CHATBOT ASSISTED INQUIRY AND TICKET BOOKING SYSTEM FOR CINEMA

BY

KHIU ZHEN HUNG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology
(Kampar Campus)

JANUARY 2020

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: CHATBOT ASSISTED INQUIRY AND TICKET BOOKING SYSTEM FOR CINEMA

Academic Session: JANUARY 2020

I KHIU ZHEN HUNG

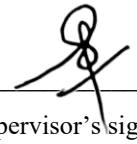
(CAPITAL LETTER)

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:
81, Jalan Harmoni 1,
Taman Harmoni Jeram,
31850 Kampar, Perak

Ts. Dr. Lim Seng Poh
Supervisor's name

Date: 22/4/2020

Date: 22/4/2020

CHATBOT ASSISTED INQUIRY AND TICKET BOOKING SYSTEM FOR CINEMA

BY

KHIU ZHEN HUNG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology
(Kampar Campus)

JANUARY 2020

DECLARATION OF ORIGINALITY

I declare that this report entitled "**CHATBOT ASSISTED INQUIRY AND TICKET BOOKING SYSTEM FOR CINEMA**" 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 : Khiu Zhen Hung

Date : 22/4/2020

ACKNOWLEDGEMENTS

I would like to express million thanks and appreciation to my supervisor, Dr. Lim Seng Poh who had supervised my progress and gave me a lot of advices and guidelines. In addition, Dr. Lim had spent plenty of valuable time to point out my mistake as well as provided suggestion to solve the problem faced.

Lastly, I would like to thank my friends for the moral support when I was facing challenges. Finally, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

Chatting Robot (Chatbot) has gradually played an important role in our daily life. It is widely used in daily life due to high response rate and ability to handle multiple users at the same time. Besides, implementation of chatbot in the entertainment industry allows the customers to clear their doubt anytime and anywhere as long as there is an internet connection. However, the chatbot-based inquiry and ticket booking system for cinema is not developed because there is no centralized database that contains the cinema and movie details, which causes the customers need to obtain the info manually. Besides, the existing ticket booking system is also not user-friendly due to the complex and non-standardized interface. Thus, this project aims to develop a chatbot-based inquiry and ticket booking system for cinema to eliminate complex steps and provide convenience to them by booking tickets using quick reply button. For this project, the main focuses are developing chatbot for inquiry related to FAQ of cinema information with English and Bahasa Malaysia translation as well as ticket booking using natural language. Therefore, waterfall model is used as the methodology for the development of the chatbot-based cinema inquiry and ticket booking system, organizing database for the details of the cinemas and movies, generating knowledge-based to clear the user doubt and booking tickets. The accuracy of the chatbot to provide relevant answers is discussed in this project. In conclusion, this project avoids the customer to go through the complex step of getting the desired answer and booking a ticket for a movie.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Overview

Chatbot is an auto-response system for interaction between users within a specific domain using natural language (Rosruen & Samanchuen, 2018). Hence, it played an essential role for the staff who worked in first line support. This is because chatbot is assisting the staff for clarifying the doubt of the customers simultaneously and within a short period based on the response provided in the knowledge-based.

However, the chatbot-based application is less emphasized in the entertainment industry which included cinema inquiry and ticket booking system. The chatbot system can be used to replace the Frequently Asked Questions (FAQ) section in order to ensure the standardization of the answers given to the customers (Muangkammuen, Intiruk & Saikaew, 2018). In other words, this can avoid the staff from giving different and inaccurate answers to the customers and hence lead to misunderstand.

Besides, the chatbot-based cinema system assists the customers for booking tickets by eliminating using the complex and many-steps system. Furthermore, implementation of the chatbot allows the information being gained by using natural language and without the need of referring to several different websites of cinema. In fact, the queue for obtaining the ticket on the spot indirectly increased the workload of the agent at the ticketing counter. Hence, the proposed system eases the customers by shortening the queue at the counter need to be developed.

Apart from that, the inquiry system is also the FAQ system and used to clear the doubts of the customers. The increase of customers forces the company to hire a huge number of agents to support the demands of customer service. This is because the customer satisfaction and the profit of the company are affected by the excellent customer service (Muangkammuen, Intiruk & Saikaew, 2018). In fact, the customer service acts as a platform for the customers in order to voice out their complaints and get the answer from the authority to solve their hesitations and doubts. Therefore, it is a value-added feature with the presence of the chatbot-based inquiry system which serves the customers with different languages and within short period.

For the ticket booking system proposed in this project is an E-ticket booking system. E-ticket is the abbreviation of electronic ticket which is generated by the computer. The purpose of developing the E-ticket booking system not only to ensure the process of ticket booking, processing and marketing more efficient and reliable, but also avoid the wastage of the paper (Kazi, et al., 2018). By implementing such a ticket

booking system, it can reduce the paper from being thrown after use once, especially for the customers who left the cinema's hall after watching a movie. Some of the customers even pollute the environment by simply throwing it but not towards the dustbin or recycle bin. Thus, the proposed system can save resources by avoiding physical printed paper tickets.

In short, a system should be developed in order to bring much more convenience and user-friendly for the customers no matter in ticket purchasing or question answering. It also brings a lot of benefits to the company as well as they did not have to spend a lot of human resources in counter handling the ticket purchase and inquiry.

1.2 Problem Statement

The main purpose of developing chatbot-based inquiry and ticketing system benefits the fast pace improvement technology by easing the customer to get the desired answer for the doubt or ticket booking for movies and reducing the workload of the cinema staff. There are mainly two types of problems faced by the customers during getting answers for inquiry and during ticket booking via online system or counter of the cinema.

Firstly, when the customers want to clarify on certain inquiries, the answer or response provided by the staff is inconsistent and various responses provided by different staffs also misleading the customers. This causes the customers to be dissatisfied with the service of that particular cinema. Although there is a standardized response in the FAQ section of the website of cinema but the list is extremely long. The situation goes worse when customers unable to find the response they want. In case the customers want to obtain the information of different cinemas and movies, the customers have no choice but to visit each and every official website of the cinema. This needed a lot of manually work to achieve the desired answer.

Besides, it is a critical problem not only for those who were busy queuing up in front of the counter to obtain tickets but also staff who worked at the counter especially popular movies are launched. Hence, the cinema company has developed a website for booking ticket purpose. However, this led to issues of complex and unstandardized as well as messy interface that full of advertisement among different cinemas cause the occurrence of confusion to the user. In other words, older customers have difficulty purchasing the ticket using that website as they are not familiar with it. Some of the

ticket booking system become meaningless because the receipt is generated instead of a valid ticket as the users still need to queue up in front of the counter. The operators of the cinema also prepare the automated ticketing machine in the cinema to reduce the workload of the staff and bring convenience to the customers especially during peak hours. However, this does not helpful to a lot of customers as they do not know the way of operating the complex machine due to many steps need to be performed.

Therefore, the problem faced by the customers and staff of the counter is explored and solved in this project by implementing chatbot.

1.3 Project Objectives

The main objectives of this project are intended:

- To develop a chatbot-based inquiry and ticketing system.
- To generate a knowledge-based for standardization of the inquiry response.
- To design a standardized user interface and organized database for the proposed system.

1.4 Project Scopes

The scopes of this project are stated as follows:

- Only focus on cinema in Malaysia – Tanjong Golden Village (TGV), Golden Screen Cinema (GSC) and Lotus Five Star (LFS).
  These cinemas are famous in Malaysia and the up-to-date information easily to be obtained from the official website.

- Only generate knowledge-based with limited inquiries related to information of cinema and movie.
  The inquiries trained in the knowledge-based are based on the FAQs section of each cinema. In this project, the similar question and answer is created once to avoid redundancy.

- Only generate and send the copy of ticket to the email of the customer.

The ticket sends to the email of the customer to act as a copy since the conversation of chatbot is transient. The email platform is used because the email was a free platform.

- Only generate Quick Response (QR) code for the ticket to avoid ticket obtaining at the counter.
  The QR code being generated in order to ease the customers by avoiding them from collecting the tickets at the counter. This can reduce the burden of staff when checking the ticket since all information is embedded in the QR code.

- Will not focus on payment of the ticket of the movie.
  The payment will not be involved in this project since the connection to the bank for payment using Application Program Interface (API) causes the testing to be difficult. However, purchased history is recorded.

- Only focus Translation on Inquiry
  The translation of Bahasa Malaysia utterance for the chatbot to classify the intent and then translate the response back to Bahasa Malaysia. However, there is no translation on ticket booking.

1.5 Impact, Significance and Contribution

The problems of inquiry in existing solutions are inconsistencies of response given by staff and a long list of FAQ sections. The problems of ticket booking are the long queue in front of the counter to get the ticket, complex and messy interface of the online system. Furthermore, some cinema online ticketing system generate receipts instead of valid tickets force the customers to line up for getting valid ticket before entrance to the hall. The staffs at the counter are overloaded with the dull and repeated task. Therefore, the simple and convenient factors need to be considered to provide customers a user-friendly system that ease them by avoiding manually searching.

The main contribution was customers can make everything to be settled just with the natural language no matter questioning inquiry or ticket booking. Apart from that, there is translation being done at inquiry where the response is returned in Bahasa

Malaysia in case the utterance is in Bahasa Malaysia. Yet another contribution is eliminating the complex steps of searching suitable nearby cinema. This is because the chatbot is able to recommend the nearby cinema by given the current location of the customers. Besides, the customers can also reduce the step of looking for suitable time and movie to watch. The chatbot provides the information of movies and respective showtimes in different cinemas nearby. This eased the customers from choosing an interesting movie without doing comparison manually. Also, the customers can avoid from being frustrated when searching inquiry in the lengthy FAQ section and cause them to overlook the inquiry. Lastly, the ticket booking system reduces the workload of the counter staff by distributing the ticket and shortening the queue of purchasing a ticket on the spot as there is E-ticket with the QR code. In other words, the E-ticket generates by this system is a valid ticket that allowed the users to enter the cinema for their favorite movies.

1.6 Report Organization

The report organization described as follows:

Chapter 2 focused on the literature review of the previous work and existing similar system. This was to generate an idea to develop the requirement of the user for the proposed system by identifying the weaknesses and adapting the strengths.

Chapter 3 discussed the methodology of the proposed system. This is the main of this chapter which covered the method and system design.

Chapter 4 was regarding the system development and discussion. This section was presented the user with system interface and functionality of the system. In addition, this chapter also included discussion of the model used and comparison of the performance.

Chapter 5 was the conclusion for the whole proposed system. This section was included the summary of the proposed system and suggestion on the future works.

CHAPTER 2 LITERATURE REVIEW

2.1 Overview

Nowadays, chatbot has evolved into part of our life to convenience us by providing the respective function. As mentioned in the previous chapter, the chatbot application is less emphasized in the cinema industry. Thus, the customers are unable to enjoy the benefits provided by the chatbot. Therefore, this chapter discusses about the related work of chatbot application, inquiry system and ticket booking system to provide brief knowledge on those keywords. In addition, the related application is compared among each other in order to identify the current problem and improvement will be suggested. For the ease of reference, a table is created with the column of application name, strengths, weaknesses and proposed system.

2.2 Review on Related Works

An interactive chatbot using Artificial Intelligence Markup Language (AIML) which is derived from Extensible Markup Language (XML) for answering the FAQ of university (Ranoliya, Raghuwanshi & Singh, 2017). AIML matched the utterance with content within pattern tag, <pattern></pattern> and replied the response based on element in template tag, <template></template>. If no category is matched, a default answer such as “I do not know the answer” is given. AIML is simple to be used due to it only consists of a few types of tag but whenever the utterance for matching increases, the AIML code needs to be modified.

Yet another application of chatbot is purposely for a medical consultant system called MedBot (Rosruen & Samanchuen, 2018). MedBot implemented Dialogflow powered by Google where training using huge numbers of utterances with labels (supervised learning). Dialogflow ease the integration process among instant messaging (IM) platforms and is able to support more than 20 languages. However, the limitation of the free version is having a limited number of requests which are unable to withstand the requests from the public who wish to get a medical consultant (Dialogflow, 2019).

Apart from that, a group of researchers has developed a ticketing chatbot service using Wit.AI powered by Facebook (Handoyo, et al., 2018) which emphasizes on the serverless feature using the webhook. The incoming message triggered the webhook and the response sent to the client side via GET and POST request. In case the incoming message contains any matched named entity such as time, location, date and so on, that

information is extracted by Named Entity Recognition (NER). Wit.AI is free to be used without limiting the number of requests and it is also able to reduce the time processing because the chat history is recorded and trained immediately. The disadvantage is being unable to handle the situation out of topic and input contains typing error.

Following that, the researchers have developed an E-ticketing system for public transport which is bus (Kazi, et al., 2018). The users need to login and select the preferable seat. After that, the payment amount is deducted automatically from the account and lastly E-ticket is generated as evidence of paying for the seat. This E-Ticketing system is able to prevent the user from missing their paper printed ticket as the E-ticket is stored in the electronic device. Besides, the users can also be able to identify the number of available seats which is able to check using the system. However, this user-friendly ticket purchasing system required few steps to be followed during ticket purchasing.

Besides, a research paper has been done regarding the intent classification (Peters, 2018). According to this paper, the first step is data generation. After that, preprocessing steps are taken to convert all words to lowercase, remove the punctuation, tokenize the texts into words. Later, the text vocabulary is generated and used to swap the tokenized text into integers. Lastly, the sequences are padded into specific length. After that, the model is trained based on deep learning architecture such as Long Short Term Memory (LSTM). The percentage of validation lost and accuracy as well as total training time is documented. Yet another chatbot system is developed using Tensorflow where the flows are almost similar to the approach proposed by (Peters, 2018). The only difference is building vocabulary dictionaries where Bag of Words is used to represent the present of word (1) and absent of word (0).

Last but not least, a model that applied Convolutional Neural Network (CNN) for text classification (Cha & Lee, 2018). The researcher had proven that CNN is not only useful in image classification but also text classification. The convolutional layer is able to produce feature maps which hold the embedding information of the keyword. The feature maps that hold the weights of the keywords undergo a fully connected layer for the outputting the useful features for the softmax classifier to make prediction. In other words, the fully connected layer eliminated those features that are not active before reaching the softmax classifier to avoid unnecessary information confusing the predictions.

According to a paper researched on LSTM and simple Recurrent Neural Network (RNN) on the generative model of Bahasa Indonesia Conversation, the performance of LSTM is better than simple RNN (Prabowo, et al., 2018). The LSTM performed better than RNN in terms of response speed and providing the expected response. This is because the LSTM is designed to remember the order of the words while RNN assumes the word is independent.

Lastly, the Gated Recurrent Unit (GRU) Sequence to Sequence (Seq2Seq) is research on the field of man and machine conversation system (Huang, et al., 2020). This research focuses on studying the generation model of dialog by implementing GRU Seq2Seq model since GRU is able to improve LSTM by reducing the number of gates and improving RNN in terms of vanishing gradient descent issue. The Seq2Seq contains three parts which include Encoder, Decoder and linkage page in between both of them. The GRU is applied in both Encoder and Decoder for the text generation. The Encoder is used to understand the meaning of the word and Decoder is used to predict the response to be returned.

2.3 Review on Similar Systems

This section discusses about the similar application regarding to their functionalities, strength and weakness. There are few systems is reviewed on this section.

2.3.1 TNB Chatbot – Lisa

Figure 2-1 shows the interface of chatbot of Tenaga Nasional Berhad (TNB) which is Lisa. Lisa is a virtual assistant applying to Tenaga Nasional Berhad to help the customers to solve their inquiries (TENAGA NASIONAL BERHAD, 2019).

First of all, the users need to select the language that they wish to be used throughout the conversation by clicking on one of the quick reply buttons as shown in Figure 2-1 with the text of “English” and “Bahasa Malaysia”. After the language selection, users can click the link on the menu of the “Current Hot Topics & Most Popular Question” as shown in Figure 2-2 to get the desired answer. However, users need to input the question manually when the question is not found in the menu.

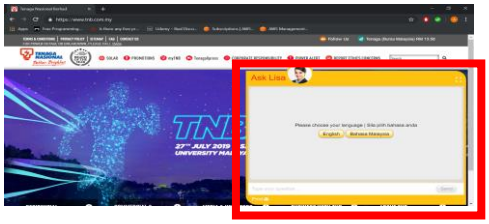


Figure 2-1 TNB Chatbot - Lisa Interface

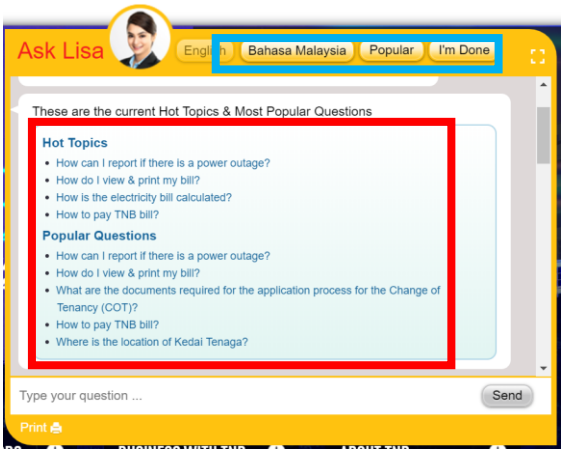


Figure 2-2 Language Switching and FAQs Topic of TNB Chatbot – Lisa

Special Features

1. The language can be switched just with one click, which is shown in Figure 2-2 with blue rectangle.
2. The hot topics and popular questions are suggested to ease the users in selection without typing.
3. This virtual assistant presents sentiment analysis because when users type in sentences with negative words such as “I don’t like you”, it will apologize to the users and request the users to provide feedback for improvement.
4. There is a “I’m done” button for the users to feedback about the satisfactory level of using Lisa.
5. There is also a “Print” button which enables users to print the conversation record.

Strengths

1. Easy to use due to the suggestion menu and tidy interface.
2. It supports two languages which are English and Bahasa Malaysia.
3. Switching language just with one click.
4. For the question asked by Lisa, quick reply button is provided, so that users do not required manually typing the answer for reply.

5. It provides print feature for the users who need the conversation record for future used.

Weaknesses

1. It is less possible to answer the question if the way of asking is changed.
2. When the users want to select topics from the menu, the users have to scroll up the conversation for selection.

2.3.2 Microtel Technology Facebook Page Chatbot

Figure 2-3 shows the chatbot of Microtel Technology which made use of Dialogflow integration (Microtel Technology, 2019).

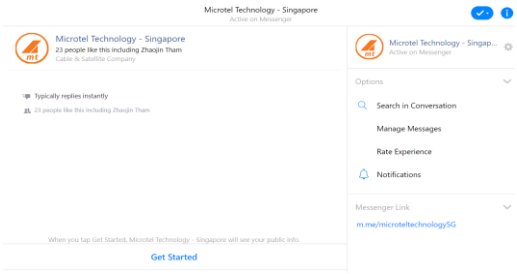


Figure 2-3 Facebook Messenger Chatbot of Microtel Technology - Singapore

When the users click the "Get Started" button, the webhook is triggered as mentioned in the related work section so it will reply the message as shown in Figure 2-4.

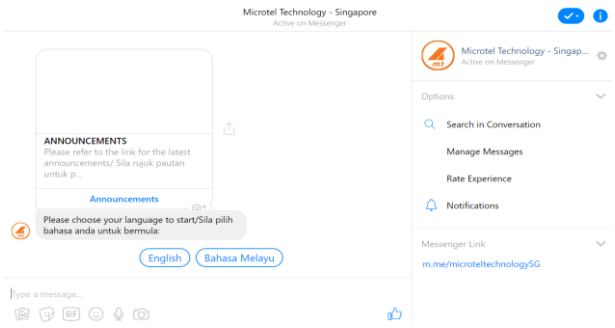


Figure 2-4 Language Selection Interface of Facebook Messenger Chatbot of Microtel Technology - Singapore

After selection of language, the chatbot introduces itself and then allow the user to choose for the interested option as well as allow the user to enter the question manually. After giving the response, the chatbot requests for the feedback from the user following with the "Back to Menu" and "End Chat" quick reply button. When the former is clicked, users can continue asking for other questions while the latter

terminates the conversation with a sentence of thanks and appreciation. The chatbot also contains the sentiment analysis feature which is when the negative emotion words are the input of the users, it prompts the users whether they want to talk with a live agent or end the chat. If the user input is out of scope, the chatbot apologizes to the users for being unable to give help.

Special Features

1. This chatbot does not require users to register but login with a Facebook account then can use it.
2. This chatbot supports sentiment analysis and natural language processing because users are able to use their own words to ask for the question.
3. This chatbot contains a conversation flow to guide the users when there are several steps to perform certain action.
4. This chatbot is able to talk to a live agent which is able to solve the problem of users that chatbot is unable to solve.

Strengths

1. Integration with Facebook eases the usage as users already familiar with the interface of Facebook.
2. There is a flow of conversation that guides users to solve their problem.
3. It supports two languages which are English and Bahasa Malaysia.
4. Allow users to chat with a live agent when there is input with negative emotions.

Weaknesses

1. There is only a single login method which is Facebook, users might not want their personal information known by the other competitors.
2. Unable to switch the language just with one click.

2.3.3 Watson Assistant Demo

Figure 2-5 displayed the Watson Assistance for demonstrating purpose which focuses on the banking and finance industry (Watson, 2019).

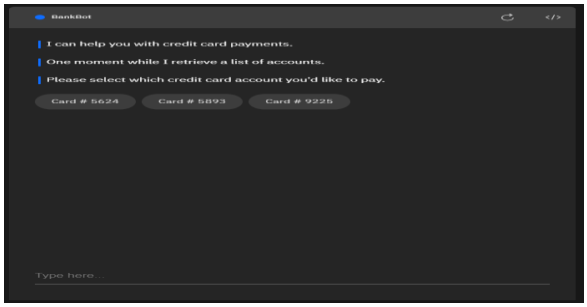


Figure 2-5 Watson Assistant Interface

This demonstrates that chatbot contains mainly three services which are making a payment, booking an appointment and recommending a credit card.

For the making payment service, the chatbot will provide the quick reply button for the card selection and then prompt the date for the payment to be done. Before the payment is done, the chatbot will send a confirmation message to make sure the information is correct and check whether balance is sufficient. Lastly, a receipt of payment will be displayed with the information that prompted previously.

As to book an appointment, it is simple to be used as the users only need to provide the information to the chatbot. For example, “I want to meet Peter tomorrow at 8am”, then the chatbot will extract the information from the text. If there is a lack of required information such as location, the chatbot asks for the location. After that, the chatbot converts the word “tomorrow” into the date and then confirms the information with the user. Once the users confirm the correctness of the appointment, the chatbot will display the remainder for the users.

The credit card recommending service is almost similar to the other services mentioned. However, the uniqueness of this service is that it contains the flow of conversation that will react differently based on the selection of the user.

Special Features

1. The chatbot will make sure the information is correct with the confirmation.
2. The chatbot is able to extract the information in the user input.
3. The chatbot is able to generate receipt, reminder and chat history of credit card recommendation.
4. The chatbot contains the flow of conversation which is able to react according to the response of users.

Strengths

1. Confirm with the users before taking the action.

2. Able to extract the information from the input of user
3. Able to convert unstructured data (user input) into structured data (reminder).
4. Support the flow of conversation that guides users step by step.

Weaknesses

1. Does not support other languages besides English.

2.3.4 cinema.com.my

This is the interface of the cinema.com.my as shown in Figure 2-6 (CinemaOnline, 2019).

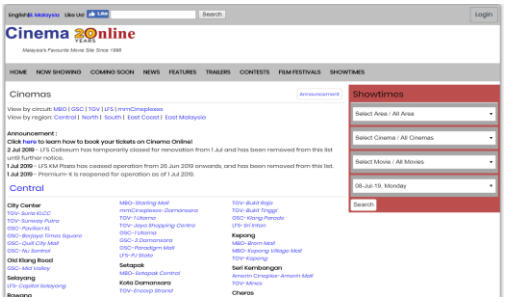


Figure 2-6 Cinema Selection Interface of cinema.com.my

The users need to select the nearby cinema form the list to get the movie names and showtimes. After user click on the movie name, the details of movie is shown and comparison of cinema feature as well as search engine is there which is shown in Figure 2-7.

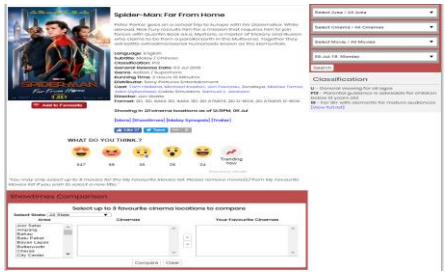


Figure 2-7 Interface for Comparison of Movie Showtimes

There is also a ticket booking service in this website, but it does not support all of the cinemas.

