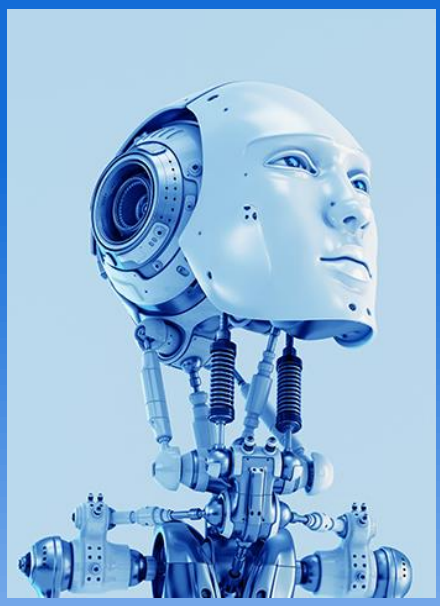
Master of Technology (IS)



Pepper Project Group

CAO LIANG	A0012884E
GENG LIANGYU	A0195278M
HAN DONGCHOU FRANCIS	A0195414A
ONG BOON PING	A0195172B
TAN CHIN GEE	A0195296M

ISS Chatbot System

ISS Chatbot System

Project Report

Contents

Executive Summary

In this project, we are tasked to build a chatbot system for the ISS website. The purpose of this chatbot would be to answer enquiries related to ISS programs, courses, and related information. This project serves to leverage the techniques learned in the Cognitive Systems course.

We employed the Dialogflow agent for this implementation together with a Python program and its associated libraries. This combination provides the capability for two levels of response, first is a fulfillment from the Python program, and second is Dialogflow.

This project gives us an opportunity to convert our learning into practical use. In this case, providing a way for users to interactively use the ISS website.

Business Problem Background

When we first got together to discuss how we could execute such a project, there were several problems that we looked at:

1) Where do we find the knowledge base for information related to ISS courses?
2) How do we put together a GUI for the chat?
3) How do we capture all the Intents?
4) What are the different variations of Intents we will cover?
5) How do we deal with the Entities?
6) What are the things we'll do in Fulfillment?
7) How do we handle queries not found in the list of configured Intents?

Project Objectives & Success Measurement

The objective of this project is to provide an operational Chatbot which is capable of the following:

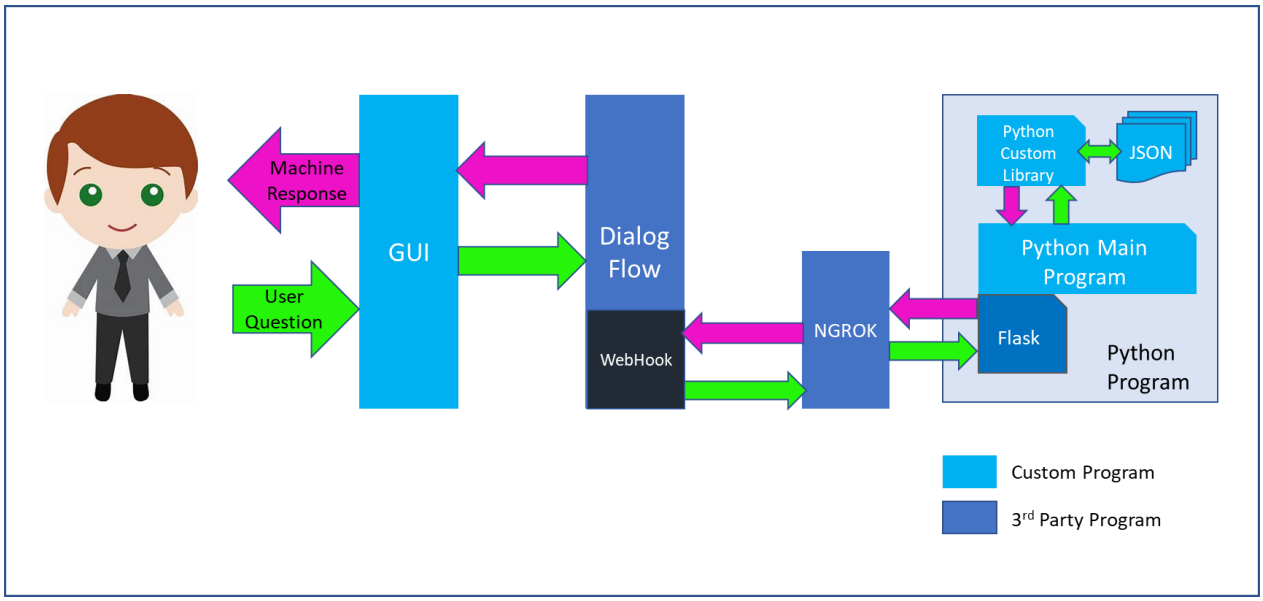
1) Take in user questions in a chat GUI
2) Process the question to understand the intent
3) Evaluate the Entities of the question
4) Generate a response through Fulfilment
5) Otherwise use the Knowledgebase
6) Otherwise use the default response in Dialogflow

Given that this is a 2 week project, we are also mindful that the scope of questions that this Dialogflow can cover is also limited. We have come up with a list of flowcharts to show the type of test scenarios.

