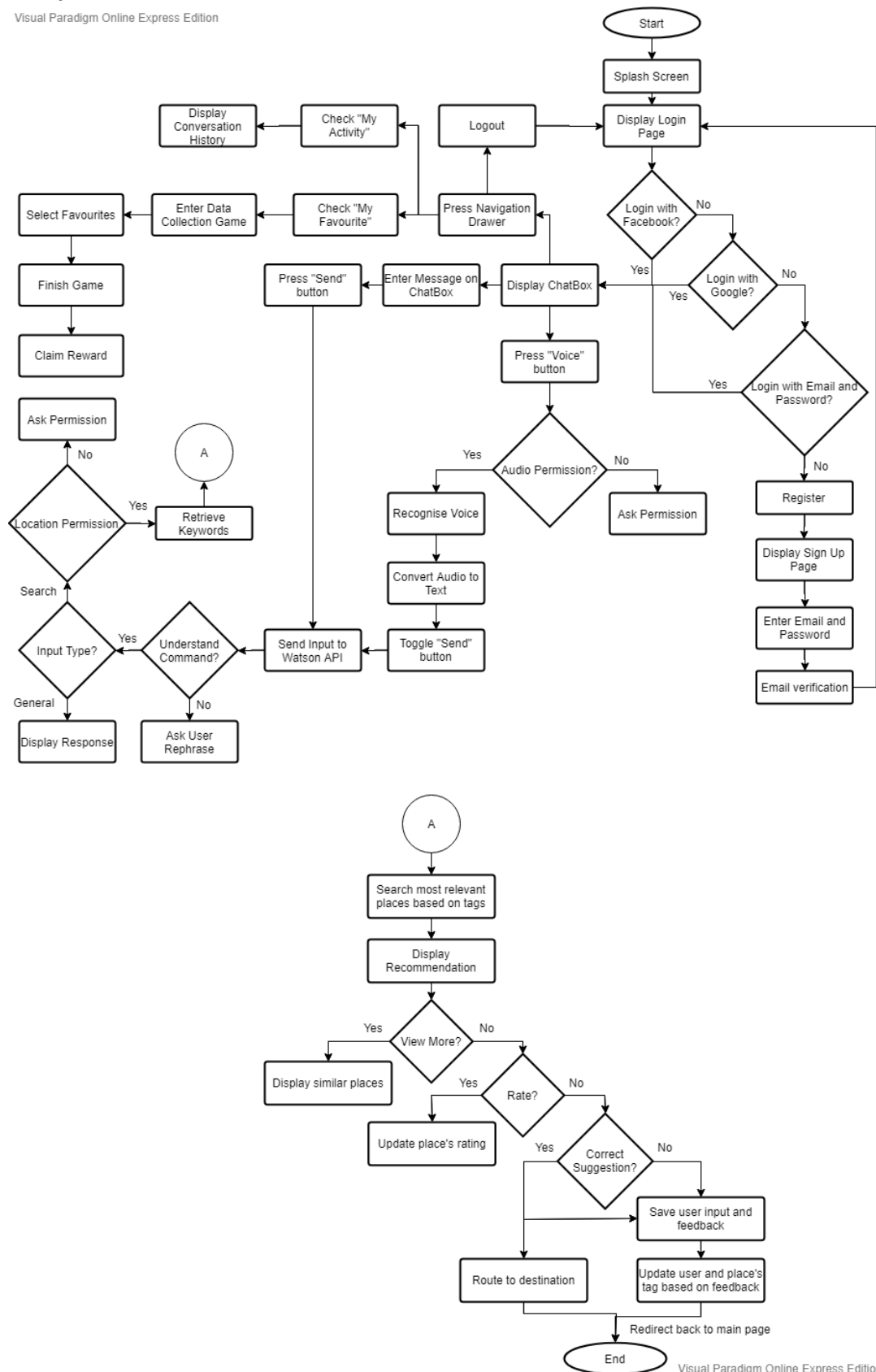


Figure 3.2.1: System Flowchart.

The figure above illustrates the overall system flow of the application. When the user runs the mobile application, a splash screen with KamparBot logo will be displayed first, followed by the login page. On the login page, the user can choose to sign in with a Facebook account, Google account, or email address and password. If the user does not have an account, they can proceed to the sign-up page. An email verification is needed upon registration of a new account to authenticate the user's identity.

Once signed in successfully, the user will be redirected to the main page and can now interact with the chatbot, by either texting or speaking. For voice input, microphone permission needs to be allowed by the user to enjoy this feature. After sending user input, the message will be retrieved by the Watson API for interpretation and translation. If the chatbot does not understand the command, it will ask the user to rephrase the input. Meanwhile, if the questions asked are general, the chatbot will reply immediately to the user. For those inputs that require searching in the database, the system will access the user's location (need to provide location permission) and extract the keywords from the user input. These keywords will be used to search for the most nearby and relevant places based on the tags associated and recommend it to the user. The user may choose to route to the destination, view more similar places, rate the place, or feedback the correctness of the response given by the chatbot. These feedbacks will be stored into the database for further enhancement of the chatbot.

Other than that, there is also a navigation drawer on the main page, which can redirect the user to "My Activity", "My Favourite" or even logout from the application. All the conversation history will be stored inside "My Activity" tab. In "My Favourite" page, the user may play a game to earn rewards. The user just has to select his favourite things from three categories: Lodging, Foods, and Tourist Attractions, and he will be able to claim his reward. Basically, this game is designed to gather the user's interests in these three categories in order to label the users with different tags for customising the response they received after that to achieve the personalisation purpose.

3.3 Use Case Diagram

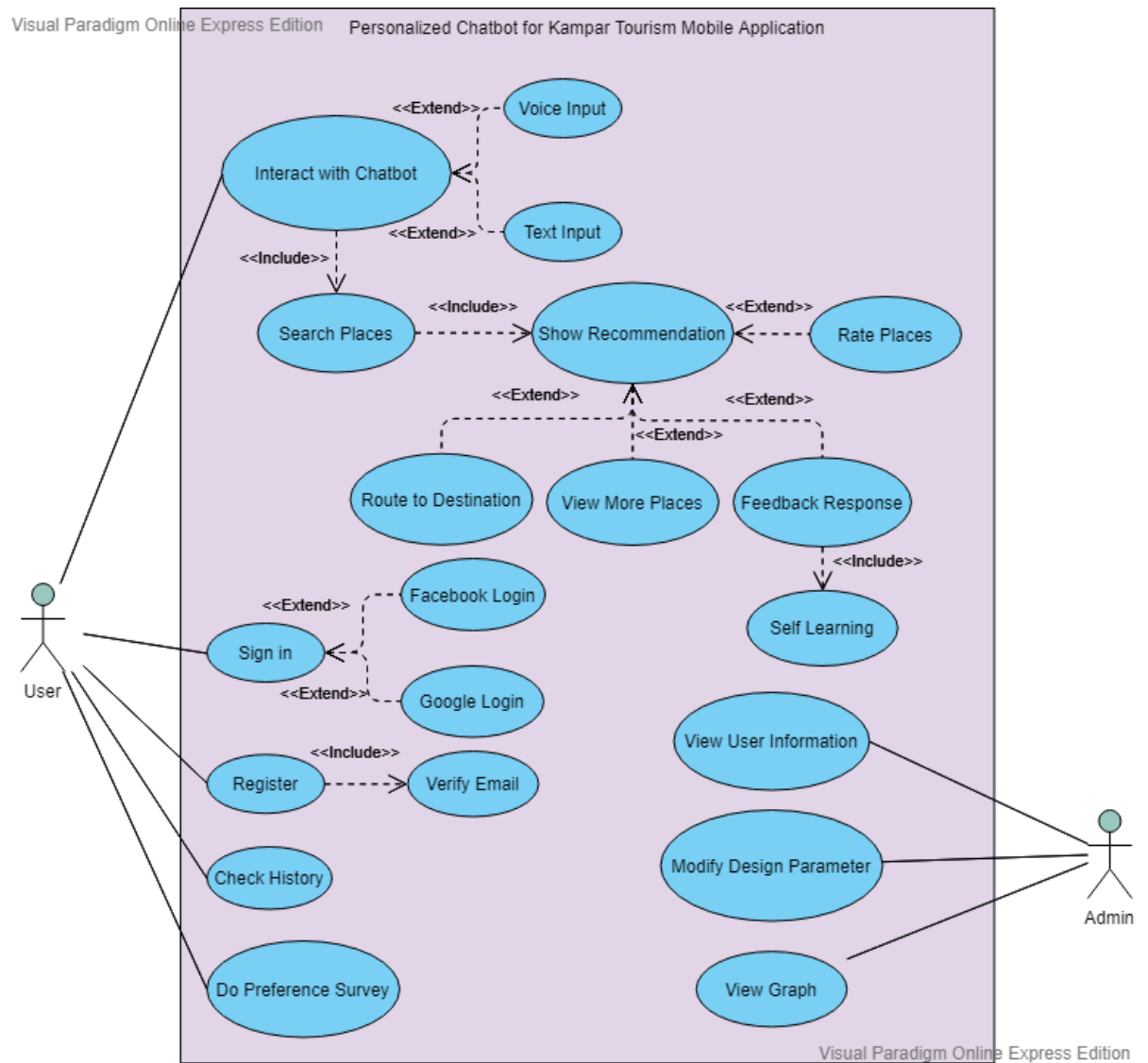


Figure 3.3.1: Use Case Diagram.

