THIRD SPRITE TASK

Description

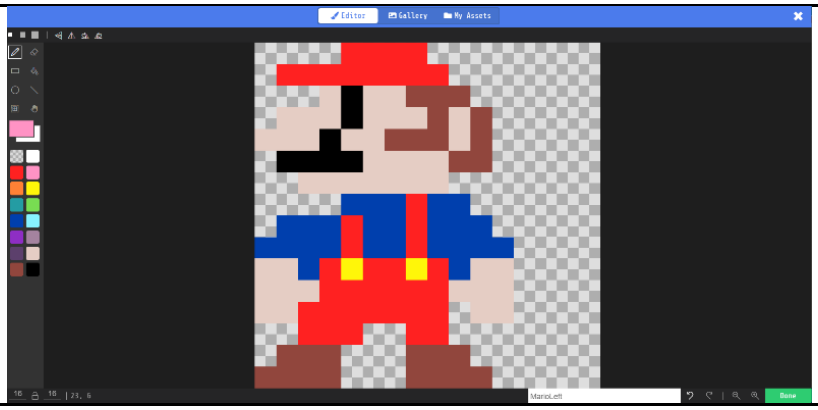
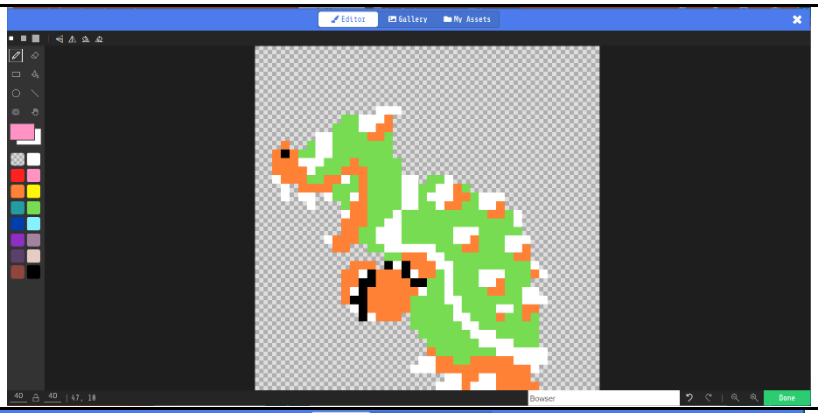
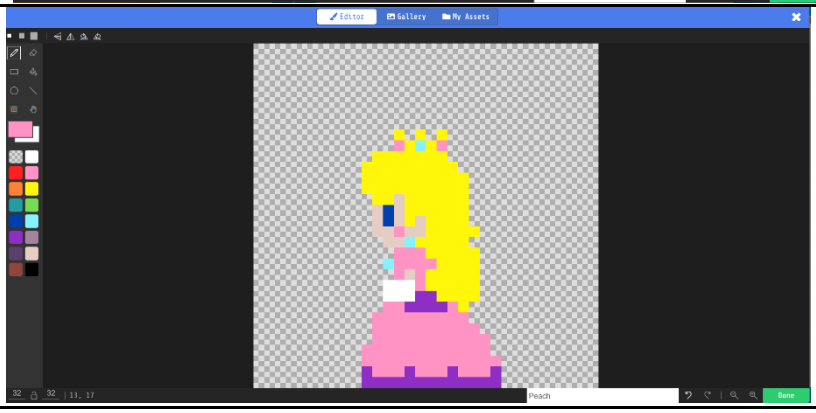
In this project, we will create what is known in video games as a cinematic. This involves creating a video sequence where the player does not have control and is used to advance the game's story or reinforce the main character's plot, introduce characters or enemies, set the atmosphere, etc. For this purpose, we will use the famous plumber Mario, who has just defeated Bowser and has come to rescue Princess Peach.

