Dale Rivera ID: 922792075
Github: darvr1 CSC317 Web Development

Assignment 3 - Javascript Individual Game Assignment

Note:
Works best on firefox (text renders a little differently on chrome).
The goal is to protect the earth from incoming asteroids. A and D to move left and right. Holding down the spacebar makes you move faster. If there seems to be a problem with the controls being too fast, check the last paragraph in 'Issues and Resolution' (tldr I don't know how to fix it).

Description:
This assignment is to create a video game using JavaScript.

Approach / What I Did:

Initially, I wanted to make a bullet hell game but before I did that I needed to understand JavaScript first because it was new to me. I looked up tutorials and watched youtube videos about the language and eventually came across the html <canvas> element. I spent about a week learning how to use <canvas> and throughout the process, led me to learning other topics about JavaScript such as event listeners. Once I felt comfortable using <canvas>, I started creating my bullet hell game but it was harder than I thought.

I looked up games made on canvas and saw an asteroid shooter type game so I used that as inspiration. My game ended up becoming this asteroid shooter where instead of the player as a moving spaceship, the player is controlling a weapon on earth that shoots asteroids.

Issues and Resolutions:

My first problem was collision detection. I started by adding boxes and making them hit each other and doing a console log message to see if my collision function works; it didn't work. It took me a while to figure it out but I needed four different if statements to check if the bottom of the first box and the top of the first box hits, and the same process for the other three sides which was tedious and a little annoying to do because the boxes' origin start at the top left. My solution to the problem is to use circles instead because <canvas> draws it from the middle of the shape so it makes collision detection easier by only checking the position of the two circles plus their radii which is much better than checking the four sides of a square.

My second problem was the different font styling between chrome and firefox. I created my game in the virtualbox so I used firefox to play my game. When I played it on chrome on windows, the texts were not aligned even though none of the font style and font size changed. I looked up the problem and it had something to do with the native browser font styles and a solution was to use some kind of css called 'reset css.' I didn't use the method to solve it because of time limitation and I'm not sure if it will mess up my game.

My third problem was creating a proper scaling difficulty. Due to the nature of the game, increasing the amount of asteroids is the only way to keep engagement but my problem was that everytime the player hits an asteroid, 'difficultyScaling' increases by a flat amount. This is bad because at a certain point, too many asteroids will start spawning from seemingly out of

nowhere. While it wasn't effective, I counteracted this by having an exponential decay function so it takes longer to reach an "infinite" amount of enemies spawning.

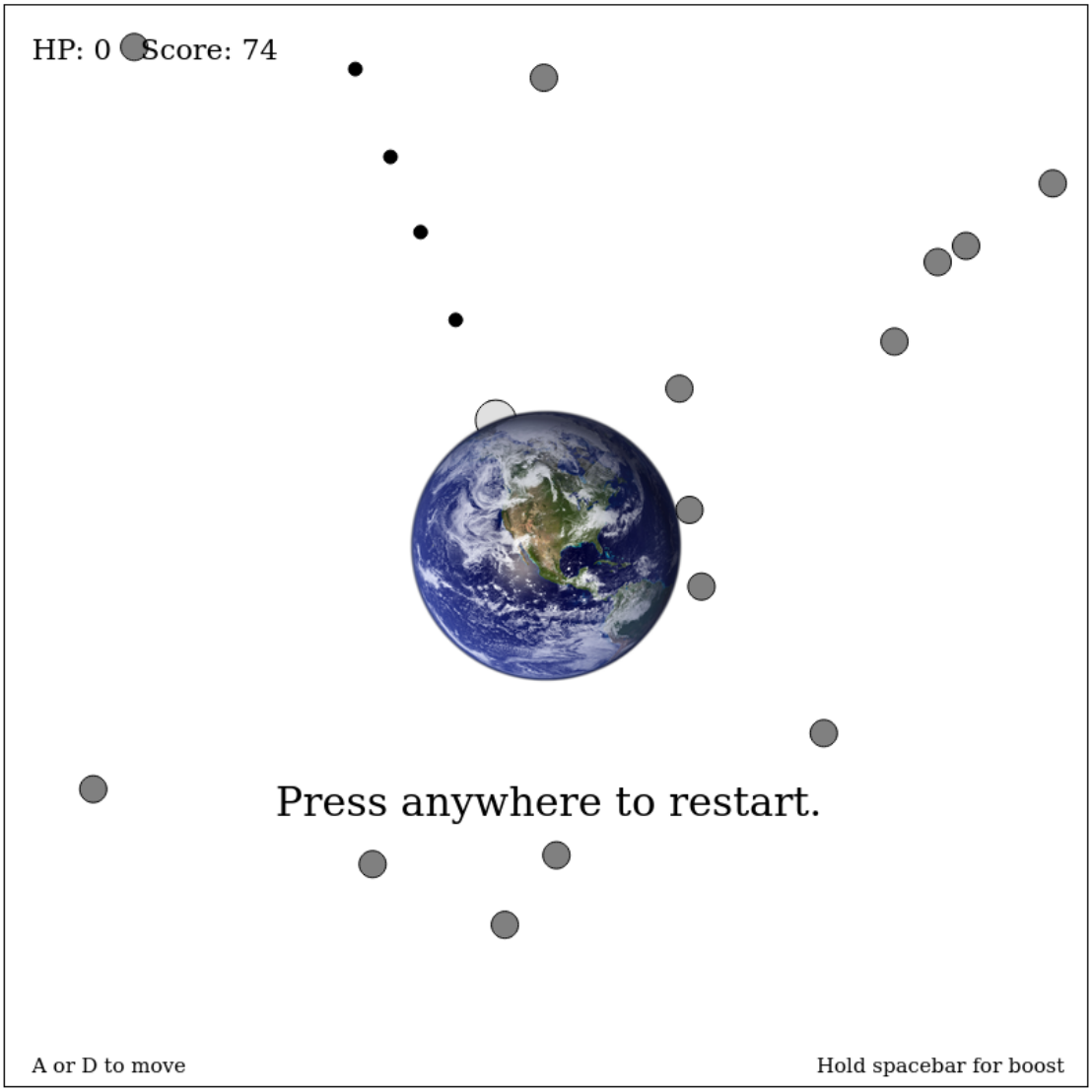
Lastly, I tried the game on my home computer but the speed at which the player moves is very different. The game was running way faster. I did not find any fix for this but my assumption is that the refresh rate of my monitor was higher than 60.

Analysis: (Reflect on what you have learned and how decisions made could impact users.)

Creating this project helped me improve my trigonometry (polar coordinates), and although minor, understand user controls. The former was the case because the asteroids move to the center of the screen. I obviously had to get some angle to get each asteroid's velocity to move towards the center of the canvas. The more I played with the math, I realized I can use polar coordinates from calculus 2.

I also learned how to optimize user controls. I said it wasn't that big of a deal but I think it was still a good change. When I was testing my game, I didn't realize holding down the spacebar can put a strain on my hand very quickly, especially for a game that requires precise movement. I inverted the key's effect so instead of constantly holding it down for most of the time, the player only needs to hold it for a short amount of time. Like I said it is a very minor change but if it was a problem to me, then it's definitely a problem to others.

Screen shots:


HP: 0 Score: 74
Press anywhere to restart.
A or D to move
Hold spacebar for boost