A Project Report on

Chatbot-Pizza Ordering

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Computer Engineering

by

Sayali Kamble(16102039)
Zahid Khan(16102051)
Vishal Jain(16102044)

Under the Guidance of

Prof. Archana Kotangale

Department of Computer Engineering
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Approval Sheet

This Project Report entitled ***"Chatbot-Pizza Ordering"*** Submitted by ***"Sayali Kamble"(16102039),"Zahid Khan"(16102051),"Vishal Jain"(16102044)***is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Prof. Archana Kotangle
Guide

Prof. Sachin Malave
Head Department of Computer Engineering

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"Chatbot-Pizza Ordering"*** submitted by ***"Sayali Kamble" (16102039), "Zahid Khan" (16102051), "Vishal Jain" (16102044)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Engineering***,to the University of Mumbai,is a bonafide work carried out during academic year 2019-2020.

Prof. Archana Kotangle
Guide

Prof. Sachin Malave
Head Department of Computer Engineering

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Sayali Kamble(16102039)

_____

Zahid Khan(16102051)

_____

Vishal Jain(16102044)

_____

Date:

Abstract

Chatbots are computer programs that are developed using Artificial Intelligence for providing an easy interference between the computer and humans . The interaction can be textual or auditory depending upon the need. The technology at the core of the rise of the chatbot is Natural Language Processing(NLP). The project aims to develop a 'Pizza Ordering Chatbot' for which We are going to develop a chatbot using dialogflow that incorporates machine learning expertise and products such as google cloud speech to text conversation . Chatbots can be used in Customer service , sales/marketing and also as a human resource .This pizza ordering Chatbot is beneficial in many ways as it offer mass communication service , improves customer satisfaction and reduces cost .

Contents

List of Figures

List of Abbreviations

GCP : Google Cloud Platform
ML : Machine Learning
NLP : Natural Language Processing
GUI : Graphical User Interface

Chapter 1

Project Conpetion and Initiation

1.1 Introduction

One of the most emerging trend in the development of robotics is Chatting robot.
Chat bot is a computer program which conducts a conversation via textual or auditory method.
This chatbot project is a android application chatting interface for ordering pizza that will be developed using Artificial Intelligence algorithms.

1.2 Objective

1.Quick and easy to use interface for customers.

2.To support and scale up the business of a Pizza Restaurant .

3.Knowing the order patterns for frequent customers.

1.3 Problem Definition

To Design and Develop a Chatbot for a pizza restaurant that would overcome the problems like unable to keep track of ordering patterns for frequent customers and customer feedback .

1.4 Scope

The Proposed chatbot will be useful in easy handling a pizza restaurant app . This chatbot will help a customer to order a pizza using a text or voice based chat .
The customer will also be able to search for different variety of pizza options available using the chat console . The customer can also apply the coupons and then calculate the total payable amount for the pizza .

1.5 Technology Stack

1.Dialogflow :
Dialogflow is an end-to-end , build-once deploy-everywhere development suite for creating conversational interfaces for websites , mobile applications ,messaging platforms and IoT devices .

2.Web Application :
A web application (or web app) is an application software that runs on a web server, unlike computer-based software programs that are stored locally on the Operating System (OS) of the device.
HTML
HTML stands for Hyper Text Markup Language .
HTML is the standard markup language for creating Web pages .
HTML describes the structure of a Web page .
HTML consists of a series of elements
CSS
CSS is a language that describes the style of an HTML document.
CSS describes how HTML elements should be displayed.
JAVASCRIPT
JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

3.Flask :
Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries.
It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.
However, Flask supports extensions that can add application features as if they were implemented in Flask itself.

1.6 Benefits for Society

1.Accessible anytime :
More time is been wasted till operators connect customers to a customer care executive. They are replacing live chat and other forms of slower contact methods such as emails and phone calls.

2.Handling Capacity :
Unlike humans who can only communicate with one human at a time, chat bots can simultaneously have conversations with thousands of people. No matter what time of the day it is or how many people are contacting you, every single one of them will be answered immediately.

1.7 Application

1. Content delivery:
A lot of publishers are also harnessing AI and machine learning technology within their chatbots to anticipate what content their consumers may be interested in.

2. Book Flights:
Chatbot gives their customers the ability to search for and book flights in a text-based conversational manner. Instead of drop-down menus, customers enter the information themselves.

3. Companionship:
A Russian company has developed its companion chatbot for Senior People and Patients with Alzheimer's Disease.
The primary function of the chatbot is to be a virtual companion – To speak with senior people on general topics like the weather, nature, hobbies, movies, music, news, etc.

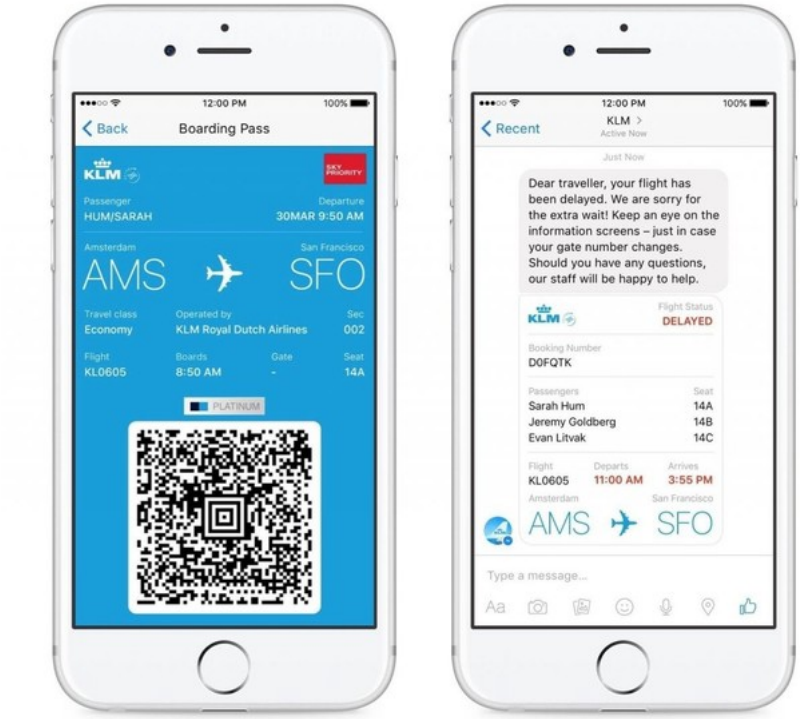

Figure 1.1: Flight Booking Chatbot

Figure 1.2: Companionship Chatbot

Chapter 2

Literature Review

A. Nsmav-Bot: Intelligent Dual Language Tutor System
Sambit Mohapatra ; Neeraj Shukla ; Simoni Jain ; Shweta Chachra 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)
To deal with low literacy rate of developing countries the paper aims to develop a bot that's going to act as a tutor of standard education for age groups.
The bot will be built with the Microsoft Bot Framework using Node.js and will be integrated with the Facebook Messenger. It will be designed to communicate intelligently in English as well as Hindi languages trained using Recast.ai and Dialogflow so as to target the rural along with the urban population.

B. Design of information system architecture for the recommendation of tourist sites in the city of Manta, Ecuador through a Chatbot
David Arteaga ; Juan Arenas ; Freddy Paz ; Manuel Tupia ; Mariuxi Bruzza 2019 14th Iberian Conference on Information Systems and Technologies (CISTI)
The current paper will present a solution for the recommendation of tourist places of the city of Manta, Ecuador.
These recommendations will be performed through the decision trees algorithm and will have a chatbot as user interface in Facebook Messenger. The presented solution integrates different components, and this paper will describe the proposal by means of diagrams of software architecture. Within these components, the integration of Web services such as IBM Watson Assistant and Google Dialogflow will be presented.

C. An Intelligent web-based voice chat bot
Salomon Jakob's du Preez , Manoj Lall and Saurabh Sinha, An Intelligent Web-Based Voice Chat Bot, in EUROCON 2009, EUROCON '09. IEEE, 2009, p. 386
This paper deals with the working of AIML based chat robot. A Java Program is developed which convert AIML files into database. This program is embedded into website which can

in turns help its customers to develop bots. The major technological enhancement in this research is integrating speech recognition and text to speech converter. This empowers the bots to respond to user queries using voice instead of text and humans to chat with bots using voice instead of text messages.

D. Chinese Intelligent Chat Robot Based on the AIML

Wei Yun-Gang, Sun Bo, Sun Ming-Chen, Zhao Cui-Yi, and Ma PeiZi, "Chinese Intelligent Chat Robot Based on the AIML language", in Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics, 2014, p. 368.

This paper explains the language enhancements in the field of Chatting Bot Development system. Here, the bot developed is in Chinese Language known as Chinese Intelligent Chat Robot Xiao Hui-hui. This major technological enhancement in this research paper is language enhancement that AIML can be used not only to develop bots in English Language but also in many other foreign languages such as Chinese, Japanese, Indonesian, Hindi, Marathi etc.

E. Development and Implementation of a chat bot in a Social Network

Salto Martínez Rodrigo, Jacques García Fausto Abraham, Development and Implementation of a Chat Bot in a Social Network, in Ninth International Conference on Information Technology - New Generations, 2012, p. 751

This paper describes the linking of chat bot with social network. It describes that how a chat bot can be linked with Twitter to entertain the users. It can also be used for advertisements. The bot is linked with Twitter since it parts from a simple concept, the exchange of short messages no longer than 140 characters which drastically reduces the amount of information and the way it is published. The algorithm process in this bot is divided into three different parts:
1 Message reception.
2 Message processing.
3 Generation of a suitable reply.

Chapter 3

Project Design

3.1 Proposed System

The given system will be a chatbot that will be simply integrated with a pizza restaurant web page .
The process of ordering a pizza becomes much more simpler with the help of the chatbot .
A customer will be provided with a single interface to interact for ordering a pizza also the customer can search for variety of pizzas , choose, customize and check for prices of the pizza using the chatting interface .
The chatbot is made with the help of dialogflow (former, api.ai) which is connected with the web page by means of 'webhook' which is nothing but built-in code provided with dialogflow connectivity .
Dialogflow uses various Machine Learning algorithm to the train the 'intents' provided by the developer . It also has an easy interface to change the training dataset whenever required.
This chatbot will an easy and time saving way to order the pizza from a website .

