**<Hal 2001>**

**Special Course in Software Engineering Project Report**

***Version <1.0>***

**Author 1, Author 2, … Author N**

**(Group #)**

|  |  |  |  |
| --- | --- | --- | --- |
| Version history | | | |
| Version no. | Date | Description | Author |
| 1.1.1 | 9.5.2022 | Complete of Requirements Specifications and Analysis | Haomin Yu, Junyu Zhao |
| 1.1.2 | 9.9.2022 | Edit Design | Peiyuan Zu |
| 1.1.3 | 10.6.2022 | Edited 5.2 Integration part and added the 5.1.2 trying RedisTracker part. | Xinyuan Chen |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[The list of tables 5](#_Toc116035741)

[The list of figures 6](#_Toc116035742)

[1. Introduction 7](#_Toc116035743)

[1.1 Purpose 7](#_Toc116035744)

[1.2 Overview 7](#_Toc116035745)

[1.3 Definition, Acronyms, and Abbreviations(FOR LATER) 7](#_Toc116035746)

[2. Background (FOR LATER) 9](#_Toc116035747)

[2.1 Chatbot Background 9](#_Toc116035748)

[2.2 Salient Concepts 9](#_Toc116035749)

[2.3 Report summary 9](#_Toc116035750)

[3. Requirements Specifications and Analysis 10](#_Toc116035751)

[3.1 Problem Statement 10](#_Toc116035752)

[3.2 Proposed Solution 10](#_Toc116035753)

[3.3 Information gathering \*\*（没填好） 10](#_Toc116035754)

[3.4 Functional Requirements 11](#_Toc116035755)

[3.5 Non-functional Requirements 12](#_Toc116035756)

[3.6 User classes and characteristics 12](#_Toc116035757)

[3.7 Constraints 13](#_Toc116035758)

[3.8 Assumption and Dependencies 14](#_Toc116035759)

[3.9 Example use cases (要改) 15](#_Toc116035760)

[4. Chatbot Design 16](#_Toc116035761)

[4.1 System architecture 16](#_Toc116035762)

[4.2 Dialog Design 16](#_Toc116035763)

[4.3 Data Design 18](#_Toc116035764)

[4.4 System Operation 19](#_Toc116035765)

[5. Chatbot Development 20](#_Toc116035766)

[5.1 Development Environment 20](#_Toc116035767)

[5.2 Integration 22](#_Toc116035768)

[5.2.1 Tracker Order 22](#_Toc116035769)

[5.2.2 RedisTrackerStroe 22](#_Toc116035770)

[6. Testing 25](#_Toc116035771)

[6.1 Internal Testing 25](#_Toc116035772)

[6.2 User Testing 25](#_Toc116035773)

[7. Conclusion 26](#_Toc116035774)

[References 27](#_Toc116035775)

[References 28](#_Toc116035776)

[Appendix A – List of User Stories 30](#_Toc116035777)

[Appendix B – List of functional and non-functional requirements 32](#_Toc116035778)

[Appendix C – Dialogs 33](#_Toc116035779)

[Appendix D – Git Repository 36](#_Toc116035780)

[Appendix E – Seven-category assessment (Internal Testing) 37](#_Toc116035781)

[Appendix F – Seven-category assessment (User Testing) 38](#_Toc116035782)

# The list of tables

**No table of figures entries found.**

# The list of figures

**No table of figures entries found.**

# Introduction

## Purpose

The HAL Chatbot project demonstrates the entire process in a structure based on analysing, requirements extracting, designing, developing, and testing. This report considers the ones that are fans of the movie 2001: A Space Odyssey as the target audiences, from which they can fully understand the crucial points of what the chatbot is capable of and how does it doing it.

## Overview

The customer experience in the media and entertainment industry can be measured in two main ways called primary services and additional services.

The primary service is the event, show or display itself that the customer comes to.

Additional services support the primary service, such as the pre and post ticketing experience, readily available information, staff assistance, etc. HubSpot Research reports that 90% of customers consider immediate response time to be very or extremely important in customer service. For sales and marketing inquiries, that percentage drops slightly - to 82 percent. According to Zendesk's Customer Experience Trends report, about half of customers consider 24/7 support to be an important part of good customer service.

And with conversational AI chatbot technology, you can get.

1. Robotic process automation for common tasks

2. 24/7 customer service automation

3. reduced customer service queue times

4. secure self-service options for customers

HAL 2001 is a fan site developed and maintained by a group of fans of the movie 2001: A Space Odyssey. The Hal chatbot should engage users on trivial topics and provide responses that HAL from the movie/novel would have provided. These responses are likely to be very sharp, straightforward, smart, but a little dark to a regular human being. In addition, the bot should also help address customer grievances related to the merchandise they purchase from the website.

The industrial domain for Hal is entertainment, and in generalchatbots are are designed and developed for a specific domain. These domains are narrow and applicable to to the concern of the organization they serve. Hence chatbots are custom and purpose built as an extension of the organization's operation, usually to allow customers to self-service.

## Definition, Acronyms, and Abbreviations(FOR LATER)

*Provide definitions for all terms, acronyms, and abbreviations required to properly interpret how they are used in this document.*

* AI （Artificial Intelligence）：
* Slot
* Intent
* Entity
* 。。。

**\*\* Everyone or anyone from the group can take responsibility of this section. Ideally, everyone should contribute, at least in proof reading.**

# Background (FOR LATER)

*This section covers the background information necessary for a reader to understand the report to follow. Essentially, this section covers why your chatbot should matter to the people of the industry that it is relevant to, what benefits can they get from the said chatbot, and the concepts that underlie the implementation of the chatbot, which can also help a layman follow this document without much difficulty. All the necessary background information should be conveyed in no more than* ***two to three pages****.*

## Chatbot Background

*Discuss the domain to which your chatbot falls in, to motivate the rationale and necessity of your chatbot . Use scientific literature to justify the existence of your own chatbot, especially in the context of the industry domain you are targeting*

## Salient Concepts

*Next, describe* *the following salient concepts that play a major role in the functioning of your chatbot. Besides the concepts listed below, you are encouraged to include other concepts that are relevant to your chatbot:*

* *Natural Language Processing (NLP)*
* *Natural Language Understanding (NLU)*
* *Artificial Intelligence (AI)*
* *Deep Learning*
* *IBM Watson Assistant / RASA Framework*

## Report summary

*Summarize the different sections to follow, by describing what the readers can expect from each section going forward*

# Requirements Specifications and Analysis

## Problem Statement

1. What can be done to resolve the issue when a customer becomes upset and agitated?

2. Is there a way or means for the customer to connect to a human service?

3. Is there a certain automatic help function when the customer makes an inquiry?

4. When the customer wants to make a complaint, how to resolve it?

5. if the customer feedback is relatively simple similar to the problem of querying the courier number, how to give a quick answer?

6. if the customer encounters a problem is more difficult, how to further solve the problem?

7. how to solve the customer consultation time if it is not within the time of manual customer service?

8. Can all the questions asked by customers be reasonably solved?

## Proposed Solution

1. Chatbots generally output relatively interesting information or use a HAL-like response mechanism, similar to introducing information about websites and topics such as weather, jokes, etc. that can lighten and enhance the user's mood.

2. chatbots can generally answer all of the user's questions, and when the user's questions are more complex or the customer no longer wants to communicate with the chatbot, the chatbot can connect to a human supervisor who will never take over the next step.

3. The chatbot will respond to the user's needs and will make alternatives for the user to choose from. The chatbot can be helpful and proactive in solving any problem the customer has.

4. When a user wants to file a complaint, the chatbot will reassure the customer and connect to a human supervisor to respond to the user's complaint and display the human supervisor's name for tracking.

5. The chatbot can help the user with simple questions such as delivery numbers, links and addresses, and will automatically provide links to help the customer solve the problem.

6. If the user encounters a problem that the chatbot cannot solve, the chatbot will provide feedback to the human supervisor and follow up with him/her. Even if the customer is not willing to accept the alternative solution provided by the chatbot, he/she can follow up by asking for the user's phone number and having the human supervisor make a phone call.

7. The chatbot is online 24 hours a day to help users solve problems and communicate with them, assuming the human supervisor is not online.

The chatbot is online 24 hours a day, and if the user is not satisfied with the solution provided, the chatbot can contact the human supervisor for the next step.

## Information gathering \*\*（没填好）

*Summarize the process of preparing, conducting, and processing the interview(s) held to elicit, confirm, and approve the user requirements. List the individual requirements, as user stories.*

Questions.

