Zachary Kuchar Senior Project Paper 2 2-27-19

My Senior Project has been coming along through the first three weeks. Most game functionality has been completed. These next few weeks will involve finishing up the functionality that I have yet to program, as well as implement an online mode for games that could do so.

Week four starts off with finishing up the work I am performing on the solitaire game. I start by fixing the problems involving moving multiple cards at once. The problem seemed to occur with how I was handling the condition. While trying to change what the number was by subtracting or adding a one, a two-digit number with a one at the end would be created. This would make the program act like the set of cards to be moved could not be moved. Luckily, I was able to fix this problem with a similar method that I had used before. By putting a plus sign in front of the values that I was using, I was able to put adjust the values in the manner that I wanted. The next thing that I worked on was methods of dealing with an empty row, and how to move a king. Starting off, I set up the view so that I card back face would be displayed when a row is empty. I set up the id of a card displayed in a blank row to a value of empty. When moving a kind, it checks if the id pulled in is the word empty. If it is not, then it will not allow the move to happen. Finally, I moved into setting up how to remove a row of cards and calculate how to win the game. This part of the game was a bit tricky to set up. I needed to look at each row and determine if a row contained all the necessary cards in the correct order. If the row did, I would remove those respective cards and then place a king card down in a separate row to show that we have completed a row. I had to do some finicking with some if statements and variables in order to get this to work properly. I set up different variables with if statements that functioned based off what those variables were. In the end, I managed to get the function working as I intended it to. With the game at near completion, I took some time to look into switching the app to run on Node.js. Unfortunately, this presented more trouble then I thought that it would. There were multiple files required for a Node app to be deployed to Heroku. Even while being guided through an example, I experienced problems. I had managed to successfully run the app locally, however, I was not able to upload to Heroku successfully. Eventually, I was able to figure out the problem, and then get a test app working on Heroku in Node.js. Sadly, I could not get any further work done before the week ended.

I started off Week five by learning some things about node.js apps in Heroku. The first job was to determine how to load up a webpage. This would allow the main page to be loaded when the app is first run. The next step was to learn about WebSockets. The objective here was to learn how to set up WebSocket servers for the games to be given an online mode. Taking the time to learn about these topics online had been extremely hard and time consuming. At first, I managed a simple way to use express to load in the HTML. However, I struggled to load in additional files, such as the JavaScript and CSS. Eventually, I managed to find a way to figure it out. Afterwards, I spent time learning how to connect the server with WebSocekts. It took some time to learn how to run it correctly. After some time, I managed to figure it out, and the website was running on Node like it had previously had been with php. While working on this part of the project, I noticed that I had forgotten to set up a reset for the card game war. I decided to take a moment of time to make sure this part of the website was running correctly. The initial reset of the cards displays the card backs incorrectly, however, the process as a whole works as intended. Afterwards, I managed to find a way to set up message communication with the web socket server. I spent the remainder of my time setting up communication between the server and the webpage. I set up a basic scenario to test if communication worked properly with JSON objects. I would take the time to send messages with a stringified object. The server would send an object with information, and an action to determine what function to call. Unfortunately, a bug seems to prevent one of the correct functions from being called. Since, time was running short, I opted to save the problem for the following week and proceed to demo the project working on GitHub working through node.js.

Week six started with continuing work on the new server for use in online games. After fixing the previous error, I started off with work into integrating the online crazy eights game code into the new system. I set up an online button so that the setup in front of the user changed from an offline game into an online one. Then, I tweaked the original workings to update the status of the game through the original html document. An early issue that I noticed, was the inability to have multiple games going on at one time. Early attempts to fix this issue ended up less than successful. In the end, I opted to hold off on this problem until the main game was working online. During the setup, I also had some early trouble with removing and resetting the event listeners that were present on the given sections of the webpage. I found a way to fix this problem by cloning each of the objects that had event listeners. This allowed me to clear each of the existing event listeners on the objects present, so that I could put new event listeners on the objects for use in online play. I then continued to work on the other online functions for the game, including selecting a card to add to the pile, and picking up a card from the deck. After working through some of the kinks, I managed to get the game to be fully operational. The game was fully capable of being played with multiple people. As a final tweak, I set up the ability for a player to quit, and the other player to get notified. If a player were to quit out of an online game, the opponent would proceed to get a notification that their opponent has quit the game, and that they have won. In addition, when any player quits the game, by toggling the online mode, they will go back to a normal offline game, and they can once again can play against the computer. With time running low for the week, I opted to save on the multiple people problem for the next week. As a final procedure for the week, I began to make setups for the online modes of other games, mainly Snip snap snorum. I got the server to successfully send a message to and from regarding a test message. Hopefully by the end of the next week, I will be able to set a functioning version of the game for online play. Then, hopefully by the end of spring break, I can get go fish working in online as well.

My Senior Project is coming along quite nicely. Progress has been made on each game, and addition features are starting to be implemented. Given time, my project will hopefully have the polish that it needs.