3.1.1 Activity Diagram

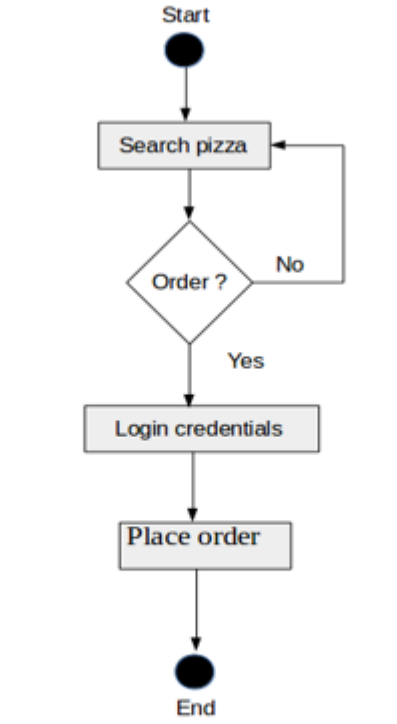

Figure 3.1: Activity Diagram

3.1.2 Use Case Diagram

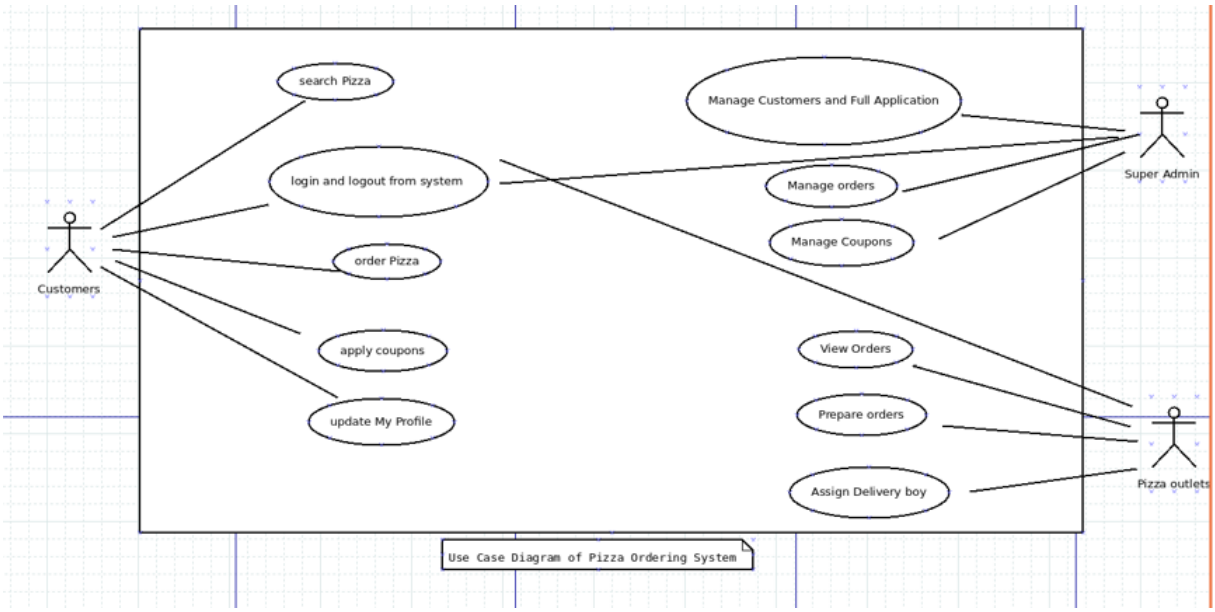

Figure 3.2: Use Case Diagram

3.1.3 Description of Use Case Diagram

1.Customers can search for different type of pizzas , if they wants to order a pizza then they need to provide login id and password details and then the order gets placed . Customers can also check prices of the pizza , apply suitable coupons and update their profiles .

2.A pizza delivery outlet can see the placed orders , prepare for them and subsequently assign a delivery boy to drop the order .

3.A super admin can will have full control over the application .

3.2 Module DIALOGFLOW

Dialogflow (formerly Api.ai) is a Google-owned developer of human computer interaction technologies based on natural language conversations. The company is best known for creating the Assistant , a virtual buddy for Android, iOS, and Windows Phone smartphones that performs tasks and answers users question in a natural language. User: We, Machines!

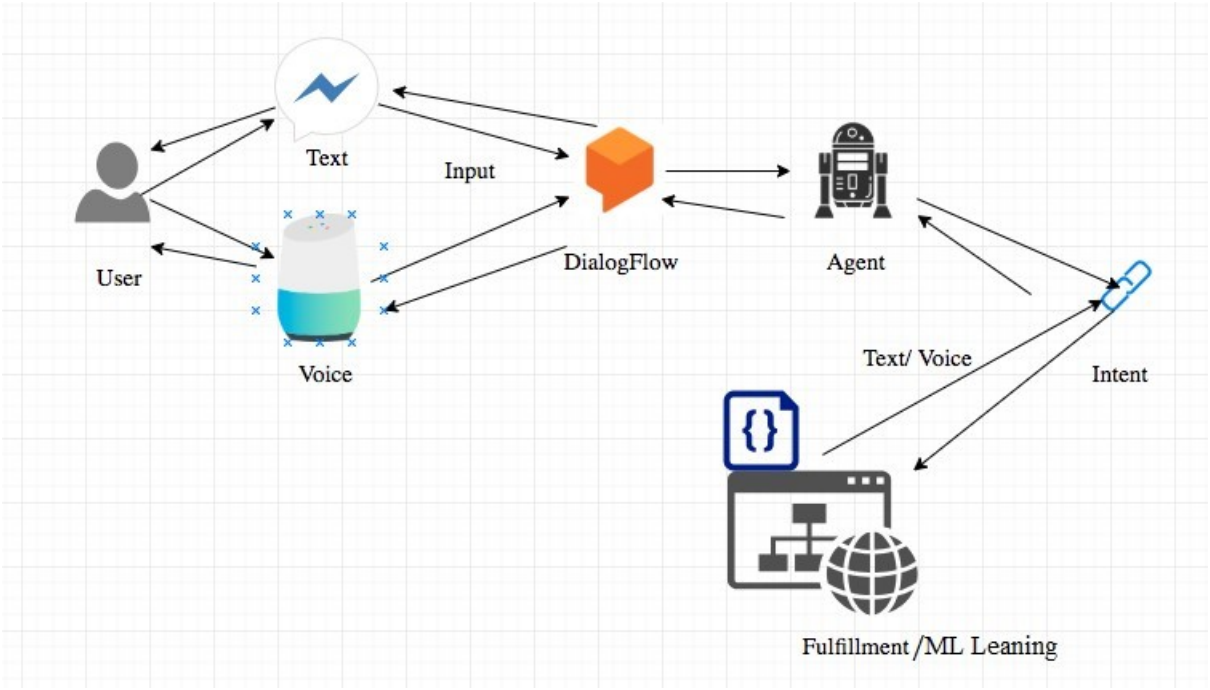

Figure 3.3: Dialogflow

Text / Voice : The user interacts with an app like facebook messenger / google home to start the interaction with the bot.
Dialogflow: Bot platform
Agent: A module within dialogflow which incorporates Natural Language Processing to understand what the user meant and to figure out what action has to be carried out. The agent transforms the user request into machine readable actionable data.

3.2.1 Module 1 - Intents

Intents are the words that replace the entire sentence written by a end-user . Intents basically categories the users intentions behind a conversation .

Dialogflow matches the end-user expression to the best intent in the given agent . This is also called as intent classification . Intents are defined are with a prefix '#'.
A basic intent contains the following :
1.Training Phrases : These are the sentences provides by the developer as a form of examples.
2.Actions : When an intent is match dialogflow performs the given action for that intent.
3.Parameters : When an intent is match dialogflow gives the values of the expression of the end-user as the parameter .
4.Responses : The defined text/speech form the response for the end-user's queries .

3.2.2 Module 2 - Entities

Each intent parameter has a type called the entity type , which dictates exactly how data from and end-user sentence is extracted .Entities are nouns that we use in normal world .Entities are defined by the prefix '@'.
The further classification of entities can be :
1. Entity Type :Entity type defines the information that the system wants to extract from user input .
2. Entity Entry : Entity entry provides a set of words or phrases that are considered equivalent . Each entity type there are many entity entries.
3.Entity Reference Values and Synonyms : For each entity entries the system defines a reference value and one or more synonyms .

Dialogflow provides some system entities . System entities are pre-build entities that facilitates handling popular common concepts .
Some of the system entities are :
@sys.date-time
@sys.date
@sys.time
@sys.number

3.2.3 Module 3 - Contexts

Dialogflow contexts are similar to natural language context . Using contexts , one can control the flow of a conversation .
Configuration of context for an intent is done by setting input and ouput contexts. whenever an intent is matched , any configured output contexts for that intent become active .
Dialogflow will only match intents that are configured with input contexts that match the currently active contexts .

Chapter 4

Project Implementation Part-1

4.1 Module-1
Dialogflow Console

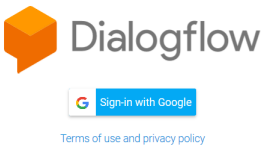

Figure 4.1: Dialogflow Registration

4.1.1 Prebuilt Agents in Dialogflow

Prebuilt agents are a collection of agents provided by Dialogflow for common use cases.These agents can be used to establish a base for building conversations for dining out, hotel booking, navigation, etc.
Prebuilt agents include intents and entities for their use cases, but you need to provide intent responses. Responses often depend on your specific scenario or need to be retrieved via fulfillment.

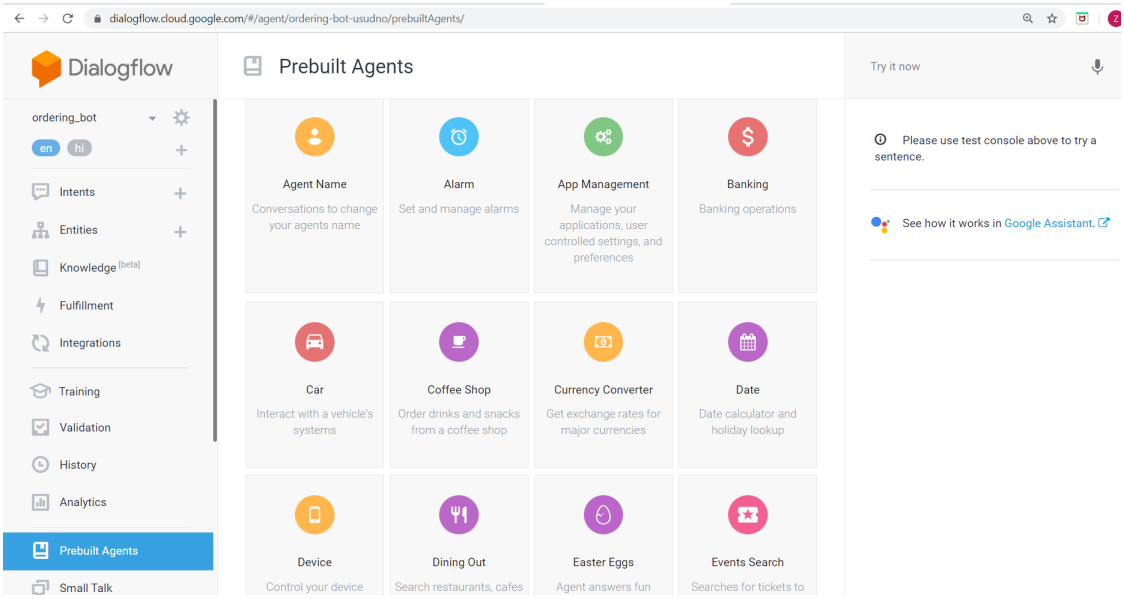

Figure 4.2: Prebuilt Agents

4.1.2 Intents of the Dialogflow Agent (ordering-bot)

Various intents for welcoming the customer , booking the pizza were created. "Small Talks" intents were imported from the prebuilt agents . Intents for changing the toppings , size of the pizza and confirming the bookings were also created.

