



DHANALAKSHMI SRINIVASAN COLLEGE OF ENGINEERING

AI BASED DISCOURSE FOR BANKING INDUSTRY

TEAM ID: PNT2022TMID43887

INDUSTRY MENTOR : DURGA PRASAD

FACULTY MENTOR : INDHU D

Team Leader : INIYAVAN D

Team Members : STELLA GRACY S

DEVIGA M

EZHIL ARASI N

PROJECT REPORT

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

. 2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

. 3.2 Ideation & Brainstorming

. 3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

. 5.2 Solution & Technical Architecture

. 5.3 User Stories

6. PROJECT PLANNING & SCHEDULING.

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

. 9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

INTRODUCTION

OVERVIEW :

Industries are forced to evolve and update their practices due to technological advances and the contemporary market. The banking sector is one of the most developed sectors and is always looking for the latest technological solutions that improve its efficiency.

Net banking websites are complex and involve navigating through a lot of pages to find the information you need. Bank staff undergoes a lot of stressful situations when communicating with clients directly. Such situations can be avoided gracefully by using chatbots.

Only 32% of companies in the finance industry currently use AI chatbots, and 37% are planning to start using them within 18 months said a report from Salesforce. This results in a potential growth rate of 118% which indicates the demand in the industry.

A smart chatbot takes a query from the user in natural language and gives the appropriate response for the same. This paper aims to discuss the relevance of chatbots in the banking sector and explore how chatbots can be implemented using natural language processing techniques that can be used in the banking industry.

LITERATURE SURVEY:

EXISTING PROBLEM:

This paper [1] presents the use of the RASA framework for building smart context-remembering chatbots, it also describes how Rasa NLU works and how its performance is elevated by using intent recognition and entity extraction. It also compares the accuracies of entity extraction using Rasa NLU and a NN, results show Rasa NLU performs better to extract entities when whole sentences are provided as compared to neural networks which require segmented inputs. This paper discusses Rasa by implementing a

chatbot related to the finance domain, using which the users can inquire about stock-related information.

RASA NLU can introduce a vital component in intelligent chatbot systems. We can compose the system to extract the entity after intent recognition. This can be further improved for complicated sentences and more entities.

This paper [2] briefly discusses advancements in the field of AI and how this has led to major shifts in some organizations about how they operate. It further mentions how the banking industry has moved to use chatbots for providing an interface to customers so that they can have an assistant throughout the day for service. This paper also gauges the ability of current chatbots to provide all the services that a user needs.

It includes several strategies for managing dialogue in the banking and finance industry based on ontology. Although further use of AI can make the chatbot not only respond to questions but also self-learning to improve itself in more stages, improving user service quality and also reducing human load.

REFERENCE:

- [1] Jiao, Anran. (2020). An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network. Journal of Physics: Conference Series. 1487. 012014. 10.1088/1742-6596/1487/1/012014.
- [2] Fathima, Sasha & Student, Suhel & Shukla, Vinod & Vyas, Dr Sonali & Mishra, Ved P. (2020). Conversation to Automation in Banking Through Chatbot Using Artificial Machine Intelligence Language. 10.1109/ICRITO48877.2020.9197825.
- [3] Singh, Netra & Singh, Devender. (2019). Chatbots and Virtual Assistant in Indian Banks. Industrija. 47. 75-101. 10.5937/industrija47-24578.
- [4] Petr Lorenc, “Joint model for intent and entity recognition” in arXiv:2109.03221v1 [cs.CL] 7 Sep 2021
- [5] The Rasa documentation. [Online]. Available: <https://rasa.com/docs/rasa/2.x>
- [6] Django documentation. [Online]. Available: <https://docs.djangoproject.com/en/4.0/>

- [7] Adamopoulou, Eleni, and Lefteris Moussiades. "An overview of chatbot technology." In IFIP International Conference on Artificial Intelligence Applications and Innovations, pp. 373-383. Springer, Cham, 2020.
- [8] Cahn, Jack. "CHATBOT: Architecture, design, & development." University of Pennsylvania School of Engineering and Applied Science Department of Computer and Information Science (2017).
- [9] Hien, Ho Thao, Pham-Nguyen Cuong, Le Nguyen Hoai Nam, Ho Le Thi Kim Nhung, and Le Dinh Thang. "Intelligent assistants in higher- education environments: the FIT-EBot, a chatbot for administrative and learning support." In Proceedings of the ninth international symposium on information and communication technology, pp. 69-76. 2018.
- [10] Waterman, Chinia. 2018. "Consumer Online Banking Trends 2018". Humley. <https://humleyai.com/2018/09/18/consumer-online-banking-trends-2018/>.
- [11] "Chatbots, A Game Changer For Banking & Healthcare, Saving \$8 Billion Annually By 2022". 2017. Juniperresearch.Com. <https://www.juniperresearch.com/press/chatbots-a-game-changer-for-bankinghealthcare>.

PROBLEM STATEMENT :

Banking is an important industry since it deals with monetary operations that anyone may use, yet banks are frequently unable to satisfactorily respond to client questions about their goods or services, which lowers customer satisfaction. An intelligent system must be implemented to help clients navigate all of the financial services the bank offers in order to offer people the best possible option. The users are customers of banks who require a service that is available around-the-clock to answer all of their questions and direct them through the various banking procedures. Therefore, a better and more intelligent method of communicating with clients must to be developed to ensure effective service delivery.

It is intended to serve as the all-encompassing virtual assistant that enables users to ask banking related queries without going to the bank or calling customer care centres and to offer pertinent recommendations.

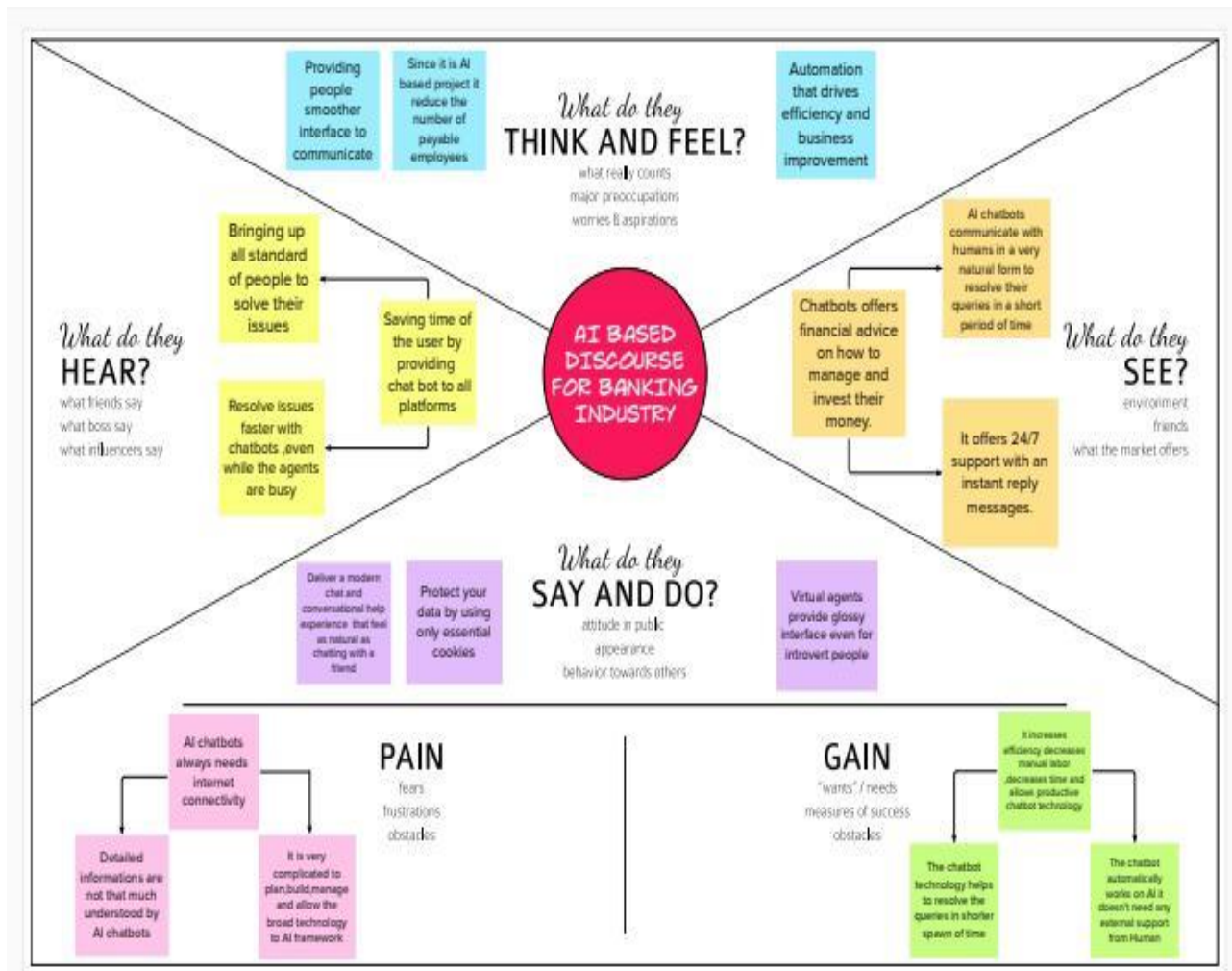
| | |
|---|--|
| Who does the problem affect? | Bank Clients |
| What are the boundaries of the problem? | Clients who have queries about banking or are attempting to use several bank services |
| What is the issue? | For simple questions, customers must routinely visit banks, Banks are unable to effectively respond to large numbers of customer inquiries |
| When does the issue occur? | When a client physically visit a bank |
| Where does the issue occur? | It happens in the banking sector |
| Why is it important that we fix the problem ? | It quickly and effectively responds to customers questions in a way that is economical |
| What solution to solve this issue? | Any common banking questions including account creation, loans, net banking, other services, etc. Should be able to be answered by a chatbot. AI |

| | |
|---|---|
| | chatbots can assist clients in efficiently and rapidly finishing their work |
| What methodology used to solve the issue? | Artificial Intelligence imitates the human brain to make chatbot conversations more realistic |



IDEATION AND PREPOSED SOLUTION :

EMPATHY MAP :



IDEATION AND BRAINSTORMING

Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare

1 hour to collaborate

2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#)

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM STATEMENT

The aim of our project is to build a chatbot using IBM Watson's assistant. This chatbot should be able to answer any general banking queries on account creation, loan, net banking, other services etc. The chatbot should provide 24/7 customer support with all the necessary data for solving their queries which reduces their time on moving to banks directly.

Key rules of brainstorming

To run a smooth and productive session

Stay in topic.

Encourage wild ideas.

Defer judgment.

Listen to others.

Go for volume.

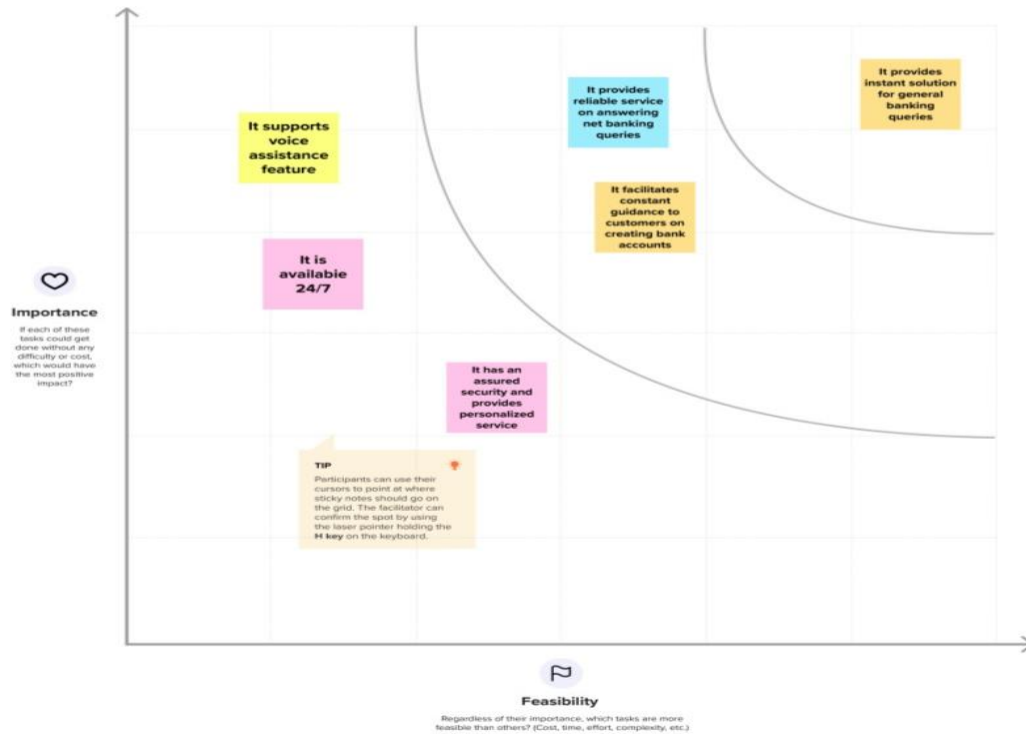
If possible, be visual.

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

🕒 20 minutes



Idea 1: It provides instant solution regarding general banking, financial, loan queries.

Providing a solution instantly is one of the key reasons behind creating a chatbot. It must be able to understand and address the queries of the customer effectively and as quick as possible. It must be available 24x7 for customers to access and use.

Idea 2: It provides a secured environment for the banking transaction.

The chatbot must provide an environment in which customers conversation, banking transactions, transactions histories should be maintained safe and secure.

Idea 3: It facilitates constant guidance and reliable service to customers on. creating bank accounts and net banking.

To support and provides guidance to customer on creating account and using net. banking facility to make the services reliable and available 24x7 to answer all the queries.

PREPOSED SOLUTION :

✓ The solution to the problem is Artificial intelligence in the banking sector makes banks efficient, trustworthy, helpful, and more understanding. It is strengthening the competitive edge of modern banks in this digital era. The growing impact of AI in banking sector minimizes operational costs improves customer support and process automation.

✓ Nearly 40% to 50% of financial and banking service providers are using AI in their processes to harness the power of next-generation AI capabilities. The companies believe that AI is the future of banking sector which can perform a range of banking operations in faster, easier, and more secure ways.

✓ AI banking Chatbots help customers in many ways. AI-based chatbot service for financial industry is one of the significant use cases of AI in banking sector. AI chatbots in banking are modernizing the way how businesses provide services to their customers

✓ AI chatbots in the banking industry can assist customers 24*7 and give accurate responses to their queries. These chatbots provide a personalized experience to users.

✓ AI chatbots in banking is providing a better customer experience.

✓ Hence, AI chatbots for banking and finance operations let banks attract customer attention, optimize service quality, and expand the brand mark in the market.

PROBLEM SOLUTION FIT :

YOUR SOLUTIONS

This problem can be solved by using an Automated solution, such as Chatbot, which can handle all simple queries

You could reduce your employee's workload by having Chatbot handle all of the simple customer requests

It understands human languages and assist them in text-based communication

PROBLEMS/PAINS

Limited response

Need to be Maintained

Misreads the customer's query

Unsuitable for outdated customer's

Losing Customer insights

CUSTOMER'S LIMITATION

Misunderstood the customers query, Internet Access is required, Duted Mobile Experiences

Bank's

AVAILABLE SOLUTIONS

Simple Banking queries can be resolved quickly Sieves lot of times, 24/7 Availability.

PROBLEM ROOT/CAUSE

Stow response from Human agent

Guiding customer create Bank Account

Limited only on working days

Longer to resolve complaints

Waiting in queue for assistance

Cannot able to a queries repeatedly

BEHAVIOR

Guiding customer create Bank Account

Limited only on working days

Longer to resolve complaints

Waiting in for assistance

Cannot able to a queries repeatedly

Answer loan Queries

A rower general Banking queries

Answer queries regarding Net Banking

Automated customer service

CUSTOMER SEGMENTS

Bank's Customer

REQUIREMENT ANALYSIS :

FUNCTIONAL REQUIREMENTS:

Functional Requirements describe the service that the banking management system must offer, they are subdivided into three access levels: Admin Mode, Teller Mode, and Customer Mode:

CUSTOMER:

Sign in with login and password.

Update personal details.

Change password.

View balance.

View personal history of transactions.

Transfer money.

Withdraw.

Submit Cash.

TELLER:

Sign in with login and password.

Change password.

Register new bank customers.

View customer information.

Manage customer accounts.

ADMIN:

Sign in with login and password.

View manager and customer details.

Add or update bank branch details.

Add or update manager details.

