Project Development Phase
Delivery of Sprint- 4

Date	05 November 2022
Team ID	PNT2022TMID28540
Project Name	AI-based discourse for Banking Industry

Creating Assistant & Integrate With Flask Webpage

You will be creating a banking bot in this activity that has the following capabilities

- The Bot should be able to guide a customer to create a bank account.
- The Bot should be able to answer loan queries.
- The Bot should be able to answer general banking queries.
- The Bot should be able to answer queries regarding net banking.
- With the help of this bot ,you can get all the required details related to banking.

Let us build our flask application which will be running in our local browser with auser interface.

In the flask application ,users will interact with the chatbot, and based on the userqueries they will get the outcomes.

Build Python Code

1:ImportingLibraries

The first step is usually importing the libraries that will be needed in theprogram.

```
from flask import Flask, render_template
```

Importing the flask module into the project is mandatory. An object of the Flask class is our WSGI application. Flask constructor takes the name of thecurrent module(name).

2 . Creating our flask application and loading

```
app = Flask(__name__)
```

- **Routing to the HTML Page**

Here, the declared construct or issued to route to the HTML page created earlier.

The ‘’ route is bound with the bot function. Hence, when the home page ofa webserver is opened in the browser, the HTML page will be rendered.

```
@app.route('/')
def bot():
    return render_template('chatbot.html')
```

Main Function

This is used to run the application in localhost.

```
if __name__ == '__main__':
    app.run()
```

Build HTML Code

- We use HTML to create the front-end part of the webpage.
- Here, we have created 1 HTMLpage-Chatbot.html
- Chatbot.html displays the home page which integrates with WatsonAssistant.
- A simple HTML page is created. Auto-generated source code fromIBM Watson Assistants is copied and pasted inside the body tag