Project Solution

Architecture

This application leverages the combination of DialogFlow with Python program and its associated libraries. The libraries is mainly Flask. We also used BeautifulSoup for scraping web site information. We utilized Python to do Fulfillment with the DialogFlow. DialogFlow does the Intent Classification and Slot Detection before reaching the Python program. The Python program reads the Intent of the question and draws out the appropriate response from the JSON files. In addition, we developed a simple web frontend for the DialogFlow GUI.



Project Solution

Sources of Information

JSON Files

Considerable amount of effort has been put into the JSON files to create a repository of information. These includes some web pages from the ISS website as well as piecemeal information which we manually entered into the JSON files.

To read and extract information from the web pages, we used the BeautifulSoup library. Challenges encountered here were mainly due to the fact that we have to take care of the various HTML tags in the pages.

A Python library (datautil.py) is created to interface with the JSON files. All reading and extractions are done via this library.

KnowledgeBase

There is an FAQ in the ISS website which we have extracted into a CSV format and uploaded into the KnowledgeBase of the DialogFlow. In the process of creating this KnowledgeBase we have selected FAQ as the option.

Default Response

Some default responses are set up for each Intent in case the Python program is not able to come up with a valid response.

Project Solution

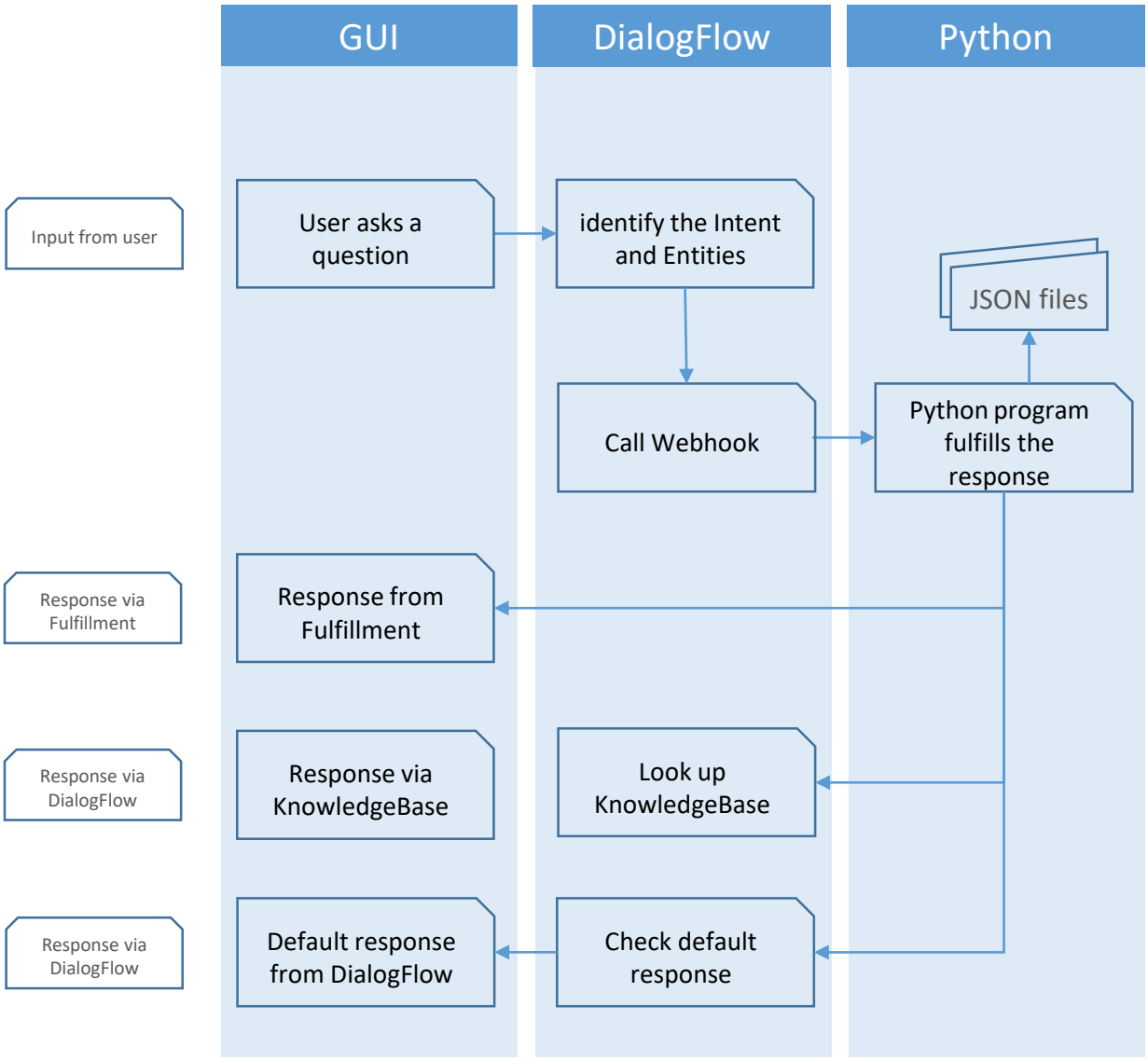
Handling Fallback Intent

A Fallback Intent can be found in the DialogFlow by default. We are calling Webhook in case the Fallback Intent is triggered. The Webhook channels the question to the Python program which will try to process it further and detect any Intent present. If a detection happens, then the Python program will try to retrieve a response through the JSON files. If not, it will return it back to the DialogFlow for some default response.

