Weather Bot using Microsoft Azure LUIS (Procedure and Outcome)

Contents

Preface

This book is intended to help all the data scientists out there. It is a step by step guide for creating a chatbot, in this case, an azure bot right from scratch and then deploying it to the cloud platform. This book takes a simple example of a weather query and tries to explain the concepts simply, extensively, and thoroughly to create a chatbot right from scratch and then its deployment to a cloud environment.

1 Introduction:

A chatbot is an application that can initiate and continue a conversation using auditory and/or textual methods as a human would do. A chatbot can be either a simple rule-based engine or an intelligent application leveraging Natural Language Understanding. Many organizations today have started using chatbots extensively. Chatbots are becoming famous as they are available 24*7, provide a consistent customer experience, can handle several customers at a time, are cost-effective and hence, result in a better overall customer experience.

1.1 Uses

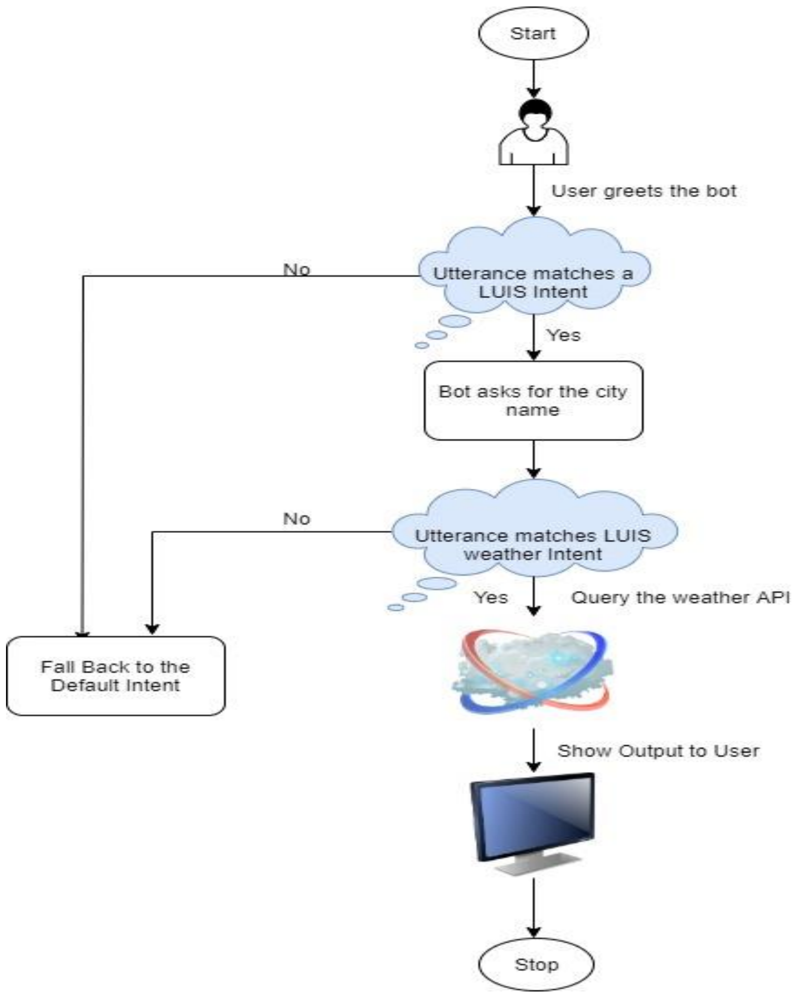
- Customer support
- Frequently Asked Questions
- Addressing Grievances
- Appointment Booking
- Automation of routine tasks
- Address a query

1.2 Prerequisites

The prerequisites for developing and understanding a chatbot using Microsoft Azure are:

- An Azure account.
- A fundamental understanding of python and flask

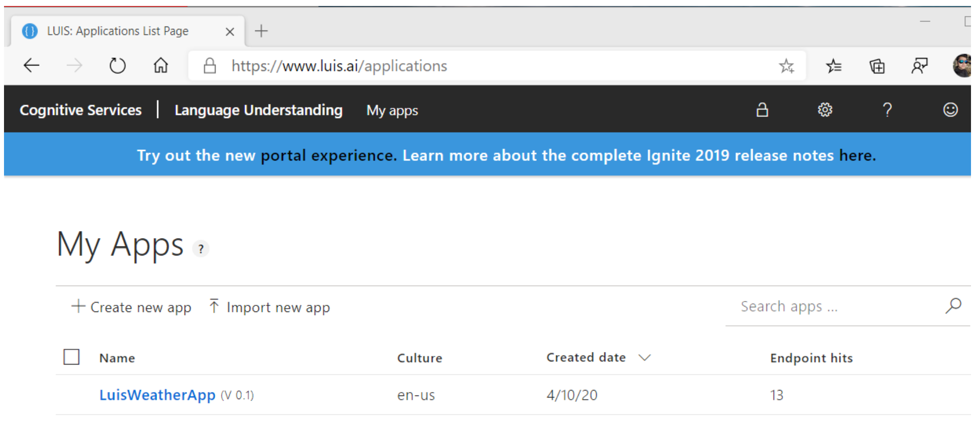
2 Application Architecture


Start
User greets the bot
Utterance matches a LUIS Intent
No
Yes
Bot asks for the city name
Utterance matches LUIS weather Intent
No
Yes
Query the weather API
Fall Back to the Default Intent
Show Output to User
Stop

3 Implementation

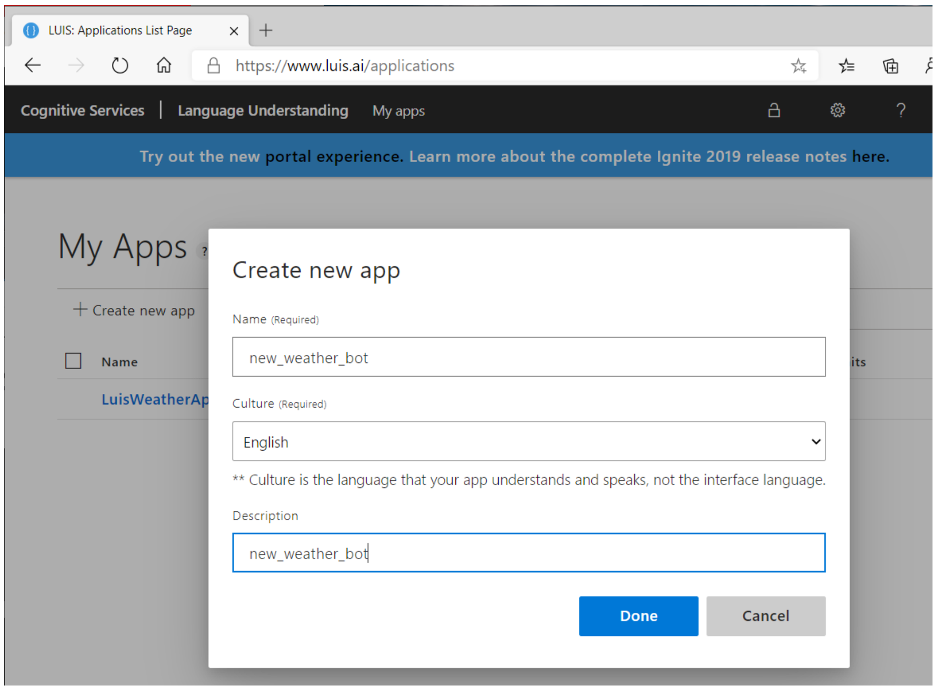
3.1 Creating a LUIS App

- Go to https://www.luis.ai and create an account if you already don't have one.
- Click on 'create new app' to create a new app by as shown:


LUIS: Applications List Page
https://www.luis.ai/applications
Cognitive Services
Language Understanding
My apps
Try out the new portal experience. Learn more about the complete Ignite 2019 release notes here.
My Apps
Create new app
Import new app
Search apps ...

Name	Culture	Created date	Endpoint hits
LuisWeatherApp (V 0.1)	en-us	4/10/20	13

Figure 1: This is the My Apps Dashboard. A new app is created by clicking 'create new app'.


