Proposal: Chatbot System

Team Members:

- Omnia Ahmed
- Mariem Mohsen
- Sara Mohamed

Overview:

The chatbot project focuses on building a conversational AI that interacts with users in real time using **NLP models**. The system integrates a **FastAPI backend** to ensure responsiveness and is optimized for handling diverse queries across multiple domains.

Project Objectives:

1. **Data Collection and Preprocessing:**
    a. Collect conversation datasets or scrape FAQs.
    b. Implement NLP pipelines to prepare the data for modeling.
2. **Chatbot Models:**
    a. Build intent recognition and response generation models using **Transformer architectures** and **BERT embeddings**.
3. **Model Evaluation and Optimization:**
    a. Evaluate performance with metrics like **Accuracy**, **BLEU**, and **ROUGE**.
    b. Tune models based on feedback and improve response generation.
4. **API Deployment and Real-Time Interaction:**
    a. Deploy the chatbot using **FastAPI**, enabling real-time conversation flow.
    b. Handle input/output efficiently for smooth interactions.

Project Phases:

- **Week 1: Data Collection and Preprocessing**
  Deliverable: Cleaned dataset and NLP preprocessing pipeline.
- **Week 2: Model Development**
  Deliverable: Trained intent recognition and response generation models.
- **Week 3: Model Evaluation and Optimization**
  Deliverable: Performance metrics and optimized models.

- **Week 4: API Deployment and Documentation**
  Deliverable: FastAPI deployment and final documentation.

Tools and Technologies:

- **Programming Language:** Python
- **Libraries:** TensorFlow, PyTorch, spaCy, NLTK
- **NLP Models:** Transformer-based architectures (BERT, GPT)
- **Deployment:** FastAPI

Current Status:

- **Data Collection and Preprocessing:** Completed initial steps.
- **Modeling:** Developing and fine-tuning the intent recognition and response generation models.
- **API Deployment:** Preparing the FastAPI backend for testing.

Expected Outcomes:

- A chatbot capable of engaging users across various domains.
- Optimized accuracy and performance using advanced NLP metrics.
- Seamless real-time interaction using a FastAPI backend.

Conclusion:

This chatbot project integrates NLP techniques and deep learning models to create an interactive system for real-time conversations. Deployed through **FastAPI**, the chatbot ensures high responsiveness and provides relevant answers based on user inputs.