	CSE 299: Junior Design Project Department of Electrical and Computer Engineering			
	Semester:	Spring 2025	**Section:**	**18**
	Weekly Progress:	1	**Date:**	31st Jan, 2025

Faculty Name:	Dr. Shafin Rahman
Group Number and Project Title	Group # 7: Peoject Title:Bank Customer ChatBot
Student Name (ID)	Soleman Hossain 2021682042
Email Address:	soleman.hossain@northsouth.edu

Last week's Target	Follow-up
1. Get to know the basic structure of Chatbot's pipeline. Also, how does RAG work? 2. We targeted the requirements like the packages, vector database, programming language, dependencies, and versions that we will need for our ChatBot 3. We have decided to conduct an online meeting regarding how to go forward with the task that we have selected for the 1st week. We also planned an approach for the workflow of the ChatBot Pipeline 4. We just planned to implement the vector database for our first week using ChromaDB.	Write a status update on the tasks mentioned on the left. This could be empty for the first week. 1. Done, 2. Done, but considering other alternatives for better usage in future. 3. Done, but they can be changed for the betterment of the ChatBot's Performance. 4. Done

Work narrative of the current week

- **Tasks completed:** We have used ChromaDB for our vector database to store the PDF in chunk form and use it for retrieval. PDF loader has been implemented using UnstructuredPDFLoader() class. After loading the PDF, we have set the chunk size = 4000 with overlapping = 1000, initially. We have used the 'nomic-embed-text' text model of 'Ollama' for embedding. Imported packages and modules are: chromadb, langchain_community.document_loaders, langchain_text_splitters import, langchain_community.embeddings and langchain_community.vectorstores
- **Challenges faced:** During the task, we faced the problem of dependencies clashing, and version clashing of packages as all the machines do not acquire the exact requirement.
- **Progress made:** For the first week, we only implemented the code as it can load a PDF, then break it into chunks and store it in the database for future usage.

  [We are expecting future update for better workflow and design in the pipeline]

Target for next weekly update

1. Looking forward to performing retrievals from the created vector database for a given query by the user.
2. Planning to find better approaches.
3. Exploring more chunking strategies and better models for embedding packages and modules.
4. Planning to implement a minimalist and attractive UI with lots of surprises.

	CSE 299: Junior Design Project Department of Electrical and Computer Engineering			
	Semester:	Spring 2025	**Section:**	**18**
	Weekly Progress:	2	**Date:**	**14th Feb, 2025**

Faculty Name:	Dr. Shafin Rahman
Group Number and Project Title	Group # 7: Project Title: Bank Customer ChatBot
Student Name (ID)	Soleman Hossain 2021682042
Email Address:	soleman.hossain@northsouth.edu