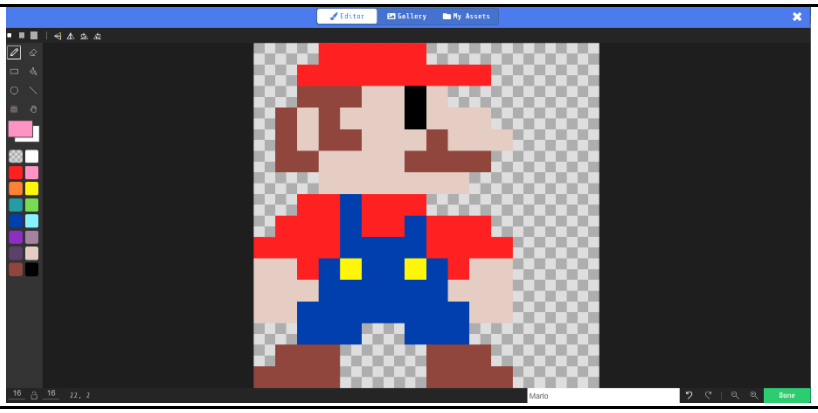
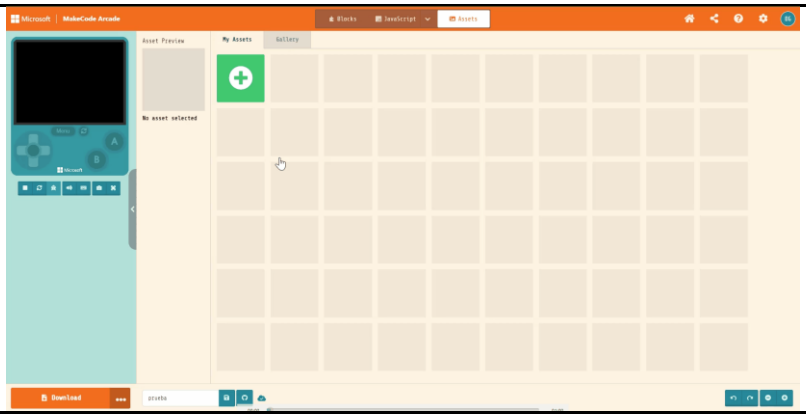
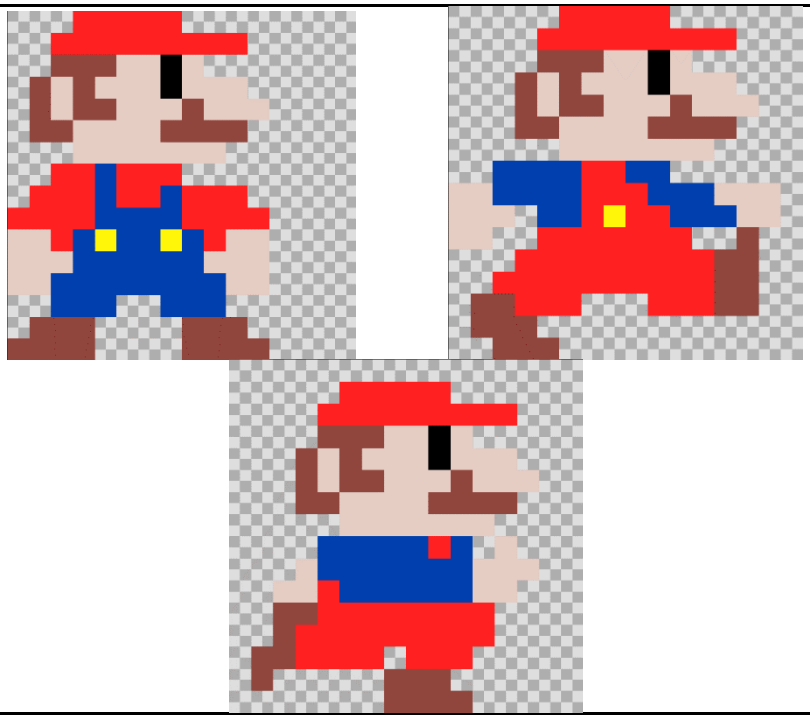
To do this, we will access MakeCode Arcade and perform the necessary operations.

Goals

- Draw a representative map with different elements to determine different scenarios
- Work on animations of different shapes.
- Create an animation for our character.
- Create an introduction to the cinematic.
- Perform the cinematic with the different programming elements.
- Provide context to the player through expressive game elements or displayed messages.

Game programming

Here we provide a link with part of the assets created and the initial programming: https://makecode.com/_ApoXXtEx2JrX Here is a summary of the assets created and the scenario creation.
ASSET CREATION
MAIN SPRITE CREATION
We recommend using a 16x16 px grid for the 'MarioLeft ' Sprite. 
ADDITIONAL SPRITE CREATION
We will create the Bowser Sprite using a 40x40 px grid. 
We will create the Peach Sprite using a 32x32 px grid 

We will create the Mario Sprite using a 16x16 px grid. Here, Mario is facing right, and we will attach the animation later.	
TILEMAP CREATION	
In the Assets, we will create the scenario with the different elements that you can find in the editor itself. It has dimensions of 40x8 px.	
ANIMATION SPRITE CREATION	
With our Mario Sprite, we will create the different Sprites that will compose the animations. All of them will be 16x16 px in size.	