LUIS: Applications List Page
https://www.luis.ai/applications
Cognitive Services
Language Understanding
My apps
Try out the new portal experience. Learn more about the complete Ignite 2019 release notes here.
My Apps
Create new app
Name
LuisWeatherAp
its

Create new app

Name (Required)

new_weather_bot

Culture (Required)

English

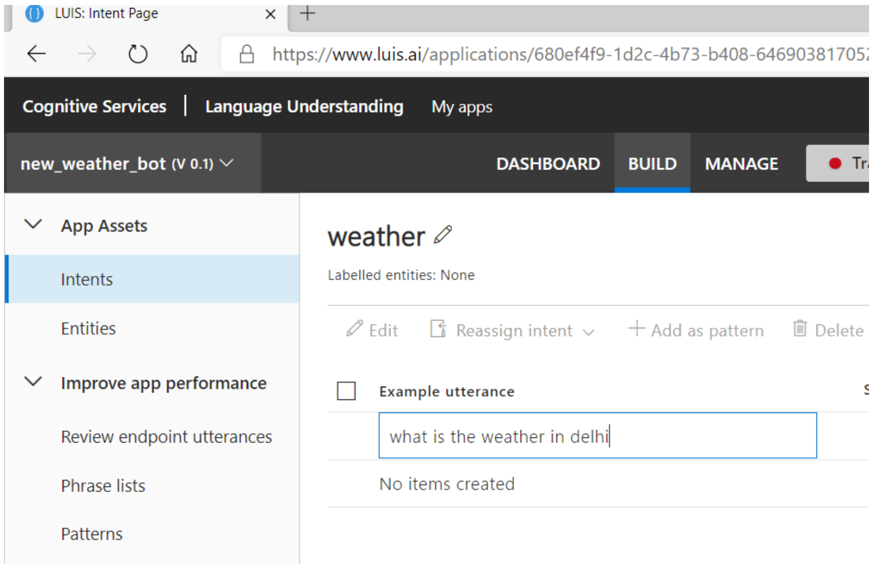
** Culture is the language that your app understands and speaks, not the interface language.

Description

new_weather_bot

Done Cancel

Figure 2: Details regarding the new app is entered.



weather

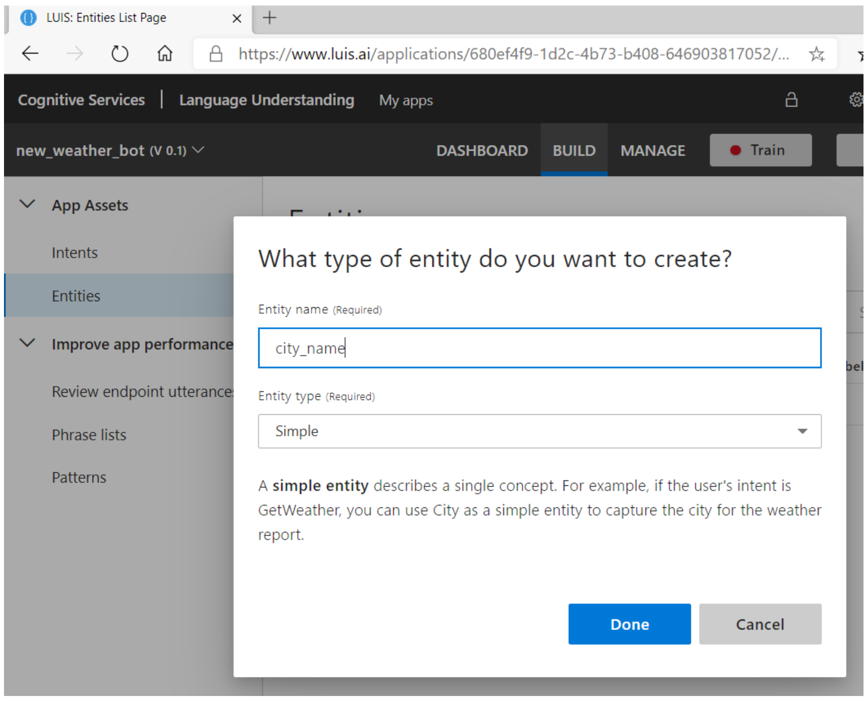
Labelled entities: None

Example utterance

what is the weather in delhi

No items created

Figure 3: Intents are created here.


LUIS: Entities List Page
https://www.luis.ai/applications/680ef4f9-1d2c-4b73-b408-646903817052/...
Cognitive Services | Language Understanding My apps
new_weather_bot (V 0.1)
DASHBOARD BUILD MANAGE Train
App Assets
Intents
Entities
Improve app performance
Review endpoint utterances
Phrase lists
Patterns
Done
Cancel

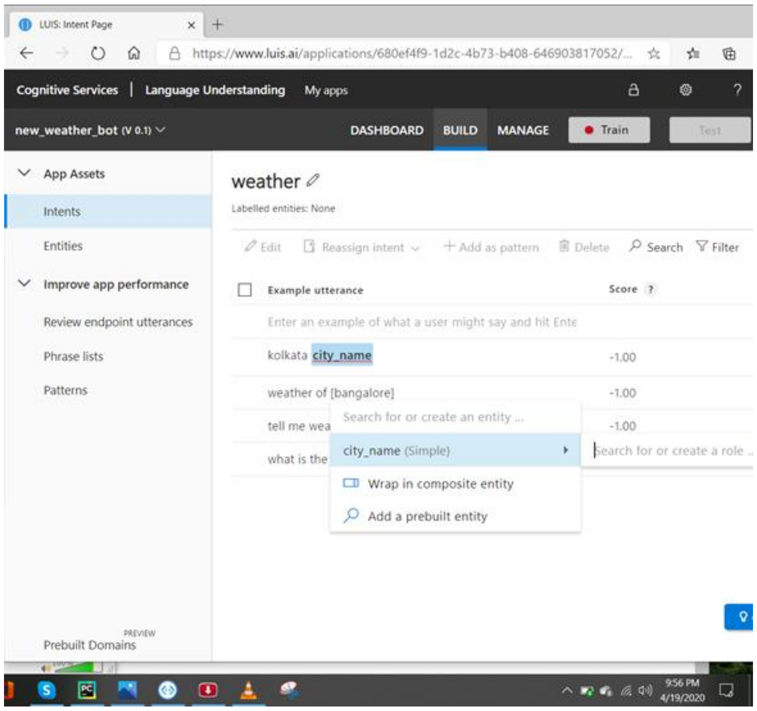
What type of entity do you want to create?

Entity name (Required): city_name

Entity type (Required): Simple

A **simple entity** describes a single concept. For example, if the user's intent is GetWeather, you can use City as a simple entity to capture the city for the weather report.

Figure 4: Entity is created here.


LUIS: Intent Page

https://www.luis.ai/applications/680ef4f9-1d2c-4b73-b408-646903817052/...

Cognitive Services | Language Understanding My apps

new_weather_bot (V 0.1) DASHBOARD BUILD MANAGE Train Test

App Assets
Intents
Entities
Improve app performance
Review endpoint utterances
Phrase lists
Patterns
Prebuilt Domains PREVIEW

weather
Labelled entities: None

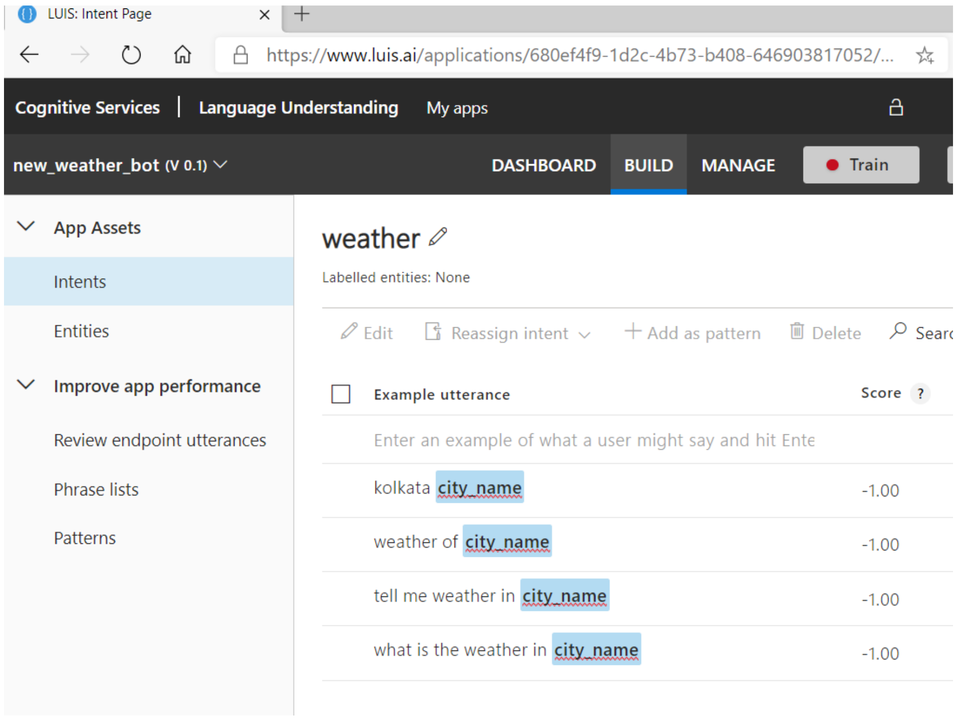
Edit Reassign intent Add as pattern Delete Search Filter

Example utterance	Score
Enter an example of what a user might say and hit Ente	
kolkata city_name	-1.00
weather of [bangalore]	-1.00
tell me wea	-1.00
what is the	

Search for or create an entity ...
city_name (Simple)
Wrap in composite entity
Add a prebuilt entity
Search for or create a role ..