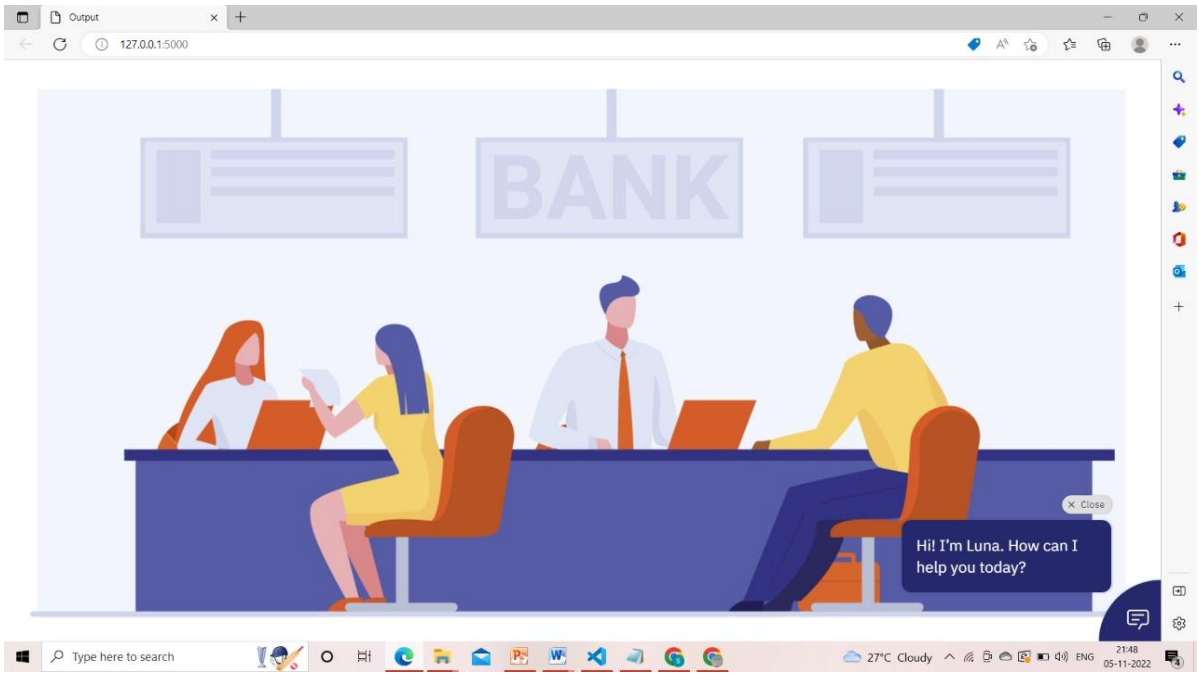
Run The Application

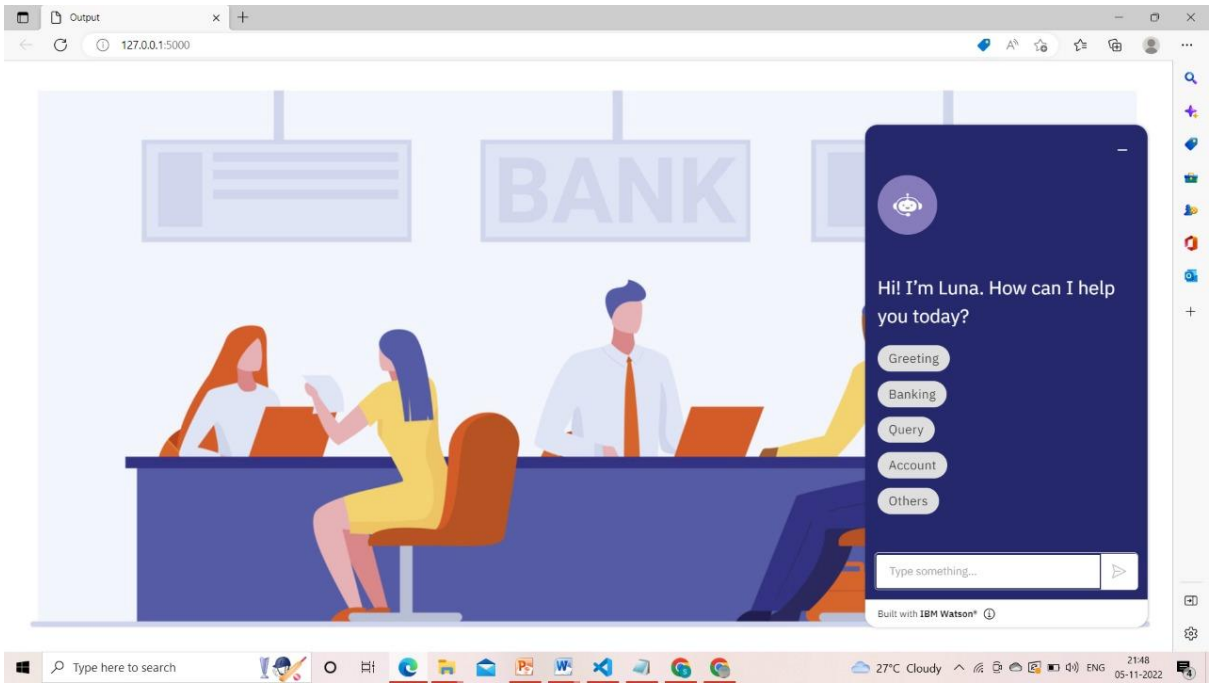
Run the application

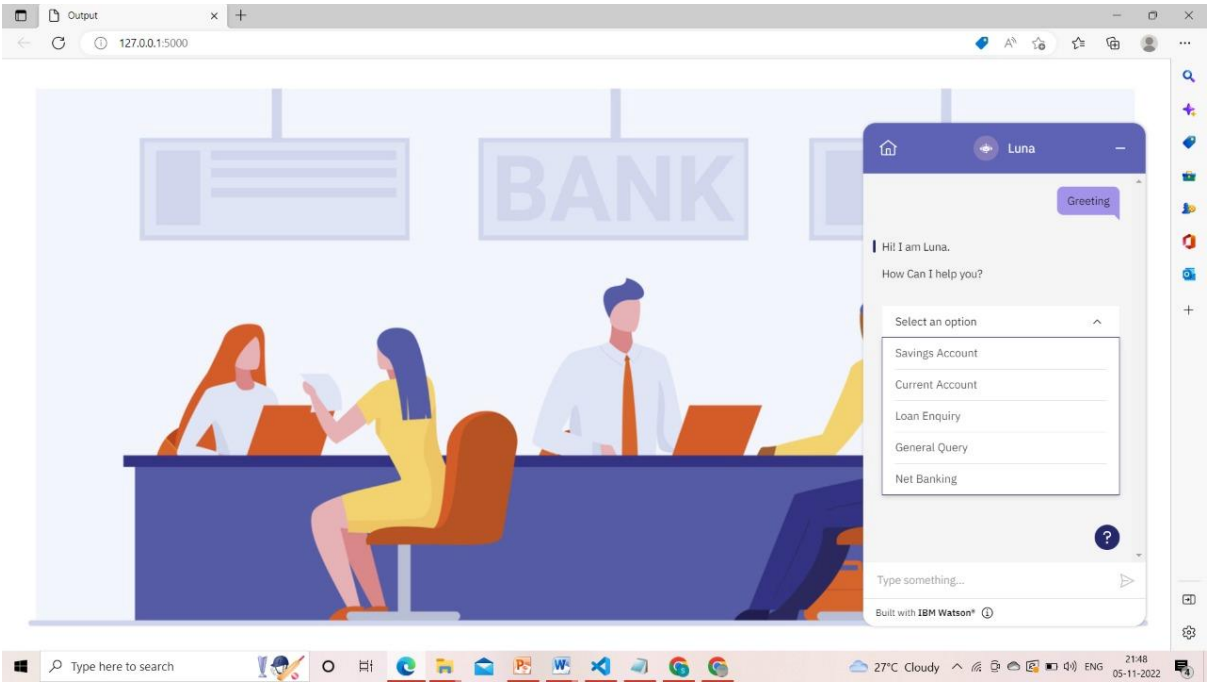
- Open the anaconda prompt from the start menu.
- Navigate to the folder where your app.pyresides.
- Now type the “pythonapp.py” command.