3.4 System Functionalities

3.4.1 Navigation Drawer

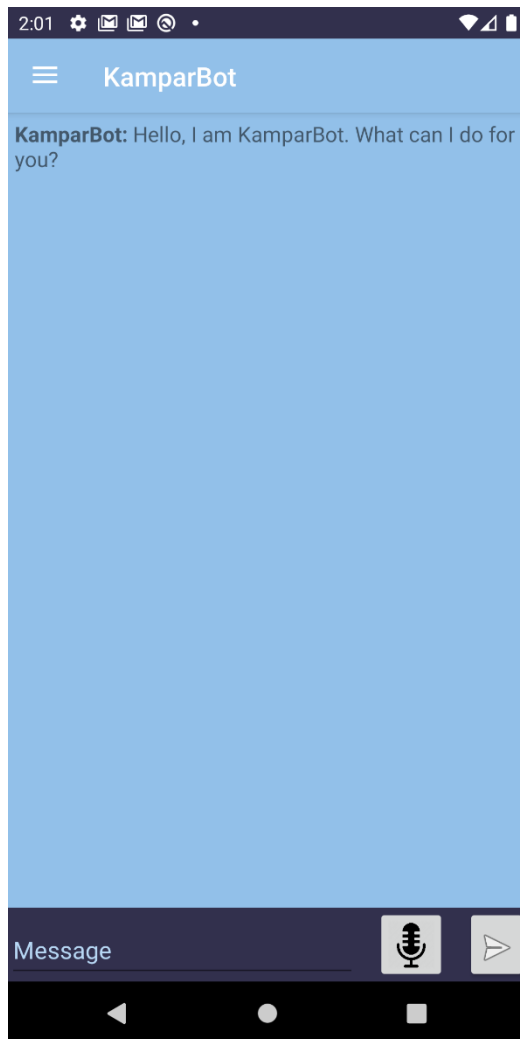


Figure 3.4.1: Main Page.

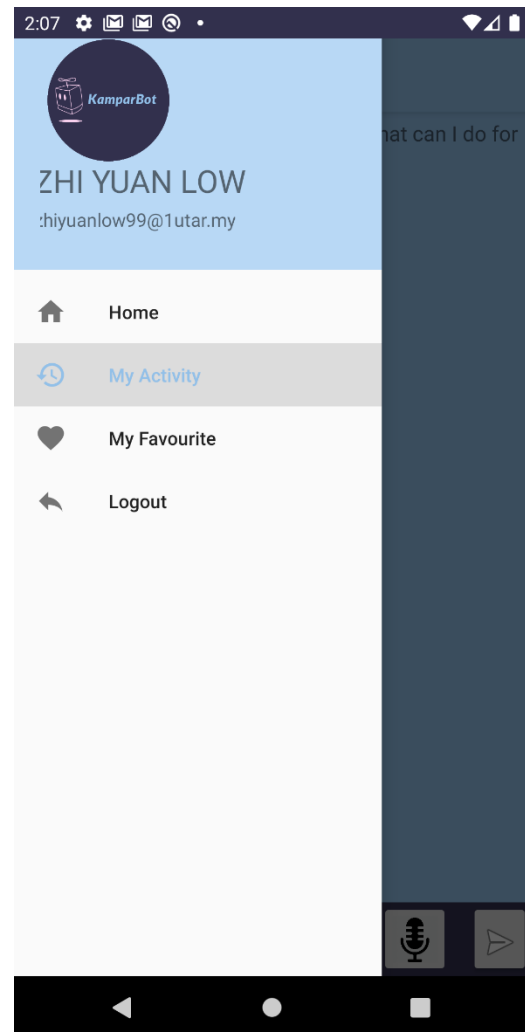


Figure 3.4.2: Navigation Drawer.

Figure 3.4.1 shows the main page of new version of KamparBot. A navigation drawer was added and located on the top left corner of the main page. After clicking on the navigation drawer icon, the drawer will be expanded and there are four menus inside, which are “Home”, “My Activity”, “My Favourite” and “Logout”. For chatbot, it will be located at “Home”. Meanwhile, the “Logout” menu allows the user to logout of the application.

3.4.2 “My Activity” Page

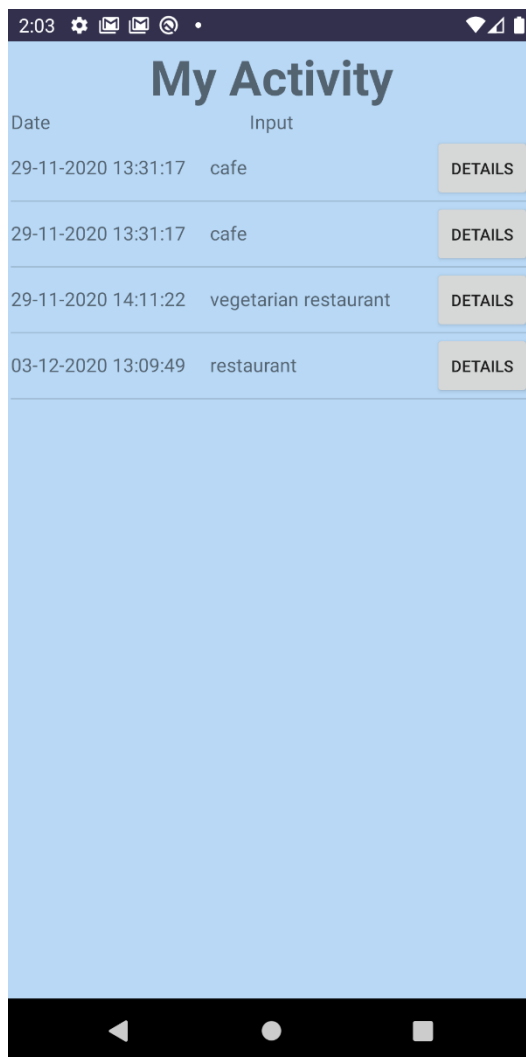


Figure 3.4.3: “My Activity” page.

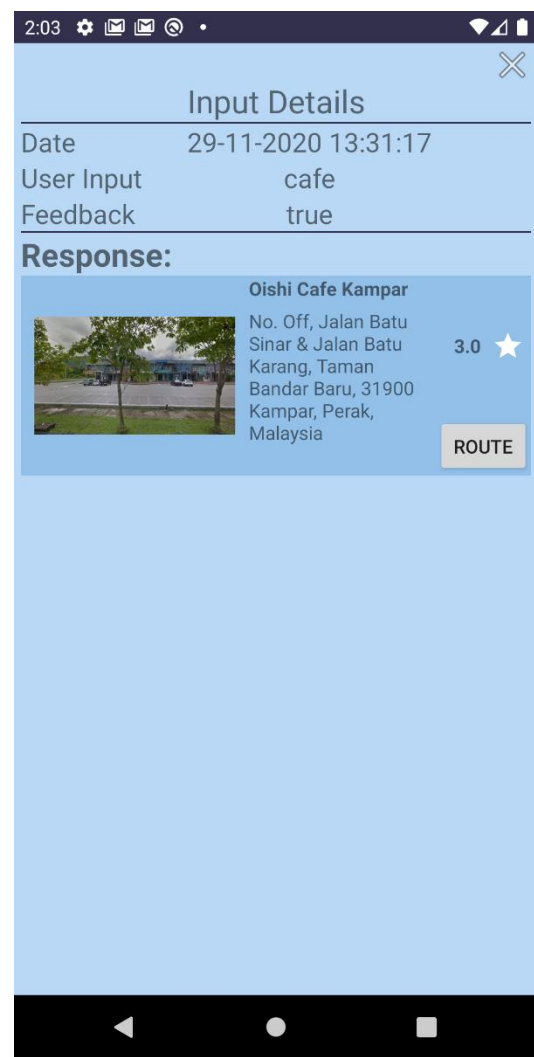


Figure 3.4.4: User Input Details.

Figure 3.4.3 illustrates the page for “My Activity” menu. Basically, this page will display all the user’s chat histories with the chatbot. If the user clicks on “Details” button, the system will display more information as shown in Figure 3.4.4. The feedback field shows the user’s previous feedback (true or false) on the chatbot’s recommended place. This field is very important as it will be used to train the chatbot for learning the user profile. Besides that, there is also a route function provided for the user if he wishes to revisit the place again.

3.4.3 “My Favourite” Page

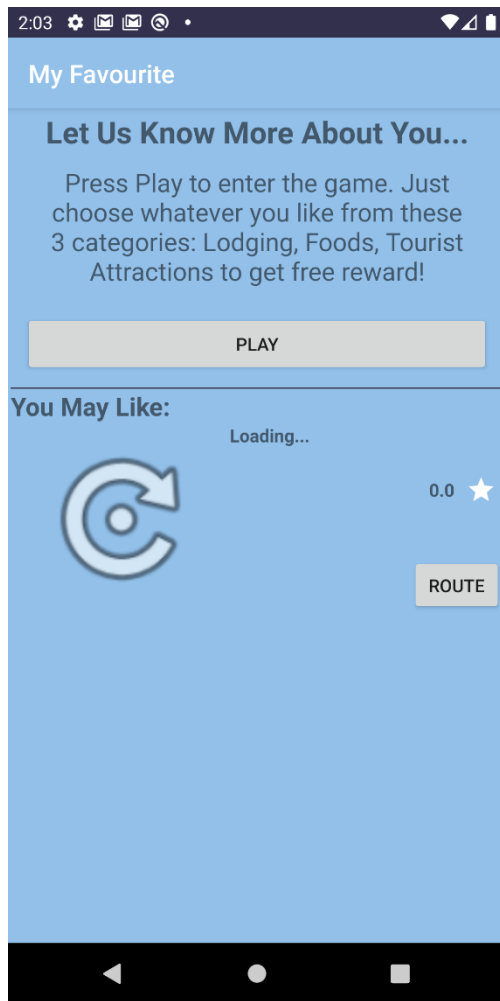


Figure 3.4.5: “My Favourite” page.