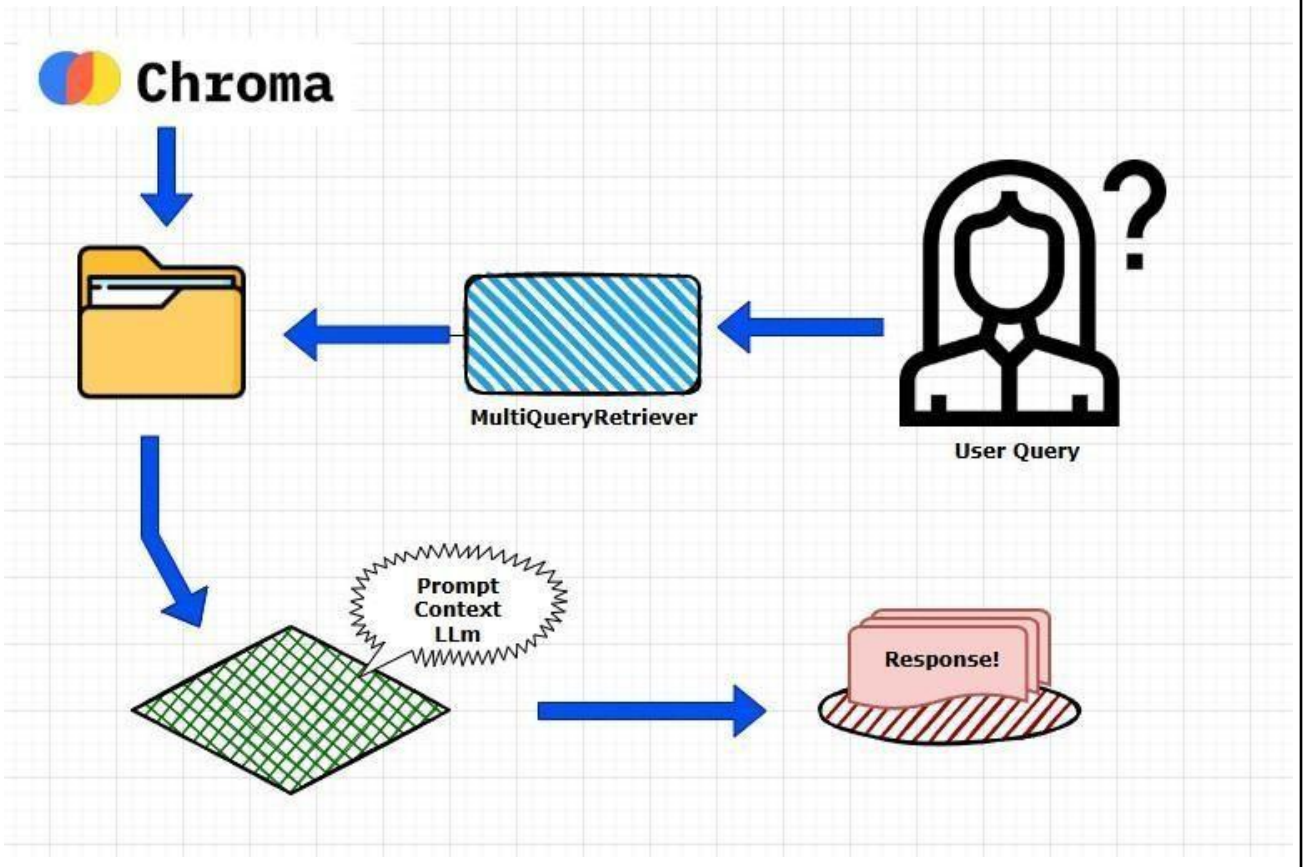
Last week's Target	Follow-up
1. Develop retrievers that can access an existing vector database located in the specified directory and utilize it to generate responses based on user queries. 2. Optimize the workflow for better efficiency and performance. 3. Explore various embedding models, embedding strategies, and vector databases to enhance the chatbot's overall performance and integrate the most effective solutions into its pipeline.	1. Completed. 2. Implementing a multimodal RAG chatbot for improved response quality is in progress. 3. Done. Explored a few models, strategies, and databases. We have also finalized the model and database we are going to use currently for the project. If we implement a better multimodal Chatbot later, we may need to swap a few elements of the pipeline.

Work narrative of the current week

1. Set up the retriever using the LLM models **"llama3.2"** and **"mistral"** for this week.
2. Set the **ChatOllama()** class to initialize the model.
3. Created a query prompt template with **PromptTemplate()** from the langchain-ChatPromptTemplate, PromptTemplate module.
4. Accessed the existing vector database, labeled it for easy reference, and enabled passing it as an argument when needed.
5. Configured the retriever using **MultiQueryRetriever()** from the langchain-MultiQueryRetriever module.
6. Developed a RAG prompt template using **ChatPromptTemplate()** and built the final chain for query processing.
7. Integrated **"langchain"** to seamlessly combine the retriever, prompt, and LLM models into a structured processing chain.

[We are expecting future updates for better workflow and design in the pipeline]

8. This week’s workflow



[Group Diagram]

Target for next weekly update

1. Implement an advanced multimodal RAG chatbot to enhance overall performance.
2. Test the pipeline with different LLMs and compare their performance.
3. Develop a user-friendly interface for the chatbot.
4. Search for a suitable knowledge base to create test sets for chatbot evaluation.
5. Gather the necessary resources to conduct a thorough evaluation.