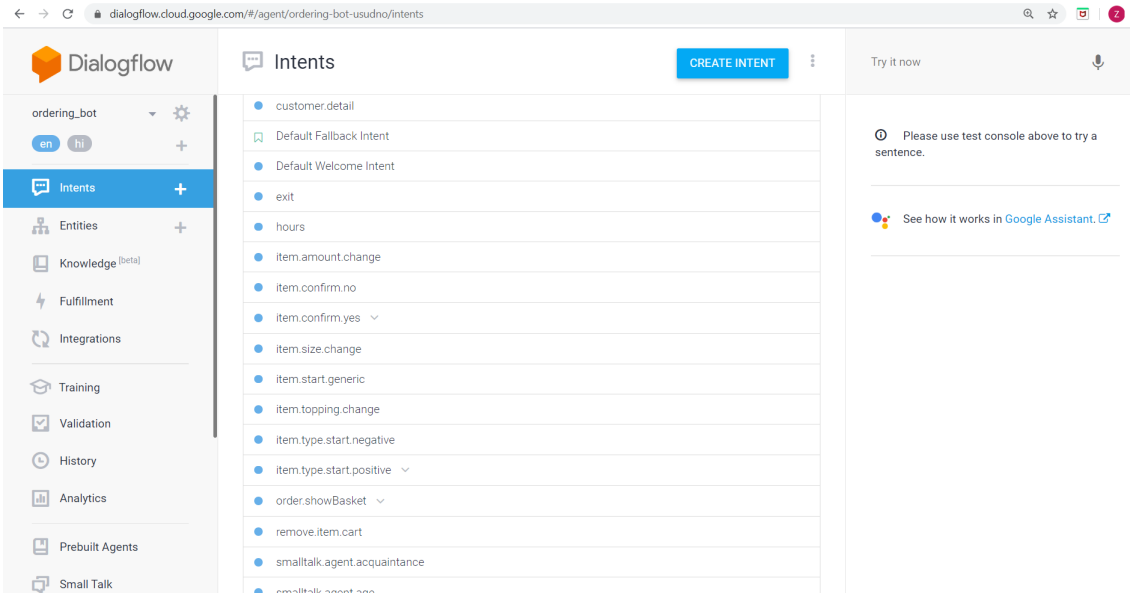

Figure 4.3: ordering-bot Intents

4.1.3 Intent for Booking Pizza

Intent with name "item.start.generic" was created which was provided with training phase coversation for what kind of pizza the user wants to booking such as " i want cheese pizza".

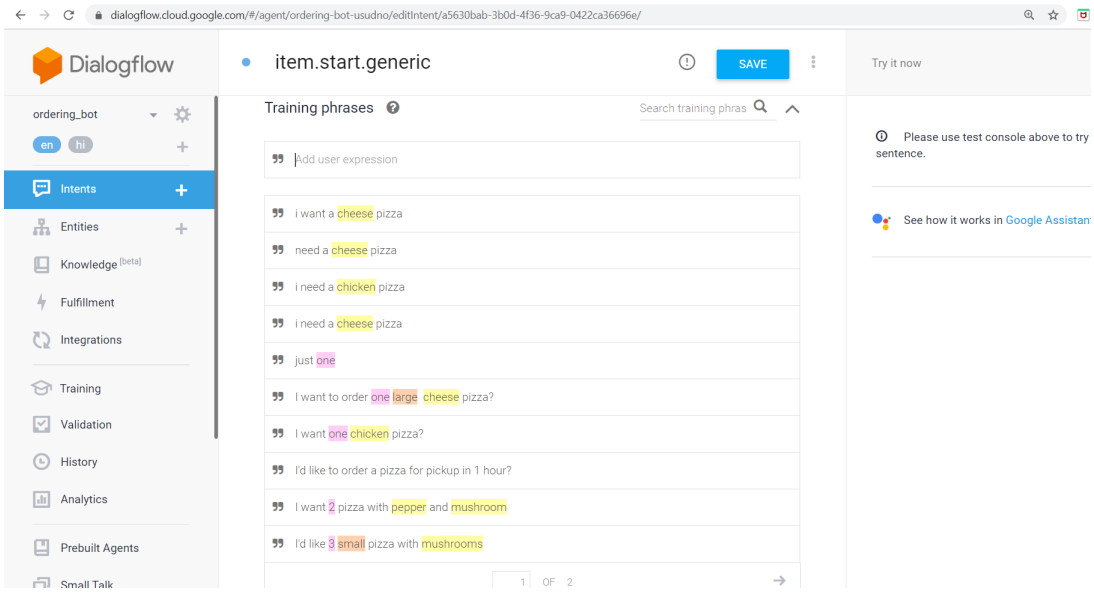

Figure 4.4: Generic Intent

4.1.4 Actions and Parameters passed to the Intent

Actions
The action field is a simple convenience field that assists in executing logic in your service. When building an agent, one can set this field to any text they find useful.
Parameters
When an intent is matched at runtime, Dialogflow provides the extracted values from the end-user expression as parameters. Each parameter has a type, called the entity type, which dictates exactly how the data is extracted. Unlike raw end-user input, parameters are structured data that can easily be used to perform some logic or generate responses.

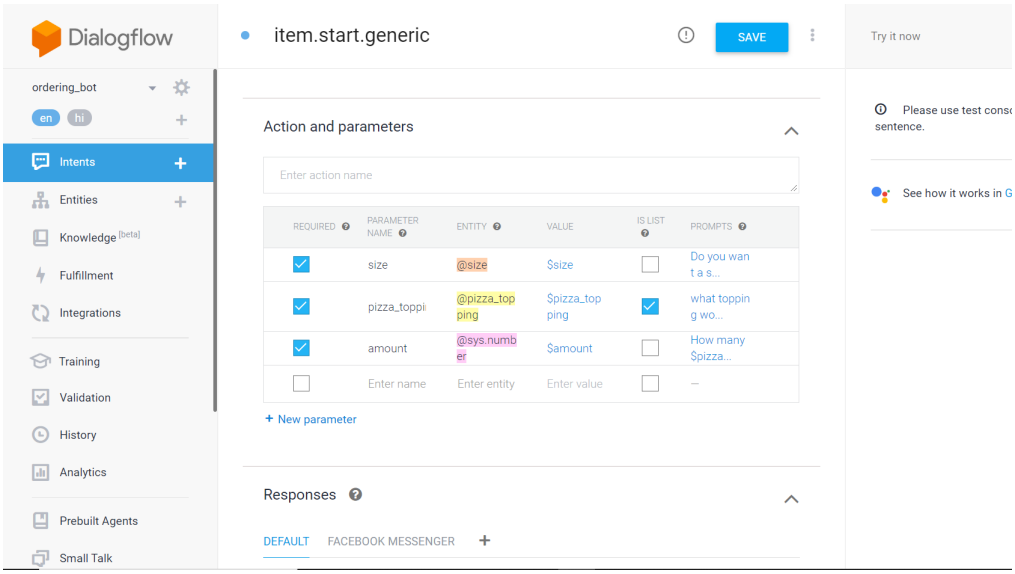

Figure 4.5: Actions and Parameters

4.1.5 Webhook enablement for the Intent

Webhook for each Intent must be enabled for the fulfilment to connect the particular intent.

