Daniel Baliczek

Randy Sirota

CIS 356: Mobile Application Develepoment

November 29, 2016

Project: Peg Game

Throughout the process of making the Peg Game App, we have overcome many obstacles. Initially, we were very confused on how the movement would take place, and we wanted to write a function that would check an array for possible moves and then move the pins. Unfortunately we could not get that to work, and with the approaching deadline, we decided to find a work around and hard code the each possible movement in.

There were really three parts to moving a peg. First we had to identify the possible moves for each peg. The next step was to implement the moving of the peg. And the last part was deleting the peg that we skipped over and changing the color of the remaining blue pegs to gray. In our proposal, we wanted to implement the each as a different function, however, we decided to keep it all in one function to keep things simple and easy for us to keep track of. This took place of our getPosition(), renderBoard(), and changePosition() functions.

We both took part in doing this, and we pieced together the code to make the movement. Once the movement was made, we had to create a check to see if the user had won. This was by far the hardest part of the assignment.

Looking back, we each took it upon ourselves to do different tasks. While we both helped out on the movements aspect, Daniel focused mainly on creating a win condition and setting up a timer for the score, while also creating the icons and Gui. Randall on the other hand took it upon himself to make the tutorial and researched how to save the scores.

While keeping the score, we decided it would make more sense to just keep track of the fastest time, as this would be the main factor in calculating the score anyway. Our getScore() function was implemented, and it was accompanied by a saveScore() function that saved the high score even when the app was closed. After that we implemented a clear scores function to reset the scores.

Unfortunately, we could not think of a way to create a higher difficulty for the game. We thought about adding more pegs, but the problem with this is that it would not fit on a mobile version. Instead, we decided that we would need to cut this feature out, as it would not fit on the screen well and was also not achievable within our time frame. All in all, we split the work pretty equally for the project, but we set our expectations too high when it came to the difficulty setting.