	CSE 299: Junior Design Project Department of Electrical and Computer Engineering			
	Semester:	Spring 2025	**Section:**	**18**
	Weekly Progress:	3	**Date:**	**28th Feb, 2025**

Faculty Name:	Dr. Shafin Rahman
Group Number and Project Title	Group # 7: Project Title: Bank Customer ChatBot
Student Name (ID)	Soleman Hossain 2021682042
Email Address:	soleman.hossain@northsouth.edu

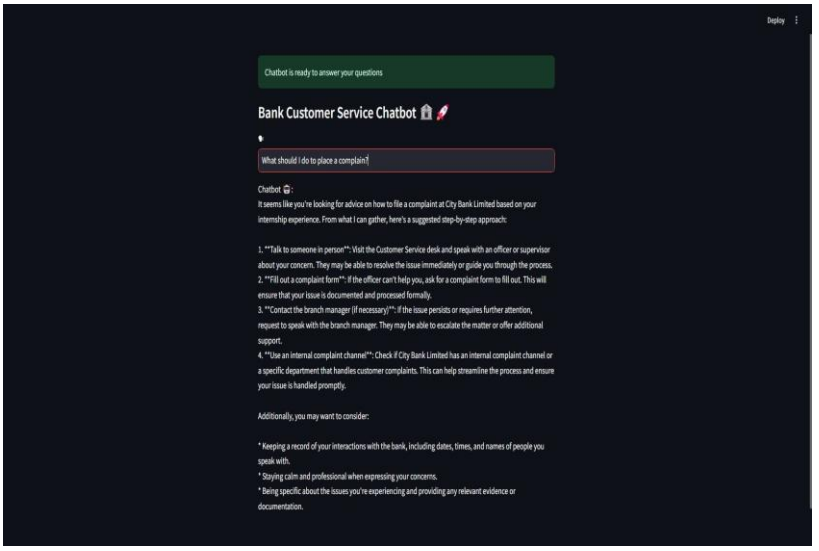
Last week's Target	Follow-up
1. Develop an advanced multimodal RAG chatbot, test it with different LLMs, and compare their performance. 2. Develop a user-friendly interface for the chatbot. 3. Find a suitable knowledge base and gather all required resources to create comprehensive chatbot evaluation test sets.	1. Completed. 2. Created a simple Streamlit UI initially but still looking for a better approach. 3. Considered some approach already. Still looking for a better approach to evaluate the Bot

Work narrative of the current week

1. Retrieve some queries using both **'llama 3.2'** and **'mistral'** and evaluate the model initially.

2. Developed an advanced multimodal RAG chatbot using **Ollama 3.2** and **Mistral**, testing their performance across various tasks. Ollama 3.2 excelled in retrieval efficiency and handling multimodal inputs, while Mistral provided more coherent and contextually rich responses.

3. Started with a simple **Streamlit** UI to get the chatbot up and running, but I'm still exploring better alternatives. The goal is to find a more flexible and efficient solution that improves usability and overall performance.



4. Researched and considered some relevant **knowledge bases** for creating detailed test sets to evaluate the chatbot. Also gathered the necessary resources, including datasets and tools, to conduct practical assessments of its retrieval accuracy and response quality.

[We are expecting future updates for better workflow and design in the pipeline]

Target for next weekly update

1. Describe the primary metrics and performance indicators to be evaluated before starting the chatbot evaluation phase.

2. Conduct manual testing by asking a wide range of users for thorough input on the accuracy, performance, and user experience of the chatbot.

3. Create a plan for automated testing that includes finding and choosing pertinent terms, information, and requirements to expedite the assessment procedure.

4. Gather and arrange all required materials, including testing tools, datasets, and documentation, to enable a comprehensive, methodical, and repeatable assessment of the chatbot's functionality.