1. When communicating with customers, does the chatbot make the communication easy?

2. How does the chatbot solve the user's problem?

3. When a chatbot is unable to solve a problem, what method is used to continue the problem?

4. Can chatbots be hosted in the cloud?

1) Yes. --Are there terms and conditions that dictate where chatbots can be located and integrated?

2) No. -- How do you implement such a full-featured intelligent chatbot?

5. Is there a provision for the tone and language used by the chatbot?

Answer.

1. Yes. Chatbots can be very sharp, direct, and clever, but a bit dark for the average person. The chatbot will provide funny, "hal-like" responses to user input on jokes, other light-hearted topics, and even philosophy.

2. The bot is also supposed to help resolve customer dissatisfaction with the items they purchase from the site. Merchandise dissatisfaction is mainly related to order delays, damaged packages, incorrect packages, etc. Chatbots resolve customer grievances in a typical "HAL" fashion.

3. If any problems are encountered in processing any of the above user requests, the chatbot should be able to provide an alternative solution to the customer's query. If the chatbot connects to a human supervisor and chats with them, the chatbot should indicate that it is connected to a human supervisor and display the supervisor's name. If the chatbot asks the user to provide details of the query in an email, the chatbot should provide the user with the email id to which they can send the query. If the chatbot asks the user if they would like to call the executive immediately, the chatbot should first ask for the user's phone number and then ask permission for one of the executives to call the number.

4. Right. The company will integrate the chatbot into its main application, so the necessary provisions should be made to enable this integration in the future. This requirement is non-negotiable.

5. The chatbot should have a direct and decent tone, using formal language. However, it is always objective, even if this means offending someone.

## Functional Requirements

1. Ability to respond to user questions 24 hours a day

2. the ability to have chatbots integrated into the application

3. the ability to send emails to users

4. send the user's phone number or provide a link to the user

5. the ability to provide a chat tone similar to HML

6. the ability to send timely information about movies, scripts, and actors for HAL 2001

7. allow users to participate in trivial topics

## Non-functional Requirements

1. memory footprint, without taking up too much memory

2. precise language of the chatbot, with the corresponding HAL style, even if to some extent offensive

3. the corresponding time, when the user asks a question, to be able to give a timely answer

4. the number of users, to be able to correspond to the number of users for certain statistics, statistical planning chart

## User classes and characteristics

Most of the people who visit the website are fans of HAL 2001, so we can classify them according to the frequency of using this movie. Those who have primary knowledge or want to know about the movie HAL 2001, and who generally do not use it very often, are classified as user category A.

Characteristics of user category **A.**

1. want to know about the movie HAL, have some interest in the plot of the movie and the comprehensive information of the actors, but have not yet made more understanding.

2. have watched the movie HAL 2001, have a relatively vague understanding of the specific information of the movie, and want to explore more information. 3.

3. did not have a clear memory of the lines in the movie and was not able to directly accept or respond to the chatting atmosphere guided by the chatbot's lines.

4. not very receptive to the trivial discussions about the movie on the website.

Users who like the movie, visit the site at certain times, visit the site a certain number of times, and have some potential purchasing power for the products on the site are classified as User Category B.

Characteristics of user category **B**:

1. they want to know more about the plot of the movie, comprehensive information about the actors and actresses, as well as film footage and behind-the-scenes information, and want to get more information about the movie HAL 2001.

2. people who have a desire to buy products on the website, who can generate some spending power on the website, and who are interested in the peripheral products of the movie.

3. people who have a certain degree of knowledge and memory of the classic lines from the movie/novel, who can respond to the specific lines sent by the chatbot, and who understand and adapt to the chatting atmosphere of the chatbot on the website.

4. have a willingness to participate in and speak up on trivial discussions about the movie on the website.

People who love the movie, visit the site regularly, and buy products on the site are classified as User Category **C**.

Characteristics of User Category C:

1. they know the plot of the movie, comprehensive information about the actors and actresses, as well as the movie highlights and behind the scenes.

2. they buy products on the website frequently and love the peripheral products of the movie/novel

3. have a good understanding of the lines in the movie/novel, and like and enjoy the chatting atmosphere provided by the chatbot with HAL's specific environment.

4. are very involved in the trivial discussions on the site about the movie, and often speak in the topics.

According to the above classification, it can be found that the user A chatbot only serves as a relative introduction to the movie details and provides some links to the users of this category to facilitate their inquiries about the movie/novel, while the chatbot atmosphere with HAL style is not used very often and does not make certain purchases for the products offered on the website. Therefore, I think that chatbots with HAL chatting atmosphere will make this type of users uncomfortable because of the dark atmosphere in the plot, so I think chatbots should use an introductory and enthusiastic tone for this type of users. User B can accept and have some understanding of the lines, and can participate in trivial topic discussions. This type of user is the one who should have a HAL tone, and the chatbot should guide them for carrying out topic discussions, and push some goods in time, they have some interest in the goods, and are potential buying customers. User C should be the most important group of customers, they enjoy the HAL chatting atmosphere and can actively answer the chatbot's questions, this group of people has a strong interest in the products, in pre-sales and after-sales issues, the chatbot should produce a satisfactory response mechanism for these customers. For this type of users, the chatbot should use a HAL chat atmosphere and push new products in a timely manner.

## Constraints

Describe any items or issues that would limit the options available to chatbot developers. These may include: company or regulatory policies; hardware limitations (time requirements, memory requirements); interfaces with other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communication protocols; security considerations; design conventions or programming standards (e.g., if the customer's organization will be responsible for maintaining the delivered software.

Chatbots with HAL style cost more

Because of the need to carry the unique linguistic atmosphere of the HAL 2001 movie/novel in the HAL website, even chatbots based on artificial intelligence and machine learning can be difficult to implement, even if they seem to be the next stage of evolution. Algorithms and frameworks can be sourced from service providers, but aligning them with the features in the entire HAL 2001 movie/novel requires providing the chatbot with relevant information about the entire movie in a predefined format, a massive undertaking, and the task of fine-tuning the AI chatbot for the website to take advantage of all its offerings is a task for experts, which requires significant investment and time.

Limited database and limited responsiveness to customers

In that since most Bots are either support chatbots or service-based chatbots, the number of responses to end users is limited to the scope of their messages. Similar to the three categories of users ABC of HAL 2001, since menu-based chatbots are limited to their queries and try to classify users, but they cannot deviate from other domains or accumulate information from current external scenarios into their responses. As a result, it is limited to the majority of users and is not fully accurate in categorizing customers accordingly

Machine Learning Security Vulnerabilities in Chatbots, Vulnerable to Attacks

Chatbots are particularly vulnerable to machine learning attacks, and because of the different classifications of customers in the HAL2001 website, chatbots often interact with each other as users, and these interactions are often completely unsupervised. Data poisoning is a machine learning attack in which hackers contaminate the training data of a machine learning model. They do this by injecting adversarial inputs that are intentionally altered data samples with the aim of inducing the system to produce incorrect outputs. Systems like customer service chatbots that are continuously trained with user input data are particularly vulnerable to such attacks. Most modern chatbots can operate automatically and answer customer queries without human intervention. Typically, the conversation between the chatbot and the user is never monitored unless the query is escalated to a human. This lack of oversight makes chatbots a prime target for hackers to exploit.

Language Complexity in HAL 2001

Modern conversational interface applications rely heavily on advances in artificial intelligence and the proliferation of connected devices to provide users with shortcuts to perform simple tasks, such as getting answers to simple questions (e.g., who is the female star of HAL 2001) or completing a quick task (e.g., alerting the user that a package of purchased goods will arrive the next day), among other simple tasks. But as with software, language is based on a set of rules that evolve and develop over time. But speaking humans, unlike computers, are not constrained by these rules and are free to form words and sentences as a way to convey a message. Similar to regional dialects aside, what is needed for the HAL 2001 website is the unique language approach of the movie/novel, which traditionally would make it more difficult for a computer to understand a similar discourse. While machine learning has improved the computer's ability to recognize language, we have not yet developed artificial intelligence to the point where it can keep pace with the rapid evolution of language and understand each particular way of speaking at the right time.

## Assumption and Dependencies

Assumptions.

1. the chatbot in this project requires a specific language environment for HAL 2001 movies/novels, for which certain development conditions are required by the third party developing the chatbot. Assuming that no specific language environment is constituted, then the language environment that is wanted to be developed cannot be formed. It will cause some loss to the user's experience.

2. Whether the chatbots in HAL2001 website have consistent and rich personalities: for chatbots to be like credible people, they must be able to simulate the assumption of having unique personalities.

3. The environment in which the chatbot operates is also a hypothetical factor, and the pairing of different environments can affect many aspects of the chatbot.