Project Solution

Process Flow

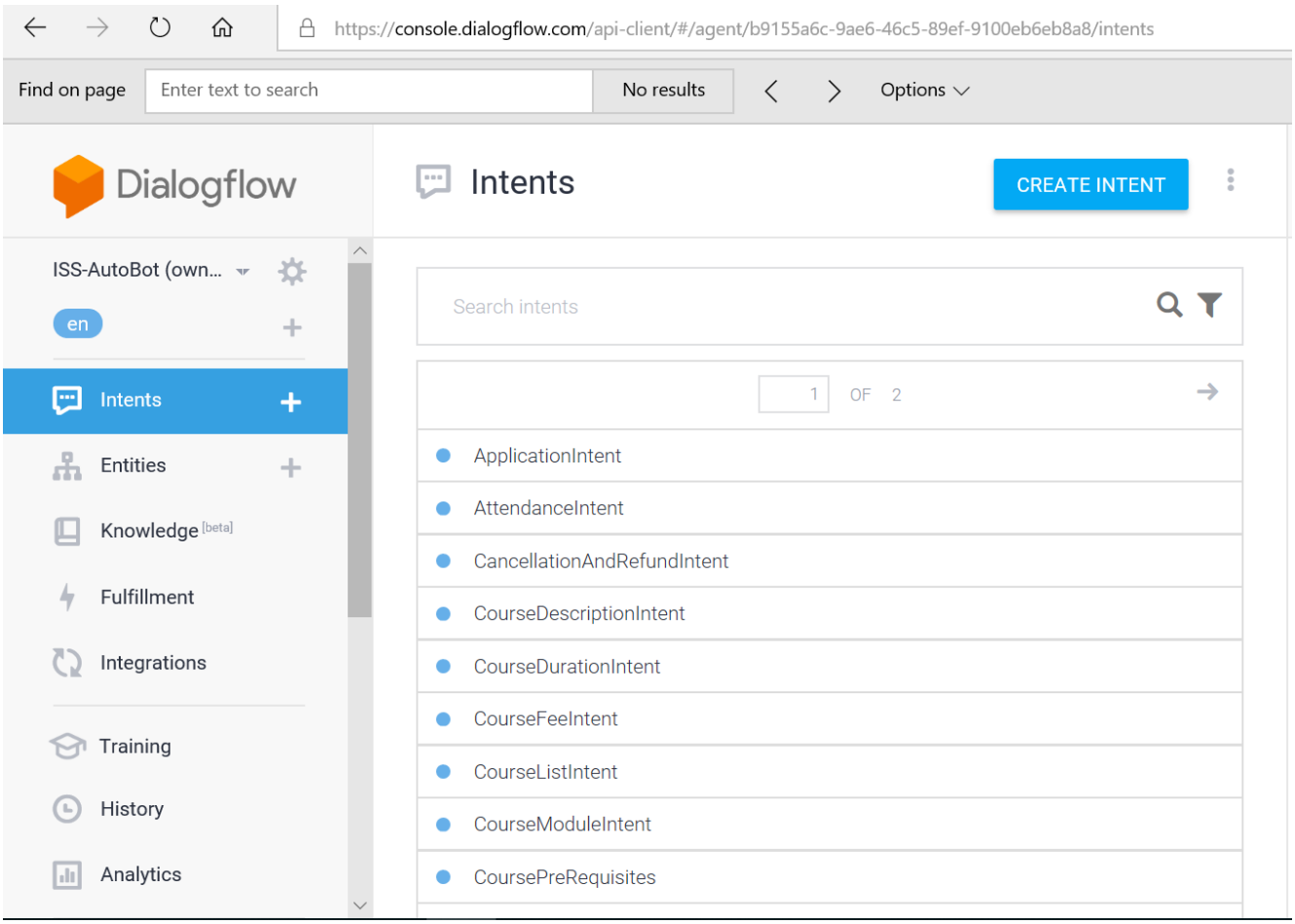
1) Users asks a question
2) DialogFlow processes the question to assess the Intent and Entities
3) Go to Python program
4) Python program picks up the Intent and Entities and then look up a JSON file for response
5) Response generated thru Fulfillment via Python
6) If there's no response from Python then look up Knowledgebase from DialogFlow
7) If still no response then the Default Response from DialogFlow will be triggered