	CSE 299: Junior Design Project **Department of Electrical and Computer Engineering**			
	Semester:	Spring 2025	**Section:**	**18**
	Weekly Progress:	4	**Date:**	**14th March, 2025**

Faculty Name:	Dr. Shafin Rahman
Group Number and Project Title	Group # 7: Project Title: Bank Customer ChatBot
Student Name (ID)	Soleman Hossain 2021682
Email Address:	soleman.hossain@northsouth.edu

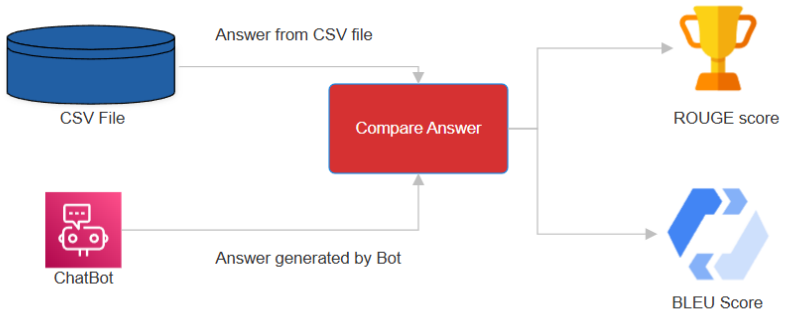
Last week's Target	Follow-up
1. Select primary metrics to test the bot's performance and evaluation. 2. Conduct manual testing with a diverse use of the bot 3. Develop a comprehensive plan for automated testing. 4. Gather and organize necessary testing materials and resources.	1. Almost Done. 2. Half Done. We are still in the process. 3. Still ongoing progress. We have to make a detailed plan. That's what takes time. 4. Done.

Work narrative of the current week

Here is a summary of the student's work on the project during the current week. This narrative should include:

- **Tasks completed**: The main evaluation criteria for the chatbot—accuracy, reaction time, and engagement—are almost complete and are presently undergoing validation based on input from the team. Although more users are required for a more varied viewpoint, manual testing has begun with preliminary user feedback on performance and user experience. Although the automated testing plan is being developed and includes a list of key phrases and tools, more work is needed to finish the setup. Every testing resource that is required, such as tools, datasets, and documentation, has been arranged and prepared for the assessment procedure.
- **Challenges faced:** The process probably encountered a number of difficulties. It might have been difficult to define the appropriate performance indicators that would have matched both technical outcomes and user pleasure. It might have been challenging to assemble a varied user base for manual testing in order to guarantee thorough feedback. It may have taken some time to choose the right frameworks and tools for automated testing in order to strike a balance between accuracy and speed. Furthermore, it might have taken more time and effort to coordinate the relevant resources and guarantee compatibility.

- **Progress made:** Considerable progress has been made thus far. Accuracy, reaction speed, and user satisfaction—the main measures used to assess the chatbot—have almost been finalised and are undergoing final validation. With intentions to broaden testing to a larger user population, manual testing has already begun and useful user feedback regarding performance and experience has been gathered. The strategy for automated testing is well under way; important terms and tools have been selected, and the setup is still being finalised. Furthermore, all necessary testing resources—such as datasets and tools—are arranged and prepared for the assessment stage.



[An initial ongoing **demo** testing phase]

We are thinking of **ROUGE** and **BLEU** as our automated testing to evaluate our ChatBot. But the Plan can be changed if we see a better approach.

[We are expecting future updates for better workflow and design in the pipeline]

Target for next weekly update

Here are next week's targets:

1. **Finalize Primary Metrics:** Get the team to validate the key performance indicators (KPIs) and get them ready for the assessment stage.
2. **Extend Manual Testing:** To get more varied input on chatbot performance and user experience, expand the user pool for manual testing.
3. **Finish Automated Testing Setup:** Complete the automated testing strategy, including the framework and tool selection, and begin configuring the testing infrastructure.
4. **Implement Voice Interaction System:** Start working on the voice interaction system and ensure it integrates seamlessly with the chatbot's current functionalities.