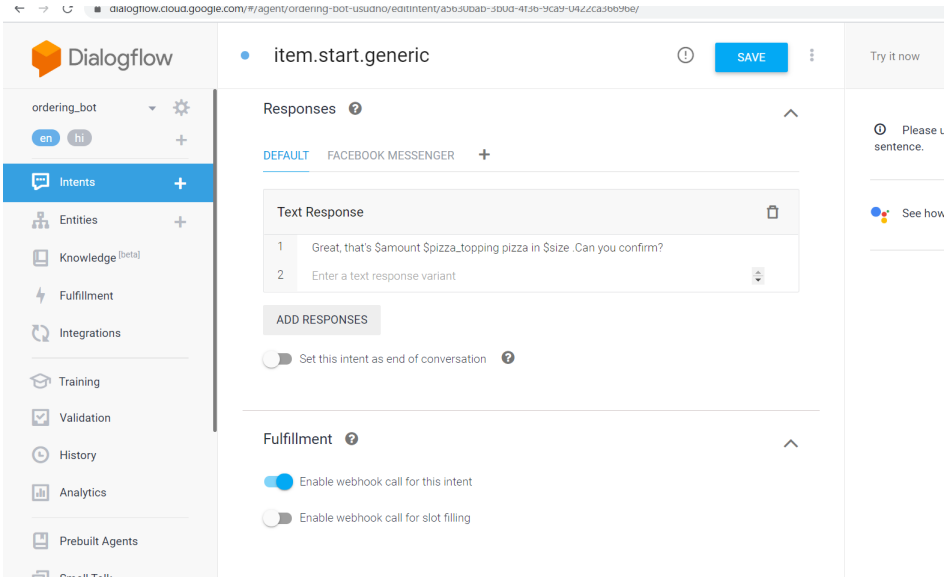

Figure 4.6: Webhook Enabled

4.1.6 Entities of the Project :

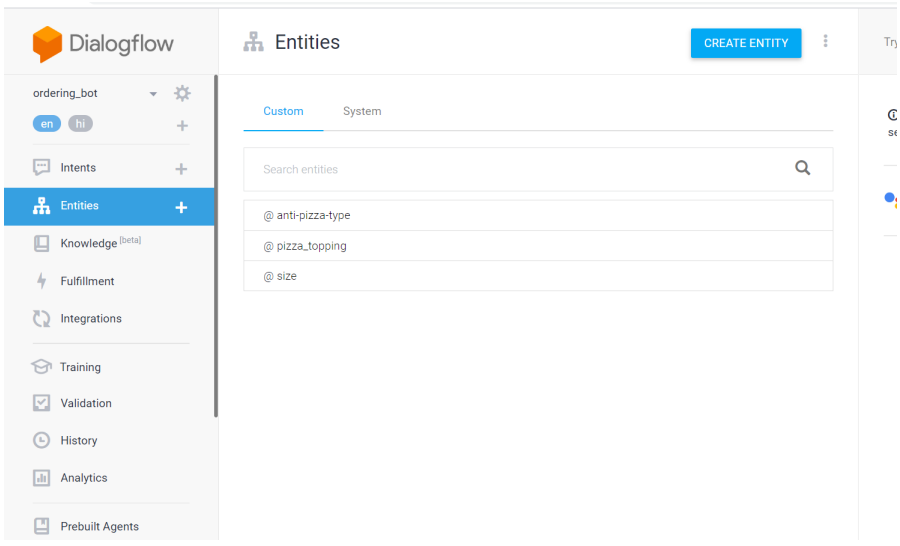

Figure 4.7: Created Entities

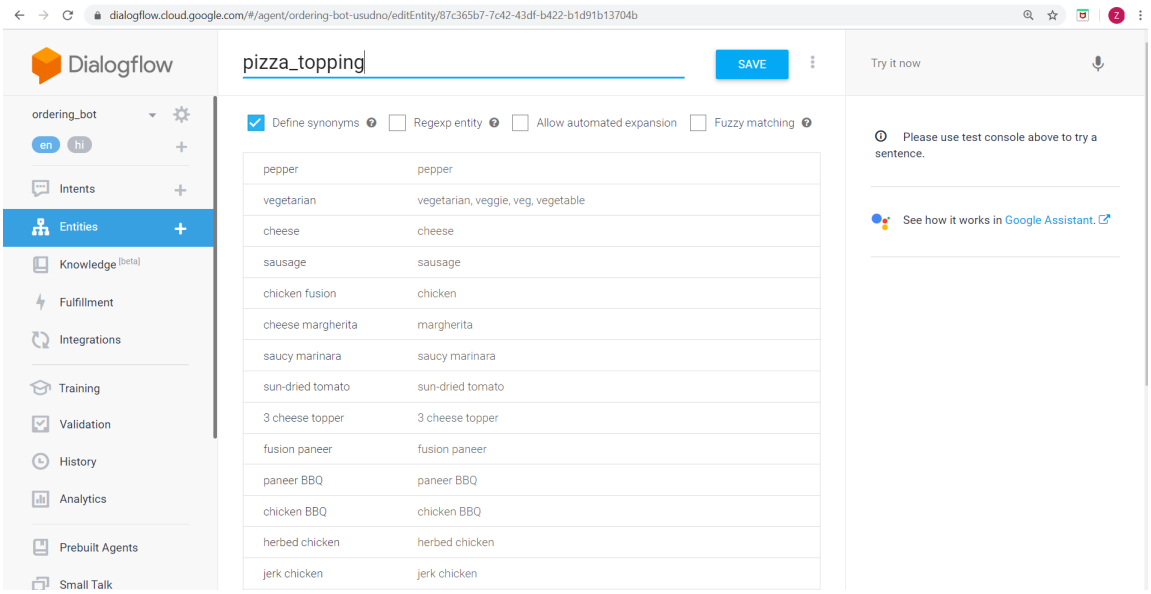

Figure 4.8: Pizza-topping Entity

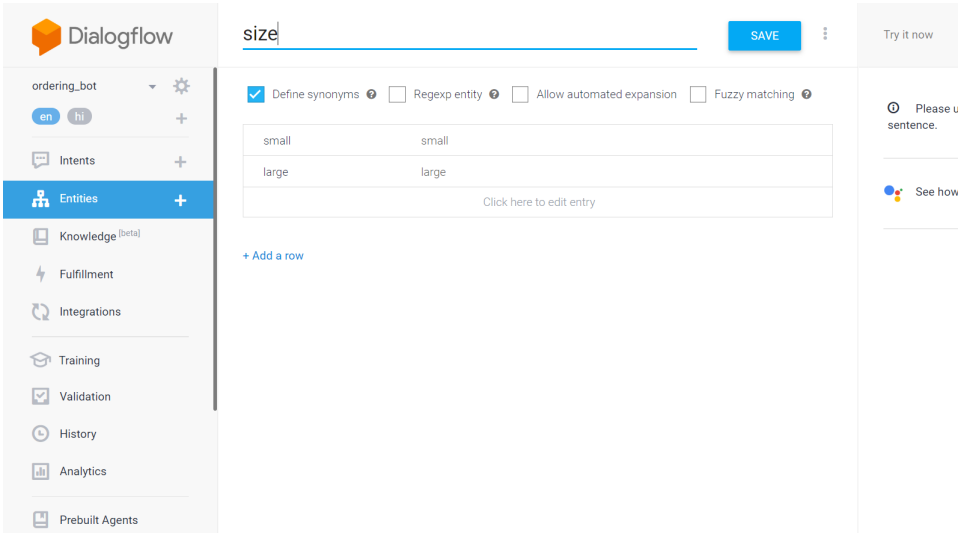

Figure 4.9: Size Entity

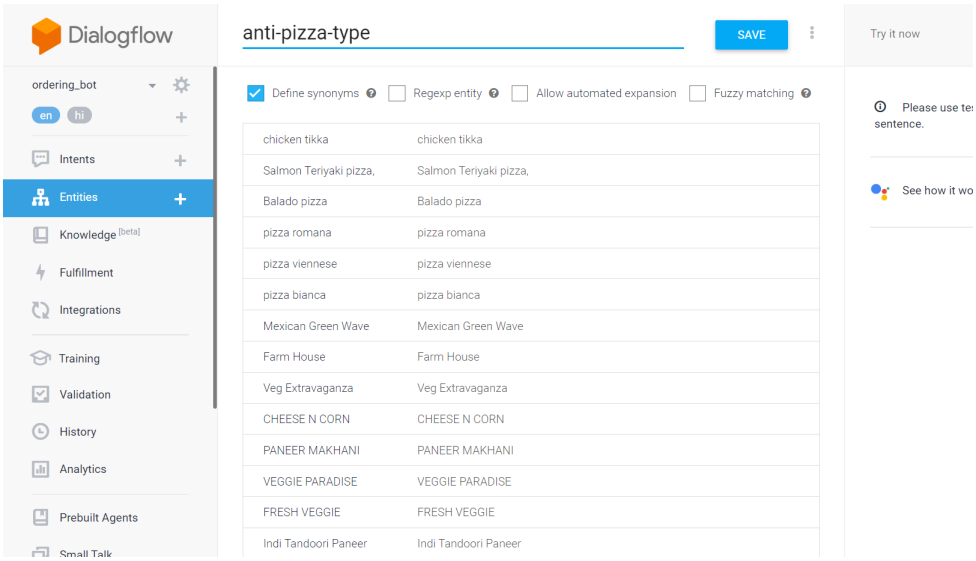

Figure 4.10: anti-pizza type Entity

4.1.7 Knoweldge Base of Dialogflow Intent

A knowledge base represents a collection of knowledge documents that you provide to Dialogflow. Your knowledge documents contain information that may be useful during conversations with end-users. Some Dialogflow features use knowledge bases when looking for a response to an end-user expression. This guide describes how to create and manage knowledge bases.

