Zach harwood

330.01

project 2 game

Documentation

For this project we were asked to create our own experience or game using JavaScript. Me and my partner chose to recreate the arcade game centipede in java script. My partner handled most of the first part of the project media requirements. My partner created all our sprites in a sprite editor named pascal. He created all the sprites you can see on the page so that we could stay consistent with how our game looked. We ran out of time when it came to animations and to particle systems however if we had time we would have animated the ship so that it swayed when it moved or animated the legs on the centipede. We would have added a particle to the ships weapon so that the movement could be conveyed more easily. To keep on theme my partner also found a pixel font to be used for all of the games font. Lastly, we chose a chiptune type music to get the feel of the old arcade games from which this game originated

Next, we move onto the interaction which was a hassle to do when working on our ship. The only controllable sprite on screen is our space ship it can be moved in all directions and shoot a laser from its cannon. We used a key daemon to make sure the controls worked as smoothly as possible. The laser is controlled by the space button and can fire many times in succession. If we had more time I would have added a cooldown time to the laser so that we could regulate the number of bullets fired from the gun. We also made it so that you can not move past a certain point in each direction calling this area the play zone. After you try to leave the play zone you are bounced back.

We now move onto the usability requirements. We made it so that if you exit the game in the browser the game becomes paused and when you click the window again the game un pauses. The game is simple and teaches the play with the starting screen. The game invites the player to play with the controls and figure what is the best away to defeat the centipede. You can even dodge the centipede when it enters the play area. Every time the play defeats the centipede it spawns a new centipede to the top of the screen. This new centipede is faster then the last one and has more segments this causes a slight issue where if the game lasts to many rounds the centipede can over accelerate leaving the play screen. Our game tells you all you need to know from its title screen to the game over screen which displays your points and asks if you want to play again which restarts the page.

For the game design requirements, we mainly followed our original plan. The game is played by one person piloting a ship. The game has rules like you can’t leave the play area and the centipede gets faster, the game doesn’t have a win state, but it does have a loss state when the player gets hit. When you achieve the loss state you see how many points you gained. The player has to make meaning full decisions like where do I want to shoot or where do I want to stand to give myself the best chance of surviving. Our game has a moderate amount of death like how you can use the quirky blaster in different ways depending on how you hold the space bar or tap it or even using it against a wall. The experience over all is also fun seeing how long you can last and if you can beat your own high score.

Lastly the coding requirement for this project I met all the requirements of the tech stack I chose. I used ES6 classes, ES6 modules, and I transpiled my code to ES5. Me and my partner made sure to use the right code in the correct places. if I were to redo this code I would use more classes to make my code more manageable. Me and my partner also made sure to create a function whenever we were not sticking to the concept of D.R.Y. The only code we need to borrow was how to make delta time which came from a tutorial found here <https://isaacsukin.com/news/2015/01/detailed-explanation-javascript-game-loops-and-timing>.