The use case diagram provides an overview of the basic services:

Use-Case diagram

NON-FUNCTIONAL REQUIREMENTS:

Non-functional requirements specify criteria that can be used to judge the operation of a system as a whole rather than specific behaviours. They describe emergent properties like security, performance, and availability and, unlike the functional requirements that can be worked around, are essential to fulfill for a usable system. The estimation of whether the product fulfills the non-functional requirement or not usually reduces to a Boolean

answer: yes or no.

For a bank management system, the most important non-functional requirements include security, performance, usability, and availability.

SECURITY

Bank management systems are notorious for being subject to malicious attacks, so security is the major requirement for the system. Unauthorized access to the data is not permissible. The data must be backed up daily and stored in a secured location, at a distance from different facilities of the system.

Online transactions and stored digital files must be encrypted according to 128-bit or 256-bit AES encryption standards. The system also must employ firewall software as a defence against network attacks.

From the client-side, the system must provide an automatic log-out after an inactivity period, accept only secure passwords that have sufficient length and non-alphabetic characters, and block login attempts after several unsuccessful trials.

PERFORMANCE

The bank management system is a multi-client system that must reach response time targets for each of the clients during simultaneous calls and must be able to run a target number of transactions per second without failure. The system must

effectively utilize the hardware and energy resources to minimize operational costs.

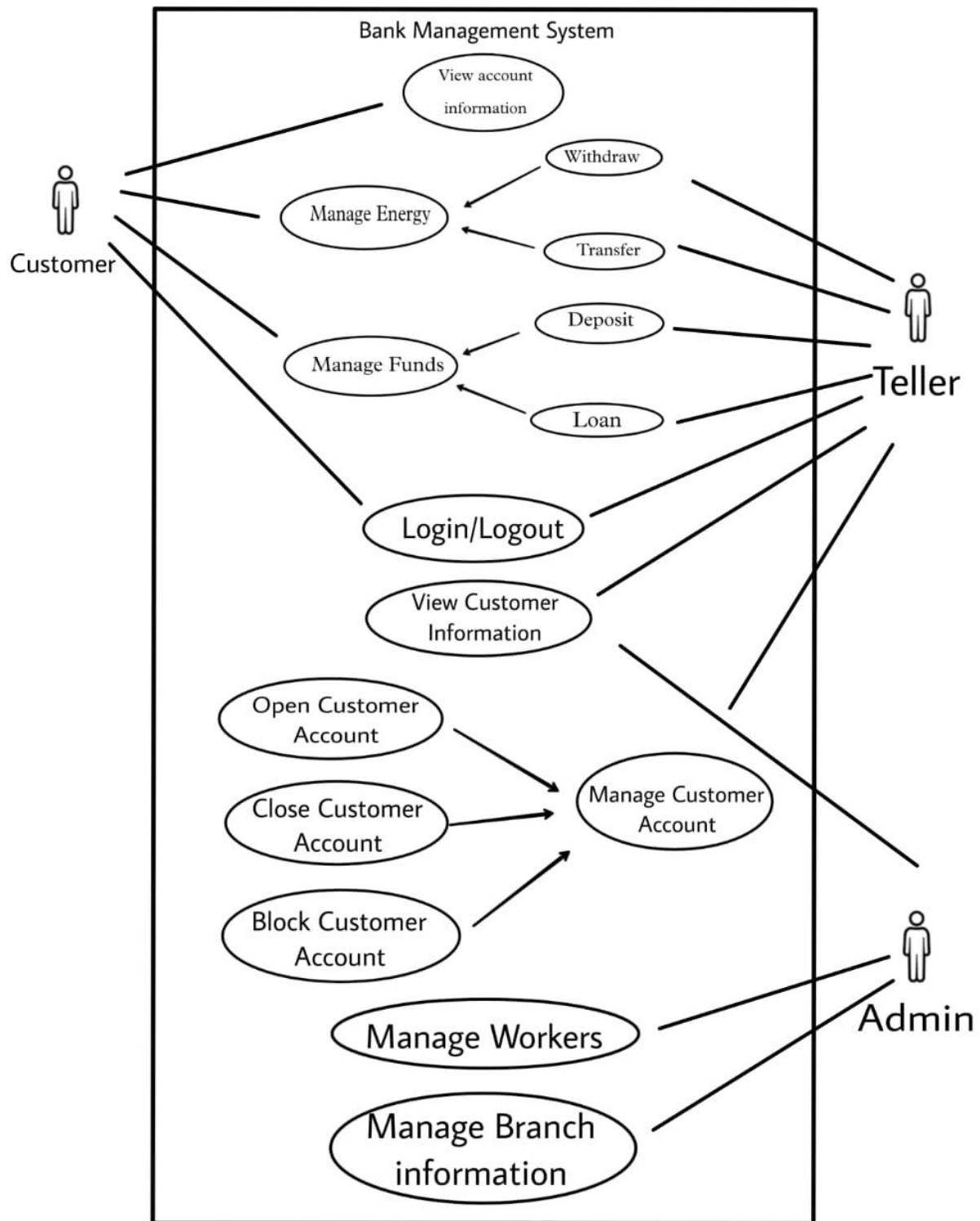
USABILITY

The system must provide different graphical interfaces for customers, tellers, and admins. All system interfaces must be user-friendly and simple to learn, including helping hints and messages and intuitive workflow, especially in a client interface: the client must be able to fast learn and use the interface without prior knowledge of banking terminology or rules.

The interfaces must automatically adjust to devices with different screen sizes, and allow to change typeface size and colour scheme to improve readability.

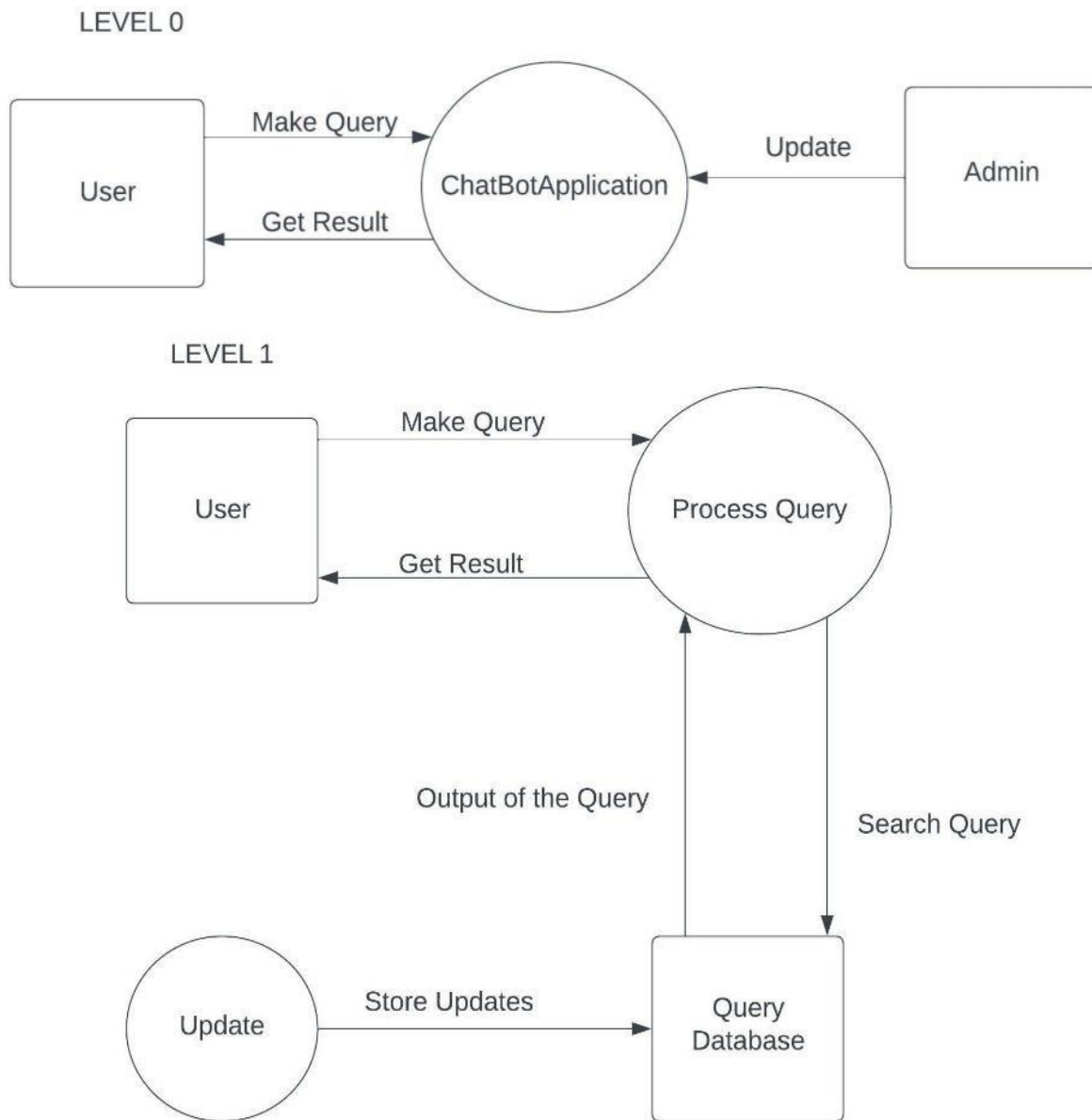
Availability

The system must be available during bank working hours. The mobile banking and ATM must be available round-the-clock with minimal maintenance times, reaching 99.999% availability time per year.

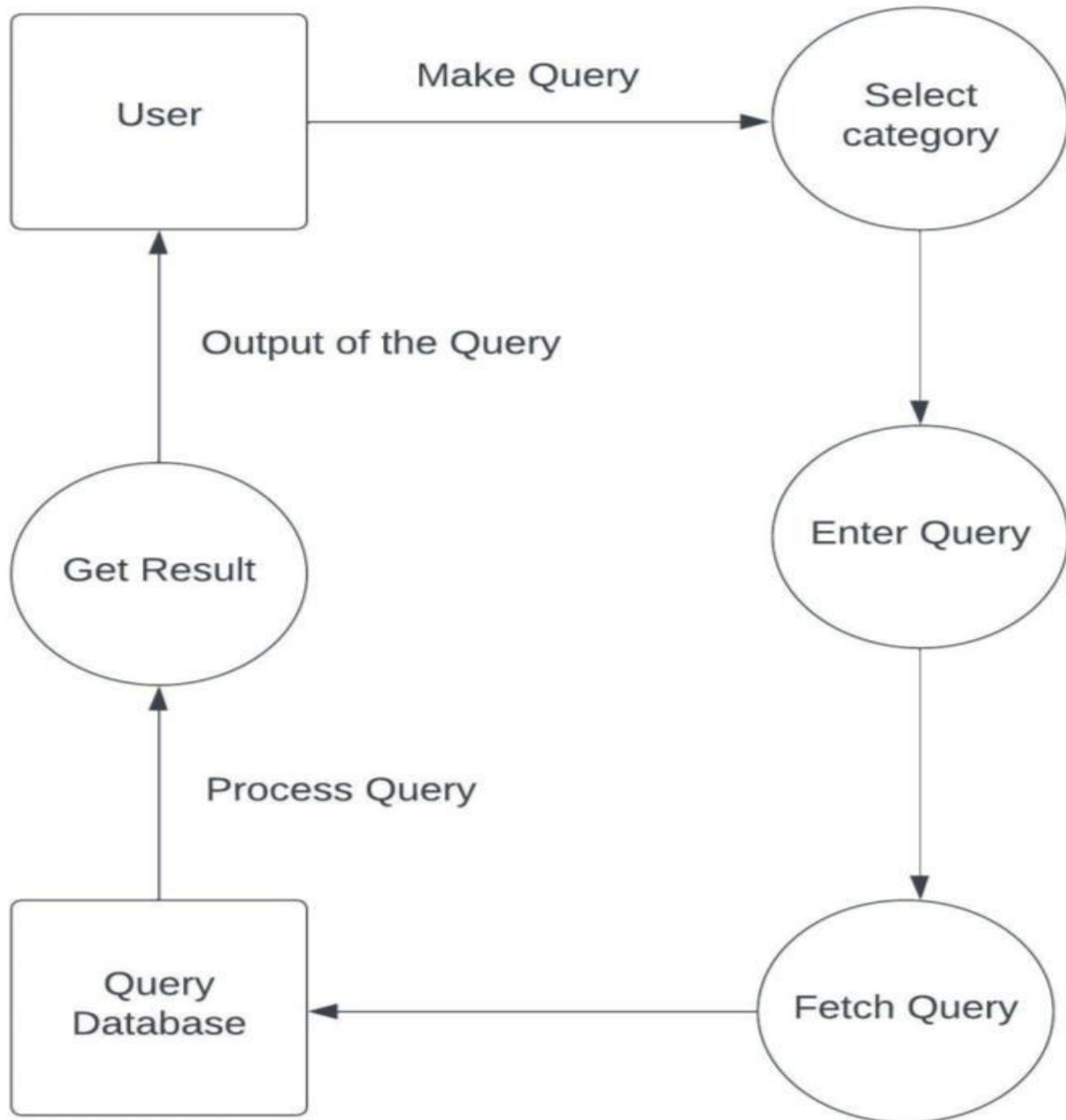


PROJECT DESIGN :

DATA FLOW DIAGRAM :

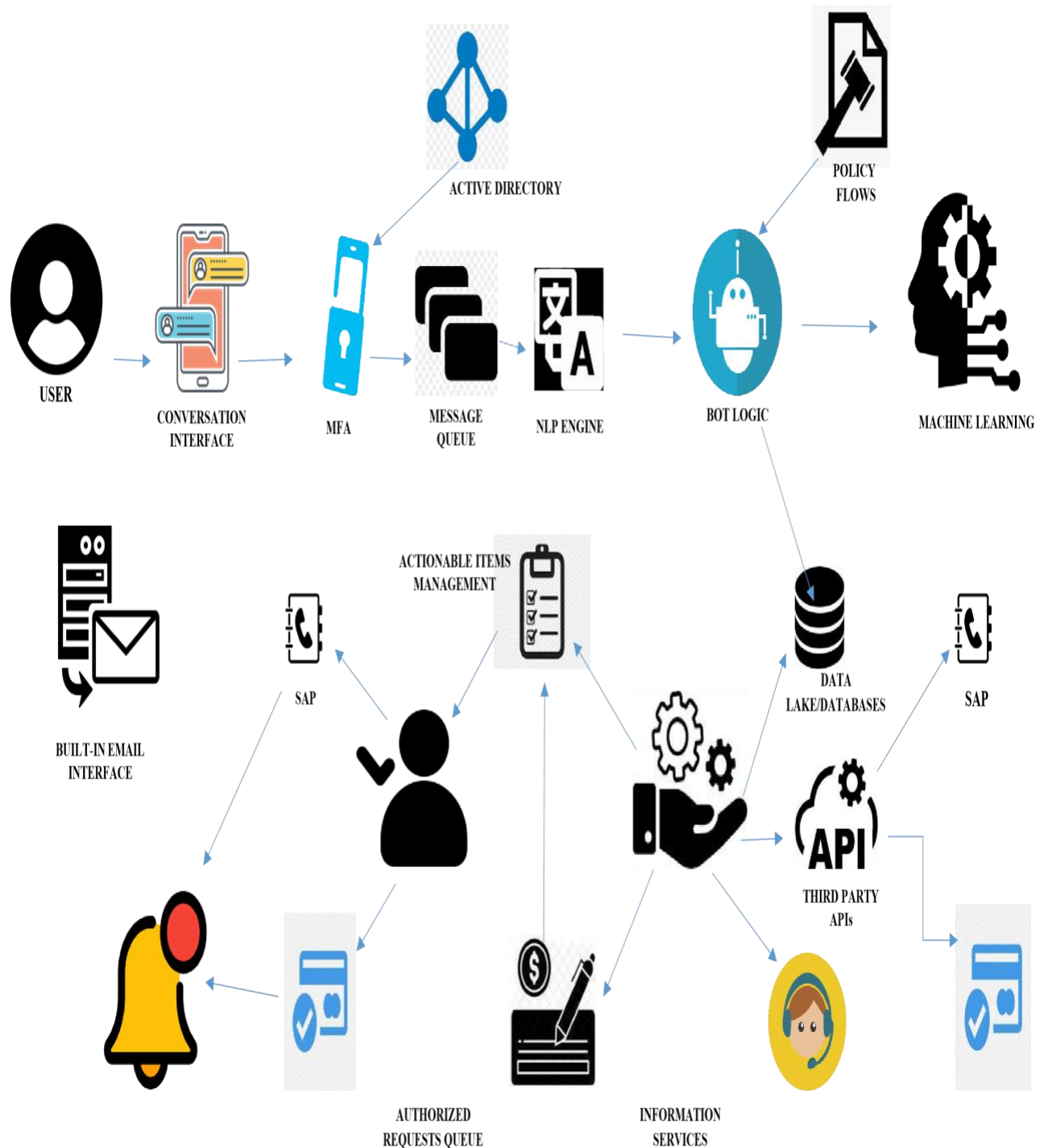


LEVEL 2



SOLUTION AND TECHNICAL ARCHITECTURE :