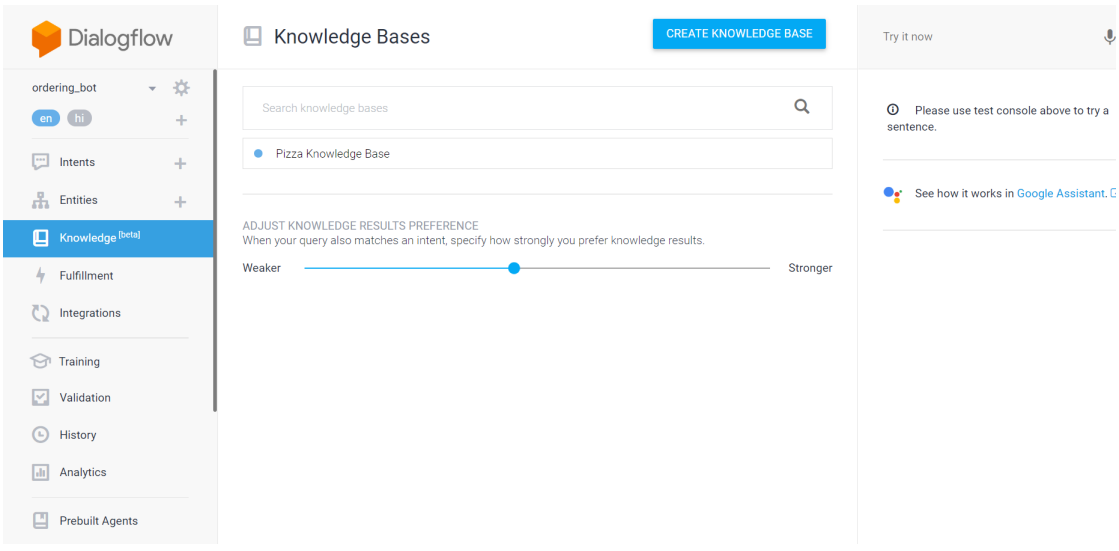

Figure 4.11: Knowledge Base1

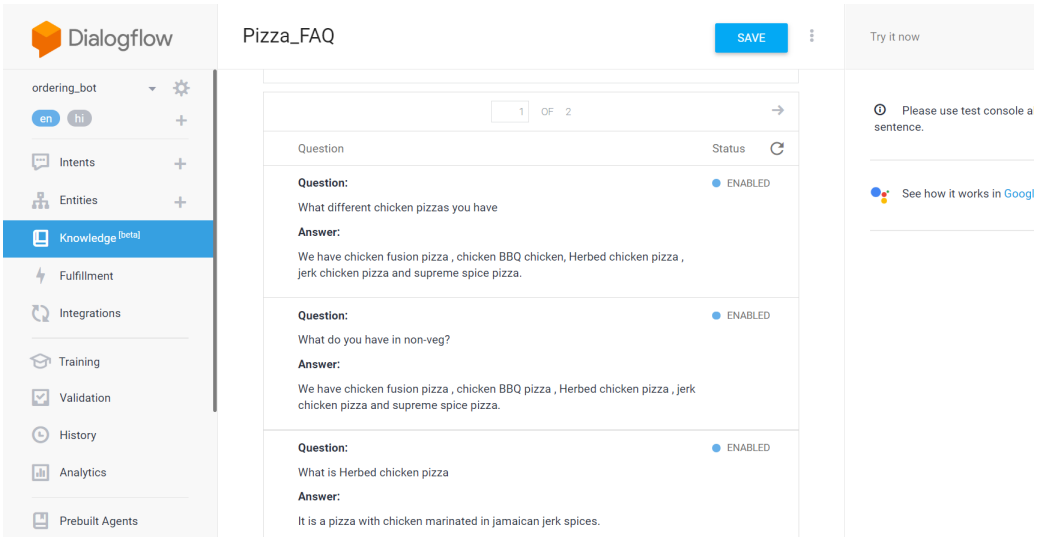

Figure 4.12: Knowledge Base2

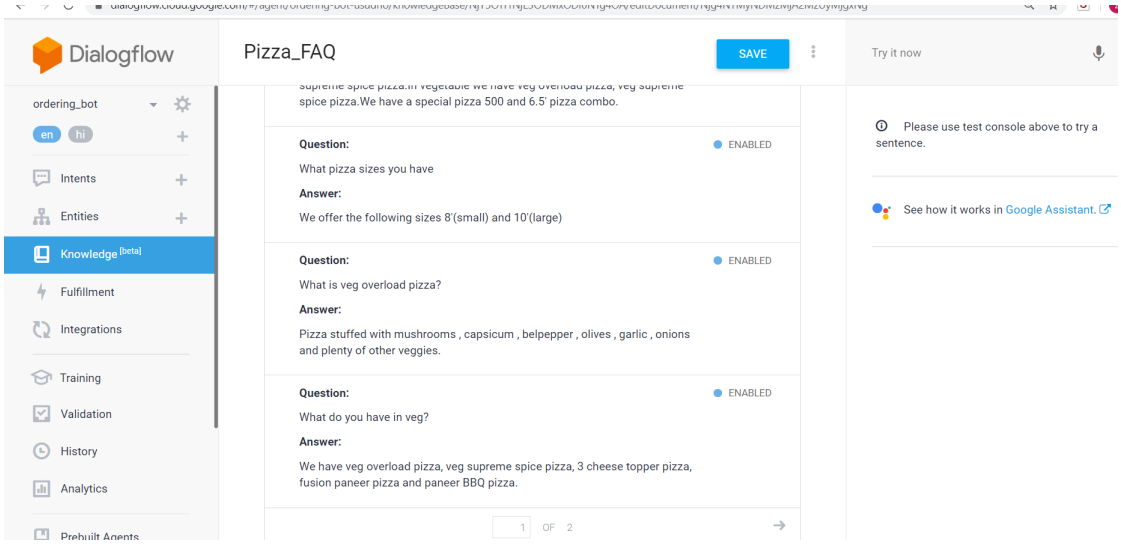

Figure 4.13: Knowledge Base3

4.1.8 Developer defined Training Set :

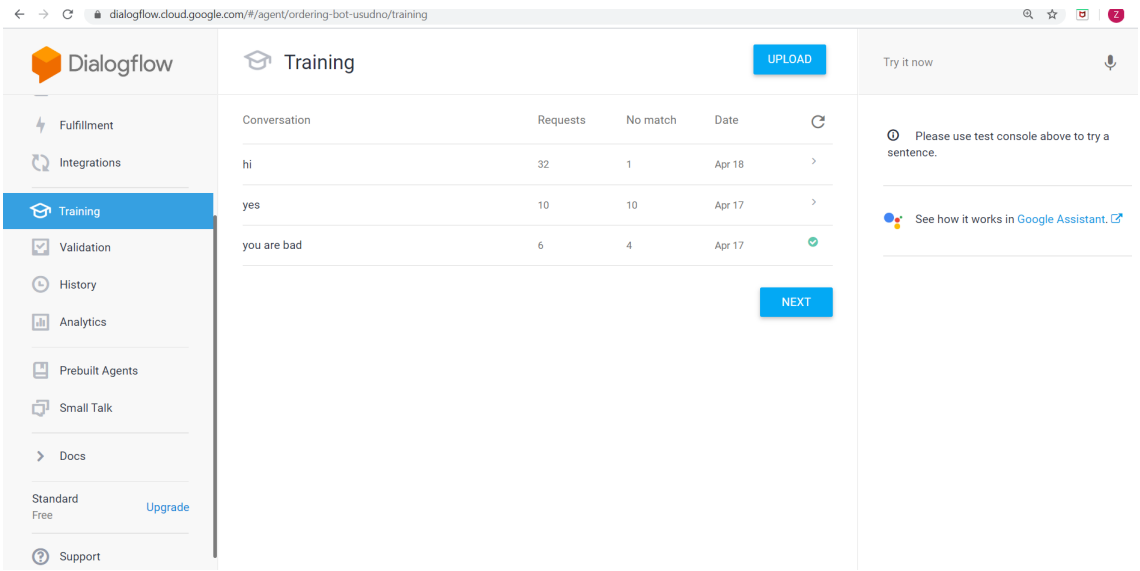

Figure 4.14: Training set1

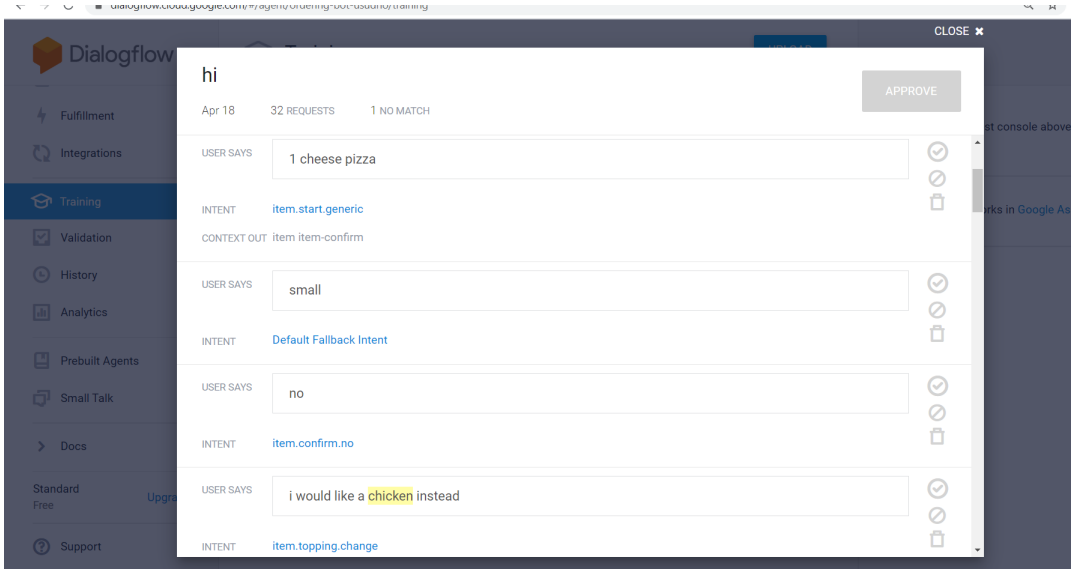

Figure 4.15: Training set2

4.1.9 History for keeping a check on Previous Training

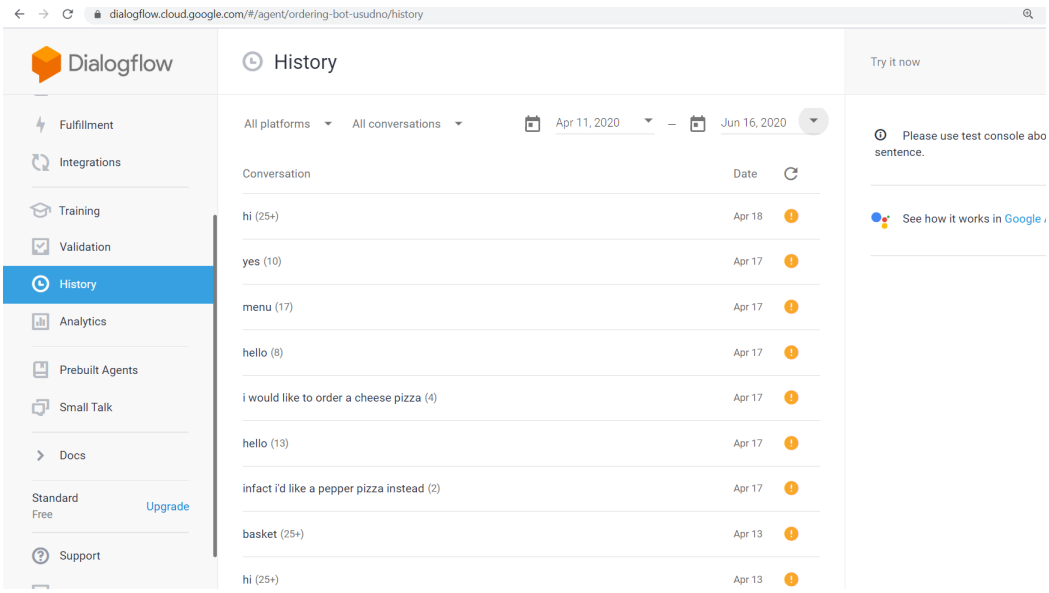

Figure 4.16: History

4.1.10 Fulfilments

When an intent with fulfillment enabled is matched, Dialogflow sends a request to your webhook service with information about the matched intent. Your system can perform any required actions and respond to Dialogflow with information for how to proceed. The following diagram shows the processing flow for fulfillment.

