Noah Falanga

Capstone Assessment

The project that my team is working on is smart music recommendation app using the Spotify API. There is an intention to create multiple ways of recommending music, including playlist creation, toggles for how similar or different the user wishes the music to be to their established taste, and if time permits, a social aspect that could allow for mapping of people with similar taste as you. In the lens of my academic experience, I will be utilizing skills learned both through class and internships to create this app. As I am a computer science major, I have taken classes that taught software design, software engineering cycles, databases, web development, and artificial intelligence. Through co-op, I took the theory learned in class and put it into practice, allowing me to fortify this knowledge into tangible skills. I plan to utilize and showcase all these skills in this application, showing that my time spent studying has been fruitful and effective.

As mentioned previously, there were a slew of courses that I took that provided me with skills that I will be using on this project. I could list very many, but for the sake of brevity, I will focus on core skills that I believe to be most relevant to our project at hand, and listing the courses that helped me with those skills. The skill of knowing efficient software engineering practices was taught in the class of the same name: Software Engineering, EECE 3093. This skill and the knowledge gained from the class will be used to help plan our project, execute that plan, and make changes effectively. I was taught how to use relational databases in course CS 4092, Database Design and Development. I took many classes that taught many aspects and types of artificial intelligence. Those classes were CS 5173, Deep Learning, CS 4033, Artificial Intelligence Principles and Applications, and CS 5135, Learning Probabilistic Models. Finally, I acquired web development skills through a small amount of exposure to it in CS 4065, Computer Networks.

The skills learned in class were important, but putting them through trial at an actual job was what solidified them. My co-ops also had the benefit of allowing me to develop not only more hard skills, but soft skills as well. My first co-op was at CADTalk as a Junior Software Developer. There I bolstered my effective communication skills, as I had to work with both co-located and remote coworkers in different meetings. I also participated in agile sprint meetings, which helped improve my software engineering skills. Because I worked so closely with the CEO, and our work was a lot of building from scratch as opposed to expanding pre-existing code, I got to learn a great deal about intelligent software design as well. More skills learned there were C# and .NET, and additional database knowledge. At my next internship as an AI R&D Software Development intern at Siemens DISW, I expanded my knowledge on artificial intelligence, in addition to gaining cloud development experience. In terms of soft skills, I developed independent drive, as my work was largely independent and almost completely remote, so I had to take charge of my own productivity. My final internship was done at Worldpay, where the only relevant skills gained here and not mentioned elsewhere was web development.

I am excited to work on this project because it combines two things that I'm incredibly passionate about: computer science and music. I have felt that Spotify, while its in-house recommendations are decent, often fails to give good artists and songs to users consistently. Its social aspect, while present, it also a bit lacking. Providing a solution to this would be greatly benefit me and many of my friends, as they are also music fanatics who are a bit disappointed in Spotify's lack in this area. Our preliminary approach to designing a solution involved meetings where we discussed lacking areas of Spotify. We discussed the problems that were present, and proposed solutions that could potentially be better. For example, we all disliked the blend feature, as it often catered to one person's taste more than the other. A proposed solution was a feature not unlike blend, but categorized people's tastes, assigned them a score, and instead of pulling songs strictly from two people's profiles and putting them in a playlist, we would pull songs from the original user and a pool of songs that was created by people with a given similarity score.

We have a collection of expected results involved in our solution. The ones that we plan to have for certain are an aesthetic, easy to navigate graphical user interface, a collection of recommendation algorithms to choose from and modify, and full integration with Spotify via its API. If feasible, we also plan on implementing a social aspect as well, allowing users to connect and share music with both strangers and friends. Our criteria for success for the graphical user interface is difficult to quantify, but we would consider it a success if, when showing the app to friends and classmates as a test-case, they are able to navigate it successfully and rate it as visually appealing. The prediction algorithms will be considered a success if they are both different enough from Spotify's prediction algorithms, produce playlists of which the end user is satisfied with in terms of variety and selection, and are able to be modified by the user depending on select parameters such as genre, similarity, and many others. It is also important to note that the second point does not necessarily need to expose the user to songs that they know they would enjoy, but rather to introduce them to songs that fit their criteria; if a user listens to mostly heavy metal, for example, and chooses to have a playlist that pulls mainly from hip-hop and indie songs, its likely that they will not enjoy a good portion of those songs. However, as long as the playlist generated satisfies the parameters input, then it is still considered a success.