Sprint 02 Progress Report

by Evan Arroyo and Joshua Sims

The goals set for sprint 02 were to establish a GitHub repository for the management of the project, to develop the foundational aesthetics of the game, to come as close as possible to a playable game with keyboard input, and to become more familiar with the Kinect.

Our GitHub repository, which is accessible via the link at the bottom of this report, helps us track issues related to the project and changes made to the project’s source code. GitHub alone has limited issue tracking capabilities so we have incorporated the use of ZenHub which is an extension of GitHub. With the inclusion of ZenHub, we were able to create epics (issues that describe large, overarching goals), associate more granular issues with those epics, and thereby more aptly track this sprint’s progress in relation to the overall project. In addition to creating issues, we have stored all of our C# script files in our GitHub repository. However, we have been unable to store the entirety of the project in the repository because its size exceeds GitHub’s limitations. Once our repository was set up, we proceeded to develop the game.

Before the development of the game mechanics, we set up the Pong arena and the foundational aesthetics of the game. The construction of the arena was simple. After a short time, the controls and logic behind arranging the arena’s constituent parts within the Unity editor became intuitive to us. The arena’s skin was also easy to include; there are many tools available for designing skins, but we used only the coloring tool. After completing the arena and placing the paddles and ball within it, we set the view of the game by moving a camera object above and behind player 01. We were then able to enter gameplay mode with a proper view of the game and were thus ready to develop the game mechanics.

A playable game of Pong implies several basic features about the ball and the paddles. The ball must behave correctly: it must bounce off of the paddles and barriers in a sensible fashion and it must disappear when reaching a goal zone. Furthermore, after reaching a goal zone, the ball must reappear in the center of the arena where it proceeds to move again. We implemented all of the aforementioned behavior. However, error handling has not been thoroughly developed so there is no guarantee that the ball will *always* behave correctly. We have inserted an epic into our GitHub repository which addresses the completion of error handling for every aspect of the game. Error handling tasks focused on specific aspects of the game will be represented as children issues of the error handling epic. Although it was not a goal for this sprint, error handling will be implemented for each of the features discussed in this report (as well as every other feature that will eventually be a part of the game). Following the behavior of the ball, we developed the behavior of the players’ paddles. Both paddles may now be user-controlled. Player 01 controls his paddle via the left and right arrow keys whereas player 02 controls his paddle via the “a” and “d” keys. In the future, we will allow competing users to play in a split screen view. Aside from user vs. user play, a user can play against an AI-controlled player. The AI that we implemented is simple – it moves side to side at a constant rate, thus imposing a basic challenge to the player. The ability of the AI-controlled paddle to move back and forth between the arena barriers relies on the same programming which confines the paddles to moving between the barriers and which enables the paddles/barriers to deflect the ball. This commonality was a great convenience to us. Lastly, after completing the game mechanics, we experimented with the Kinect to determine its capabilities.

Using the Kinect was a great experience. We were unable to connect it to a PC because the appropriate cord on the Kinect did not seem to fit in any of the PC ports. However, we did connect the Kinect to Evan’s Xbox One. We were surprised by how good the resolution was. The camera captured a large portion of Evan’s room and tracked our bodies well. Although we have not yet developed content using the Kinect, we now have a better idea of how we can use it when we do develop content with it. There is substantially more experimentation that we can and will do with the Kinect, but we feel content with what we have experienced thus far.

[The GitHub repository for this project](https://github.com/J-o-s-h-S-i-m-s/3D-Gesture-Controlled-Arcade-Games)