Project Implementation

DialogFlow Setup

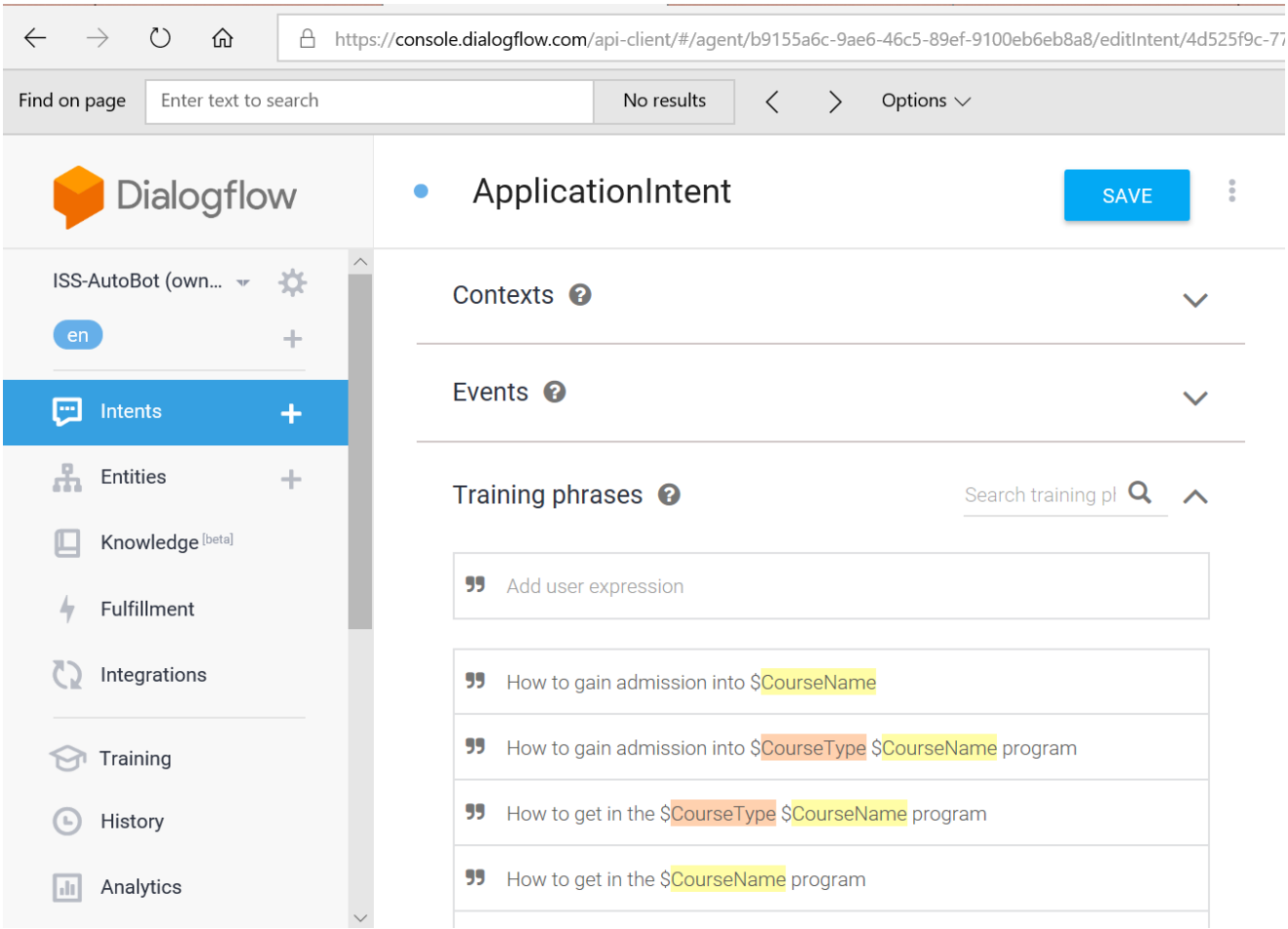
We set up a list of Intents to cover a broad scope of questions.



Project Implementation

DialogFlow Setup

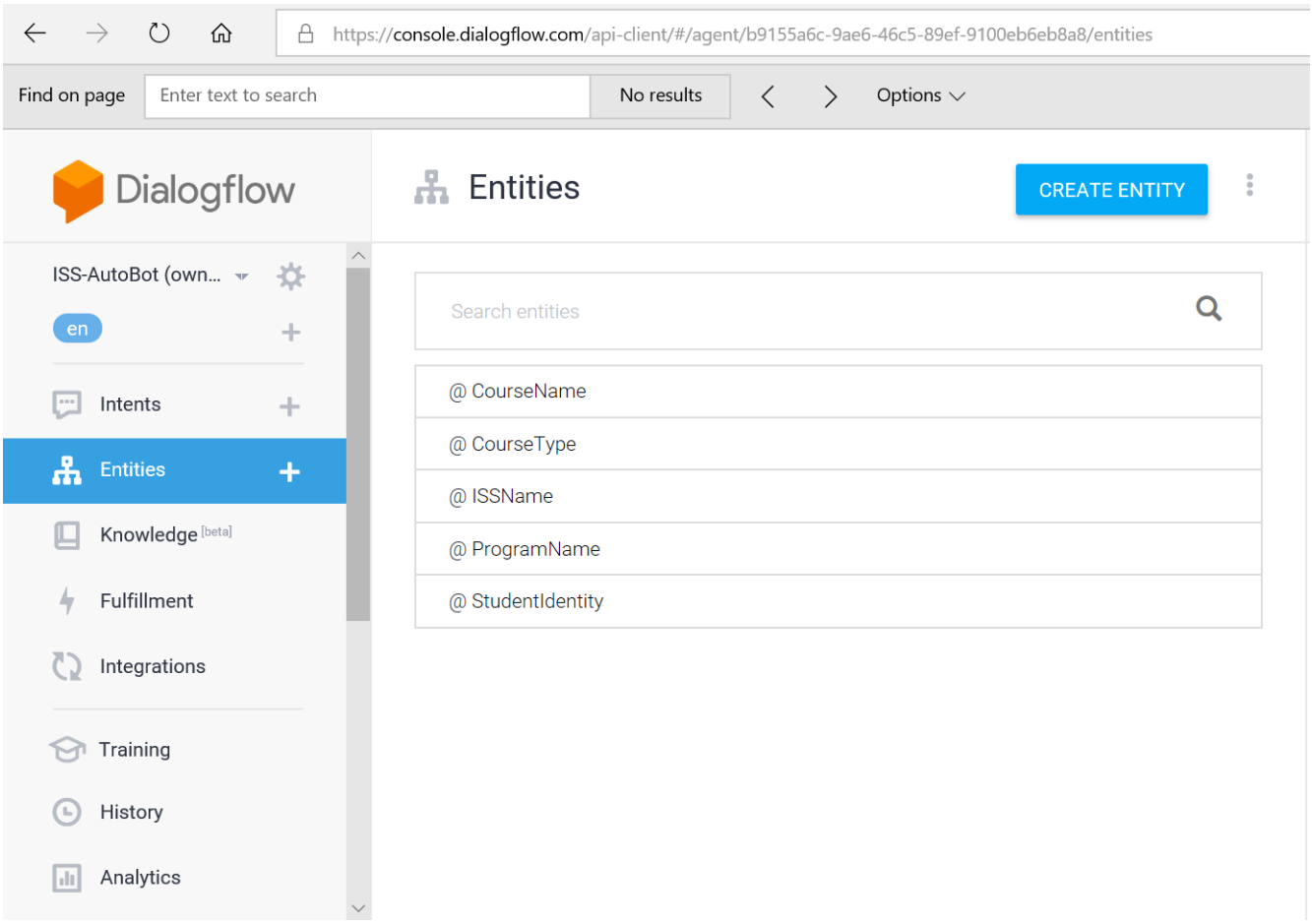
For each Intent, we set up some Training Phrases in order to cover the different variations of questions. These Training Phrases will have some defined Entities embedded in order for DialogFlow to pick up them up together with the Intent.