Since AI chatbots rely on natural learning processing (NLP) to understand the semantics of incoming messages, unless a natural learning processing (NLP) parser is trained on the domain, the accuracy of identifying the intent and topic of interest will be unacceptable."

No chatbot implementation will understand domains. Developers need to train chatbots on custom domains to recognize scenarios

## Example use cases (要改)

图示

描述已自动生成

# Chatbot Design

## System architecture

In 1950, alan turing put forward the Turing Test ("Can Machines Think"), and the first known chat robot was Eliza developed in 1966, with the purpose of responding to users through dialogue and question-and-answer as a psychological trainer. She uses simple pattern matching and template-based response mechanism [1]. Of course, as an earlier chatbot, her conversation ability is relatively poor. In 2000, ALICE won the "Most Humanized Computer". This chatbot relies on a simple pattern matching algorithm, and the underlying intelligence is based on Artificial Intelligence Markup Language (AIML)[2], enabling developers to define the building blocks of chatbot knowledge [3].

Pattern matching is a typical stimulus-response block. Input information (stimulus), the chatbot produces an output response consistent with the user's input. In this mode, the chatbot's dialogue is relatively rigid, repetitive, predictable and impersonal. Without storing past information, it is not suitable for project development.

NLP (Natural Language Processing) is a field of artificial intelligence. It explores the manipulation of natural language texts by computers, collects, understands and uses human language knowledge to develop technologies, so that computers can understand and naturally express the tasks to be performed [5]. Natural language understanding NLU is the core of any natural language processing task, and it is a technology to realize natural user interface, which can make appropriate response according to the user's intention [6]. The machine recognizes the user's intention, and extracts entities in specific fields.

At present, there are three mainstream chat robot models: rule-based model, retrieval-based model and generation-based model.

The rule-based model is more of the architecture type of early chat bots. The corresponding system is selected based on a set of fixed predefined rules, and there will be no new text content based on the recognition of the vocabulary form of the input text. The knowledge used is manually coded and presented in a dialogue mode. Expanding the scale of rule database will allow more types of user input, but chatbots will not give too many corrections to user errors and grammatical errors.

Based on the retrieval, it provides greater flexibility. Before the matching method is applied, the robot will select some corresponding candidates from the index, and finally choose the best choice. [7]

Compared with the other two models, the chat robot based on the generation model is more like a human being, using machine learning algorithm and deep learning technology, so the cost of construction and training will increase relatively.

We will use rasa architecture. rasa has an extensible architecture, and its main components are NLU and dialogue management.

NLU is the part that deals with intention classification and corresponding retrieval. The dialogue management component decides the next action of the dialogue according to the context, and the dialogue history of the robot will be stored in tracker store.

## Dialog Design

*First, get the functions that users want from the demand, and determine the basic functions of the chat robot: it can respond to users' questions at any time, send emails and phone numbers to customers, send information about hal2001 movies, chat with customers using HML, and expect to participate in trivial topics. After determining the basic functions, determine the character and tone of the chat robot, which will be explained in detail later. After the basic intonation is determined, the conversation design will be carried out. The design should simulate possible users and special situations as much as possible, and finally the user test will be carried out.*

*Chat Robot According to the big five personality theory, chat robot will be an understanding and extroverted robot with a humorous tone.*

*For instance*

*User: query the current weather.*

*Bot: Now the weather is light rain to heavy rain. You'd better take an umbrella, brother.*

*User: ok, thank you.*

*Bot: For me, this kind of thing is trivial.*

*User: query the current weather.*

*Bot: It's sunny now, with a temperature of 35 degrees. Be careful not to get heatstroke.*

*User: ok, thank you.*

*Bot: For me, this kind of thing is trivial.*

When the user asks about the weather, the chat robot will remind the user according to the weather conditions of the day. When it rains, remind him to bring an umbrella, and pay attention to sun protection in sunny days.

About commodity inquiry (Happy Path 1)

User: query commodity information.

Bot: Oh, my God, there must be something wrong with it. Can you tell me the order number of the goods quickly?

User:00001

Bot: OK, no problem. Your goods arrived on September 5, 2022.

User: thanks.

Bot: You're welcome. It's easy.

About commodity inquiry (Sad path)

User: Why is there something wrong with my product?

Bot: Oh, my God, there must be something wrong with it. Can you tell me the order number of the goods quickly?

User:00001

Bot: OK, no problem. Your goods have been transferred to the courier company on September 5, 2022. You can go to the relevant courier company for further details.

User: I can't accept this result.

Bot: Well, it seems that we can't keep talking.

Bot: Please choose some services.

1. Transfer to the human supervisor, who will solve the problem.

2. This is the email ID XXXXXXXXX of the executive. Please contact him actively, and he will solve the problem.

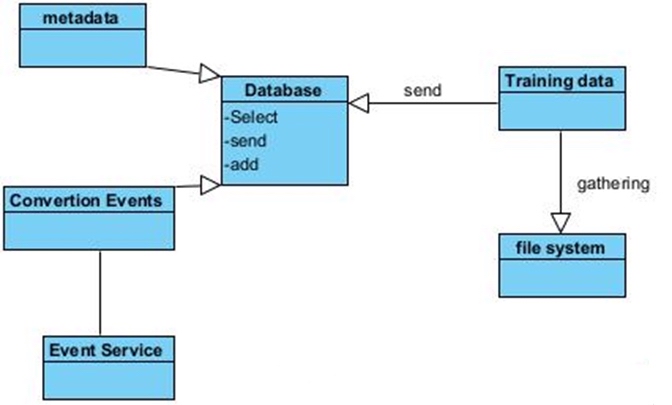
Please tell me your telephone number, and I will dial the high-level telephone for you, and he will solve the problem.

When the customer tries to inquire about the purchased goods, the robot will let the user enter the product id, and if there is no problem with the purchase of the goods, it will politely inform the user of the specific information of the goods. When the customer uses a grumpy and impatient tone, the chatbot tries to calm the user and ask for the product id.

If the specific information can't be inquired, three ways are given for users to choose.

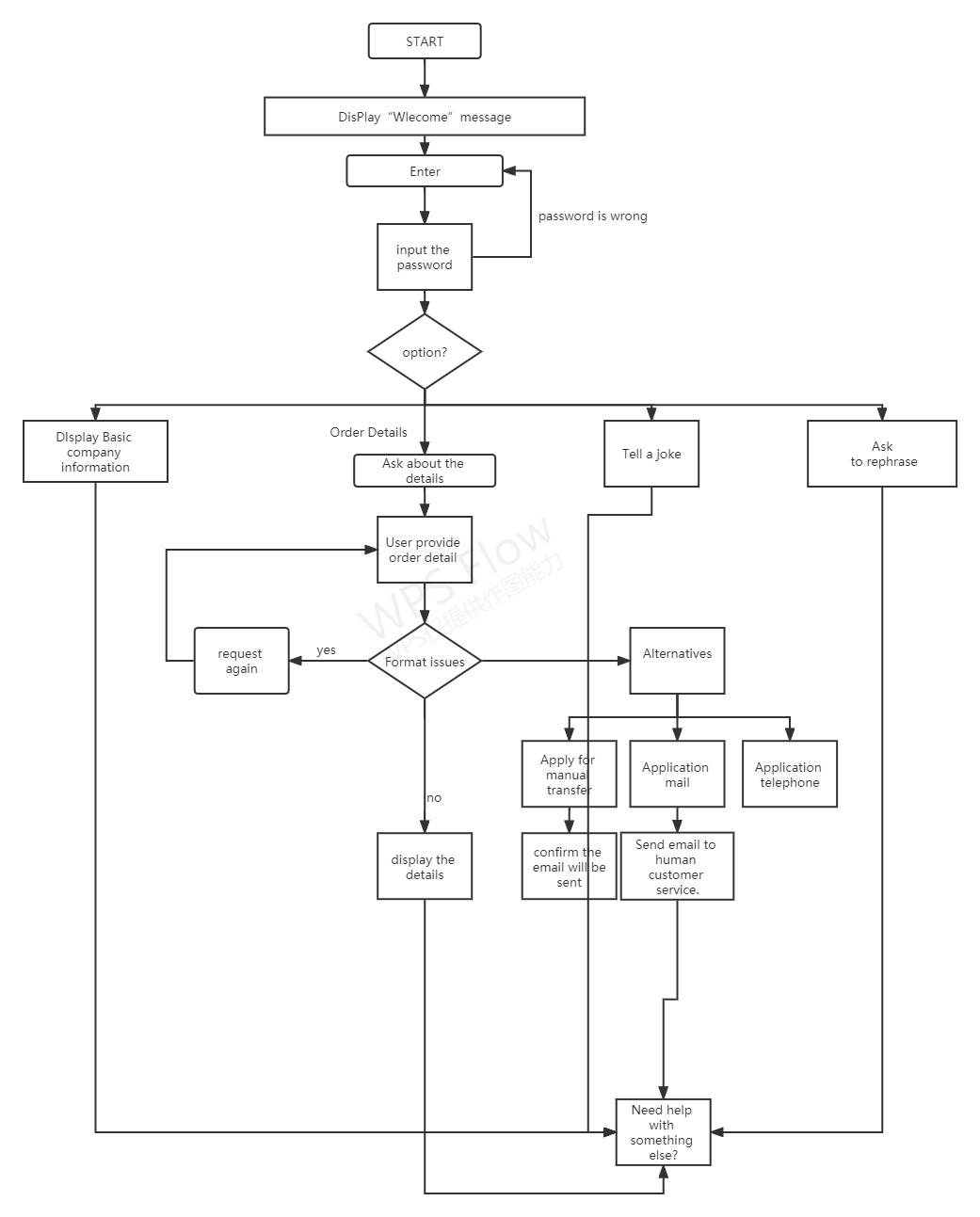
More dialogues will be shown in the appendix.