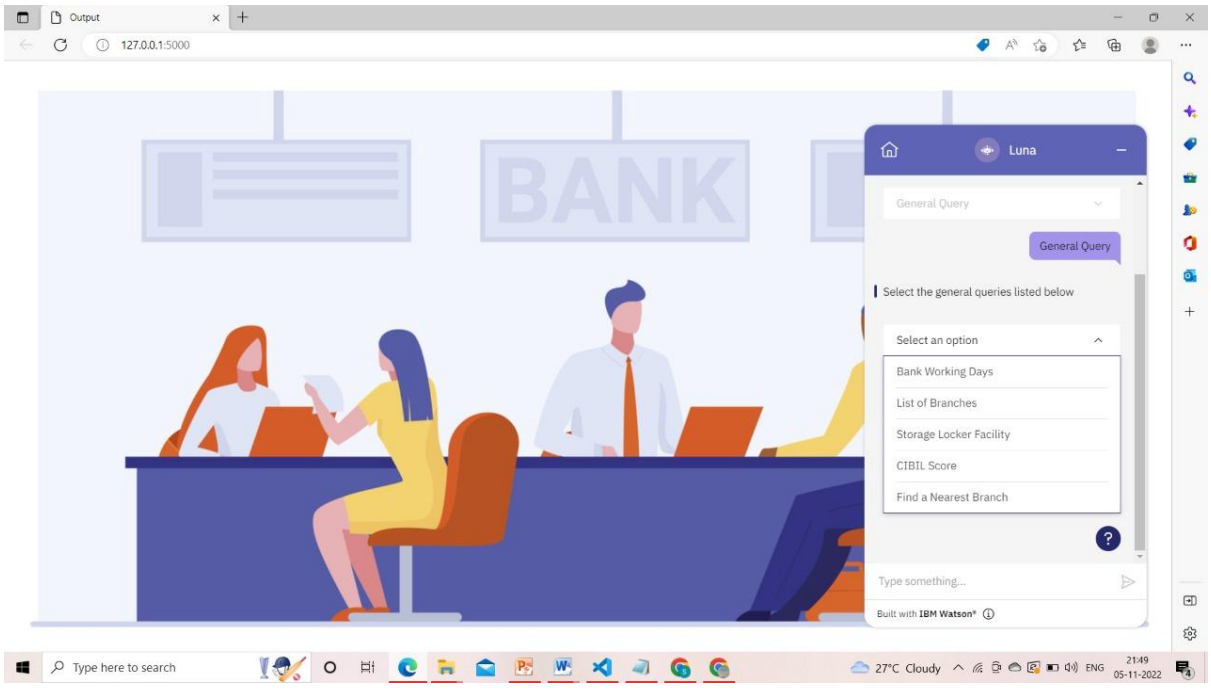
- It will show the localhost where your app is running on http://127.0.0.1.5000/
- Copy that localhost URL and open that URL in the browser. Itdoes navigate me to where you can view your webpage.

