The application I tried to create was a fake Spotify web application that uses an AI API to generate fake artist information based on user prompts. Initially I tried using OpenAI to generate artist details like name, album names, album image, and songs within that album. I quickly ran into usage limits and a pay wall which prevented me from making enough requests on the free tier to fully test the funcitonality. After that I experimented with several different Google Gemini models to see if I would be able to get some kind of different results, but it came back to the same thing. Because of the limitations, I was not able to confirm if the AI output was going to be formatted correctly, but I did try to make sure that it would format the output as best as I could. I did leave the env file in there for anyone willing to try and test it. While using AI during the development process, it was very interesting to see how it works it did simplify the process for coding, but it made quite a few errors. Which led to a good amount of debugging. But it was a very good tool that helps immensely by saving time as well as some headaches. To compensate for my inability to actually test the AI generation, I did add in a random method that was a fallback net in case the AI generation wouldn’t work. It's within a try and catch that would try the AI generation and then make a call to the random method. It pulls together a string of random words within a list to create the artist's info. But there are no album pictures or actual audio with the tracks. The front-end stuff is pretty minimal and really only for functionality. Overall, this assignment was challenging but it was enjoyable. It allowed me to apply some concepts that I have been learning at my internship while also experimenting with AI tools.