MAIN PROGRAMMING

ON START GAME CREATION

To start, we will create an " on start" block and place inside it " set tile map ’’’leftMario’’ we declare it as the Player and position it using " set ‘leftMario’ position to ‘x’ 130 ‘y’ 75".

Next, we repeat the same process with " Bowser" we declare it as an and position it using " set ‘Bowser’ position to ‘x’ 80 ‘y’ 70".

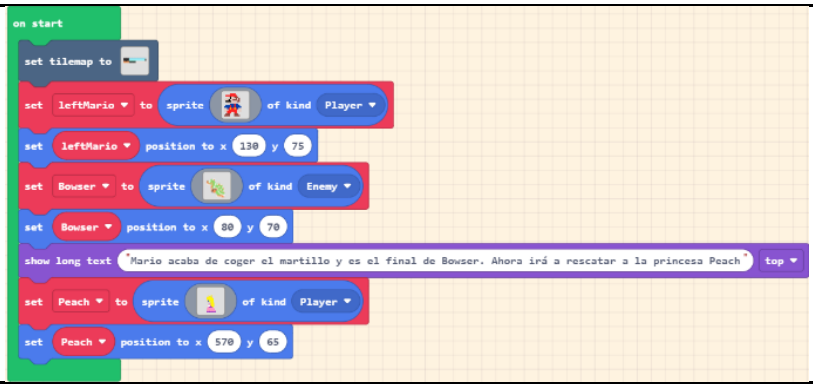
Finally, we do the same with Peach. Declare it as the Player and position it using " set ‘Peach position to ‘x’ 570 ‘y’ 65".

PLOT INCORPORATION

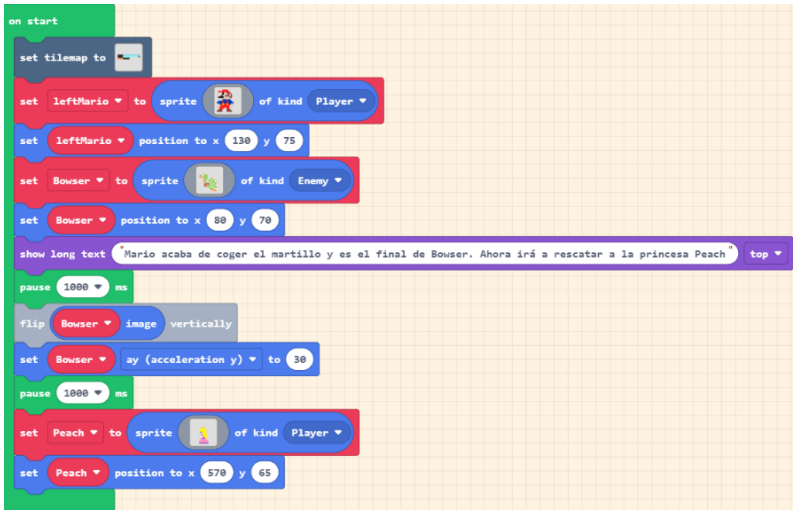
Let's create a small atmosphere to set the mood and give a sense of what is happening.

It is the end of the game, and Mario has defeated Bowser and saved Princess Peach. Mario is going to reunite with Peach, and she will thank him for saving her and her kingdom.

First, we will display a text to set the context.
To do this, go to ‘’ Game" and the " show long text" block. Replace "write the desired text" with the actual text you want to display. Place this block before creating Peach

