

## Project Development Phase

### Delivery of Sprint - 4

Date	04 November 2022
Team ID	PNT2022TMID03423
Project Name	AI-based discourse for Banking Industry

### Creating Assistant & Integrate With Flask Web Page

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

1. The Bot should be able to guide a customer to create a bank account.
2. The Bot should be able to answer loan queries.
3. The Bot should be able to answer general banking queries.
4. The Bot should be able to answer queries regarding net banking.
5. 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 a user interface.

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

### Build Python Code

#### 1: Importing Libraries

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

```
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 the current module (`__name__`).

#### 2: Creating our flask application and loading

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

### 3: Routing to the Html Page

Here, the declared constructor is used to route to the HTML page created earlier.

The '/' route is bound with the bot function. Hence, when the home page of a web server 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 web page.
- Here, we have created 1 HTML page-Chatbot.html
- Chatbot.html displays the home page which integrates with Watson Assistant.
- A simple HTML page is created. Auto-generated source code from IBM 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.py resides.
- Now type the "python app.py" command.
- It will show the local host where your app is running on <http://127.0.0.1:5000/>
- Copy that localhost URL and open that URL in the browser. It does navigate me to where you can view your web page.

Then it will run on localhost:5000

The image displays three sequential screenshots of a web browser window, showing the development and deployment of a chatbot interface for a banking industry discourse page.

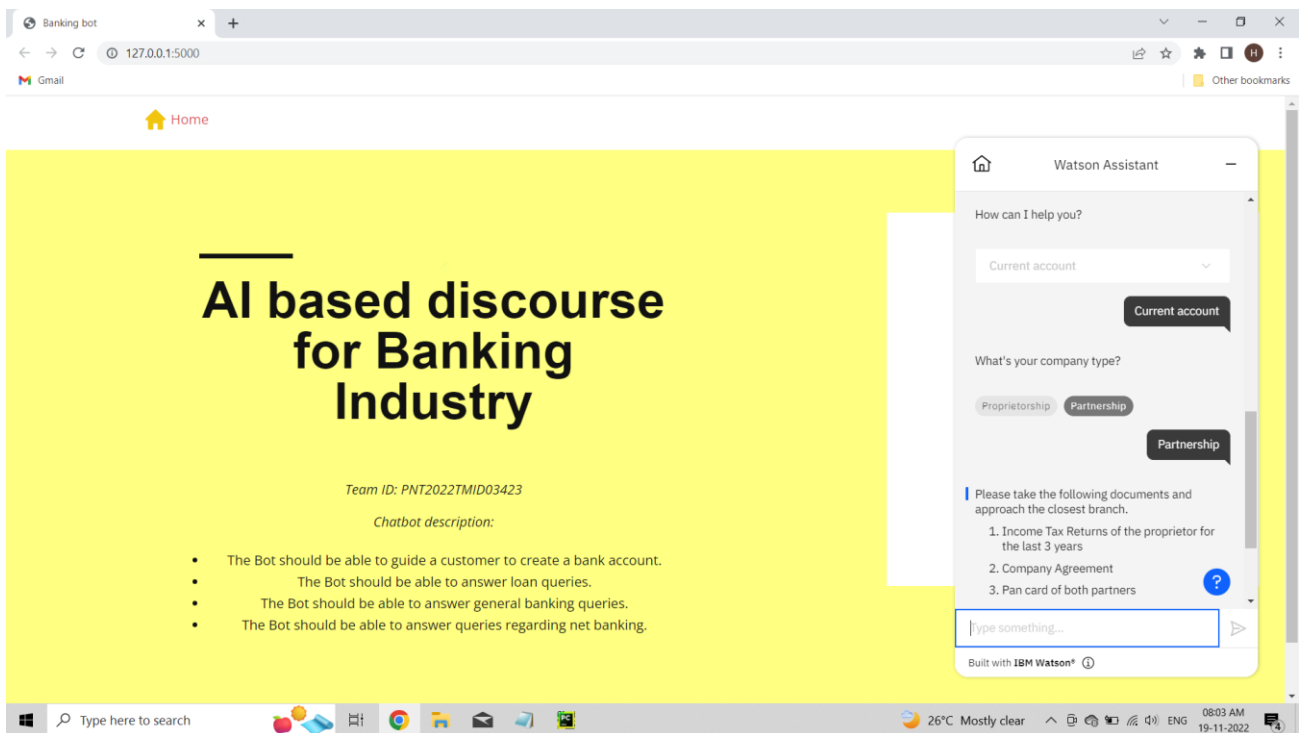
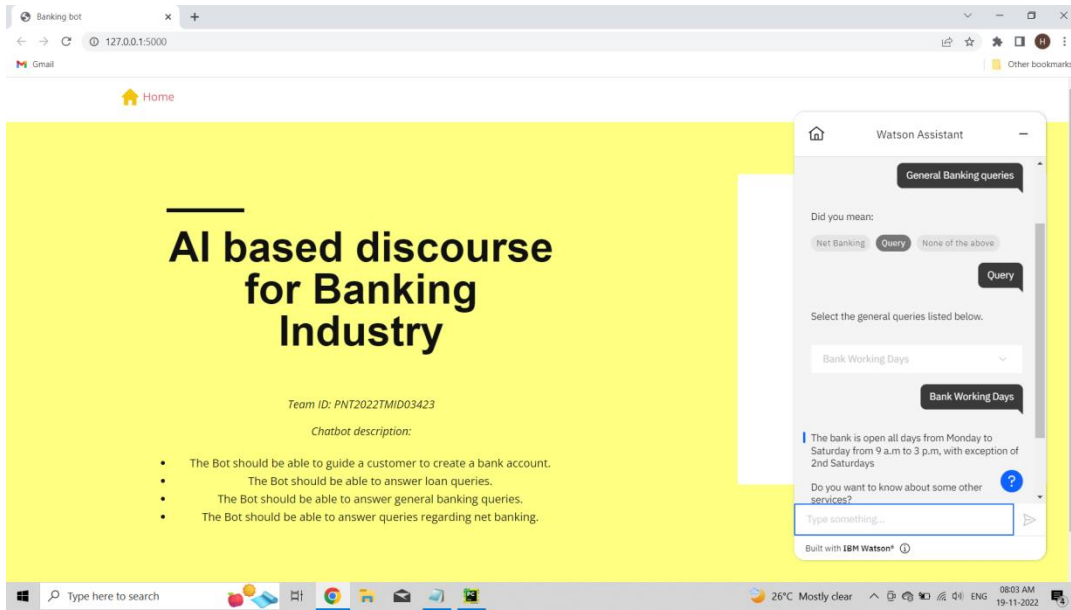
**Page Content (Common to all screenshots):**

- Page Title:** AI based discourse for Banking Industry
- Team ID:** PNT2022TMID03423
- Chatbot description:**
  - 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.

