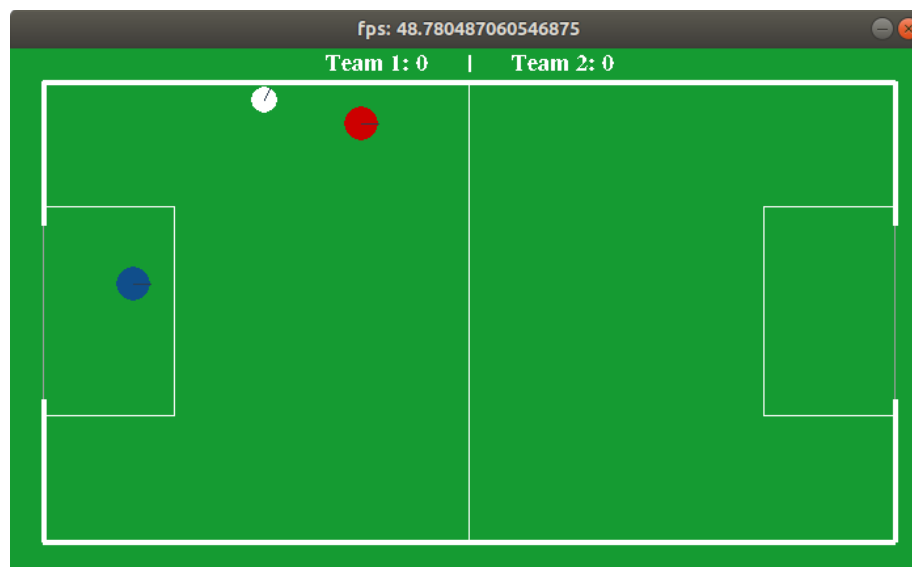


For the final project in computer science 310, we are developing a game along with AI to play the game. The basic idea behind our game is that it is a soccer game. One human player will control one player and we are developing an AI for the user to play against. It's somewhat simple, as it is a top down view of a soccer field, and the players are circles on the field. We have made a lot of progress with this project, namely finishing the development of the mechanics of the game as well as starting to develop the artificial intelligence.

The first thing we did to complete this project is actually create a game to develop an AI for. This stage took a long time to develop, as we had to do preliminary research on different tools to use, as well as actually learn the framework and develop the game. We chose to use Pygame with the Pymunk library. Pygame allows you to create games in python and Pymunk works with Pygame by adding a physics engine that you can use. Implementing a physics engine was the most difficult part, since we wanted the ball to bounce and move around the field in the most fluid way possible. We had to first learn how Pymunk works, and then play with the various settings and code to get the results we expected. We set up the field, the player objects, and the ball all as physics objects, so that they interacted with each other in the ways we wanted. This included adding different weights, and pulses which occurred when a player would touch the ball. You can see in the below image what the game looks like, as well as all of the different physics objects in the game.



The next thing we had to make in the game outside of the physics of the objects interacting with each other is actually being able to control the players. We used some Pygame functions in order to collect the user's pressed keys on each frame. We then used these keys to determine which direction the user would go. So, if the user pressed up, then the user would move up the Y axis. However, this seemed very rigid and not intuitive, so we implemented some

smoother movement mechanics. To do this, we stored the user's speed in each of the four directions. Then, we would increase these speeds depending on the keys that the user pressed, so if the user presses up, then the up speed will increase a certain amount, but if the up key is not being pressed, then the up speed will decrease a certain amount each time. This made it so that the player would move around the field with acceleration and deceleration in each direction making for a more fluid experience.

The final task we have completed so far is starting to develop the AI. We want the user to be good enough to play the game against a human and be able to score goals on the enemy as well as protect their own goal. The first step of this process is getting some functionality to get the AI to move around. To do this, we created a function which just determined a series of actions the AI can go through in order to move to a specified X, Y location. This function returns a series of key presses that would need to be carried out in order to move the AI to that location. Then, this series of key presses will be input into the movement function we talked about earlier in order to move the object. In order to determine the key presses, we calculated the Y distance the AI is away from the specified point as well as the X distance. Then, we divide the each distance by 3 because that is the top speed, and then that is the amount of key presses you should input in that direction. Then, we input all of the Y key presses first followed by the X key presses. This moves the character up or down to the ball and then left or right to the ball.

We faced many challenges in implementing our AI as well as deciding which library would be best for creating this sort of game. One of the first challenges we encountered was when we attempted to make our own physics engine. We soon realized this would not be possible and began looking at libraries in python which would support our game and still give us the opportunity to implement an AI. After searching through the web we found that pymunk would be our best choice as we saw good game examples with it that we could see us modifying to work for our game. Other challenges that we have encountered are glitches within the field of play. The physics engine on pymunk works as it should but when the ball is pressed against a corner at times it will pass through the barrier. This is something we can't do much about since it has to do with the actual physics object in the pymunk library. Other challenges we faced was deciding whether or not we wanted the user to actually shoot the ball or to just push it instead. We had a shooting function which we implemented but decided the AI would run better without it so we omitted it from the code.

In addition, we found that we are going to next implement a finite state machine which will allow the AI to be in either an attack, defense, or midfield state. This will help the AI accomplish the tasks of scoring goals and defending goals and know when to do when. As of right now the AI can track the ball by taking the slope. What we plan to do for the attack and defense is that when the ball has crossed a half of the pitch say the half the AI needs to score on it will track the ball and attempt to push it into goal. If the ball is on the side of the pitch the AI is supposed to defend then it will go to the goal line and track the ball from there. The midfield state would be of attempting to get the ball away from the defense side and pushing it back to the attacking side of the pitch.