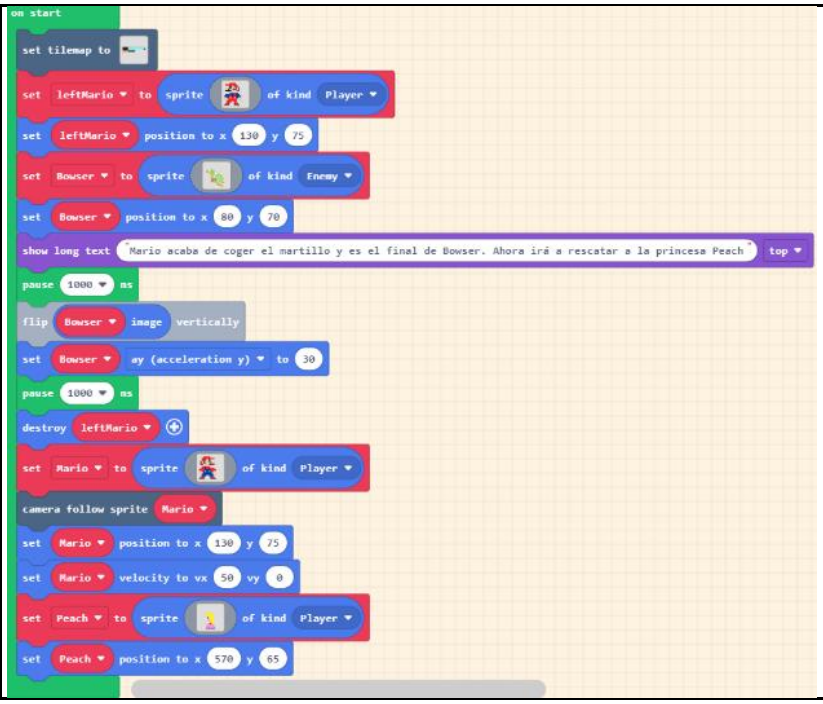


Afterwards, we will add a 1-second pause to create more suspense. Next, under the "Images" category, we select the block " flip ‘(Bowser) image' vertically." This will make the sprite flip vertically.
We give it acceleration (to make it fall) by using the block " set (Bowser) ‘ay (acceleration y) to 30".
Finally, we add another 1-second pause to allow it enough time to fall.
We place all of this between the previous text and the creation of Peach.



CINEMATIC MECHANIC

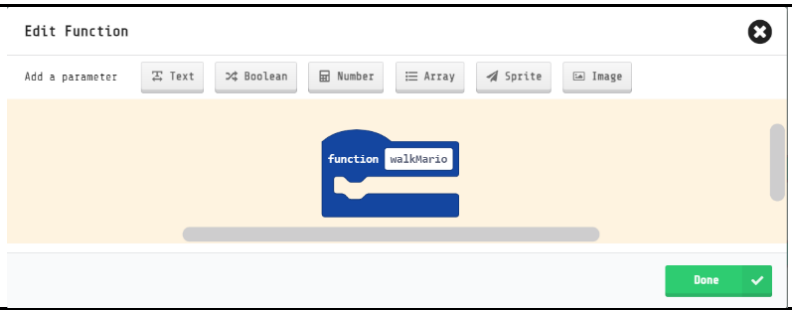
Let's begin with the cinematic sequence.
To do this, we will start by replacing the leftMario Sprite. This can be achieved by destroying the existing Sprite and placing the new one in the same position. Use the block " Sprites, destroy leftMario) " to destroy the existing Sprite. Then, use the block "set ‘Mario’ to (Sprite ‘Mario’ of Kind Player)" to set the new Sprite. Place it in its position, give it velocity, and make the camera follow it camera follow sprite (Mario).



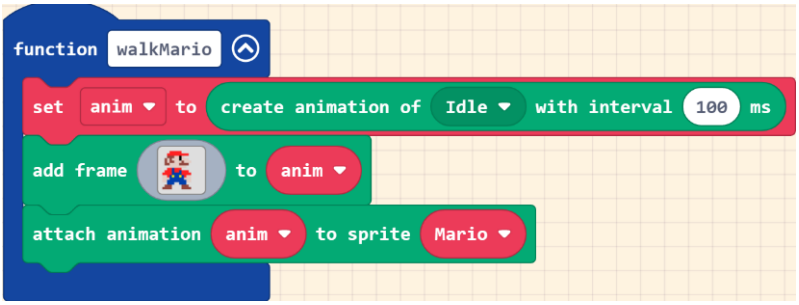
ANIMATION CREATION

Let's create a function that will handle the animation of Mario when he is walking or standing still.

