INTERVUE ASSIGNMENT PROJECT

Overview

This project is an interview chatbot that utilizes Google's Gemini 2.0 Flash model for generating responses. It is built using FastAPI for the backend and React for the frontend, providing a seamless user experience for conducting interviews.

Source Code:

- Github -> https://github.com/namra4122/Intervue_Assignment

Deployed Project Link:

- Vercel Link -> https://intervue-assignment-beta.vercel.app/
- Backend Docs -> https://namra4122-intervueassignmentbackend.hf.space/docs
- Vide Demo of the Assignment -> https://www.youtube.com/watch?v=NfPMjMX4rLY

Project Structure

```
Intervue_Assignment/
├── cli/
│ ├── config/
│ │ └── interview_flow.json
│ ├── main.py
│ ├── requirements.txt
│ └── src/
│ ├── services/
│ │ ├── flow_service.py
│ │ └── llm_service.py
│ └── utils/
├── web/
│ ├── backend/
│ │ ├── config/
│ │ │ └── interview_flow.json
│ │ ├── main.py
│ │ ├── requirements.txt
│ │ └── services/
│ │ ├── flow_service.py
│ │ └── llm_service.py
│ └── frontend/
```

```
│ ├── src/
│ │ ├── components/
│ │ │ ├── ChatInterface.tsx
│ │ │ ├── MessageBuble.tsx
│ │ │ └── WelcomePage.tsx
│ │ ├── App.tsx
│ │ └── index.tsx
│ └── package.json
└── README.md
```

Versions

CLI Version

The CLI version allows users to interact with the chatbot directly in the terminal.

Requirements

- `langchain`
- `langgraph`
- `google-genai`
- `python-dotenv`

Running the CLI Version

1. Navigate to the `cli` directory.
2. Install the required packages:

```
pip install -r requirements.txt
```

3. Run the application:

```
python main.py
```

Web Version

The web version provides a user-friendly interface for interacting with the chatbot.

Frontend Requirements

- `react`
- `react-router-dom`,
- `tailwindcss`

Frontend Implementation

The frontend is built using React and provides a user-friendly interface for interacting with the chatbot.

Key Components

- **WelcomePage.tsx**: Allows users to enter their name and initialize the chat session.
- **ChatInterface.tsx**: Manages the chat interface, displaying messages and handling user input.
- **MessageBuble.tsx**: Renders individual chat messages, distinguishing between user and bot messages.

Running the Frontend

1. Navigate to the `web/frontend` directory.
2. Install the frontend dependencies:

```
npm install
```

3. Start the frontend application:

```
npm run dev
```

Backend Requirements

- `langchain`
- `langgraph`
- `google-genai`
- `python-dotenv`
- `fastapi[standard]`
- `uvicorn`

Running the Backend Web

1. Navigate to the `web/backend` directory.
2. Install the required packages:

```
pip install -r requirements.txt
```

3. Start the FastAPI server:

```
uvicorn main:app --reload
```

4. Navigate to the `web/frontend` directory.
5. Install the frontend dependencies:

```
npm install
```

6. Start the frontend application:

```
npm run dev
```

API Endpoints

The backend provides the following API endpoints:

- **Initialize Chatx**
- **Endpoint:** `POST /api/init`
- **Request Body:**

```
{
    "username": "string",
    "session_id": "string"
}
```

- **Response:**

```
{
    "response": "string",
    "session_id": "string",
    "current_node": "string"
}
```

- **Chat Response**
- **Endpoint:** `POST /api/chat`
- **Request Body:**

```
{
    "session_id": "string",
    "message": "string"
}
```

- **Response:**

```
{
    "response": "string",
    "session_id": "string",
    "current_node": "string"
}
```

- **Reset Chat**
- **Endpoint:** `POST /api/reset`
- **Request Body:**

```
{
    "session_id": "string"
}
```

- **Response:**

```
{
    "response": "string",
    "session_id": "string",
    "current_node": "string"
}
```

- **Get Chat History**
- **Endpoint:** `GET /api/sessions/{session_id}/history`
- **Response:**

```
{
    "history": [{
        "role": "string",
        "content": "string"
    }],
    "current_node": "string"
}
```

Conclusion

This project serves as an interactive interview chatbot, leveraging advanced language models to facilitate engaging conversations. The CLI and web versions provide flexibility in how users can interact with the chatbot.