	CSE 299: Junior Design Project **Department of Electrical and Computer Engineering**			
	Semester:	Spring 2025	**Section:**	**18**
	Weekly Progress:	5	**Date:**	**28th March, 2025**

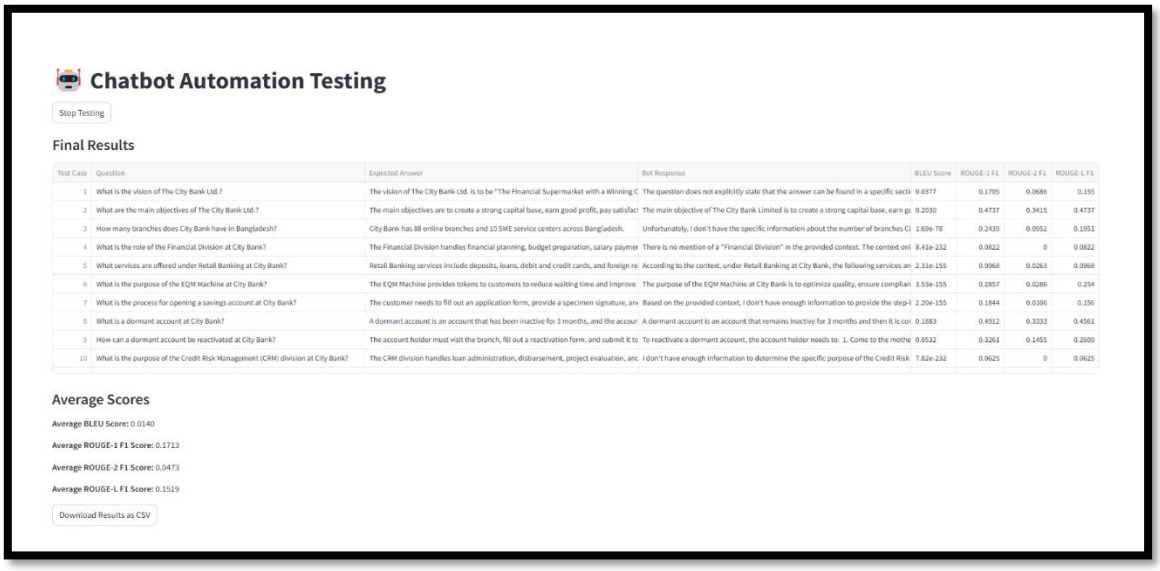
Faculty Name:	Dr. Shafin Rahman
Group Number and Project Title	Group # 5: Project Title: Bank Customer ChatBot
Student Name (ID)	Soleman Hossain 2021682
Email Address:	soleman.hossain@northsouth.edu

Last week's Target	Follow-up
1. Finalize primary matrices. 2. Extend manual testing. 3. Fully automated testing setup 4. Implement a voice interaction system.	1. Done. 2. Ongoing 3. Done 4. Ongoing.