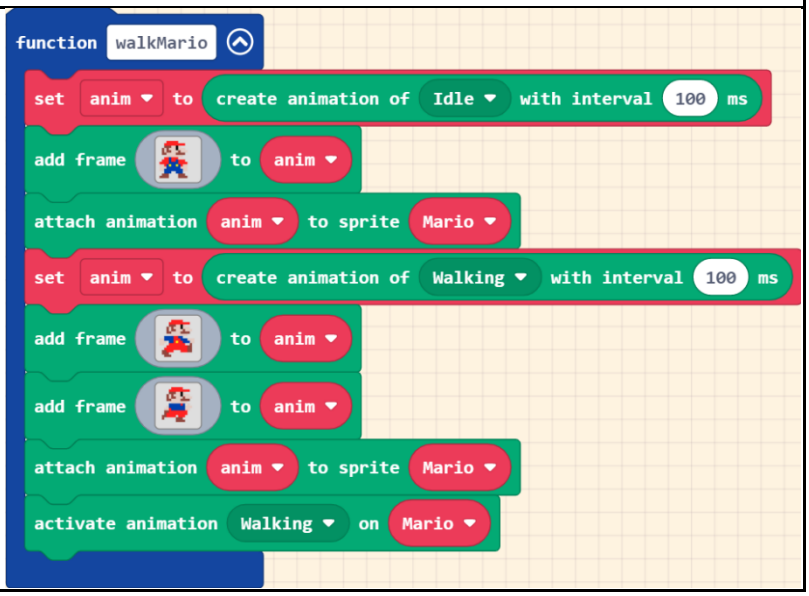
Functions walkMario

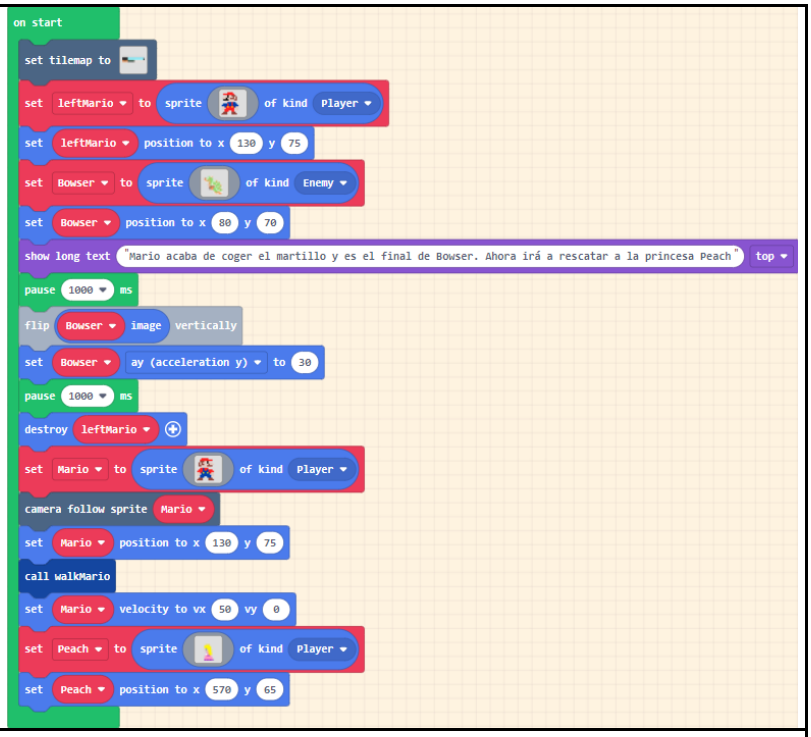
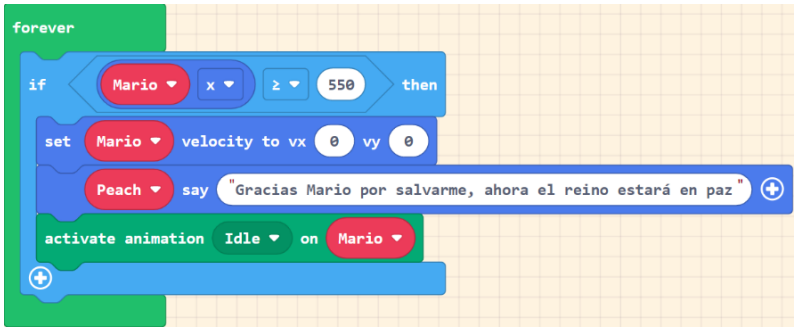


Let's start with the idle animation for Mario.
First, go to the "Animation" category.
Use the block " set anim to créate animation of 'Idle' with interval 100 ms".
Next, add the frame of Mario in the idle position to the animation.
Finally, assign this animation to the .



