

VIP VP4 - Textbook Summarizer using ChatGPT

Fall 2023 API Team Project Proposal

Membership:

Member		
Veena Gonugondla	Skills	Java, Python, JavaScript, HTML/CSS
	Credits	2
	Responsibility	<ul style="list-style-type: none">– How to utilize OpenAI API and integrate it within our existing application.– Researching best methods for keyword extraction (MonkeyLearn vs NLTK).– Researching prompt engineering for more specific responses from ChatGPT (Ex: telling ChatGPT to create an appropriate summary using the keywords)
Divya Mundkur	Skills	Java, Python, Google Cloud, C
	Credits	1
	Responsibility	<ul style="list-style-type: none">– Researching about calling an API that given a text returns a list of keywords with their relative weights of occurrence.– Once these keywords are extracted, determine a way of sending ChatGPT the text and keywords to generate a summary that includes the keywords.
Pearl Jain	Skills	Java, JavaScript, Python, Flask, SQL
	Credits	1
	Responsibility	<ul style="list-style-type: none">– Ways to integrate OpenAPI into the existing application with addition of a ChatBot like feature.– Addition of a database schema and system to store prompts, summaries and responses from ChatGPT.
Rachit Gupta	Skills	Python, Flutter, Java, Kotlin, MySQL
	Credits	1

	Responsibility	<ul style="list-style-type: none"> – Integrating the OpenAI API functionalities with our custom RESTful API, making querying more consistent throughout the project. – Researching the most optimum keyword extraction library to use—something that’s free, open source, and doesn’t have a querying limit. (e.g. NLTK) – Optimizing the keyword extraction to summarization generation pipeline
--	----------------	--

Project Goals:

Through this project, we want to be able to actively contribute to the existing ChatBot recommender project that was started a few semesters ago. We want to tackle the challenge of how to integrate the ChatGPT API and MonkeyLearn API to summarize a given chapter in a textbook that highlights the keywords of the chapter. This would then help the other sub-teams of the project work on connecting chapters with similar keywords through a graph network. We hope to be able to achieve this by doing our research on how we can leverage OpenAPI, and integrate that into our application to generate useful and concise summaries.

Project Timeline:

- Week 1-2: Team formation Project Selection
- Week 3-4: Sub-team formation, Project proposal draft and refinement
- Week 5: Create repository, fork existing project on GitHub enterprise, look into the existing code on Github, and research advantages and disadvantages of MonkeyLearn vs. NLTK.
- Week 6: Create Flask API schema, Integrate OpenAI API and test API calls on sample text.
- Week 7: Build a keyword extraction pipeline for textbook keyword extraction.
- Week 8: Research prompt engineering and ways to optimize prompt into ChatGPT.
- Week 9: Feed keywords and passage into ChatGPT that composes accurate summary using keywords.
- Week 10-11: Meet with front-end team to integrate both of our work into existing application
- Week 12: Improve the keyword extraction.
- Week 13: Testing, adjusting and optimizing accordingly.
- Week 14: Final Presentation.

Milestones:

Our first milestone would be to test the API calls on a small sample text. We would first test the call with the MonkeyType API to test if the keywords extracted are relevant to the text sent in. We would then test this sample with ChatGPT's API to see if the summary generated by it given these keywords would concisely explain the content while also allowing it to be linked to different chapters. Our responsibilities are outlined above in the membership chart.

Project Description:

The project is based on digitizing a textbook with easy access to chapters. Our goal is also to link each subsection to other related subsections for readers to navigate the textbook with ease. To do so we will be needing keyword extractions and ChatGPT API calls to link these chapters. Our aim is also to use this ChatGPT API integration for more functionality such as chapter summarization. Our project includes the ChatGPT API integration with the existing frontend framework. We are currently using the Python Flask framework and there will be a need to store the calls into a database. The user will be able to see a visually appealing graph representing each subsection of a chapter and linking them to subsections of other chapters based on their similarity.

Foreseeable Challenges:

Group Challenges: We're all new members to this VIP so it may take some time to get to speed and familiarize ourselves with the code in the existing project.

Veena: I see some possible challenges in finding ways for better keyword extraction. However, we can use MonkeyLearn, a machine learning platform for Text Analysis. It features a pre-trained keyword extractor that we can use which will hopefully make this process easier. With better keyword extraction, we can create more accurate summaries which will be more helpful to students.

Divya: I see potential problems with the scalability of the project. Since we have to feed in such large data sets to the APIs, I am not sure if we would exceed the limit of calls we can make to these APIs. I also see a potential challenge with learning the tech stack that is currently being used as I primarily have experience with Java. However, I do believe I can overcome this issue by researching and studying the existing code to see how we can adapt it.

Pearl: Feeding data into the ChatGPT API may be a potential issue from the backend aspect of the project in terms of transferring data from the user interface. There could also be a gap in connecting all features from backend to the API while also integrating it into the front-end, all in a given period of time.

Rachit: I believe the biggest challenge for us would be to enhance the usability and extend this project to work beyond just the technicalities of the VIP. I would love to see it being used by students who take the ECE 2026: Signal Processing class in the coming semesters.

Future Additions: Once we are done with the minimum viable project to meet our project goals, we hope to incorporate more of the features that ChatGPT offers into our project. Considering the textbook in discussion would not change unless there is a newer version, we would also like to incorporate caching to avoid multiple API calls to fetch the subchapter connection every time. Depending on the output results of ChatGPT and the kind of prompts given to it, we aim to design a database schema. We plan to build the database system using MySQL or SQLite depending on integration ease. We will also be looking into NoSQL databases like MongoDB considering the responses from the API will be in an unstructured format.

Implementation and Teamwork: In addition to the general group meeting time at 12:30 pm on Wednesdays, our group plans to also meet virtually every week on Friday at 4:00pm to update each other on our progress and debug any issues. We have also created a group chat on iMessage to regularly communicate.

Implementation Tools/Resources/File Management: We will be using the provided ECE 2026: Intro to Signal Processing textbook as the basis of our project. We will use GitHub and Git to collaborate on a single codebase by working on different branches. Once we are done with our individual parts, we plan to spend time combining our work together by merging branches.