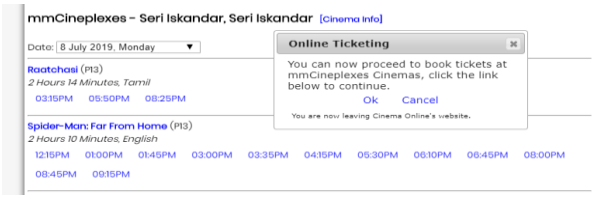


Figure 2-8 Interface for Ticket Booking Service

For the cinema that supported the ticket booking service, the time of the movie is blue in color whereas black in color if not support that particular function which is shown in Figure 2-8. When the users click on the time, the dialog box will pop up which contains “OK” button (shown in Figure 2-8) that will redirect the users to the official page of the cinema for seat selection and payment.

Special Features

1. Most of the cinema in Malaysia can be found on this website.
2. Presence of features of searching the movie using search engines.
3. Able to redirect the users to the cinema official webpage for ticket booking.

Strengths

1. Contain most of the cinema.
2. Able to search the desired movies and cinemas using search engines.
3. Able to filter the movies, cinemas, showtimes and area.
4. The review of the movies can be seen in terms of facial expression.

Weaknesses

1. The interface is messy and not user-friendly.
2. Many steps need to be taken for reviewing the movies details.
3. Not all of the cinemas support ticket booking service.

2.3.5 FAQs Section of TGV Cinema

This is the interface of the FAQs section of TGV cinema as shown in Figure 2-9 (TGV Cinemas Sdn Bhd., 2019).

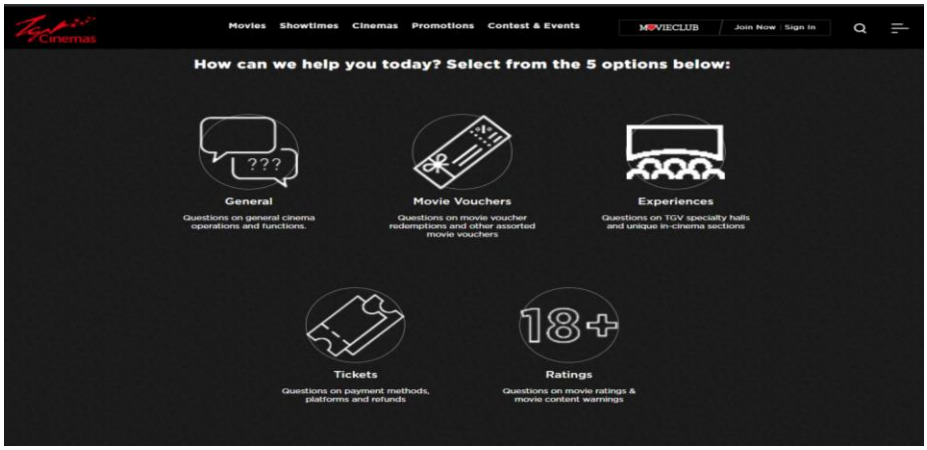


Figure 2-9 Interface of FAQs Section of TGV Cinema

According to Figure 2-9, the question and answer are categorized into General, Movie Vouchers, Experiences, Tickets and Ratings. The users can gain explanation or solution by selecting the category. After that, the users need to go through the topic as well as questions displayed.

If the users find the question that they wish to ask, the users need to click on the "+" button in order to get the response or solution. In case the users are unable to find any relevant solution to clear their doubts, they can contact the customer service hotline given in the bottom of the FAQs section. However, for those who wish to complain about the services or products, there is an email given in the FAQs section.

Special Features

1. The questions and respective answers are categorized to ease the users to search based on their inquiries.
2. The answers of the questions are hidden by default to make the interface neater.
3. The questions are grouped together according to the topic.

Strengths

1. The questions and respective answers are categorized.
2. The answers of the questions are hidden by default.
3. The questions are grouped by the topic.
4. The answer or solution provided is short and clear.

Weaknesses

1. Users need to read all the questions to find the solution.
2. The solution provided without the visual aids that make the users more understand.

2.3.6 Quick Book Section of TGV Cinema

This is the quick book section of TGV Cinema as shown in Figure 2-10 (TGV Cinema Sdn. Bhd., 2019).

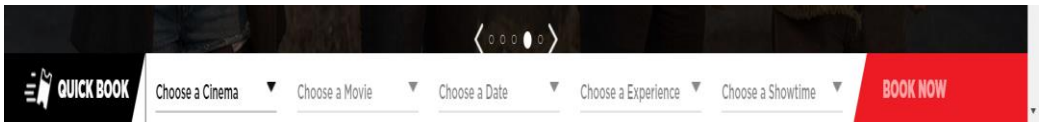


Figure 2-10 Quick Book Section of TGV Cinema

This particular section is easier for the customers for ticket booking. According to Figure 2-10, customers can select the Cinema based on the respective location. After that, customers can select their favorite movies and the dates as well as showtime. As to the Experience drop-down list, it confuses the customers since it does not provide any help text or so-called hint.

When the customers select the adult constraint movie, it displays the notification as shown in Figure 2-11. This can make sure the underage customers are alert regarding the content of the movie selected.

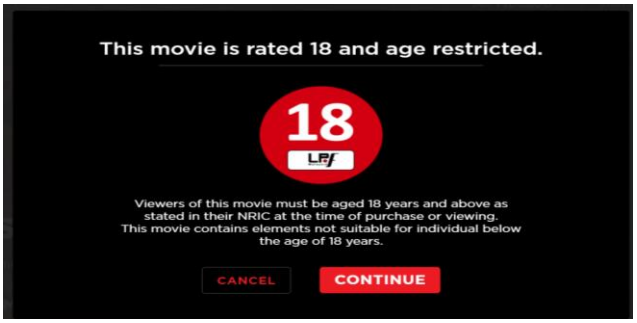


Figure 2-11 Adult Constraint Notification

After that, customers can login to the system using their credential details or continue as a guest. If the customers choose to continue as guest, they need to provide the name, email and phone number. Following that, the customers are free to choose their preference seats and then select the combo set of drinks and snacks. Lastly, the customers need to provide the payment method for completion of ticket booking.

Special Features

1. Quick booking of tickets with a simple interface.
2. Alert the customers regarding the content of the movie is not for underage.
3. Easier for customers since they need not login to the system.

Strengths

1. Provide the customers with the option of continuing without login.
2. Allow the customers for ticket booking with just a few clicks.

3. Alert generated for those underage customers.

Weaknesses

1. Delay when selecting movie, date and time.
2. Present of confusing selection which is Experience.

2.3.7 Easy Booking Section for LFS Cinema

Figure 2-12 shows the interface of the Easy Booking Section of LFS Cinema (Lotus Fivestar Cinemas (M) Sdn. Bhd. , 2019).

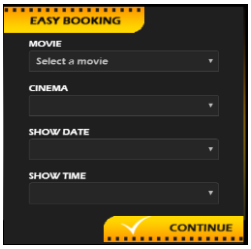


Figure 2-12 Easy Booking Section of LFS Cinema

This interface is for the ease of customers for ticket booking. Customers only need to select the movie, cinema, date and time. During the selection of seats, customers can select whether they are child or adult. Lastly, customers can check out the system by providing the payment account and their information such as name, email and phone number for finishing the ticket booking step.

Special Features

1. Does not require user login during ticket booking.
2. Able to choose whether child or adult since prices of both categories are different.

Strengths

1. Simple interface and easy to use.
2. The step for ticket booking is less and clear.
3. Able to select whether adult or child.

Weaknesses

1. Does not alert users with the age constraint content.

2.3.8 Zendesk Chatbot Demo

This is the interface of Zendesk Chatbot demonstration on Zendesk official website as shown in Figure 2-13 (Zendesk, 2019).

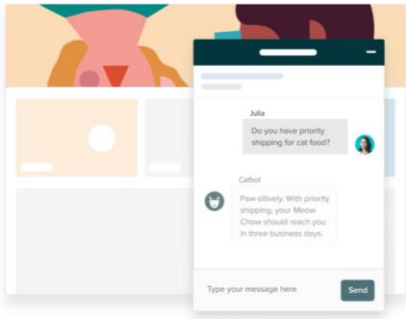


Figure 2-13 Zendesk Conversation Demo

This chatbot is an auto reply system for specific domains. When the customers ask a question, the chatbot figures out the answers and reply accordingly. When the customers wish to place an order, the Zendesk Chatbot redirects the customers to the live agent as shown in Figure 2-14.

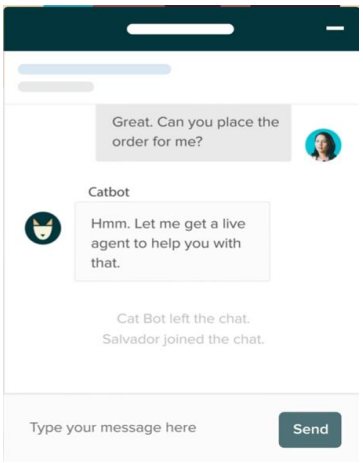


Figure 2-14 Redirect Customers to Live Agent

Special Features

1. Able to auto reply when customers ask for specific questions.
2. Able to redirect the customers to live agents for specific intent.

Strengths

1. Reply to the inquiry of the customer automatically.
2. Able to detect intent of user and redirect to live agent.

Weaknesses

1. Unable to integrate with social chat
2. Chatbot system must integrate with Customer Relationship Management System.

2.4 Comparison of Applications with the Proposed System

Table 2-1 had shown the comparison of features present in the application. The numbering below are represented by section numbering. For example, 2.3.1 is referred to TNB Chatbot.

Application / Feature	Chatbot				Inquiry System	Ticket Booking System			The Proposed System
	2.3.1	2.3.2	2.3.3	2.3.8	2.3.5	2.3.4	2.3.6	2.3.7	
Sentiment Analysis for user input	√	√							√
Feedback after conversation end	√	√							√
Selection in Quick Reply Button	√	√	√						√
Guide users using conversation flow	√	√	√						√
Redirect conversation to live agent		√		√					√
Automatic retrieve information from user input			√						√
Able to search or filter the information						√			√
Provide short and clear answer		√	√	√	√	√			√
Alert for age constraint content							√		√
Does not required Login for further process							√	√	

Simple to use for ticket booking							√	√	√
Language switching	√	√							√

Table 2-1 Features Comparison of Applications

Therefore, the features that presented in the proposed system are sentiment analysis for user input, provide feedback form after conversation end, allow the user to print the conversation record, provide quick reply button for the selection, guide user with the conversation flow, automatic retrieve information from user input, able to search and filter the information and provide short and clear answer. In addition, the feature of redirecting the users to the live agent when chatbot is unable to answer the inquiry is also involved in this project. This is because that particular feature is essential as when the system is unable to solve the problems of the user and the problems cannot be left as unsolved so it should send notification to the live agent for them to communicate. Although this is not an automated system but not all of the problems solve by the agent so it can reduce a lot of the workloads. Last but not least, the proposed system also alerts the user regarding the movie of age constraint and be simple to use for ticket booking. The proposed system also included language switching but it is different to Lisa and Microtel Technology Facebook since the switching is done without manual click. However, Login is required for this system due to not involve payment after booking tickets.

2.5 Summary

This chapter has explored the background of the chatbot, inquiry system and ticket booking system. Several previous works have been reviewed in order to understand and improve the existing systems through the proposed system. Therefore, the weaknesses mentioned in the previous section will be taken into consideration and improvement will be done in this project to ease the users in terms of convenience and time saving.

CHAPTER 3 RESEARCH METHODOLOGY

3.1 Overview

This chapter discusses the methodology used in developing the proposed system. The methodology chosen is further elaborated regarding the way to implement this project. The advantages and disadvantages are discussed and risks are taken into consideration in order to avoid it. Each phase of the methodology is further explained with the works to be done. In addition, the tools, hardware and software are discussed. Furthermore, the issues and the challenges during the implementation are also highlighted. Lastly, the timeline is generated in order to ensure the project was able to complete within the time allocated.

3.2 Methodology

The methodology that is being chosen for the project of chatbot assisted inquiry and ticket booking for cinema is waterfall model. Figure 3-1 shows the waterfall model as referred from (Sommerville, 2011) which is one of the models in Software Development Life Cycle (SDLC).

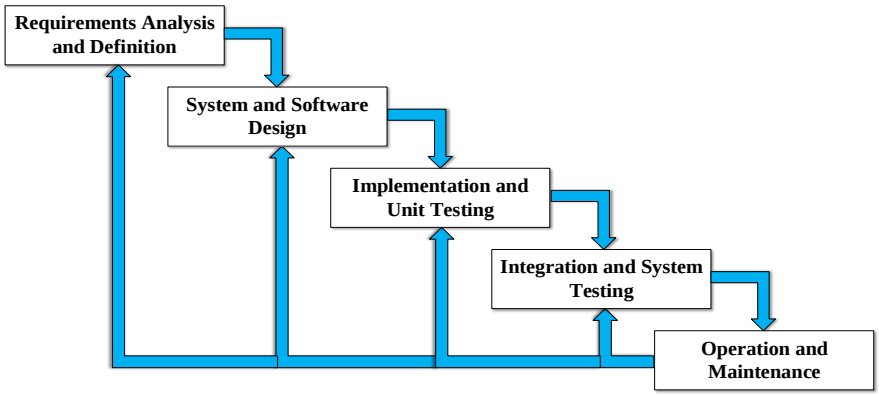


Figure 3-1 Waterfall Model

The waterfall model is an approach for project management that flows sequentially. In addition, the project is done based on the allocated date, requirement set as well as expected outcomes (Gallagher, Dunleavy & Reeves, 2019). The model is proceeded phase by phase after the completion of a phase and the backtrack of the phases is not allowed (Mkrtchyan, 2017). In other words, the development team needs

to understand and decide an upfront plan to avoid repetition of similar steps. Therefore, the requirement needs to be analyzed before development starts.

Waterfall model is selected due to the several advantages. The advantage is project scope determined during the early stage of the project. This eases the management on the cost and time because the minimum changes are done to the project scope during the development phase. Besides, the structured approach of the waterfall model allows the identification of the risk and figures out a plan to reduce the unpredictable risk causing the project failure. Also, system was documented in details and it is essential especially when the project needed to be referenced and improved. Lastly, the requirement and expected outcome is understood before the project starts to be developed.

However, the model also contains disadvantages and will be tolerated. In this project, the disadvantage of unable to perform well in a large-scale project is tolerated as the proposed system is small and requirement also being fixed during the analysis phase. The ambiguous documentations of previous steps are also tolerated due to the proposed system being done by a single person. Besides, since the requirements are determined based on the improvement of the weaknesses of previous work, so the disadvantage of unable to cope with frequent customers involvement for requirement gathering can also be avoided.

The following section discusses and explains in details for each and every phase in the waterfall model.

3.2.1 Requirement Analysis and Definition

In this phase, the problem statement and motivation are identified since the solution is not being figured out when there is no problem faced by that particular domain. In addition, the target users are also being identified in order to develop a system that is user-friendly to the target user. Then, the background information is researched for proposing better solutions and solving the current problems. The project objective and project scope also being determined.

After the initial work mentioned above is accomplished, the user requirements are gathered by reviewing the existing or similar application. The special features, strength and weakness will be identified. After that, the requirements of the users are obtained by gathering the special features and suggestions for enhancing the

weaknesses based on Chapter 2. The proposed system is built based on the strength of the applications.

In addition, justification of the features is based on the problem statement, project objectives and project scope to eliminate meaningless features. To ease the usage and save time, the Quick Reply button is replaced the manually typing for the fixed response such as “Yes” or “No”. In fact, the high quality chatbot not only should guide the conversation flow by Quick Reply button to avoid invalid input but also need to be able to retrieve information from user input automatically. A better experience is provided when the natural language of question answering with a short and clear answer is used to replace the form filling process.

Lastly, sentiment analysis is an element to avoid the chatbot keep on giving the wrong answer by routing the user to the live agent after several times of detecting negative sentiment. Users also have the opportunity to provide feedback according to the response given by the chatbot to increase the accuracy. In this proposed system, the dataset of sentiment analysis is obtained from (Kaggle, 2019) which is related to Twitter due to the similarity of condition. In addition, the data created by referring to the FAQs sections shown in the official website of each cinema. For the translation dataset, it is the translation of questions and responses in the FAQs sections.

3.2.2 System and Software Design

In this phase, the unstructured user requirements are converted into the system architecture to avoid misunderstands. The hardware and software are decided by considering the factors such as cost effectiveness, technical support provided by vendor and ease of use.

In this project, the system overview is represented with the Use Case diagram which included functionality for different actors like Admin (Developer), Cinema Admin, Agent and Customer. Each of the actors is given specific permission while the Admin is having the full right on the whole system for the debug purpose. After listing the flow of web pages in the proposed system, the activity diagrams are generated to show the interaction between actors and system.

Entity Relationship Diagram (ERD) diagram is designed to show the entity name and respective attributes as well as relationship between entities. Class diagram is generated based on the name and attributes of the entities in ERD. The functions of

the class diagram include setter and getter as well as function of interacting with the Stored Procedure of database. Lastly, the activity diagram also being used for representing the training and prediction of the model of deep learning.

As to the software that needed for this project are Visual Studio 2019, Internet Information Services Manager (IIS) and Microsoft SQL Server Management System 18 (SSMS). For deep learning, Spyder (Anaconda) and Git Bash are used. The Visual Studio 2019 and Spyder (Anaconda) are the software needed for coding the website of the proposed system and coding for deep learning natural language processing respectively. After that, IIS is used for hosting the application website in own computer while Git Bash is used for the purpose of hosting the source code of Python to provide services. However, NGROK is used to allow the port to be accessed by the outsider using a public address. The SSMS is used to store the data of the proposed system such as conversation record, responses and so forth. The tools that are used included ASP.Net Framework for developing the website in C# whereas Pytorch is used for machine learning model.

3.2.3 Implementation and Unit Testing

In this phase, the software design from the previous step is implemented using tools and software mentioned by programmed the module on the user interface design and functionality from previous phase. For example, the chatbot is divided into several modules such as getting user input, sentiment analysis, output message and intent classification. This particular module is undergone unit testing before integration because the bugs difficult to be trackbacked and fixed due to the size of the integrated system is extremely huge.

Unit Testing also known as White Box Testing because the internal structure and design as well as coding of the function are being tested (Guru99, 2019). This mainly focused on verifying whether the system is able to process the predefined input and produce expected output. For instance, the message retrieved by the system is compared with the original to determine whether the system failed due to words containing space.

White Box testing provides more understanding on the flow of the source code as every intermediate result displayed so it can apply automation testing. Applying the technique to this project, the test cases generated according to the function and the

intermediate output is compared with the expected output. However, the print statement for displaying the intermediate result is not used but replaced with the breakpoint function of Visual Studio 2019. Once the intermediate output is different from the expected output, the testing is stopped and source code is being fixed. After fixing the bugs for that particular part, the testing is running again to avoid changing of source code which affected the correctness of the result in previous testing.

3.2.4 Integration and System Testing

During this phase, the actions taken included integration of unit modules built from the previous stage and system testing to ensure the system work as expected and achieve the requirements that had been declared at the previous stage (SOFTWARETESTINGHELP, 2019). In addition, the database and user interface are also tested to make sure all of the modules after integration are error free.

Bottom-Up integration testing approach is selected and it starts from the innermost unit and then gradually moves up to the system. The advantage of Bottom-Up approach is easily to detect the flaws of the system during integration as major flaws come when integrating the innermost unit due to logical error and bug. Hence, Bottom-Up integration testing approach is selected and applied to this project.

By using this approach, all module of the proposed system are classified into simple module and complex module. For example, the module of getting user input (utterance) is classified as simple module and less focused much during the integration testing. However, the intent classification module is classified as the complex module as there are a lot of test cases used to assess it for detecting the accuracy. After the classification, the modules are then integrated from simple module to complex module to ensure the defects and bugs as well as errors can be resolved before the final testing which is functional testing.

In fact, the functional testing also known as black box testing in which complete functionalities of system are tested without knowing the internal structure of the system. This testing usually is tested for the functional and non-functional features of the system but mainly focused on functional features (Software Testing Fundamentals, 2019). The black box testing included testing for the component of the user interface such as textbox and button, testing for accuracy of the classification of sentiment analysis, intent and intent of inquiry and testing for the correctness of the conversation flow.

3.2.5 Operation and Maintenance

In this phase, the system is deployed for the practical used and maintenance of the system is carried out. However, since this project does not involve the customer so User Acceptance Testing (UAT) is done after with the system hosting on the Internet Information Services Manager (IIS). If there was error found in the hosted system, the system is fixed and deployed again to the IIS. In case, the hosted system is able to run smoothly without error, the public access is allowed using NGROK and documentation being done for future use. As to the maintenance, the errors of the system that are not found during UAT is fixed and enhancement is carried out when the new requirements are found until the deadline of the proposed project.

3.3 System Design

The following section displayed the overview of the proposed system with the Unified Modeling Language (UML) diagrams.

3.3.1 Use Case Diagram

Figure 3-2 shows the use case diagram contains 4 actors which were Admin, Customer, Cinema Admin and Agent. Each of them has their respective role and functionality. Admin has the full right to the system except for Registration. Some of the functionalities are not related to Admin but Admin needs the right to detect the bug in specific web pages. Cinema Admin and Agent contain similar roles except for Manage Agent and View Chat Record are granted to Cinema Admin. Customer is only able to access certain websites due to the access control done by Admin. In addition, all of the actors can search and filter for a certain webpage by using the search bar and clicking on the header of each column.

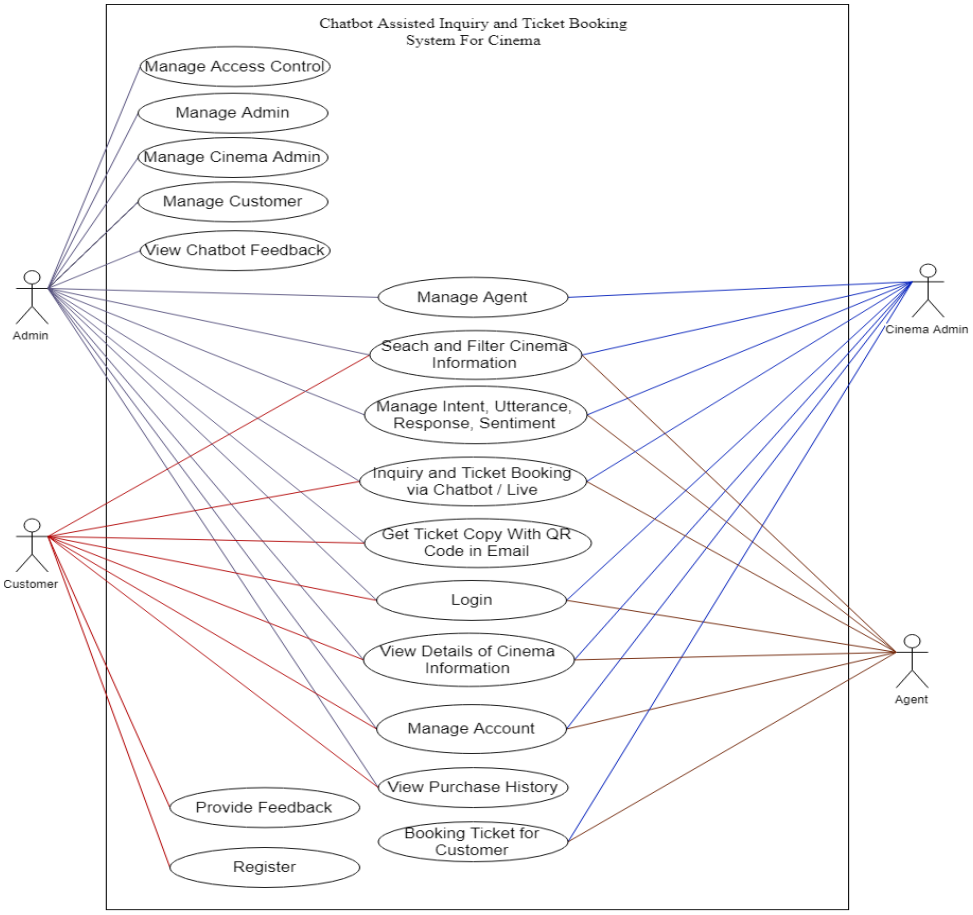


Figure 3-2 Use Case Diagram of Proposed System

3.3.2 Overview of Webpage

Figure 3-3 shows the overview of the web pages contain in the proposed system. Master Page (Index.master and Master.master) is inherited by others website in order to reduce the effort of coding for similar content such as Menu. The details of each webpage are discussed in Chapter 4.

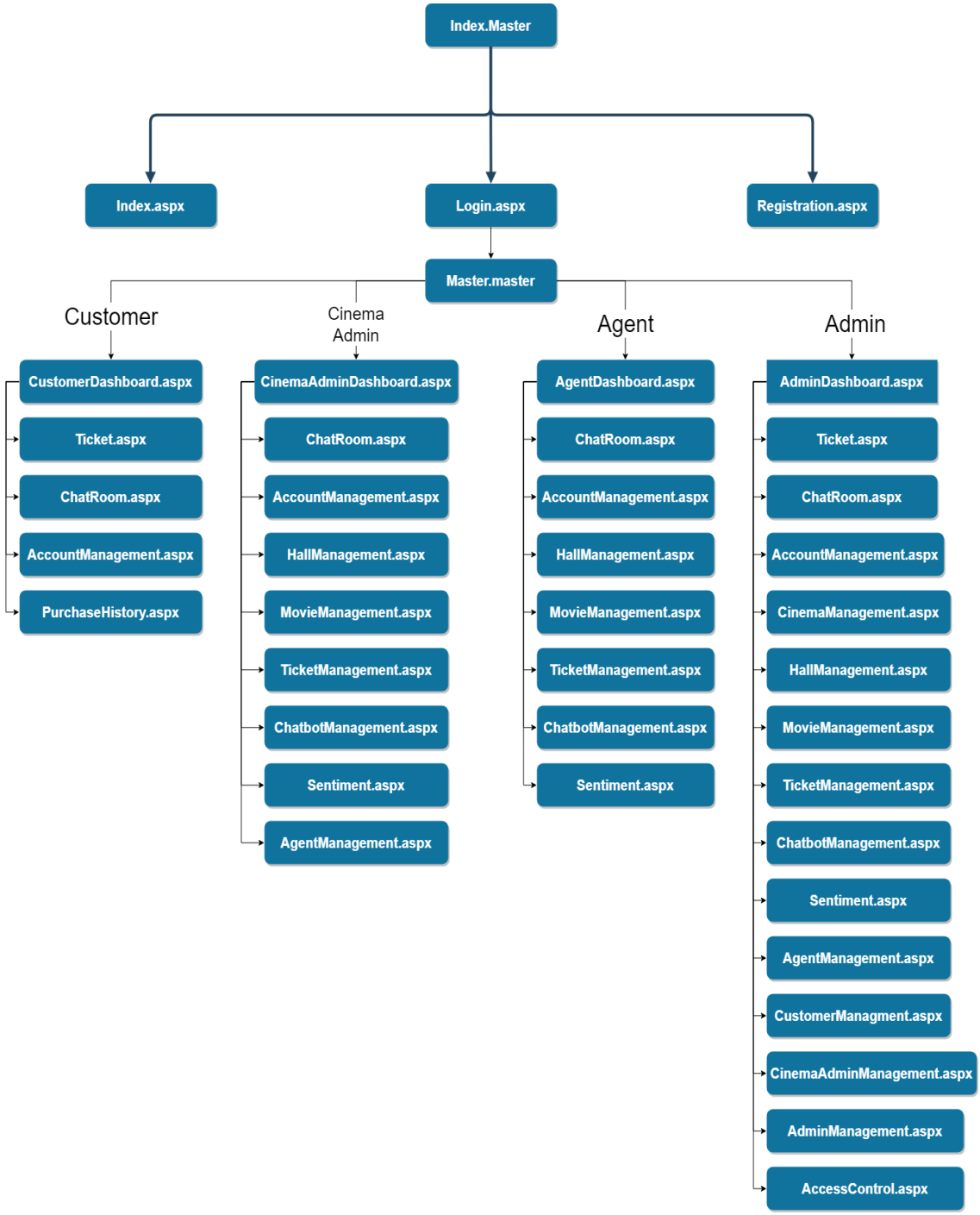


Figure 3-3 Overview of Webpage

3.3.3 Activity Diagram for Registration

Figure 3-4 shows the Activity Diagram for Registration where it is only applicable for Customers. After the form is filled, validation is taking place to avoid left blank and incorrect format. When there are invalid fields, Customers are required

to fill it again. Then the password is hashed before stored into the database. After the registration success, the Customers is redirected to the Login Page.

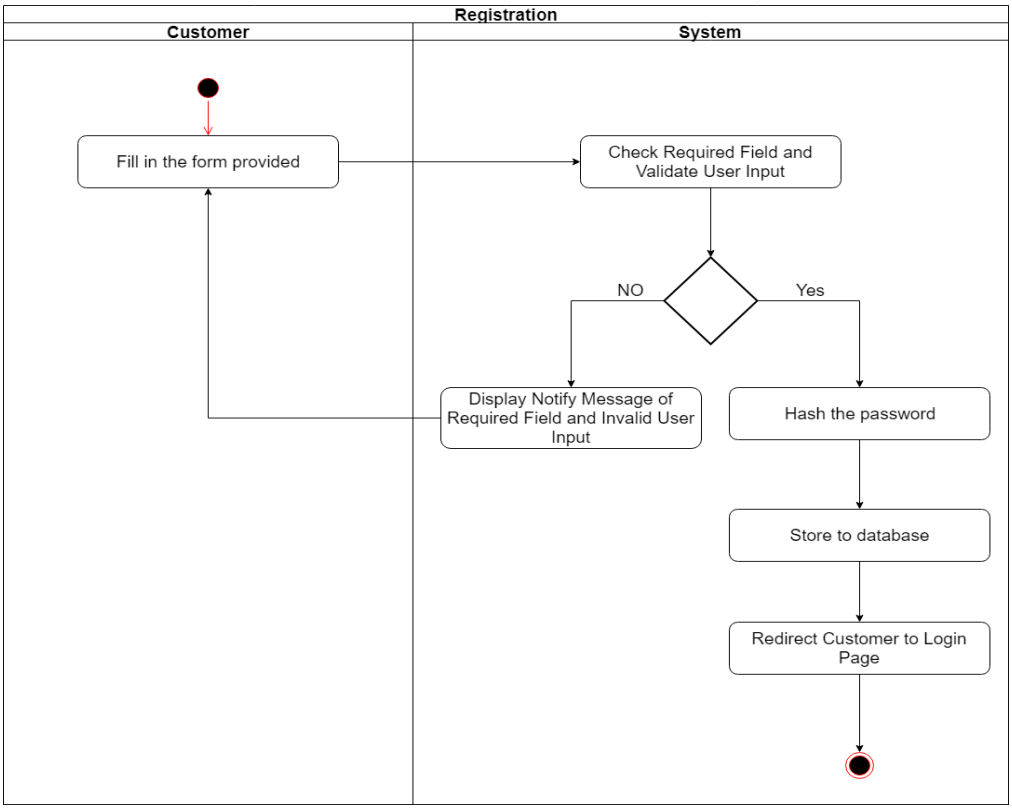


Figure 3-4 Activity Diagram for Registration

3.3.4 Activity Diagram Login

Figure 3-5 shows the Activity Diagram for Login. Users need to provide the username and password. Soon, the system hashes the password received from users and compares the password retrieved from the database based on the username given. In case, the login credentials do not match or do not exist, user are required to fill in again. When the login credentials are valid, users are redirected to respective Dashboards based on their Roles.

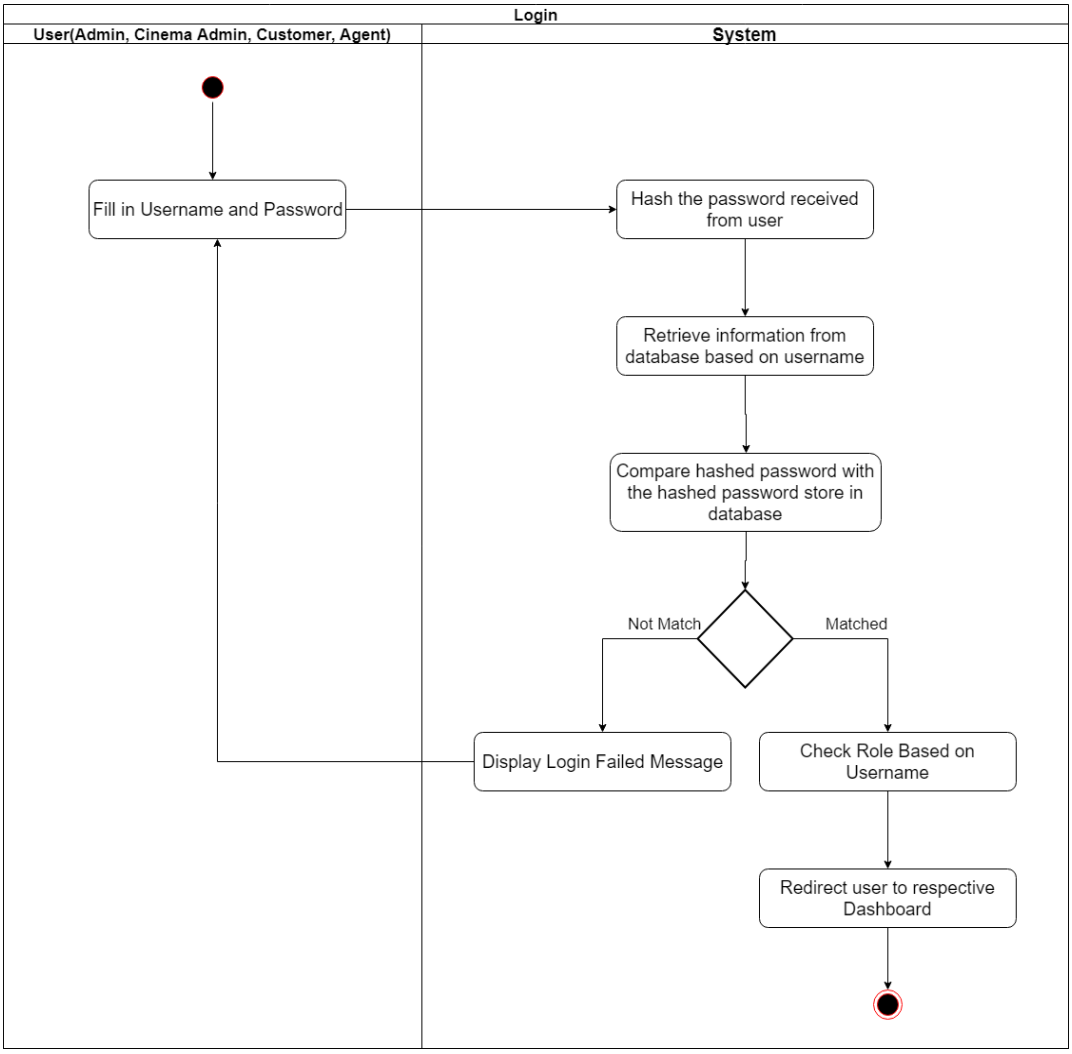


Figure 3-5 Activity Diagram for Login

3.3.5 Activity Diagram for Conversation

Figure 3-6 shows the Activity Diagram for Conversation. First, users need to select either “Ticketing Booking” or “Inquiry”. For the “Ticketing Booking”, users need to provide the ticket details and then confirm the details. After that, the system sends the ticket via email. As to the “Inquiry”, the system detects the language of utterance and classifies as well as retrieves the response from the database. Lastly, the system displays the response according to the language of input. In case, the system accumulates 3 times unable to get the meaning of user, the user is routed and handled by agent.

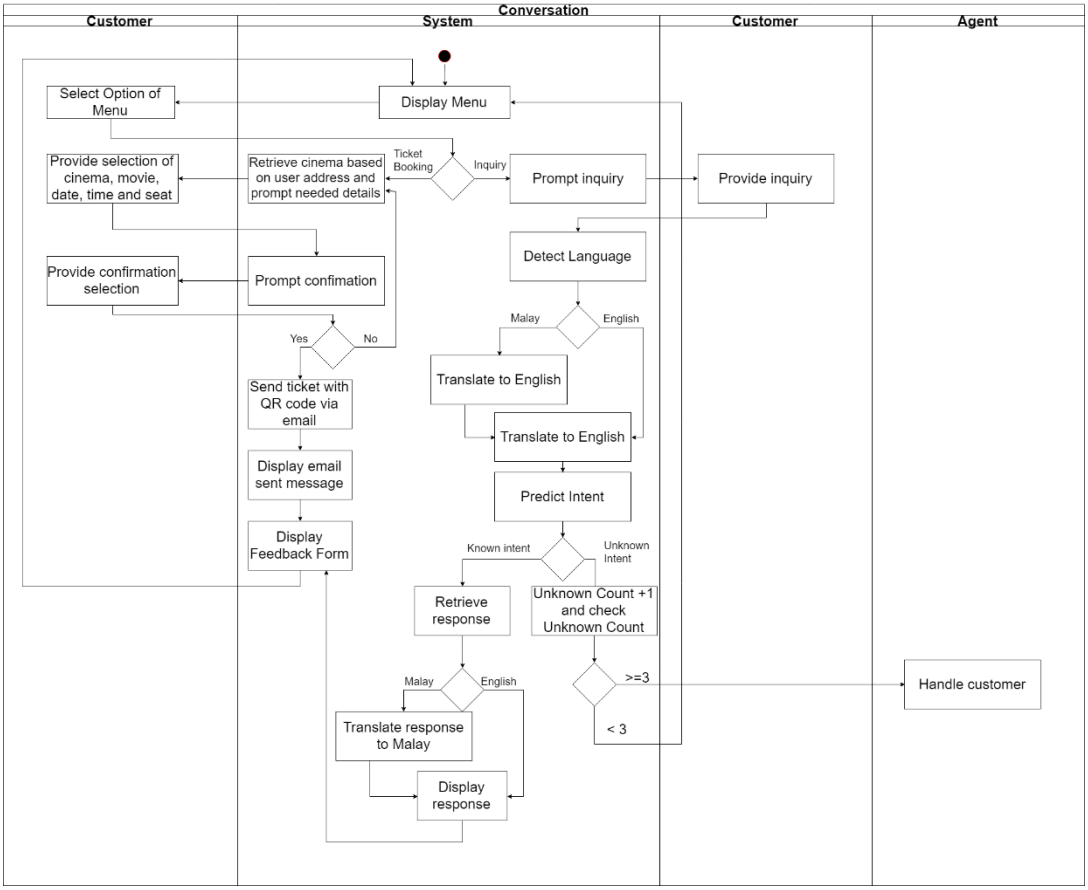


Figure 3-6 Activity Diagram for Conversation

3.3.6 Activity Diagram for Sentiment and Intent Training

Figure 3-7 shows the Activity Diagram for Sentiment and Inquiry Intent Training where the model is separate but the flow of development is the same. At first, the dataset is loaded from database and then preprocessed the text by change to lowercase while label is preprocessed using one hot encoding. Following that, the dataset is split into train, validate and test set. After that, the word embedding of Glove is loaded for transfer learning the weight of each word which represent the relationship between words. After that, the vocabulary dictionary is built using train set and the model started to be train using train set and evaluate by validation set after defining those hyperparameters such as input dimension, hidden dimension, number of epoch and so forth. If the current loss is lower than best validation loss, the model is saved to achieve the earlier stopping effect. The test set is used to test the model saved for obtaining the accuracy and loss in unseen data. Lastly, the record of loss and accuracy are recorded and model is trained with different architecture such as Long Short Term

Memory (LSTM), Recurrent Neural Network (RNN) and so forth which discussed in Chapter 4.

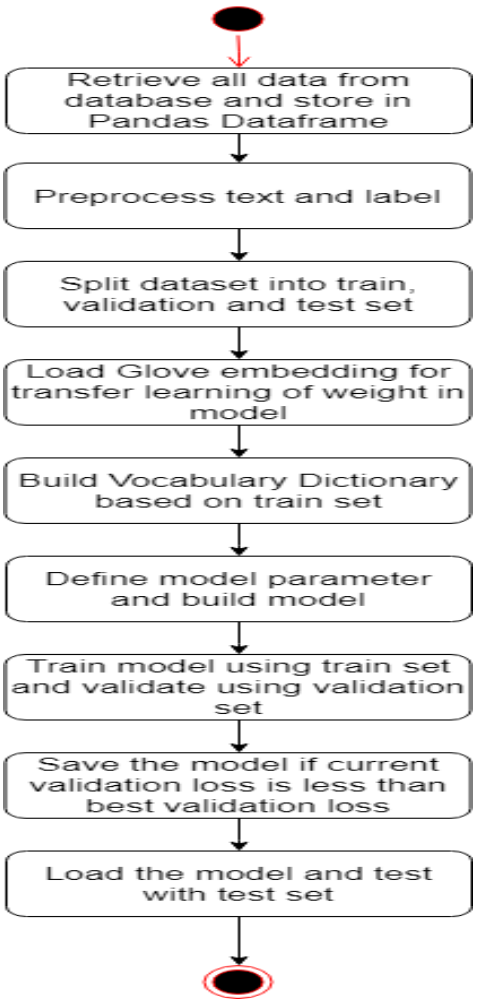


Figure 3-7Activity Diagram for Sentiment and Intent Training

3.3.7 Activity Diagram for Sentiment and Intent Prediction

Figure 3-8 shows the Activity Diagram for Sentiment and Inquiry Intent Prediction. As mentioned in the previous section where three of them are different models but steps are the same. The preprocessing steps are similar with 3.3.6 Activity Diagram for Sentiment and Intent Training. After that, the model used to perform the prediction is selected and loaded after comparison on the performance of difference models. The prediction result is then returned and outputted.

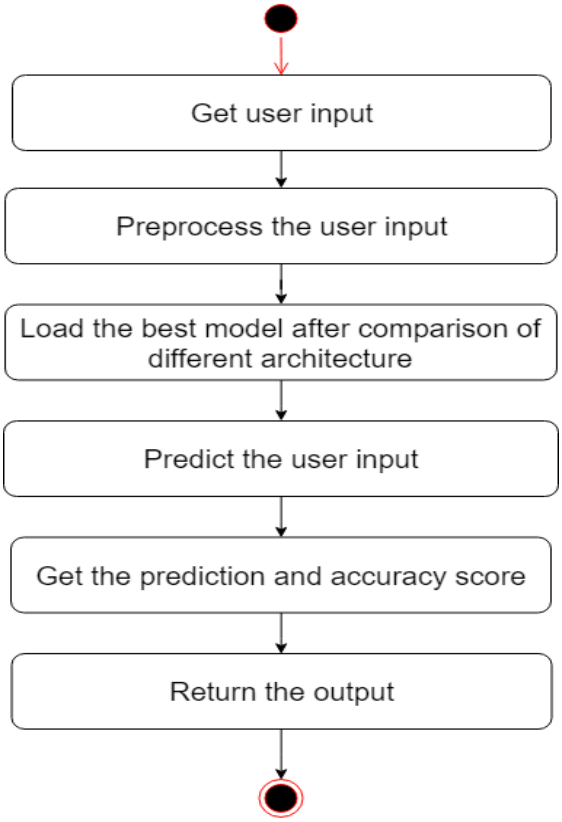


Figure 3-8 Activity Diagram for Sentiment and Intent Prediction

3.3.8 Class Diagram

Figure 3-9 shows the Class Diagram that contains Account, Cinema, Cinema User, Movie, Movie Details, Ticket, Hall, Inquiry Intent, Utterance, Response, Sentiment, Contact, Access Control and Login Record. All of the attributes and methods are shown in Figure 3-9. The methods are used to query the database for “Select”, “Insert”, “Update” and “Delete”. Each of the attributes and their respective description is discussed in detail in Section 3.3.10 Database Design. As to the methods started with “Get” is to “Select” the data from the database while methods started with “Insert” is to “Insert” the data to the database. In addition, the “Update” methods are “Update” the data in the database whereas “Delete” methods are remove the data from being viewed by user.

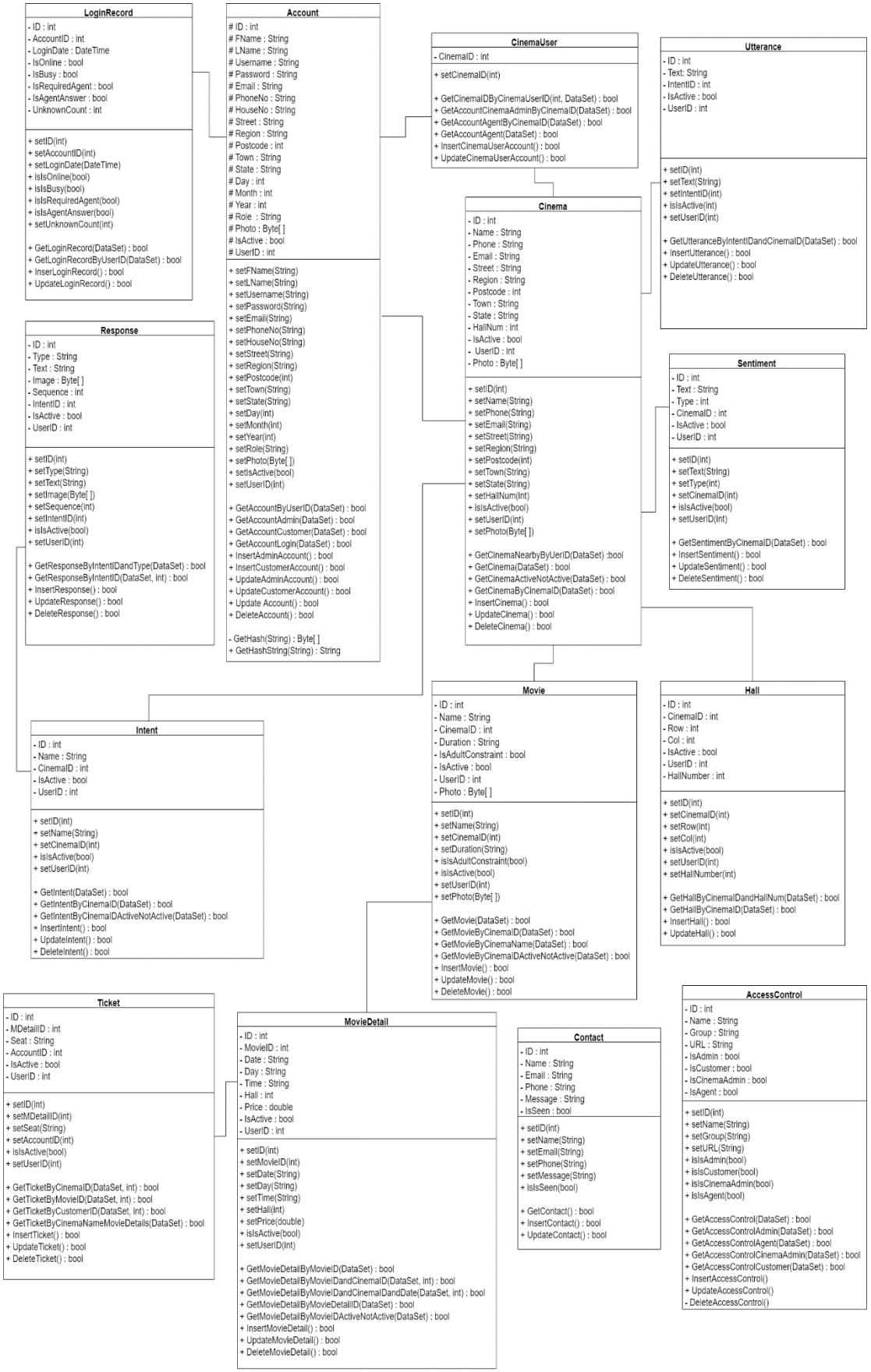


Figure 3-9 Class Diagram for Proposed System

3.3.9 Entity Relationship Diagram (ERD)

Figure 3-10 shows the Entity Relationship Diagram (ERD) for the proposed system. The Entities are similar with the Classes in Class Diagram as shown in Figure 3-9. As to the attributes, the details description is discussed in Section 3.3.10 Database Design.

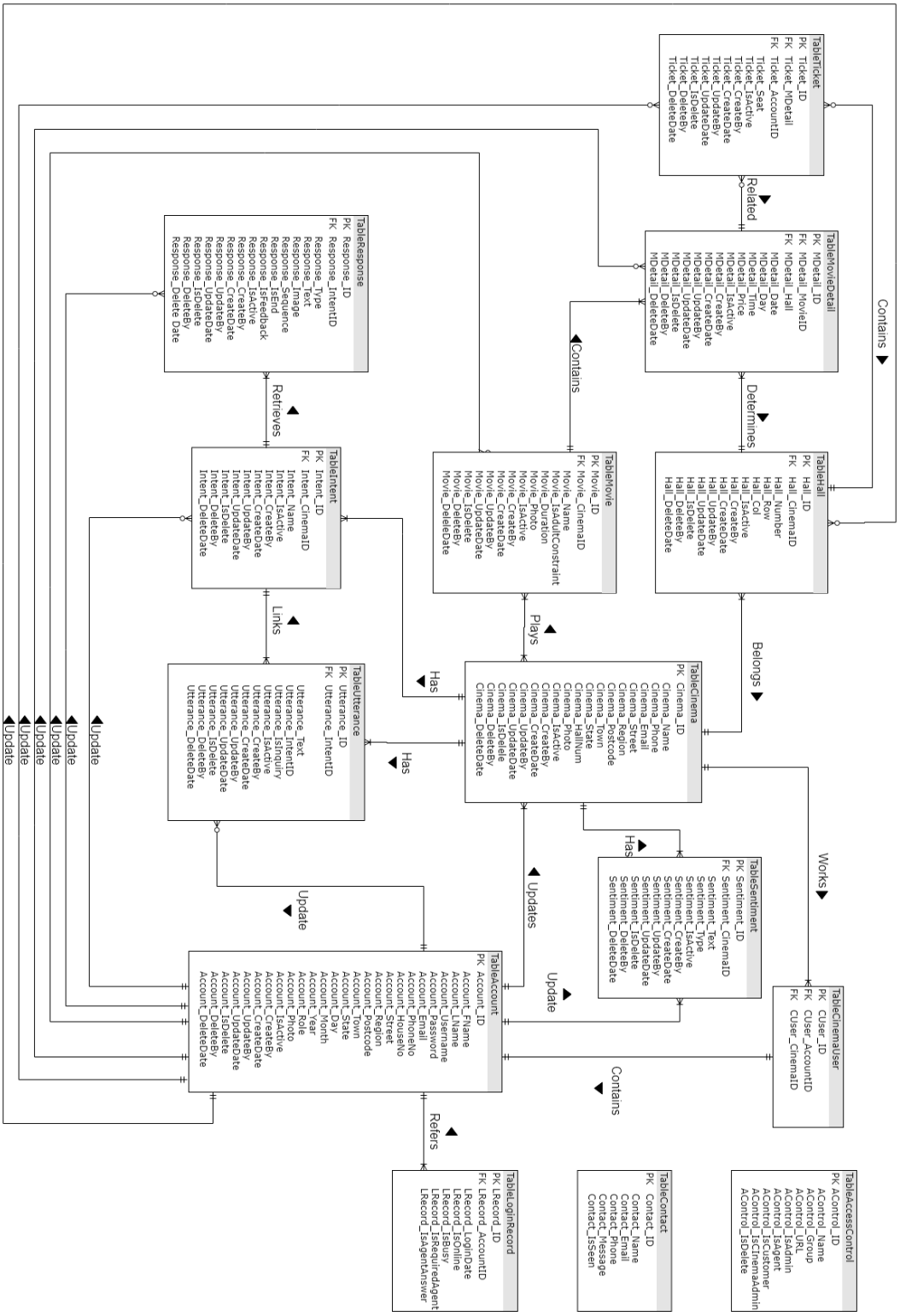


Figure 3-10 ERD for Proposed System

3.3.10 Database Design

All of the tables in this section shows the Entities in Entity Relationship Diagram (ERD) and their respective attributes. Each and every attribute within the Entities are described with variable name, data type, length and description of the use of that particular attribute.

Variable	Data Type	Length	Description
AControl_ID	Int		Primary Key
AControl_Name	Varchar	100	Record name displayed in Menu
AControl_Group	Varchar	100	Record name displayed in Drop Down List
AControl_URL	Varchar	1000	Record file path of website for
AControl_IsAdmin	Bit	1	Determine whether allow Admin to access the webpage
AControl_IsAgent	Bit	1	Determine whether allow Agent to access the webpage
AControl_IsCustomer	Bit	1	Determine whether allow Customer to access the webpage
AControl_IsCinemaAdmin	Bit	1	Determine whether allow Cinema Admin to access the webpage
AControl_IsDelete	Bit	1	Determine whether the record is deleted

Table 3-1 Table Access Control

Variable	Data Type	Length	Description
Account_ID	Int		Primary Key
Account_FName	Varchar	100	Record first name of the user
Account_LName	Varchar	100	Record last name of the user
Account_Username	Varchar	100	Record Login Credentials – Username
Account_Password	Varchar	100	Record Login Credentials – Password

Account_Email	Varchar	100	Record email of the user
Account_PhoneNo	Varchar	20	Record phone number of the user
Account_HouseNo	Varchar	10	Record house number of Address
Account_Street	Varchar	100	Record street of Address (Jalan Aman)
Account_Region	Varchar	50	Record region of Address (Taman Asia)
Account_Postcode	Int		Record postcode of Address
Account_Town	Varchar	50	Record town of Address (Ipoh)
Account_State	Varchar	50	Record state of Address (Perak)
Account_Day	Int	2	Record day of Birth Date
Account_Month	Int	2	Record month of Birth Date
Account_Year	Int	4	Record year of Birth Date
Account_Role	Varchar	50	Record role of user (Admin, Customer, Agent, CinemaAdmin)
Account_Photo	Image		Record photo of the user
Account_IsActive	Bit	1	Determine whether the user still able to Login to the system
Account_CreateBy	Int		Record person who create the user account
Account_CreateDate	Datetime		Record date of create the user account
Account_UpdateBy	Int		Record person who update the user account
Account_UpdateDate	Datetime		Record date of update the user account
Account_IsDelete	Bit	1	Determine whether the user account is deleted
Account_DeleteBy	Int		Record person who delete the user account
Account_DeleteDate	Datetime		Record date of delete the user account

Table 3-2 Table Account

Variable	Data Type	Length	Description
Cinema_ID	Int		Primary Key
Cinema_Name	Varchar	100	Record name of cinema
Cinema_Phone	Varchar	20	Record phone number of cinema
Cinema_Email	Varchar	100	Record email of cinema
Cinema_Street	Varchar	100	Record street of Address (Jalan Aman)
Cinema_Region	Varchar	50	Record region of Address (Taman Asia)
Cinema_Postcode	Int		Record postcode of Address
Cinema_Town	Varchar	50	Record town of Address (Ipoh)
Cinema_State	Varchar	50	Record state of Address (Perak)
Cinema_HallNum	Int	2	Record total Number of hall in cinema
Cinema_IsActive	Bit	1	Determine whether the user still able to Login to the system
Cinema_CreateBy	Int		Record person who create the cinema
Cinema_CreateDate	Datetime		Record date of create the cinema
Cinema_UpdateBy	Int		Record person who update the cinema
Cinema_UpdateDate	Datetime		Record date of update the cinema
Cinema_IsDelete	Bit	1	Determine whether the cinema is deleted
Cinema_DeleteBy	Int		Record person who delete the cinema
Cinema_DeleteDate	Datetime		Record date of delete the cinema

Table 3-3 Table Cinema

Variable	Data Type	Length	Description
CUser_ID	Int		Primary Key

CUser_AccountID	Int		Record Account ID of Cinema Admin/Agent (reference Account_ID in TableAccount)
CUser_CinemaID	Int		Record Cinema ID of Cinema Admin/Agent belongs to (reference Cinema_ID in TableCinema)

Table 3-4 Table Cinema User

Variable	**Data Type**	**Length**	**Description**
Contact_ID	Int		Primary Key
Contact_Email	Varchar	100	Record email of person who filled in the contact form
Contact_Phone	Varchar	20	Record phone number of person who filled in the contact form
Contact_Message	Varchar	3000	Record the message from contact form
Contact_IsSeen	Bit	1	Record whether the message read by admin
Contact_Name	Varchar	100	Record name of person who filled in the contact form

Table 3-5 Table Contact

Variable	**Data Type**	**Length**	**Description**
Hall_ID	Int		Primary Key
Hall_Number	Int		Record the number of hall (Hall 1, Hall 2, …)
Hall_CinemaID	Int		Record Cinema ID where the hall belongs (reference Cinema_ID in TableCinema)
Hall_Row	Int		Record the number of rows in the cinema
Hall_Col	Int		Record the number of columns in the cinema

Hall_IsActive	Bit	1	Determine whether the hall is open to user for further process
Hall_CreateBy	Int		Record person who create the hall
Hall_CreateDate	Datetime		Record date of create the hall
Hall_UpdateBy	Int		Record person who update the hall
Hall_UpdateDate	Datetime		Record date of update the hall

Table 3-6 Table Hall

Variable	**Data Type**	**Length**	**Description**
Intent_ID	Int		Primary Key
Intent_Name	Varchar	100	Record Inquiry Intent Name
Intent_CinemaID	Int		Record Cinema ID where Inquiry Intent belongs (reference Cinema_ID in TableCinema)
Intent_IsActive	Bit	1	Determine whether the record is shown in another webpage other than Indent.aspx
Intent_CreateBy	Int		Record person who create the inquiry intent
Intent_CreateDate	Datetime		Record date of create the inquiry intent
Intent_UpdateBy	Int		Record person who update the inquiry intent
Intent_UpdateDate	Datetime		Record date of update the inquiry intent
Intent_IsDelete	Bit	1	Determine whether the inquiry intent is deleted
Intent_DeleteBy	Int		Record person who delete the inquiry intent
Intent_DeleteDate	Datetime		Record date of delete the inquiry intent

Table 3-7 Table Inquiry Intent

Variable	Data Type	Length	Description
LRecord_ID	Int		Primary Key
LRecord_AccountID	Int		Record the user account id login (reference Account_ID in TableAccount)
LRecord_LoginDate	Datetime		Record the datetime user login
LRecord_IsOnline	Bit	1	Determine whether user join chat room
LRecord_IsBusy	Bit	1	Determine whether user chat with agent and vice versa
LRecord_IsRequiredAgent	Bit	1	Determine whether user call for agent
LRecord_IsAgentAnswer	Bit	1	Determine whether agent handling the user
LRecond_UnknownCount	Int		Determine how many time chatbot unable to answer

Table 3-8 Table Login Record

Variable	Data Type	Length	Description
Movie_ID	Int		Primary Key
Movie_Name	Varchar	100	Record name of movie
Movie_CinemaID	Int		Record Cinema ID where movie belongs (reference Cinema_ID in TableCinema)
Movie_IsAdultConstraint	Bit	1	Determine whether movie is under category P18
Movie_Duration	Varchar	100	Record the duration of the movie
Movie_Photo	Image		Record the photo of the movie
Movie_IsActive	Bit	1	Determine whether the movie can be booked
Movie_CreateBy	Int		Record person who create the movie
Movie_CreateDate	Datetime		Record date of create the movie

Movie_UpdateBy	Int		Record person who update the movie
Movie_UpdateDate	Datetime		Record date of update the movie
Movie_IsDelete	Bit	1	Determine whether the movie is deleted
Movie_DeleteBy	Int		Record person who delete the movie
Movie_DeleteDate	Datetime		Record date of delete the movie

Table 3-9 Table Movie

Variable	**Data Type**	**Length**	**Description**
MDetail_ID	Int		Primary Key
MDetail_MovieID	Int		Record Movie ID (reference Movie_ID in TableMovie)
MDetail_Date	Varchar	100	Record the date movie play
MDetail_Day	Varchar	100	Record the day movie play
MDetail_Time	Varchar	100	Record the time movie play
MDetail_Hall	Int		Record the hall number (reference Hall_Num in TableHall)
MDetail_Price	Float		Record the price of ticket
MDetail_IsActive	Bit	1	Determine whether the movie detail can be booked
MDetail_CreateBy	Int		Record person who create the movie detail
MDetail_CreateDate	Datetime		Record date of create the movie detail
MDetail_UpdateBy	Int		Record person who update the movie detail
MDetail_UpdateDate	Datetime		Record date of update the movie detail
MDetail_IsDelete	Bit	1	Determine whether the movie detail is deleted

MDetail_DeleteBy	Int		Record person who delete the movie detail
MDetail_DeleteDate	Datetime		Record date of delete the movie detail

Table 3-10 Table Movie Detail

Variable	**Data Type**	**Length**	**Description**
Response_ID	Int		Primary Key
Response_Type	Varchar	100	Record type of response (Text, Image, Quick Reply, Carousel)
Response_Text	Varchar	3000	Record the text response
Response_Image	Image		Record the image response
Response_Sequence	Int		Record the order of message to be send
Response_IntentID	Int		Record Inquiry Intent ID (reference IIntent_ID in TableInquiryIntent)
Response_IsActive	Bit	1	Determine whether the response need to be sent
Response_CreateBy	Int		Record person who create the response
Response_CreateDate	Datetime		Record date of create the response
Response_UpdateBy	Int		Record person who update the response
Response_UpdateDate	Datetime		Record date of update the response
Response_IsDelete	Bit	1	Determine whether the response is deleted
Response_DeleteBy	Int		Record person who delete the response
Response_DeleteDate	Datetime		Record date of delete the response

Table 3-11 Table Response

Variable	Data Type	Length	Description
Sentiment_ID	Int		Primary Key
Sentiment_Text	Varchar	3000	Record the text of sentiment for training
Sentiment_Type	Int	1	Record the sentiment (1=Positive, 0=Negative)
Sentiment_CinemaID	Int		Record Cinema ID where sentiment belongs (reference Cinema_ID in TableCinema)
Sentiment_IsActive	Bit	1	Determine whether the sentiment need to be trained
Sentiment_CreateBy	Int		Record person who create the sentiment
Sentiment_CreateDate	Datetime		Record date of create the sentiment
Sentiment_UpdateBy	Int		Record person who update the sentiment
Sentiment_UpdateDate	Datetime		Record date of update the sentiment
Sentiment_IsDelete	Bit	1	Determine whether the sentiment is deleted
Sentiment_DeleteBy	Int		Record person who delete the sentiment
Sentiment_DeleteDate	Datetime		Record date of delete the sentiment

Table 3-12 Table Sentiment

Variable	Data Type	Length	Description
Ticket_ID	Int		Primary Key
Ticket_MDetailID	Int		Record the Movie Detail ID (reference MDetail_ID in TableMovieDetail)
Ticket_Seat	Varchar	10	Record the seat

Ticket_AccountID	Int		Record the Customer ID who bought the ticket (reference Account_ID in TableAccount)
Ticket_IsActive	Bit	1	Determine whether the ticket is valid
Ticket_CreateBy	Int		Record person who create the ticket
Ticket_CreateDate	Datetime		Record date of create the ticket
Ticket_UpdateBy	Int		Record person who update the ticket
Ticket_UpdateDate	Datetime		Record date of update the ticket
Ticket_IsDelete	Bit	1	Determine whether the ticket is deleted
Ticket_DeleteBy	Int		Record person who delete the ticket
Ticket_DeleteDate	Datetime		Record date of delete the ticket

Table 3-13 Table Ticket

Variable	**Data Type**	**Length**	**Description**
Utterance_ID	Int		Primary Key
Utterance_Text	Varchar	3000	Record text of utterance
Utterance_IntentID	Int	1	Record Intent ID (1=Inquiry, 2=Ticket Booking, 3=Others)
Utterance_IsActive	Bit	1	Determine whether the utterance is valid
Utterance_CreateBy	Int		Record person who create the utterance
Utterance_CreateDate	Datetime		Record date of create the utterance
Utterance_UpdateBy	Int		Record person who update the utterance
Utterance_UpdateDate	Datetime		Record date of update the utterance
Utterance_IsDelete	Bit	1	Determine whether the utterance is deleted

Utterance_DeleteBy	Int		Record person who delete the utterance
Utterance_DeleteDate	Datetime		Record date of delete the utterance

Table 3-14 Table Utterance

3.4 Methods and Technology Involved

In this project, the programming languages use are C# and Python. ASP.NET C# is the object-oriented programming framework for developing the website where it allows tidy up the code and reused. In addition, ASP.NET is the combination of user interface and server-side technologies in which developers can code for frontend (Interface) and backend (Logic) separately. Furthermore, ASP.NET allows the development of websites using drag and drop and the design of the component can be done using the "Properties" section. Python is used in Pytorch for deep learning classification of sentiment and intent as well as translation. The reason to choose Pytorch is because it can build the layers of architecture and is easy to debug. Pytorch is also popular among researchers so there are a lot of supports for the error such as syntax error and runtime error. Yet another reason for choosing Python instead of other languages is because Google has provided a free environment for the model training with the condition not exceeded 12 hours per day.

Besides, this project is developed using the web platform as it is supportive in Android, IOS and Windows. This is allowed the developer less worry about the platform problems and focused on the development of the functionality of the proposed system. In other words, the web platform is able to display the content of the chatbot more dynamically as the web platform can be displayed in different size of devices such as smart phone, laptop and tablet.

Last but not least, the supervised machine learning is used for training the intent, sentiment and translation model. This is because supervised machine learning can avoid the users of the system teaching the chatbot with some nonsense such as rude words and incorrect information. Therefore, reinforcement learning is not use in this project. After training the model, the model is saved so that the prediction of the user input for sentiment analysis and intent become faster.

Lastly, the website system of chatbot is hosted using the Internet Information Services Manager (IIS). This is because it is prebuilt in Windows 10 and only configuration needed to be done for the first time. IIS is also suitable to be used especially during the testing phase because the deployment and remove from deployment can be easily done as it is in local host. Nonetheless, the deep learning natural language processing models is hosted using Git Bash with the help of Flask that made the services available to the ASP.NET C# since both languages were unable to communicate directly.

3.5 Tools, Hardware and Software

The Hardware, Software and Tools required in the development and implementation of the proposed system were shown in Table 3-15, Table 3-16 and Table 3-17.

Hardware:

1. Laptop

Processor	Intel(R) Core (TM) i7-8750H CPU @ 2.2GHz
RAM	12GB
Storage	512GB SSD (NVME PCIe) + 1TB 5400rpm HDD
Graphic Card	NVIDIA GeForce GTX 1050Ti GPU
Battery	48Wh, 120W charger

Table 3-15 Hardware Specification

2. Software:

Operating System	Windows 10
Programming Environment	Microsoft Visual Studio Community 2019, Spyder (Anaconda)
Database System	Microsoft SQL Server Management Studio 18
Hosting Environment	Internet Information Services Manager, Git Bash
Browser	Google Chrome, Mozilla Firefox

Table 3-16 Software Specification

3. Tools:

Framework	ASP.NET C#, Python
Web Development Framework	DevExpress
Deep Learning Library	Pytorch
GET, POST Request Framework	Flask (Python)
Real Time Functionality Library	SignalR
Model Training Platform	Google Colab

Table 3-17 Tools Specification

As shown in Table 3-15, the specification of the hardware which is a laptop and it is set to be quite high for the training of the model. Even though the model is trained in Google Colab but a smaller dataset is used to train the model to make sure it is error free in another computer. However, the training speed does not affect the performance of the system as the model is saved during training.

As to the software which is shown in Table 3-16, the operating system of Windows 10 will be chosen. After that, Microsoft Visual Studio Community 2019 is used as the programming tool because it contains a lot of convenience tools for debugging such as breakpoint. In addition, Microsoft Visual Studio Community also provides the auto-indentation feature and notifies the error made. In addition, the Spyder (Anaconda) is also used in this project because Pytorch is supportive in Python language. Spyder is used instead of Jupyter Notebook because of the easier debug as Spyder provides the view of the contents stored in the variables. The Microsoft SQL Server Management Studio is used because the syntax is simpler and easier. Microsoft SQL Server Management Studio also provides technical support on the official website regarding the bugs, scripts, notes and patches as well as no additional charging when downloading it (Guru99, 2019). The hosting environment used is Internet Information Services Manager because it is already built in Windows 10 and only required configuration to make use of it. Besides, the Git Bash is also being used because IIS is unable to host the Python extension of source code. Therefore, Git Bash is used to act as the hosting platform for the Python Source Code. The browser used is Google Chrome and Mozilla Firefox because they supported WebKit features. However, the WebKit JavaScript features and similar JavaScript features are taken into consideration in order to allow the system supported by all of the browser.

Last but not least, the tools as shown in Table 3-17 which is ASP.NET C# is used with the DevExpress for developing the website which is the interface of the chatbot. As to the Python language, it is used with the companion of Pytorch. Therefore, Flask is used since ASP.NET C# is unable to direct communicate with Python and the role of Flask is to provide services for the ASP.NET C#. Following that, the SignalR library is used which allowed the users getting the message update without keeping refresh the browser for obtaining the message. Following that, Google Colab is used to train the huge model since the time taken for model training in Laptop mentioned above is slow. With the help of Google Colab, the testing and validation of models' accuracy can be done faster as compare to Laptop.

3.6 Timeline

Figure 3-12 shows the timeline of the proposed system. In the timeline, there are mainly five steps from waterfall model. The timeline looks like a waterfall as it is generated based on the style of waterfall model which back to previous stage is almost impossible. This is because the start date of next task is set just right after the completion of previous tasks. The details of timeline are shown in Figure 3-11 and Figure 3-12.

	Task Name	Duration	Start	ETA
1	Complete project execution	128 days	14.10.2019	21.11.2019
2	Requirement Analysis and Definition	17 days	14.10.2019	30.10.2019
3	List out user requirement	2 days	14.10.2019	15.10.2019
4	Draw UML Diagram	8 days	16.10.2019	23.10.2019
5	Research on NLP	5 days	24.10.2019	28.10.2019
6	Check completeness and correctness of Diagrams	2 days	29.10.2019	30.10.2019
7	System Design and Unit Testing	17 days	31.10.2019	16.11.2019
8	Design interface of webpage	6 days	31.10.2019	05.11.2019
9	Provide functionality and validation to webpage	6 days	06.11.2016	11.11.2019
10	Develop Deep NLP and Train Model	3 days	12.11.2019	14.11.2019
11	Test Deep NLP model	2 days	15.11.2019	16.11.2019
12	System Integration and System Testing	3 days	17.11.2019	19.11.2019
13	Integrate all source code	2 days	17.11.2019	18.11.2019
14	Testing integrated system	1 day	19.11.2019	19.11.2019
15	Reporting and Documentation	2 days	20.11.2019	21.11.2019
16	Writing Report	2 days	20.11.2019	21.11.2019
17	Maintenance	89 days	13.01.2020	10.04.2020
18	Refinement of existing interface	7 days	13.01.2020	19.01.2020
19	Improvement of Deep NLP	35 days	20.01.2020	23.02.2020
20	Research on Translation using Seq2Seq	21 days	24.02.2020	15.03.2020
21	Improve the Chatbot with different response type	7 days	16.03.2019	22.03.2019
22	Integrate and undergo black box testing	7 days	23.03.2020	29.03.2020
23	Writing Report and documentation	7 days	30.03.2020	05.04.2020
24	Checking and refinement of report	5 days	06.04.2020	10.04.2020

Figure 3-11 Timeline

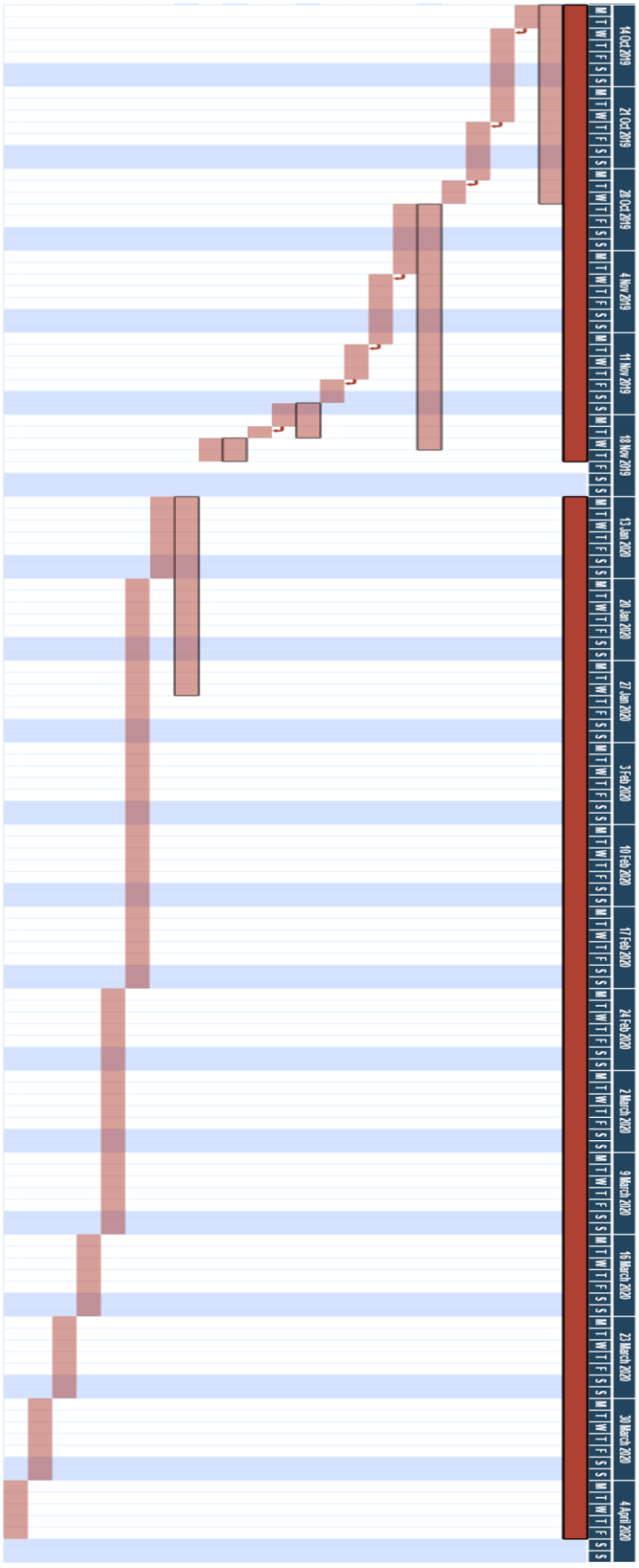


Figure 3-12 Timeline

3.7 Summary

This chapter has discussed the methodology of the waterfall model. There are mainly five stages in the waterfall model. In every stage, the task is being done and accomplished within the duration allocated. This is to ensure the development of the project does not fall after time and is able to be delivered on time. In addition, by following the stages in the waterfall model the bug and error can be reduced as testing will be carried out during the integration of the module into a single system. Then, methods and technology involved are also determined so that the tools, hardware and software are able to be identified. Lastly, the system overview with the representation of UML diagram is shown so that the requirement and process of the system is clear.

CHAPTER 4 SYSTEM DEVELOPMENT AND DISCUSSION

This chapter shows the user interface and the functionality available in the proposed system. The demonstrations of system are based on different actors as drawn in use case diagram. The similar interface is mentioned in the starting paragraph of each actors.

4.1 General Function for All User

These functionalities are applied to all of the actors shown in the Use Case in Figure 3-2.

4.1.1 Main Layout

This is the main page (Figure 4-1) where the user will be caught sight of initially when browsing the Cinema Chatbot Web Application.

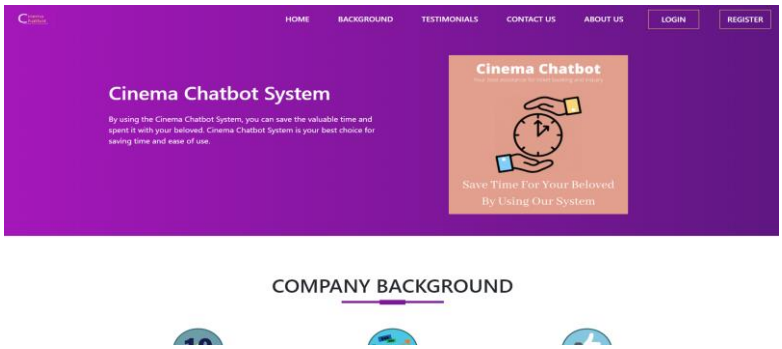


Figure 4-1 Main Page of Proposed System

When the user scrolls down the same webpage, the About Us and Contact Us sections will be shown as in Figure 4-2 below. The user can contact the admin of the webpage by filling in the form. There are validations which are included not to allow null and validate format of Email and Phone Number in the form before submitting to the admin.

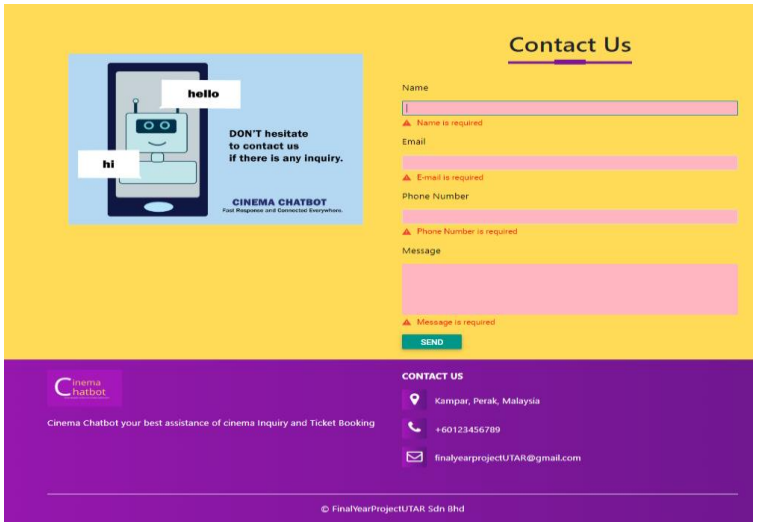


Figure 4-2 Contact Us and About Us Layout

Before proceeding for other functionalities, users must Login via the login credentials. The "Username" and "Password" are required fields as shown in Figure 4-3. When the account is invalid there is an error message as shown in Figure 4-4. If the valid Login Credentials are correct, then the user is redirected to respective Dashboard.

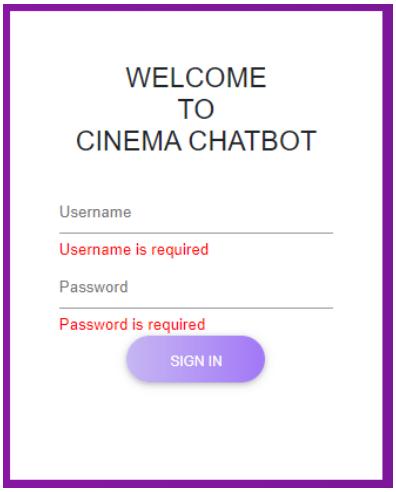


Figure 4-3 Username and Password are required

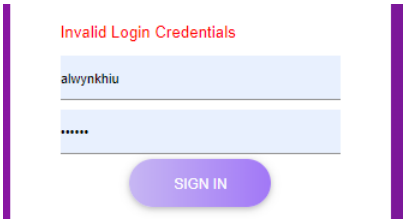


Figure 4-4 Invalid Login Credentials

4.1.2 Account Management

Figure 4-5 shows the Account Management Interface. This is the interface for all users to modify respective information. There is validation of required fields and format for the fields. As to the password, it has been masked. However, when the users modify the password, the plain text is shown to avoid typing error as shown in Figure 4-6. As to the "Birth Date", users can modify in the calendar displayed as shown in Figure 4-7.

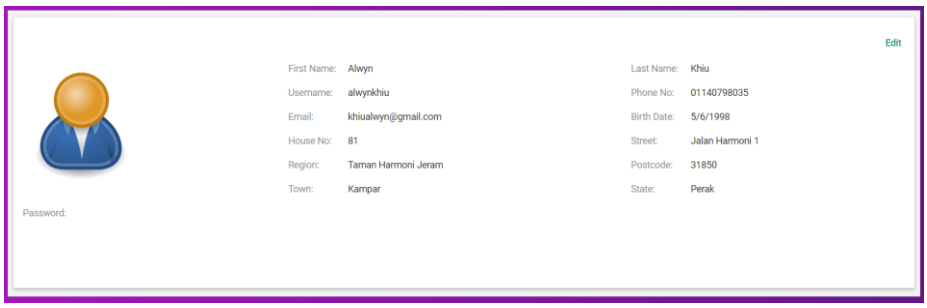


Figure 4-5 Account Management Interface

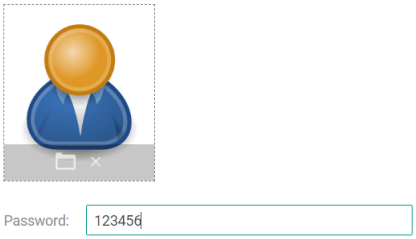


Figure 4-6 Password in Plain Text when Modification

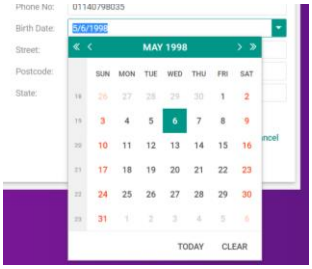


Figure 4-7 Calendar for Birth Date Field

4.1.3 Chat Room

Figure 4-8 shows the interface of Chat Room with the message that sent by the chatbot.

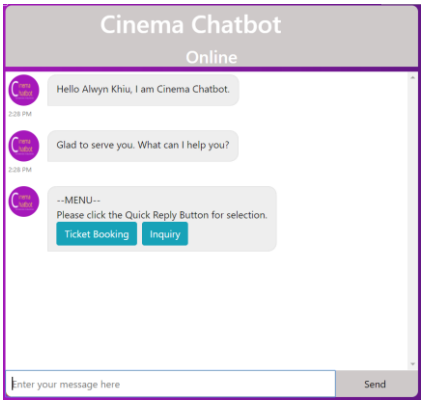


Figure 4-8 Chat Room Interface

When the "Ticket Booking" Quick Reply Button (Figure 4-8) is clicked, the required information such as cinema, date and time will be prompt using Quick Reply as shown in Figure 4-9. As to the movie, it is prompted by showing the Carousel and Quick Reply so that user need not manually key in the movie name. The "P18" with red word and white background is used to alert the user regarding the movie only for those who above 18 years old. Figure 4-11 shows the seat in red means already booked by others and green means still available. After that, the email will be sent to the mailbox of the user as shown in Figure 4-12.

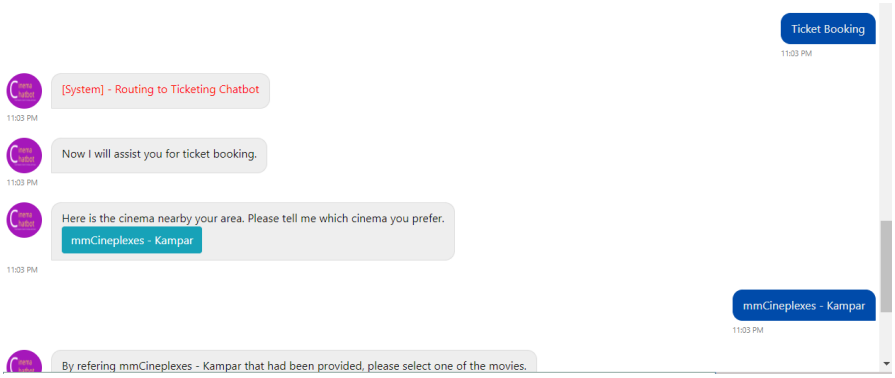


Figure 4-9 Ticket Booking with Quick Reply Button

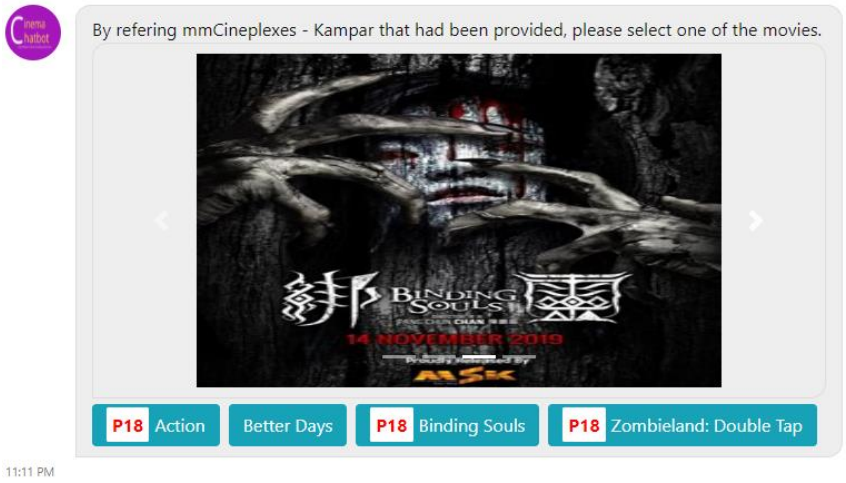


Figure 4-10 Movie Selection with Carousel and Quick Reply Button

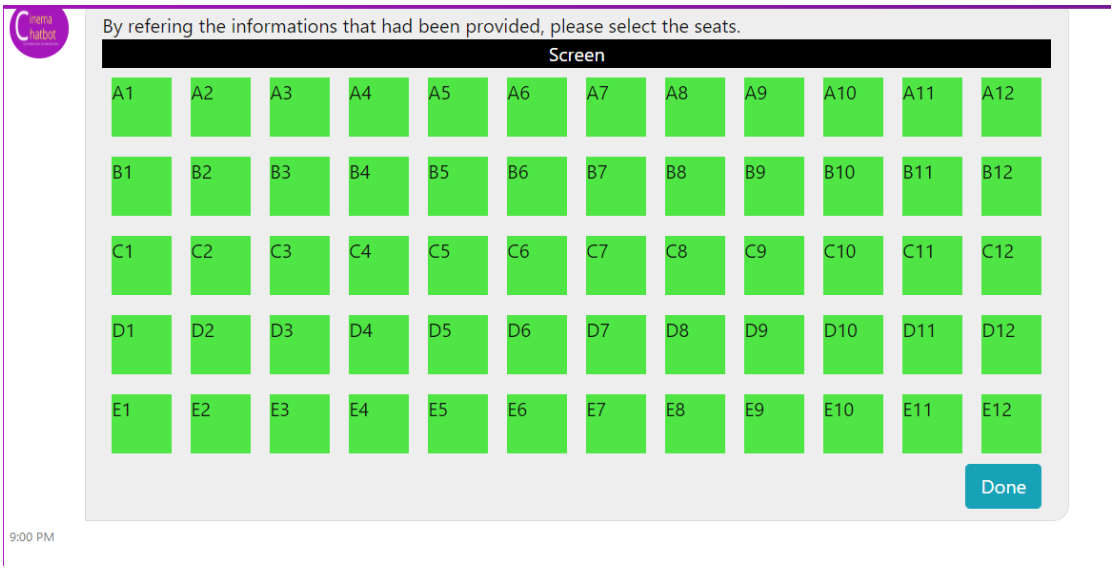


Figure 4-11 Seat Selection in ChatRoom

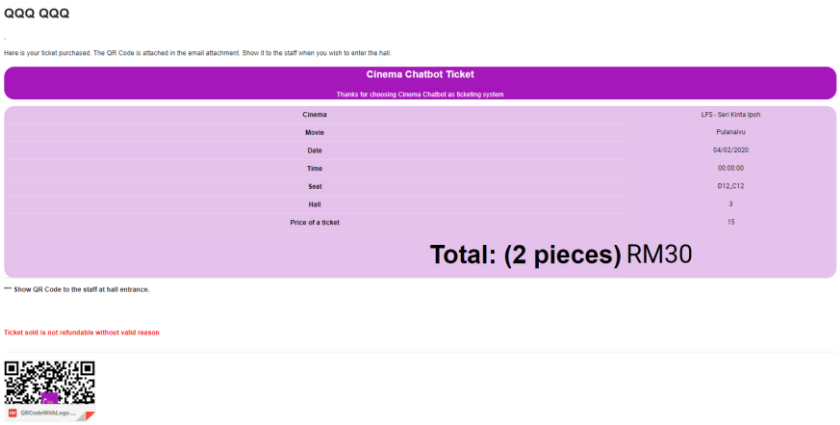


Figure 4-12 Email with Ticket Details and QR Code Attachment

In Figure 4-13, it shows the bot response after processing in Pytorch. It is similar to Figure 4-14 but the Figure 4-14 shows the inquiry asked in Bahasa Malaysia and response also in Bahasa Malaysia. This means that the inquiry in Bahasa Malaysia undergo translation to detect the intent and then response also undergo the translation from English to Bahasa Malaysia. This happens without any clicks for language switching. Figure 4-15 shows the message replied by chatbot when negative sentiment is classified. Nonetheless, positive sentiment does not return any of the messages. There are total four types of response which included Text (Figure 4-13 displaying response for user utterance), Quick Reply (Figure 4-13 for selecting Ticket Booking or Inquiry), Carousel (Figure 4-10 for displaying image of all movies available) and Image (Figure 4-16 for displaying “U”, “P13” and “P18”). If the chatbot classify more than 3 times with unknown intent, the user will be route to live agent as shown in 4.3 User Interface and Functionality of Agent (Figure 4-30).

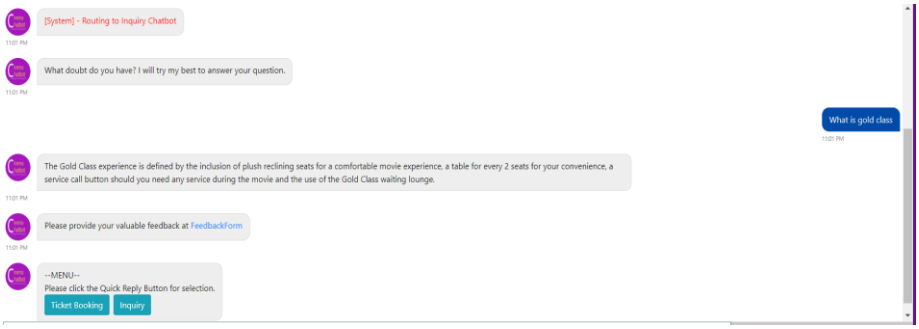


Figure 4-13 Message of User and Bot Response

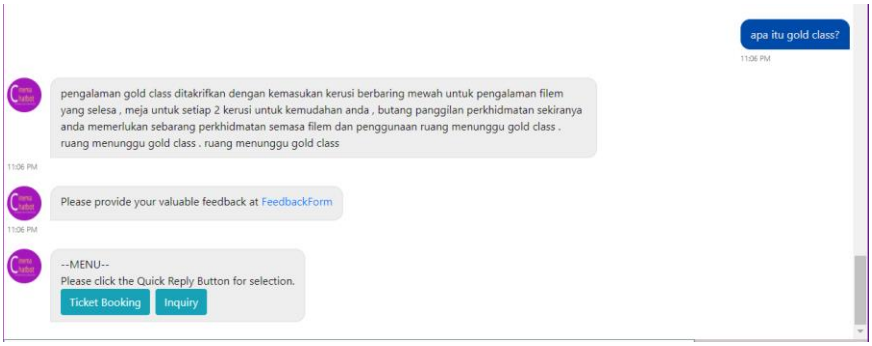


Figure 4-14 Message of User and Bot Response with Translation of Bahasa Malaysia

Figure 4-15 Message Reply for Negative Sentiment

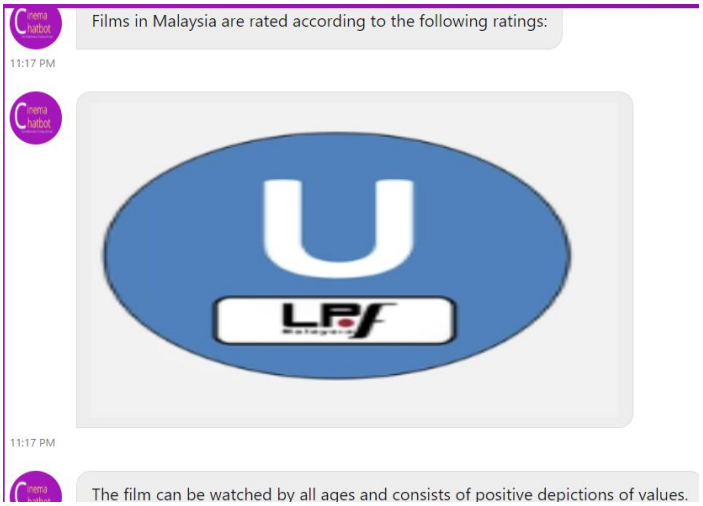


Figure 4-16 Response with Image and Text

4.2 User Interface and Functionality of Customer

The following functionalities are for the Customer actor in the Use Case in Figure 3-2.

4.2.1 Registration

New customers can register themselves with an account by clicking on the "Register" button and fill in all of the details required in the form. In case, the "Confirm Password" and "Password" do not match, this is an error message as shown in Figure 4-18. The "Email" and "Phone Number" also will be validated as shown in Figure 4-17. Lastly, customers are required to upload the photo as well (Figure 4-19).

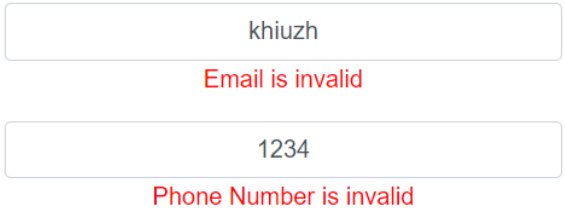


Figure 4-17 Invalid Email and Phone Number

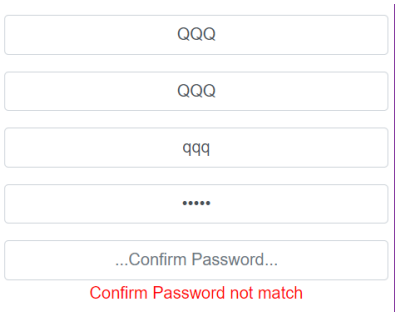


Figure 4-18 Password and Confirm Password Not Match

Figure 4-19 Field to Upload Image

4.2.2 Customer Dashboard and Menu

Figure 4-20 shows the user interface of "Customer Dashboard". The "Logout" button redirected customer to Login Page when being clicked. When the customers scroll down the same webpage, there are two quick booking buttons which are "Chatbot Ticket Booking" and "Manual Ticket Booking" as shown in Figure 4-21. The former is ticket booking with the chatbot while the latter is manually selecting cinemas, movies and so forth. The "hamburger menu icon" button shown in Figure 4-20 is to toggle the menu. The list of menu items such as "Account Management", "Chat Room" and so on is shown in Figure 4-22.

When "Home" is clicked, the Customer is redirected to the main page as mentioned before. However, the difference compared to previous is that the customers

are only provided the “Logout” button and “Name” button (changed according to the customer name). The “Logout” button redirected customers to “Login” Page while the “Name” button redirected customers back to Dashboard as shown in Figure 4-20.

Figure 4-20 Customer Dashboard Interface

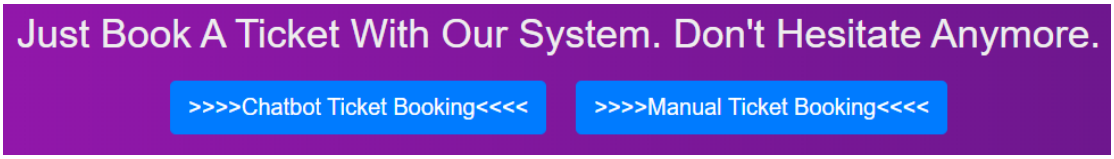


Figure 4-21 Quick Booking Buttons

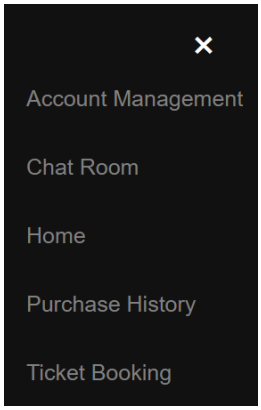


Figure 4-22 Menu with Drop Down List

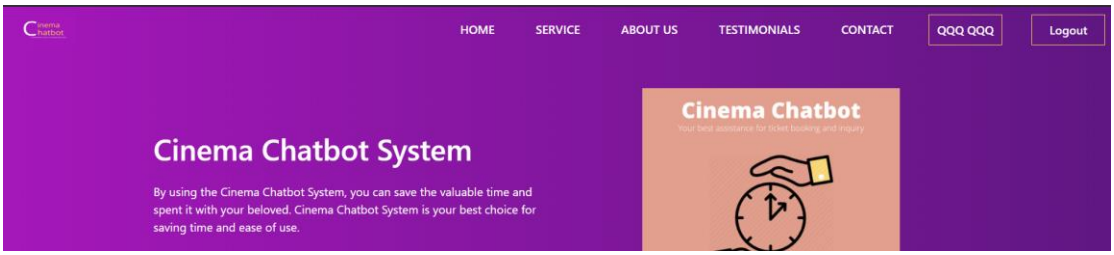


Figure 4-23 Logout and Name button

The “Chat Room” and “Account Management” are same as in Section 4.1.2 Account Management and Section 4.1.3 Chat Room.

4.2.3 Ticket and Purchase History

This is the interface of the Ticket for reservation purpose as shown in Figure 4-24. Customers are required to select the “Cinema”, “Movie”, “Date” and “Time”. After those selections are made, the seat will be displayed and they can select the preference seat as shown in Figure 4-25. The “Red” seats mean already booked by others while “Green” seats mean available. After clicking on the “Purchase” button, the customers are notified with alert message that ticket is sent via email (Figure 4-26) and then redirected to “Purchase History” with ticket details as shown in Figure 4-27. The email contains the information of ticket which is shown in Figure 4-12.

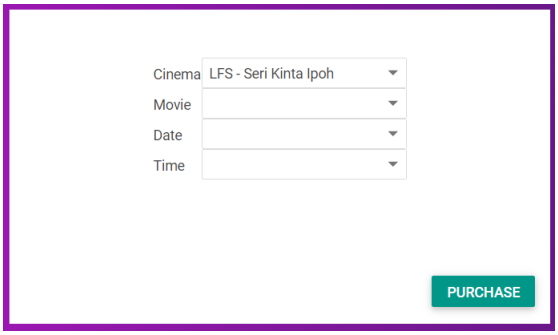


Figure 4-24 Ticket Booking Interface

Figure 4-25 Seat Selection Interface

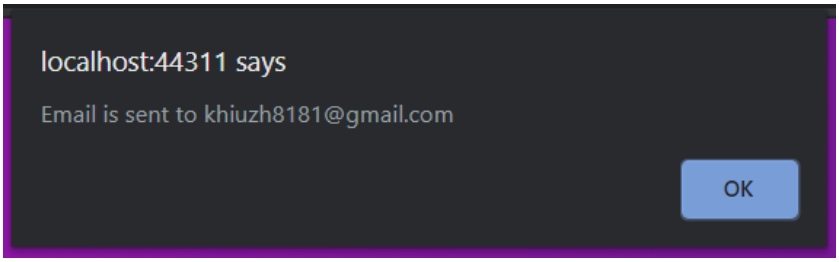


Figure 4-26 Alert Message for email sent

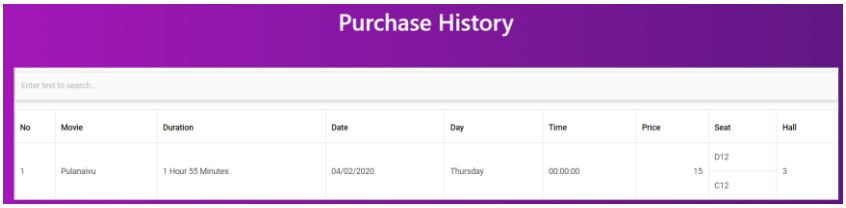


Figure 4-27 Purchase History Interface

4.3 User Interface and Functionality of Agent

The following functionalities are applied to the Agent actor in Use Case Diagram in Figure 3-2. This is the user interface for Agent as shown in Figure 4-28 and Figure 4-29. The functionality of Agent can refer to Section 4.1.2 Account Management. As to the Agent, there are additional functionalities such as "Chatbot Management". However, there is a huge difference from the aspect of Chat Room. When the users required the Agent, the system stopped the chatbot response and route to the Agent as shown in Figure 4-29. When Agent answers the call from the user, the page is directed to the Chat Room which allows the conversation between Agent and User as shown in Figure 4-30. However, when there is no user request for Agent, the Agent Dashboard is refreshed automatically after a certain time. This is to check whether the User requested help.

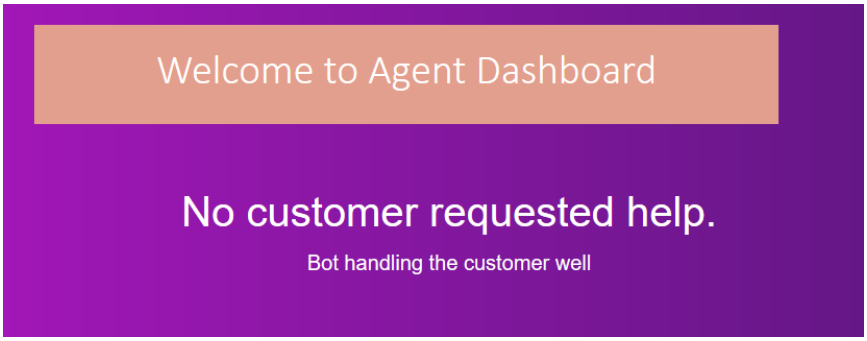


Figure 4-28 Agent Dashboard Without Call

Figure 4-29 Agent Dashboard with Call

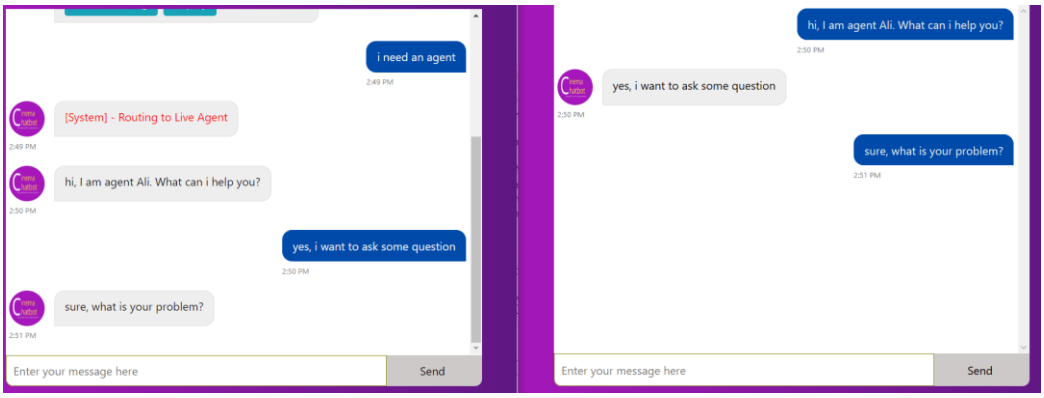


Figure 4-30 Chat Room Between Agent and User

4.3.1 Chatbot Management

Figure 4-31 shows the interface for Chatbot Management.

Intent Management

The “No” column displays the index of intent accordingly. “Intent” is the name of intent while “Update By” and “Update Name” are for recording who and when the record is being modified. “Is Active” is checked when that particular is needed for training. When the Agent presses the “New” button, the popup form as shown in Figure

4-32 is shown. “Update” button plays the role of saving the record to the database. When the “Delete” button is clicked, an alert message is popup for the agent to confirm the deletion of the row as shown in Figure 4-33. Once the “OK” button is clicked, the deletion done and the record will be disappeared. The “Search” bar is to search the content as shown in Figure 4-34. When the Agent clicks the “Header” of the column, the record is sorted as shown in Figure 4-35. Lastly, the Agent can also rearrange the column as shown in Figure 4-36.

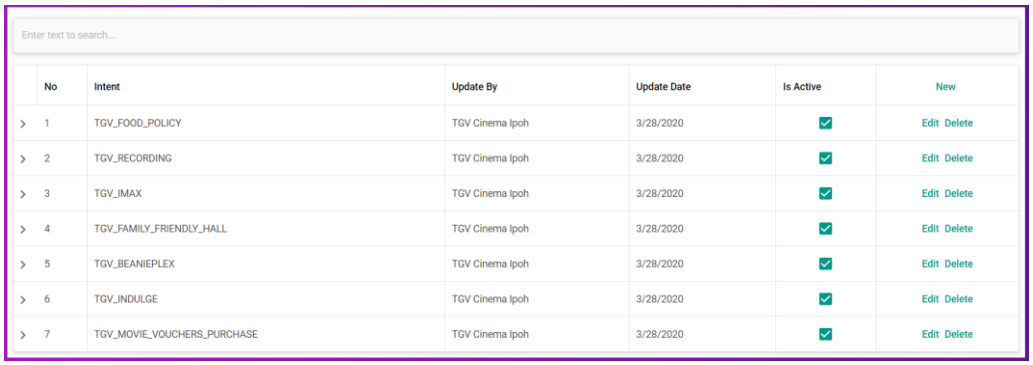


Figure 4-31 Intent Management Interface

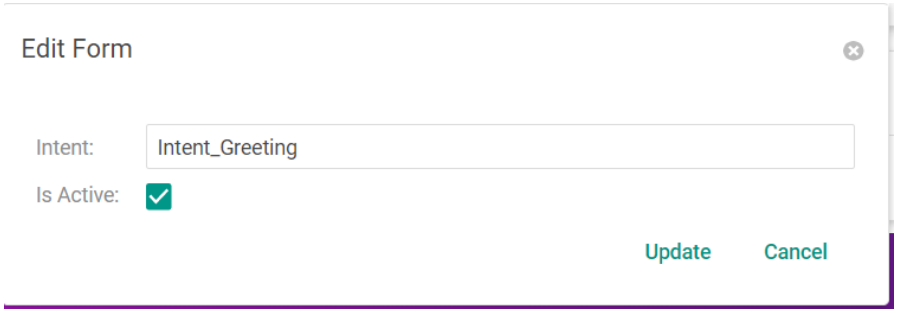


Figure 4-32 Insert, Update form

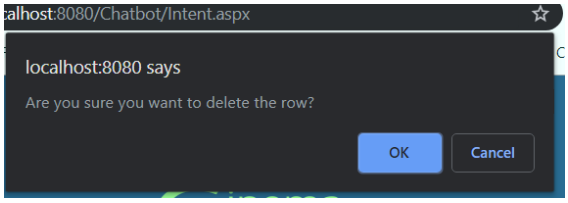


Figure 4-33 Alert Message for Confirmation of Record Deletion

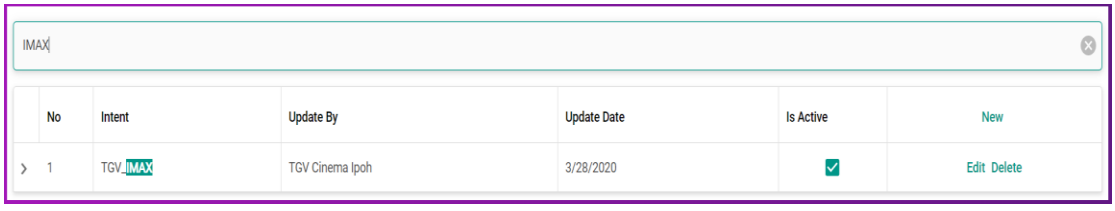


Figure 4-34 Search Function

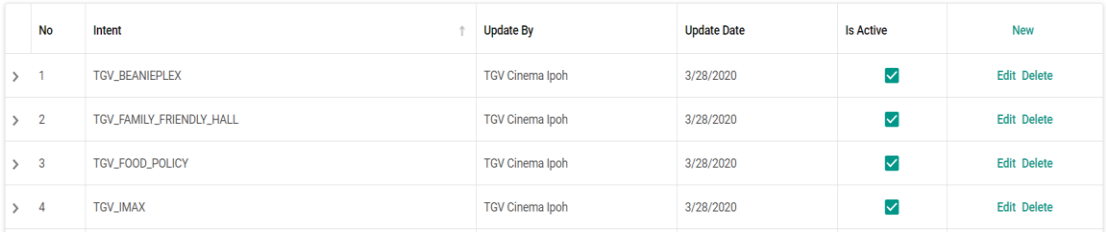


Figure 4-35 Sort the Intent Alphabetically Ascending

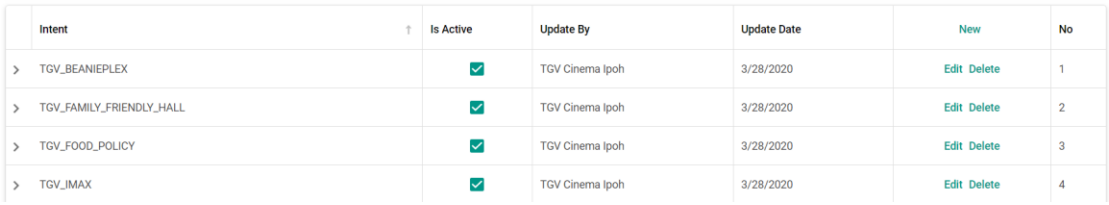


Figure 4-36 Rearrange the column of records

Utterance Management

When the user clicks on ">" at the leftmost of the row, the interface of Utterance Management as shown in Figure 4-37. The "No", "Update By", "Update Date" and "Is Active" are the same functions as mentioned previously. Same goes to the "Add", "Edit", "Delete", "Search", "Sort" and "Rearrange Column" functions. The "Train" button is to train the model in Pytorch and the message is shown as in Figure 4-38. As shown in Figure 4-39, the edit form only required the user to enter the Utterance text.

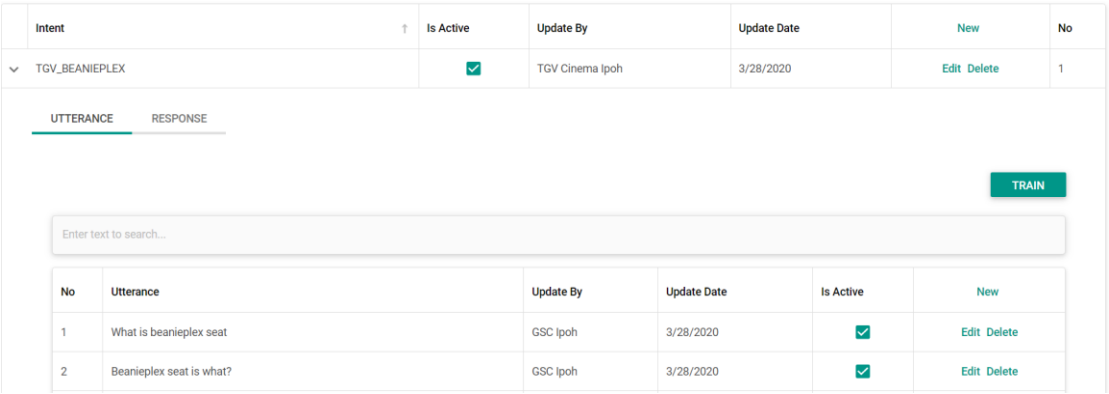


Figure 4-37 Utterance Management Interface

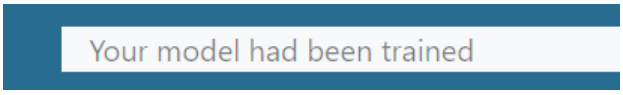


Figure 4-38 Message Displayed After Model Finish Training

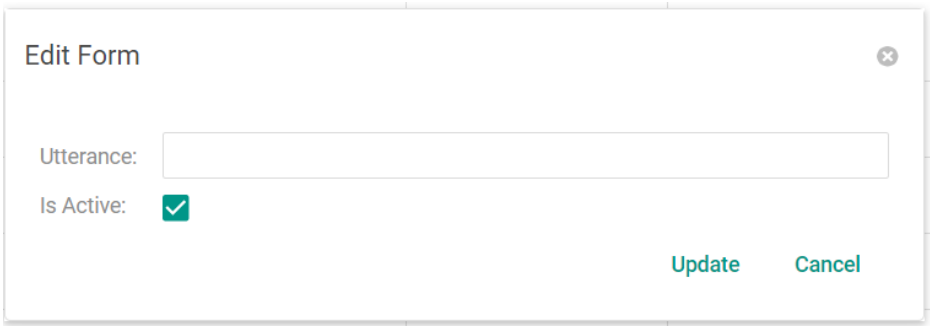


Figure 4-39 Insert, Update Form for Utterance

Response Management

When the user clicks the "Response" tab next to "Utterance" tab, the interface of Response Management as shown in Figure 4-40. There are 4 different tabs which are "Text Response", "Image Response", "Quick Reply Response" and "Carousel Response". The column of "Sequence" is used to determine the order of messages being shown in the Chat Room.

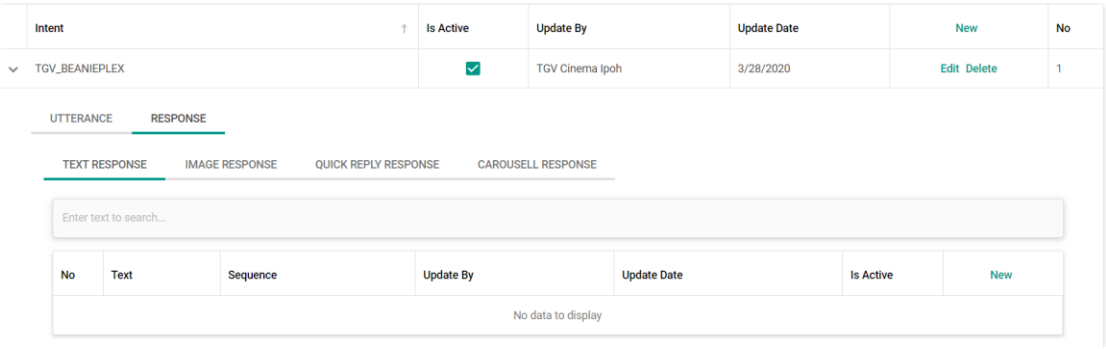


Figure 4-40 Interface for Response Management

4.3.2 Sentiment Management

Figure 4-41 shows the interface of Sentiment Management. The "Test" column is the data for training and the "Sentiment" column is labeled for "Positive" and "Negative". The "Train" button is used to train the sentiment.

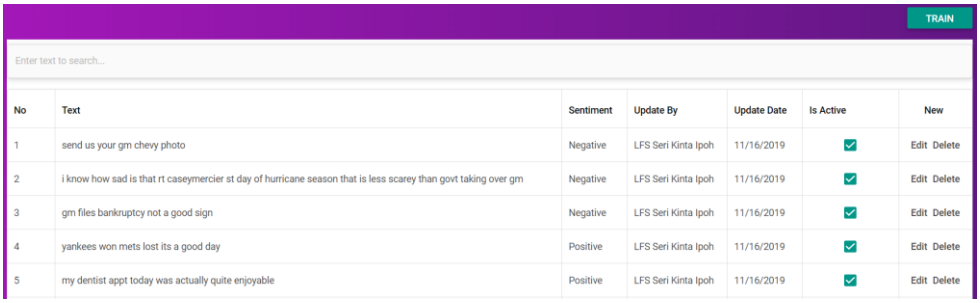


Figure 4-41 Sentiment Management Interface

4.3.3 Hall Management

Figure 4-42 shows the interface of Hall Management. The drop-down list is for selection of cinema hall. As to the "Number of Row" and "Number of Column" is to determine the number of rows and columns of that particular hall. The "Simulate Hall View" button is to generate the hall view (Figure 4-43) and save the row and column to the database.

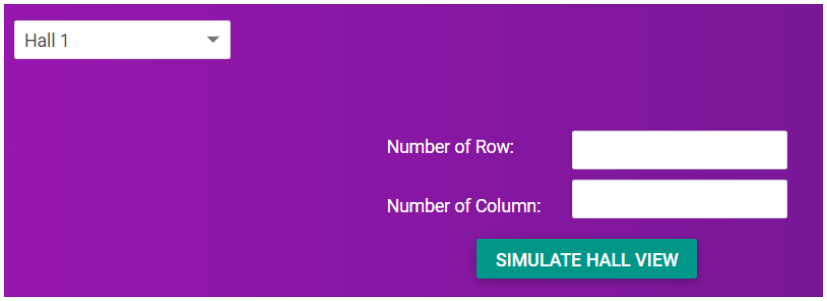


Figure 4-42 Hall Management Interface

Figure 4-43 Hall View Generated

4.3.4 Movie Management

Figure 4-44 shows the interface for Movie Management. The “Movie” column is for movie names, “Duration” is the time for the movie to be finished and “Is Adult Constraint” is used to determine whether the movie is for 18 years old and above. “Photo” is used to display the photo of the movie. Figure 4-45 shows the Photo Uploader which is used to upload the photo and save it in the database. In Figure 4-44, there is a “>” symbol besides the numbering of each row which is used to show the details of the movie (Figure 4-46) such as showtime, price and so on. The “Date”, “Day”, “Time”, “Hall” and “Price” are the details of the movie and these details are used in ticket booking.

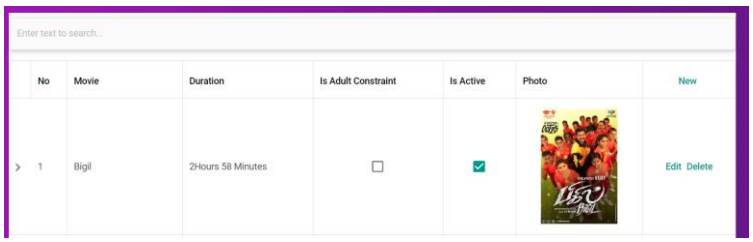


Figure 4-44 Movie Management Interface

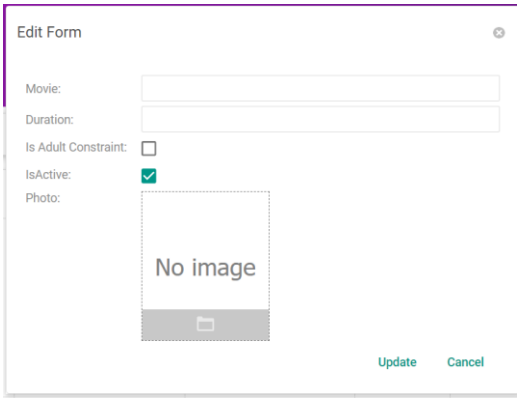


Figure 4-45 Edit Form of Movie Management

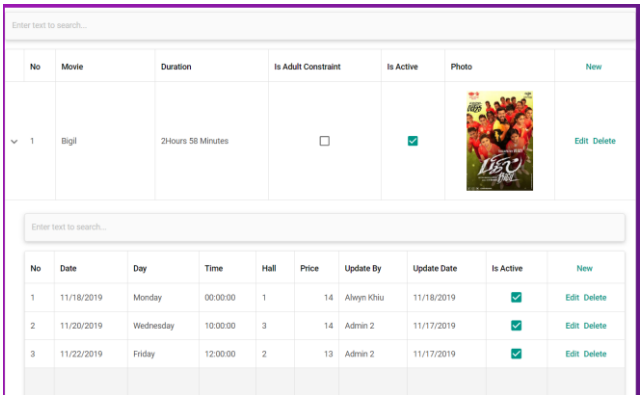


Figure 4-46 Details of Movie Interface

4.3.5 Ticket Management

Figure 4-47 shows the interface of Ticket Management. As mentioned in Section 4.3.4 Movie Management, the details of "Movie", "Duration", "Date", "Day", "Time" and "Price" are the details of the movie. The "Seat" is the location that is purchased by the customer while the "Customer" column displayed the name of Customer who books the ticket.

As to the "Create Ticket" button is for the agent to help the customer to book the ticket at the counter. Therefore, it does not send the email with ticket details with QR code attachment. The ticket booking interface is similar with Figure 4-25 but there is no "Cinema" drop-down list because it is impossible for the agent of cinema to book the tickets from competitors. The "Show/Delete Ticket" button is to route the user interface as in Figure 4-25 back to interface as in Figure 4-47.

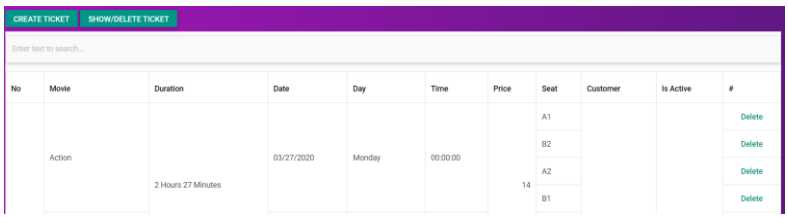


Figure 4-47 Ticket Management Interface

4.4 User Interface and Functionalities of Cinema Admin

Figure 3-2 shows the functionalities that are applied to the Cinema Admin actor in Use Case Diagram. The functionalities of Cinema Admin can refer to Section 4.1.2 Account Management and Section 4.1.3 Chat Room. In addition to that, all of the functionalities in Agent are a subset of the functionalities in Cinema Admin.

Figure 4-48 is the Dashboard interface of the Cinema Admin. The two graphs showed the sale of tickets per day basis and per movie basis. As to the word cloud below the graphs is for decoration and it makes up of all of the Utterances related to that particular cinema.

Figure 4-48 Cinema Admin Dashboard Interface

4.4.1 Agent Management

The main difference of the Cinema Admin and Agent is Agent Management. Since the Agent is not able to create the account by own. Therefore, Cinema Admin takes the responsibility to create the account for the agents. This is the interface of the Agent Management as shown in Figure 4-49 where it is similar to Figure 4-5. However, the Agent Management contains additional functionality such as “Insert”, “Update” and “Delete”. The “New” button presence in every card view is because this eases the Cinema Admin to insert new records directly instead of scroll up and look for the “New” button.

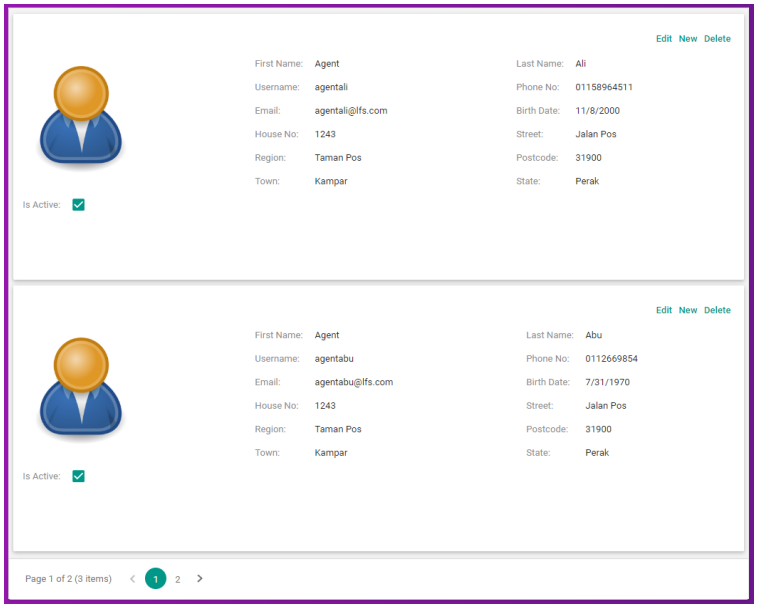


Figure 4-49 Agent Management Interface

4.5 User Interface and Functionalities of Admin

Admin is the person who has full access rights to the system. This means that all of the functionalities mentioned previously are a subset of Admin. Figure 4-50 and Figure 4-51 show the Dashboard of the Admin. In Figure 4-50, the colored card showed the number of users registered in this system while Figure 4-51 showed the record filled by users in Figure 4-2. The "Edit" button is used to update the "Is Seen" column.