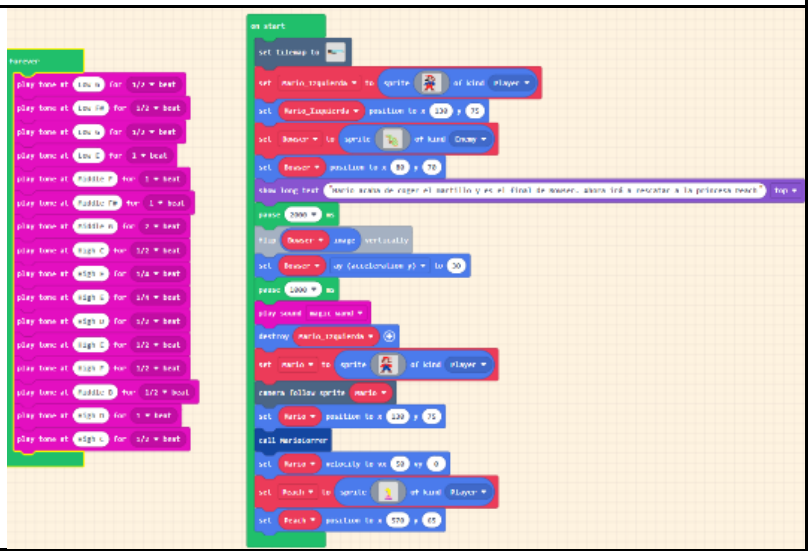
Next, let's continue with the walking animation for Mario. It follows the same steps as before, but we will incorporate the frames of Mario walking.
Again, go to the "Animation" category.
Use the block " Animation set anim to create animation of 'Walking' with Interval 100 ms".
Add the frames of Mario walking to the animation.
Assign this animation to the Mario Sprite.
Finally, activate animation 'Walking' on (Mario).



Finally, we call the "walkMario" function in the " on start" block to initiate Mario´s movement while walking.	
FINAL CINEMATIC CREATION	
Now let's create the ending of the cinematic sequence. We will create a " forever " block. Inside the loop, we will use an "if" condition: if the x- position of Mario is higher than or equal to 550, we will execute the following code. Set (Mario) velocity to vx 0 vy 0 in both the x and y directions to make him stop moving. Use the block " Peach say 'Whatever we want'" to make Peach say a specific message. Replace "Whatever we want" with the desired dialogue. activate animation 'Idle' on (Mario .	

ADDITIONAL DETAILS

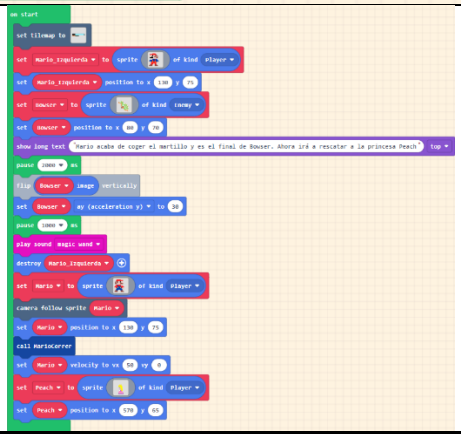
To add the final touches, let's incorporate some sound effects and background music.



MELODIES CREATION

To ensure that the background music plays throughout the cinematic sequence, we will create a "forever loop ’ and place all the music inside it.
For the melodies, go to the " Music" category and use the " play tone at (Middle C) for ½ beat" block. Replace "Middle C" with the desired musical note and adjust the duration of the note accordingly. Add as many notes as needed to create your melody

Finally, let's add a sound effect when Bowser is defeated.
Use the " Music " block and the " play sound ‘magic wand'" block to play the sound effect. Place this block after the last pause to give it the effect of Bowser's defeat.



Thanks to this programming, we have learned how to create animations and cinematic sequences. This makes our games visually more appealing and interesting to play.

Glossary

Functions: It is a subroutine that consists of a set of instructions and can be called from the main program.

Walls: Objects or areas where the various elements of the game cannot pass through.

Camera: An object within a game scene that serves as the player's viewpoint in relation to the game.

Narrative: Part of a video game that serves to build a story.

Effect: Something applied to the environment, objects, characters, and other elements to convey realism or a certain sensation within the game.

Music: Combination of sounds and silences that create a rhythm.

Image: Visual element displayed on a screen that represents something (a landscape, people, etc.).

Acceleration: The rate at which an object's velocity changes over time.

Velocity: A physical quantity that relates an object's displacement to the change in time.