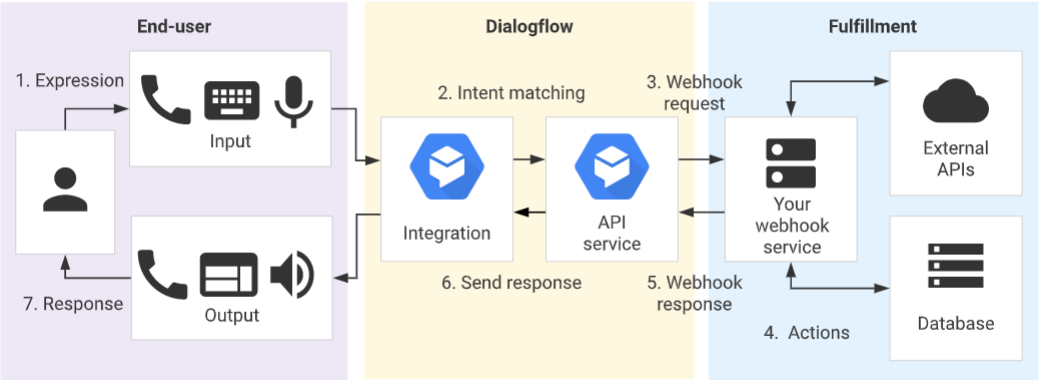

Figure 4.17: Fulfilment Flow

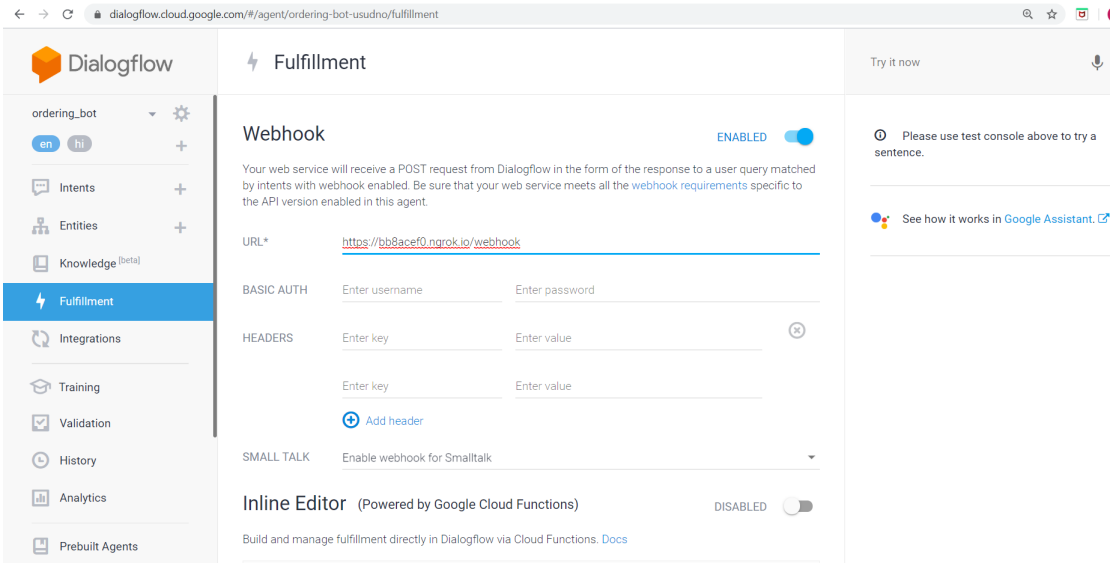

Figure 4.18: Fulfilment

4.2 Module-2
Webhook using Python Flask

Webhooks are "user-defined HTTP callbacks". A webhook is an HTTP request that is sent automatically whenever certain criteria are fulfilled. The request is done as an HTTP POST request. A webhook can be created in any server-side programming language like Python, PHP or Node.js.
We use webhooks when we want to create some useful chatbot with complex actions or we want dynamic responses. In Dialoflow, a webhook can be used to fetch data from your server whenever a certain intent having webhook enabled is invoked. The information from the intent is passed to the webhook service to receive the result.

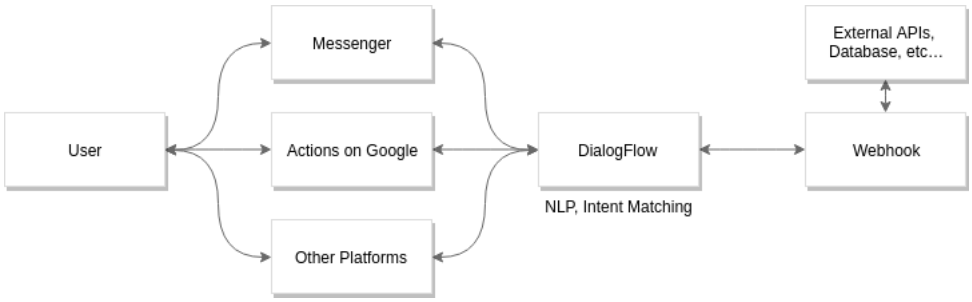

Figure 4.19: Interaction with Dialogflow and Webhook

4.2.1 Creating the UI for PizzaBot

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css">
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
    <title>Pizza ChatBot</title>
  </head>
  <body>
  <style>
      a {
        color: #8ebf42;
      }
    </style>
      <div class="container h-100">
        <div class="row align-items-center h-100">
            <div class="col-md-8 col-sm-12 mx-auto">
                <div class="h-100 justify-content-center">
                    <div class="chat-container" style="overflow: auto; max-height: 80vh">
                        <!-- chat messages -->
                        <div class="chat-message col-md-5 offset-md-7 bot-message">
                          Hello!  What pizza is your tummy craving for today?
                        </div>
                    </div>
                    <form id="target">
                       <input class="input" type="text" value="" placeholder="Enter message..." id="input_message"/>
                       <input type="submit" hidden>
                    </form>
                </div>
            </div>
        </div>
    </div>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"></script>
    <script src="{{ url_for('static', filename='custom.js')}}"></script>
  </body>
</html>
```

Figure 4.20: HTML code

```css
body {
  background: #fdb93c;
}

.chat-container {
  background: #FFFAF0;
  /*margin: 0px;*/
  padding: 0px;
  width: 500px;
  /*margin: 35px 0px;*/
  margin-left: 15%;
  margin-right: 15%;
}

.chat-message {
  padding: 6px;
  border-radius: 6px;
  margin-bottom: 3px;
}

.bot-message {
  background: yellow;
  max-width: 300px;
  color: white;
  margin-left: auto;
}

.human-message {
  background: red;
  max-width: 300px;
  color: white;
  margin: 13px 1px;
}

.input {
  width: 500px;
  /*margin: 35px 0px;*/
  margin-left: 15%;
  margin-right: 15%;
```

Figure 4.21: CSS code

Chapter 5

Project Implementation Part-2

5.1 Module-1 Localhost Codes

5.1.1 index.py for connecting dialogflow and the frontend webpage

```
from flask import Flask, request, jsonify, render_template
import os
import dialogflow
import requests
import json
import pusher

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

# run Flask app
if __name__ == "__main__":
    app.run()

@app.route('/webhook', methods=['POST'])
def webhook():
    data = request.get_json(silent=True)


def detect_intent_texts(project_id, session_id, text, language_code):
    session_client = dialogflow.SessionsClient()
    session = session_client.session_path(project_id, session_id)

    if text:
        text_input = dialogflow.types.TextInput(
            text=text, language_code=language_code)
        query_input = dialogflow.types.QueryInput(text=text_input)
        response = session_client.detect_intent(
            session=session, query_input=query_input)
        return response.query_result.fulfillment_text

@app.route('/send_message', methods=['POST'])
def send_message():
    message = request.form['message']
    project_id = os.getenv('DIALOGFLOW_PROJECT_ID')
    fulfillment_text = detect_intent_texts(project_id, "unique", message, 'en')
    response_text = { "message":  fulfillment_text }
    return jsonify(response_text)
```

Figure 5.1: index.py File

5.1.2 Creating virtual Environment and enabling Flask App

The Commands below will instruct flask to use index.py as the main entry file and startup the project in development mode.
The web application is running on port 5000, we could see our app with the welcome message at http://127.0.0.1:5000/.

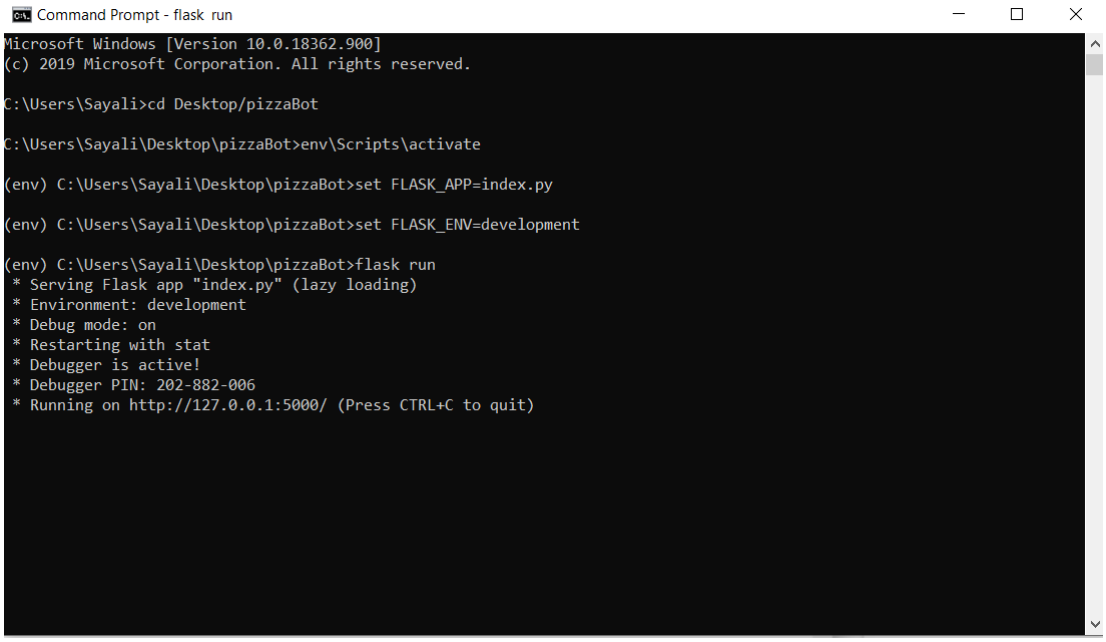

Figure 5.2: Flask App Initiation

5.1.3 jQuery code

The below code will listen for when the user submits the message, then call the submit-message function to submit the input text from the user.

```
function submit_message(message) {
        $.post( "/send_message", {message: message}, handle_response);

        function handle_response(data) {
          // append the bot repsonse to the div
          $('.chat-container').append(`
                <div class="chat-message col-md-5 offset-md-7 bot-message">
                    ${data.message}
                </div>
            `)
          // remove the loading indicator
          $( "#loading" ).remove();
        }
    }

$('#target').on('submit', function(e){
        e.preventDefault();
        const input_message = $('#input_message').val()
        // return if the user does not enter any text
        if (!input_message) {
          return
        }

        $('.chat-container').append(`
            <div class="chat-message col-md-5 human-message">
                ${input_message}
            </div>
        `)

        // loading
        $('.chat-container').append(`
            <div class="chat-message text-center col-md-2 offset-md-10 bot-message" id="loading">
                <b>...</b>
            </div>
        `)

        // clear the text input
        $('#input_message').val('')

