# **Hiring AI Assistant Chatbot**

## **Project Overview**

The Hiring Assistant Chatbot is an AI-powered tool designed to streamline the hiring process by generating technical interview questions based on a candidate's skills and job role. It also provides a chatbot interface to clarify doubts. The application is built using Streamlit and integrates with NVIDIA AI models to generate intelligent responses.

## **Installation Instructions**

### **Prerequisites**

* Python 3.x installed
* NVIDIA API key set as an environment variable (NVIDIA\_API\_KEY)
* Required dependencies installed

### **Setup Steps**

1. **Clone the repository:**

git clone <repository-url>

cd <repository-folder>

2. **Create a virtual environment (optional but recommended):**

python -m venv venv

source venv/bin/activate # On Windows use: venv\Scripts\activate

3. **Install dependencies:**

pip install -r requirements.txt

4. **Run the application:**

streamlit run ai\_assistant.py

## **Usage Guide**

1. **Sign Up / Login**
   1. Users can create an account or log in using their credentials.
2. **Create a New Chat**
   1. Enter a chat name and generate technical questions based on the candidate's information.
3. **Switch Between Chats**
   1. Previously generated questions will be retained when switching between chats.
4. **Generate Technical Questions**
   1. Provide the candidate's tech stack and role to generate relevant interview questions.
   2. If incorrect or missing details are provided, a warning message will prompt the user to enter proper details.
5. **Doubt Clarification**
   1. Users can ask doubts, and the AI model will generate relevant responses.

## **Technical Details**

* **Framework:** Streamlit
* **Backend:** Python
* **Database:** JSON-based storage for user authentication and chat history
* **AI Model:** NVIDIA AI API using meta/llama-3.1-405b-instruct
* **API Integration:** OpenAI Python SDK with NVIDIA AI API

## **Prompt Design**

The AI model is prompted dynamically based on the user's inputs.

* If both role and tech stack are invalid, the chatbot requests the user to provide valid information.
* If only one of the two is valid, the chatbot adapts and generates questions accordingly.
* The AI model is configured with a low temperature (0.2) and a moderate top\_p (0.7) to ensure precise and relevant responses.

## **Challenges & Solutions**

1. **Handling Incorrect Inputs**
   1. Implemented a check to warn users if they provide invalid or missing role/tech stack details.
2. **Retaining Chat History**
   1. Ensured that previously generated questions remain visible when switching between chat sessions.
3. **Double Login Button Click Issue**
   1. Refactored session state handling to ensure login works with a single click.
4. **API Key Issues**
   1. Added a check to ensure an API key is set before making requests to prevent runtime errors.