Final Delivery Guidelines

Deadline: 03/01/25

This final delivery involves submitting your project's code through a GitHub repository. Groups must ensure that the codebase is of the highest quality, easily maintainable, and well-documented. Adhere to best coding practices by ensuring the code is modular and well-organized. Additionally, provide clear and comprehensive documentation, including a well-structured README, to facilitate understanding and future development of the system. The code will be evaluated based on the following aspects:

Code Quality

- Ensure the code is readable with proper use of comments, meaningful variable names, and consistent formatting (e.g., following Python's PEP 8).
- Maintain modularity by organizing the code into reusable and testable components.

Code Organization

- Structure the GitHub repository to follow the provided template, ensuring that all required files are present, correctly named, and well-organized. Additional files or folders can be added as needed.

Documentation

- Provide a comprehensive README.md file that follows the provided template. You may include additional sections if necessary, but ensure the README is clear and provides all relevant information for understanding and using the project.

Groups must submit a **link to their private GitHub repository via Moodle**. The repository should be named after your company (e.g., "CompanyName"), with "CompanyName" replaced by your actual company name. The repository must be set to private, and groups should invite me (**tiagomosantos**) as a collaborator to allow me to access the code.

Chatbot Functionalities

Grading Breakdown

8 points (40% of the final grade)

Objective

Groups are tasked with developing a chatbot using LangChain and the OpenAI API. The system should be built with solid object-oriented programming principles, follow Python best practices, and deliver an intelligent, responsive user experience. The chatbot should be capable of handling a variety of tasks, from answering company-related queries to performing complex database operations and data retrieval.

Requirements

Intention Router

- **Test:** Verify that the intention router functions correctly.
- **Evaluation:** The router should accurately direct user requests to the appropriate functionality, ensuring efficient handling of complex intents. Does it handle fallback cases where the user input is non-related or small talk?

Intention Router Accuracy

- **Test:** Generate at least 50 messages per intention as specified in the Readme Template. Additionally, generate at least 25 small-talk messages related to your company and 25 off-topic messages unrelated to the company, labeled as "None."
- **Evaluation:** Report the accuracy for each intention, as well as the overall accuracy. Accuracy should be calculated as the percentage of correct responses out of the total inputs for each intention.

Prompt Injection Tolerance

- **Test: Assess:** Input validation by providing potentially malicious prompts (e.g., prompt injections).
- **Evaluation:** The chatbot should handle and prevent prompt injections through input filters.

Memory Implementation

- **Test:** Verify that the chatbot retains context across multiple interactions.
- **Evaluation:** The chatbot should maintain conversation continuity, offering relevant responses based on memory and improving over time.

Intentions

- **Test:** The groups should provide at least 3 examples of test messages for each chatbot's functionality/intention and the expected behavior. You should have at least 8 user intentions.
- **Evaluation:** Verify if each functionality works as the group described. The quality of your prompts and the correct use of chains.

Company Information Response Capability

- **Test:** Verify if the chatbot accurately answers company-related questions based on preloaded information.
- **Evaluation:** The chatbot should correctly respond to a variety of queries, such as details about the company, services, or policies.

Database Modification Capabilities

- **Test:** Evaluate CRUD (Create, Read, Update, Delete) operations on the database using your chatbot.
- **Evaluation:** Ensure that all at CRUD operations function properly with appropriate validation.

Retrieval-Augmented Generation (RAG)

- **Test:** Test the chatbot's ability to use Pinecone embeddings of your PDF files for contextual responses. Groups should provide at least 5 user messages that will leverage RAG.
- **Evaluation:** At least one user intention should leverage RAG to enhance response accuracy and relevance.

User Interface and Experience

Grading Breakdown

4 points (20% of the final grade)

Objective

Groups should design and implement an engaging, functional user interface using Streamlit. The interface should facilitate seamless interactions between the user, the chatbot, and the underlying database. Focus on creating a smooth and intuitive user experience that reflects the brand's identity while incorporating robust features such as secure login and efficient navigation.

Requirements

Home Page

- **Test:** Ensure the home page provides a clear introduction to the startup with easy navigation options to other pages on the website.
- **Evaluation:** The home page should be engaging, with a concise introduction to the company and a simple, intuitive layout with clear navigation links.

About Page

- **Test:** Verify that the About page displays the company's mission, vision, and values.
- **Evaluation:** The page should present this information clearly and in a visually appealing way, contributing to the overall branding of the startup.

Chatbot Interaction Page

- **Test:** Ensure there is an interface for users to interact with the chatbot, including a visible chat history.
- **Evaluation:** The page should provide a seamless chat experience, displaying a clear history of messages. It should also have authentication for secure login before users can interact with the chatbot.

Login/Register Functionalities

- **Test:** Ensure users can securely log in and register using the website, this interacts with your database.
- **Evaluation:** The login page should be intuitive, with proper validation using your database. The register should create a new user profile in the database and after that I should be able to login.

User Experience (UX)

- **Test:** Evaluate the overall UI/UX of the site.
- **Evaluation:**
    - The interface should be intuitive, making it easy for users to navigate and interact with key features.
    - Components such as buttons, menus, and forms should be functional and responsive, ensuring a smooth user experience across devices.

Other Relevant Pages (Optional)

- **Test:** Verify the existence of any additional relevant pages.
- **Evaluation:** These pages should be consistent in design and contribute to the overall functionality and user experience of the website.