Sprint 03 Progress Report

by Evan Arroyo and Joshua Sims

The goals set for sprint 03 were to achieve proper deflection of the ball, to display the scores of the match as the match is happening, to implement a more sophisticated AI, and to start developing with Kinect.

In order to achieve proper deflection of the ball, we first had to know exactly how the ball should be deflected. We investigated this by playing the classic Pong game – specifically by playing the game at PongGame.org. We discovered that the ball is deflected as though the striking face of the paddle was rounded, like a semi-circle. However, the paddles in the classic Pong game are not rounded; they are rectangles. Therefore, the paddles must have been programmed to behave as though they were rounded. We attempted to do the same – to implement an invisible, rounded face for the paddles, but after unsuccessful attempts at creating well-suited invisible collider components, we decided to just make the paddles rounded. After successfully achieving proper ball deflection by making the paddles rounded, we had to reprogram the feature which keeps the paddles confined to the Pong arena.

Displaying the scores for a Pong match required delving into an aspect of Unity that we had not yet used. We were unfamiliar with how to implement a user interface. The code we wrote to display the scores was not difficult to understand, but it was quite new to us, so it took time to research the relevant methods and data structures.

The artificial intelligence that we developed this sprint is far more sophisticated than what we developed last sprint. Rather than moving side to side at a constant rate, oblivious to the ball’s position, the paddle controlled by the AI now follows the ball’s location on the z axis, moving only where necessary to deflect the ball. By changing the value of a single field, AISpeed, the AI can be made more or less capable of deflecting the ball – specifically, the AI will be quicker to get to the ball the higher that the value of that field is.

We did not implement gestural control of the paddles via the Kinect, but we have become more familiar with the Kinect and we can now link it to a PC rather than an Xbox. Fortunately, the gestural control of the paddles is not a prerequisite for any other functionality; it is a self-contained issue. Therefore, we believe that implementing gestural control during the next sprint (rather than having implemented it this sprint) will not involve rewriting the code for other, existing features.