SOLUTION ARCHITECTURE :



TECHNICAL ARCHITECTURE STEPS:

1. User queries to the Chat Bot
2. Bot previews the query
3. Query is transferred to Watson Assistant
4. Natural Processing Language is used to understand the query
5. Watson finds the relevant response from cloud database
6. Queries and responses (sent and received) is stored in cloud database
7. All queries and related information is sent to the bank for improvement

TECHNICAL ARCHITECTURE :

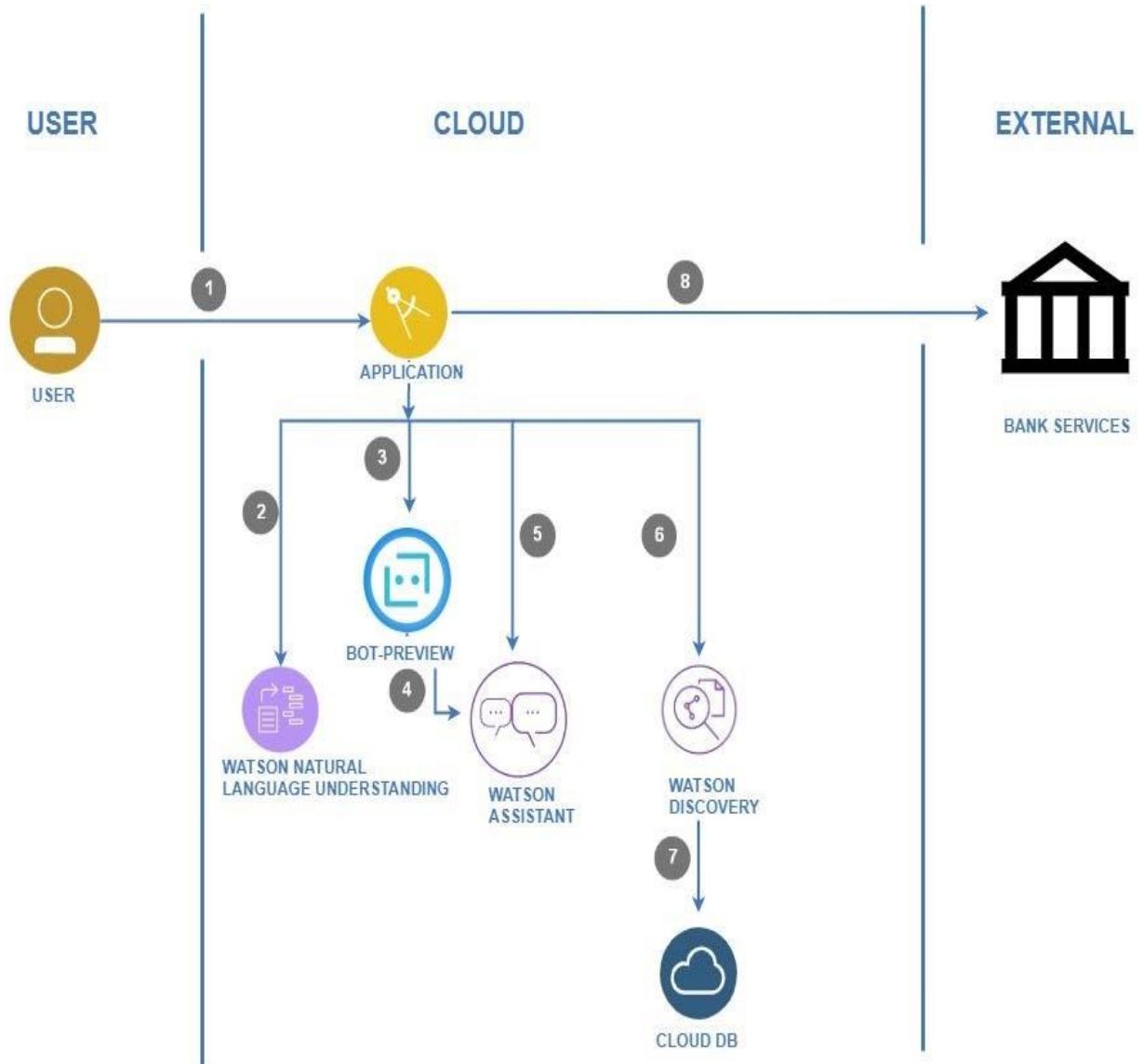


Table 1: Components and Technologies

| S. No | Component | Description | Technology |
|--------------|----------------------|--|-------------------------|
| 1 | Bot Previews | A simple page is presented to the user with a chat layout that has an input box field available to get user queries and present options for the user to select | HTML, CSS, Java Script |
| 2 | Application Logic- 1 | An input bar is provided that enables the user to type queries | Java/Python |
| 3 | Application Logic-2 | Regularly asked queries or options are presented for the user | IBM Watson STT service |
| 4 | Application Logic-4 | Processes responses to customers queries and displays a relevant response | IBM Watson Assistant |
| 5 | Cloud Database | Queries and answers to queries are stored in the cloud and are accessed whenever a query is asked | IBM cloudant DB |
| 6 | External API -1 | It provides an interface between the application and the cloud to send the query from the application to the cloud | Watson Assistant v2 API |
| 7 | External API -2 | A cloud based API that supports several cloud based applications and operations | IBM Cloud API |
| 8 | Deep learning model | It is trained with several queries and uses that knowledge to provide relevant | Deep Learning |

| | | | |
|---|-------------------------------|--|-------------------------|
| | | responses to queries with a good enough accuracy | |
| 9 | Infrastructure (server/cloud) | Application Deployment on local system cloud Local Server Configuration : Flask application | Python Flask, IBM Cloud |

Table 2 : Application and Characteristics

| S. No | Characteristics | Description | Technology |
|-------|-------------------------|--|---|
| 1 | Open-source Frameworks | List the open source frameworks used | Python Flask, CSS Frameworks |
| 2 | Security Implementation | General access control and the built in security features of IBM Cloud are present | IBM Watson Assistants, IBM Cloudant DB |
| 3 | Scalable Architecture | The architecture consist of three tiers, the client side, the web server and the cloud server. Each of these can be scaled as per requirements | Client Side :Flask(Python) Web Server :IBM Watson Assistant Cloud Server :IBM Cloud |
| 4 | Availability | The chatbot is available 24/7 on almost all devices that support an internet browser | IBM Cloud, Flask (Python) |
| 5 | Performance | Responds to several thousands of queries at the same time | IBM Load Balancer, IBM Cloud |

USER STORIES :

AI CHATBOT

It is one of the best advantages of using Artificial Intelligence in the banking sector. AI banking Chatbots help customers in many ways. AI-based chatbot service for financial industry is one of the significant use cases of AI in banking sector. AI chatbots in banking are modernizing the way how businesses provide services to their customers.

AI chatbots in banking industry can assist customers 24*7 and give accurate responses to their queries. These chatbots provide a personalized experience to users. Hence, AI chatbots for banking and finance operations let banks attract customer attention, optimize service quality, and expand the brand mark in the market.

AI ENHANCES CUSTOMER EXPERIENCE

AI banking apps can do wonders. AI mobile banking apps for Android/iOS are aimed to improve customer experiences and service quality. Implementation of AI and Machine Learning in banking help companies in tracking user behavior and delivering highly personalized services to customers.

Intelligent mobile apps using ML algorithms can monitor user behavior and derive valuable insights based on user search patterns. These insights would help service providers in providing personalized recommendations to end-users.

Hence, 70% of the banks are looking ahead to integrating AI in mobile banking apps and stepping forward to embrace the golden opportunities of AI in banking industry.

BRINGS AUTOMATION & MAKES THE PROCESS SEAMLESS

The use of Artificial Intelligence in Banking will accelerate automation and make your process seamless.

Automation is one of the best AI use cases in finance and banking sector. AI has great potential in the banking industry. AI software helps banks in streamlining and automating every task which is done by humans and making the entire process simple and virtual.

Therefore, AI applications can reduce the workload of bankers and optimize the quality of work. Through customized AI banking apps and AI Chatbot services, users can request service at any time and get accurate responses from AI virtual banking assistants all the time

DATA COLLECTION & ANALYSIS

There are many fold benefits of AI in Banking and Finance and automated data collection and analysis is one of them.

Artificial intelligence in the banking sector can efficiently perform data collection and analysis processes. AI machines process massive data sets and extract valuable insights into data. This analysis will help banks to predict the future of their business and market trends with ease.

Further, customer data analysis through AI-powered mobile banking apps will also play a vital role in delivering personalized services and enhancing the overall user experience. Moreover, banks can also make effective business decisions with the insights derived from the customer data and offer them more personalized service recommendations.

AI FOR PORTFOLIO MANAGEMENT

It is one of the best benefits of AI in Banking and Finance sector. Wealth and portfolio management can be done more powerfully with artificial intelligence. It's a fact that advanced technologies bring everything to our fingertips. AI helps those users who cannot visit the banks frequently. This innovative AI technology can manage banking services and strengthen mobile banking operations.

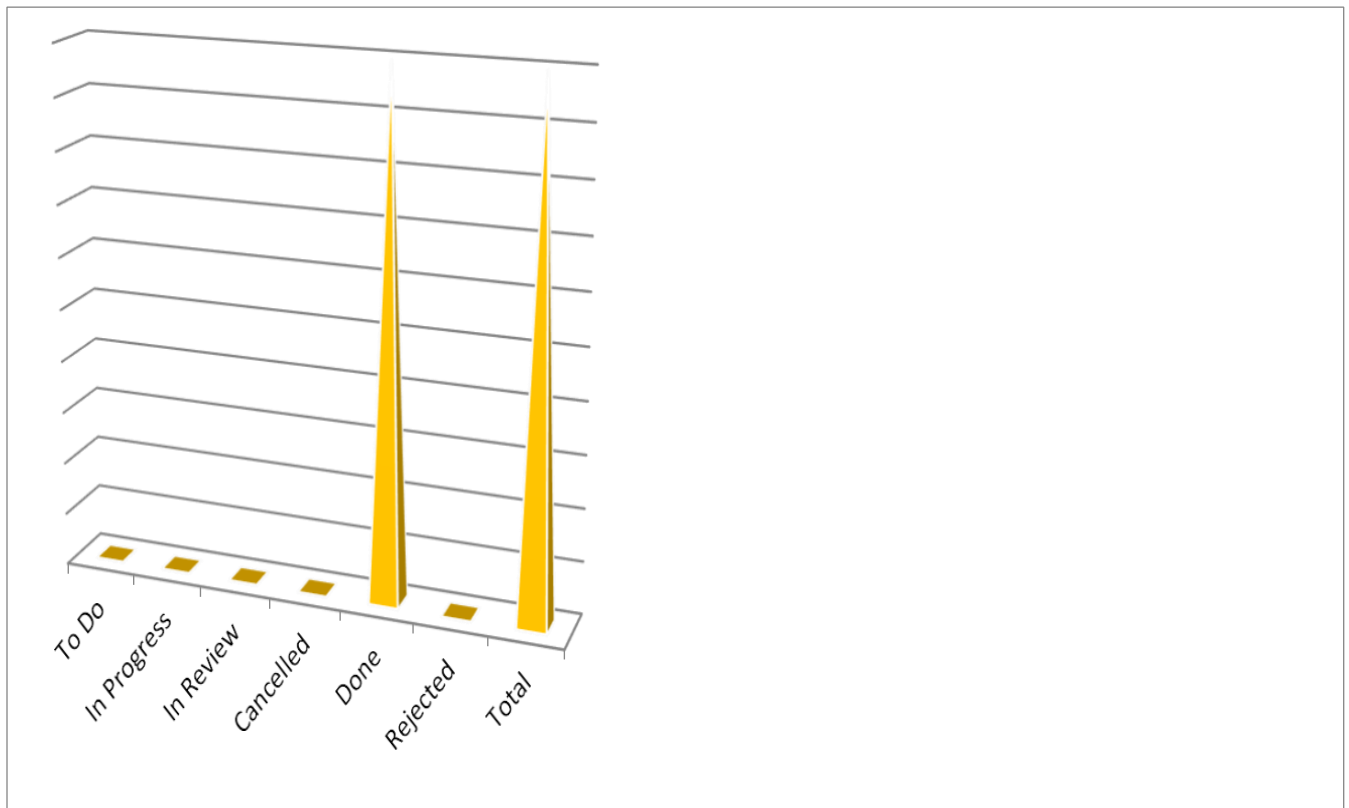
On remitting money through digital banking apps, AI apps will track and send immediate transaction alerts to the users, if they trigger any suspicious transactions. AI machines immediately alert the user. Hence, AI ensures safe transactions

On the other hand, AI also plays a crucial role in the debit/credit card management system. It can automate the credit and debit card management system and makes the process safer. Artificial intelligence technology in banking eases the card authentication process and makes transactions safe and secure. Hence, AI systems advance mobile banking services.

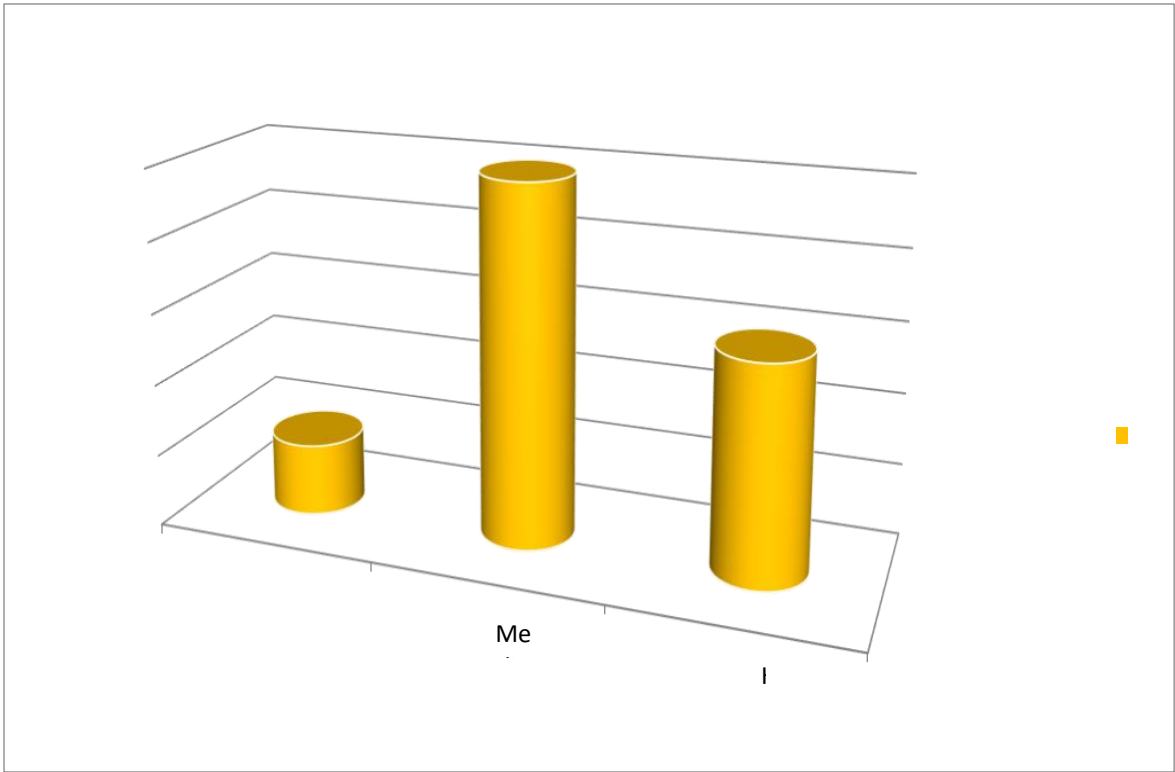
PROJECT PLANNING AND SCHEDULING :

SPRINT PLANNING AND ESTIMATION :

BURNDOWN CHART



PRIORITY BREAKDOWN



SPRINT DELIVERY SCHEDULE

SPRINT 1

IBM Watson Assistant Lite Upgrade Banking ChatBot Learning center

Greeting

Customer starts with:
Greeting:

1 0 0
total step end steps re-ask steps

Conversation steps

1 This step has no content
Continue to next step

New step

Customer starts with:

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 4

Enter a phrase:

Key
Hi
Hello
Greeting

Preview

Type here to search 28°C Partly cloudy 22:13 29-10-2022

IBM Watson Assistant Lite Upgrade Banking ChatBot Learning center

Actions

Actions

Created by you

Set by assistant

Variables

Created by you

Set by assistant

Set by integration

Saved responses

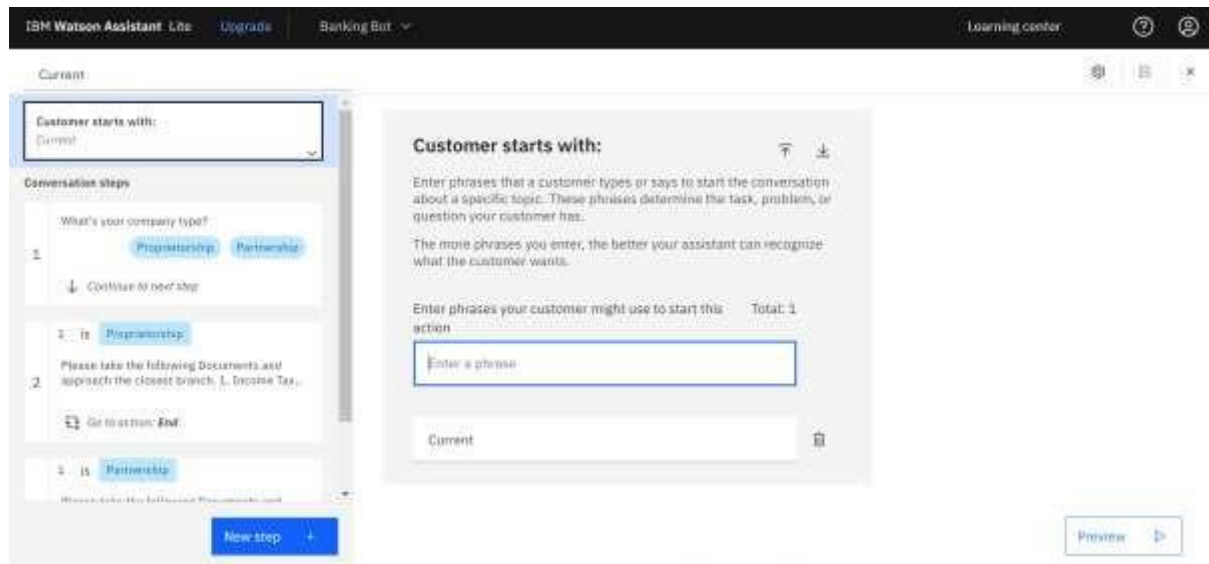
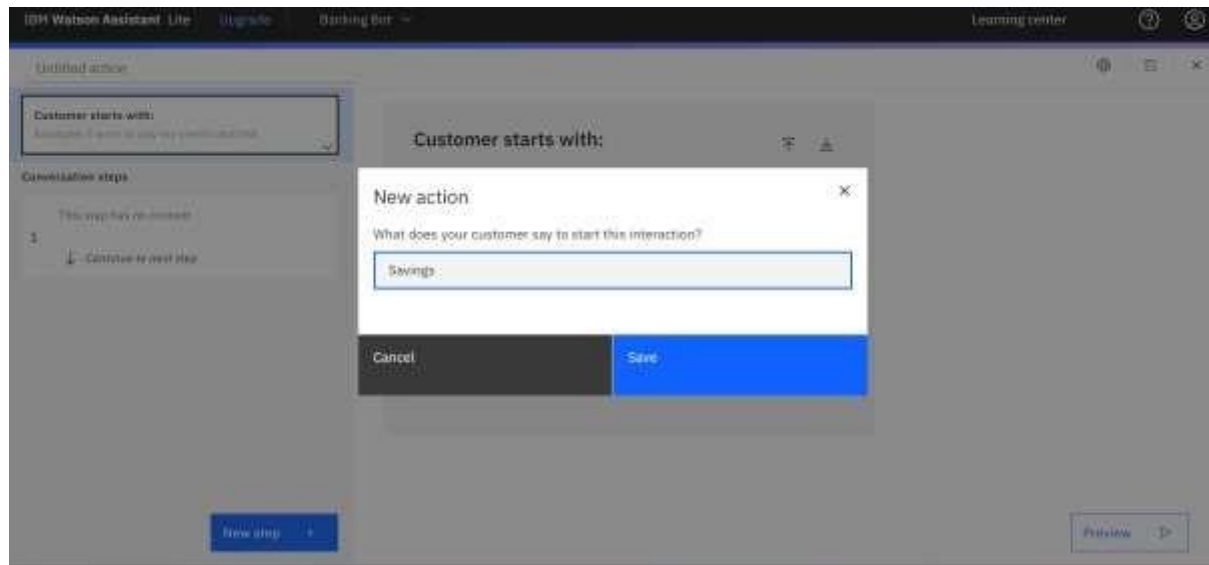
| Name | Last edited | Status |
|----------|---------------|--------|
| Index | 6 minutes ago | ✓ |
| Greeting | 2 minutes ago | ✓ |

Items per page: 50 Showing 1-2 of 2 actions 1 1 of 1 pages













Preview





Type here to search 28°C Partly cloudy 22:27 29-10-2022

SPRINT 2



SPRINT 3

| | | |   New action + |
|--------------|-------------|---|--|
| Name | Last edited | Status | |
| Current | 2 days ago |  |  |
| Index | 2 days ago |  |  |
| Register | 3 days ago |  |  |
| Greeting | 2 days ago |  |  |
| End Greeting | 2 days ago |  |  |

| | | |   New action + |
|---|-------------------|---|--|
| Name | Last edited | Status | |
| Net Banking | 3 minutes ago |  |  |
| End | 2 days ago |  |  |
| Loan | 2 days ago |  |  |
| Query | a few seconds ago |  |  |
| Savings | 16 minutes ago |  |  |
| Current | 2 days ago |  |  |
| Items per page: 50 ▾ Showing 1–10 of 10 actions | | | 1 ▾ 1 of 1 pages   |

SPRINT 4

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 FLASH ALLOCATION 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.

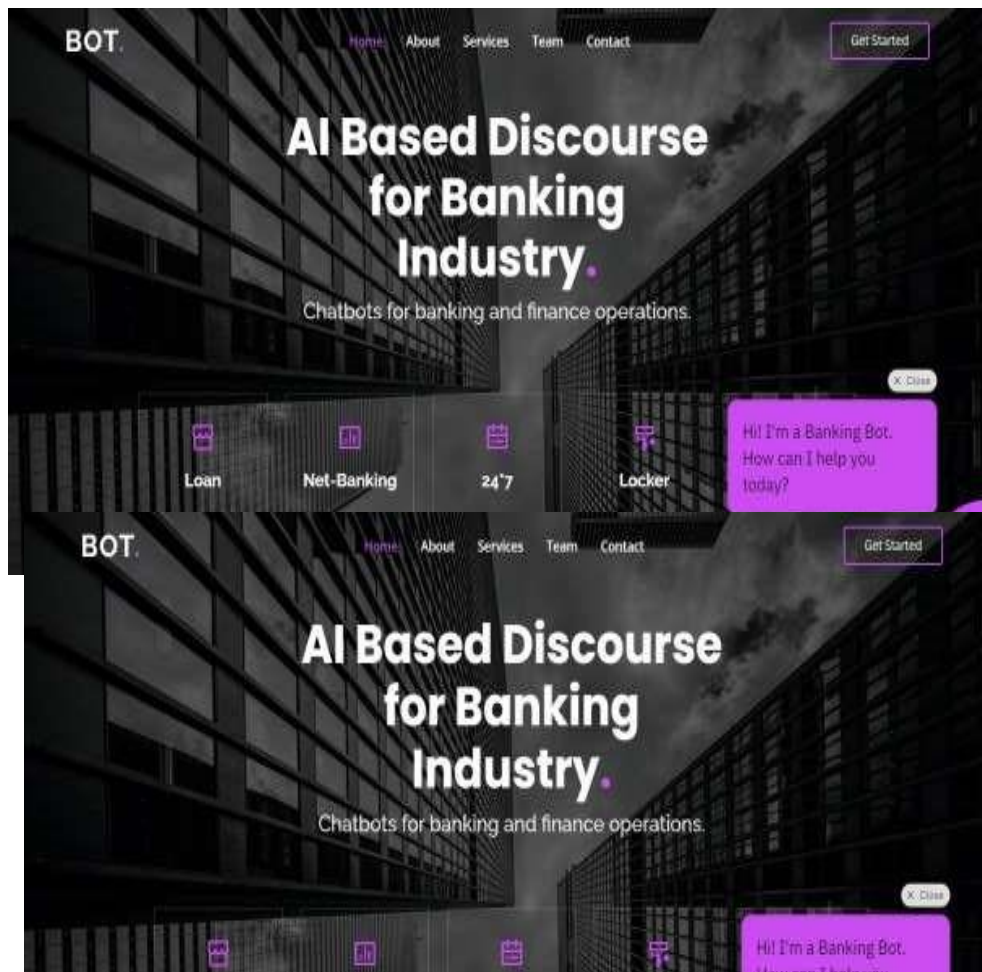
```
if __name__ == '__main__':  
    app.run()
```

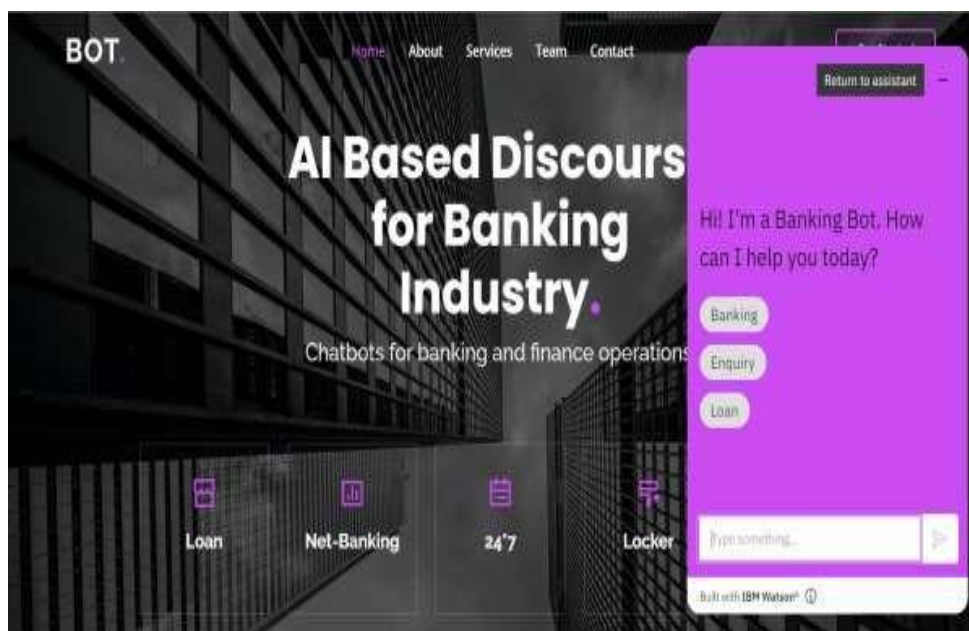
BUILD HTML CODE

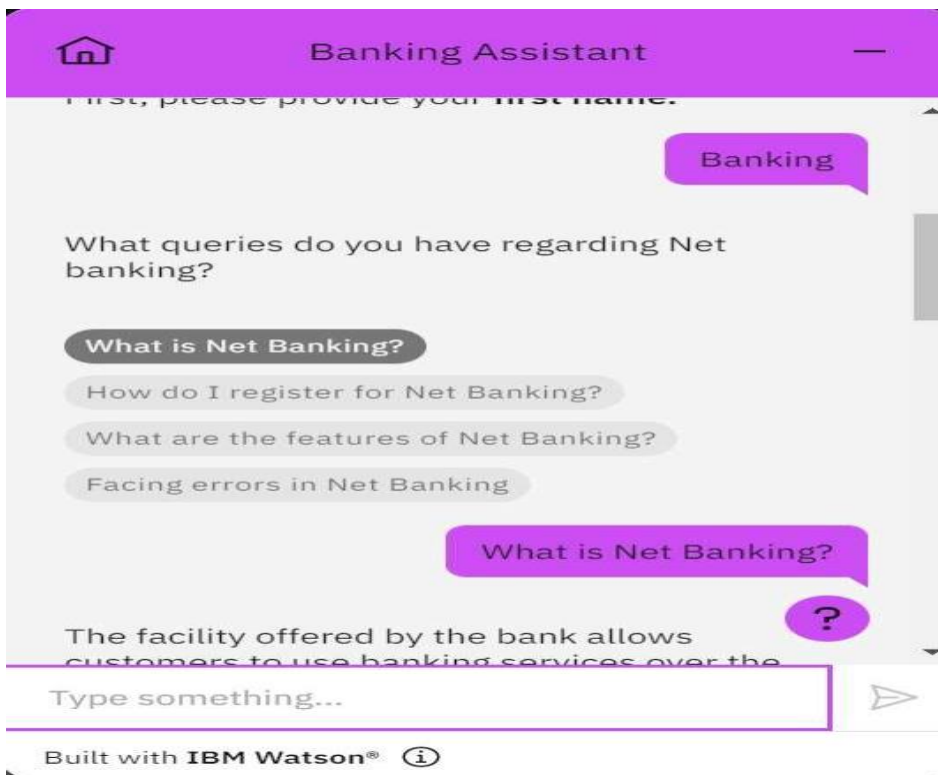
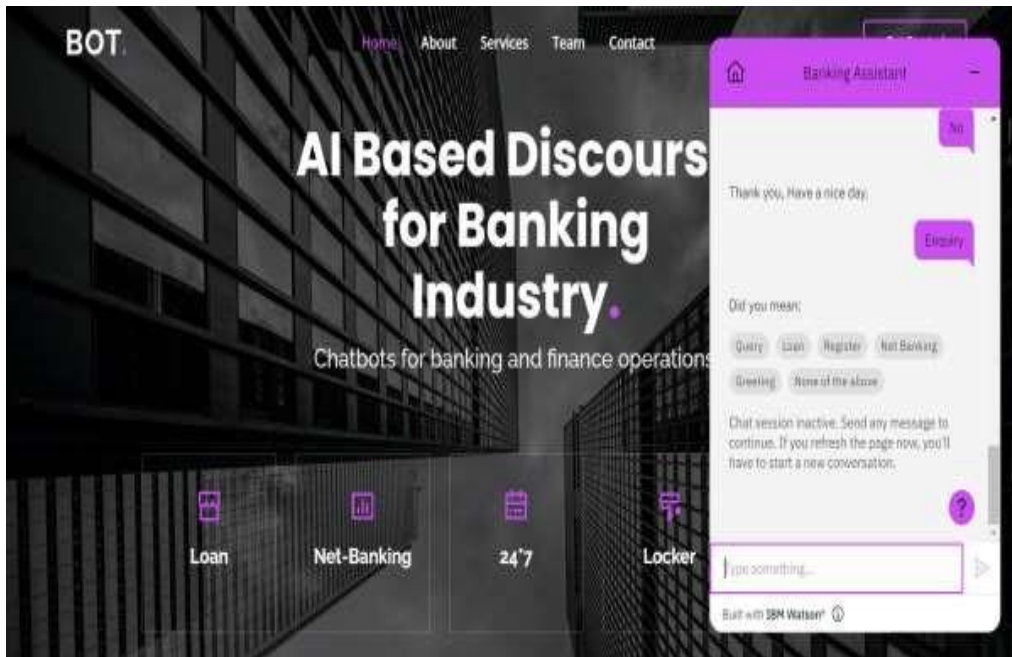
- We use HTML to create the front-end part of the web page.
- Here, we have created 1 HTML page-Chatbot.html
- Chatbot.html displays the home page which integrates with Watson Assistant.
- A simple HTML page is created. Auto-generated source code from IBM Watson Assistants is copied and pasted inside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

RUN THE APPLICATION

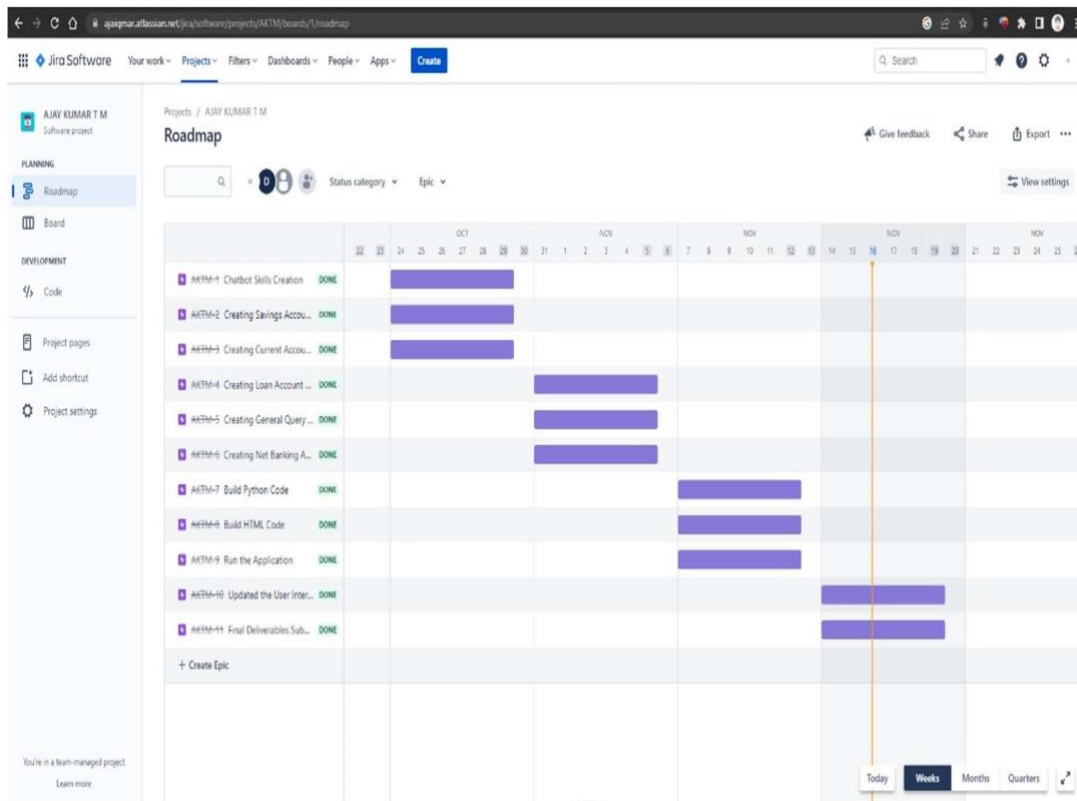
- Open the anaconda prompt from the start menu.
- Navigate to the folder where your app.py resides.
- Now type the “python app.py” command.
- It will show the local host where your app is running on http://127.0.0.1:5000/
- Copy that localhost URL and open that URL in the browser. It does navigate me to where you can view your web page.