9:56 PM 4/19/2020

Figure 5: Utterances are entered.


LUIS: Intent Page

https://www.luis.ai/applications/680ef4f9-1d2c-4b73-b408-646903817052/...

Cognitive Services | Language Understanding My apps

new_weather_bot (V 0.1) DASHBOARD BUILD MANAGE Train

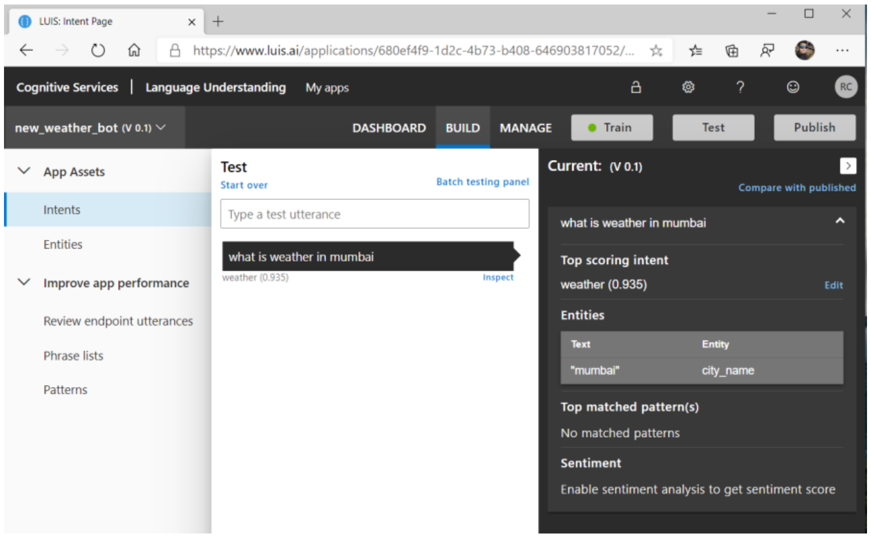
App Assets
Intents
Entities
Improve app performance
Review endpoint utterances
Phrase lists
Patterns

weather
Labelled entities: None

Edit Reassign intent Add as pattern Delete Searc

Example utterance	Score
Enter an example of what a user might say and hit Ente	
kolkata city_name	-1.00
weather of city_name	-1.00
tell me weather in city_name	-1.00
what is the weather in city_name	-1.00

Figure 6: Training of LUIS app is executed.



what is weather in mumbai

weather (0.935)

Top scoring intent

weather (0.935)

Entities

Text	Entity
"mumbai"	city_name

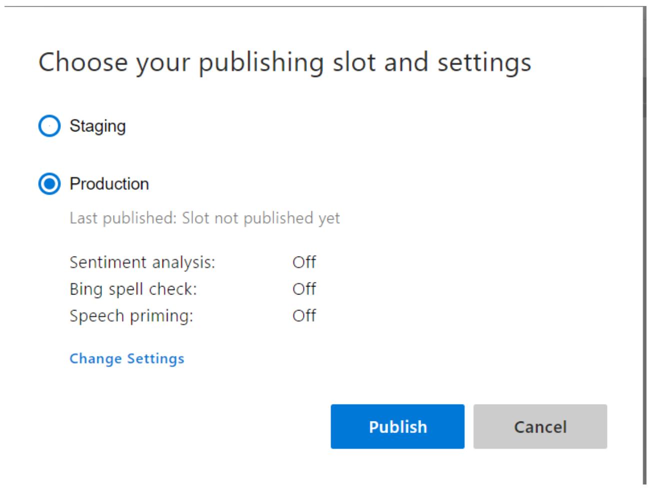
Top matched pattern(s)

No matched patterns

Sentiment

Enable sentiment analysis to get sentiment score

Figure 7: Testing of intents is done and accuracy is finally observed.



Choose your publishing slot and settings

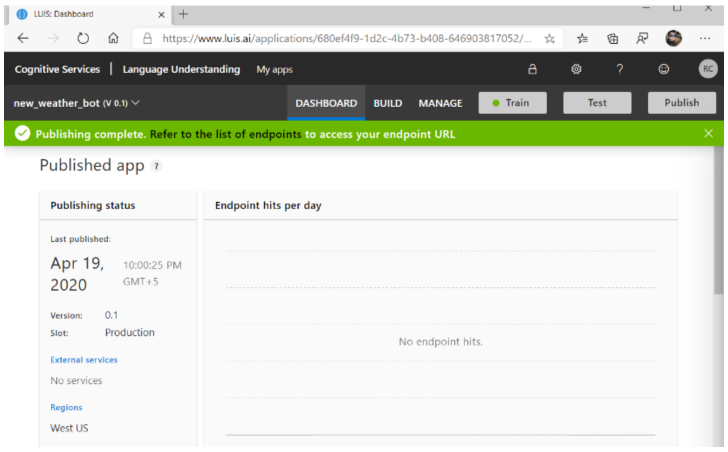
( ) Staging

[✓ Production]

Last published: Slot not published yet

Sentiment analysis:	Off
Bing spell check:	Off
Speech priming:	Off

Figure 8: Publishing is done here.


LUIS: Dashboard
https://www.luis.ai/applications/680ef4f9-1d2c-4b73-b408-646903817052/...
Cognitive Services | Language Understanding My apps
new_weather_bot (V 0.1)
DASHBOARD BUILD MANAGE Train Test Publish

Publishing complete. Refer to the list of endpoints to access your endpoint URL

Published app

Publishing status

Last published:

Apr 19, 2020 10:00:25 PM GMT+5

Version: 0.1

Slot: Production

External services

No services

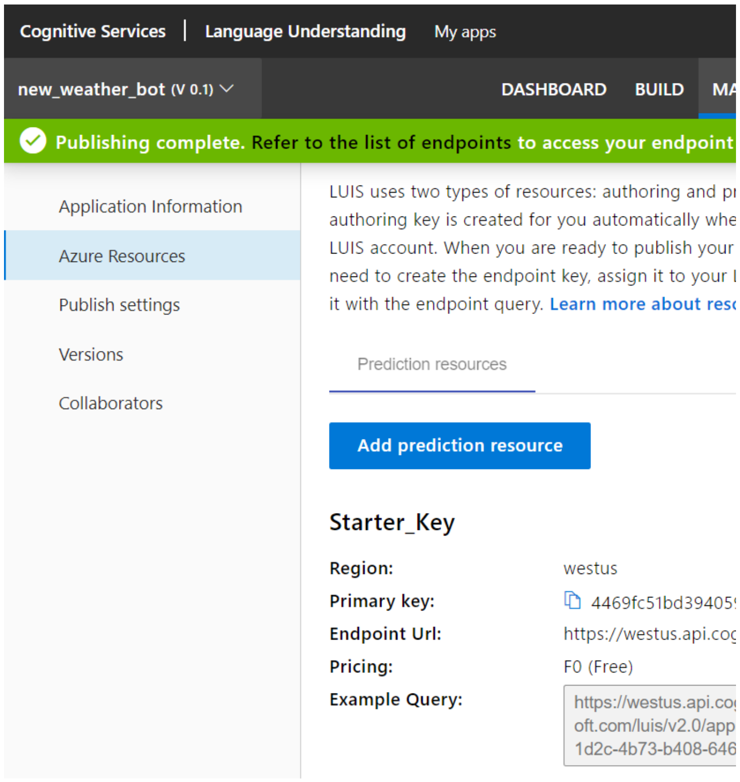
Regions

West US

Endpoint hits per day

No endpoint hits.

Figure 9: Above image represents Published app status.



