AI Assignment

Task 1: Get Matching Person Names

Objective:

Build a name-matching system that finds the most similar names from a dataset when a user inputs a name.

Key Steps:

- **Data Preparation:** Store similar names (like Geetha, Gita, Gitu, etc.) in a list, take at least 30 names.
- **Similarity Matching:** When a user enters a name, find the most similar name(s) using any library, Vector DB Search or anything, that is your choice.

Expected Output:

- **Best Match:** The closest matching name with a similarity score.
- **List of Matches:** A ranked list of other similar names with scores.

Task 2: Local LLM Integration & Chatbot

Objective:

Set up an AI model on a local server, fine-tune it, and build a chatbot interface.

Key Deliverables:

- **Server Setup:**
  - Install an open-source model or smaller models if resources are limited.
- **Fine-Tuning:**
  - Collect/prepare datasets specific to the chatbot's use case i.e. Train with Recipes data using custom datasets.
- **API Integration:**
  - Expose the model through an API that accepts queries and returns JSON responses.
- **Chatbot Development:**
  - Build a chatbot UI (CLI, Web, or Mobile) that sends queries to the API and displays the response conversationally.
  - Connect with Python API frameworks (FastAPI)
- **Expected Output:**
  - When user enters ingredients, it should suggest us a recipe based on it. For example, user enter Egg, Onions. It should answer with a recipe for it

Deliverables of above two tasks:

- Provide **complete project code** with all required files and dependencies.
- The code should be **fully runnable on a standard Windows/Linux laptop** without any external setup, except for the clearly documented steps.
- Include **README documentation** with:
  - Setup instructions
  - Installation commands (if any)
  - How to run the project
  - Sample input and expected output for verification
- Ensure the project can be **tested and verified locally** by our team.