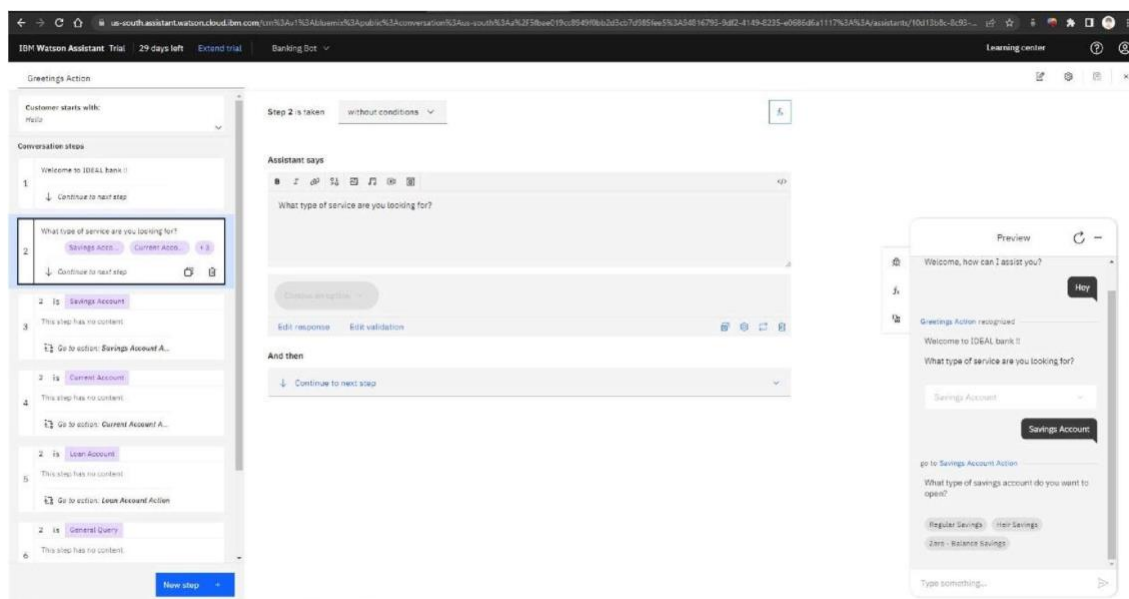


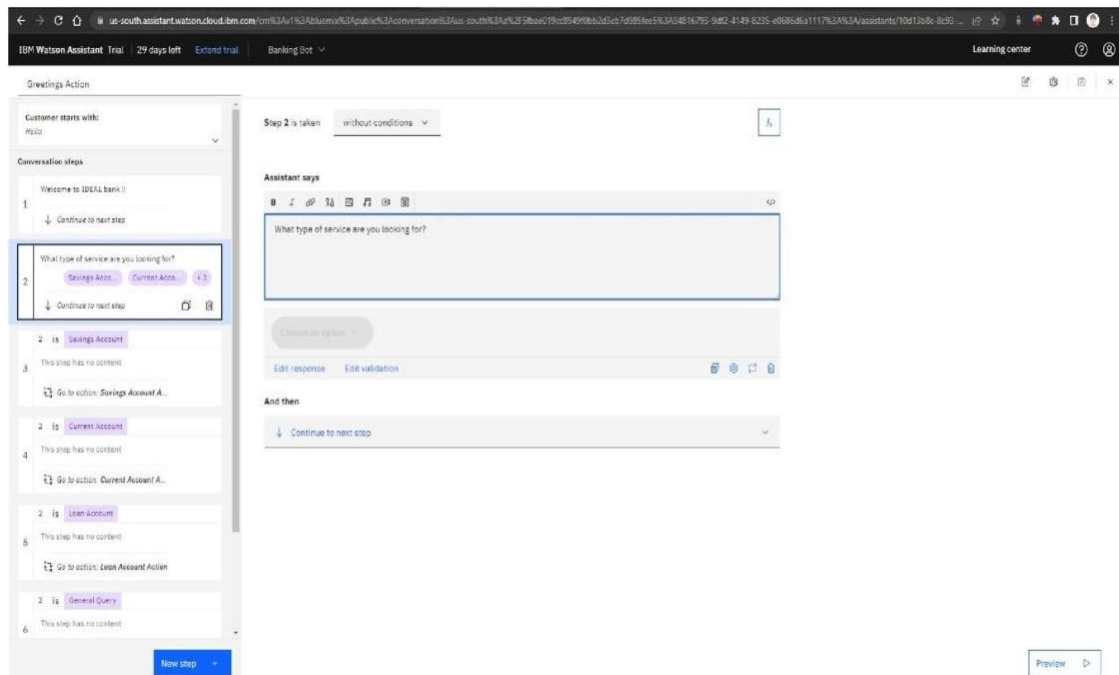
REPORTS FROM JIRA :



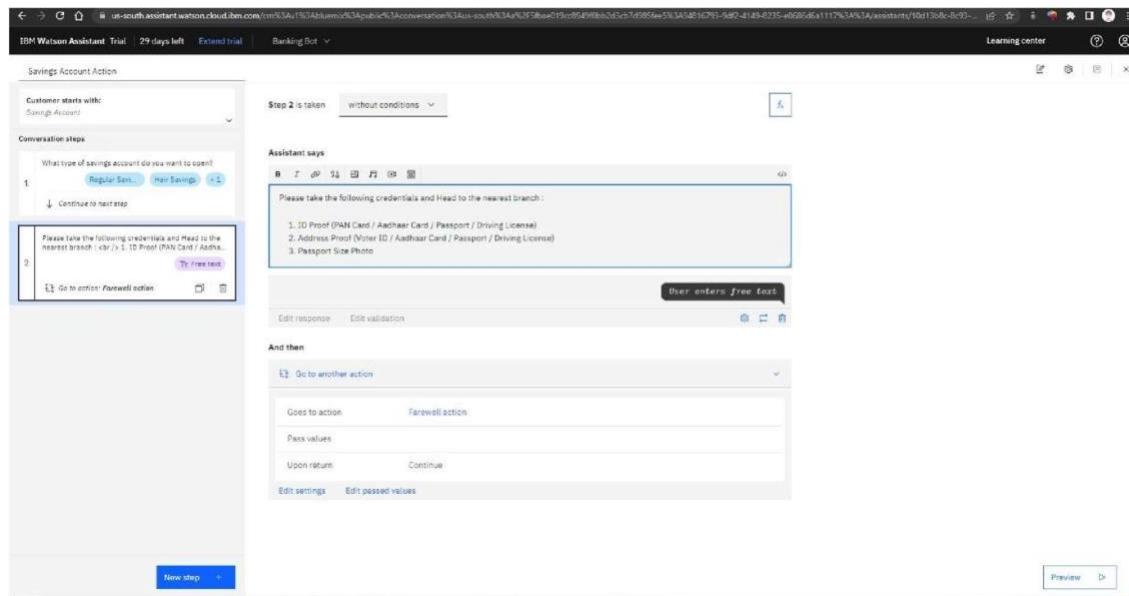
CODING AND SOLUTIONING : FEATURE -1

Service Selection – Lets you select the feature that you want to select and look up



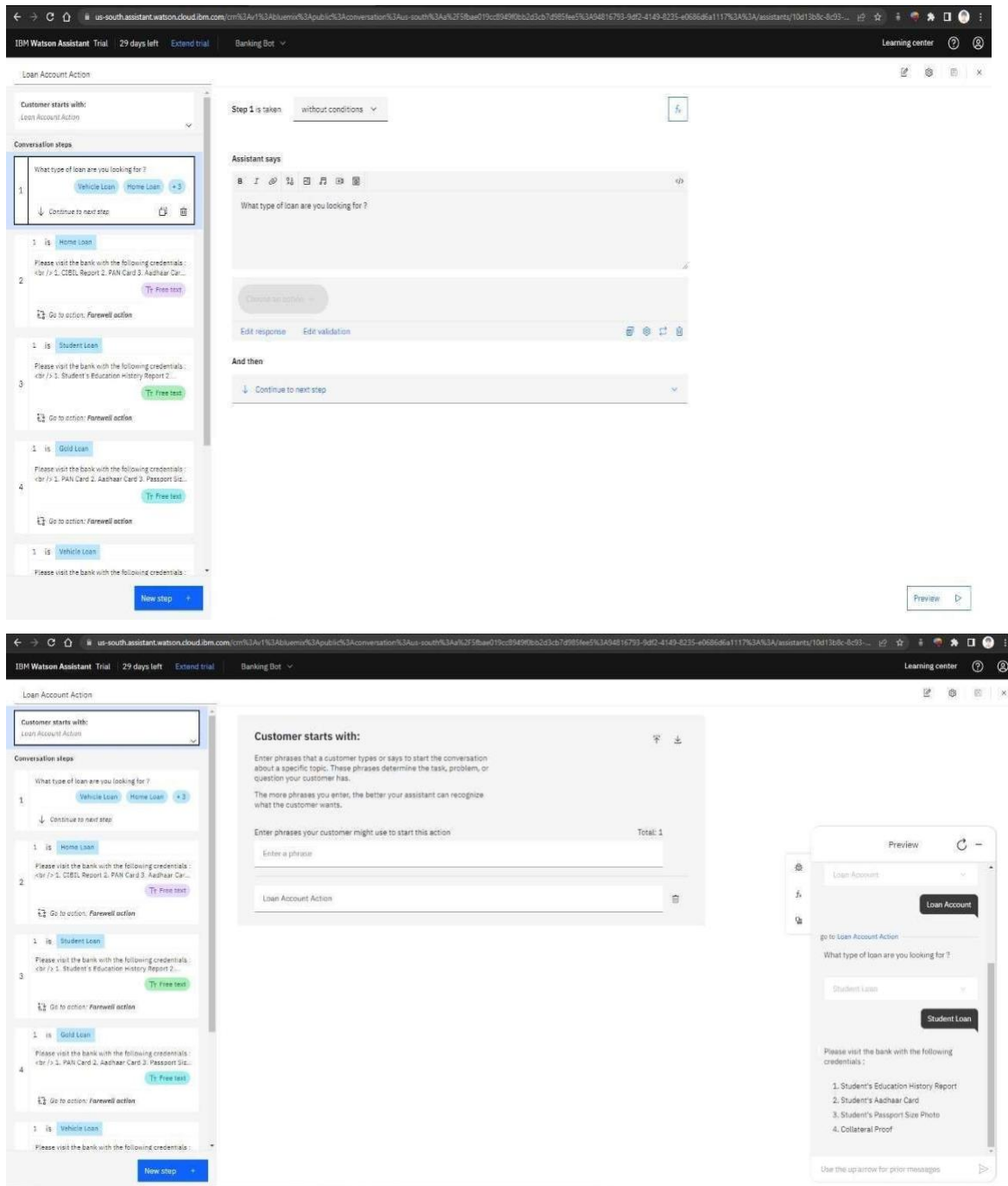


ACCOUNT TYPE REQUISITES

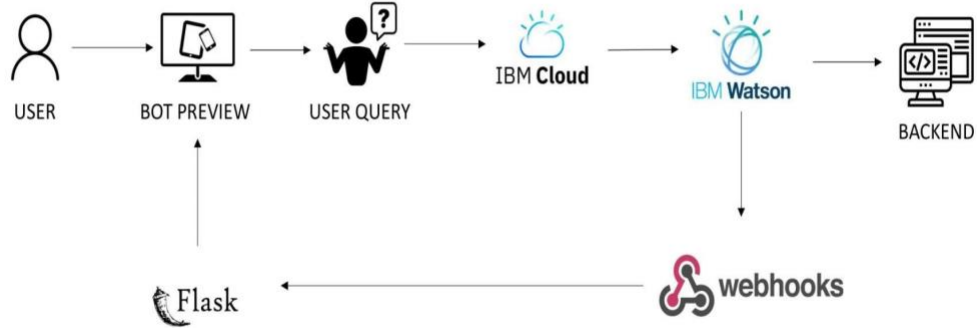


FEATURE -2

LOAN SELECTION



DATABASE SCHEMA :



TESTING :

TEST CASES :

| S. No | Test Cases | Passed/Failed |
|-------|----------------------------|---------------|
| 1 | Providing List of Queries | Passed |
| 2 | Creating a Bank Account | Passed |
| 3 | Prerequisites for Services | Passed |
| 4 | Loan Offers | Passed |
| 5 | Timings and Venue | Passed |
| 6 | Troubleshooting Help | Passed |
| 7 | Login/ Logout | Passed |

USER ACCEPTANCE TESTING :

| S. No | Test cases | Yes/No |
|-------|------------------------------------|--------|
| 1 | Key word driven | Yes |
| 2 | Responds in manually drafted rules | Yes |
| 3 | Manages multiple users | Yes |
| 4 | Conversational paradigm | Yes |
| 5 | Learns from real interactions | Yes |
| 6 | Training via historical data | Yes |
| 7 | Has decision-making skills | Yes |

RESULTS :

PERFORMANCE METRICS

Building the AI bank of the future

AI bank of the future Can banks meet the AI challenge?

Artificial intelligence technologies are increasingly integral to the world we live in, and banks need to deploy these technologies at scale to remain relevant. Success requires a holistic transformation spanning multiple layers of the organization.