Project Implementation

DialogFlow Setup

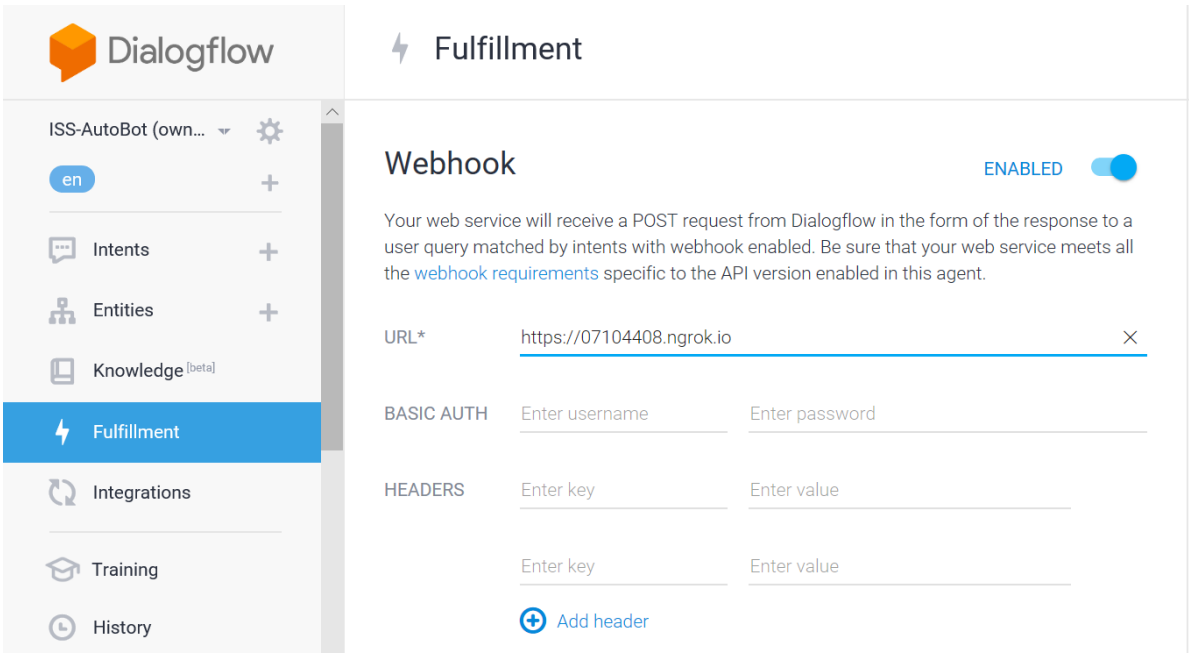
The Entities are defined here.