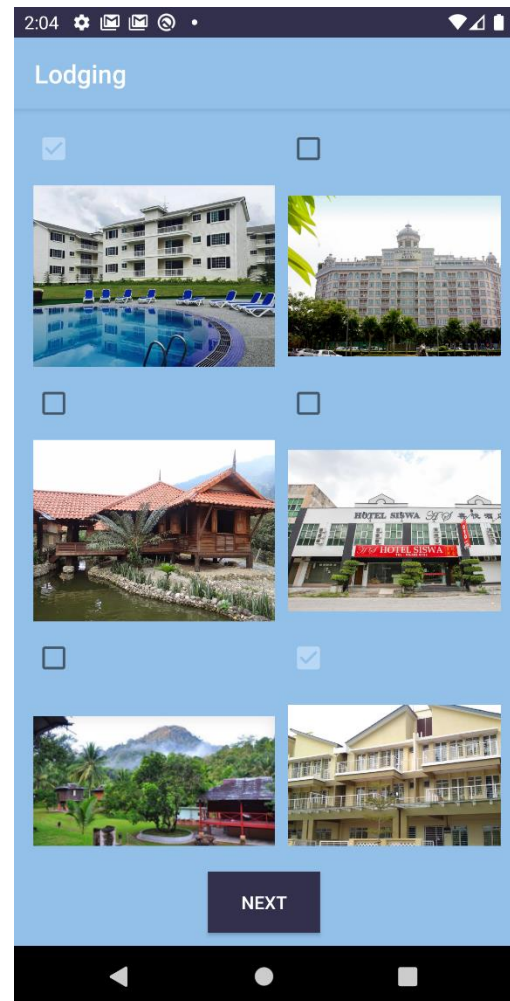


Figure 3.4.6: Game Page 1 - Lodging.

Figure 3.4.5 illustrates the page for “My Favourite” menu. This menu is designed to get user’s favourite things for three categories (Lodging, Foods, and Tourist Attractions). Besides that, there is also a recommendation field inside the menu, where the user can find a list of tourism places recommended after the user profile was learnt by the chatbot using the content-based filtering.

To motivate the user to share their interests, some rewards will be credited to the user after finishing the game. There will be six pictures for each category inside the game. As shown in Figure 3.4.6, the user is required to choose any two of the six pictures he likes the most. All these pictures contain respective labels and tags in database. In this case, the tags will be hostel, resort, and hotel.

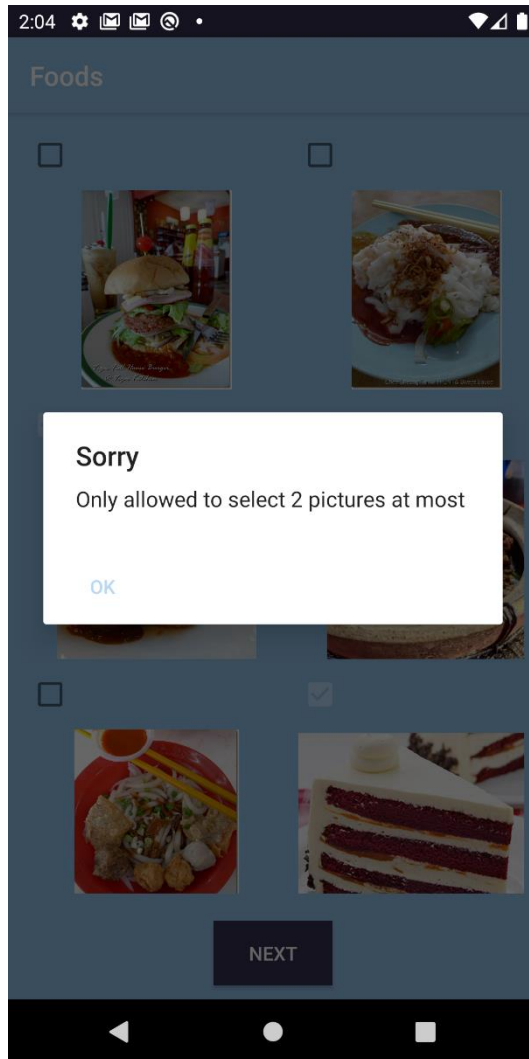


Figure 3.4.7: Game Page 2 – Foods

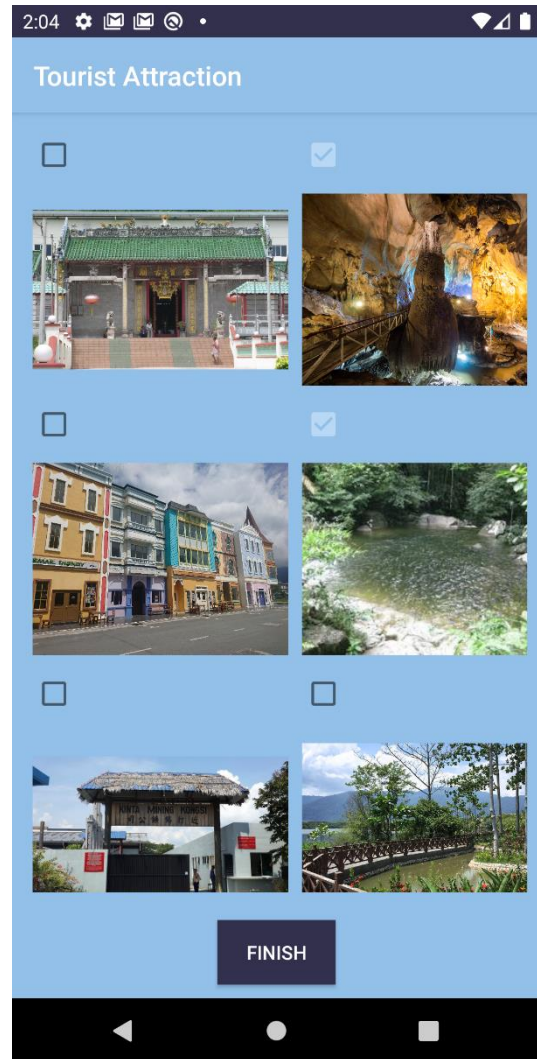


Figure 3.4.8: Game Page 3 - Attractions.

Figure 3.4.7 shows the error message when the user selected more than two pictures under a category. By limiting the maximum selection, the chatbot can get to know the user's preference more easily and accurately. This can also prevent the user from selecting all choices directly, which will cause the user's actual interest to be unknown.

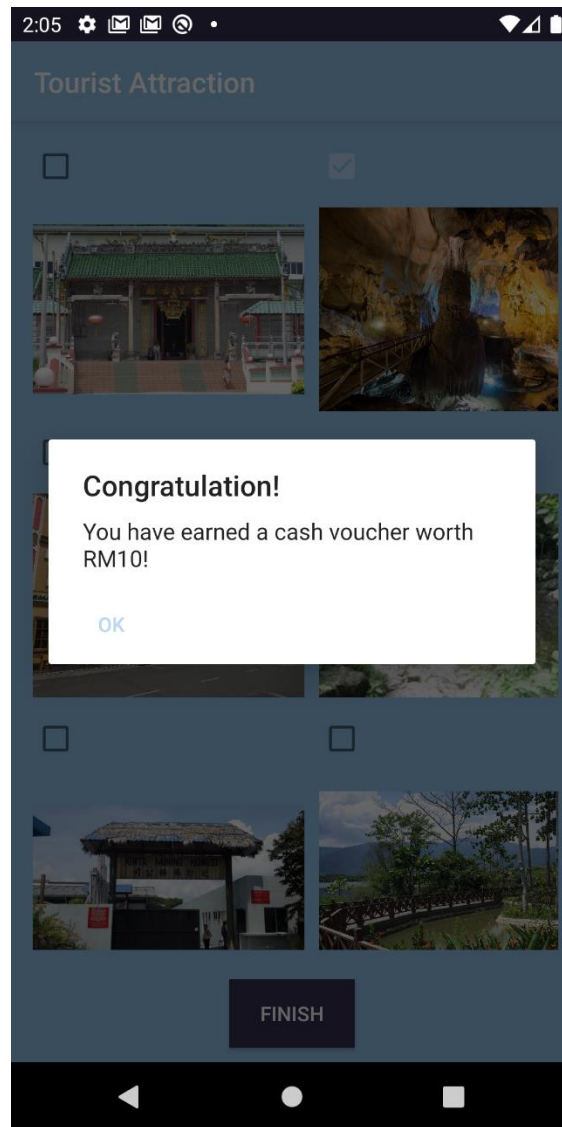


Figure 3.4.9: The End of Game.

As shown in Figure 3.4.9, a pop-up message will be shown after the user completed the game. For now, it is just a message telling the user that a cash voucher is credited as reward, since it is not the main focus of this project. Hence, it will be a future work if the budget is sufficient.

3.4.4 Recommender System

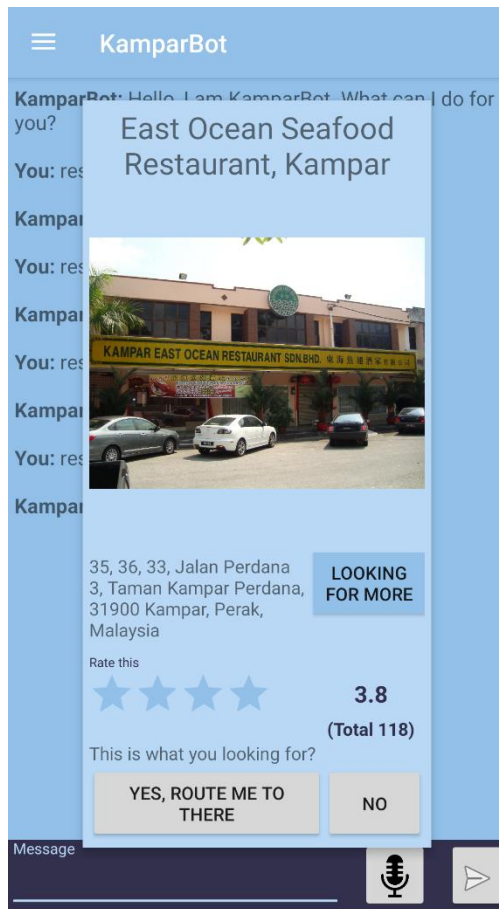


Figure 3.4.10: Result 1.

