Olivia Axiuk
CART 351
Professor Sabine Rosenberg
Due 2025-11-19
Project 02 Peer-Review

Hugo Waterfall and André Neder de Almeida

Project: Family Feud

https://github.com/Hugo-Waterfall/CART-351-Web-Files---Hugo-Waterfall/tree/main/CART-351-Project-II_Hugo_Andre

Hugo and André's application was based around the *Family Feud* game show, asking the user different questions and accepting a typed answer. What surprised me initially was the fact that the program actually also accepted answers for the poll aspect of *Family Feud*, ranking the answers in a JSON file based on how many times it was entered. In this way, the program could dynamically change depending on the amount of people who answered this poll, and on what kind of answers were entered as well. During the demo, I tried this myself. To test it out, I entered an answer first during the game, initially receiving a few points for it. Then, I entered this same answer in the poll section of the program and, there having been few players before me, upon replaying the game and entering the same answer, I saw now that the word was higher up in ranking based on the polls. Although it would usually be separate people playing and answering, it was interesting to see it happen in real time. Functionally, the game works well with regards to storing and displaying the information,

However, there were a couple of things which I found could be improved. Mainly, I found that the input from the player end of the application was finicky when it came to capitals and lowercase. The experience would be much smoother if this was fixed, as it was frustrating as a player to have put in a good answer, only to be met with a large red X due to having capitalized the response. As well, I did find that the visual aspect of the game could have been more fleshed out, such as the white background, which may look more polished if it was turned to blue, matching the rest of the game's design language.

Hubert Sia
Project: Branching text-based adventure game
https://hubertsia.github.io/cart351/index.html#
Note; Unfortunately Hubert's link to the GitHub repository is not working, so I have provided the link to his project Website (which should eventually host the link to the repository here instead)

Hubert's application was very familiar in terms of its functionality, being a path-based text adventure game. The user was greeted by a short text, then prompted to choose from the several options displayed in order to continue the story. The UI was simple and easy to understand, making use of the application smooth, so that it was easy to play the game without thinking too much about the surrounding interface. However, it would have been interesting to see the immersion aspect of the game be more developed. The interface seemed streamlined for a game set in what I understood to be a medieval fantasy, and so an interface design leaning more into those visual aspects would have been interesting. As well, the simple inclusion of images may have helped the different story beats feel more involved.

Although the application worked smoothly, I do feel like there could have been a way to also integrate the data-storing affordances of the JSON files in a more engaging way. Data storage in digital storytelling has a great amount of interesting potential, and it would've been nice as a player to really feel in collaboration with the story. While Hubert's project stored the answers, it would be nice if, for example, a player could choose a series of character traits at the beginning which would affect the outcomes of player actions down the line.

Eamon Foley

https://eamon1000000.github.io/CART351_WEBSITE/index.html
Note; Unfortunately Eamon's link to the GitHub repository is not working, so I have provided the link to his project Website (which should eventually host the link to the repository here instead)

Eamon's application, interestingly, also focused on a fortune-telling application which reflected the themes of Julia and I's project. However, instead of displaying a pre-determined text by the project creator's, the application instead returned fortunes which had been previously entered by past application users. This creates a direct link between past and present users, making the application not only interactive with itself, but also with those that had previously decided to leave a fortune. Eamon's application, however, did not display the past results of previous users, unlike our own project, which I find would be an interesting extension to the program. Not only then would everyone be able to view these fortunes, but it would also take advantage of the uniqueness of each result, having been entered by individuals.

Furthermore, I found it interesting that there was an option to enter a positive or negative fortune to the bunch, introducing a more conceptual aspect to the overall experience. However, I

do wonder if the fact that the application removes the pulled fortune from the well could eventually lead to issues if there are fewer fortunes entered than given. Although, I do believe such a problem could be avoided if the program did not delete the resulting fortune if the total amount of fortunes currently in the program were under a specified total amount. In any case, such a state was not reached as I used the program, and I found the overall experience of the fortune well enjoyable.