Project Implementation

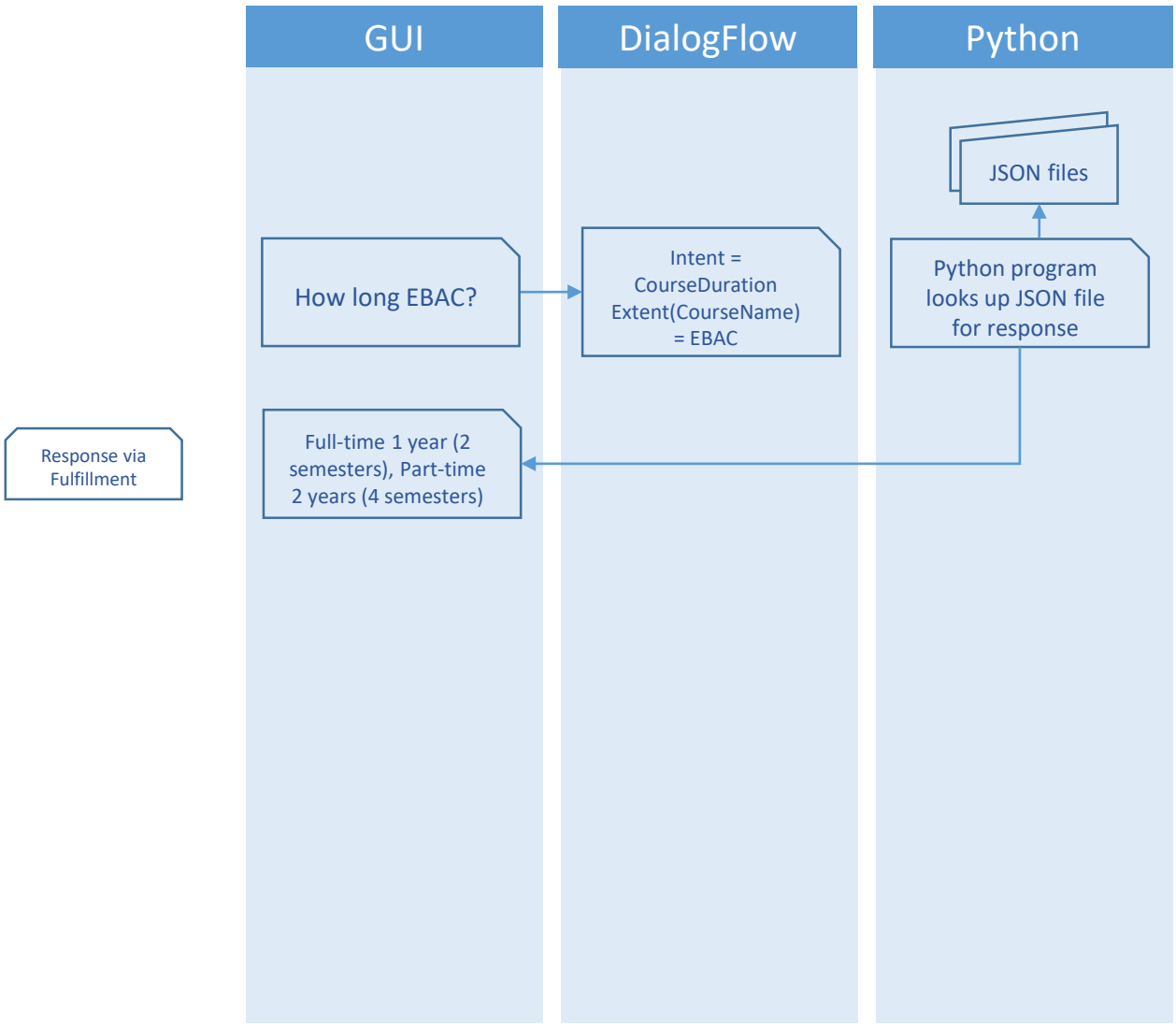
DialogFlow Setup

The Entities are defined here.