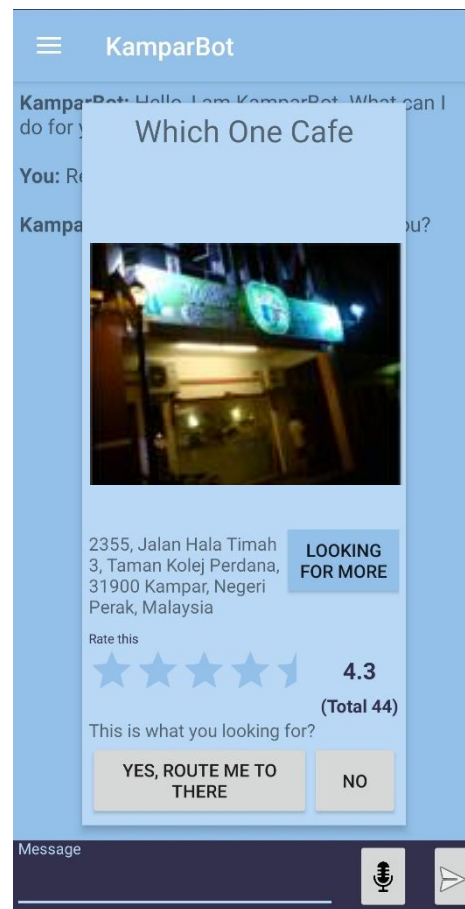


Figure 3.4.11: Result 2.

Figures above illustrates the results when different users searching for nearby restaurant. For result 1, the user had been tagged with “local foods” according to his selection in the game. Meanwhile, the user in result 2 had been tagged with “café”. The comparison between the results shows that the different users manage to receive different recommendations based on their interests analysed by the chatbot using the content-based filtering technique.

Chapter 4 Methodology and Tools

4.1 Methodology

The methodology chosen for this project is prototyping approach. It is a form of iterative development process, which requires frequent feedback from users to refine the application until a complete system is developed. Instead of determining all the system requirements at initial stage first before starting any development, prototyping emphasises more on rapid development and continuous refinement of the system. For this methodology, user involvement is very important, since they will be the person who try out the prototype themselves as well as comment on it, so that the developer able to further improve the system, and eventually comes out with a perfect system which satisfies the targeted customers. There are several phases in prototyping approach, which are requirements gathering, quick design, build prototype, user evaluation, refining prototype, implementation as well as maintenance.

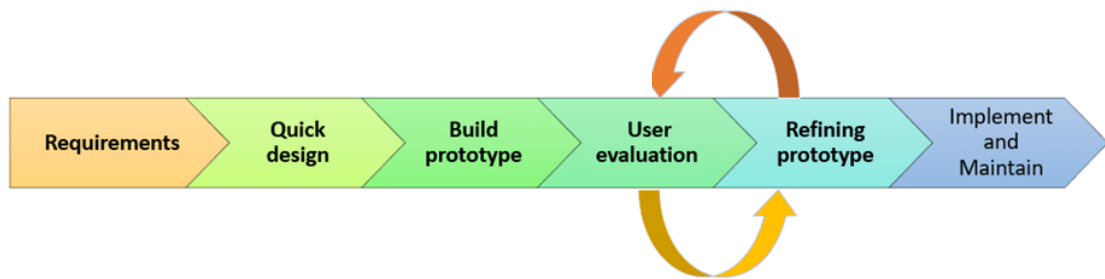


Figure 4.1.1: Prototyping approach.

- Requirements gathering

Identify the desired requirements of the project, such as the user registration, login, personalised chatbot and suitable recommender system.

- Quick Design

Design the user interface, database, and the system architecture.

- Build Prototype

First version of system prototype with minimal number of useful functionalities is developed for user reviewing.

- User Evaluation

At this stage, users highly involve in evaluating the system features by testing the system prototype themselves. If there is any dissatisfaction, user may point out to the developer and ask for enhancement.

- Refining Prototype

An iterative process whereby the system prototype will be modified according to what the users actually want, until the system is completely acceptable by users.

Any improvements done will be added to the system in next iteration.

- Implementation and Maintenance

Final version of the system prototype is implemented and delivered to the market. The process will move to maintenance phase now.

4.2 Tools / Technologies Involved

Platform

1. Android

Android is an open-source mobile operating system which is famous nowadays with the most users. Hence, it is chosen as the OS of the mobile application to be developed in this project, as it is free of charge and having a high percentage of mobile users.

Programming Language

1. Java

Java is an open-source object-oriented programming language, which is popular for developing software. Currently, Java is ranked the second most well-known programming language, just after the Python (statisticstimes.com, 2020). In

addition, Java application able to run in any devices if the respective Java Virtual Machine (JVM) is installed previously. In this project, Java is chosen for developing the mobile application.

2. Extensible Markup Language (XML)

Android's XML is a markup language used to rapidly design the User Interface (UI) layout of the mobile application in Android Studio.

Software

1. Android Studio

Android Studio is an Integrated Development Environment (IDE) for Google's Android operating system, designed mainly for developing Android-based mobile application. It supports most of the programming language, such as Java, Kotlin and C++. In this project, Android Studio is chosen as the platform for developing the chatbot mobile application in Java.

2. Visual Paradigm

Visual Paradigm is a Unified Modeling Language (UML) diagramming tool that helps the developer to transform the actual system requirements into system design. In this project, Visual Paradigm will be used to draw UML diagrams, Entity Relationship Diagram (ERD) and also Gantt chart for system design.

3. Google Speech-to-Text API

It is an API used to interpret and translate the audio to text. In this project, this API will be implemented on the chatbot for voice recognition purpose.

4. Watson API

It is an API used to provide the conversation service of chatbot in any application. In this project, it will be mainly used for handling the conversational flow between the user and the chatbot.

Database

1. Firebase

A cloud-hosted database used to store data as Javascript Object Notation (JSON) format. The data is synchronized to every user connected in real time. In this project, Firebase will be used to store all the information of the users, and chatbot.

Hardware

1. Laptop

Model: Acer Nitro AN515-44

Processor: AMD Ryzen 7 4800H with Radeon Graphics 2.90GHz

GPU: NVIDIA GeForce GTX1650

RAM: 16GB

System Type: 64-bit Operating System, x64-based processor

2. Smartphone with Android OS

Model: Huawei nova 5t

Processor: Huawei Kirin 980

RAM: 8GB

Android Version: 10

4.3 User Requirements

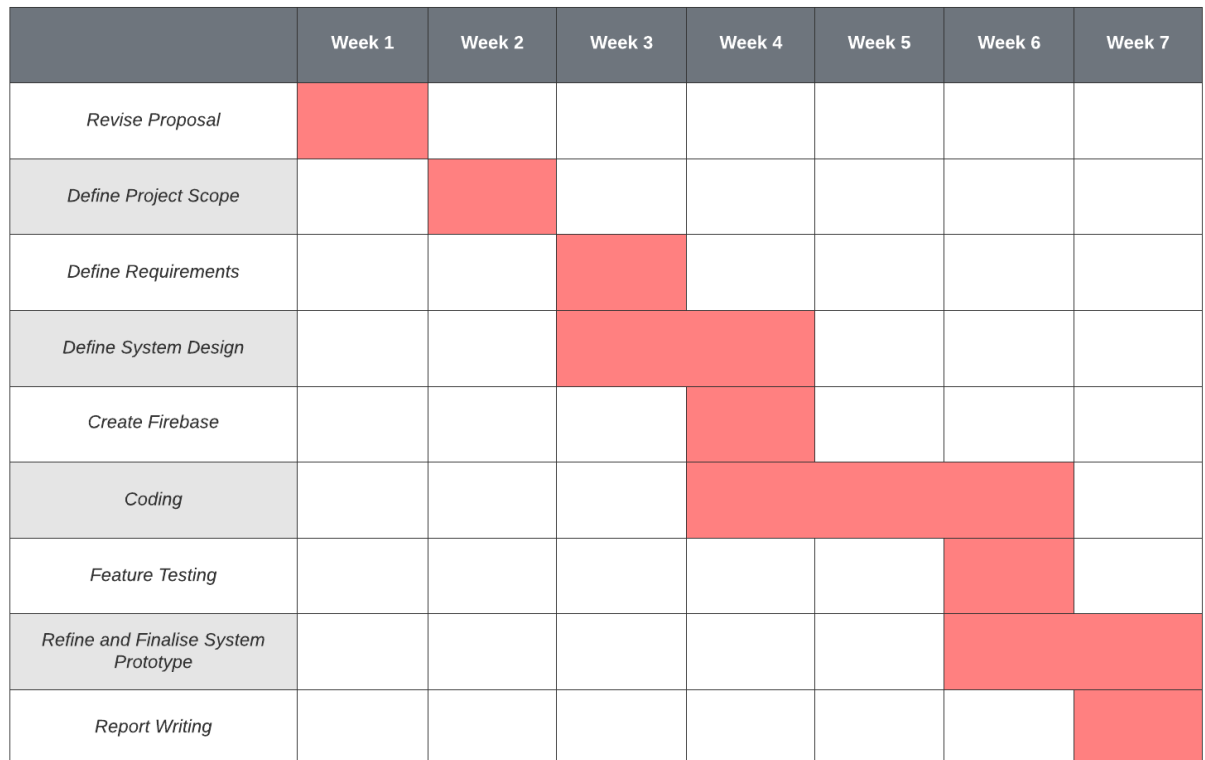
Functional Requirements

1. User must be able to register new account.
2. User must be able to sign in with email address and password.
3. User must be able to sign in with Facebook account.
4. User must be able to sign in with Google account.
5. User must be able to interact with chatbot, either via text or voice.
6. User must be able to view the response given by chatbot.
7. Different users must be able to view different responses given by chatbot based on their preferences.
8. User must be able to route to their destination.
9. User must be able to view more similar places.
10. User must be able to rate the place.
11. User must have the ability to feedback the correctness of response.
12. User must be able to view conversation history.
13. User must be able to play games for determining interests.
14. User must be able to logout.