Publishing complete. Refer to the list of endpoints to access your endpoint

LUIS uses two types of resources: authoring and pr
authoring key is created for you automatically whe
LUIS account. When you are ready to publish your
need to create the endpoint key, assign it to your l
it with the endpoint query. Learn more about reso

Starter_Key

Region:	westus
Primary key:	4469fc51bd39405
Endpoint Url:	https://westus.api.cog
Pricing:	F0 (Free)
Example Query:	https://westus.api.cog oft.com/luis/v2.0/app 1d2c-4b73-b408-646

Figure 10: Primary key mentioned above will serve as LUIS API Key.

3.2 Create a python app

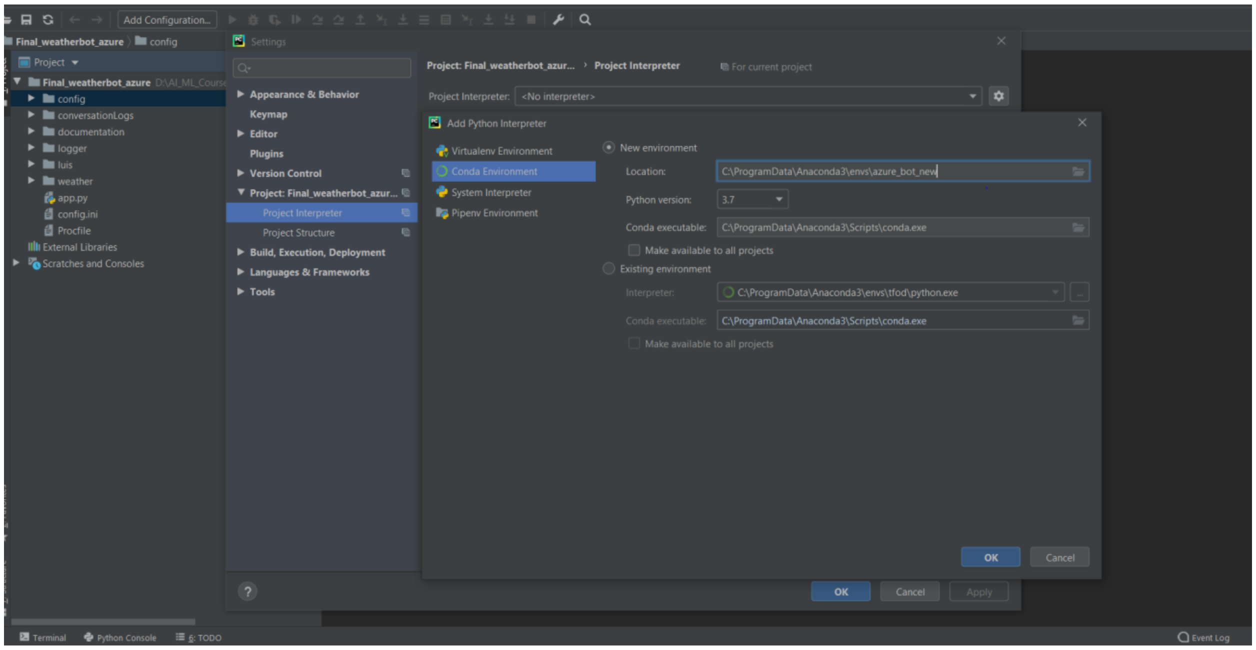
3.2.1 Subscribing to the weather API

- Go to https://home.openweathermap.org/, sign in/signup, and create an API Key for calling the current weather data API.
- This will act as the weather_api_key.

3.2.2 App creation and Integration with LUIS

- Create a folder for your chatbot called azurePythonBot.
- Open the folder through pycharm.
- Create a file called app.py and put the following code.

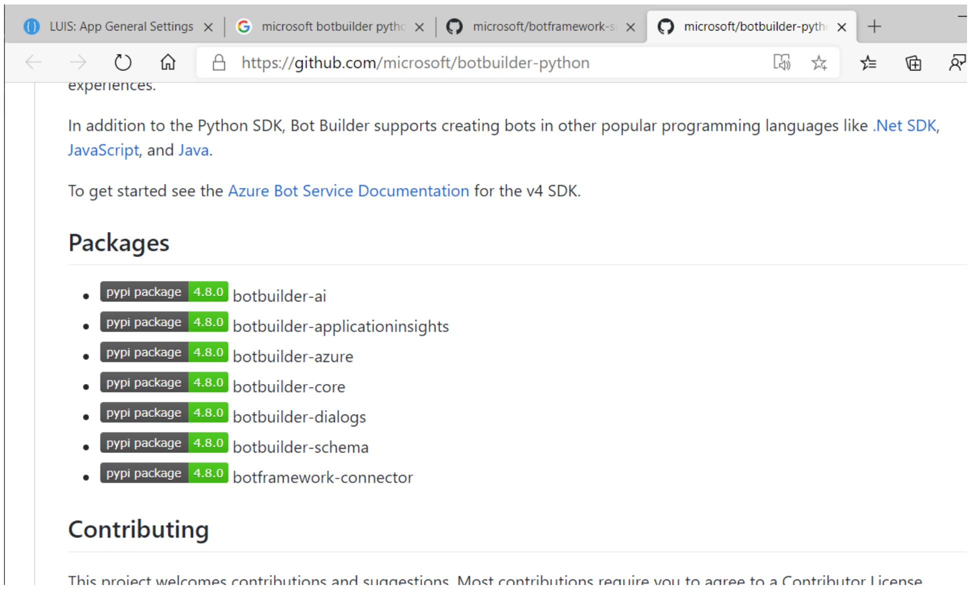
3.2.3 CODE Screenshot



Project Interpreter: <No interpreter>

New environment
Location: C:\ProgramData\Anaconda3\envs\azure_bot_new
Python version: 3.7
Conda executable: C:\ProgramData\Anaconda3\Scripts\conda.exe
Make available to all projects
Existing environment
Interpreter: C:\ProgramData\Anaconda3\envs\tfod\python.exe
Conda executable: C:\ProgramData\Anaconda3\Scripts\conda.exe
Make available to all projects

Figure 11:Set Environment in pycharm


LUIS: App General Settings
microsoft botbuilder pytho
microsoft/botframework-s
microsoft/botbuilder-pyth

https://github.com/microsoft/botbuilder-python

experiences.

In addition to the Python SDK, Bot Builder supports creating bots in other popular programming languages like .Net SDK, JavaScript, and Java.

To get started see the Azure Bot Service Documentation for the v4 SDK.

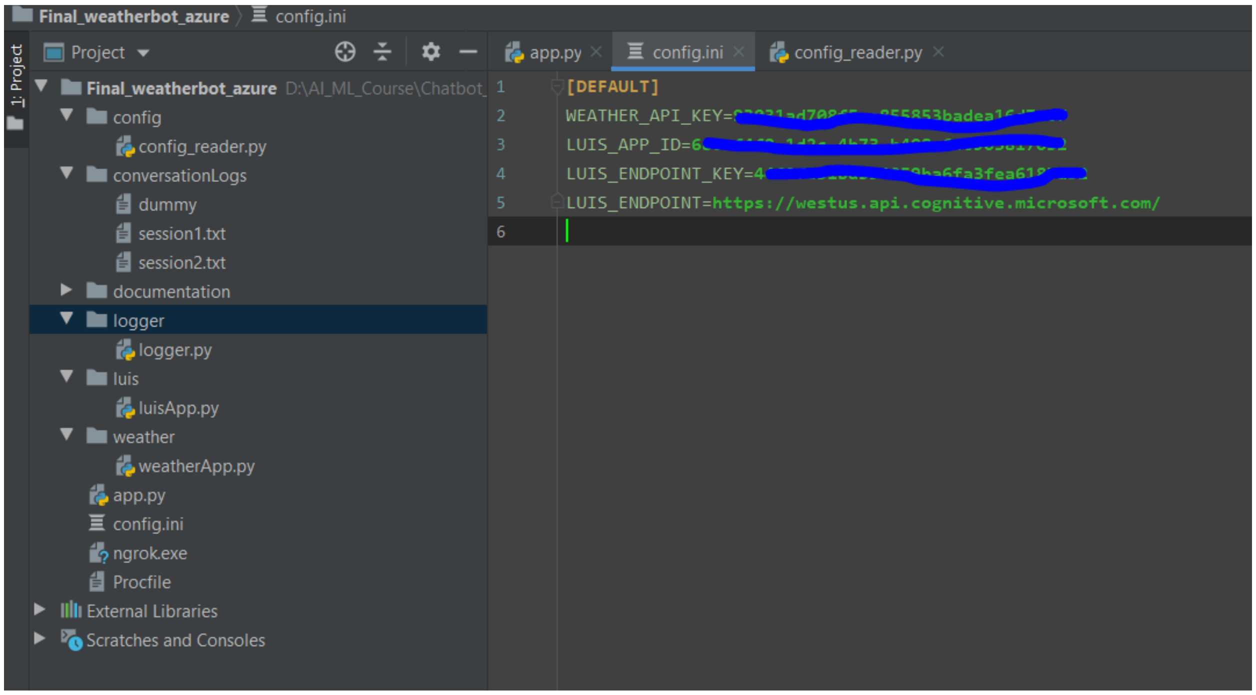
Packages

- pypi package 4.8.0 botbuilder-ai
- pypi package 4.8.0 botbuilder-applicationinsights
- pypi package 4.8.0 botbuilder-azure
- pypi package 4.8.0 botbuilder-core
- pypi package 4.8.0 botbuilder-dialogs
- pypi package 4.8.0 botbuilder-schema
- pypi package 4.8.0 botframework-connector