## Data Design



The database will accept metadata, training data and dialogue events, and can select and add them. The training data will eventually be collected in the file system, and the dialogue events will be transmitted by the event service module.

## System Operation



When entering the chat robot, the robot will introduce itself and then let you log in. After logging in, you will have a variety of functions to choose from. Of course, it can also be triggered in small talk. If the user is in a bad mood, the robot will tell jokes to make the user happy. If the user tries to chat with the robot on some profound topics, the robot will try to respond. When the customer asks to inquire about product information, if the chat robot can handle it, it will continue to handle it. If the relevant information can't be processed, it will be given to the user to choose three manual takeover schemes. After a period of no response, the robot will try to talk to the user again. If there is still no response, it will be regarded as the user leaving, and goodbye politely.

# Chatbot Development

*This section will describe the overall development efforts, including integration, undertaken for your chatbot. This section should not take more than* ***five to six pages.***

## Development Environment

*Depending on the way used, describe your development environment, which includes both hardware and software specifications.*

Development environments are as follows:

**For the software:**

Processor Intel (R) Core (TM) i7-9750H CPU @ 2.60GHz 2.59 GHz

Tape RAM 16.0 GB (15.8 GB available)

System type 64 bit operating system, x64 based processor

**For the hardware:**

Using PC server.

**For the developing tools:**

Visual Studio Code + SQLite

Rasa Version : 3.2.8

Minimum Compatible Version: 3.0.0

Rasa SDK Version : 3.2.1

Python Version : 3.7.13

Operating System : Windows-10-10.0.19041-SP0

Python Path : E:\Anaconda\Ana\envs\installingrasa2\python.exe

1. Necessary components like stories, intents, slots, entities, custom actions are vital, but it’s also necessary to add some more rules, forms, and fallbacks as we need to build a more powerful chatbot. All of which could be for either retrieving information from an external source, or storing user-provided information on external database, or a combination of both.
2. Secondly, tools like rasa-core and rasa-nlu are obviously needed because of different training models and natural language process to comprehend intents and fixes slots.

Rasa NLU: used to understand user messages, including intention recognition and entity recognition. It will convert user input into structured data.

Rasa Core: It is a dialogue management platform for holding dialogues and deciding what to do next. Rasa x is also used, which is a tool that helps you build, improve, and deploy AI Assistants supported by the Rasa framework. Rasa X includes user interfaces and REST APIs.

1. Technologies.

Firstly, it has a multi-channel customer experience. Rasa includes 10 built-in messaging channels, plus endpoints for custom channels. Engage customers across messaging platforms, powered by a single backend assistant.

Secondly, High performance architecture, making overwhelmed call centers a thing of the past. Rasa’s robust architecture scales up to meet high traffic demand, without putting pressure on human support agents.

Lastly, Versatile, reusable infrastructure. Avoiding reinventing the wheel with every new virtual assistant. Rasa’s technology is transferable across use cases, so you can re-use skills and apply work from previous projects to new opportunities.

To act more vivid and more like a personal assistant, we decided to add one more function to fill the slot which saves the users’ name as an entity, and will be added in the next conversation flows with the name on.

1. APIs：

**Enabling the HTTP API**

By default, running a Rasa server does not enable the API endpoints. Interactions with the bot can happen over the exposed webhooks/<channel>/webhook endpoints.

To enable the API for direct interaction with conversation trackers and other bot endpoints, add the --enable-api parameter to the run command: rasa run --enable-api

Note that starting the server with an NLU-only model, not all the available endpoints can be called. Some endpoints will return a 409 status code, as a trained dialogue model is needed to process the request.

By default, the HTTP server runs as a single process. We can change the number of worker processes using the SANIC\_WORKERS environment variable.

**NLU-Only Server**

You can run an NLU-only server and use the HTTP API to connect to it. Connecting to an NLU server#

You can connect a Rasa NLU-only server to a separately running Rasa dialogue management only server by adding the connection details to the dialogue management server's endpoint configuration file:

endpoints.yml

nlu: url: "http://<your nl u host>:<your nlu port>"token: <token> # [optional] token\_name: <name of the token> # [optional] (default: token)

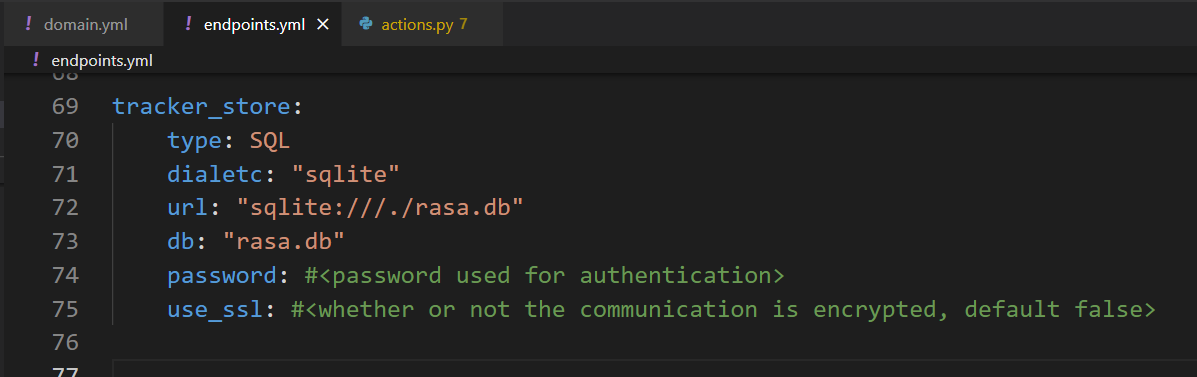
The token and token\_name refer to optional authentication parameters. The dialogue management server should serve a model that does not include an NLU model. To obtain a dialogue management only model, train a model with rasa train core or use rasa train but exclude all NLU data. When the dialogue management server receives a message, it will send a request to http://<your nlu host>:<your nlu port>/model/parse and use the parsing information returned.