Non-functional Requirements

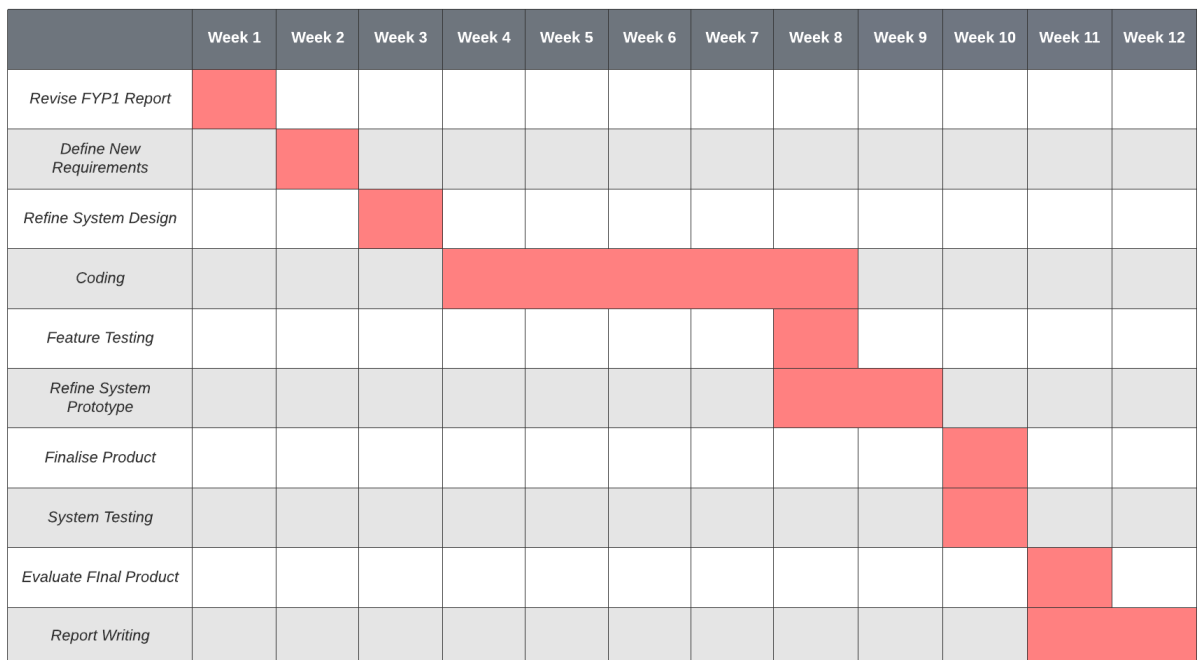
1. User must be able to run the mobile application on any Android device.
2. User must be able to get correct response from the chatbot.
3. User must be able to receive response within 5 seconds.
4. User must be able to use the mobile application without crashing.
5. User must be able to view appropriate error messages when system error.

4.4 Timeline



Legend: ■ Duration

Figure 4.4.1: FYP1 timeline.



Legend: ■ Duration

Figure 4.4.2: FYP2 timeline.

Chapter 5 System Specification

To achieve the objectives proposed above, several methods were applied:

5.1 IBM Watson Assistant

As stated above, IBM Watson Assistant was chosen to develop the chatbot part of this project.

Plan	Features	Pricing
Lite	<p>Everything you need to get started, free for as long as you need it</p> <p>Up to 1,000 unique monthly active users (MAUs) chatting with your assistant Up to 10,000 messages per month</p> <p>--- Features ---</p> <ul style="list-style-type: none">- World-class conversational AI with Watson- Make your website assistant your own with Webchat - deploy Webchat in minutes, or use our fully extensible architecture- Keep your assistant up-to-date by automatically scanning existing webpages and documents or by using some of our prebuilt dialog content- Connect to any application or database with a prebuilt integration, or build your own custom integration on top of API endpoints- Create engaging user interactions using images, buttons, and more- Analyze and enhance your assistant with our analytics dashboard- Take comfort knowing your assistant is reliably hosted on IBM Public Cloud <p>--- Limits ---</p> <p>Up to 5 Skills (Dialog, Action, Search) 7 days of usage analytics Session inactivity timeout 5 minutes Services are deleted after 30 days of inactivity</p> <hr/> <p>Everything you need to get started, free for as long as you need it</p> <p>Lite plan services are deleted after 30 days of inactivity.</p>	Free

Figure 5.1.1: Subscription Plan of IBM Watson Assistant.

According to Figure 5.1.1, IBM provides its client to subscribe a lite version of Watson Assistant for free, with some limitations. However, it was good enough for this project using lite plan, since not many users and messages will be created for this assistant.

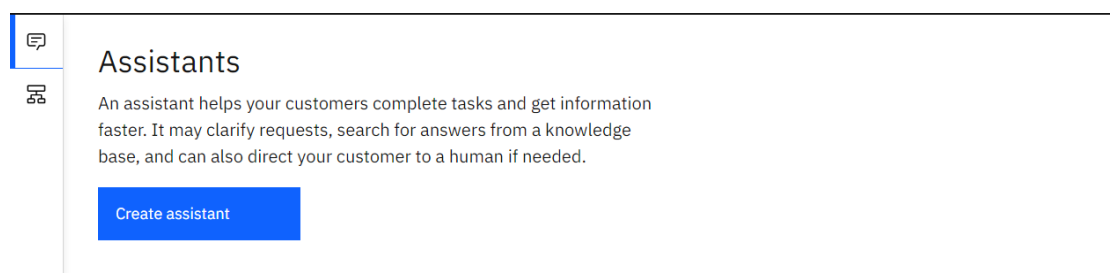


Figure 5.1.2: Create Assistant Page.

In Watson Assistant, an assistant is also known as the chatbot, which is designed to help the customers with their problems. At here, several assistants can be created.

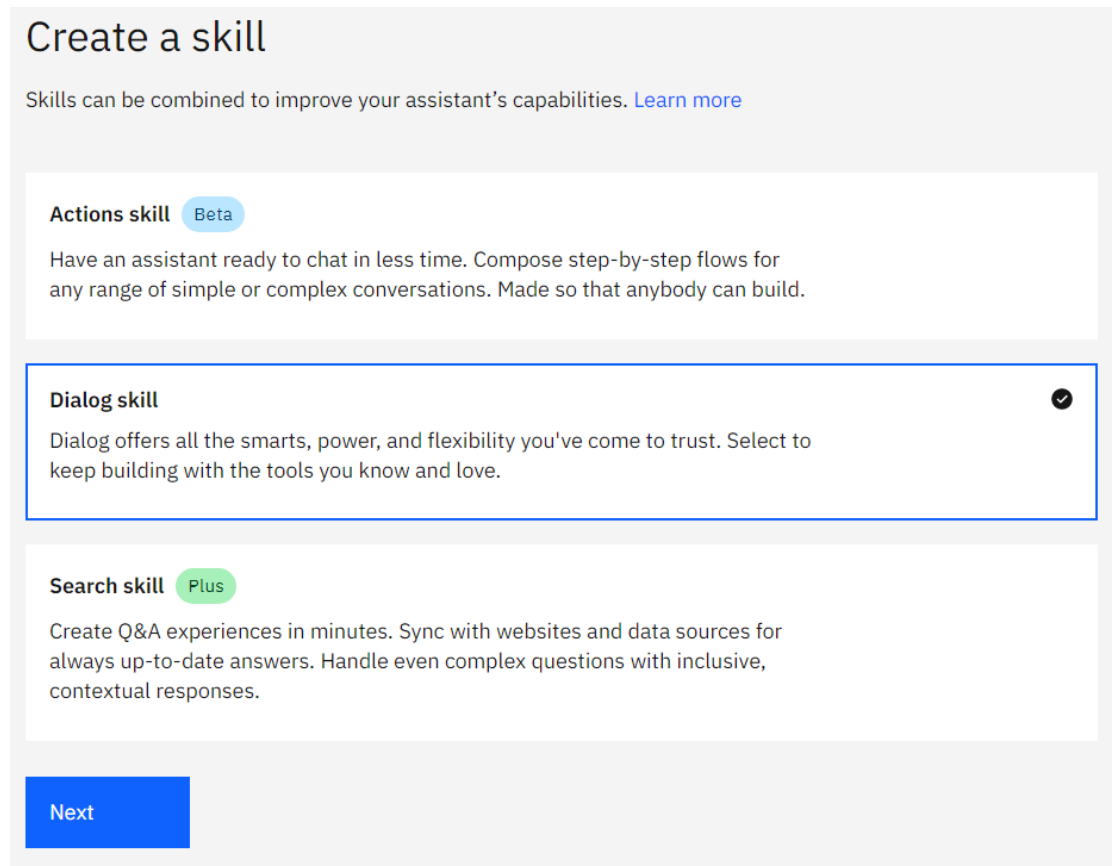


Figure 5.1.3: Create Skill Page.

Before the assistant manages to accomplish its task, a skill must be created, configured, and added to that assistant. For this project, a dialog skill is created to manage the conversational flow using natural language processing.

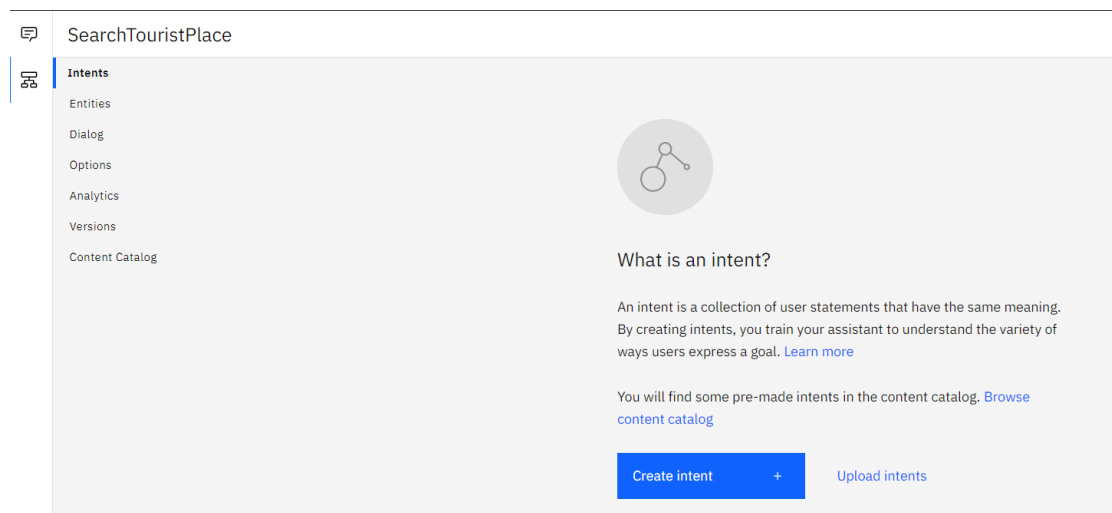


Figure 5.1.4: Intent Page.