Contributing

This project welcomes contributions and suggestions. Most contributions require you to agree to a Contributor License

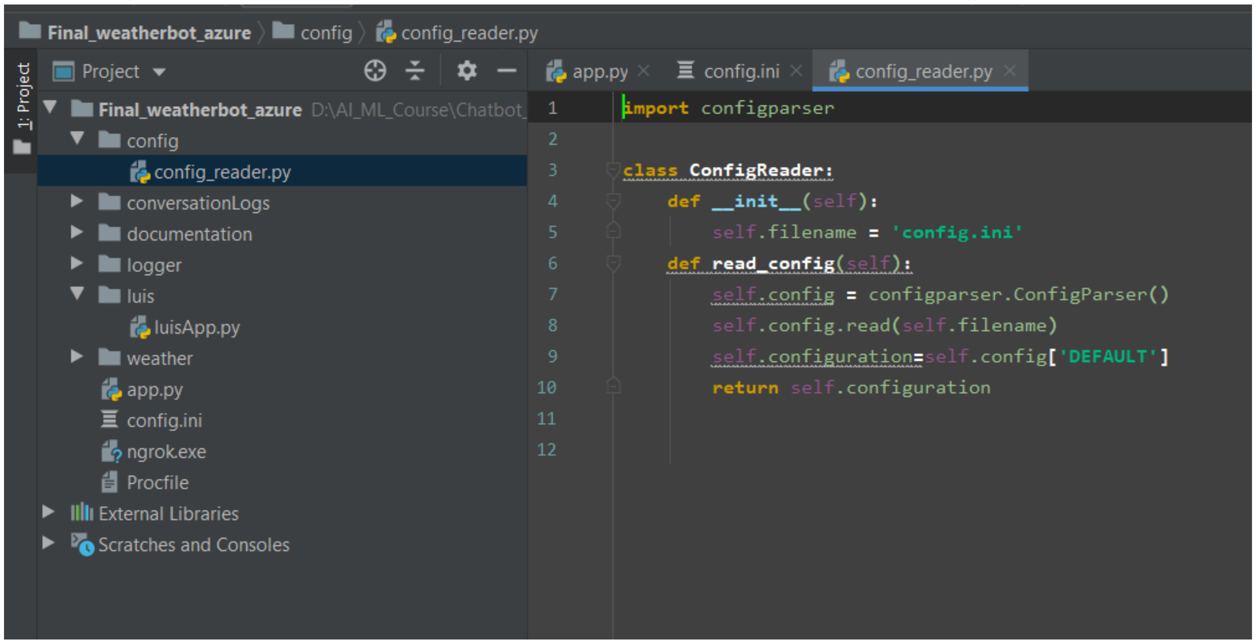
Figure 12: Install all depedency



```
[DEFAULT]
WEATHER_API_KEY=[redacted]
LUIS_APP_ID=6[redacted]
LUIS_ENDPOINT_KEY=4[redacted]
LUIS_ENDPOINT=https://westus.api.cognitive.microsoft.com/
```

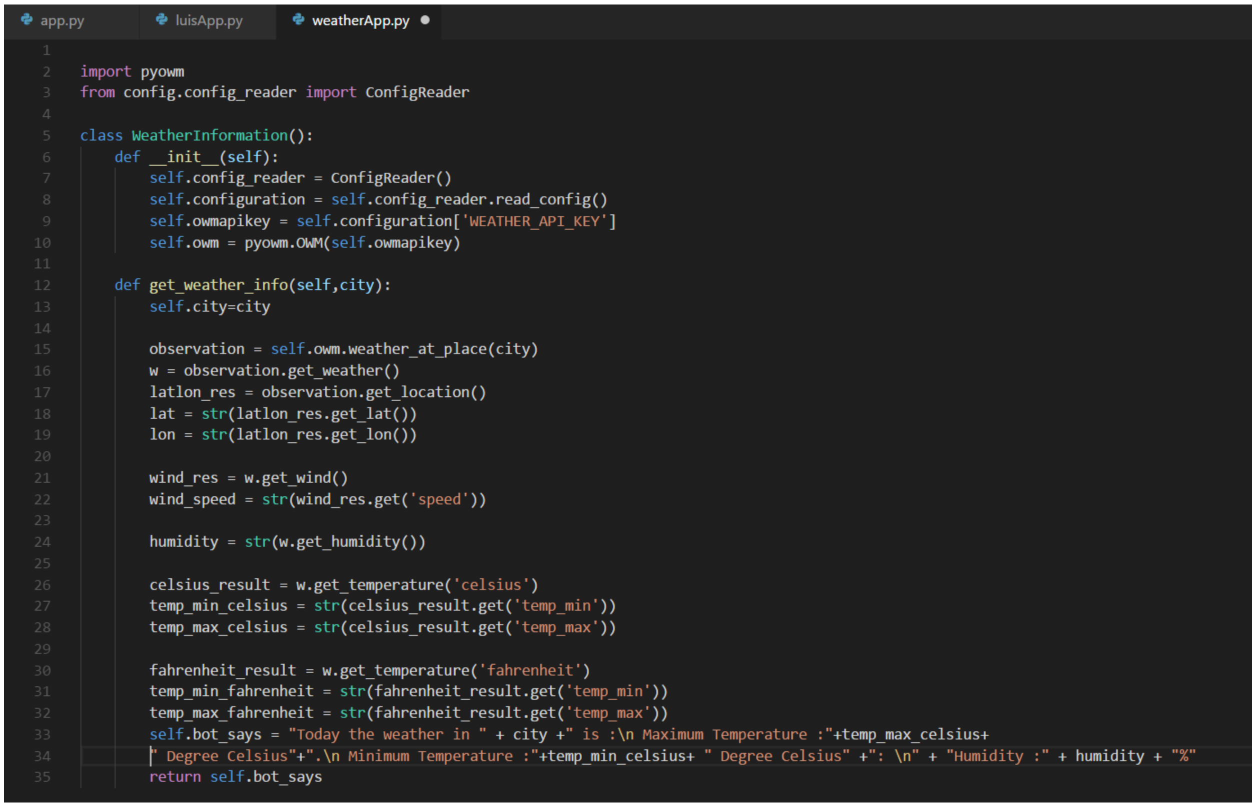
Figure 13: Folder structure and config file

