SE 492
Team May24-49
Generative AI to Assess Learning

Present:
Akpobari Godpower
Alex Vongphandy
Abram Demo
Henry Duwe (Client/Advisor)
Absent:
Drake Rippey, he was feeling sick and let us know beforehand that he wouldn't attend

Agenda: We had already had one meeting before this but it was a short meeting, mostly reviewing our work last semester. For this meeting, we want to begin by discussing what each member of the team would likely be working on for the majority of the semester as well as the immediate steps for each member so we have a focus for the upcoming weeks. We then plan to discuss some of the more low-level actions for each team member, including what the expected common case should look like, as well as how the overall program will connect to Canvas. We want to spend some time looking over the schedule to see if it's still manageable. We want to leave some time to discuss problems that have come up since the beginning of this semester. Finally, we want to end by discussing both the requirements specific for SE492 and things for everyone to work on before our next meeting, which will be next week.

Meeting Notes:

- Began by discussing our positions for the semester, including what we would likely be working on.
- Discussed the current structure of our database and what we may have to /want to change in order to make it work better.
- Talked about an error involving GPT being incorrect in regard to a math question, it calculated 5-2 to be 4. Which obviously isn't correct but was wrong as well in coming to those numbers in the first place. Seems reproducible, but isn't always bad at math when given different numbers.
- Discussed how to connect to Canvas, including what we would eventually upload and whether we could do it or not.
- Uploading PDFs using the REST API is hard, so in the future, we should consider using the Python canvas API.
- Made the decision to focus more on the common case for the average student instead of testing so many edge cases early on.

- Potential solutions were brought up to solve the math problem, including more testing, adding a wrapper that helps with math, or simply giving the AI answers beforehand.
- Ability to automate testing by use of serialized langchain conversations, this way we can test specific things without worrying about varying AI behavior.
- Ended with the understanding that the agent developers would continue to work on conversation flow and the frontend/backend developers would work on making communications between the different modules work
- We didn't get to talk about requirements for SE492, this is because our meeting got cut short, we also didn't have a long discussion on the next steps for everyone for the same reason.

Summary
Main Points:

- We have encountered an issue where OpenAI is claiming an answer to a question is actually something that is incorrect. AI doubles down incorrect behavior when we try to reason with it. After having discussed it in the meeting, we believe that this is simply because GPT3.5 is just not great at math. This behavior is consistent, and we were able to reproduce it every time. Interestingly, it isn't always bad at math, and in some other cases with slightly different numbers, it works as expected.
- We discussed how we want our program to interact with Canvas. Based on our design doc, we plan on running it as an LTI (Learning Tools Interoperability), with a slight difference as an LTI asks for a URL submission, but we plan on submitting a PDF. After some discussion with our advisor, it seems like this is still possible.
- To better automate development that makes it easier to test specific changes, we discussed creating more serialized 'scripts' that we could simply run on OpenAI using LangChain so that we can get to the desired change faster and with less use of tokens. By doing this we would spend less money and remove unwanted variables when developing by starting with the desired state that we want to check a change in every time and not having to rely on OpenAI's sometimes random responses.
- A possible future decision that we haven't quite decided on is potentially creating a branch off of the main program website so that the frontend developers can test a basic LTI to make sure that it successfully connects to both Canvas and the backend SQL database.

Decisions Made:

- For developing the actual agents that will be used to administer the assessment, we decided to focus more on the common case for the time being and circle back to include more of the edge cases that we have pointed out may come up in our design doc. This change was brought up because we were worried about these edge cases early on, but our advisor thinks that we need a base common case that will work for the average student before getting into that.
- For frontend development, We made the decision to use the Canvas Python API for future calls to communicate with Canvas instead of using the default REST API. This decision was made for a couple of reasons. Firstly, we will be using Python, so it makes sense to use the API also, some API calls are tedious and tough to make using simple GET and POST operations. The operation that was giving us the most trouble was uploading files from your device onto Canvas, it appears that the Canvas Python API will make this much easier for us.

Future Actions To Take:

- Researching and experimenting with GPT to see what may possibly fix the issue of GPT giving incorrect information for the quizzes. We came up with a few ideas, including upgrading to GPT4.0 as well as simply telling GPT what the correct answer is so that it can correctly grade the students. The immediate action will be to do more research and more testing to see if we can find a simple solution.
- An issue we ran into that we made no decision on was that when we were using LangChain to save conversation history, GPT was overloaded, and ran couldn't process all of the conversation. While this isn't a pressing issue because it's the first time that it has happened, so we don't believe it will come up very often, but it does remain a future issue that we will need to address. We believe that this will be especially apparent when doing user testing and we get to see how many users actually do take too much time writing responses. We currently have no prediction of how many students or what percent will be affected by this.

Next Steps for the Project:

- For our Frontend and Backend developers, the next steps include getting a running database as well as determining how to automate the process of uploading a fully graded PDF of the assessment, one that Canvas will be able to read and automatically put into the grade book.
- For agent developers, the next steps include continuing to work on the conversational agent as well as implementing the proctor agent so we have a seamless conversation between the agent and the student. This includes the action mentioned above of finding ways to prevent GPT from producing incorrect answers.