Inside the dialog skill, there are several things to be configured. An intent can be defined as the goal of the users want to achieve. At here, a collection of the user

statements with similar meaning can be created as an intent for later training purpose. Meanwhile, an entity can be defined as a keyword or noun of the business term to be recognized by the assistant, so that it can provide respective responses to the user queries.

← | #Yes

User example

Type a user example here

Add unique examples of what the user might say. (Pro tip: Add at least 5 unique examples to help Watson understand)

Add example

☐ User examples (8) ↑

☐ Correct

☐ It's correct

☐ Sure!

☐ True

☐ Ya

☐ Yeah

Figure 5.1.5: Example of Intent #Yes with User Examples.

As shown in the figure above, an intent called “Yes” was created to express the meaning of confirmation. Several different user examples were added into the intent for the assistant to learn and understand this intent after training.

← | #Foods

User example

Type a user example here

Add unique examples of what the user might say. (Pro tip: Add at least 5 unique examples to help Watson understand)

Add example

- ☐ Foods
- ☐ Ho Chiak
- ☐ Hungry
- ☐ Kopi
- ☐ Kopitiam
- ☐ Nearby foods
- ☐ Nearby Restaurant

Figure 5.1.6: Example of Intent #Foods with User Examples.

By localising the user example of an intent, one of the objectives of this project, to allow the user to ask questions using local slang without affecting result could be achieved. As shown in the Figure above, several local slangs to express meaning of foods such as kopi, kopitiam and ho chiak can be learnt easily by the Watson Assistant without paying much effort to create a new model. For the assistant to work well with local slang, more and more local expression of an intent can be added into skill to train it.

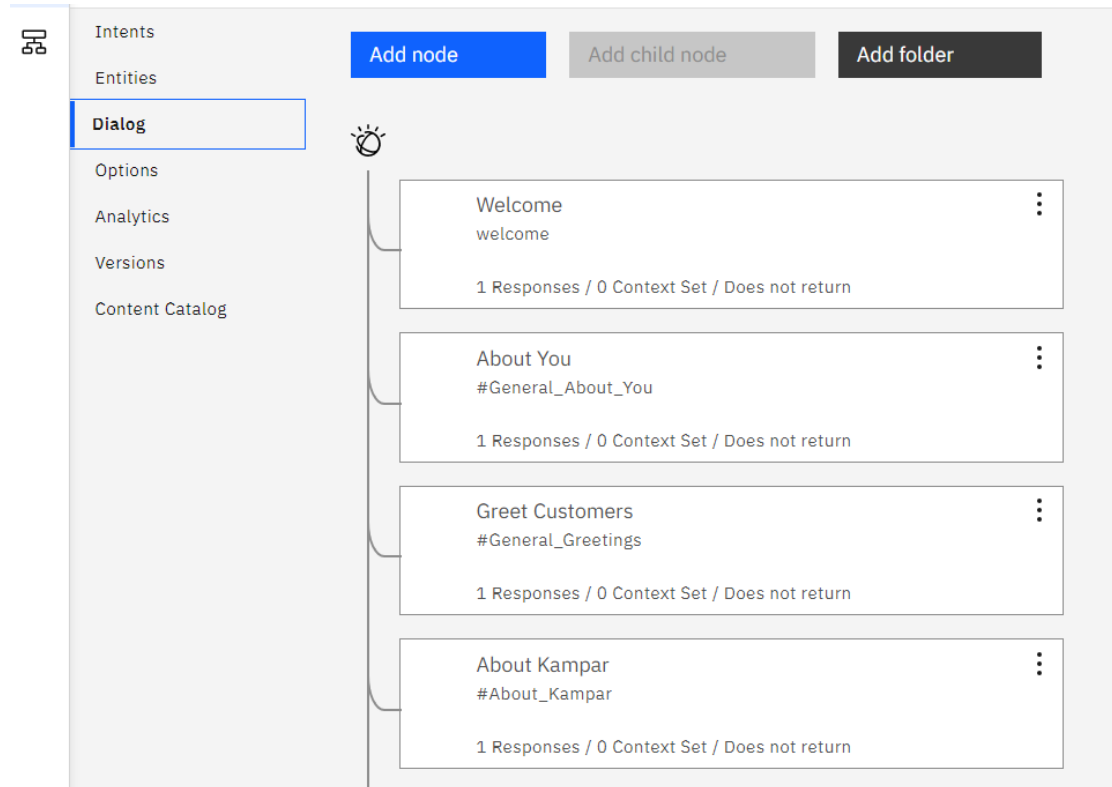


Figure 5.1.7: Dialog Tree (Conversational Flow).

Dialog tab is the focus of the dialog skill of the assistant as it defines the conversational flow between the assistant and the users. Initially, there will be two pre-defined nodes inside the dialog tree, which are Welcome node and “Anything else” node. For Welcome node, it will be triggered once the chatbot is accessed. Meanwhile, if all the nodes do not recognise any intent from the user query, “Anything else” node will be triggered. It is basically used to reply the user when the question is not understandable by the assistant.

The screenshot shows the configuration for a dialog node in Watson Assistant. It is divided into three main sections:

- If assistant recognizes:** This section contains a trigger phrase "welcome" with a trash icon and a plus sign to add more triggers.
- Assistant responds:** This section allows configuring the assistant's response. It includes a dropdown menu set to "Text", a sample response "Hello, I am KamperBot. How can I help you?", and a text area for "Enter response variation". Below this, it states "Response variations are set to sequential. Set to random | multiline" with a "Learn more" link. At the bottom of this section is a link "Add response type +".
- Then assistant should:** This section is for configuring the next action. It includes the instruction "Choose whether you want your Assistant to continue, or wait for the customer to respond." and a dropdown menu currently set to "Wait for reply".

Figure 5.1.8: Details of Node in Dialog Tree.

Figure above illustrates the configuration of the welcome node inside the dialog tree, which will be triggered once the chatbot is activated. Inside each dialog node, the client has to specify what the assistant should response when certain intents or entities are found, and when they are not found. By configuring these settings, another objective of this project can be accomplished. A chatbot which allows the user to retrieve the tourism information of respective tourism places in Kampar can be developed easily using Watson Assistant.

5.2 Content-Based Filtering Recommender System

To achieve the last objective of this project, which is to ensure that different users receive personalized responses from the same chatbot based on their interests, a well-defined recommender system is required.

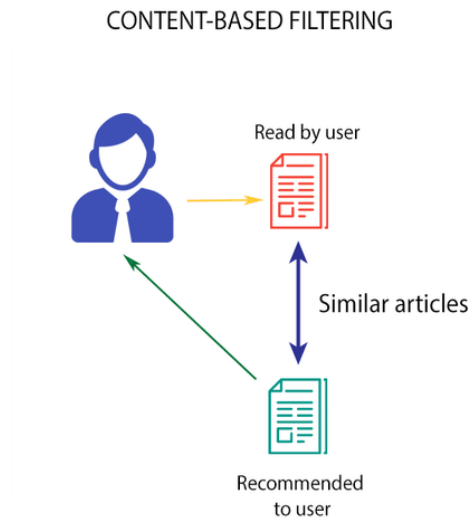


Figure 5.1.9: Content-Based Filtering.

As mentioned just now, a content-based filtering recommender system will filter the results based on item features. In this case, the chatbot should be able to recommend the users with respective tourism places based on their tags. For example, in this project, restaurant will have 3 tags, which are local foods, fast foods, and café. Based on the users' previous rating on the places they visited together with these tags, a similarity measure using dot product was calculated. With this, a weighted rating value of the user interest for each place in Kampar can be figured out easily. Eventually, the recommender system will select the top few tourism places with highest weighted value to the users.

Chapter 6 Implementation and Testing

6.1 Implementation

The main intent of this project is to develop a recommender system model which manages to customise the response based on different users' profile, and therefore improve their experience in communicating with the chatbot. As stated before, the content-based filtering model was chosen to be implemented in this project, and the important feature of the item will be the tag or label of the restaurant, accommodation, or tourist attraction place. Since the implementation of content-based filtering model was similar for either restaurant, accommodation or tourist attraction, foods recommender system was chosen to be illustrated below:

location_id	name	Local foods	Café	Fast foods
0	Chan Siew Heng Claypot Chicken Rice	1	0	0
1	Restaurant Jia Kampar	1	0	0
2	Spade's Burger Kampar	0	0	1
3	Chicken Bun Restaurant Yau Kee	1	0	0
4	Villa 21 Café	0	1	0
5	Vegan Life	0	1	0
6	East Ocean Seafood Restaurant	1	0	0
7	Restaurant Fook Kee	1	0	0
8	Kam Ling Restaurant	1	0	0
9	Chill Out Bistro	0	1	0
10	Ah Boy Noodle	0	1	0
11	Medan Selera Kampar	1	0	0
12	Dai Ga Jay	1	0	0
13	Mala Hotpot	0	1	0
14	Mosel Western & Fusion Cuisine	0	1	0

Figure 6.1.1: Restaurants with One-Hot Encoded Labels in Kampar.

user_id	location_id	rating
1	0	5
1	1	5
1	3	4
1	12	4
1	24	2
1	25	2
1	27	3
2	4	5
2	6	3
2	9	4
2	10	2
2	14	5
2	17	4

Figure 6.1.2: User Rating on Restaurant in Kampar.