Provide all the relevant details



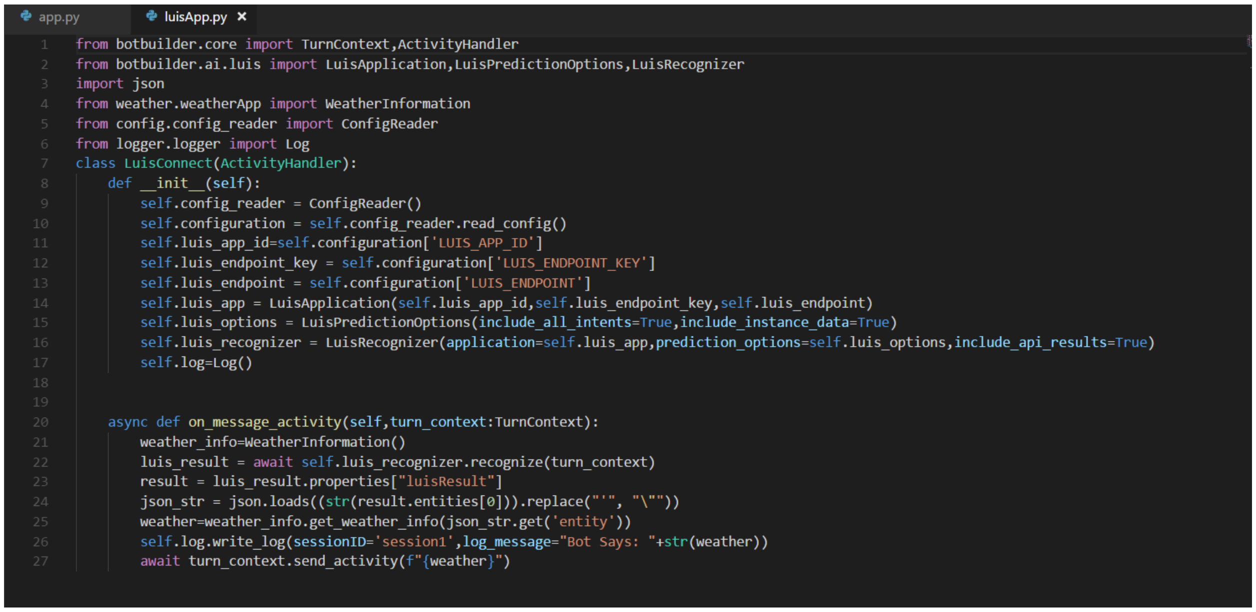
```python
import configparser

class ConfigReader:
    def __init__(self):
        self.filename = 'config.ini'
    def read_config(self):
        self.config = configparser.ConfigParser()
        self.config.read(self.filename)
        self.configuration=self.config['DEFAULT']
        return self.configuration
```

Figure 14: Config reader file



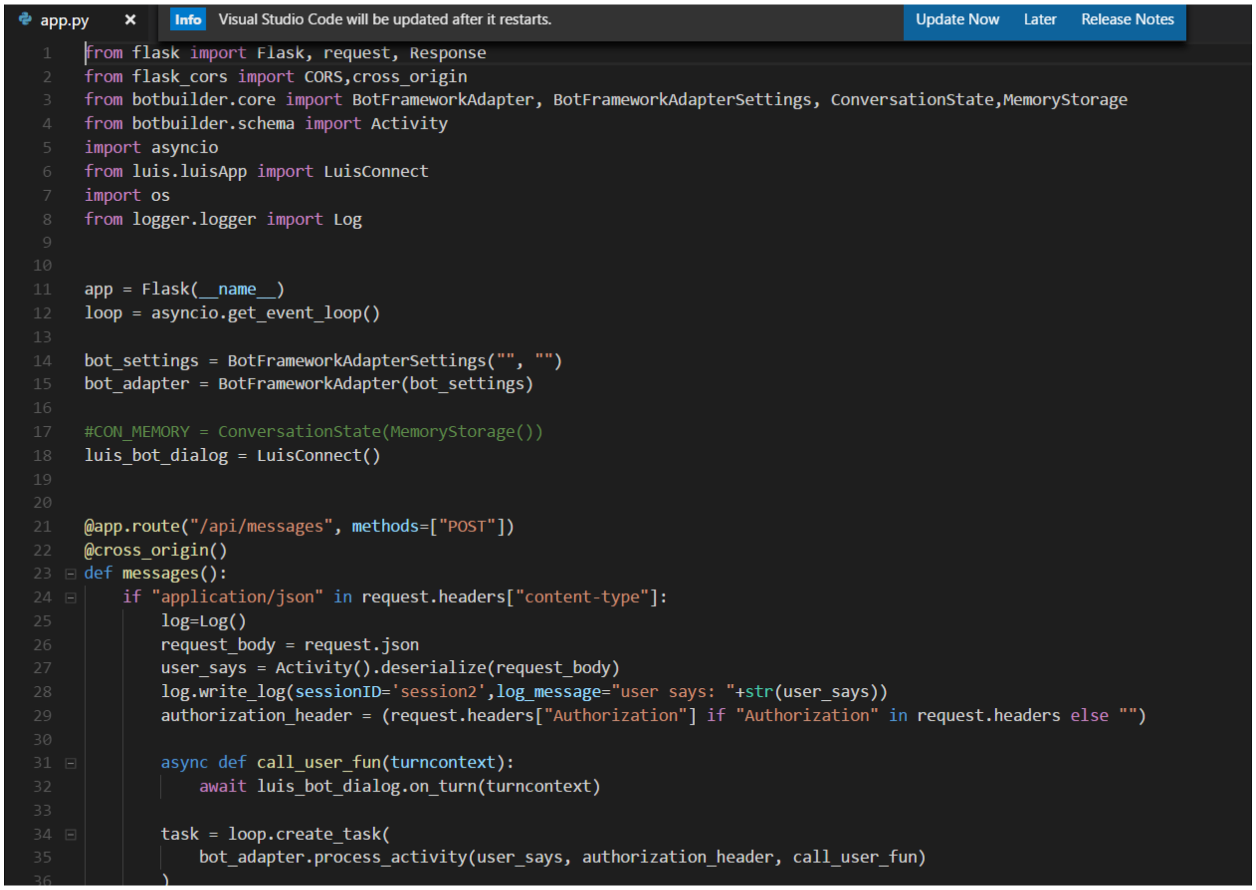
```python

import pyowm
from config.config_reader import ConfigReader

class WeatherInformation():
    def __init__(self):
        self.config_reader = ConfigReader()
        self.configuration = self.config_reader.read_config()
        self.owmapikey = self.configuration['WEATHER_API_KEY']
        self.owm = pyowm.OWM(self.owmapikey)

    def get_weather_info(self,city):
        self.city=city

        observation = self.owm.weather_at_place(city)
        w = observation.get_weather()
        latlon_res = observation.get_location()
        lat = str(latlon_res.get_lat())
        lon = str(latlon_res.get_lon())

        wind_res = w.get_wind()
        wind_speed = str(wind_res.get('speed'))

        humidity = str(w.get_humidity())

        celsius_result = w.get_temperature('celsius')
        temp_min_celsius = str(celsius_result.get('temp_min'))
        temp_max_celsius = str(celsius_result.get('temp_max'))

        fahrenheit_result = w.get_temperature('fahrenheit')
        temp_min_fahrenheit = str(fahrenheit_result.get('temp_min'))
        temp_max_fahrenheit = str(fahrenheit_result.get('temp_max'))
        self.bot_says = "Today the weather in " + city +" is :\n Maximum Temperature :"+temp_max_celsius+
        " Degree Celsius"+".\n Minimum Temperature :"+temp_min_celsius+ " Degree Celsius" +": \n" + "Humidity :" + humidity + "%"
        return self.bot_says
```