## Integration

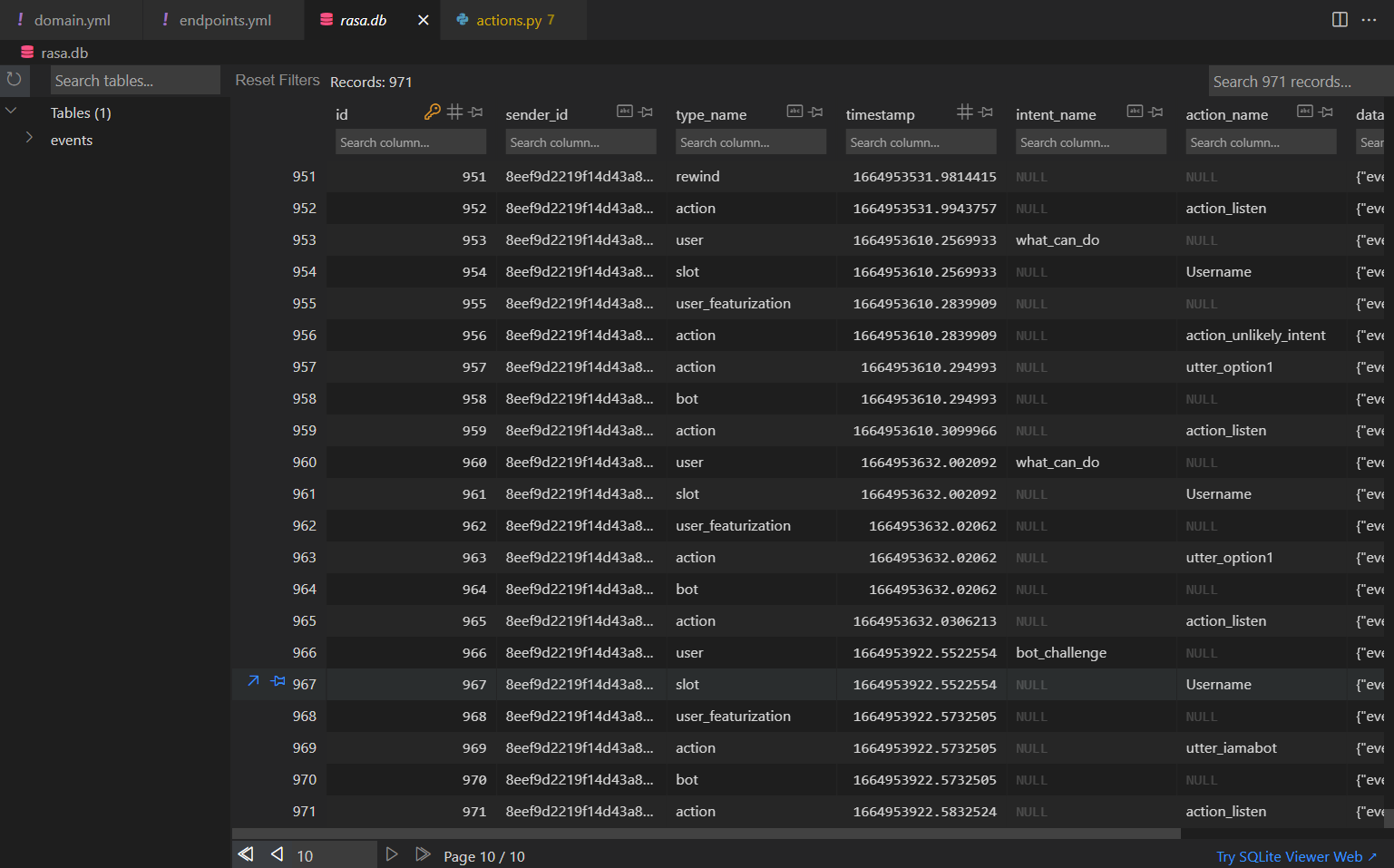
Since we are using RASA core(3.x.x) from the start, and it’s not easy to downgrade the core to 2.8.x to adapt to Rasa X. Therefore, we choose to use Tracker store be our only viable alternative, even though it can be quite annoying. Also, I tried to use to use RedisTrackerStore, which occasionally work but not full functional, those tries are attached in 5.2.2.

### Tracker Order

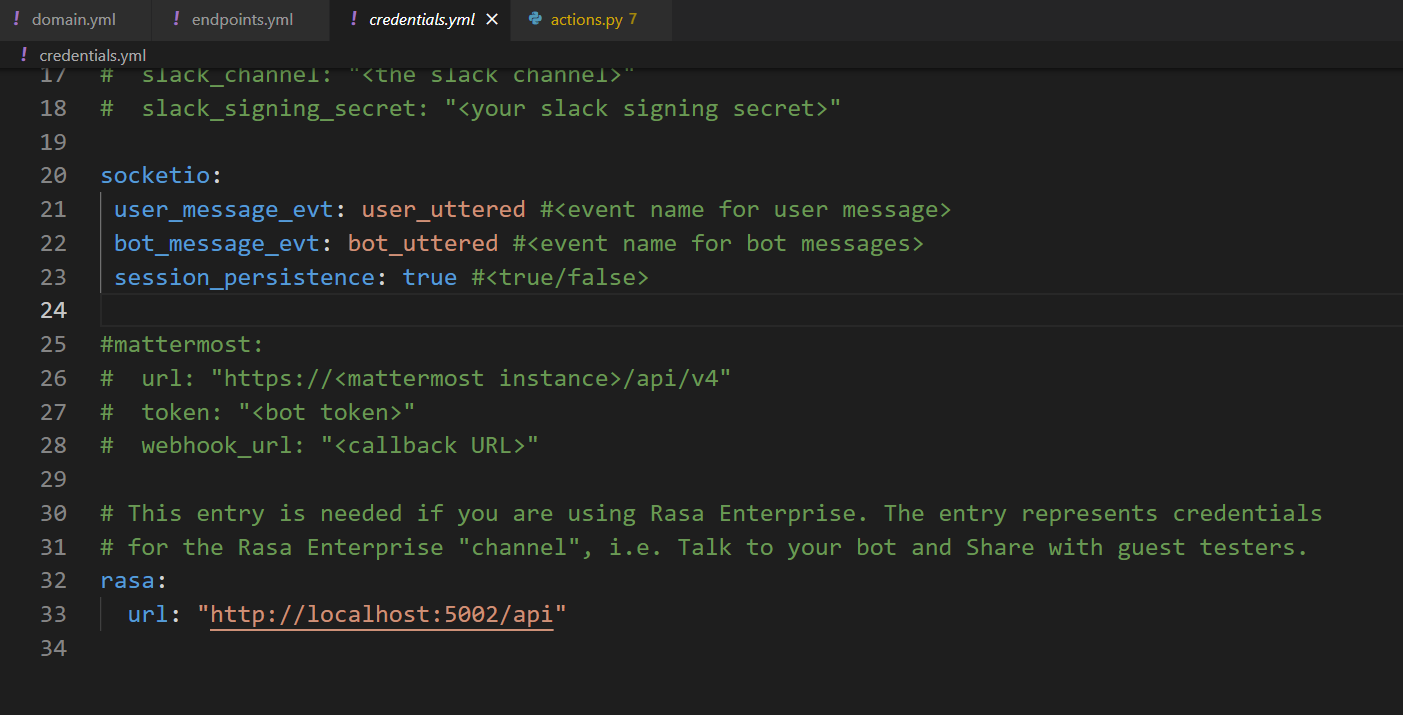
Implementing Tracker Order is easy, there are three files need to be edited. First, of course endpoints.yml:



In which ,we used SQLite to store the tracker data, and we use the self-contained SQLite Viewer to check the dialogue data:

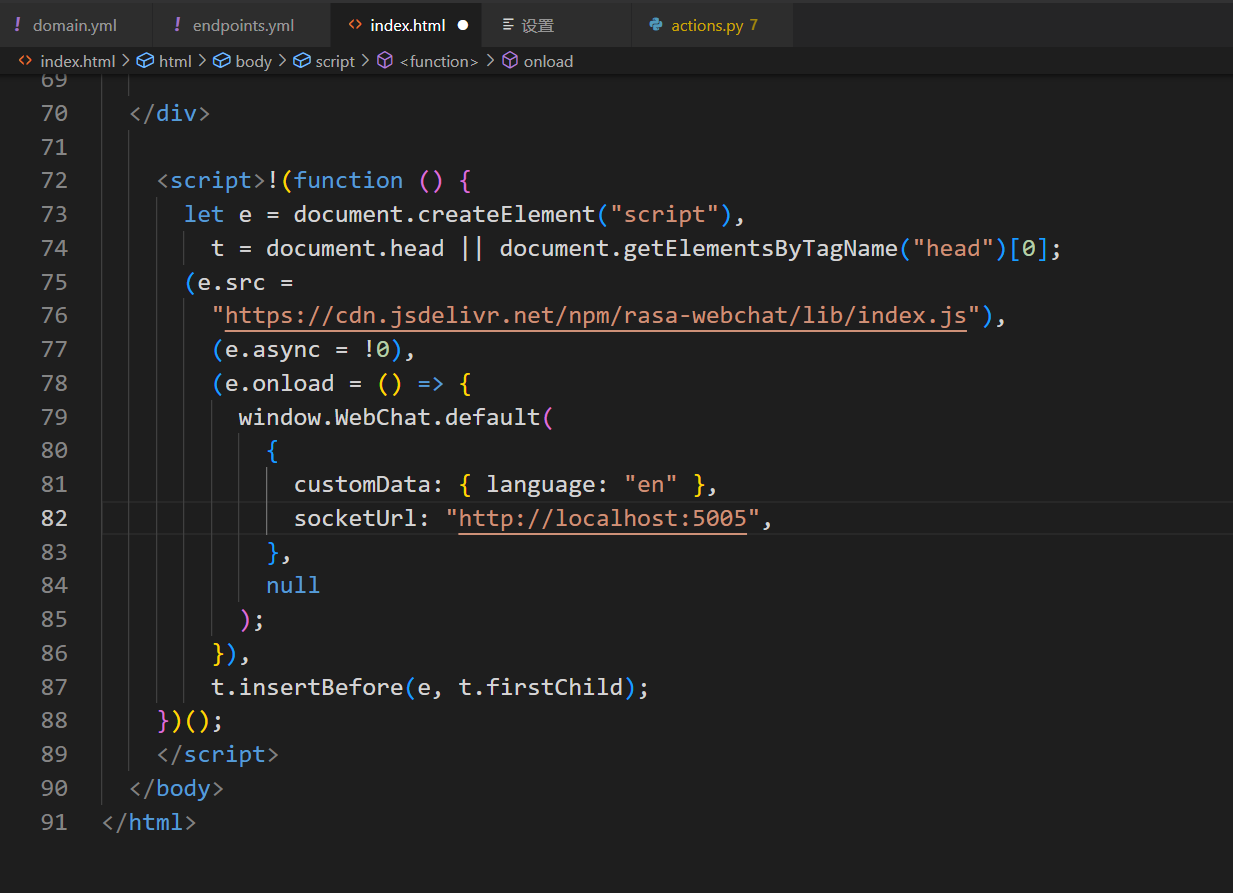


Secondly, to connect with HTML5, I choose to use the Socket.IO, since Socket.io provides event-based real-time bi-directional communication that is a very important requirement for real-time data transfer between Browser and Webserver. WebSocket is a new communication protocol in HTML5, featuring a server that can actively push information to the client and a client that can actively send information to the server, which is a true two-way equal dialogue and belongs to a kind of server push technology. And for this specific case, it should be filled with event name for both user message and bot message in order to save enough data to analyze. Meanwhile, the session\_persistence should also be turn on, which directs a client’s requests to the same backend web or application server for the duration of a “session” or the time it takes to complete a task or transaction.(quate [What is Session Persistence? - NGINX](https://www.nginx.com/resources/glossary/session-persistence/))

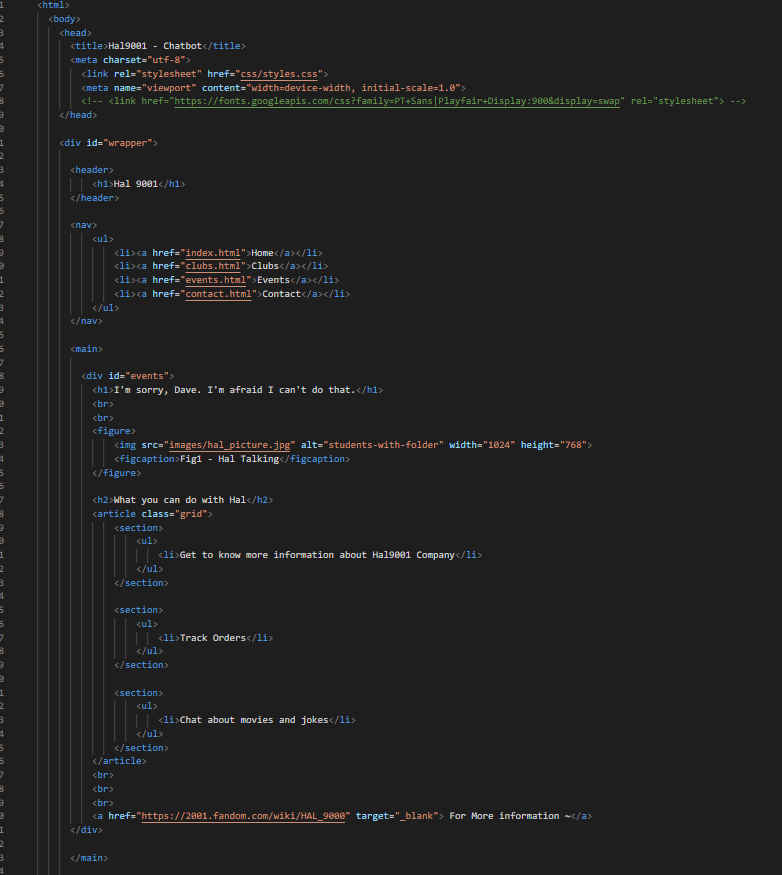


Thirdly, I added some elements in index.html ,there are some crucial points should be pointed out.

* The **Header** to communicate with each other is very small, probably only 2Bytes
* socketUrl should point ar localhost. And can be changed to other channels if needed.

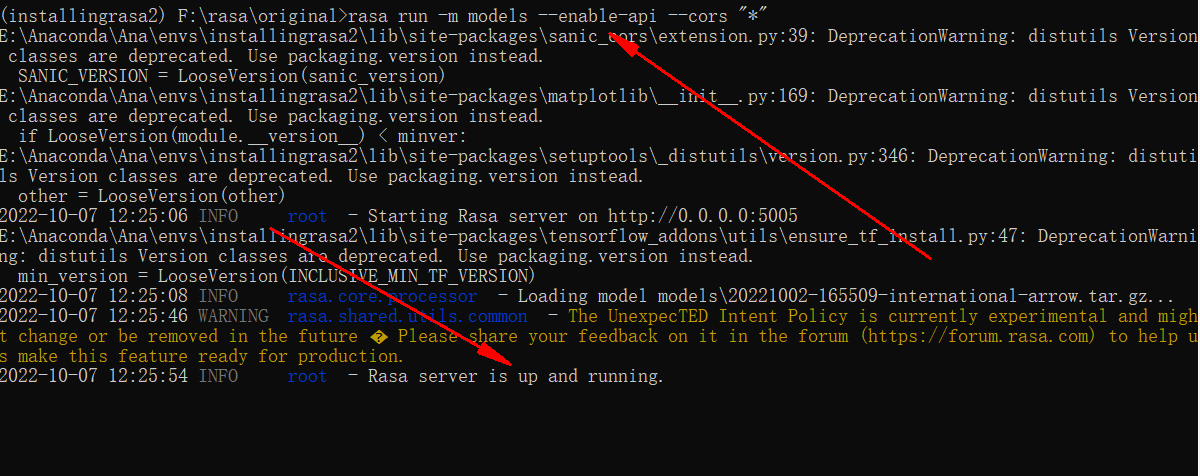


I added some elements to the index.html and css file in order to make the interface more beautiful and to meet the testing requirements at the final delivery.

