Project Description:

Scrapyard Scramble - A strategy card game I created where two inventors build robots in order to defeat the other robot. Both players pick cards from the same “scrap pile” so they are trying to choose the best resources in order to defeat the other robot. The goal is to get the most points when the game is over but there are also speed and intelligence stats that for each of them the robot with the most gets an extra 10 points.

Competitive Analysis:

The design of the game is based on card drafting board games like 7 Wonders. I found that there is an online version of 7 Wonders where people can play the game against each other online. The online version seems to just be the board game put into an online environment. My project is overall much simpler than 7 Wonders and it has a lot of similar concepts to it. The main different component is that my game is built to be 2 players and I have a play against a computer option which this 7 Wonders simulator doesn’t seem to have.

Structural Plan:

All of the cards in the game are stored in separate Card objects or in subclasses of Card. This is so I can easily call the giveScore function on all of them. The bulk of what the card game runs on is the pickCard function which does every necessary action once a card is picked including choosing the next move if they are playing against the computer.

Algorithmic Plan:

The most algorithmic complex part of my project is the AI in the 1-player mode and the different difficulty levels of it. For easy mode, the game just picks a random card, in medium and hard mode the computer uses a minimax algorithm to choose the next card. The difference between the medium and the hard mode is that the medium mode only thinks 1 move ahead rather than until the end of the pack and this is done by using a max depth. The minimax algorithm works by first checking which players turn it is, and then looping through all of the possible moves and checking the score of each and then if it is the computer’s turn, returns the

max and if it is the players turn, returns the minimum. This is all done recursively and the base case is when there is only 1 card left in the pack which then it just calculates the difference in score of the two players.

Timeline Plan:

I have already completed the minimax algorithm. I plan to:

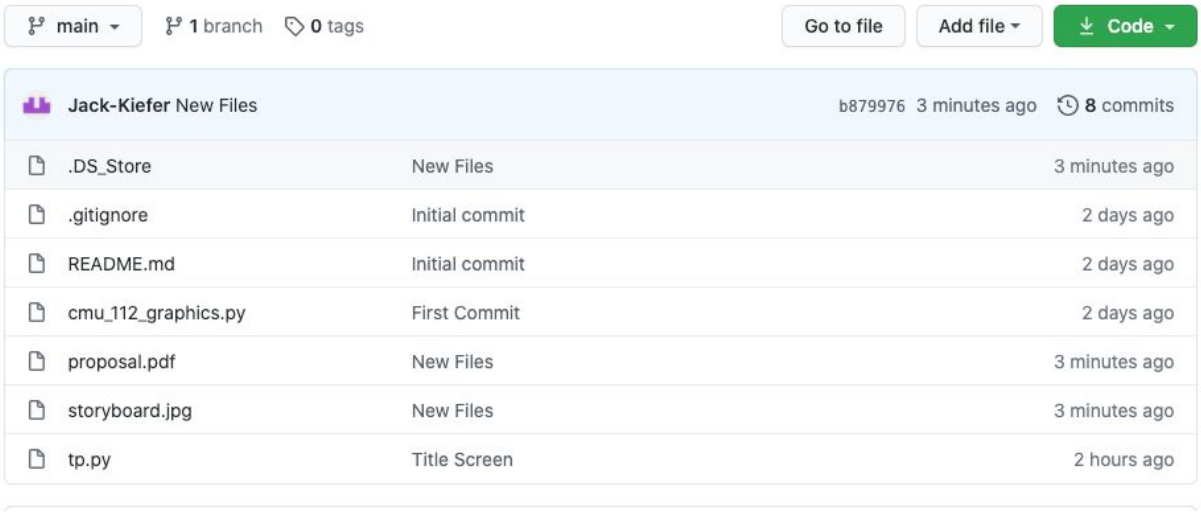
Wednesday, Dec 2 - Complete and decided deck of cards and features.

Saturday, Dec 5 - Add visual layout of picked cards and some pictures.

Monday, Dec 7 - Complete game and visuals with only some bug and visual fixes.

Version Control Plan:

I am using GitHub for version control.



Module List:

I am not planning on using any external modules. Only 112 graphics.

TP2 Update:

I decided not to include the power core as my storyboard shows. I also decided to not have the picked cards show up as pictures on either side of the screen and just decided to show

the names because most of their effects are shown in the scoreboard. Other than that I mostly stuck with my storyboard design.

TP3 Update:

The new major features I added were:

- A timed game mode that uses a chess timer and if you run out of time you lose
- A rules screen with alternating preview cards every 2 seconds
- A hint button where in vs. ai modes, you can get advice from the minimax algorithm
- New pictures made with photoshop
- Red text that shows you how the scores will change if you pick a card