Reimagining customer engagement for the AI bank of the future

Banks can meet rising customer expectations by applying AI to offer intelligent propositions and smart servicing that can seamlessly embed in partner ecosystems

AI-powered decision making for the bank of the future

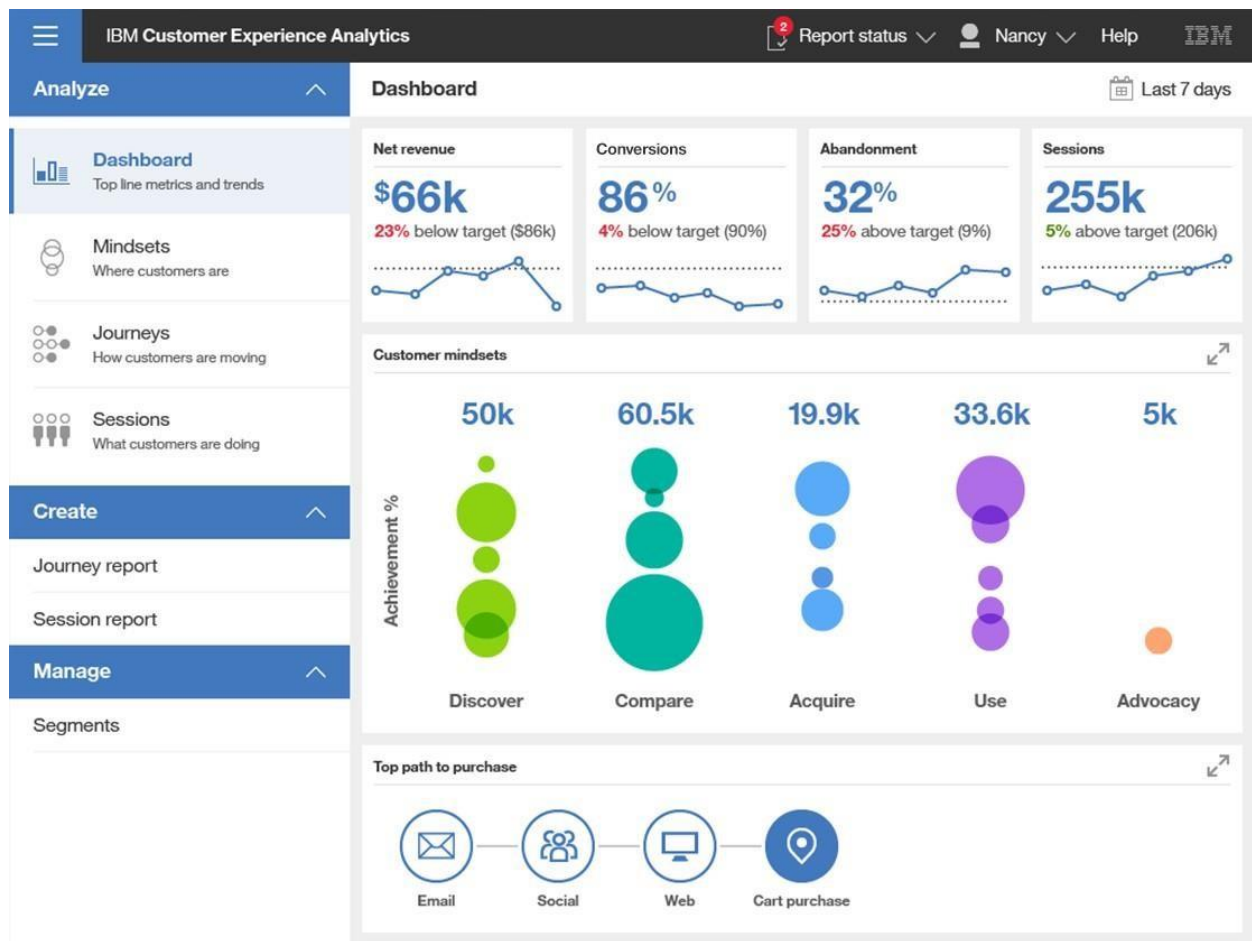
Banks are already strengthening customer relationships and lowering costs by using artificial intelligence to guide customer engagement. Success requires that capability stacks include the right elements.

Beyond digital transformations: Modernizing core technology for the AI bank of the future

For artificial intelligence to deliver value across the organization, banks need core technology that is scalable, resilient, and adaptable. Building that requires changes in six key areas.

Platform operating model for the AI bank of the future

Technology alone cannot define a successful AI bank; the AI bank of the future also needs an operating model that brings together the right talent, culture, and organizational design.



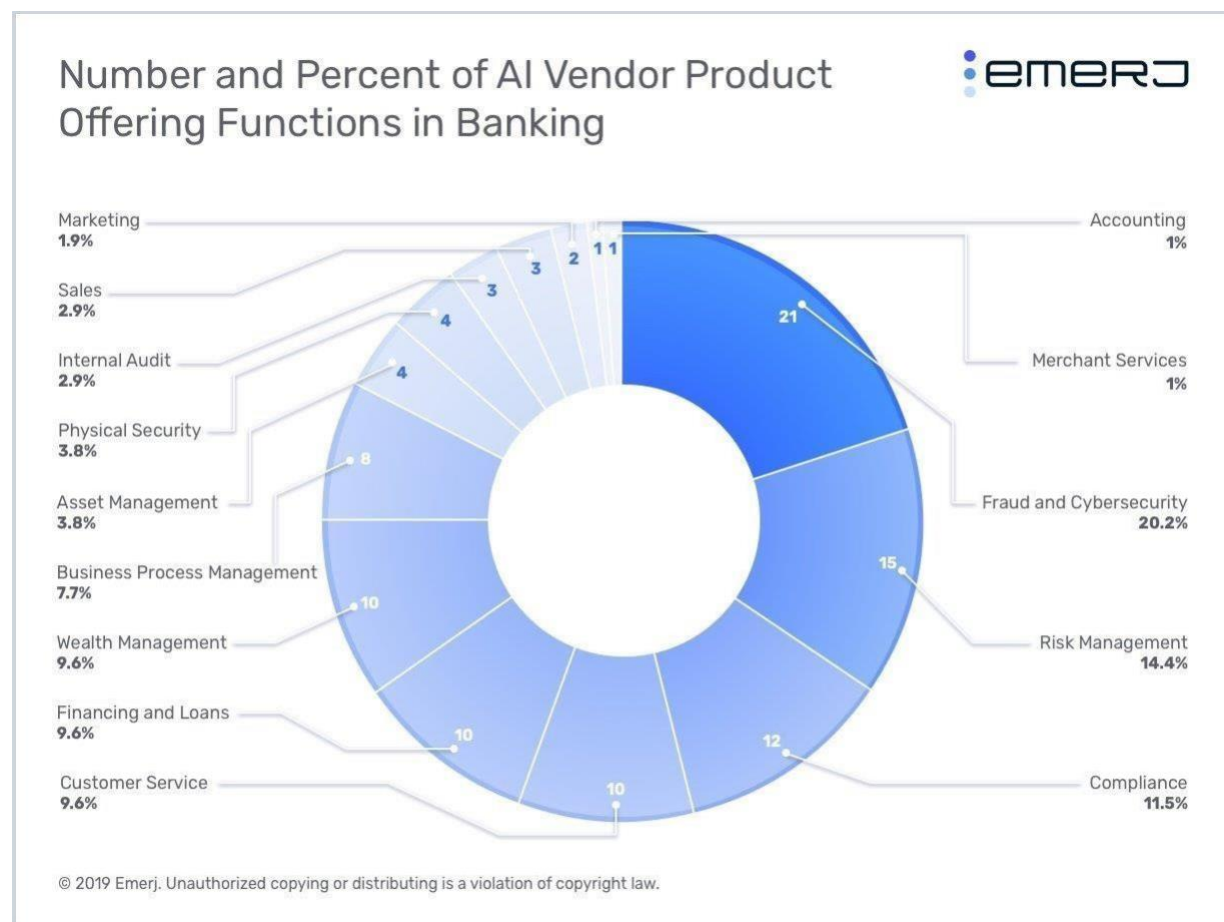
AI in Finance and Banking in India

The Indian banking sector's adoption of artificial ,though in a steep increase. In the past year, global investment in AI applications touched \$5.1 billion, up from \$4.0 billion in 2015.8 while large commercial and investment banks globally are incorporating AI and block chain for both back-office and customer facing purposes, in India, widespread adoption of these technologies has not yet come to fruition.

In the past year, several large financial services companies and Fine Tech start ups have collaborated to conduct proof of concepts (POCs) and implement some of these emergent technologies into their operations. Though the deployment of AI technologies is still nascent in the banking sector, the competitive advantage that the technologies bring has been recognized by banks with some developing 'innovation' and running - these initiatives often take the form of partnerships between banks and Fine Tech companies.

Use of AI in Banking and Finance

The adoption of AI in the banking and finance sector is a part of the larger digital wave occurring within the sector.¹⁰ The use and deployment of AI in consumer banking, financial products and back-end operations is varied and across different stages of operations. Though it is not always clear from publicly available information the exact type of AI technology that is being used, some technologies that we did find in our research include: Natural Language Processing, Natural Language Generation, Machine Learning (such as Neural networks/deep learning), and Computer Vision.



THEORETICAL ANALYSIS:

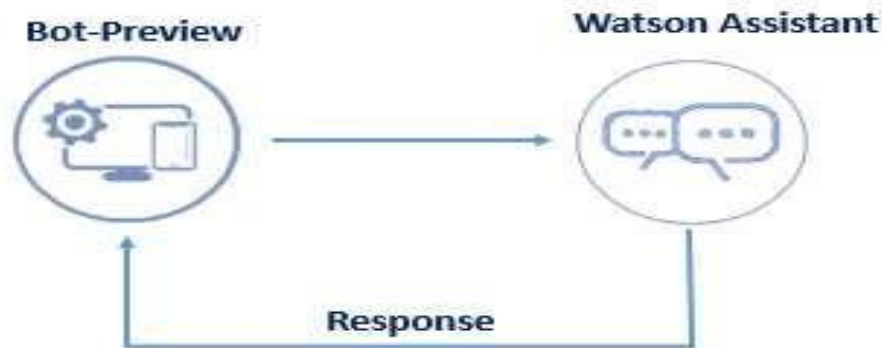
SERVICES USED:

- IBM Watson Assistant

Watson Assistant



BLOCK DIAGRAM:



HARDWARE / SOFTWARE DESIGNING:

To complete this project, you should have the following software and packages.

SOFTWARES:

- Visual studio code
- IBM Watson studio

PACKAGES:

- Flask

FLOWCHART:

To accomplish the above task, you must complete the below activities and tasks:

- Create IBM Services.
- Creating skills & Assistant for Chatbot.
- Creating Savings account action.
- Creating Current account action.
- Creating Loan account action.
- Creating a general query action.
- Creating a Net banking action.
- Create HTML web page.
- Integrate the Watson Chatbot with web page.

ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

- Round-the-clock service.
- Brand Consistency.
- Increased Productivity.
- Reduced Staffing Needs.
- Consistent Response Rate and Availability.
- Helps with Fraud Prevention.
- Chats can be saved.
- Lower costs.

DISADVANTAGES:

- Questions must be programmed beforehand.
- Impersonal
- Must keep information up-to-date.
- Technology issues.

- Needs additional measures to protect identities.

APPLICATIONS:

- Banking chatbots have all the data to predict the spending habits of customers and help them keep their finances on track.

CONCLUSION

The solution to almost all the querying applications has become chatbot for assistance and resolving. We believe that the same technology can be in banking queries as it was meant for that purpose. Though the bot would not be able to solve or satisfy all the queries for customer, it can certainly resolve issues that the user might be facing often and help the banking sectors maintain great relationships with their customers.

FUTURE SCOPE

The future of project lies entirely on how the customers get benefitted from the interaction and the interface. We would have to make improvements in the bot to make it as user-friendly as possible. The following areas could have a serious impact on our scope:

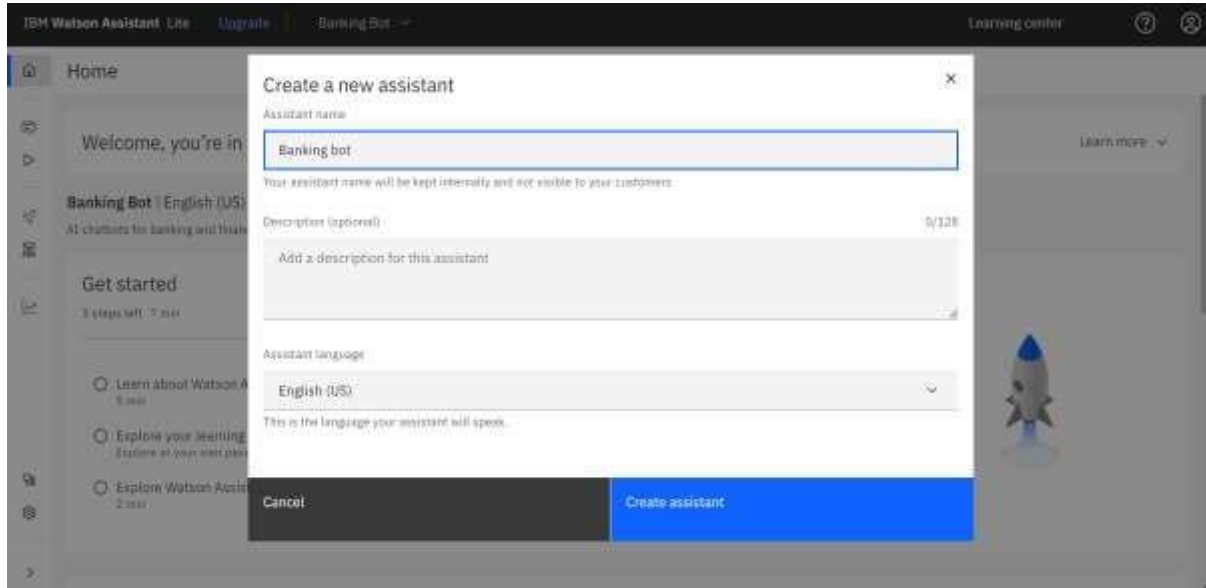
- i. Support for multiple languages
- ii. Low latency in fetching responses
- iii. Voice and video instructions

APPENDIX:

Create IBM Service

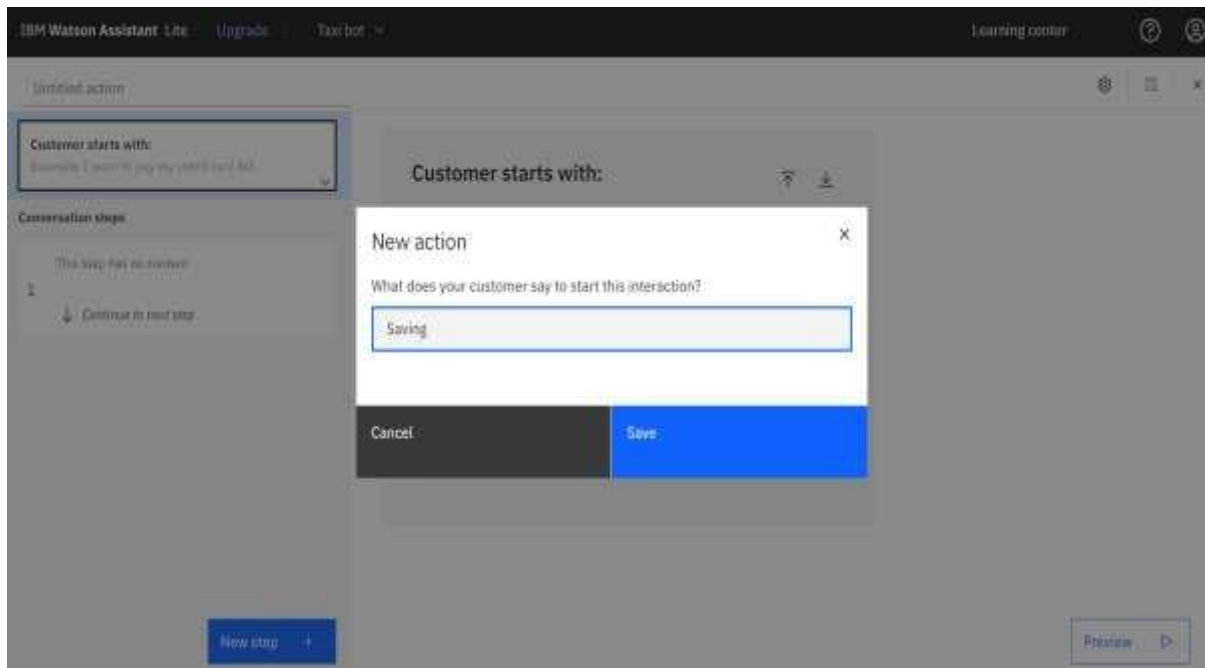
In this activity, you will be creating the Necessary IBM service. The following are the service that you have to create.

