
Name: Rafael Amorim

Student ID: 100316549

FINAL PROJEC:

2d Survival Game

I. DESCRIPTION

I developed a third person survival game, in which the main goal is to survive the enemies attacks and kill as many of them as you can so you can get a higher final score.

The player starts the game with 30 health and 7 bullets in his gun magazine. Enemies start to appear around the map and the player have to avoid hitting them as if he does he loses health.

The player can move around to run away from the monster (using the W, A, S and D keys) and can also shoot them by aiming with the mouse cursor and left-clicking.

After shooting 7 bullets the gun has to be reloaded by pressing the key R. It takes 2.5 seconds to reload the gun and it can only be reloaded when there are no bullets left in the gun.

The main goal is to achieve a high score, which is calculated based on how many enemies you have killed and how long you have survived (each second survived gives around 1 point).

II. UNDERSTANDING THE CODE

- a. The html file only contains the canvas and another DIV with the instructions on how to play the gameç
- b. The game has 3 main components:

B1) Main Menu:

The main menu only contains a simple background and the play button, which I have created using photoshop as well as free images available in the internet. When you click on the canvas it takes you to the game;

B2) The Game:

All the assets and code;

B3) Game Over screen:

The game over screen is also a simple background with texts that I have made in photoshop. I used the canvas "fillText" to display the final score in the middle of the canvas;

When you click the canvas you start playing again;

To navigate in between these 3 components I created a variable called gameStatus which starts with the value 1 – triggering the function mainMenu(), that displays the menu.

When you click the Main Menu canvas, it changes the gameStatus to 2. This status runs the function updateCanvas(), that basically runs the entire game;

After you die in the game the gameStatus changes to 3, which runs the gameOver() function, that displays your final score and gives you the possibility to play again by clicking the canvas.

Most importantly, to keep checking the gameStatus, I included all of this inside the function startGame(), and called this function with the setInterval() method to keep checking it over and over again.

c. All the game code is run in the updateCanvas() function.

This is the most important function, because it runs almost all the other functions that I have wrote.

First this function clears the canvas and draws the background image. After that it adds 1 to the variable timeCount, that I use to control how the time passes.

I used several times if statements calculating the module between the time and different numbers, this way I managed to call some functions after a certain amount of time has passed.

Several different for loops were used to create the enemies, to create the first aid packs and to create the bullets. Inside these loops I used if statements to control many important features of the game, like:

- Checking the collision between the player and the boost pack and it receives a 10 hp boost, as long as it doesn't go over 30;
- Checking the collision between the bullets and the enemies, if they collide the enemy is deleted, the score is increased by 10 and a sound is played;
- Checking the collision between the player and the monsters, if they collide the player loses hp. If the hp is 0 or less, then the gameStatus changes to 3, which runs the gameOver();

List of features to implemented:

1) Character movement using the keyboard:

W = moveUp

A = moveLeft

S = moveDown

D = moveRight

R = reload

I achieved that by creating some attributes to the Object called "player" and setting them equal to false.

After I checked if the keys were pressed (which turns the attributes equal to true) and if the keys were released (which makes it false again). Then I used this (checking if these status were true) to move the player and reload the gun (simply assigning the value 7 back to the gunMagazine);

2) Aiming and shooting using the mouse:

To draw the bullets I used the same logic that I used to create the monsters. But instead of randomly generating them, I used the onmousedown event to trigger the creation of bullets.

To shoot the bullets towards the mouse direction, I used the onmousemove event to register the mouse x and y positions into 2 variables, than I used the Math.atan2() function, which takes a y and x positions and returns the angle between them in radians. Then I used this angle to draw the bullets.

3) Enemies movement:

I used the addEnemies() function to add new enemies to an Object called "enemiesList", I also defined some attributes that I used to set their position, size, speed of movement and image;

The createEnemies() function is used to randomly generate some of the monster's attributes, so they appear in the canvas in a random position and move with random velocities.

The moveEnemies() function adds the positions (x and y) of the monster as with the speed (x and y). This way the enemy moves around the canvas. To make the game become harder, I created a speedBoost variable, which multiplies the speed of the monsters as the time passes.

4) Collisions:

This is an important part of the game, as it is important to know when the player hits an enemy, when a bullet hits an enemy and also when the player hits the first aid kit.

As all the entities are basically a rectangle I wrote a function that checks if those shapes intersect. After, I checked for each of the combinations (for exemple player and enemy, player and first aid, etc).

When they collide, the collided variable turns true. Then I used an if statement to make something happen when collided is true for each of them. So, for instance, when the player collides with an enemy he loses health and a sound is played.

5) Score, healt, bullets:

I used the canvas fillText to draw all these attributes in the canvas. Then I simply pointed the content to the variables that I have created (playerScore, player.hp and gunMagazine);