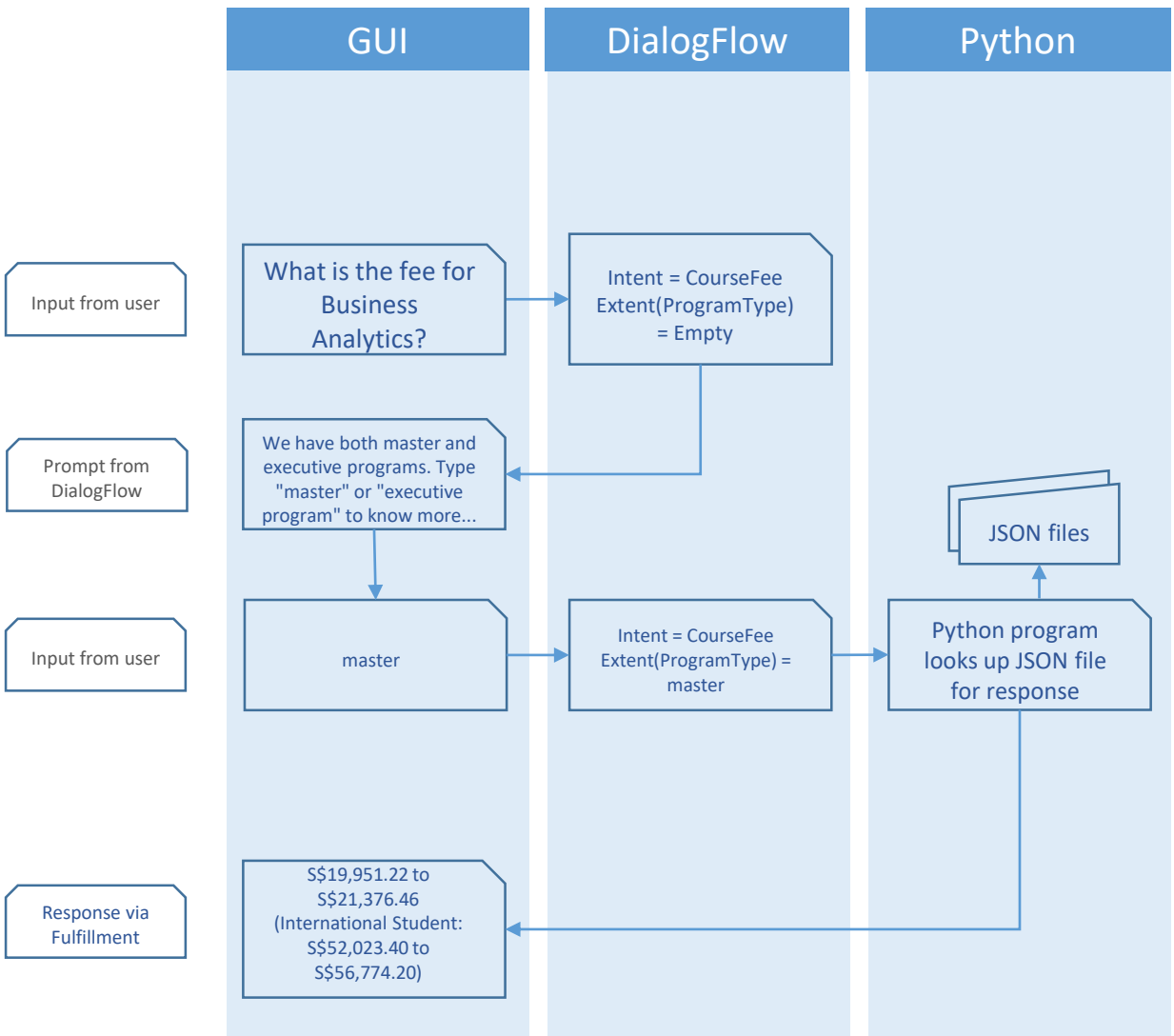
Project Performance & Validation

Scenario 1 – Simple Query



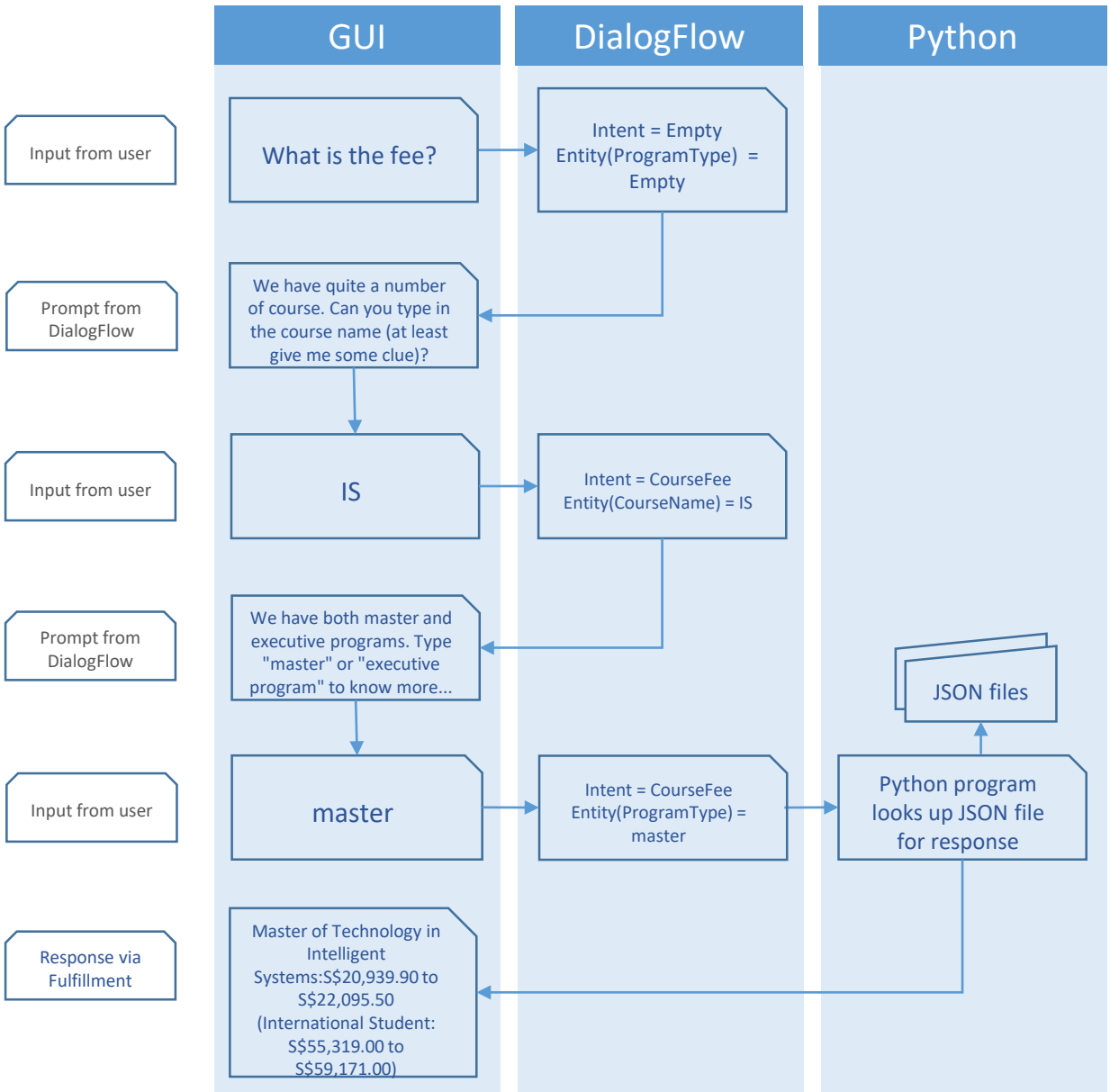
Project Performance & Validation

Scenario 2 – Missing 1 Slot (Program Type Entity)