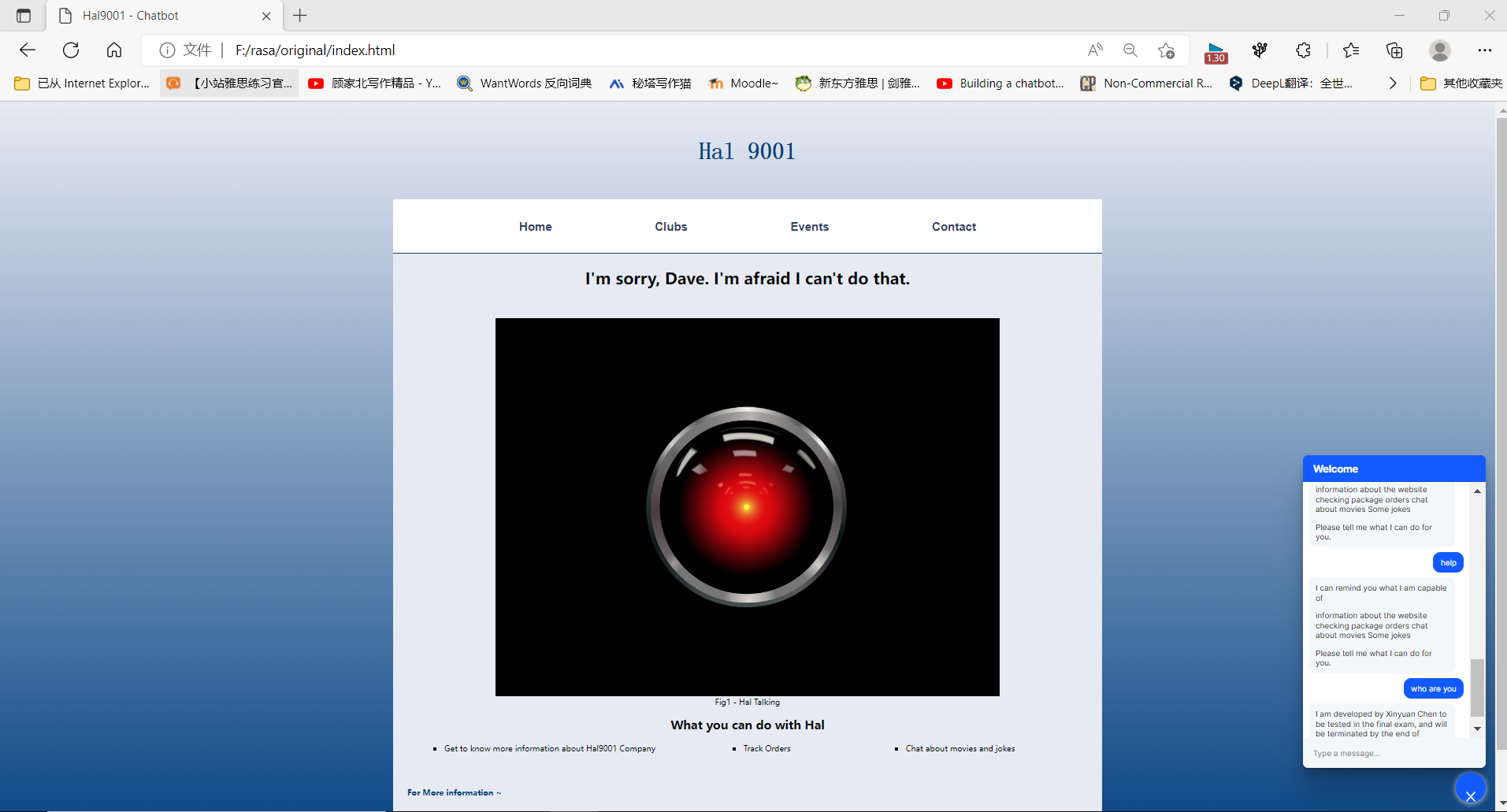


Then we are here to enable the APIs to connect with channels:

rasa run -m models --enable-api --cors "\*"



Then copy the absolute path of index.html and paste it to any web page and it shows, along with dialouge data stored in rasa.db:



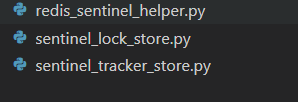
### 5.2.2 RedisTrackerStroe

Here I used RASA Session Data Store RedisTrackerStore Connected

Background:

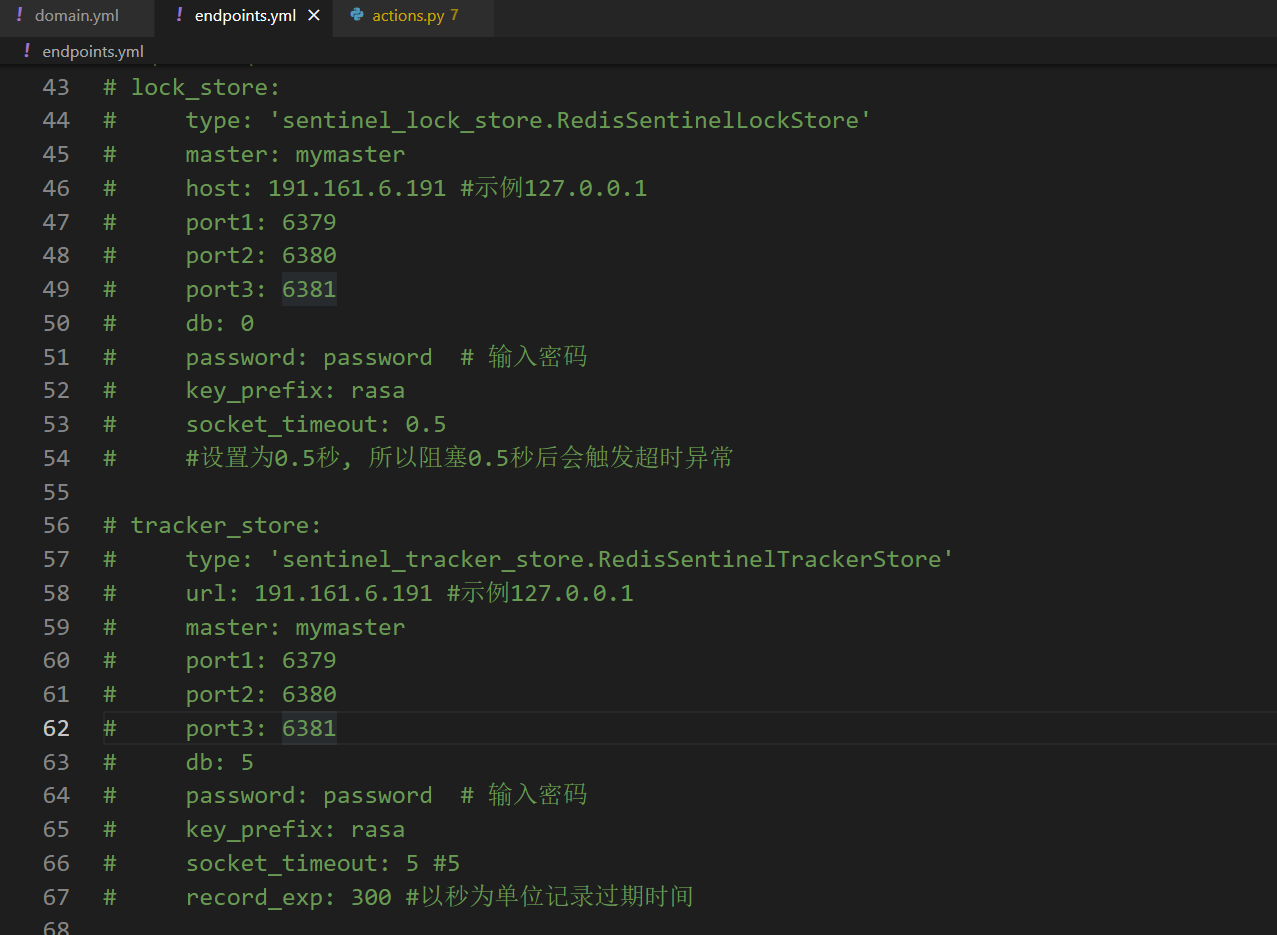
Rasa session logs are stored in memory by default, but if we want to support multiple instances of bots, we need to store rasa session data in the database, and the official documentation supports single-instance redis, which cannot support redis clusters.

So here's how to use Redis cluster for session log storage. The connection here is to a Redis sentinel cluster. According to the example, you can extend the connection to the Redis cluster by modifying the corresponding code section. This code supports rasa3 version.

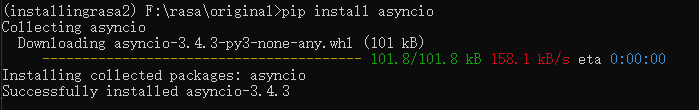


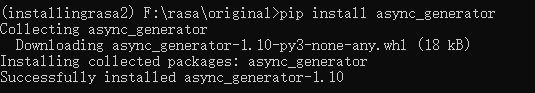
**The concrete code should be checked with the original code, here only present some rough screenshots.**

**Connecting to the endpoints.**

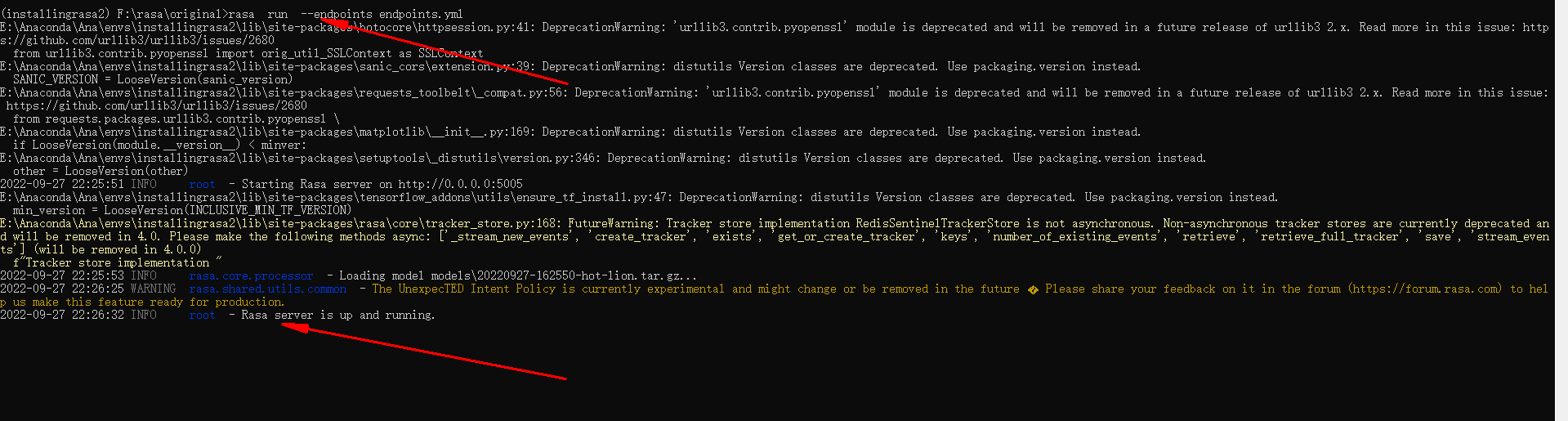


Downloaded some tools:





It can be seen from the figure, it can also run on the server.