        // send the message
        submit_message(input_message)
    });
```

Figure 5.3: custom.js File

5.2 Module-2
Google Cloud Platform (GCP)

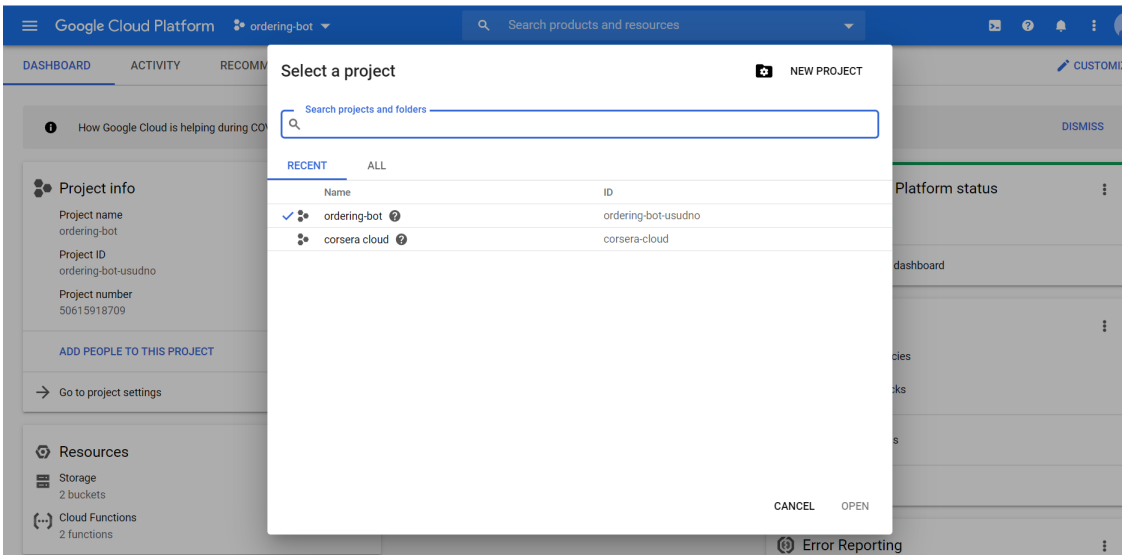

Figure 5.4: GCP Projects

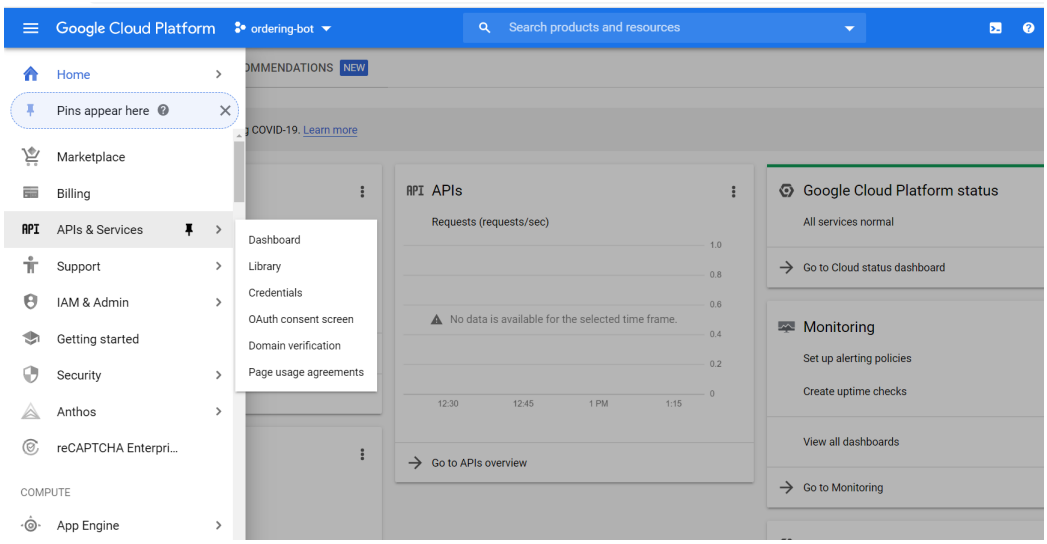

Figure 5.5: GCP API's and Services

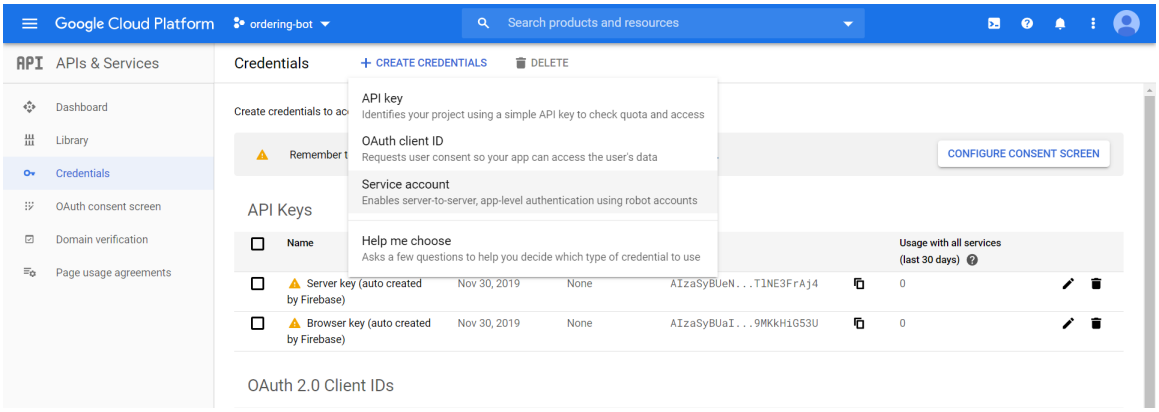

Figure 5.6: Creating Credentials for Service Account

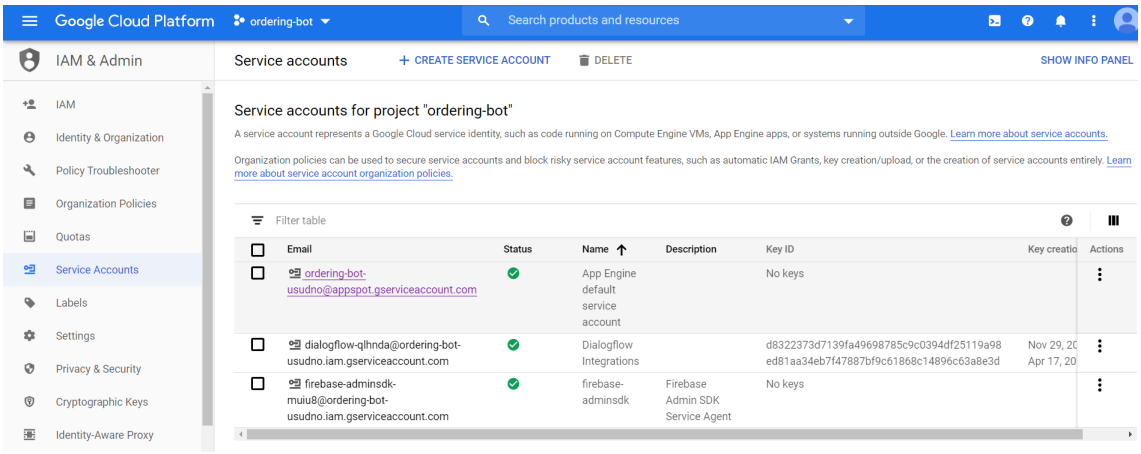

Figure 5.7: Service Account for respective Dialogflow Agent

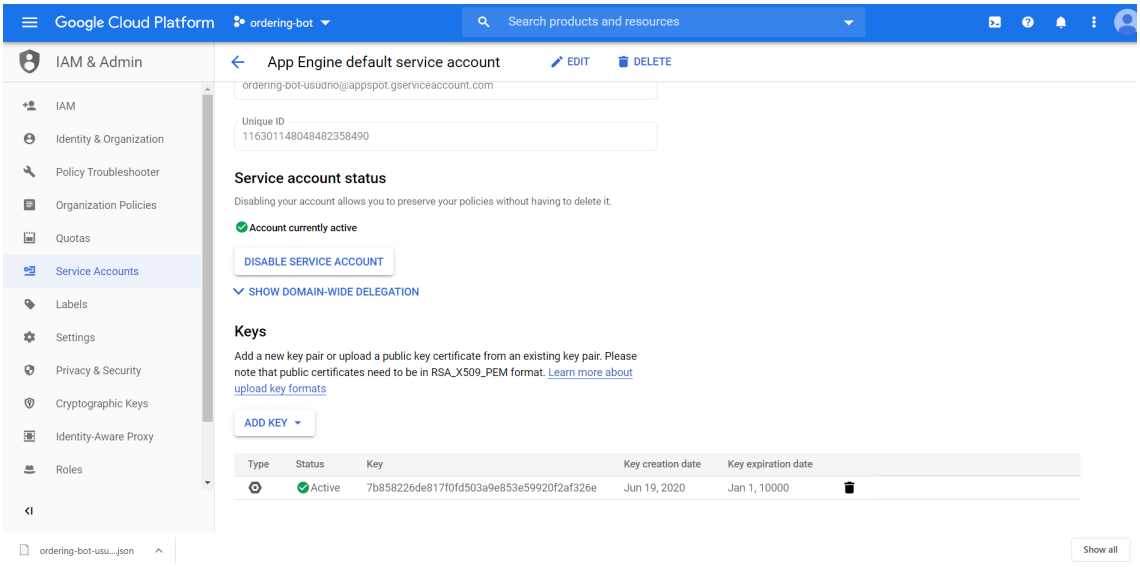

Figure 5.8: API Key

5.3 Module-3
Running Webhook using Ngrok

DialogFlow doesn't support localhost. To integrate it as webhook for DialogFlow, we need to make it live.
Ngrok establishes secure tunnels from a public endpoint such as the internet to a locally running service while capturing all traffic for detailed inspection and replay.

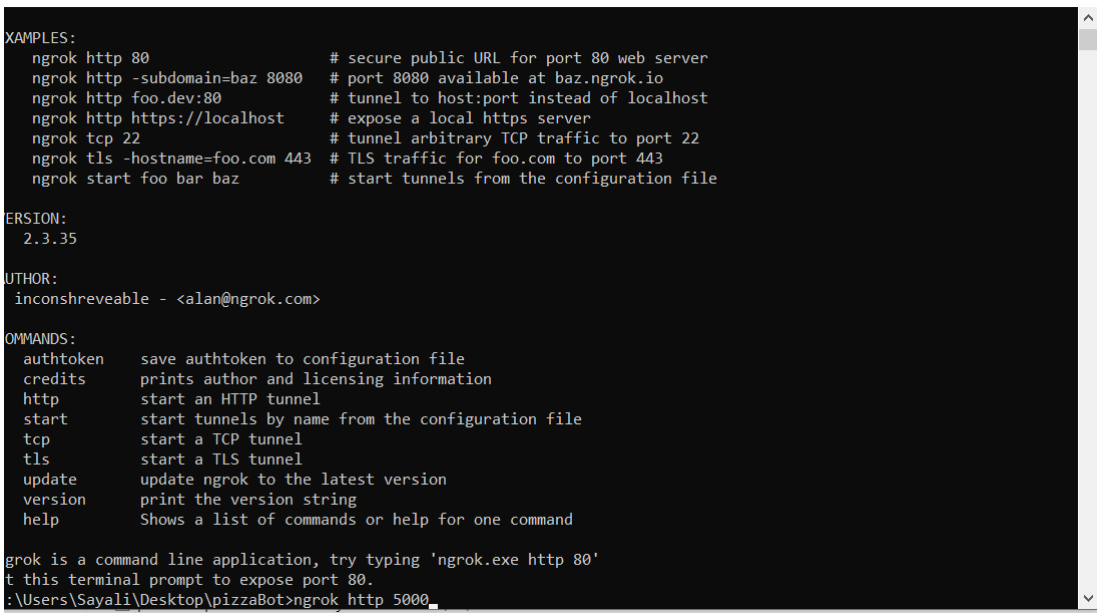

Figure 5.9: Running Ngrok

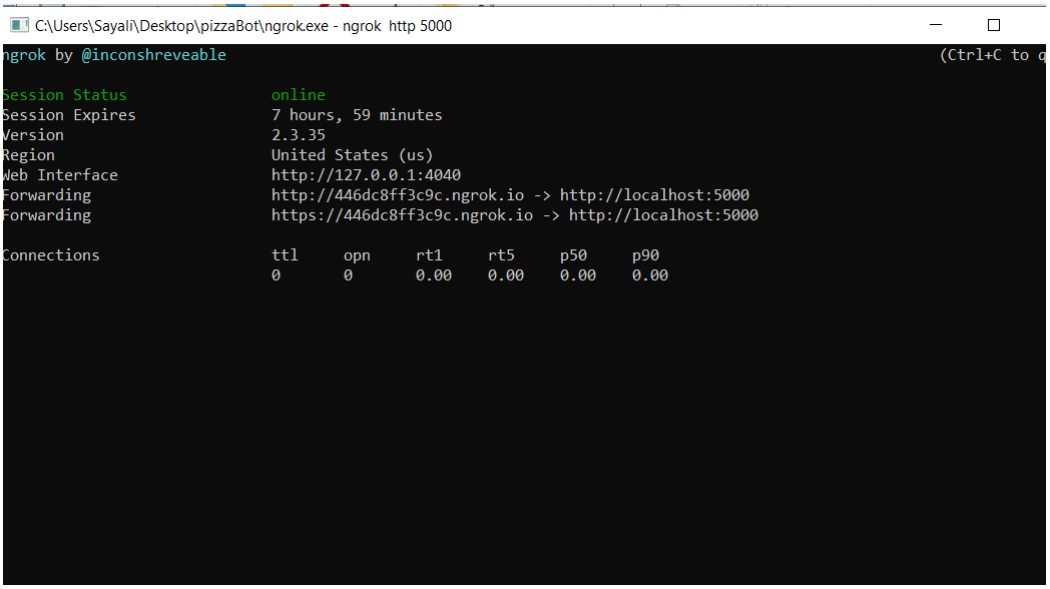

Figure 5.10: Http link generated by Ngrok

Chapter 6

Result

The following is the output on the User Interface(UI).

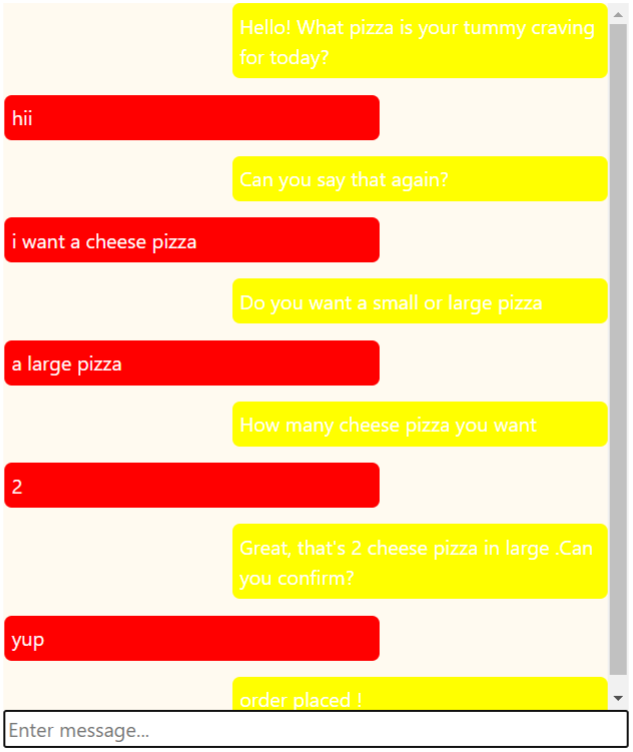

Figure 6.1: Output1

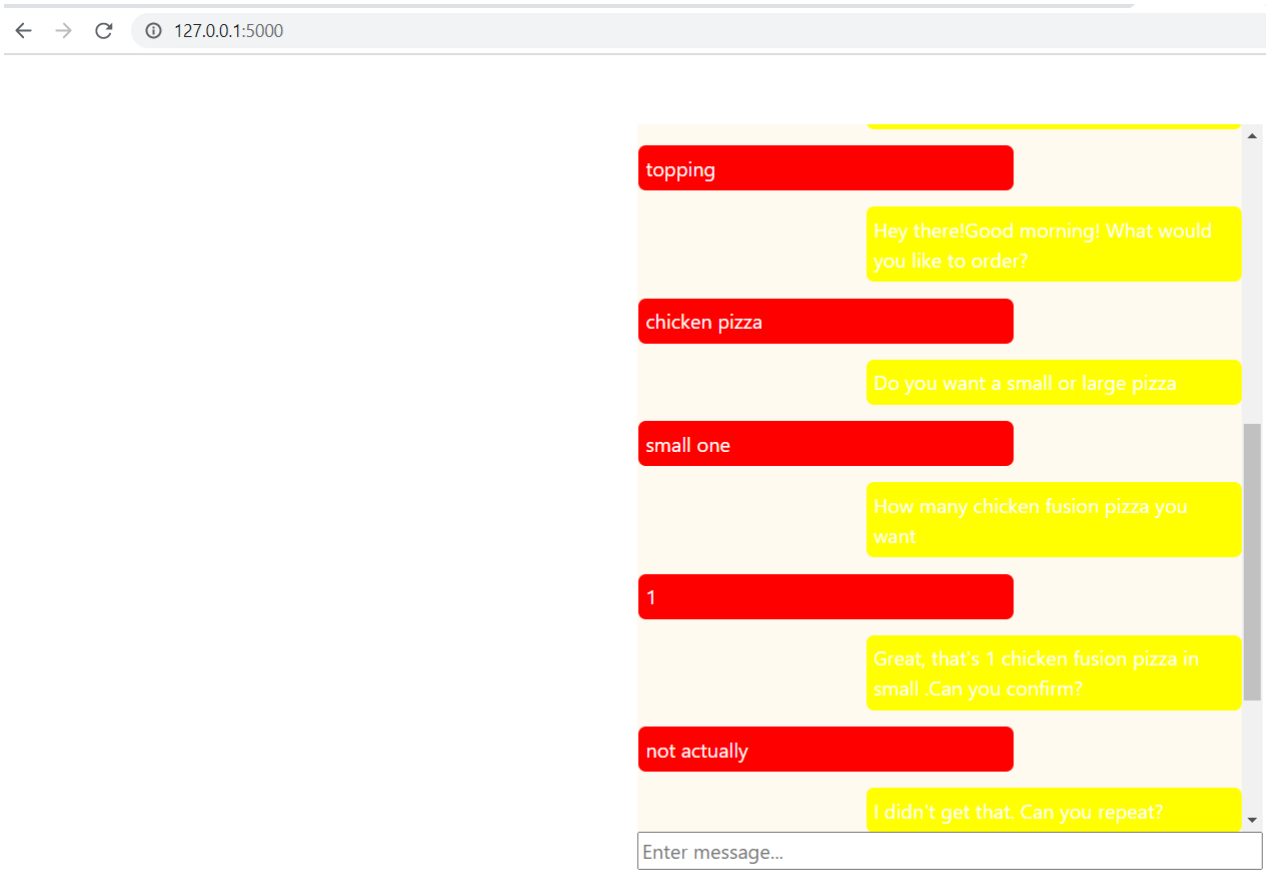

Figure 6.2: Output2

Chapter 7

Conclusion and Future Scope

Thus,a chatbot is generally used for a friendly interaction with the user and to solve the queries of the user in no time.
A usable system will be designed , developed and deployed to a web application .
The customer can search , chose , apply coupons on a particular pizza ordering .
Thus this system will be helpful an quick access chat section for pizza ordering from a given web app.
The Task of searching favourite pizza from various GUI interaction has been made easy with the help of chatbot. The user can order pizza with few chat messages with the chatbot .
Speech to text and vice versa can be added to this chatbot to make the bot more user-friendly and also a user who is unable to write/read can order pizza without and complications of choosing pizza from the GUI dropdown menu .
