

Update 3 – Progress Update Report

CSE 499A: Senior Design I | Section: 15 | Group: 03

Md. Kamrul Islam (2211745642) | Farhan Ishraque (2212002042) | Atikul Islam Nahid (2211978042) | Iram Shehzad (2131614642)

Week Update Summary

The team demonstrated strong advancements in several AI functionalities this week. We developed a working chat agent using LangGraph and OpenAI, which receives user questions and logs the chat history. We have completed the backend for a resume checker, which we developed using Python. We began work on the backend for the AI Mock Interview feature using Node.js, Express, and researched relevant papers to form a plan around API integration. We added to the AI Mock Interview backend, first establishing routes and environment variables and testing calls to the API. This component of the work offered great opportunities for applying the knowledge we had obtained in RESTful API and AI integration to our project, despite the limitations posed by technical issues.

Completed Task

- Completed to learn LangGraph.
- Studied some research papers and journals to get the idea of this technology.

Contribution

Md. Kamrul Islam:

I developed a straightforward chat agent with the help of LangGraph and OpenAI's tools. The agent accepts user questions, processes the input question using OpenAI's language model, and preserves messages to enable chat history. I set up the packages "langgraph" and "langchain_openai", established a ChatState for storing messages exchanged between the user and agent, and set up a **chat_node** to send user prompts to OpenAI's LLM and return the agent's response. Lastly, I built a StateGraph that connected everything together (START → chat_node → END) to create a functioning chatbot. I tested it with the user prompt "What is the capital of Bangladesh?" and it functioned as expected.

Farhan Ishraque:

In this update, I have completed the backend of the resume checker with Python. For the database used Firebase and MongoDB were used. In this process, I faced several challenges, including the backend not connecting initially, so I had to try alternative methods and ultimately succeed. I have been learning about Lang chain and Lang graph, as well as various models for our AI agent, and my teammates are also learning and testing these models this week. I created a backend that calculates the percentage of matching jobs and your skills. And the user has to

give the description and the position, then it will be given by AI. My mates and I wrote the weekly report together so that we wouldn't miss any content we had done that week.

Atikul Islam Nahid:

I worked on building the backend for an AI Mock Interview feature. At first, I focus on some journals that are related to AI mock interviews and which models they have used. Tried to use JavaScript, Node.js, and Express server with routes to fetch interview questions and tried to evaluate candidate responses using this, but faced some difficulties getting integrated with the frontend, as the API was not connected properly

Iram Shehzad:

I collaborated on the proposal and began writing the backend of an AI mock interview tool using Node and Express on our JobCore platform. The idea was to send requests to fetch the interview questions and then evaluate the answers with the OpenAI API. I started by setting up the server, routes, and environment variables, and then tested everything in Postman. However, I ran into issues connecting to the API, encountering various errors, such as 'The OPENAI_API_KEY environment variable is missing or empty.' Although the project was not fully functional, I learned a great deal about backend development, RESTful APIs, and integrating AI services, which I will continue to improve upon moving forward.

Work in Progress

- Resume Checker backend and frontend integration.
- Mock Interview questions, answers, and scoring based on the answers.
- Job Search from many job posting platforms.

Next Target

- Complete the chatbot with the concept of memory in LLM and context.
- Job Search from many job posting platforms.
- Resume Checker backend and frontend integration.
- Complete mock interview questions, answers, and a scoring system.

Challenges

- Unavailability of the LLM model's API.
- Unavailability of enough research papers in our domain.
- Costly existing tools or websites in the current market.