Figure 15: weatherApp python file



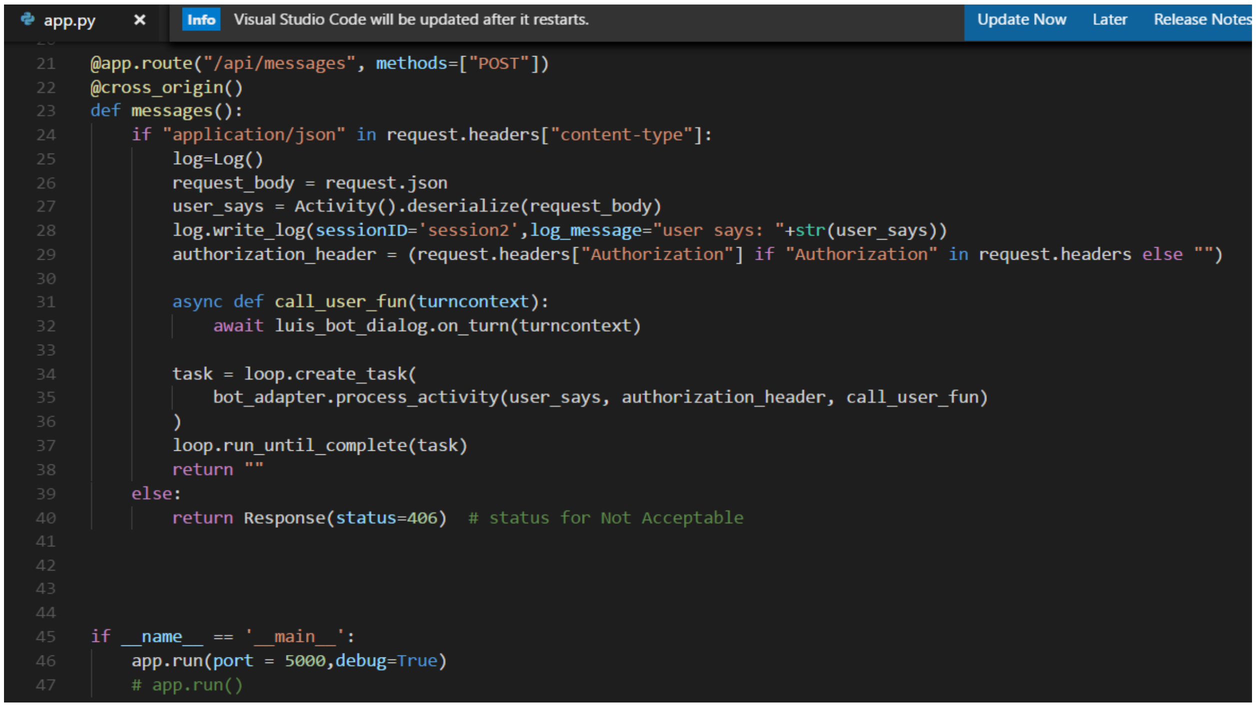
```python
from botbuilder.core import TurnContext,ActivityHandler
from botbuilder.ai.luis import LuisApplication,LuisPredictionOptions,LuisRecognizer
import json
from weather.weatherApp import WeatherInformation
from config.config_reader import ConfigReader
from logger.logger import Log
class LuisConnect(ActivityHandler):
    def __init__(self):
        self.config_reader = ConfigReader()
        self.configuration = self.config_reader.read_config()
        self.luis_app_id=self.configuration['LUIS_APP_ID']
        self.luis_endpoint_key = self.configuration['LUIS_ENDPOINT_KEY']
        self.luis_endpoint = self.configuration['LUIS_ENDPOINT']
        self.luis_app = LuisApplication(self.luis_app_id,self.luis_endpoint_key,self.luis_endpoint)
        self.luis_options = LuisPredictionOptions(include_all_intents=True,include_instance_data=True)
        self.luis_recognizer = LuisRecognizer(application=self.luis_app,prediction_options=self.luis_options,include_api_results=True)
        self.log=Log()


    async def on_message_activity(self,turn_context:TurnContext):
        weather_info=WeatherInformation()
        luis_result = await self.luis_recognizer.recognize(turn_context)
        result = luis_result.properties["luisResult"]
        json_str = json.loads((str(result.entities[0])).replace("'", "\""))
        weather=weather_info.get_weather_info(json_str.get('entity'))
        self.log.write_log(sessionID='session1',log_message="Bot Says: "+str(weather))
        await turn_context.send_activity(f"{weather}")
```

Figure 16 luisApp python file



```python
from flask import Flask, request, Response
from flask_cors import CORS,cross_origin
from botbuilder.core import BotFrameworkAdapter, BotFrameworkAdapterSettings, ConversationState,MemoryStorage
from botbuilder.schema import Activity
import asyncio
from luis.luisApp import LuisConnect
import os
from logger.logger import Log


app = Flask(__name__)
loop = asyncio.get_event_loop()

bot_settings = BotFrameworkAdapterSettings("", "")
bot_adapter = BotFrameworkAdapter(bot_settings)

#CON_MEMORY = ConversationState(MemoryStorage())
luis_bot_dialog = LuisConnect()


@app.route("/api/messages", methods=["POST"])
@cross_origin()
def messages():
    if "application/json" in request.headers["content-type"]:
        log=Log()
        request_body = request.json
        user_says = Activity().deserialize(request_body)
        log.write_log(sessionID='session2',log_message="user says: "+str(user_says))
        authorization_header = (request.headers["Authorization"] if "Authorization" in request.headers else "")

        async def call_user_fun(turncontext):
            await luis_bot_dialog.on_turn(turncontext)

        task = loop.create_task(
            bot_adapter.process_activity(user_says, authorization_header, call_user_fun)
```

Figure 17: Main app.py file (Entery File)

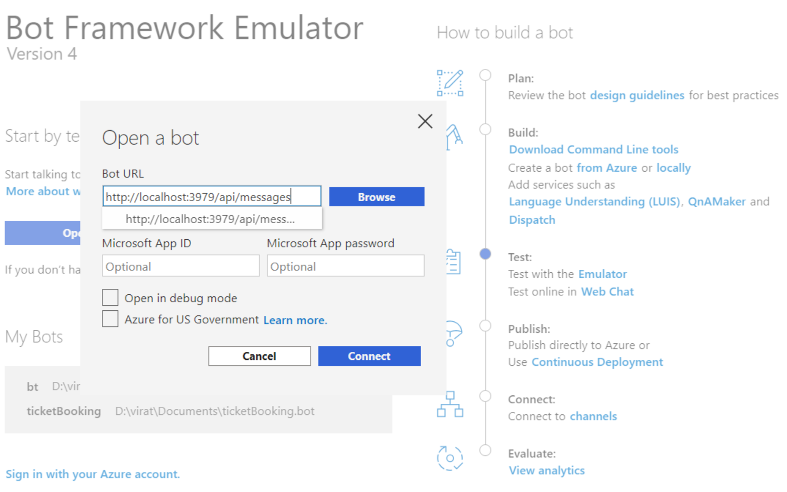


```python
@app.route("/api/messages", methods=["POST"])
@cross_origin()
def messages():
    if "application/json" in request.headers["content-type"]:
        log=Log()
        request_body = request.json
        user_says = Activity().deserialize(request_body)
        log.write_log(sessionID='session2',log_message="user says: "+str(user_says))
        authorization_header = (request.headers["Authorization"] if "Authorization" in request.headers else "")

        async def call_user_fun(turncontext):
            await luis_bot_dialog.on_turn(turncontext)

        task = loop.create_task(
            bot_adapter.process_activity(user_says, authorization_header, call_user_fun)
        )
        loop.run_until_complete(task)
        return ""
    else:
        return Response(status=406)  # status for Not Acceptable




if __name__ == '__main__':
    app.run(port = 5000,debug=True)
    # app.run()
```

Figure 18 Main app.py file continue

3.3 Install Bot Emulator and test

- Go to https://github.com/Microsoft/BotFramework-Emulator/releases and download the Bot Emulator setup file based on your computer.
- Once the download is completed, double click the installation file and it'll automatically install the Bot Emulator.
- Run the bot emulator and connect to the already running bot file(app.py) as shown:



Bot Framework Emulator
Version 4

Start by te
Start talking to
More about w
Op
If you don't ha

My Bots
bt D:\vir
ticketBooking D:\virat\Documents\ticketBooking.bot

Sign in with your Azure account.

Open a bot

Bot URL: http://localhost:3979/api/messages
http://localhost:3979/api/mess...
Microsoft App ID: Optional
Microsoft App password: Optional
☐ Open in debug mode
☐ Azure for US Government Learn more.

How to build a bot

Plan:
Review the bot design guidelines for best practices

Build:
Download Command Line tools
Create a bot from Azure or locally
Add services such as
Language Understanding (LUIS), QnAMaker and Dispatch

Test:
Test with the Emulator
Test online in Web Chat

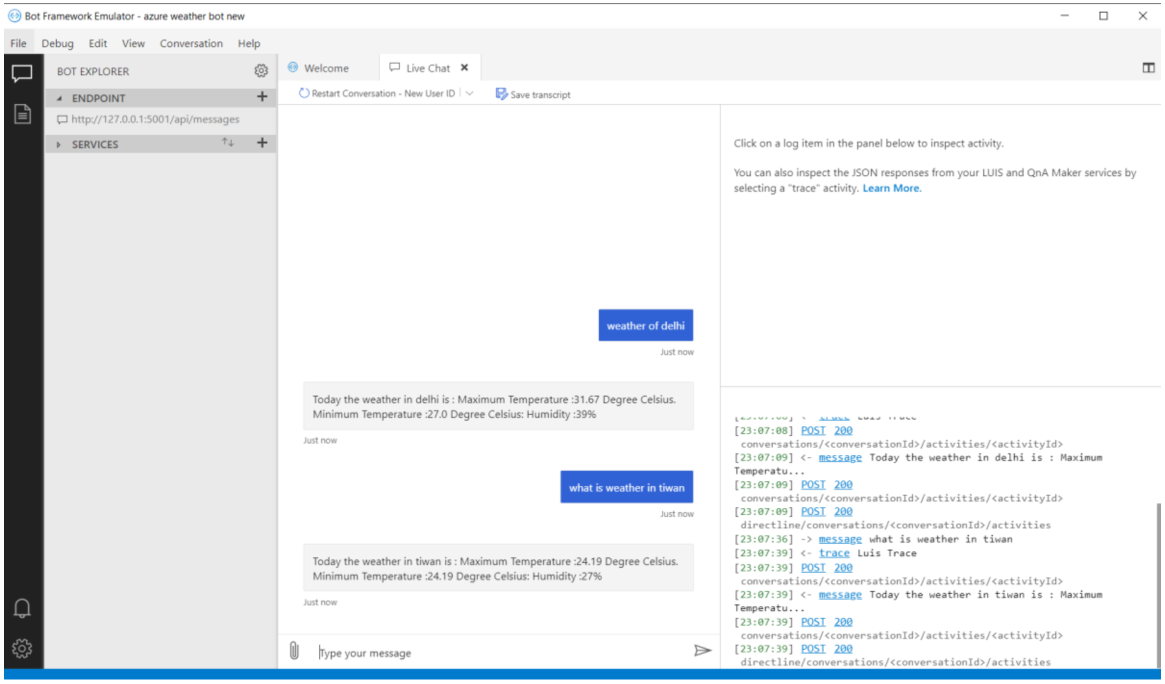
Publish:
Publish directly to Azure or
Use Continuous Deployment