Big data Analytics can be carried out on customer's previous orders and feedback to improve the quality of services if required and to make the customer's favourite pizzas available at time ordering the pizza .

Bibliography

[1] B. R. Ranoliya, N. Raghuwanshi and S. Singh, "Chatbot for university related FAQs," in Advance in Computing, Communications and Informatics (ICACCI), 2017.

[2] Chih-Wei Lee, Yau-Shian Wang, Tsung-Yuan Hsu, Kuan-Yu Chen, Hung-Yi Lee, Lin-shan Lee "scalable sentiment for sequence-to-sequence chatbot response with performance analysis" [ICASSP],2018.

[3] N. Thomas, "An e-business chatbot using aiml and lsa," in Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on. IEEE, 2016, pp. 2740–2742.

[4] B. Setiaji and F.W. Wibowo,"Chatbot using a knowlegde in Database: Human-to-Machine conversation Modelling" in Intelligent systems, modelling and simulation(ISMS),2016 7th International Conference on , Bangkok, Thailand .

[5] Rashid Khan and Anik Das. Build Better Chatbots. 2017. 120 pp.

[6] Hoang Nguyen and Jon Patrick. "Text Mining in Clinical Domain: Dealing with Noise". In: International Conference on Knowledge Discovery and Data Mining. San Francisco, CA, USA, 2016, pp. 549–558.

Appendices

Appendix-A: Registration on dialogflow console

1. Go to the site ' https://console.dialogflow.com '

2. Register with google account to make use of free services.

3. Logout from the account .

Acknowledgement

We have great pleasure in presenting the report on **Chatbot-Pizza Ordering.** We take this opportunity to express our sincere thanks towards our guide **Prof. Archana Kotangale** Department of Computer Engineering, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin Malave** Head of Department,Computer Engineering, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol Kalugade** BE project coordinator, Department of Computer Engineering, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1: SAYALI KAMBLE
Student ID1: 16102039

Student Name2: ZAHID KHAN
Student ID2: 16102051

Student Name3: VISHAL JAIN
Student ID3: 16102044