**Screenshot 1 (Top):** The page is displayed with a yellow background. A blue chatbot icon is visible in the bottom right corner.

**Screenshot 2 (Middle):** The page is displayed with a yellow background. A blue chatbot interface is overlaid on the right side of the page. The interface includes a greeting: "Hi! I'm a virtual assistant. How can I help you today?". Below the greeting are three buttons: "Open a new bank account", "General Banking queries", and "Net Banking". A text input field with a placeholder "Type something..." and a send button is at the bottom. The interface is labeled "Built with IBM Watson®".

**Screenshot 3 (Bottom):** The page is displayed with a yellow background. A white chatbot interface is overlaid on the right side of the page. The interface includes a header "Watson Assistant" and a button "Open a new bank account". Below the button is a message: "Kindly reach out to our customer care executive Contact Us @924xx78159". A text input field with a placeholder "Type something..." and a send button is at the bottom. The interface is labeled "Built with IBM Watson®".



Banking bot x +

127.0.0.1:5000

Gmail Other bookmarks

Home

# AI based discourse for Banking Industry

Team ID: PNT2022TMID03423

Chatbot description:

- 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.

Watson Assistant

Yes

How can I help you?

Loan enquiry

Loan enquiry

What type of loan are you looking at?

House loan

House loan

To be eligible for a house loan please contact our bank service providers with all existing loan details.

Type something...

Built with IBM Watson®

Type here to search

26°C Mostly clear

08:04 AM 19-11-2022

Banking bot x +

127.0.0.1:5000

Gmail Other bookmarks

Home

# AI based discourse for Banking Industry

Team ID: PNT2022TMID03423

Chatbot description:

- 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.

Watson Assistant

Net Banking

Net Banking

What queries do you have regarding NetBanking?

What is Net Banking?

How do I register for Net Banking?

What are the features of Net Banking?

Facing errors in Net Banking

How do I register for Net Banking?

Please download and fill up the net banking requisition form and submit it to your home branch.

Type something...

Built with IBM Watson®

Type here to search

26°C Mostly clear

08:04 AM 19-11-2022

### **PREVIEW OF CHATBOT:**

<https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageUrl=https%3A%2F%2Fus-south.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-23571e7e-fdd2-47af-b02a-45f7c14411ae%3A%3A6ffbfbfc1-64a0-4bb7-925c-e0f76ece7593&integrationID=d365db8b-9408-418f-a61b-e18a5f564312&region=us-south&serviceInstanceID=23571e7e-fdd2-47af-b02a-45f7c14411ae>

**Source code is attached in Final Deliverables.**

**Note:** No code for this project. So, I attached the screenshot and step to build it.