But eventually I choose to use the Tracker Order, which is more clean and easier to teach to others when applying future work.

# Testing

*This section describes both the internal testing and user testing carried out to test and evaluate the chatbot. Do not take more than* ***four to five pages*** *to write this section.*

## Internal Testing

*Describe the simulated conversational testing by listing some of the conversations* ***(screenshots****) you utilized for testing the chatbot. Pick examples that you have elaborated in the previous sections to discuss* ***user case views, dialog design,*** *and* ***system operation.***

*Next, discuss the outcomes of the testing; like the shortcomings identified and improvements recommended to the development team. Use the* ***seven-category checklist*** *already shared with you as a tool to test the chatbot.*

**we tested internally, and find most occasions, we use “ok” instead of “please recommend me some movies”, so we changed the story: chat1 movie, change the intent into “affirm” instead of “good\_movies”. Which you can see in the nlu.yml and domain.yml.**

## User Testing

*Describe the simulated conversational testing by listing some of the conversations (screenshots) you utilized for testing the chatbot. Pick examples that showcase significant use cases of your chatbot within the context of your company and industry.*

*Next, discuss your evaluation of the final chatbot using the* ***seven-category checklist*** *already shared with you as a tool of evaluation.*

**\*\* Both the *Testers* and the *Stakeholders* will take the lead on their relevant parts in section, followed by sufficient assistance from the *Developers*. The rest are also encouraged to contribute, especially in proof reading.**

# Conclusion

*This section will summarize your entire document by talking about only the key takeaways. Should not be more than* ***one to two pages****.*

*Start by summarizing the need for your chatbot within the context of its dictated domain, problem statement, the solution conceived, the steps taken to develop that solution, the benefits of using your developed chatbot, the current shortcomings, and its assessment based on the user testing.*

*Next, also briefly discuss how your chatbot could be improved, outlining potential future work.*

**\*\* Everyone or anyone from the group can take responsibility of this section. Ideally, everyone should contribute, at least in proof reading.**

# References

# References

|  |  |
| --- | --- |
| [1] | V. R. B. G. Cldiera and D. Rombach, "The goal question metric approach," *Encyclopedia of software engineering,* vol. 2, pp. 528-532, 1994. |
| [2] | V. R. C. G. &. R. H. D. Basili, "Experience factory.," in *Encyclopedia of software engineering.*, 1994. |
| [3] | Weizenbaum, J.: ELIZA—a computer program for the study of natural language communication between man and machine. Commun. ACM **9**, 36–45 (1966). https://doi.org/10.1145/ 365153.365168 |
| [4] | Marietto, M., et al.: Artifificial intelligence markup language: a brief tutorial. Int. J. Comput. Sci. Eng. Surv. **4** (2013). https://doi.org/10.5121/ijcses.2013.4301 |
| [5] | Jung, S.: Semantic vector learning for natural language understanding. Comput. Speech Lang. **56**, 130–145 (2019). https://doi.org/10.1016/j.csl.2018.12.008 |
| [6] | Wu, Y.,Wu,W., Xing, C., Zhou,M., Li, Z.: SequentialMatching Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots. arXiv:1612.01627 [cs] (2016) |
| [7] | Wallace, R.S.: The anatomy of A.L.I.C.E. In: Epstein, R., Roberts, G., Beber, G. (eds.) Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer, pp. 181–210. Springer, Cham (2009). https://doi.org/10.1007/978-1-4020-6710- 5\_13 |

Appendix A – List of User Stories

As a customer, I want the chatbot to provide interesting and "HAL-like" responses to relevant information entered by the user, information about the website such as weather, jokes or other light-hearted topics, or even philosophy.

As a customer, I want the chatbot to resolve customer grievances in a typically "HAL-like" manner.

I want to be able to resolve complaints about products such as delayed orders, damaged packaging, incorrect packages, etc.

As a customer, I would like the chatbot to be able to provide an alternative solution to the customer's query or even in some cases connect to a human supervisor and chat with them to resolve the issue, should the chatbot encounter any problems while handling any of the above user requests.

As a customer, I would expect that once the user chooses this alternative, the chatbot should indicate that it is connecting to the

human supervisor while also displaying the supervisor's name or asking the user to provide their detailed query in an email.

As a customer, I would expect that after choosing this alternative, the chatbot should provide the user with an email id and the chatbot could send their query to ask the user if they would like to speak with the executive over the phone immediately.

As a customer, I would like to see that in this alternative, the chatbot should first ask the user for their phone number and then ask permission for one of the human executives to call that number.

As a customer, I would like the chatbot to be hosted in the cloud, just like the company's main application

As a customer, I would like the company to integrate the chatbot into its main application, so the necessary terms should be done to make future integration possible, and this is a requirement that must be met.

As a customer, since the company expects customers from all walks of life, I would expect that the chatbot should have a simple, dignified tone and be able to use formal language. However, a chatbot will always be objective, even if it means to offend.

Appendix B – List of functional and non-functional requirements

***(As user stories)***

Appendix C – Dialogs

holiday words

Bot: Hello, I'm Hal Bionic Robot. Welcome to hal Fanxiang website, where you can find everything related to hal. What I can do is:

1. Provide relevant information of the company.

2. Problems related to orders

3. Tell a joke

All right, don't waste time, just choose one, brother.

User: hello

Bot: Hello, what can I do for you?

User: goodbye.

Bot: Brother, goodbye.

Query the weather

User: query the current weather.

Bot: Now the weather is light rain to heavy rain. You'd better take an umbrella, brother.

User: ok, thank you.

Bot: For me, this kind of thing is trivial.

User: query the current weather.

Bot: It's sunny now, with a temperature of 35 degrees. Be careful not to get heatstroke.

User: ok, thank you.

Bot: For me, this kind of thing is trivial.

Information about the website

User: Do you know xxx?

Bot: Oh, of course I know, xxx is (list basic information)

User: basic information of the website

Bot: "2001: A Space Odyssey" (2001: A Space Odyssey) is an American science fiction film directed by stanley kubrick and adapted from the novel of science fiction writer arthur clarke. It was released in 1968 and is known as "the milestone of modern science fiction film technology"

About commodity inquiry (Happy Path 1)

User: query commodity information.

Bot: Oh, my God, there must be something wrong with it. Can you tell me the order number of the goods quickly?

User:00001

Bot: OK, no problem. Your goods arrived on September 5, 2022.

User: thanks.

Bot: You're welcome. It's easy.

About commodity inquiry (Happy Path 1)

User: Why is there something wrong with my product?

Bot: Oh, my God, there must be something wrong with it. Can you tell me the order number of the goods quickly?

User:00001

Bot: OK, no problem. Your goods have been transferred to the courier company on September 5, 2022. You can go to the relevant courier company for further details.

User:emmm, OK, I probably get it.

Bot: If you get it, go and find it. Don't waste your time with me here.

About commodity inquiry (sad path)

User: Why is there something wrong with my product?

Bot: Oh, my God, there must be something wrong with it. Can you tell me the order number of the goods quickly?

User:00001

Bot: OK, no problem. Your goods have been transferred to the courier company on September 5, 2022. You can go to the relevant courier company for further details.

User: I can't accept this result.

Bot: Well, it seems that we can't keep talking.

Bot: Please choose some services.

1. Transfer to the human supervisor, who will solve the problem.

2. This is the email ID XXXXXXXXX of the executive. Please contact him actively, and he will solve the problem.

Please tell me your telephone number, and I will dial the high-level telephone for you, and he will solve the problem.

Appendix D – Git Repository

***(Include the procedure to set up the environment to run your chatbot. This is a common practice. Look at some git repositories to get an idea on what and how to write this part)***

Appendix E – Seven-category assessment (Internal Testing)

Appendix F – Seven-category assessment (User Testing)