Tap It Write Up

Erik Wurman and Jason Nawrocki

What we learned:

This project was more difficult than we anticipated. For starters, halfway through the semester Jason got a new Macbook preloaded with XCode 8 and Swift 3. Instead of reverting, we decided to trek ahead in Swift 3 for our project. The easiest part of our project was learning rotation and shake motion listeners, which determined if the user did the respective action and relayed that information to our model. The rest of what we learned was not so easy, but also had less to do with the mobile aspect of the project. This included all of the following: saving game high scores by writing them to a file and reading that file in; Inheritance; Using the built in Timer class; and Deploying to an iPhone for playing. The most complicated thing we learned was to manage and work simultaneous paths and then merge these paths via Github.

How much we got done versus our expectations:

We would qualify this as a successful project. We have a fully functioning and enjoyable game that utilizes much of what we learned over the course of the semester. We have 3 game modes and a tutorial, which was what we desired. Our gestures and motions are working correctly, although shake requires a rather significant shake by the user. In our pre-project presentation, we thought we may be able to add speaking into the game. This goal was never realized as we decided that the last 2 weeks would be devoted to making the game have a better user experience and to fixing bugs.

We did encounter some road blocks along the way. Specifically, Github took a while for us to figure out, but was very helpful in the long run. For a while, swiping only worked for the first person in multiplayer, but we did not find this bug until we tried to implement a replay button. This prompted us to make our Player Manager class, which gave us a bit of a detour when we thought we had finished the back end of the game. The Timer class also took some time to master, as it was new to Swift 3.

Our Player Names scene is a little broken, as it was a late addition to the game. The scene does not adjust correctly when rotated. Typing in the names of the players works, but not in the way that a user would expect. We think the correct way to have approached this would have been with a scroll view, but we frankly ran out of time to do this.

Overall, I think we set pretty high expectations, and while we didn’t necessary follow our original plan, our diversion into the front-end of our app made the game have a better overall user experience than a foray into exploring the microphone would have.

What would we have done differently?

First, our Player name scene would have been added sooner, and implemented with a scroll view. Second, the tutorial scene was also added late. Doing this over again, this scene should have been one of the first scenes made, as it allows us to test each gesture and motion as well as the game manager. Lastly, we might have explored different ways to make the model, like have three separate game managers for each game mode that all inherit from a generic model instead of the controller holding the logic for each game mode.

Grading:

What we think we deserve:

95%

Why?

We have a fun, fully functioning game that exhibits our knowledge of iOS capabilities learned over the course of this semester. We also explored other parts of iOS development not covered in the class. It took us up until the last day to get all of the features working, and all but the very last one we added are working as expected. The last one is working, just not in a user-friendly way.