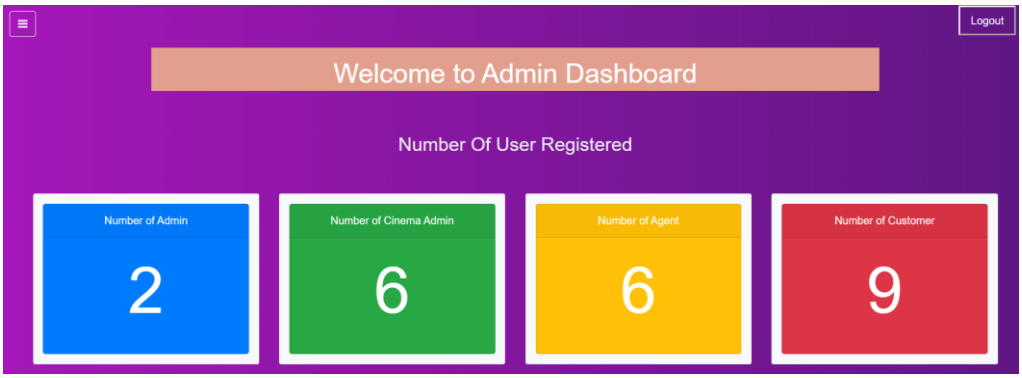


Figure 4-50 Admin Dashboard Interface - Part 1

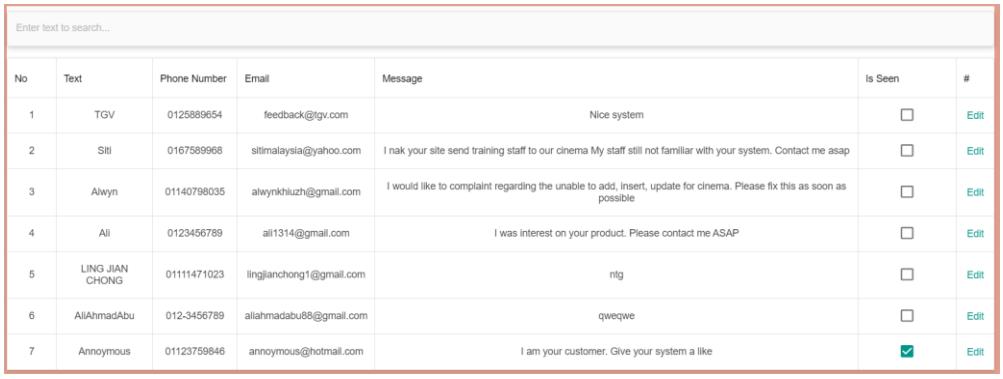


Figure 4-51 Admin Dashboard Interface - Part 2

4.5.1 Cinema Management

Figure 4-52 is the interface for Cinema Management. "Cinema" column displays the name of the cinema, "Address" column displays the address of cinema. However, the "Address" is broken into smaller sections such as "Street", "Region", "Postcode", "Town" and "Region" as portrayed in Figure 4-53. "Hall Number" column is to display the total number of halls in specific cinemas. The rest of the columns are as similar as mentioned in the previous section.

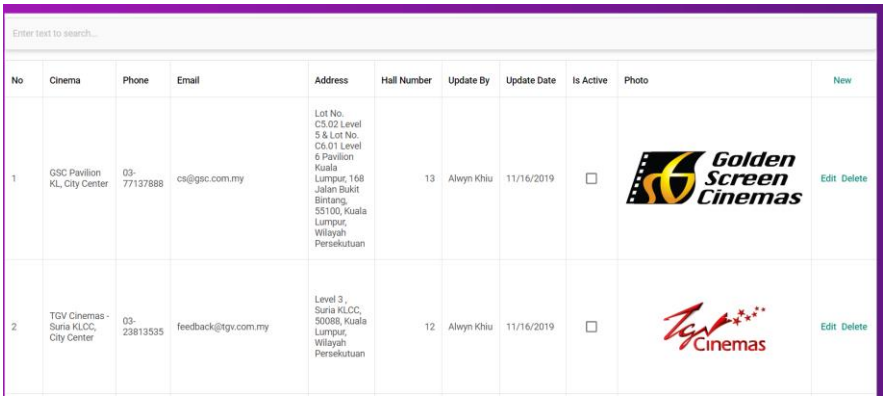


Figure 4-52 Cinema Management Interface

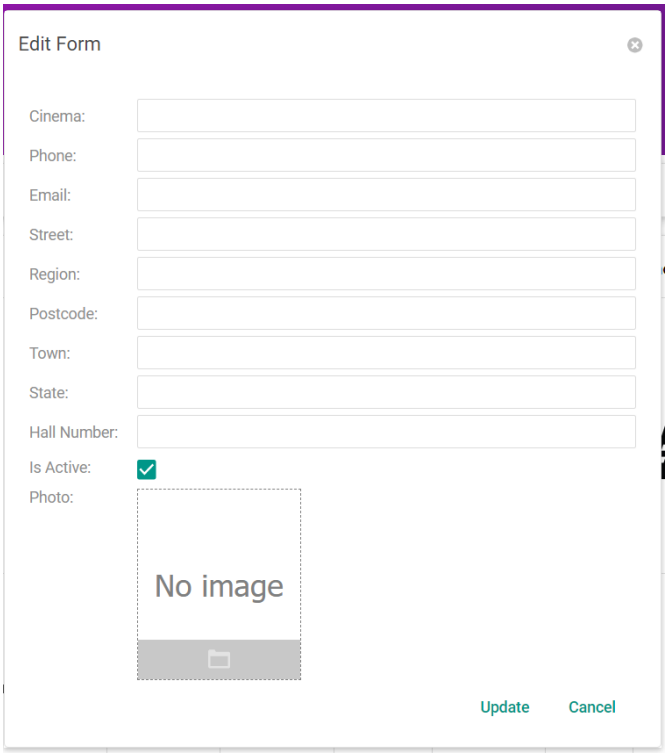


Figure 4-53 Edit Form of Cinema Management

4.5.2 Admin Management, Cinema Admin Management and Customer Management

The interface of Admin, Cinema Admin and Customer Management is similar to Figure 4-49. The difference of these management pages is for managing different roles. There is a minor difference in Cinema Admin Management which is the presence of a drop-down list for the selection of cinema as shown in Figure 4-54. As to the "Insert"

function in the Customer Management is used to help the customer who cannot register using the registration interface as shown in Figure 4-55.

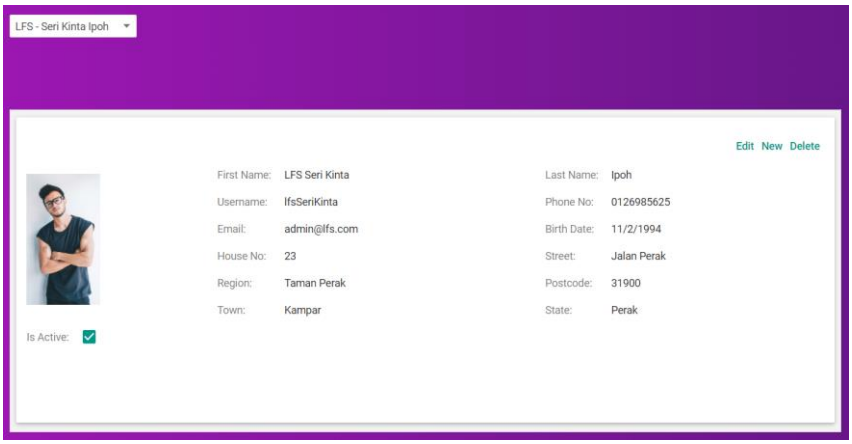


Figure 4-54 Cinema Admin Management Interface

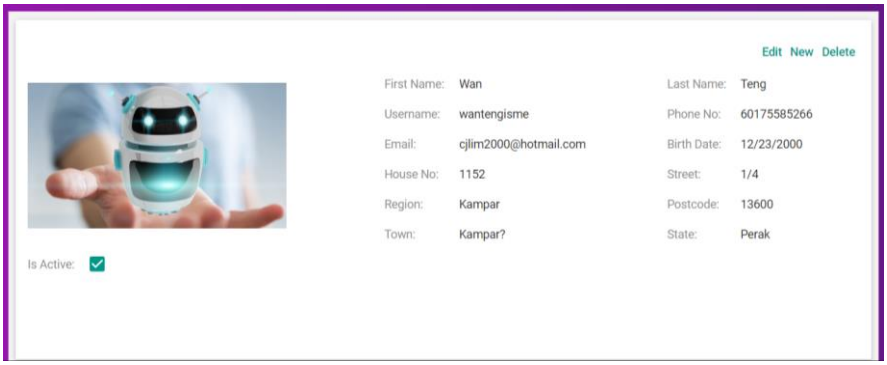


Figure 4-55 Customer Management Interface

4.5.3 Access Control Management

This is the interface for Access Control Management as shown in Figure 4-56. This is mainly for determining the access right of each actor. “Name” column is the text displayed in the “Menu” shown in Figure 4-57. For example, Ticketing Booking. “Group” is the name of the drop-down list such as Chatbot. The “URL” is the field for listing the location of a specific webpage. When the fields of “Is Admin”, “Is Customer”, “Is Agent” and “Is Cinema Admin” are checked, then that particular role has access right on it.

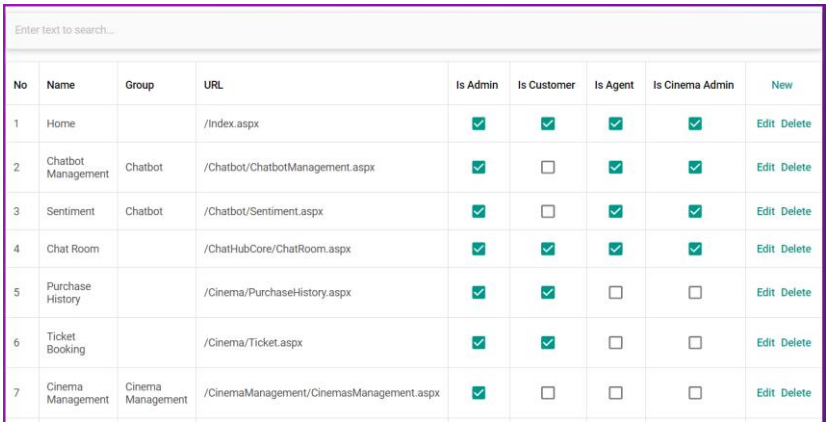


Figure 4-56 Access Control Management Interface

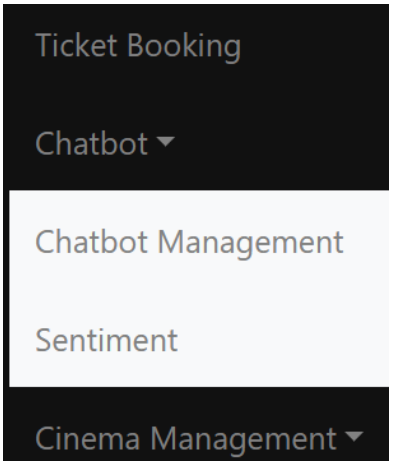


Figure 4-57 Menu

4.6 Hosting with NGROK

The proposed system is then hosted in IIS and then allowed access by public via porting to the NGROK which provided the hosting with random domain names. However, the limitation of the free version of NGROK is unable to host for more than eight hours. There is also limited concurrent access for more than three connections. The configuration of NGROK is typing the following command: “ngrok http <port in IIS>” as shown in Figure 4-58 and the result of configuration is shown in Figure 4-59. After that, the result of hosting is shown in Figure 4-60.

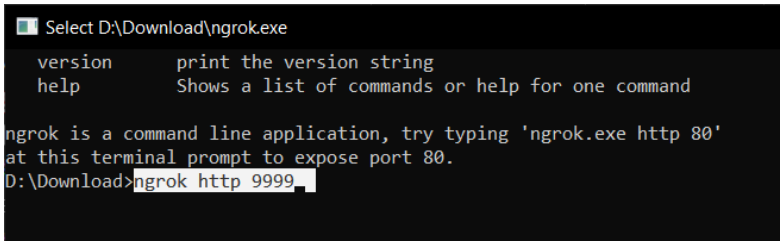


Figure 4-58 Configuration of NGROK

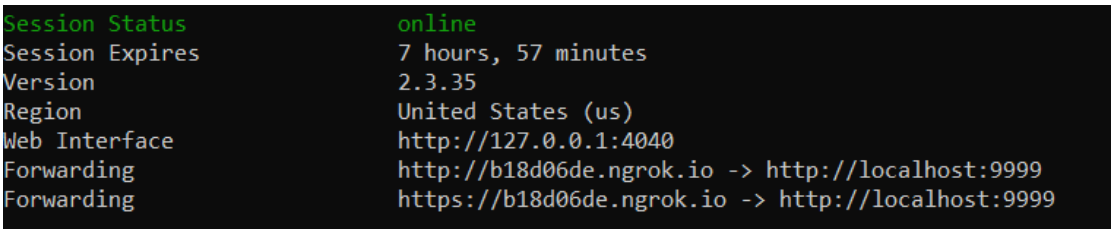


Figure 4-59 Random domain provided by NGROK

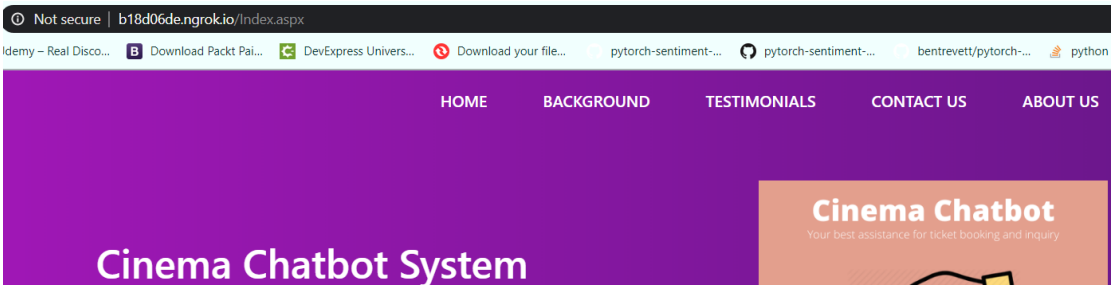


Figure 4-60 Access Proposed System using NGROK domain

4.7 Discussion of Deep Learning Algorithm

In this section, the Deep Learning Algorithms are discussed and compared to determine which algorithm should be used. For the Sentiment and Intent classification, the algorithm involved are the comparison included Long Short Term Memory (LSTM), Recurrent Neural Network (RNN), Bidirectional LSTM (BLSTM), Bidirectional RNN (BRNN) and Convolutional Neural Network (CNN). As to the translation of utterance and response, it only used the algorithm of Sequence to Sequence (Seq2Seq) with Attention and Teacher Forcing.

The comparison of the algorithm is done by shuffling the dataset each time of training and the testing is done for 5 times and then getting the average accuracy. The reason for shuffling the dataset is because to ensure the algorithm is able to perform well even the dataset for training, validation and testing is different. In addition, the average value is obtained to avoid the model getting stuck at local minimum during gradient descent for updating the parameters such as weight and bias. For better

comparison purposes, the hyperparameters such as type of optimizer, number of hidden layers, dropout rate and so forth is fixed.

The scores shown in the table are loss (above) and accuracy (below) in percentage (%) where the interpretation of the best model is having low loss and high accuracy. The loss value is obtained by subtracting the predicted value with the actual value. As for the accuracy value, it represents the number of correct predictions. For the information, the scores are obtained by testing the model with a test set.

Algorithm	Score	Number of Testing					Average
		1	2	3	4	5	
RNN	Loss	0.422	0.358	0.408	0.422	0.410	0.404
	Accuracy	83.33	83.33	86.67	86.67	86.67	85.33
LSTM	Loss	0.317	0.366	0.352	0.367	0.353	0.351
	Accuracy	86.67	86.67	86.67	86.67	86.67	86.67
BRNN	Loss	0.462	0.430	0.403	0.526	0.484	0.461
	Accuracy	80.00	80.00	86.67	80.00	86.67	82.67
BLSTM	Loss	0.357	0.363	0.316	0.349	0.260	0.329
	Accuracy	86.67	86.67	86.67	86.67	86.67	86.67
CNN	Loss	0.409	0.393	0.342	0.337	0.376	0.371
	Accuracy	86.67	83.33	86.67	86.67	86.67	86.00

Table 4-1 Result of Sentiment Analysis with different models

Algorithm	Score	Number of Testing					Average
		1	2	3	4	5	
RNN	Loss	0.045	0.042	0.053	0.034	0.039	0.043
	Accuracy	88.89	88.89	81.48	88.89	88.89	87.41
LSTM	Loss	0.019	0.032	0.027	0.030	0.026	0.027
	Accuracy	95.27	88.89	88.89	85.19	92.59	90.17
BRNN	Loss	0.034	0.034	0.024	0.040	0.042	0.035
	Accuracy	88.89	88.89	92.59	85.19	88.89	88.89
BLSTM	Loss	0.044	0.049	0.044	0.051	0.051	0.048
	Accuracy	88.89	88.89	92.59	88.89	92.59	90.37
CNN	Loss	0.009	0.011	0.011	0.012	0.010	0.011
	Accuracy	95.27	96.26	96.88	95.27	96.26	95.19

Table 4-2 Result of Intent Classification with different models

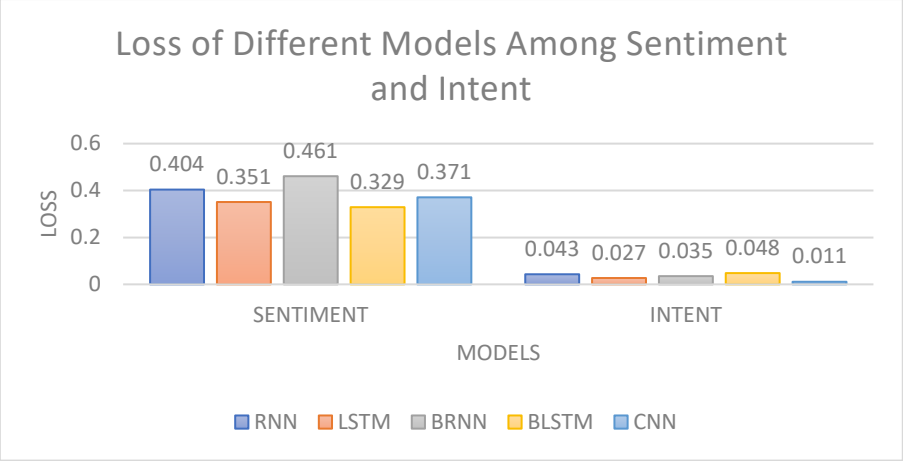


Figure 4-61 Bar Graph of Loss with Different Models

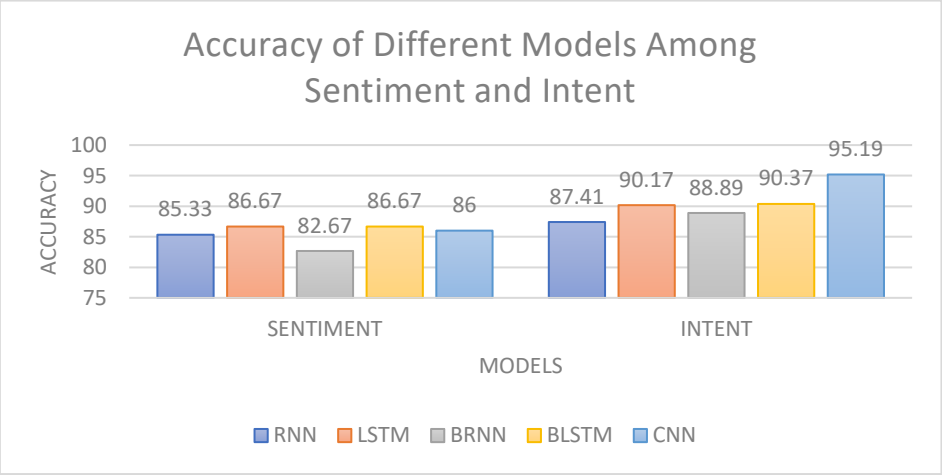


Figure 4-62 Bar Graph of Accuracy with Different Models

Sentiment

As shown in Table 4-1, the sentiment testing of model, BLSTM (0.329), LSTM (0.351) and CNN (0.371) are better than RNN (0.404) and BRNN (0.461) where the loss of BLSTM, LSTM, CNN is almost similar. The reason RNN and BRNN are unable to perform well is because some of the length of the sentiment training data is long which is up to 40 words. This is due to RNN and BRNN contain the problem of vanishing gradient descent which means the gradient descent value becomes smaller and smaller when the sequence data traversing toward the end (Hochreiter, 1998). Vanishing gradient descent problem caused the parameters such as weight and bias do not have much updated when undergoing product mathematical expression. In other words, the RNN and BRNN is unable to remember too many sequential words that had

been processed. Even though BRNN can traverse from backwards but the vanishing gradient descent issue caused the value of gradient descent to become small at the front of sequential data. Therefore, it causes the BRNN to have a high loss score.

Then, CNN is able to capture the important information in the input data which is in embedding format using the convolution layer (Cha & Lee, 2018). For example, CNN able to capture the important information in the sentence of "I hate the chatbot because unable to solve my problems" where the "hate" keyword information is able to captured by CNN. However, according to (Yang, et al., 2019), there is a lack of robust and reliable CNN model that is performs well in time series classification due to the convolution layer being unable to remember the state of previous sequential data. For instance, CNN cannot remember the "not" word and it is only able to capture the "poor" word and then detected as "Negative" in the sentence of "The service of the cinema is not poor". Therefore, the performance is slightly weaker than LSTM and BLSTM.

For the LSTM and BLSTM, both of them are able to keep or forget the sequential data and remember the information from a long chain sequence (Nguyen, 2018). In this case, the BLSTM is having the lowest loss score as compared to LSTM because it is able to traverse forward and backward which means BLSTM is able to obtain the information from both directions (Aggarwal, 2019). This causes the BLSTM to know how the structure of the sentence that categorized as "negative" and "positive".

As to the accuracy of sentiment, BLSTM and LSTM have the similar score which is 86.7% followed by CNN (86%) and RNN (85.33%) and lastly BRNN (82.67%). The result of accuracy and loss of sentiment is matched where the loss is low and the respective accuracy will be low as well.

Intent

The loss of the intent is much lower than sentiment because it is domain specific. Therefore, the model is able to predict it well as compared to sentiment which is open domain. In other words, the user input for intent is much related to the cinema such as outside food and video recording of movies. This makes the deep learning algorithm able to learn it easier. Nonetheless, the sentiment is open domain which consists of different conditions such as customer support, talking on rude words, scolding people and so on. In other words, the model is able to learn well when the domain is specific.

The CNN in intent prediction is able to achieve the low loss (0.011) and high accuracy (95.19%). This is because as mentioned previously CNN is able to capture the

keyword from the utterance. For instance, when a user is asking whether outside food is allowed to enter the cinema hall, the model is able to capture "outside food", "allow" and "enter cinema". Hence, the model managed to predict it when the new user input contains those keywords. In addition to that, the utterance inputted by the user is almost the same and the differences are only a few words or arrangement of sentence structures.

Following that, LSTM and BRNN performance is quite similar to each other. This is because the sentence length of the training data of intent classification is short. Therefore, the weakness of BRNN as mentioned previously which is vanishing gradient descent issue can be avoided. Furthermore, the BRNN is able to traverse forward and backward further solving the problem. In other words, the BRNN is able to hold the information from forward propagation and backward propagation in which during combining of two information BRNN can have information from past and future (Aggarwal, 2019).

As to the LSTM, it can remember the sequential data that have been seen previously. Therefore, the LSTM can perform quite well since it able to remember the words after another. Nonetheless, if the structure of the sentences is change from "Can you please let me know more about what is Aurum Theatre?" to "Tell me more about Aurum Theatre is what, can you?", the prediction is having high chances to get wrong. In addition, LSTM also will raise problem when two sentences having the same structure but asking the different intent. For example, "Can you please let me know more about what is Aurum Threatre?" and "Can you please let me know about where is Aurum Threatre?". In this case, the "what" is changed to "where" but the structure of the sentences are the same will also lead to miss prediction.

Besides, BLSTM will cause the model to be overfitting because traversing forward allows the model to remember the information from forward propagation and backward propagation. This led to the model trained memorizing the training data and unable to predict when user input is not recognized. Apart from that, the performance of RNN is weaker than BRNN because of vanishing gradient descent issues and it is not bidirectional. In other words, the RNN is unable to obtain the information from both directions.

Translation

Translation of Bahasa Malaysia to English and vice versa is done by the same deep learning algorithm which is Sequence to Sequence (Seq2Seq) with teacher forcing

and attention. The time series prediction algorithm used is Gated Recurrent Units (GRU) as proposed by (Huang, et al., 2020). First of all, the Seq2Seq model consists of Encoder and Decoder. The Encoder is used to encode the inputted language (source language) in order to obtain the output of Encoder which is hidden state value. After that, the output of Encoder will then be passed to the Decoder as input and further training to get the output language (target language). In other words, the Encoder is used to understanding the meaning of sentence of source language and then decoder is using the understandable meaning to build the target language. Furthermore, the GRU is selected instead of LSTM because GRU has less parameters for training which led to fast training and less memory consumption (Gao & Glowacka, 2016).

However, the usage of Encoder and Decoder for translation is weak and having a high loss score due to the input of Decoder is totally retrieved from Encoder. Therefore, the teacher forcing is added to solve the problems mentioned. Teacher forcing is a method to supervise the input of Decoder in which the actual target is compared with the output of Encoder based on time series. In case the output of Decoder is less than the threshold of teacher forcing, the actual target of that particular time frame will be used instead of the output of Encoder at the same time frame (Brownlee, 2019). This helps to reduce the mistakes done in Encoder and continuously misleading the Decoder. In short, the teacher forcing plays an important role to ensure the mistakes done for input of Decoder as less as possible.

Furthermore, the teacher forcing is insufficient to produce a good translation model. Therefore, the Attention is added in between Encoder and Decoder. Attention is used to focus the important words during the training. For example, when translating “Apa yang khas pada Rabu” to “What is special on Wednesday”, the Attention will remember “Apa” equivalent to “What”, “khas” is “special” and “Rabu” is translated as “Wednesday”. By remembering the translation of word by word using Attention will help the model able to predict the targeted language. But, if translating word by word will cause grammar error, therefore the hidden state value of Encoder is important to ensure the grammar is correct when prediction of target language.

In a nutshell, the BLSTM is used for predicting the sentiment while CNN model will be used to classify the intent of user input. In the proposed system, when the confidence level of prediction of sentiment and intent is lesser than 80% it will be categorized as unknown and respective action will be taken. As to the translation, it is predicted by using the Seq2Seq model with teacher forcing and Attention.

CHAPTER 5 CONCLUSION

In conclusion, the proposed system is able to achieve the objectives identified which are generate a knowledge-based and develop a chatbot with standardized user interface as well as organized database for assisting the customer. The contributions mentioned also bring a lot of convenience not only to customers but also cinema staff. The proposed system had improved the weaknesses of the existing system and remained the strengths in the proposed system. For the future work, the insert, update and delete of the chatbot management should be in drag and drop format where allow better visualization to the cinema admin and agent during designing the conversation flows. In addition to that, the system should be developed in mobile applications so that it does not require the browser for surfing to eliminate problems of dependencies such as WebKit. Lastly, the ticket booking should be implemented with deep learning to extract the entity from user utterances and response of inquiry required to use the generative model. The complaint section also should be included in the system so that the chatbot is not only working for ticket booking and inquiry.

REFERENCES

Lotus Fivestar Cinemas (M) Sdn. Bhd. , 2019. *LFS Cinema Home Page.* [Online]
Available at: https://www.lfs.com.my/
[Accessed 12 November 2019].

Aggarwal, R., 2019. *Bi-LSTM.* [Online]
Available at: https://medium.com/@raghavaggarwal0089/bi-lstm-bc3d68da8bd0
[Accessed 31 March 2020].

Brownlee, J., 2019. *What is Teacher Forcing for Recurrent Neural Networks?.* [Online]
Available at: https://machinelearningmastery.com/teacher-forcing-for-recurrent-neural-networks/
[Accessed 31 March 2020].

Cha, J. & Lee, J.-H., 2018. Extracting Topic Related Keywords by Backtracking CNN based Text Classifier. *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS),* pp. 93-96.

CinemaOnline, 2019. *Cinemas.* [Online]
Available at: http://www.cinema.com.my/cinemas/cinemalist.aspx
[Accessed 30 July 2019].

Dialogflow, 2019. *Dialogflow.* [Online]
Available at: https://dialogflow.com
[Accessed 30 July 2019].

Gao, Y. & Glowacka, D., 2016. Deep Gate Recurrent Neural Network.

Guru99, 2019. *Oracle Vs. SQL Server: Key Differences.* [Online]
Available at: https://www.guru99.com/oracle-vs-sql-server.html
[Accessed 29 July 2019].

Guru99, 2019. *What is WHITE Box Testing? Techniques, Example, Types & Tools.* [Online]
Available at: https://www.guru99.com/white-box-testing.html
[Accessed 27 July 2019].

Handoyo, E. et al., 2018. Ticketing Chatbot Service using Serverless NLP. *2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE),* pp. 325-330.

Hochreiter, S., 1998. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems,* Issue 6(02), pp. 107-116.

Huang, W., Dong, X., Shang, W. & Lin, W., 2020. Research on Man-Machine Conversation System Based on GRU seq2seq Model. *2019 6th International Conference on Dependable Systems and Their Applications (DSA), Harbin, China,* pp. 413-418.

Kaggle, 2019. *Kaggle Dataset Twitter Sentiment Analysis.* [Online] Available at: https://www.kaggle.com/c/twitter-sentiment-analysis2/overview [Accessed 29 October 2019].

Kazi, S., Bagasrawala, M., Shaikh, F. & Sayyed, A., 2018. Smart E-Ticketing System for Public Transport Bus. *2018 International Conference on Smart City and Emerging Technology (ICSCET).*

Li, X. et al., 2016. A User Simulator for Task-Completion Dialogues. *arXiv preprint.*

Li, X. et al., 2018. Microsoft Dialogue Challenge: Building End-to-End Task-Completion Dialogue Systems. *SLT 2018 Special Session.*

Mkrtchyan, R., 2017. *All You Need to Know About the Waterfall Model.* [Online] Available at: https://www.linkedin.com/pulse/all-you-need-know-waterfall-model-rafayel-mkrtchyan [Accessed 26 July 2019].

M. T., 2019. *Microtel Technology - Singapore Facebook Page.* [Online] Available at: https://www.facebook.com/microteltechnologySG [Accessed 30 July 2019].

Nguyen, M., 2018. *Illustrated Guide to LSTM's and GRU's: A step by step explanation.* [Online] Available at: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21 [Accessed 31 March 2020].

Peters, F., 2018. *University of Liège Library.* [Online] Available at: https://matheo.uliege.be/bitstream/2268.2/4625/6/Thesis_PETERS_Florian.pd

f

[Accessed 12 November 2019].

Prabowo, Y. D. et al., 2018. Lstm And Simple Rnn Comparison In The Problem Of Sequence To Sequence On Conversation Data Using Bahasa Indonesia. *The 1st 2018 INAPR International Conference, 7 Sept 2018, Jakarta, Indonesia,* pp. 51-56.

ProfessionalQA.com, 2019. *Bottom Up Integration Testing Approach.* [Online] Available at: http://www.professionalqa.com/bottom-up-approach [Accessed 28 July 2019].

Rosruen, N. & Samanchuen, T., 2018. Chatbot Utilization for Medical Consultant System. *2018 3rd Technology Innovation Management and Engineering Science International Conference (TIMES-iCON).*

Singh, R. et al., 2018. *Chatbot using TensorFlow for small Businesses.* Mumbai, IEEE, pp. 1614-1619.

Software Testing Fundamentals, 2019. *Black Box Testing.* [Online] Available at: http://softwaretestingfundamentals.com/black-box-testing/ [Accessed 28 July 2019].

SOFTWARETESTINGHELP, 2019. *What Is SDLC Waterfall Model?.* [Online] Available at: https://www.softwaretestinghelp.com/what-is-sdlc-waterfall-model/ [Accessed 28 July 2019].

Sommerville, I., 2011. *Software Engineering.* 9th ed. Boston, Massachusetts: Pearson Education, Inc..

T. C. S. B., 2019. *FAQs.* [Online] Available at: http://www.cinema.com.my/cinemas/cinemalist.aspx [Accessed 30 July 2019].

TGV Cinema Sdn. Bhd., 2019. *TGV Home Page.* [Online] Available at: https://www.tgv.com.my/ [Accessed 12 November 2019].

T. N. B., 2019. *TENAGA NASIONAL BERHAD.* [Online] Available at: https://www.tnb.com.my [Accessed 30 July 2019].

Watson, 2019. *Watson Assistant.* [Online]
Available at: https://watson-assistant-demo.ng.bluemix.net
[Accessed 30 July 2019].

Yang, C., Jiang, W. & Guo, Z., 2019. Time Series Data Classification Based on Dual Path CNN-RNN Cascade Network. *IEEE Access,* Volume 7, pp. 155304-155312.

Zendesk, 2019. *Zendesk.* [Online]
Available at: https://www.zendesk.com/chat/chatbot/
[Accessed 12 November 2019].

APPENDIX A FINAL YEAR PROJECT WEEKLY REPORT

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 1**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

I had redesigned the interface of the proposed system for better view in the Index.aspx and ChatRoom.aspx. In addition, the ChatbotManagement.aspx also used to integrate the Intent.aspx, Utterance.aspx and Response.aspx.

2. WORK TO BE DONE

- Review the deep learning model for chatbot by studying the architecture, strengths and weakness
- Study the Pytorch deep learning library.

3. PROBLEMS ENCOUNTERED

- CSS hard to control since lack of experiences.

4. SELF EVALUATION OF THE PROGRESS

- Satisfy.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 2**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

Finish study the architecture of deep learning and know the respective strengths and weaknesses. I also able to code and understand the coding based on the example of study materials. The architectures are more understandable. Then I start to code for the sentiment classifier.

2. WORK TO BE DONE

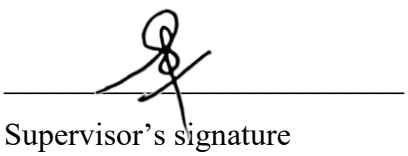
- Study and research the coding of Pytorch
- Start to code for the classification of sentiment and intent using Pytorch

3. PROBLEMS ENCOUNTERED

- Study materials do not explain the architecture and the usage clearly
- The syntax is not clear and still learning.
- The dimensions of the input and output difficult to calculate.

4. SELF EVALUATION OF THE PROGRESS

Satisfy.



Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 4**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

The deep learning architectures are produced for sentiment classifier. After that, start the coding for the intent classifier. However, there is less problems encounter in building the intent classifier based on the experience of failure previously.

2. WORK TO BE DONE

- Research the way of building the Seq2Seq model for translation
- Start to build the Seq2Seq model

3. PROBLEMS ENCOUNTERED

- Google Colab keeps freeze due to high usage

4. SELF EVALUATION OF THE PROGRESS

Satisfy.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 6**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

Successfully build the model of Seq2Seq model. The dataset of the utterances and responses is translated to Bahasa Malaysia manually since the Google Translate does not work well for the translation. However, the specific terms are remained in English. Then the model is improved using Teacher Forcing + Attention and the prediction become more make sense.

2. WORK TO BE DONE

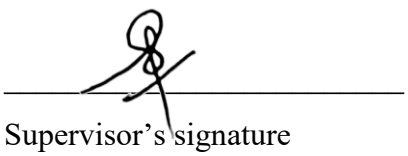
- Tidy up the code and standardize the train model and predict model.

3. PROBLEMS ENCOUNTERED

- Seq2Seq performance performed quite well but the ending of translation is not perfect.

4. SELF EVALUATION OF THE PROGRESS

Satisfy.



Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 8**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

The code is tidy up and started to integrate the code with Flask for providing the services to the ASP.NET. Finish the design of Text, Image, Quick Reply and Carousel Response. In addition, the ticket booking and inquiry bot as well as agent handle manage to be completed.

2. WORK TO BE DONE

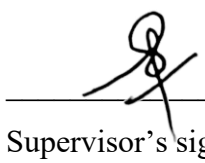
- Ensure the system work well and start the report writing.

3. PROBLEMS ENCOUNTERED

- None

4. SELF EVALUATION OF THE PROGRESS

Satisfy.



Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 10**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE

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

Finish the design of Text, Image, Quick Reply and Carousel Response. In addition, the ticket booking and inquiry bot as well as agent handle manage to be completed.

2. WORK TO BE DONE

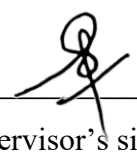
- Ensure the system work well and start the report writing.

3. PROBLEMS ENCOUNTERED

- None

4. SELF EVALUATION OF THE PROGRESS

Satisfy.



Supervisor's signature

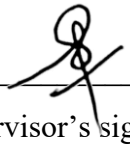
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: 3, 3	**Study week no.: 12**
Student Name & ID: Khiu Zhen Hung 16ACB01673	
Supervisor: Dr. Lim Seng Poh	
Project Title: Chatbot Assisted Inquiry and Ticket Booking System For Cinema	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Test the proposed system and started report writing. After that, let supervisor to check the report.
2. WORK TO BE DONE - Report writing - Checking of report by supervisor and present the system
3. PROBLEMS ENCOUNTERED - None
4. SELF EVALUATION OF THE PROGRESS Satisfy.



Supervisor's signature

Student's signature

POSTER

INTRODUCTION

- Auto-response system for Cinema
- Handles inquiry and ticket booking
- Reduce Agent workload
- Avoid waiting for response and queue
- Provide standardized response
- Provide translation of Bahasa Malaysia to English and vice versa

PROBLEM STATEMENT

Inconsistency answer from agent

Time Consuming for looking required answer

Heavy Workload of agent

Complex interface of different cinema

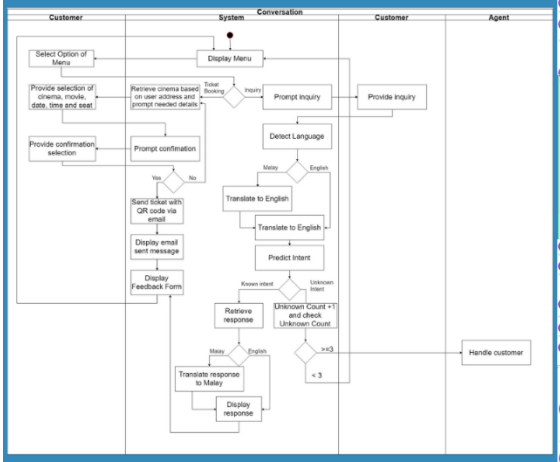
CHATBOT ASSISTED INQUIRY AND TICKET BOOKING SYSTEM FOR CINEMA

Done By: Khiu Zhen Hung

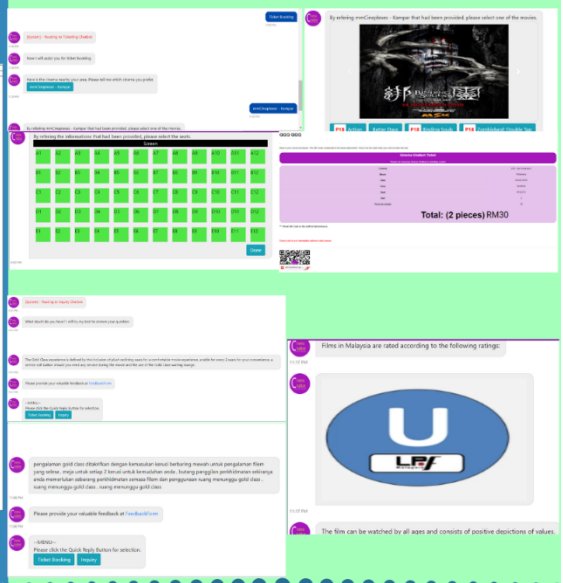
Supervised By: Dr. Lim Seng Poh

METHODOLOGY

- Waterfall Model
- Flowchart as shown below:



SCREENSHOT



OBJECTIVES

- To develop a chatbot for clarify customer doubt and ticketing system
- To generate a knowledge base for standardization of the inquiry response
- To design a standardized user interface and organized database for proposed system.

PROJECT SCOPE

1. Only focus on cinema in Malaysia
2. Only generate knowledge base with limited inquiry related to cinema and movie
3. Only generate and send the ticket's copy to the email of customer
4. Only generate QR code for ticket
5. Will not focus on payment of ticket
6. Does not involve QR Code Scanner
7. Only focus on Translation on Inquiry

DISCUSSION

- Sentiment Classifier using BLSTM (87%)
- Intent Classifier using CNN (95%)
- Translation using GRU Seq2Seq with teacher forcing and Attention

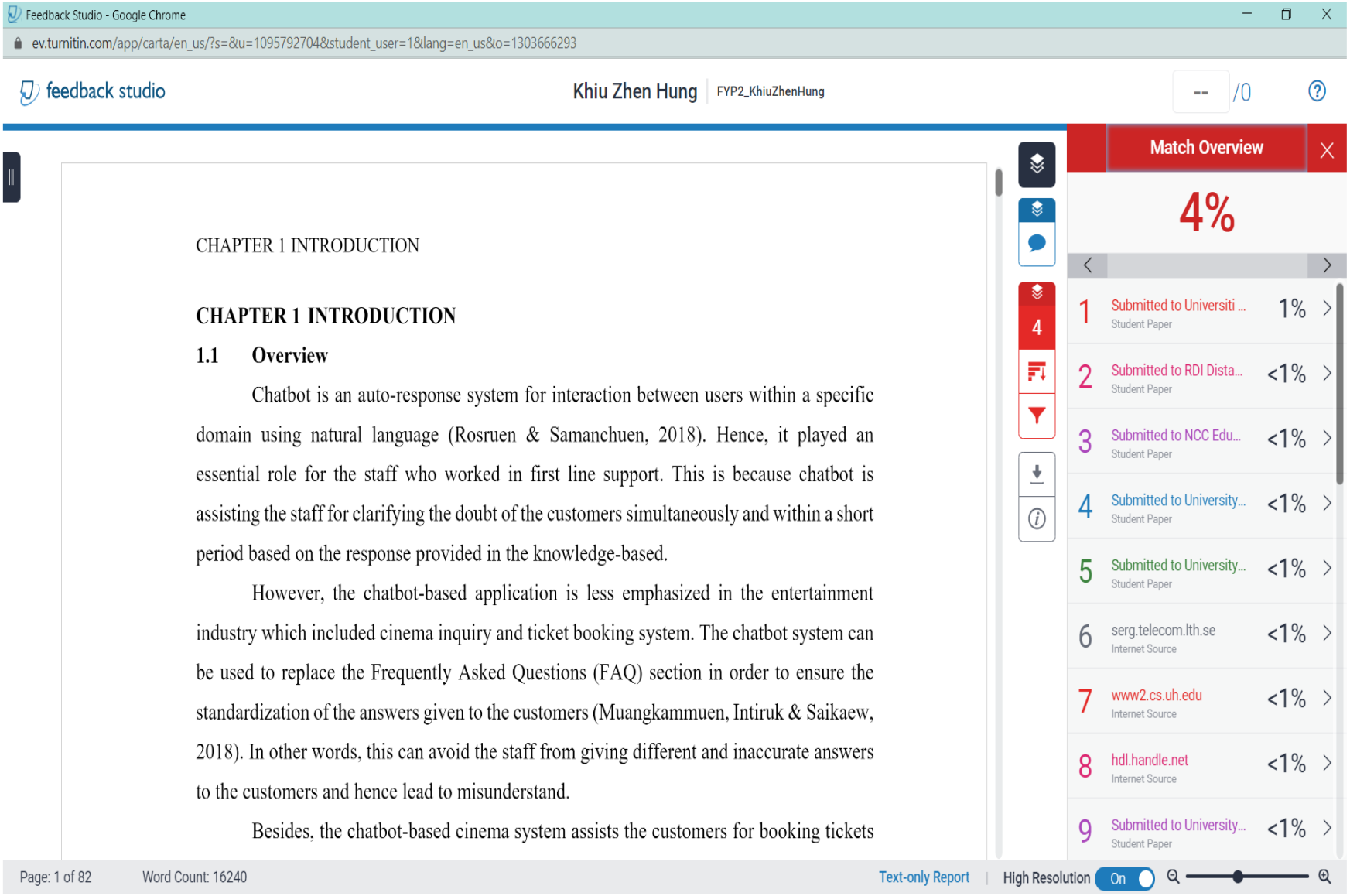
CONCLUSION

Future Work

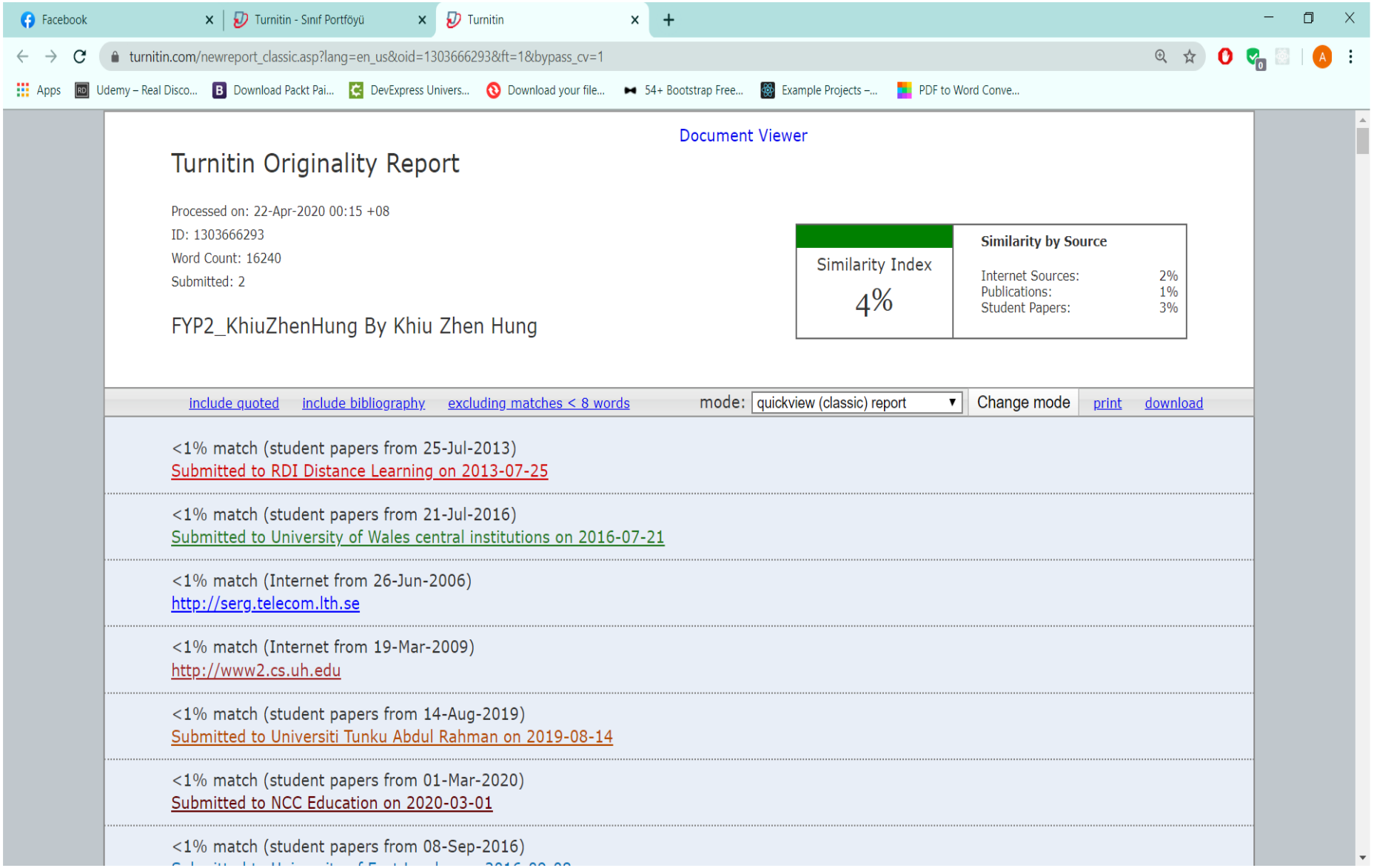
- Drag and Drop for designing the conversation flow
- Supported in web and mobile phone
- Named Entity Recognition should be implemented for capturing entity of ticket booking
- Response of chatbot should be generated using generative model
- Included complaint section in the chatbot

BCS(Hons) Computer Science
Faculty of Information and Communication Technology (Kampar Campus), UTAR

PLAGIARISM CHECK RESULT



BCS(Hons) Computer Science
Faculty of Information and Communication Technology (Kampar Campus), UTAR



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)	KHIU ZHEN HUNG
ID Number(s)	16ACB01673
Programme / Course	Bachelor of Computer Science (HONS)
Title of Final Year Project	Chatbot Assisted Inquiry and Ticket Booking System for Cinema

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 4 % **Similarity by source** Internet Sources: 2% Publications: 1% Student Papers: 3%	Ok.
Number of individual sources listed of more than 3% similarity: 0	

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: Ts. Dr. Lim Seng Poh

Date: 22/4/2020

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	16ACB01673
Student Name	KHIU ZHEN HUNG
Supervisor Name	DR. LIM SENG POH

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)

I, the author, have checked and confirmed all the items listed in the table are included in my report. _____ (Signature of Student) Date: 22/4/2020	Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction. _____ (Signature of Supervisor) Date: 22/4/2020