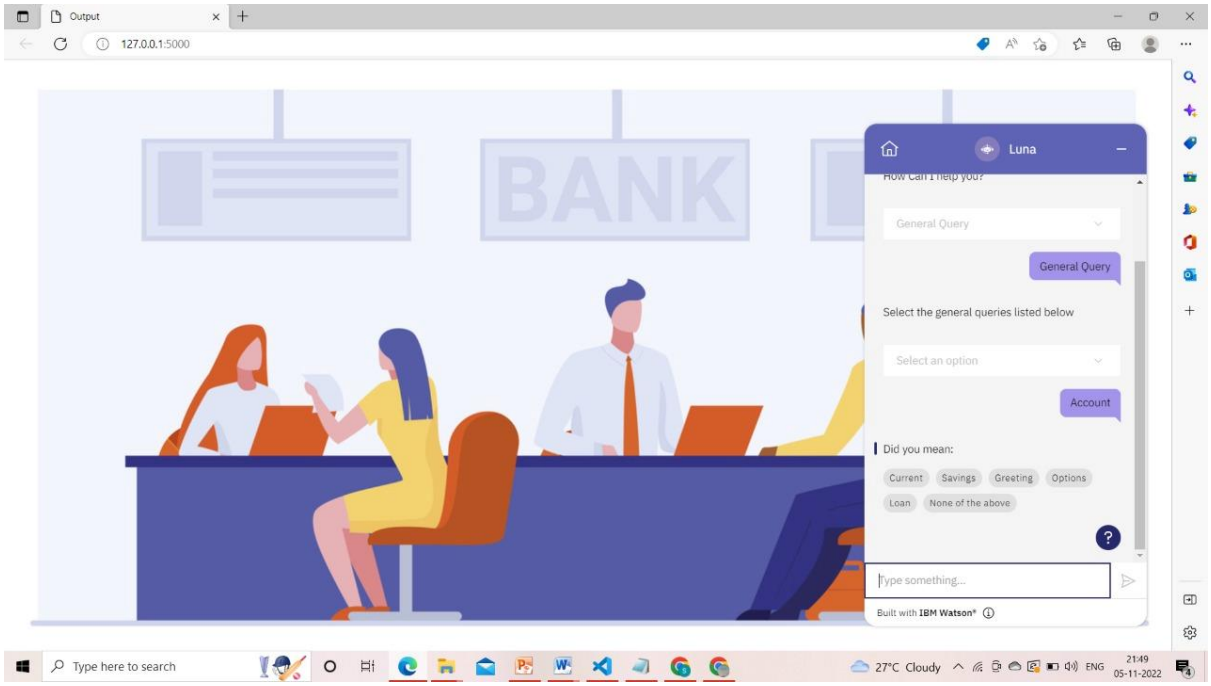
Then it will run on localhost:5000

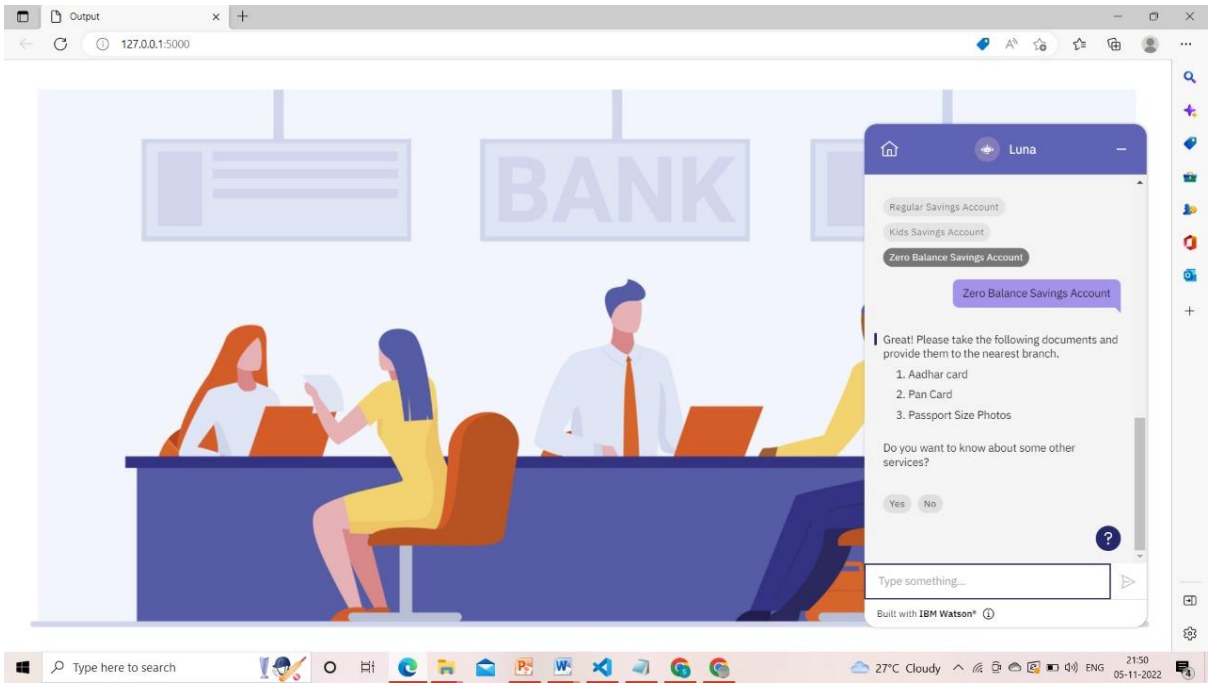


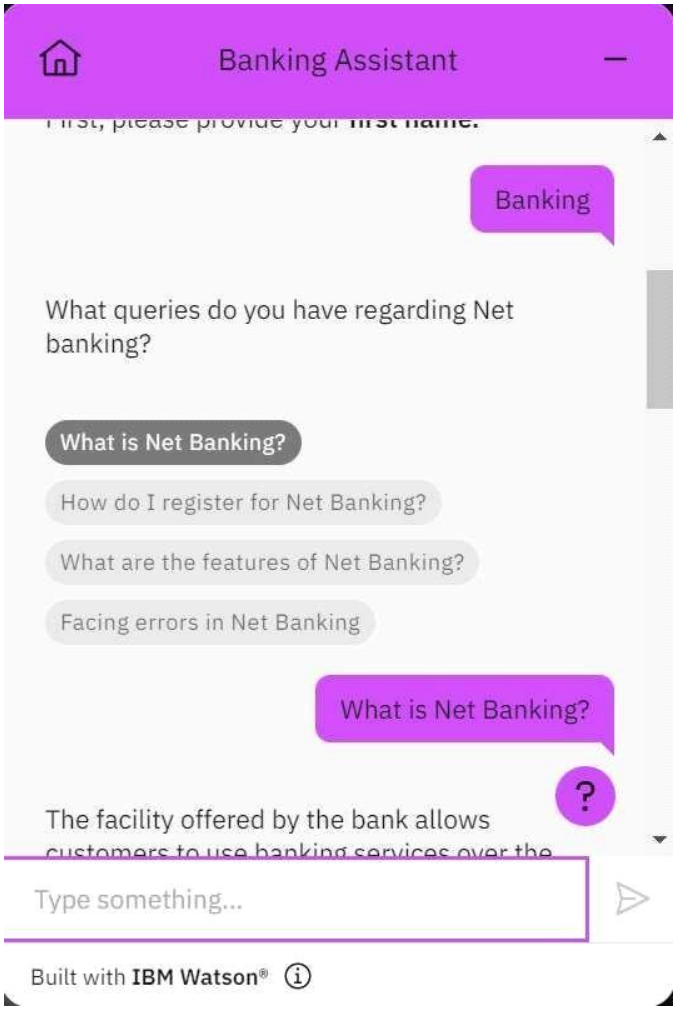



Output
127.0.0.1:5000
BANK
Luna
Greeting
Hi! I am Luna.
How Can I help you?
Select an option
Savings Account
Current Account
Loan Enquiry
General Query
Net Banking
Type something...
Built with IBM Watson®
Type here to search
27°C Cloudy
ENG
21:48
05-11-2022


Output
127.0.0.1:5000
BANK
Luna
General Query
General Query
Select the general queries listed below
Select an option
Bank Working Days
List of Branches
Storage Locker Facility
CIBIL Score
Find a Nearest Branch
Type something...
Built with IBM Watson®
Type here to search
27°C Cloudy
ENG
21:49
05-11-2022


Output
127.0.0.1:5000
BANK
Luna
General Query
General Query
Select the general queries listed below
Select an option
Account
Did you mean:
Current
Savings
Greeting
Options
Loan
None of the above
Type something...
Built with IBM Watson®
Type here to search
27°C Cloudy
ENG
21:49
05-11-2022


Output
127.0.0.1:5000
BANK
Luna
Regular Savings Account
Kids Savings Account
Zero Balance Savings Account
Zero Balance Savings Account
Great! Please take the following documents and provide them to the nearest branch.
1. Aadhar card
2. Pan Card
3. Passport Size Photos
Do you want to know about some other services?
Yes
No
Type something...
Built with IBM Watson®
Type here to search
27°C Cloudy
ENG
21:50
05-11-2022



PROJECT: https://chatbotprojectibm.000webhostapp.com/

PREVIEW OF CHATBOT:

https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImag eURL=https%3A%2F%2Fus-south.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-785992fb-b6cf-4d51-b222-23f37f3cee20%3A%3A33c532ec-f7b3-46f0-becb- d89ad77b3d68&integrationID=fafa4141-555c-427c-9e44-66a101cbb178&region=us-south&serviceInstanceID=785992fb-b6cf-4d51- b222-23f37f3cee20