

AI QGen Project

This project is a full-stack application that includes a Python backend and a React frontend. The backend uses FastAPI to serve various endpoints, and the frontend is built with React to provide a user interface for interacting with the backend.

About Backend

The backend is in the **python-backend** directory and uses FastAPI to serve various endpoints. The main entry point is `server.py`, which includes routers for different functionalities:

- 1) **askquestion.py**: Endpoint for generating interview questions.
- 2) **descriptive.py**: Endpoint for generating and submitting
- 3) **mcq.py**: Endpoint for generating and submitting multiple-choice questions.
- 4) **rag.py**: Endpoint for uploading and querying documents.
- 5) **chat.py**: Endpoint for chat functionality descriptive questions.

Backend Setup

1. Navigate to the `python-backend` directory:

```
cd python-backend  
Code .
```

2. Create a virtual environment:

```
python -m venv venv
```

3. Activate the virtual environment from Terminal of VS Code (Ctrl + ~):

```
On Windows: venv\Scripts\activate  
On macOS/Linux: source venv/bin/activate
```

4. Install the required dependencies:

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

5. Run the FastAPI server:

```
python server.py
```

6. **Add .env to project**

```
OPENAI_API_KEY="XXXX"  
TIME_GENERATED=2025-01-03 10:42:12.990444  
TOKEN_URL=https://sso-dev.johndeere.com/oauth2/ausx8znnrQpD7x0f0h7/v1/token  
CLIENT_ID=00a2acfvtgzc9aDL0h8  
CLIENT_SECRET=SAXRW-olVviVNw6G6ThBklXkl48Rpli5050hWxNfxUtyDD5xtjCY3g56CQ3uUm  
SCOPE=mlops.deere.com/model-deployments.llm.region-restricted-inocations  
DEERE_AI_GATEWAY=https://ai-gateway.deere.com/openai
```

Setup Frontend:

1. Navigate to the react-app directory:

```
cd react-frontend
```

2. Install the required dependencies:

```
npm install
```

3. Start the React development server:

```
npm start
```