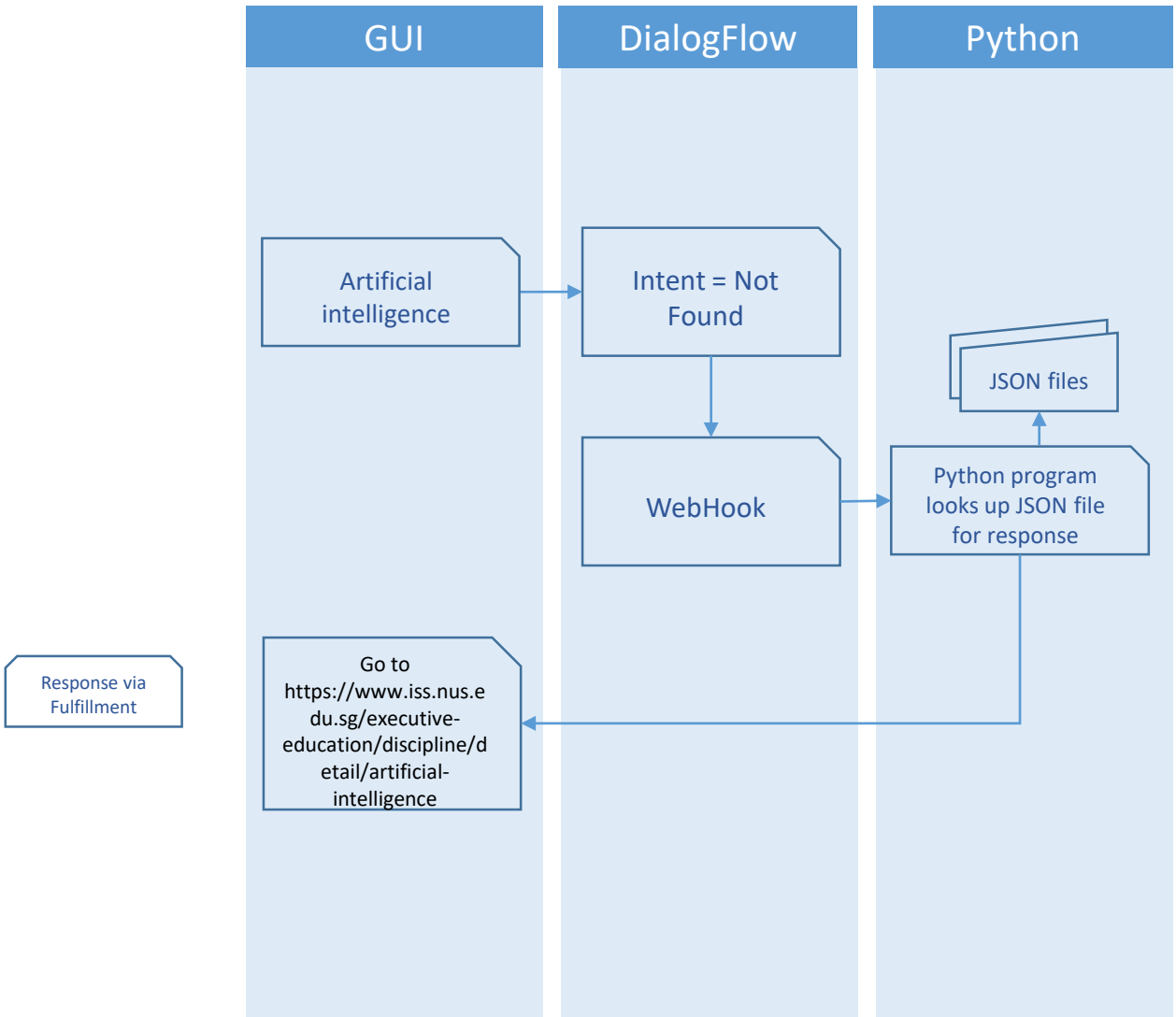
Project Performance & Validation

Scenario 3 – Missing 2 Slots



Project Performance & Validation

Scenario 4 – Only Course Name entered



Project Conclusions: Findings & Recommendation

We have met the prescribed objective of this project and that is to

1) Take in user questions in a chat GUI
2) Process the question to understand the intent
3) Evaluate the Entities of the question
4) Find the answers via Fulfilment at the Python layer, or via DialogFlow

In addition, we are able to handle exceptions with missing entities as well as vague questions.

Hence, we consider this ChatBot to be meeting the objectives and ready to be adapted for the ISS website.

More importantly, we have gained some good knowledge concerning the various solutions around the ChatBot application.