Before developing the recommender system model, 2 excel files were created to act as the real-life data. Figure 6.1.1 represents some restaurants in Kampar with their respective labels, such as local foods, café or fast foods. The label of each restaurant was one-hot encoded for later use; 0 represented false, 1 represented true. Meanwhile, Figure 6.1.2 illustrates the user rating for some of the restaurants in Kampar.

Add locationId to user input

```
In [21]: userInput = [
            {'name': 'Chan Siew Heng Claypot Chicken Rice', 'rating': 5},
            {'name': 'Vegan Life', 'rating': 3},
            {'name': 'Bak Kut The Kam Heong Food', 'rating': 2},
            {'name': 'Ming Kei Chee Cheong Fun', 'rating': 5},
            {'name': 'Gok Jie Restaurant', 'rating': 4}
        ]
inputFoods = pd.DataFrame(userInput)
inputFoods
```

Out[21]:

	name	rating
0	Chan Siew Heng Claypot Chicken Rice	5
1	Vegan Life	3
2	Bak Kut The Kam Heong Food	2
3	Ming Kei Chee Cheong Fun	5
4	Gok Jie Restaurant	4

Figure 6.1.3: User Input with Ratings.

A random user input was created to represent a user who loves local foods the most for testing purpose later. After that, a dot product was performed between the rating and the label of each restaurant visited by the user.

name	Local foods	Café	Fast foods
Chan Siew Heng Claypot Chicken Rice	1x5=5	0x5=0	0x5=0
Vegan Life	0x3=0	1x3=3	0x3=0
Bak Kut The Kam Heong Food	1x2=2	0x2=0	0x2=0
Ming Kei Chee Cheong Fun	1x5=5	0x5=0	0x5=0
Gok Jie Restaurant	1x4=4	0x4=0	0x4=0
TOTAL	16	3	0

Figure 6.1.4: Dot Product of User Ratings and Labels of Each Rated Restaurant.

The sum computed was the weight of the user favour on each label. From here, the user profile is collected (more likely to eat local foods rather than café and fast foods).

location_id	name	Local food	Café	Fast foods	TOTAL
0	Chan Siew Heng Claypot Chicken Rice	1x16=16	0x3=0	0x0=0	16
1	Restaurant Jia Kampar	1x16=16	0x3=0	0x0=0	16
2	Spade's Burger Kampar	0x16=0	0x3=0	1x0=0	0
3	Chicken Bun Restaurant Yau Kee	1x16=16	0x3=0	0x0=0	16
4	Villa 21 Café	0x16=0	1x3=3	0x0=0	3
5	Vegan Life	0x16=0	1x3=3	0x0=0	3
6	East Ocean Seafood Restaurant	1x16=16	0x3=0	0x0=0	16
7	Restaurant Fook Kee	1x16=16	0x3=0	0x0=0	16
8	Kam Ling Restaurant	1x16=16	0x3=0	0x0=0	16
9	Chill Out Bistro	0x16=0	1x3=3	0x0=0	3
10	Ah Boy Noodle	0x16=0	1x3=3	0x0=0	3
11	Medan Selera Kampar	1x16=16	0x3=0	0x0=0	16
12	Dai Ga Jay	1x16=16	0x3=0	0x0=0	16
13	Mala Hotpot	0x16=0	1x3=3	0x0=0	3
14	Mosel Western & Fusion Cuisine	0x16=0	1x3=3	0x0=0	3

Figure 6.1.5: Dot Product of User Profile and Labels of Each Restaurant.

Then, the user profile is multiplied again with each label of all the restaurants in database in order to retrieve the weight of user favour towards every restaurant in Kampar.

Final Weighted User Preference

$$= \frac{\text{Weighted User Preference for each restaurant}}{\text{Sum of user profile}}$$

Figure 6.1.6: Formula to Compute Final Weighted User Preference.

Finally, the weight of user preference computed was divided by the total value of the user profile to get a normalised weighted user preference for recommendation. This content-based filtering recommender system will output those restaurants with top final weighted user preference value to the user as the result. It will be high possibility that the user would accept the result as it matched his or her interest.

```
#The final recommendation table
foods_df.loc[foods_df['location_id'].isin(recommendationTable_df.head(10).keys())]
```

	location_id	name	Local foods	Café	Fast foods
0	0	Chan Siew Heng Claypot Chicken Rice	1	0	0
3	3	Chicken Bun Restaurant Yau Kee	1	0	0
6	6	East Ocean Seafood Restaurant	1	0	0
7	7	Restaurant Fook Kee	1	0	0
8	8	Kam Ling Restaurant	1	0	0
11	11	Medan Selera Kampar	1	0	0
12	12	Dai Ga Jay	1	0	0
15	15	Bak Kut The Kam Heong Food	1	0	0
32	32	S & J Group Food Beverage Sdn Bhd	1	0	0
37	37	U2 Food Stop	1	0	0

Figure 6.1.7: Top 10 Recommended Restaurants.

6.2 Implementation Issues and Challenges

During the implementation phase of the project, several challenges were faced. The main challenge of this project is the user data collection process. As the chatbot is targeted towards personalisation purpose, it needs to understand and gather user preferences first. Yet, it is a long term and time-consuming process, since most of their actual interests can only be extracted via their daily conversations with the chatbot. Besides that, a huge amount of data is also required to train the chatbot to learn different users' preferences. The process will be tedious for this project.

Another challenge of this project was encountered during the system integration. When the IBM Watson Assistant and the recommender system model were integrated into the mobile application, there are lack of documentation online. The previous version of IBM Watson Assistant was IBM Watson Conversation. After that, IBM had integrated some of its cloud services into a unified service called Watson

Assistant. Watson Assistant does provide an easy way to integrate into web application, Facebook Messenger, or Slack. However, the proposed application in this project is mobile based. Therefore, the only way to solve this issue is via reading the IBM documentation on new Watson Assistant v2 API.

6.3 Testing

To evaluate the project's performance and ensure it met the objectives, several testing scenarios were carried out to test the functionalities of the mobile application thoroughly.

Test Scenario: User Sign In / Register Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
Register Account	Enter valid email address and password.	Receive verification email.	Receive verification email.	Pass
	Enter invalid email address, but password is valid.	Display error message.	Display error message.	Pass
	Enter existing email address, and password is valid.	Display error message.	Display error message.	Pass

	Enter valid email address, but invalid password.	Display error message.	Display error message.	Pass
Sign In	Enter correct email address and password.	Sign in successfully.	Sign in successfully.	Pass
	Enter incorrect email address.	Display error message.	Display error message.	Pass
	Enter correct email address, but incorrect password	Display error message.	Display error message.	Pass
	Enter correct email address and password without verifying account.	Display error message.	Display error message.	Pass
	Sign in via Facebook account.	Request permission, and login to homepage after granting permission.	Request permission, and login to homepage after granting permission.	Pass

	Sign in via Google account.	Choose a Google account, and login to homepage after authenticating user.	Choose a Google account, and login to homepage after authenticating user.	Pass
--	-----------------------------	---	---	------

Table 6.3.1: Test Cases for Sign In / Register Module.

Test Scenario: Chatbot Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
Communicate with Chatbot	Make greeting with chatbot.	Reply with correct message.	Reply with correct message.	Pass
	Ask chatbot for introduction.	Reply with correct message.	Reply with correct message.	Pass
	Ask chatbot for capability.	Reply with correct message.	Reply with correct message.	Pass

	Ask chatbot about Kampar.	Reply with correct message.	Reply with correct message.	Pass
Search Tourism Place	Ask chatbot to search for place based on its category.	Reply with nearest recommended place based on category.	Reply with nearest recommended place based on category.	Pass
	Ask chatbot to search for place based on its category and tag.	Reply with nearest recommended place based on category and tag.	Reply with nearest recommended place based on category and tag.	Pass
Route to Destination	Click on Route button to route to destination.	Redirect to Google Maps and route to destination.	Redirect to Google Maps and route to destination.	Pass
View Similar Places	View more similar places.	Display a list of similar places.	Display a list of similar places.	Pass
Rate Place	Rate a new place.	Rate successfully and display score.	Rate successfully and display score.	Pass

	Rate a same place.	Display error message.	Display error message.	Pass
--	--------------------	------------------------	------------------------	------

Table 6.3.2: Test Cases for Chatbot Module.

Test Scenario: Recommender System Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
Personalize Response from Chatbot	Receive a list of tourism places of same category for user personalization.	Recommend different users with different places based on their preferences.	Recommend different users with different places based on their preferences.	Pass

Table 6.3.3: Test Cases for Recommender System Module.

Test Scenario: Navigation Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
Open Navigation Bar	Access navigation bar.	Display navigation bar from side.	Display navigation bar from side.	Pass

Redirect Page	Click on different navigation tab to change page.	Redirect to selected page successfully.	Redirect to selected page successfully.	Pass
Logout	Click on logout button.	Logout successfully.	Logout successfully.	Pass

Table 6.3.4: Test Cases for Navigation Module.

Test Scenario: “My Activity” Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
View Place History	Change page to “My Activity”	Display all place histories successfully.	Display all place histories successfully.	Pass
View Place Details	Choose to view details of one place history.	Display place details successfully.	Display place details successfully.	Pass

Table 6.3.5: Test Cases for “My Activity” Module.

Test Scenario: “My Favourite” Module

Test Case	Input	Actual Output	Expected Output	Pass / Fail
Play User Interest Game	Click on Play Game button to take user interest survey.	Display game for each category successfully.	Display game for each category successfully.	Pass
	Select at most two preferred items per category.	Store result in database for analysis later.	Store result in database for analysis later.	Pass
	Select more than two preferred items per category.	Display error message.	Display error message.	Pass
View Recommended Places	Browse a list of recommended places based on user interest.	Display a list of recommended places personalized for the user correctly.	Display a list of recommended places personalized for the user correctly.	Pass

Table 6.3.6: Test Cases for “My Favourite” Module.