- Watson Assistant

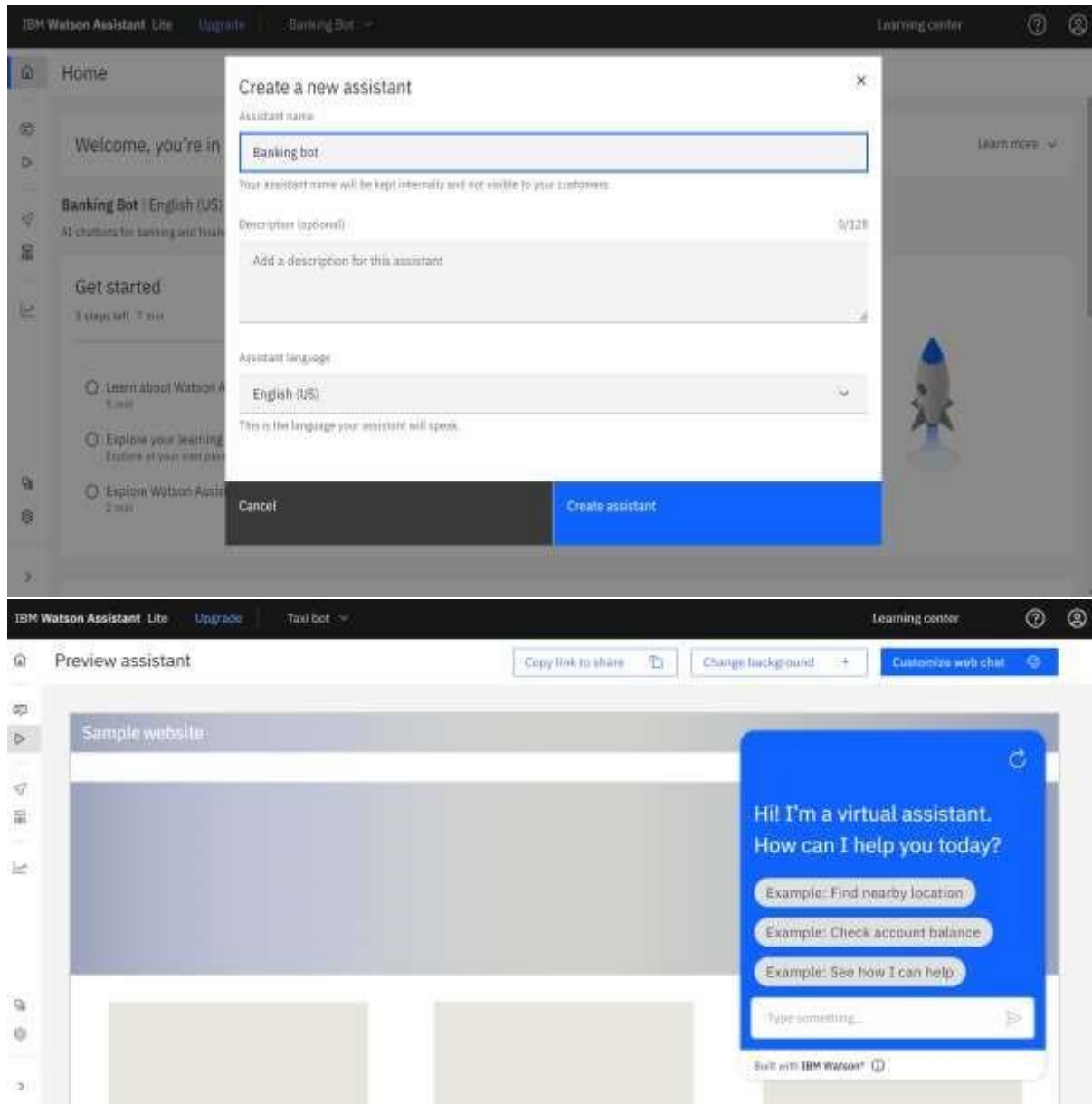


Creating Skills & Assistant For Chatbot

Skills are nothing but actions and steps. Steps are the subset of actions where conversations are built and Assistant is used to integrate skills.

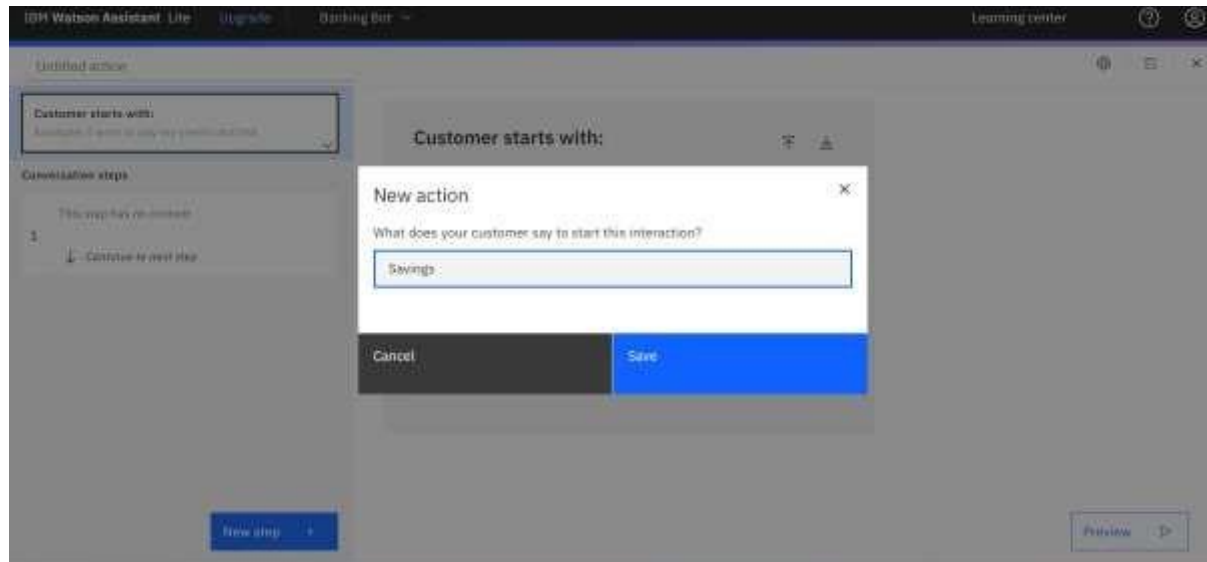


A default template chatbot is created. Need to add actions.

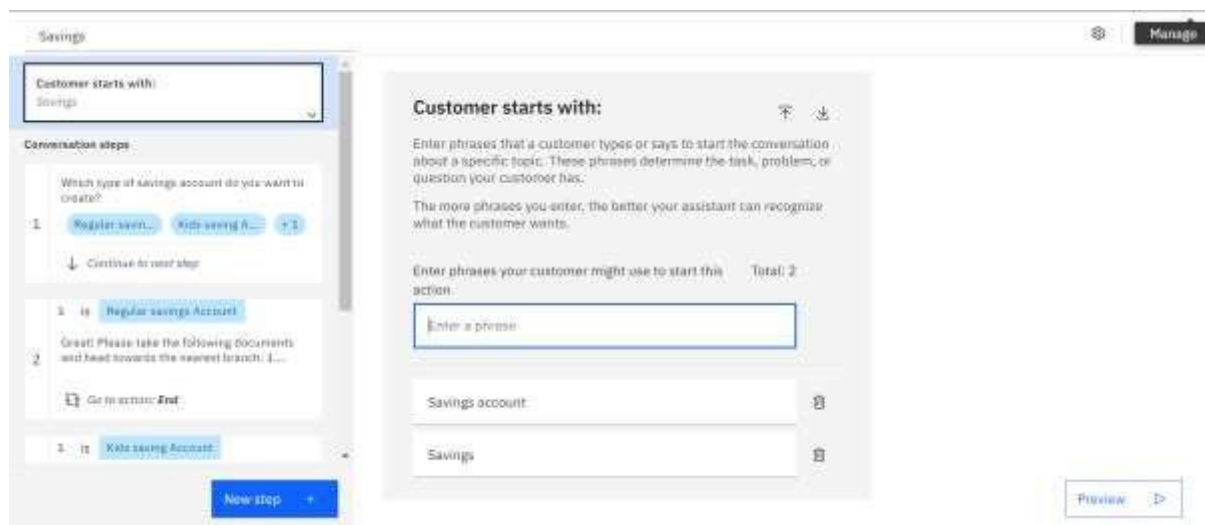


Creating Saving Account Action

Create a saving account in IBM Watson. Create new **Action** Saving.

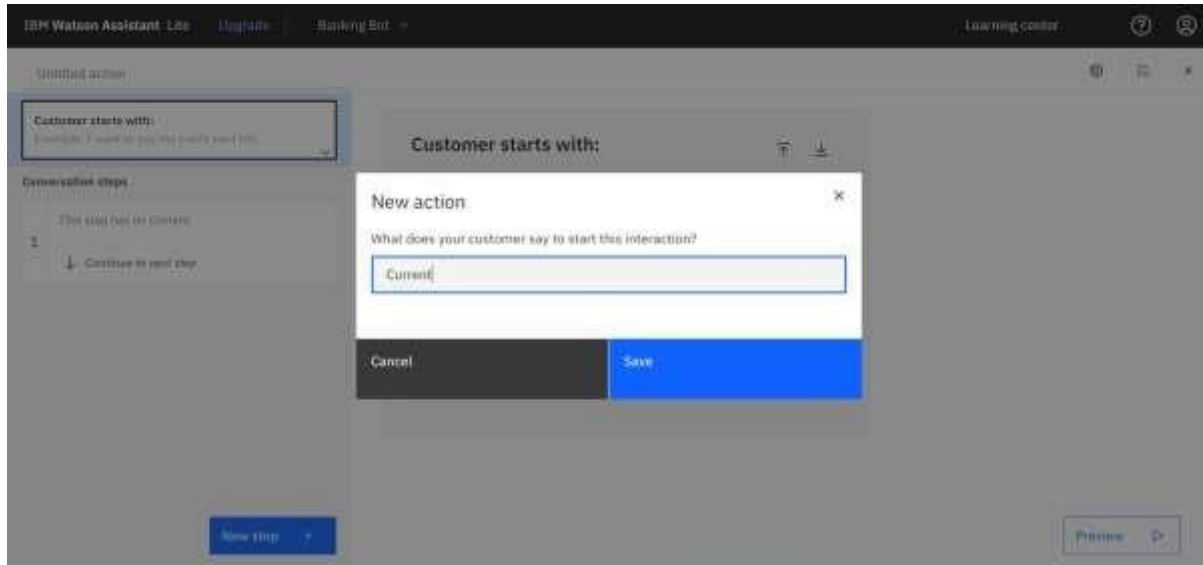


Add steps in savings action.

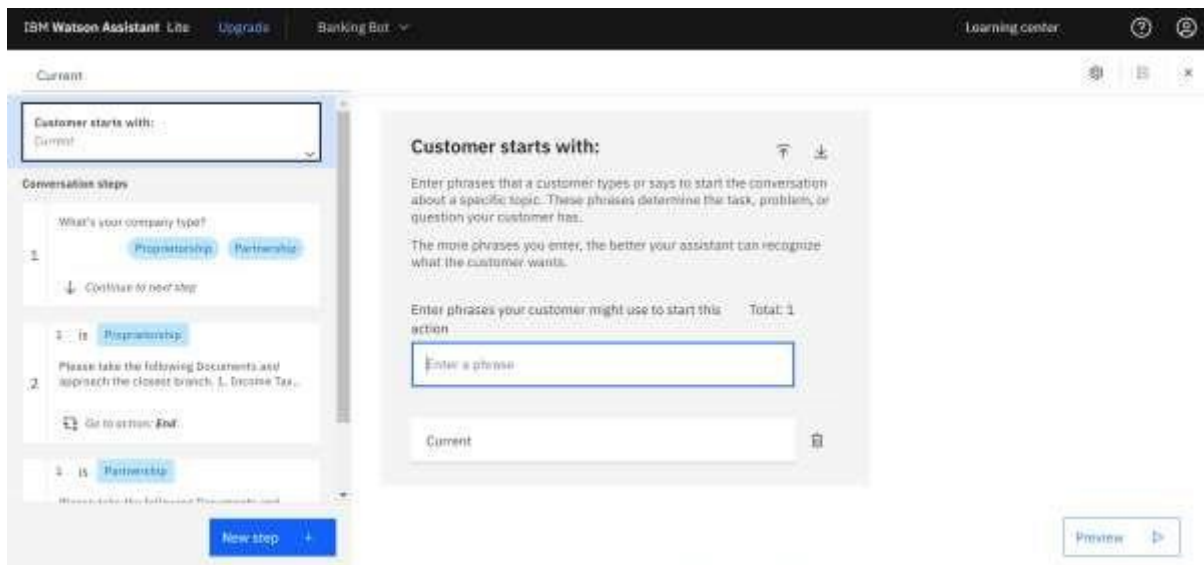


Creating Current Account Action

Create a new Action Current for the current account action.

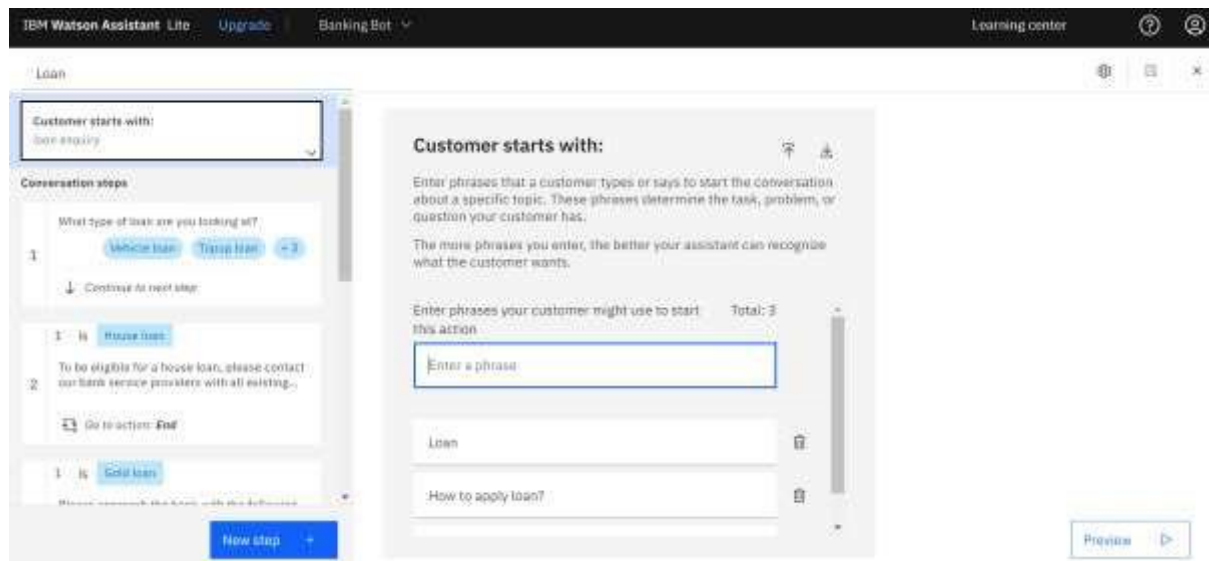


Add steps in current action.



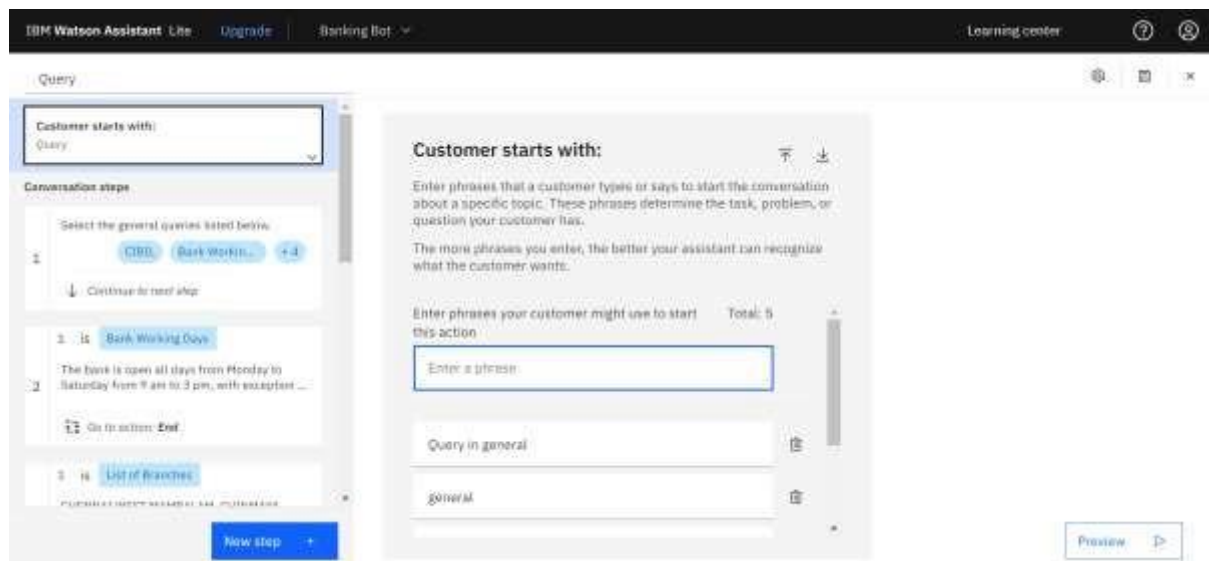
Creating Loan Account Action

Loan action is created with the necessary steps.