Connect:
Connect to channels

Evaluate:
View analytics

Figure 19: Local port URL is given.

Give the port number on which your app is running.



Bot Framework Emulator - azure weather bot new

http://127.0.0.1:5001/api/messages

Click on a log item in the panel below to inspect activity.

You can also inspect the JSON responses from your LUIS and QnA Maker services by selecting a "trace" activity. Learn More.

weather of delhi
Just now

Today the weather in delhi is : Maximum Temperature :31.67 Degree Celsius. Minimum Temperature :27.0 Degree Celsius: Humidity :39%
Just now

what is weather in tiwan
Just now

Today the weather in tiwan is : Maximum Temperature :24.19 Degree Celsius. Minimum Temperature :24.19 Degree Celsius: Humidity :27%
Just now

Type your message

```
[23:07:08] POST 200
 conversations/<conversationId>/activities/<activityId>
[23:07:09] <- message Today the weather in delhi is : Maximum
Temperatu...
[23:07:09] POST 200
 conversations/<conversationId>/activities/<activityId>
[23:07:09] POST 200
 directline/conversations/<conversationId>/activities
[23:07:36] -> message what is weather in tiwan
[23:07:39] <- trace Luis Trace
[23:07:39] POST 200
 conversations/<conversationId>/activities/<activityId>
[23:07:39] <- message Today the weather in tiwan is : Maximum
Temperatu...
[23:07:39] POST 200
 conversations/<conversationId>/activities/<activityId>
[23:07:39] POST 200
 directline/conversations/<conversationId>/activities
```

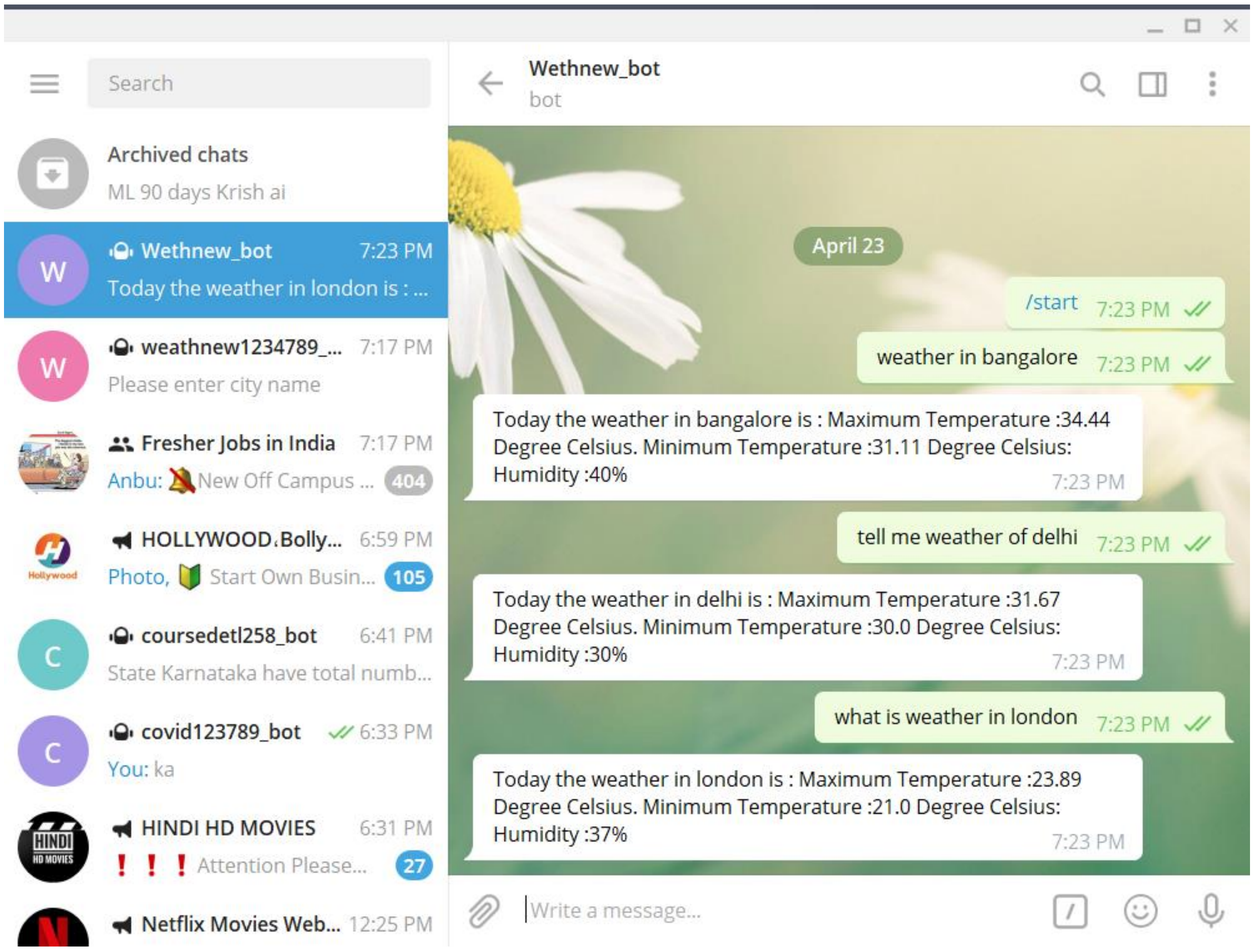
Figure 20: Live chat example in Bot Emulator.

3.4 Telegram Integration:

- Download ngrok from https://ngrok.com/download
- After extracting the zip file, open the ngrok file and run it.
- In ngrok, enter the command 'ngrok http 5000 ':

3.4.1 Then go to telegram and create your own bot using Botfather:

- Open the telegram app and search for botfather(it is an inbuilt bot used to create other bots)
- Start a conversation with botfather and enter /newbot to create a newbot.
- Give a name to your bot
- Give a username to your bot, which must end in _bot. This generates an access token. This should be enter to config file


Search
Archived chats
ML 90 days Krish ai
Wethnew_bot 7:23 PM
Today the weather in london is : ...
weathnew1234789_... 7:17 PM
Please enter city name
Fresher Jobs in India 7:17 PM
Anbu: New Off Campus ... 404
HOLLYWOOD.Bolly... 6:59 PM
Photo, Start Own Busin... 105
coursedetl258_bot 6:41 PM
State Karnataka have total numb...
covid123789_bot 6:33 PM
You: ka
HINDI HD MOVIES 6:31 PM
Attention Please... 27
Netflix Movies Web... 12:25 PM
Wethnew_bot
bot
April 23
Write a message...

/start 7:23 PM

weather in bangalore 7:23 PM

Today the weather in bangalore is : Maximum Temperature :34.44 Degree Celsius. Minimum Temperature :31.11 Degree Celsius: Humidity :40% 7:23 PM

tell me weather of delhi 7:23 PM

Today the weather in delhi is : Maximum Temperature :31.67 Degree Celsius. Minimum Temperature :30.0 Degree Celsius: Humidity :30% 7:23 PM

what is weather in london 7:23 PM

Today the weather in london is : Maximum Temperature :23.89 Degree Celsius. Minimum Temperature :21.0 Degree Celsius: Humidity :37% 7:23 PM

Figure 21: Live Telegram Chat

Thanks for Watch