

Title: PDF.ai

Who:

- Eddie Kiernan
- Amber Perillo
- Wei Jiang
- Estelle Girard
- Helen Garabedian
- Justice Asamonye

Project Description:

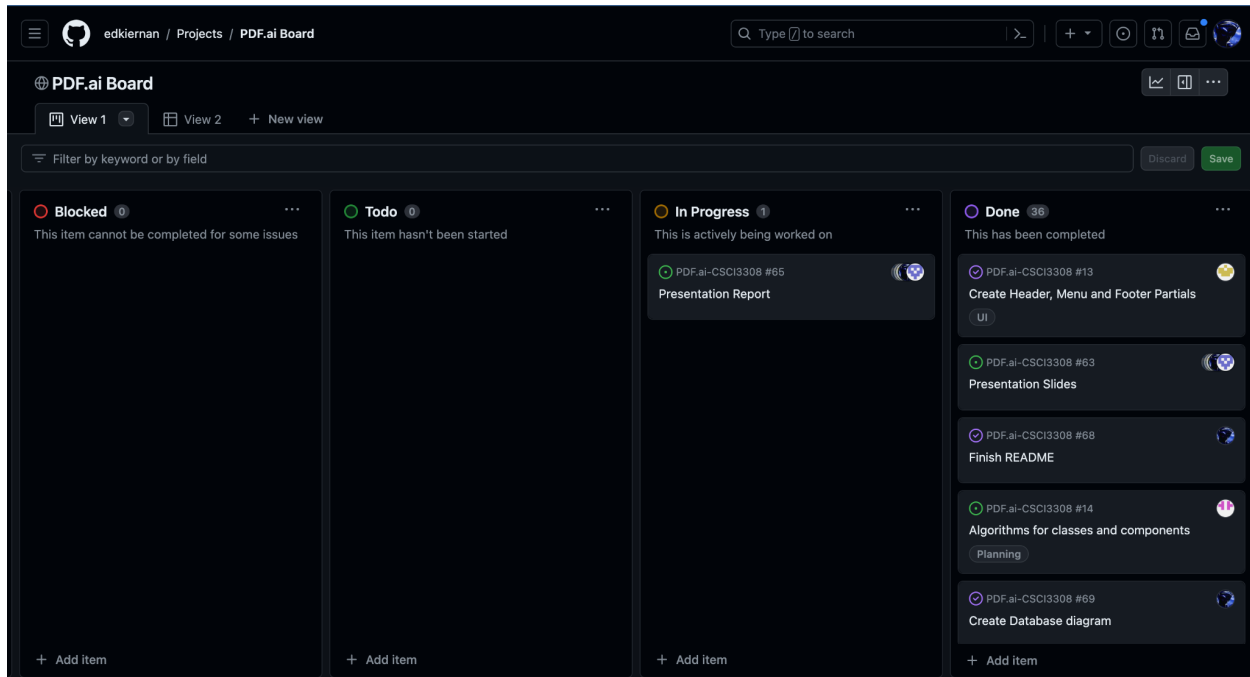
Faced with the challenge of deciphering complex textbook language and the struggle of sifting through comprehensive documents, we conceived and developed PDF.ai. As a pioneering document assistant, PDF.ai revolutionizes the process of understanding and note-taking from textbooks and documents. Its distinctive capability lies in extracting text to serve as a conversational context for AI language models, facilitating the creation of content summaries and engaging discussion about the material within the PDF, informed by the text itself.

Empirical evidence indicates that PDF.ai can yield a substantial time saving for users. Observations from personal usages have shown time savings of over 2 hours weekly when taking notes from a textbook, while enhancing overall comprehension.

The integration of Google's Vertex AI within our Node.js application, framed by Express and EJS, introduces intelligent, interactive elements, marrying real-time responsiveness with an enhanced user experience. Coupled with Google Storage, we ensure a protected repository for PDFs, providing users with seamless and quick retrieval for ongoing or future reading sessions.

Project Tracker - GitHub project board:

- Link to our Project Tracker: <https://github.com/users/edkiernan/projects/1/views/1>
- Screenshot showing our project board in our project tracker:



Video:

<https://github.com/edkiernan/PDF.ai-CSCI3308/MilestoneSubmissions/projectvideo.mp4>

(Also in Milestone Submissions folder in github repo)

VCS: <https://github.com/edkiernan/PDF.ai-CSCI3308>

Contributions:

Eddie Kiernan: The main parts of the project I contributed too are the following: created the database using SQL which saved previous users' uploaded pdfs, the front end of the login and register pages modeled off of the labs throughout the semester, fetch and getMaxPages api routes on the back end, automated test cases to test the application in the server.spec.js file, and periodically updating the README as the project advanced.

Amber Perillo: My main contributions to our project were implementing the PDF viewer partial where after the user logged in and uploaded their file, the pdf would display properly on the left side of the screen with a scroll bar and other functionalities. I was also responsible for making sure button functionalities worked such as the next page, previous page, and the jump-to-page, and that they coincided correctly with the PDF. I utilized Eddie's getMaxPages route and made sure other edge cases were taken care of such as setting the first page of the pdf properly. This was all done on the front end,

although challenging. I also contributed to some API routes such as for the login, registration, and logout pages, the front-end for the help page, and the styling and color scheme for our website as a whole.