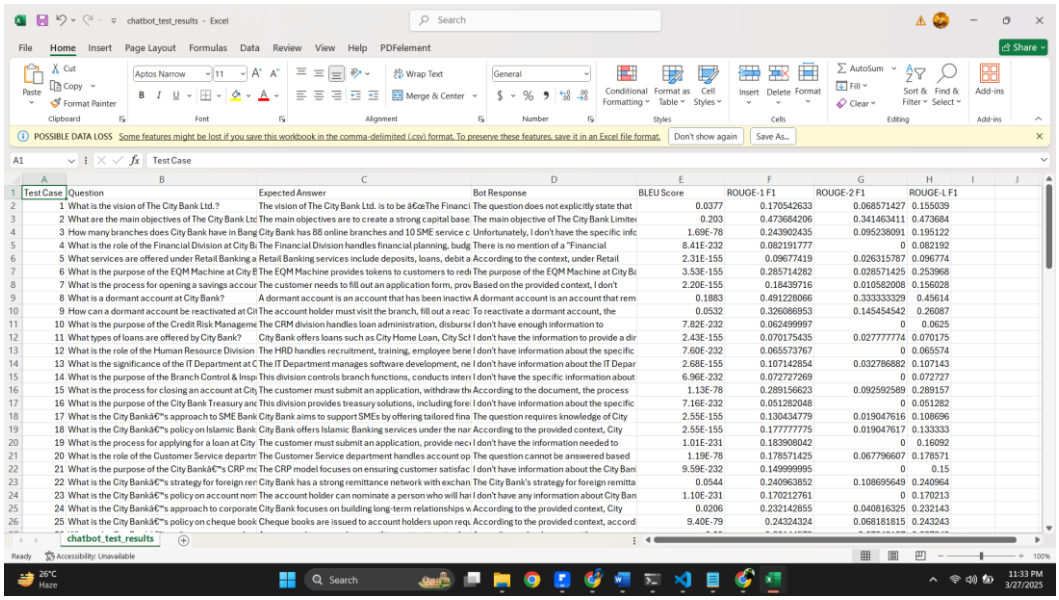
Work narrative of the current week

Here is a summary of the student's work on the project during the current week. This narrative should include:

- **Tasks completed**: This week, our team made significant progress in evaluating and enhancing our RAG-based chatbot through automated and manual testing. We implemented a comprehensive testing framework using BLEU and ROUGE metrics to objectively measure the chatbot's performance against predefined test cases. The Streamlit dashboard we developed systematically processes each question, compares the chatbot's response with the expected answer, and generates detailed scores, including BLEU for n-gram precision and ROUGE-1, ROUGE-2, and ROUGE-L for recall-based assessment. Alongside automated testing, we conducted manual validation to cross-check the results and identify cases where the chatbot's responses, while technically correct, needed contextual refinement. Additionally, we began exploring voice integration capabilities, experimenting with speech-to-text and text-to-speech libraries to enable seamless voice interactions. Our next steps involve further optimizing the testing process and improving the voice system's responsiveness. Here are some screenshot of automation testing and also we were able to add a feature where the user can download the automation testing result as exel file for better view.



Automation Testing



Automation Testing Result in CSV Format

- **Challenges faced:** While implementing the automated testing framework and voice integration system, our team encountered several challenges. One major difficulty was handling extremely low BLEU scores for responses that were semantically correct but phrased differently from the reference answers, requiring us to implement decimal precision formatting for better readability. The ROUGE metrics sometimes produced inconsistent results with very short or very long responses, making interpretation tricky. For the voice integration, we faced latency issues in real-time speech processing and accuracy problems with different accents and background noise in the STT (Speech-to-Text) conversion. Additionally, synchronizing the automated testing pipeline with manual validation proved time-consuming, as we had to ensure the metrics aligned with human judgment while maintaining testing efficiency. These challenges highlighted areas for improvement in both evaluation methods and voice system optimization.

- **Progress made:** This week, our team made significant progress in enhancing our RAG-based chatbot's evaluation framework and expanding its capabilities. We successfully implemented an automated testing system using BLEU and ROUGE metrics through a user-friendly Streamlit dashboard, enabling efficient assessment of the chatbot's responses against expected outputs. The system now generates comprehensive reports with formatted scoring, including handling of edge cases like extremely low values. Alongside automation, we conducted thorough manual testing to validate the metrics' reliability. In parallel, we achieved a key milestone by developing a functional prototype for voice integration, establishing the foundation for speech-based interactions. These advancements have strengthened our testing methodology while moving us closer to creating a multimodal conversational interface.

[We are expecting future updates for better workflow and design in the pipeline]

Target for next weekly update

Here are next week's targets:

1. **Extend Manual Testing:** To get more varied input on chatbot performance and user experience, expand the user pool for manual testing.
2. **Implement Voice Interaction System:** Start working on the voice interaction system and ensure it integrates seamlessly with the chatbot's current functionalities.

[The Team will add further improvement in the pipeline if possible and the result will be shown in the final report]