Chapter 7: Conclusion

7.1 Project Review

In summary, implementation of chatbot in customer service has brought lots of benefits to the business. However, most of the chatbot are still rule-based. The chatbot can only act as a middleman to answer specific questions defined in database. For other sectors, command-based chatbot is acceptable. But in tourism sector, different users might expect different recommendation on the tourist attractions provided by the chatbot, since everyone has their own perspectives and preferences. Hence, this project provides a solution for the tourists to inquire about tourism information in Kampar by communicating with the KamparBot, and receive their desired responses based on their needs. Through the communication, the chatbot will slowly learn the user profile from the user habits, and predict their actual interests for personalising recommendation to respective users. With this, it is believed that the users will have better experiences interacting with the KamparBot and more satisfy with it.

7.2 Novelty and Contribution

Currently, there are not many chatbot-based mobile applications for Malaysia tourism in the market. It is believed that the development of personalised chatbot mobile application for Kampar tourism can benefit the market in future. It is much more convenient for the community to use the chatbot for finding tourism information, as the chatbot can respond immediately to the users for frequently asked questions. Compared to traditional travel agents, the chatbot is always ready for serving the users. Unlike other chatbot, KamparBot can behave like a human who will understand the user preferences through daily conversation and make use of it to respond the users with their preferred answer with the aid of content-based filtering recommender system developed. It can also interpret the user query accurately even with local slang. Hence, the main outcome of this project will eventually provide the user with better experience in dealing with the chatbot for solving problem.

7.3 Future Work

For the future development, more tourism places can be added to the database of KamparBot so that there are more options of the interesting places can be recommended to the tourists through KamparBot. Currently, there are still not many records due to Covid-19 pandemic which restricts the movement of everyone.

Besides that, KamparBot is still a standalone mobile application. If it can be integrated into a tourism application platform such as TripAdvisor or GoMalaysia, it will be more convenient to the users as they might ask any questions to the chatbot while browsing the tourism places using those applications, instead of just installing another mobile application to use the chatbot.

In addition, collaborate with the local agents to make reservation for dining, staying, or visiting the tourist attraction places could be a good idea to accomplish in the future. If done so, the tourists can make any reservation easily using the KamparBot without the need to queue up in the counter physically.

Last but not least, the current skill added to the assistant was a dialog skill. If the budget is sufficient, a search skill powered by Watson Discovery can be added to the assistant also, in order to extend the existing functionalities. With search skill, the assistant will be able to query existing documents or data in the company, instead of just saying sorry to the customer when it cannot recognise any intent from the user query. If found any matching documents with the search skill, the assistant will be able to share the answer with the customer.

Bibliography

Kampar District Council 2020, *Kampar District Council*. Available from:

<http://www.mdkampar.gov.my/en/visitors/background-kampar>. [11 September 2020].

N. P. Patel, D. R. Parikh, D. A. Patel and R. R. Patel, "AI and Web-Based Human Like Interactive University Chatbot (UNIBOT)," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 148-150.

Nica, I., Tazl, O. and Wotawa, F. n.d., *Chatbot-based Tourist Recommendations Using Model-based Reasoning*. Available from: http://ceur-ws.org/Vol-2220/05_CONFWS18_paper_31.pdf. [11 September 2020].

R. Hirschmann, *Travel and tourism in Malaysia – Statistics & Facts*. Available from: <https://www.statista.com/topics/5741/travel-and-tourism-in-malaysia/>. [11 September 2020].

statisticstimes.com. n.d., *Top Computer Languages 2020 - StatisticsTimes.com*. Available from: <http://statisticstimes.com/tech/top-computer-languages.php#:~:text=Its%20share%20was%20over%202020>. [11 September 2020].

S. Dey and D. Shukla, "Analytical study on use of AI techniques in tourism sector for smarter customer experience management," 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Gunupur, India, 2020, pp. 1-5.

S. Tadvi, S. Rangari and A. Rohe, "HR Based Interactive Chat bot (PowerBot)," 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Gunupur, India, 2020, pp. 1-6.

BIBLIOGRAPHY

U. Arsenijevic and M. Jovic, "Artificial Intelligence Marketing: Chatbots," 2019 International Conference on Artificial Intelligence: Applications and Innovations (IC-AIAI), Belgrade, Serbia, 2019, pp. 19-193.

Weekly Reports

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	Study week no.: 3
Student Name & ID: Low Zhi Yuan 1702725	
Supervisor: Ts Dr Liew Soung Yue	
Project Title: Personalised A.I. Chatbot For Kampar Tourism Mobile Application	

1. WORK DONE

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

Study IBM Watson Assistant documentation.

2. WORK TO BE DONE

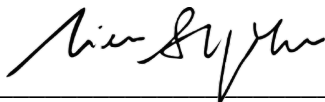
Develop new IBM Watson Assistant instance.

3. PROBLEMS ENCOUNTERED

Take more time to study and understand the whole documentation.

4. SELF EVALUATION OF THE PROGRESS

Need more time to understand IBM Watson Assistant.



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	Study week no.: 7
Student Name & ID: Low Zhi Yuan 1702725	
Supervisor: Ts Dr Liew Soung Yue	
Project Title: Personalised A.I. Chatbot For Kampar Tourism Mobile Application	

1. WORK DONE

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

Develop IBM Watson Assistant.

2. WORK TO BE DONE

Develop Recommender System(Content-Based Filtering vs Collaborative Filtering)

3. PROBLEMS ENCOUNTERED

Take some time to learn different filtering techniques for recommendation.

4. SELF EVALUATION OF THE PROGRESS

Need more time to learn more machine learning techniques for recommender system.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	Study week no.: 9
Student Name & ID: Low Zhi Yuan 1702725	
Supervisor: Ts Dr Liew Soung Yue	
Project Title: Personalised A.I. Chatbot For Kampar Tourism Mobile Application	

1. WORK DONE

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

Develop Content-Based Filtering Recommender System.

2. WORK TO BE DONE

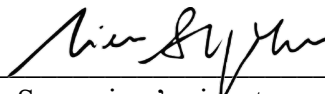
Integrate IBM Watson Assistant and Recommender System into mobile application.

3. PROBLEMS ENCOUNTERED

Not enough example to refer.

4. SELF EVALUATION OF THE PROGRESS

Need more time to understand how to integrate Watson Assistant into mobile application.



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	Study week no.: 11
Student Name & ID: Low Zhi Yuan 1702725	
Supervisor: Ts Dr Liew Soung Yue	
Project Title: Personalised A.I. Chatbot For Kampar Tourism Mobile Application	

1. WORK DONE

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

Integrate IBM Watson Assistant into mobile application.

2. WORK TO BE DONE

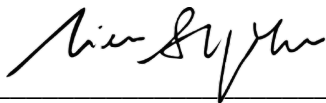
Fix bugs due to system integration.

3. PROBLEMS ENCOUNTERED

Not enough time to focus on project.

4. SELF EVALUATION OF THE PROGRESS

Need to spend more time on fixing bugs and evaluate the final outcome.



Supervisor's signature



Student's signature

Poster

Prepared By: Low Zhi Yuan | Supervised By: Ts Dr Liew Soung Yue

Personalized AI Chatbot for Kampar Tourism Mobile Application



INTRODUCTION

KamparBot is an AI Chatbot which allows the users to search for nearby interesting places, including foods, lodging, and tourist attractions. Unlike command-based chatbots, user's preferences will be extracted and analysed in this application via daily conversation and user's feedback to Kamparbot in order to achieve personalisation purposes.



OBJECTIVES

- To allow users to get tourism information in Kampar more easily and efficiently
- To allow users to receive personalised response from chatbot based on their interests

MOTIVATION

Different tourists may have different preferences, and thus it is important to understand their needs before replying to them with relevant answers. With personalisation feature implemented, different users may get different recommended places based on their personalities and interests found. Eventually, the user will be more satisfied and have a better experience with it.

DELIVERABLES

Recommendation for two different users who are finding for restaurant:





Plagiarism Check Result

Personalised A.I. Chatbot for Kampar Tourism Mobile Application

ORIGINALITY REPORT

5 %	2 %	1 %	3 %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Neelkumar P. Patel, Devangi R. Parikh, Darshan A. Patel, Ronak R. Patel. "AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)", 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), 2019 Publication	1 %
2	Submitted to Segi University College Student Paper	1 %
3	Submitted to CSU, San Jose State University Student Paper	<1 %
4	Submitted to Universiti Teknologi Petronas Student Paper	<1 %

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.: 1 of 1



FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Low Zhi Yuan
ID Number(s)	17ACB02725
Programme / Course	Bachelor of Computer Science (Honours)
Title of Final Year Project	Personalised A.I. Chatbot for Kampar Tourism Mobile Application

Similarity	Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)
Overall similarity index: <u>5</u> % Similarity by source Internet Sources: <u>2</u> % Publications: <u>1</u> % Student Papers: <u>3</u> %	Within the required range
Number of individual sources listed of more than 3% similarity: <u>0</u>	Within the required range
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 <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

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: Liew Soung Yue

Date: 16/4/2021

Signature of Co-Supervisor
Name: _____

Date: _____

FYP 2 Checklist



UNIVERSITI TUNKU ABDUL RAHMAN


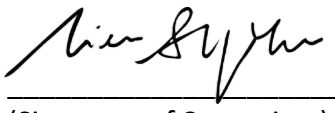
FACULTY OF INFORMATION & COMMUNICATION
TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	17ACB02725
Student Name	LOW ZHI YUAN
Supervisor Name	TS DR LIEW SOUNG YUE

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.
✓	Front Cover
✓	Signed Report Status Declaration Form
✓	Title Page
✓	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)
✓	Poster
✓	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

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

<p>I, the author, have checked and confirmed all the items listed in the table are included in my report.</p>  <p>(Signature of Student) Date: 15/4/2021</p>	<p>Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction.</p>  <p>(Signature of Supervisor) Date: 16/4/2021</p>
---	--