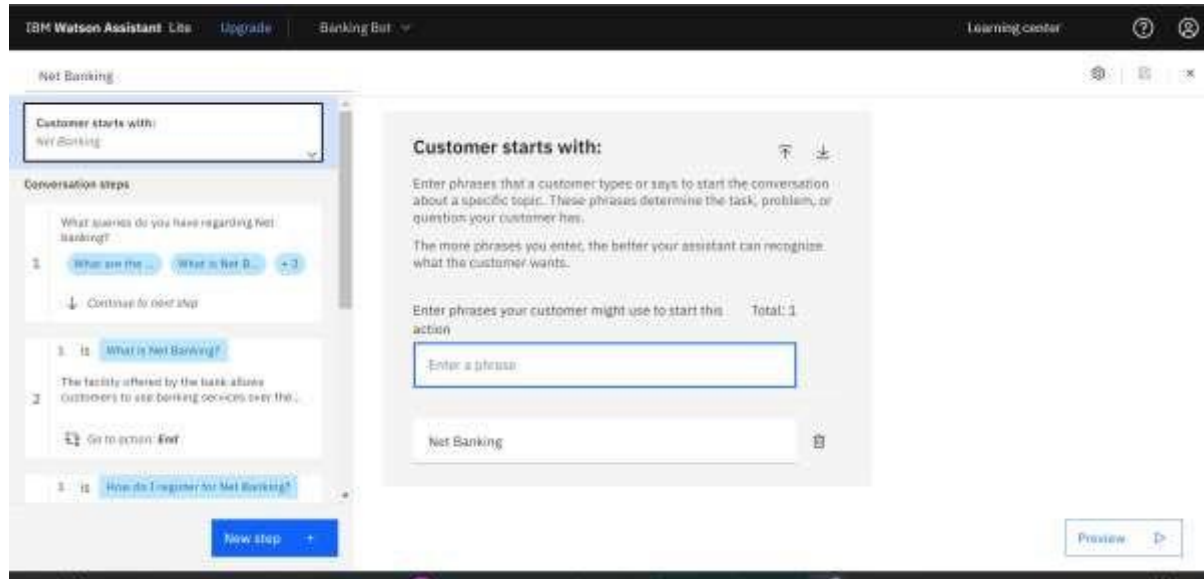
Creating General Query Action

General query action is created with the necessary steps.



Creating Net Banking Action

Net banking action is created with the necessary steps.



In addition to this greeting, end greeting ,index and end actions are also created.

| Name | Last edited | Status | |
|--------------|-------------|--------|---|
| Current | 2 days ago | ✓ | ⋮ |
| Index | 2 days ago | ✓ | ⋮ |
| Register | 3 days ago | ✓ | ⋮ |
| Greeting | 2 days ago | ✓ | ⋮ |
| End Greeting | 2 days ago | ✓ | ⋮ |

| Name | Last edited | Status | |
|-------------|-------------------|--------|---|
| Net Banking | 3 minutes ago | ✓ | ⋮ |
| End | 2 days ago | ✓ | ⋮ |
| Loan | 2 days ago | ✓ | ⋮ |
| Query | a few seconds ago | ✓ | ⋮ |
| Savings | 16 minutes ago | ✓ | ⋮ |
| Current | 2 days ago | ✓ | ⋮ |

Items per page: 50 Showing 1–10 of 10 actions 1 1 of 1 pages

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

- 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.

SOURCE CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>Banking Bot</title>
  <meta content="" name="description">
  <meta content="" name="keywords">

  <!-- Favicons -->
  <link href="assets/img/favicon.png" rel="icon">
  <link href="assets/img/apple-touch-icon.png" rel="apple-touch-icon">

  <!-- Google Fonts -->
  <link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Raleway:300,300i,400,400i,500,500i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,600i,700,700i" rel="stylesheet">

  <!-- Vendor CSS Files -->
  <link href="assets/vendor/aos/aos.css" rel="stylesheet">
  <link href="assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="assets/vendor/bootstrap-icons/bootstrap-icons.css" rel="stylesheet">
  <link href="assets/vendor/boxicons/css/boxicons.min.css" rel="stylesheet">
  <link href="assets/vendor/glightbox/css/glightbox.min.css" rel="stylesheet">
  <link href="assets/vendor/remixicon/remixicon.css" rel="stylesheet">
  <link href="assets/vendor/swiper/swiper-bundle.min.css" rel="stylesheet">
```

```

<!-- Template Main CSS File -->
<link href="assets/css/style.css" rel="stylesheet">

</head>

<body>
<script>
    window.watsonAssistantChatOptions = { integrationID: "fafa4141-555c-427c-9e44-
66a101cbb178", // The ID of this integration.    region: "us-south", // The region your
integration is hosted in.

    serviceInstanceID: "785992fb-b6cf-4d51-b222-23f37f3cee20", // The ID of your service
instance.

    onLoad: function(instance) { instance.render(); }
};
setTimeout(function(){
    const t=document.createElement('script');

    t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/"
(window.watsonAssistantChatOptions.clientVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
    document.head.appendChild(t);

});
</script>

<!-- ===== Header ===== -->
<header id="header" class="fixed-top ">
    <div class="container d-flex align-items-center justify-content-lg-between">

        <h1 class="logo me-auto me-lg-0"><a href="index.html">Bot<span>.</span></a></h1>
    <!-- Uncomment below if you prefer to use an image logo -->

```

```
<nav id="navbar" class="navbar order-last order-lg-0">
  <ul>
    <li><a class="nav-link scrollto active" href="#hero">Home</a></li>
    <li><a class="nav-link scrollto" href="#about">About</a></li>
    <li><a class="nav-link scrollto" href="#services">Services</a></li>
    <li><a class="nav-link scrollto" href="#team">Team</a></li>
    <li><a class="nav-link scrollto" href="#contact">Contact</a></li>
  </ul>
  <i class="bi bi-list mobile-nav-toggle"></i>
</nav><!-- .navbar -->
```

```
<a href="#about" class="get-started-btn scrollto">Get Started</a>
```

```
</div>
```

```
</header><!-- End Header -->
```

```
<!-- ===== Hero Section ===== -->
```

```
<section id="hero" class="d-flex align-items-center justify-content-center">
<div class="container" data-aos="fade-up">
```

```
<div class="row justify-content-center" data-aos="fade-up" data-aos-delay="150">
  <div class="col-xl-6 col-lg-8">
    <h1>AI Based Discourse for Banking Industry</h1>
    <h2>Chatbots for banking and finance operations.</h2>
  </div>
</div>
```

```
<div class="row gy-4 mt-5 justify-content-center" data-aos="zoom-in" data-aos-delay="250">
  <div class="col-xl-2 col-md-4">
    <div class="icon-box">
      <i class="ri-store-line"></i>
      <h3><a href="">Loan</a></h3>
    </div>
  </div>
  <div class="col-xl-2 col-md-4">
    <div class="icon-box">
      <i class="ri-bar-chart-box-line"></i>
      <h3><a href="">Net-Banking</a></h3>
    </div>
  </div>
  <div class="col-xl-2 col-md-4">
    <div class="icon-box">
      <i class="ri-calendar-todo-line"></i>
      <h3><a href="">24*7 </a></h3>
    </div>
  </div>
  <div class="col-xl-2 col-md-4">
    <div class="icon-box">
      <i class="ri-paint-brush-line"></i>
      <h3><a href="">Locker</a></h3>
    </div>
  </div>
  <div class="col-xl-2 col-md-4">
    <div class="icon-box">
      <i class="ri-database-2-line"></i>
      <h3><a href="">Data Security</a></h3>
    </div>
  </div>
</div>
```

</div>

</div>

</div>

</div>

</section><!-- End Hero -->

<div id="preloader"></div>

<i class="bi biarrow-up-short"></i>

<!-- Vendor JS Files -->

<script src="assets/vendor/purecounter/purecounter_vanilla.js"></script>

<script src="assets/vendor/aos/aos.js"></script>

<script src="assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<script src="assets/vendor/glightbox/js/glightbox.min.js"></script>

<script src="assets/vendor/isotope-layout/isotope.pkgd.min.js"></script>

<script src="assets/vendor/swiper/swiper-bundle.min.js"></script>

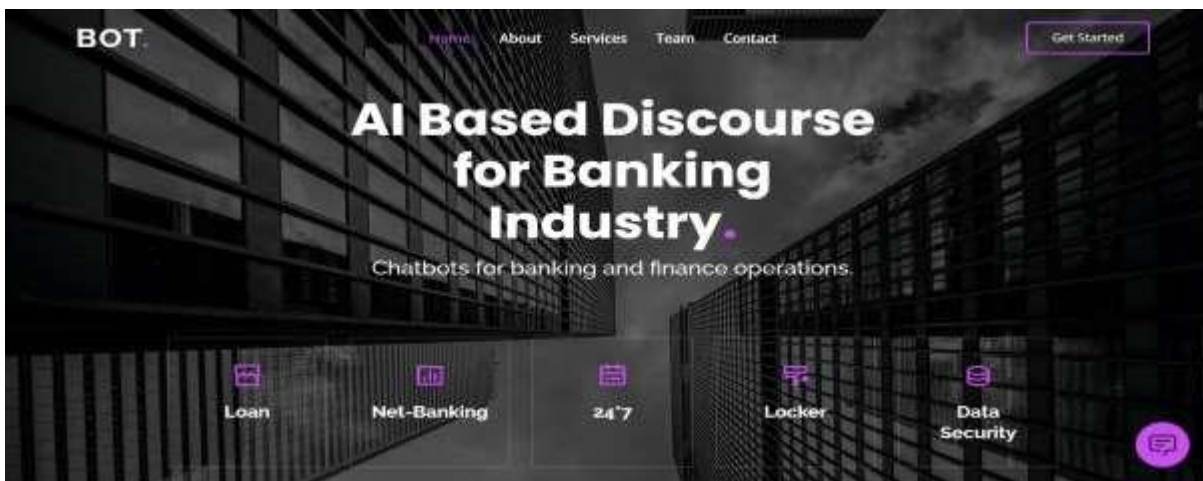
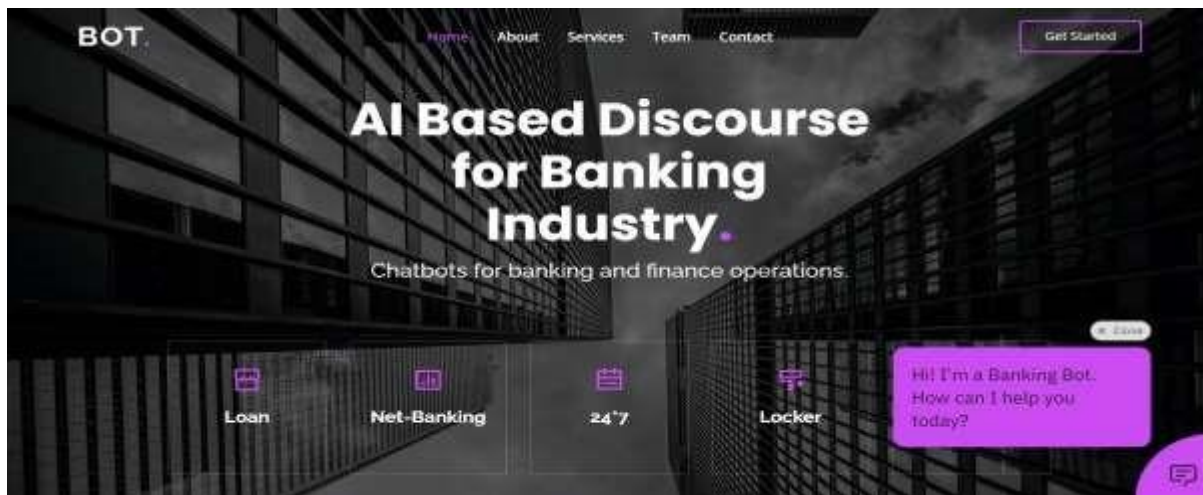
<script src="assets/vendor/php-email-form/validate.js"></script>

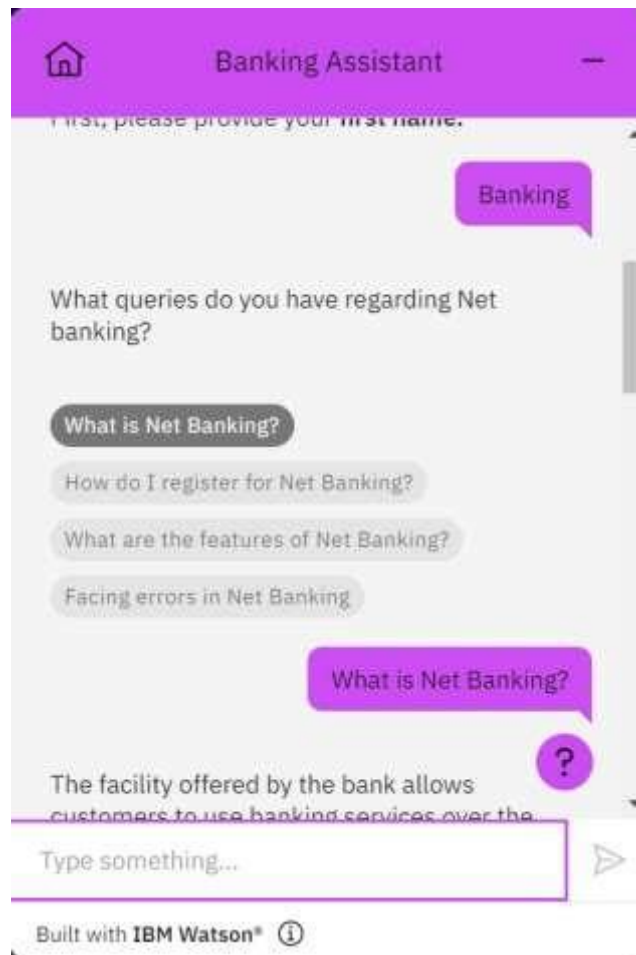
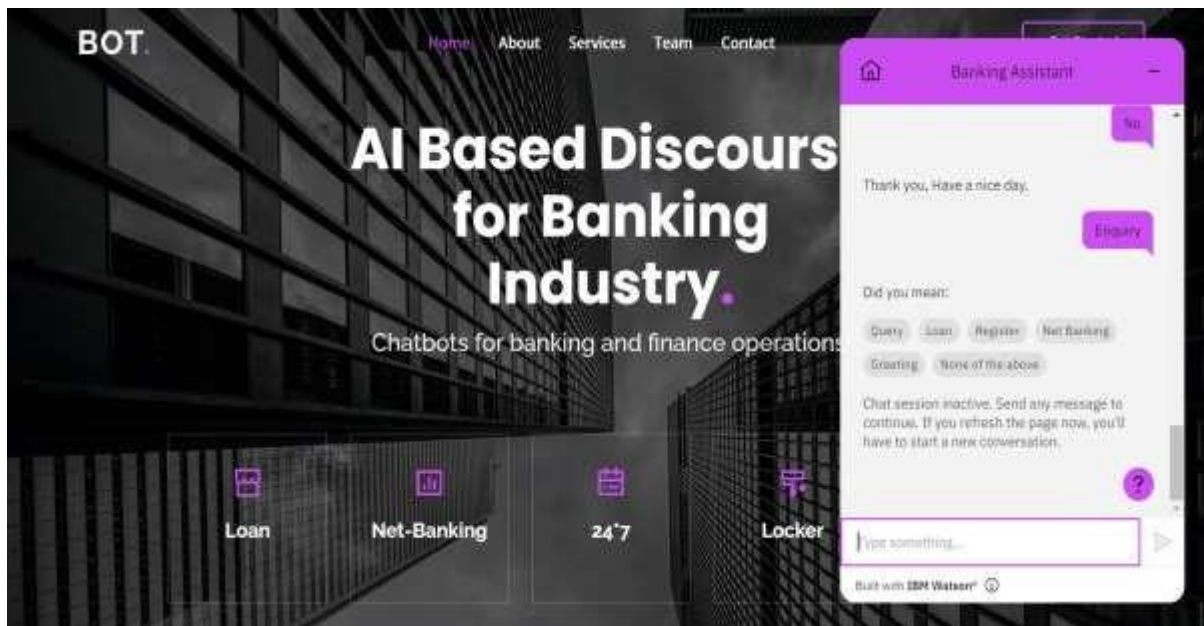
<script src="assets/js/main.js"></script>

</body>

</html>

OUTPUT:





BANKING CHATBOT:

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

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-785992fbb6cf-4d51-b222-23f37f3cee20%3A%3A33c532ec-f7b3-46f0-becbd89ad77b3d68&integrationID=fafa4141-555c-427c-9e44-66a101cbb178®ion=us-south&serviceInstanceID=785992fb-b6cf-4d51-b22223f37f3cee20>

GITHUB LINK : IBM-EPBL/IBM-Project-51468-1660979740