Wei Jiang: My primary contributions in the project were centered around the development of the Chat feature on the front-end and managing the external API integrations on the back-end. The Chat feature I designed enables users to generate summaries of the displayed PDF page and inquire about textbook content through natural human language. On the back-end, I was responsible for implementing API calls to Google Storage and Google Vertex AI, with the corresponding code housed in `message.js` and `storageHandler.js`.

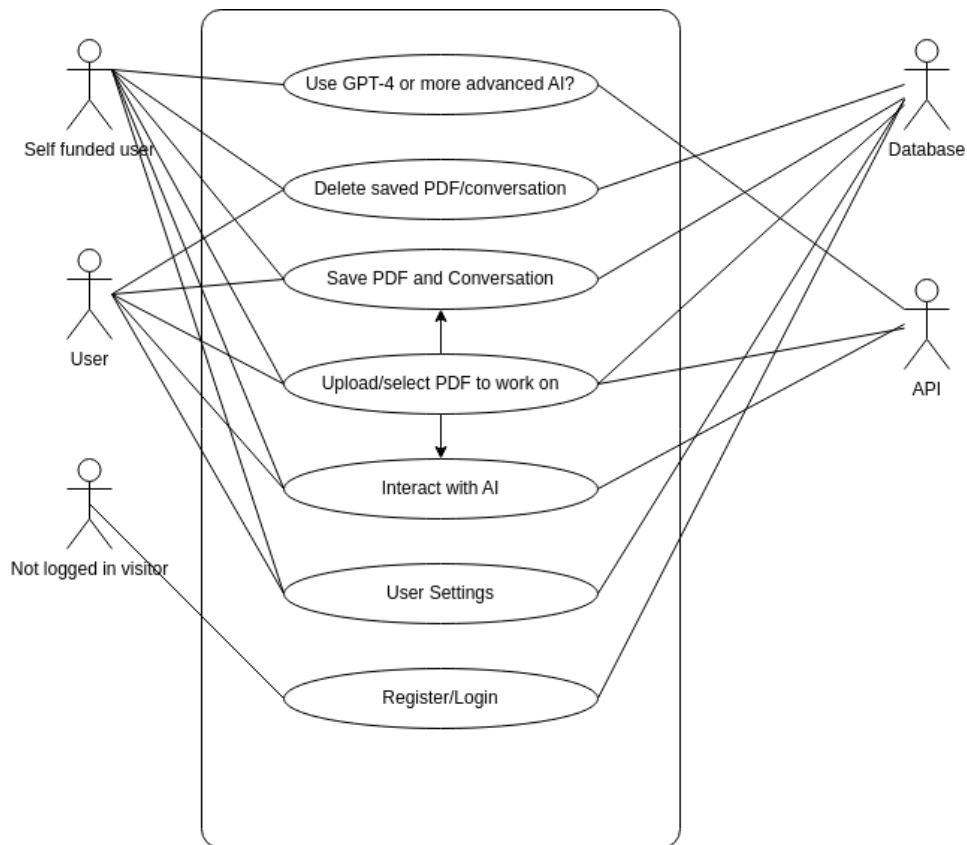
Additionally, I contributed to the backend PDF processing workflow by developing functions for loading PDFs and extracting text, as detailed in `pdfUtil.js`. I also established API routes for two key functionalities: retrieving and uploading PDF documents, and interfacing with Google's Vertex AI API to obtain text summaries, which can be found within `index.js` and `message.js` respectively.

Estelle Girard: I mostly created partials to be used throughout the website, like the headers and footers, and the upload partial, etc. I also contributed to the css styling with Helen to try to fix some issues with the display, and I created a history page.

Helen Garabedian: My main contributions to this project was on the CSS style guide that was applied across the entire site. I also worked on creating the help / how-to page.

Justice Asamonye: My main contributions to building our PDF summarizer were in the initial setup and interface of some of the different pages that the users would have access to when using the pdf summarizer. I also was responsible for creating a settings page that would give the user access to their account information as well as some of the preferences they may have while using the summarizer.

Use Case Diagram:



Test results: This project was tested in two ways. The first way was through automated test cases written in the `server.spec.js` file. The automated test cases tested four api endpoints in our application, the `/login`, `/register`, `/get-page-summary` and `/getMaxPages` endpoints. These endpoints were determined to be tested as they returned important information required to enter our application and important information regarding an uploaded PDF. For each endpoint positive and negative automated tests were run and edge cases were tested to ensure no bugs were found. The second form of testing came by observing a user outside of our team use the application. The observed user was seen intuitively registering and logging in as expected. Same can be said for the uploading of a selected PDF. When it came to chat with the bot and retrieving summarization the user was seen as impatient as the chat element takes a few seconds to return to the user. As a stretched goal, we could add a message telling the user to please be patient when using this part of the application. The observed aspects of interactions matched with our use case diagram and our automated testing, making our testing overall successful.

Deployment:

- Local
 - Localhost link: <http://localhost:3000/>
- Deployed
 - Azure link: <http://recitation-14-team-03.eastus.cloudapp.azure.com:3000/>
 - Screenshot:

