Unity Project: Make a game like The Legend of Zelda

By: Mathias, Thu, Ysihak

Press Play


Level:
1

Level 1: The Startup

Our tutorial was from a youtube video made over five years ago, which meant that our video and our software are completely different. Unity compared to itself five years ago has many changes that make following the videos instructions difficult.

Unity : A software to make the game, import C# code and animate.

Visual Studio : A software used to code and then saved to a file in a Unity project.

*GFX : Our art folder, inside contains our pixelated sprites, backgrounds, and tiles for our project.

Level 1: Baby Steps

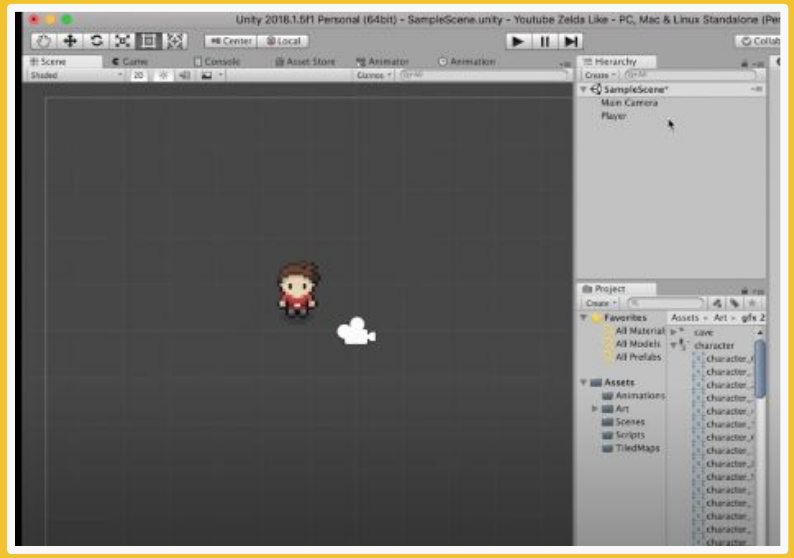
Our first step was to download the software that gives birth to our game, unity. After using a VM to download Unity because CPSD blocked it, we then had to use the same VM to download the art assets (the sprites, the overworld and objects). In our VM we used DuckDuckGo. After, we made a file in a VM of all the art assets and other blocked downloaded softwares from CPSD, we shared that file to the rest of the group in order to start the project.

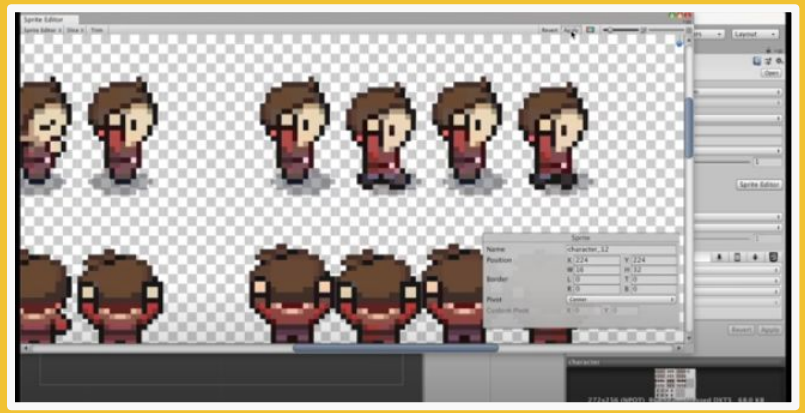
DuckDuckGo

VirtualBox

Level 1:
The Birth

Once our downloads finished, it was now time to set up our game in Unity. In Unity, we had to select what game we would be making, a 2D game. All of us had to make folders and import many other assets into our game. We also had to slice some of our art images in order to create our sprites. Once we finished naming our images and folder, it was now time for us to make our images come to life.



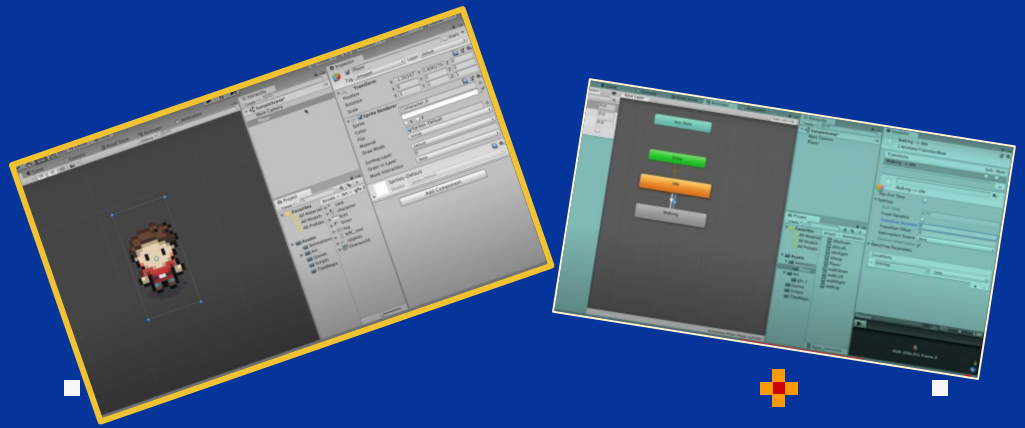



Level:
2

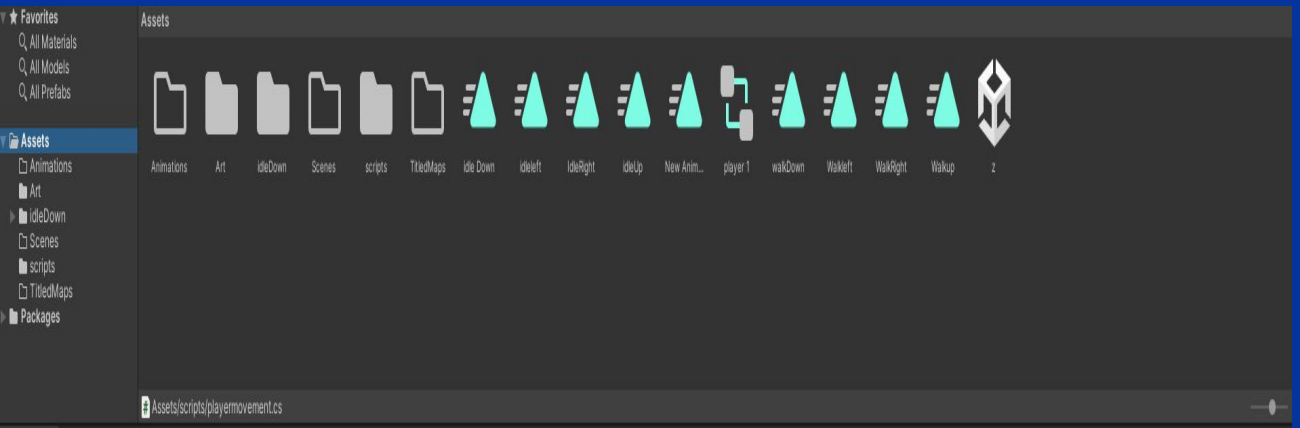
Level 2:
Movement & Animation

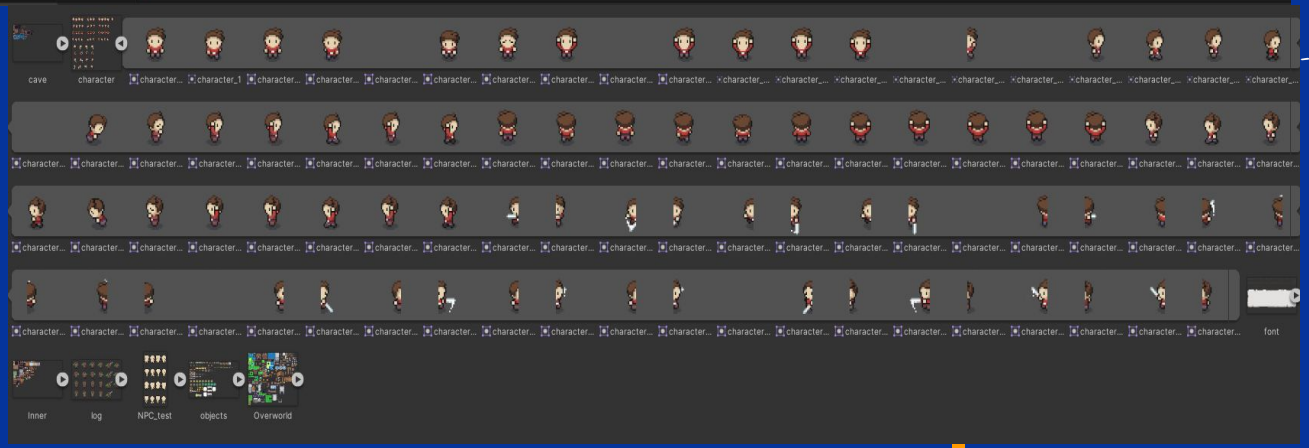
- When creating this 2D project, the first step would be to get the character to move.

- Next step would be setting up animations to build the setting of the character.



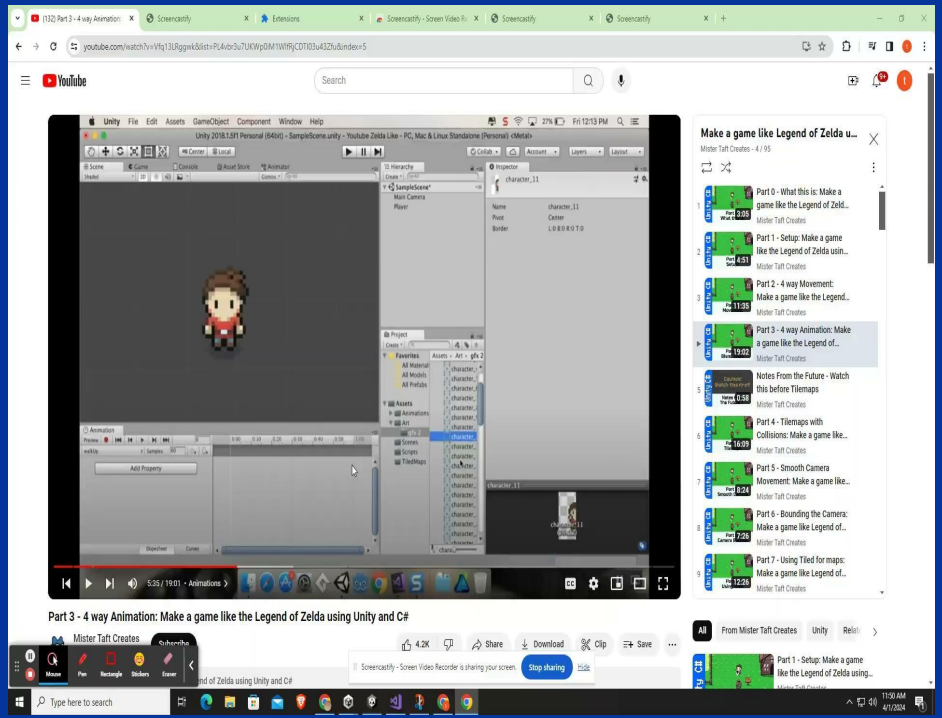
LEVEL 2 : Animations and characters





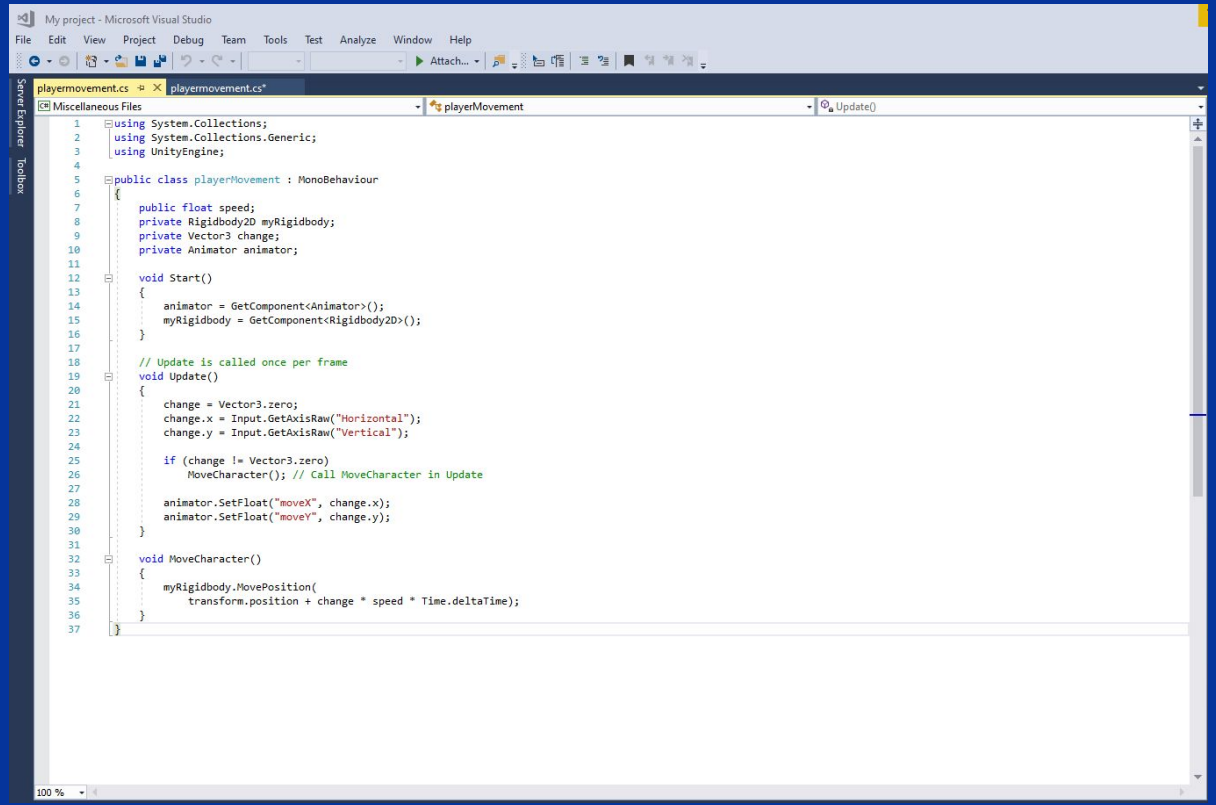
- One of the most important files connected to assets is annmations.
- Each of the 'ide/walk' files that you s correspond to the movement of the character when connected to one of these files.
- 'Idle' only corresponds to what happen when the character is standing still, 'Walk' is for when the character is moving.

Level 2 :Picking characters for specified movements



- This video will be an example of characters being add to one of the files mentioned above; 'Walk Up'.
- Provide a walkthrough on what happens in the video.
- example of what you would have to do for the character to ' Walk Up'.

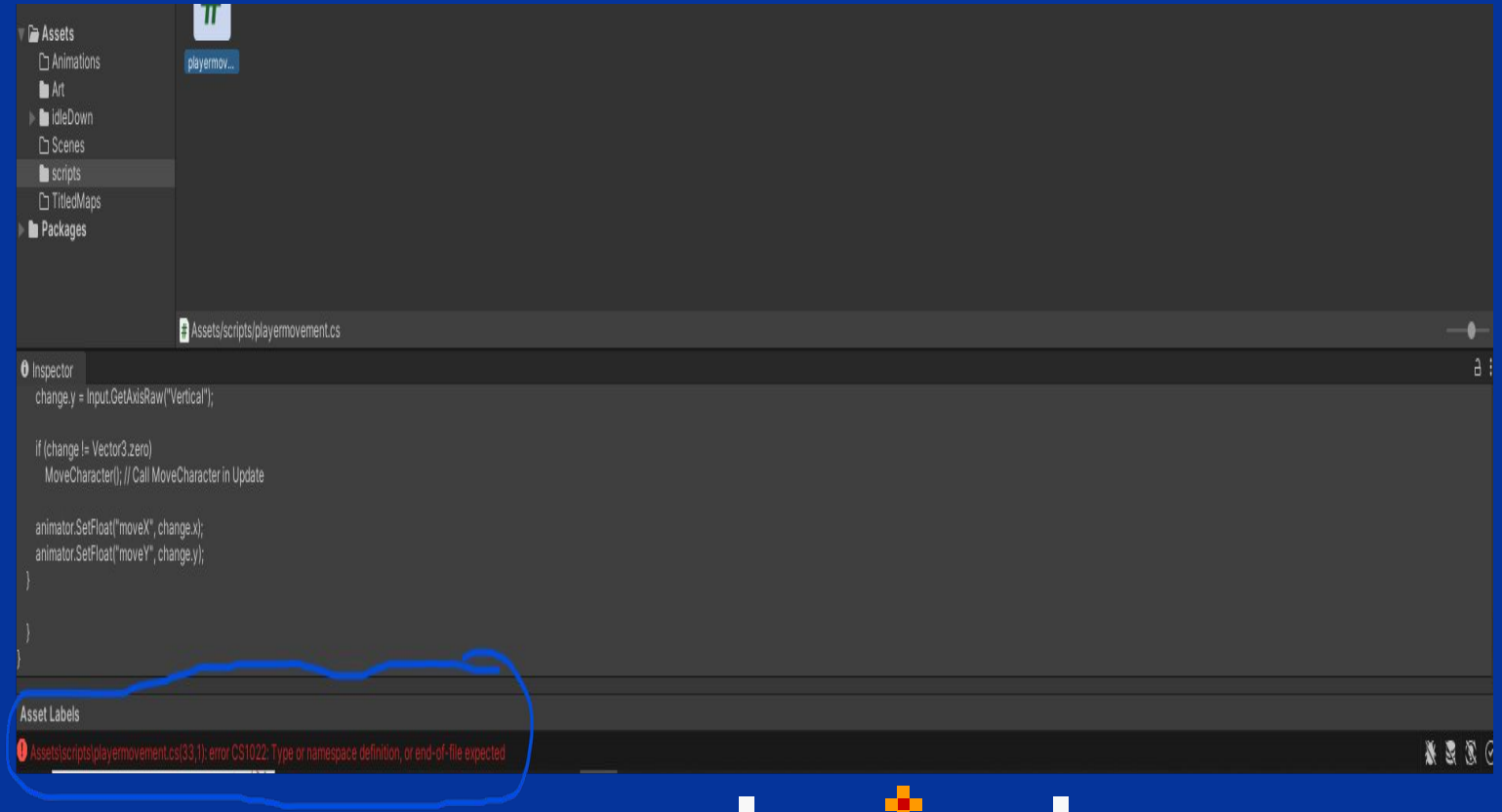
Level 2: Player Movement coding




```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class playerMovement : MonoBehaviour
{
    public float speed;
    private Rigidbody2D myRigidbody;
    private Vector3 change;
    private Animator animator;

    void Start()
    {
        animator = GetComponent<Animator>();
        myRigidbody = GetComponent<Rigidbody2D>();
    }

    // Update is called once per frame
    void Update()
    {
        change = Vector3.zero;
        change.x = Input.GetAxisRaw("Horizontal");
        change.y = Input.GetAxisRaw("Vertical");

        if (change != Vector3.zero)
            MoveCharacter(); // Call MoveCharacter in Update

        animator.SetFloat("moveX", change.x);
        animator.SetFloat("moveY", change.y);
    }

    void MoveCharacter()
    {
        myRigidbody.MovePosition(
            transform.position + change * speed * Time.deltaTime);
    }
}
```

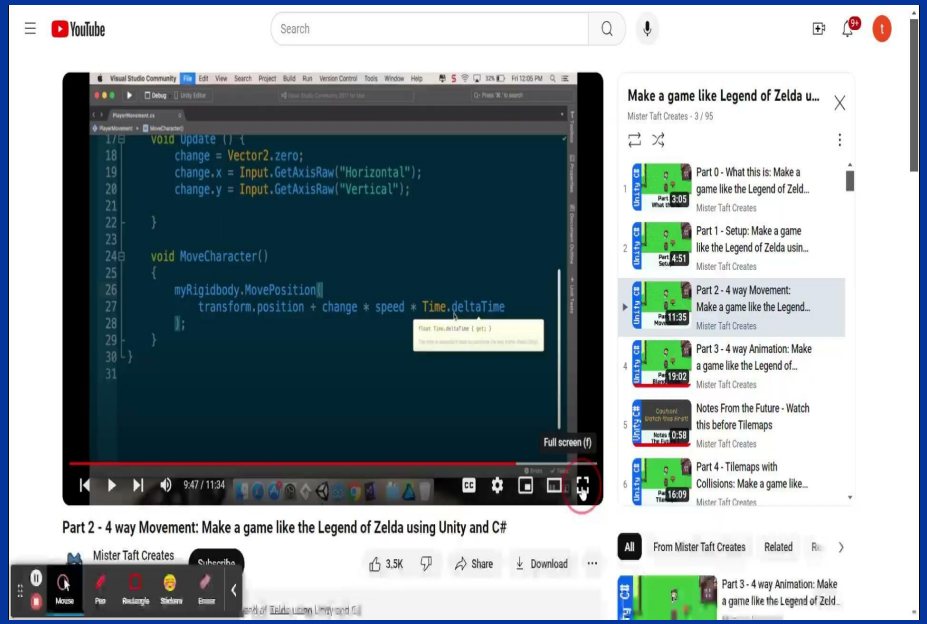
- Player movement is used throught our project.
- After adding character corresponding to each of those files, you then have add code to the player movement so that movement is possible.
- Without this aspect being there, our previous progress wouldn't/work run on unity.
- Address bugs.

Level 2: Unity error



- Tried to run player movement and didn't work.
- Error on unity can be seen below when trying to run code.

Level 2: player movement errors/character moving



- Way in which you can identify a unity error
- Example of a unity error.
- Character moving and advancing onto level 3.


Level:
3

Level 3: Title Card

Goal: A title card will pop up whenever the character move to a new room

Demo result:

Level 3: Title Card

Instruction:

Create a Canvas

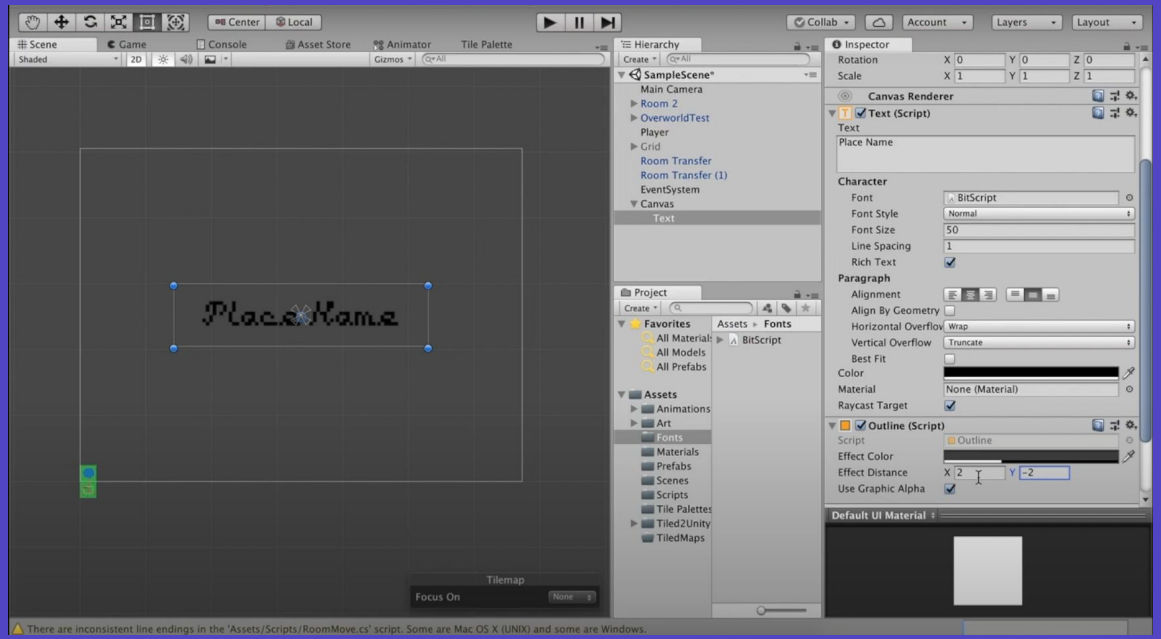
Inside the Canvas, create Text

Edit the text

Font

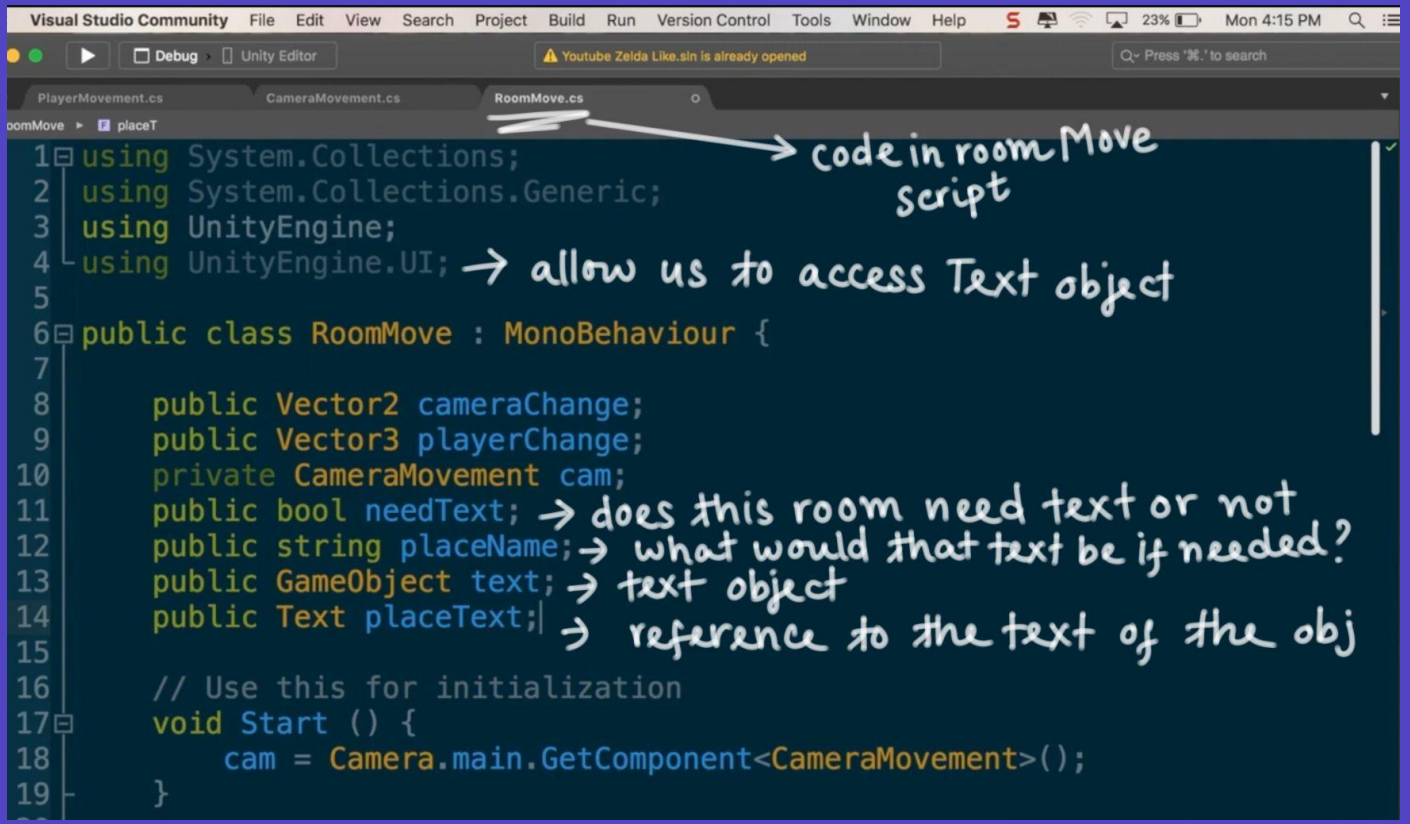
Size

Alignment



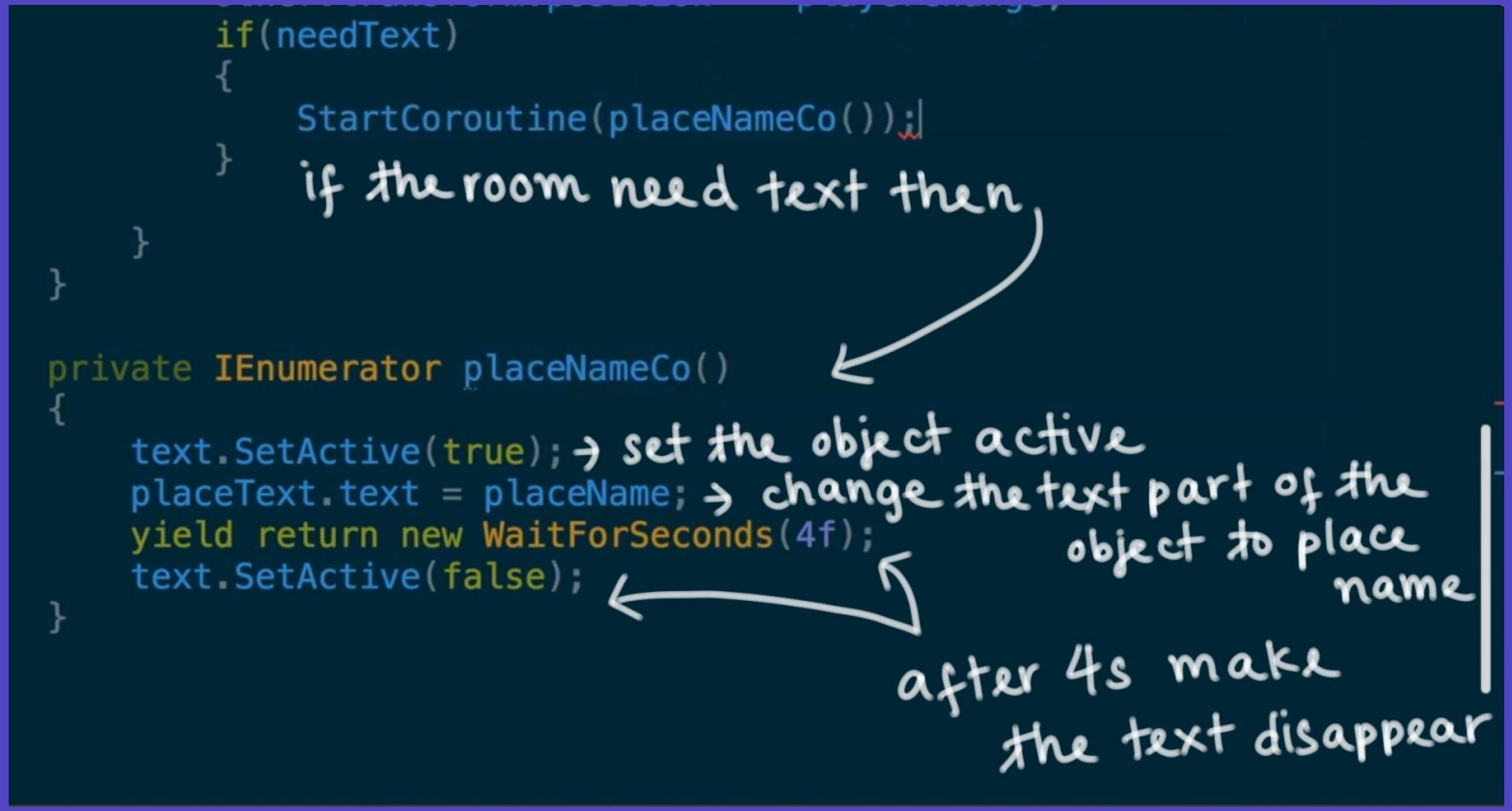
Level 3: Title Card

Visual Studio



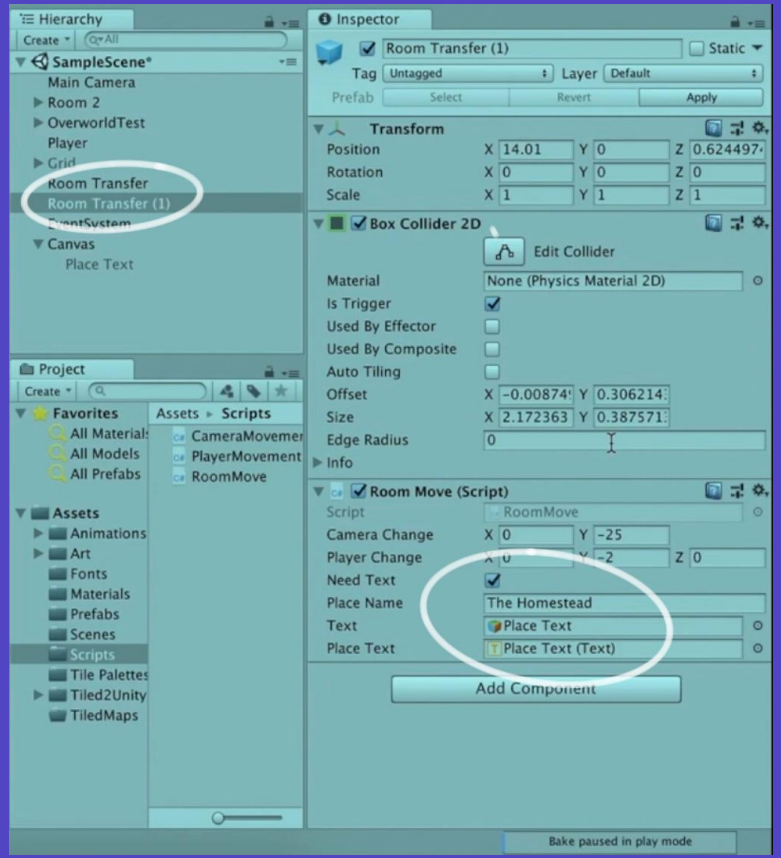
Level 3: Title Card

Visual Studio



Level 3: Title Card

Unity



Level 3: Dialog box

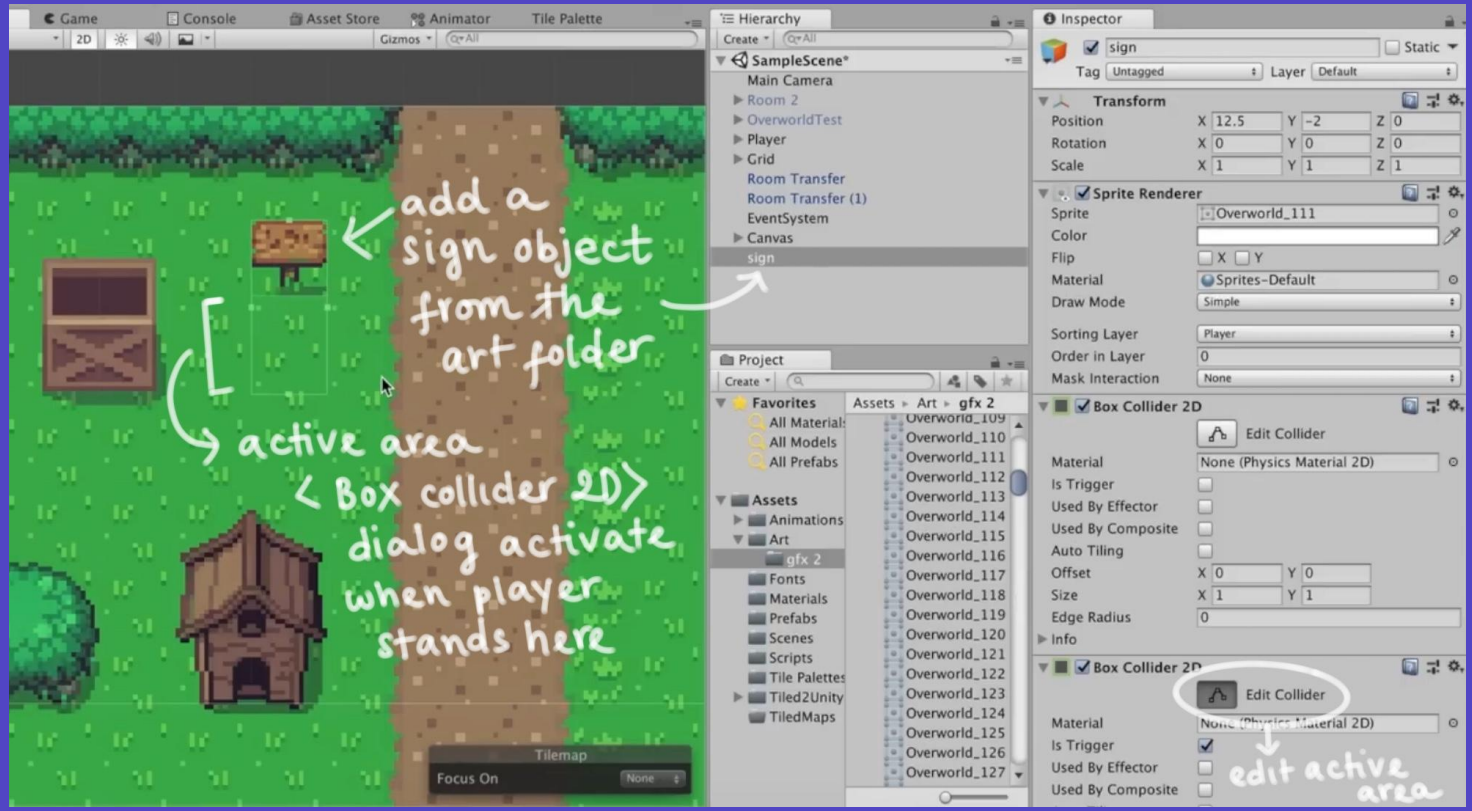
Goal:

A dialog box will pop up whenever the character is standing in front of a sign

Demo result:

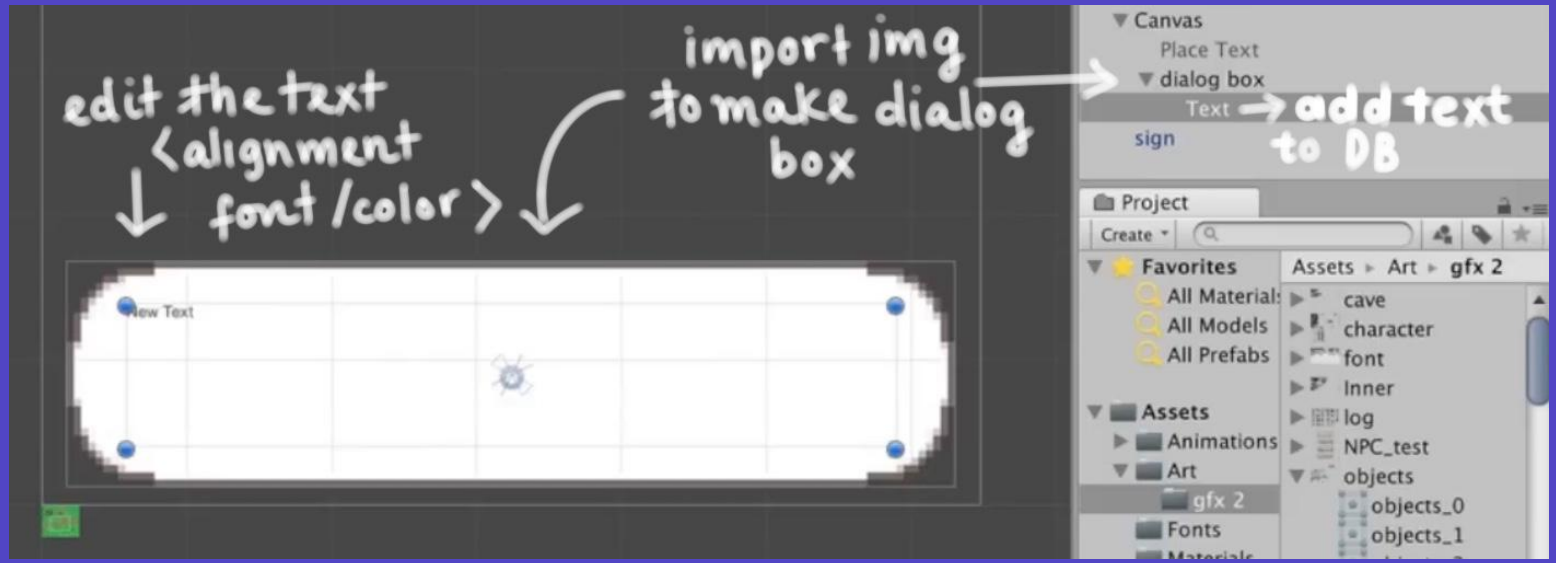
Level 3: Dialog box

Instruction



Level 3: Dialog box

Instruction



Level 3: Dialog box

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;          → import UI file
5
6  public class Sign : MonoBehaviour {
7
8      public GameObject dialogBox;   → reference to the DB
9      public Text dialogText;        → reference to the text in DB
10     public string dialog;          → sentence dialog text will change into
11     public bool dialogActive;      → determine whether DB show or not
12
```

Objects Layout

Level 3: Dialog box

```
private void OnTriggerEnter2D(Collider2D other)
{
    if(other.CompareTag("Player"))
    {
        playerInRange = true;
    }
}

private void OnTriggerExit2D(Collider2D other)
{
    if(other.CompareTag("Player"))
    {
        playerInRange = false;
    }
}
```

→ check if the object is the player

→ confirm that the player is in range

→ confirm player is out of range

Objectsally

Level 3: Dialog box

Visual Studio

```
18      // Update is called once per frame
19      void Update () {
20          if(Input.GetKeyDown(KeyCode.Space) && playerInRange)
21          {
22              if(dialogBox.activeInHierarchy)
23              {
24                  dialogBox.SetActive(false);
25              }else{
26                  dialogBox.SetActive(true);
27                  dialogText.text = dialog;
28              }
29          }
```

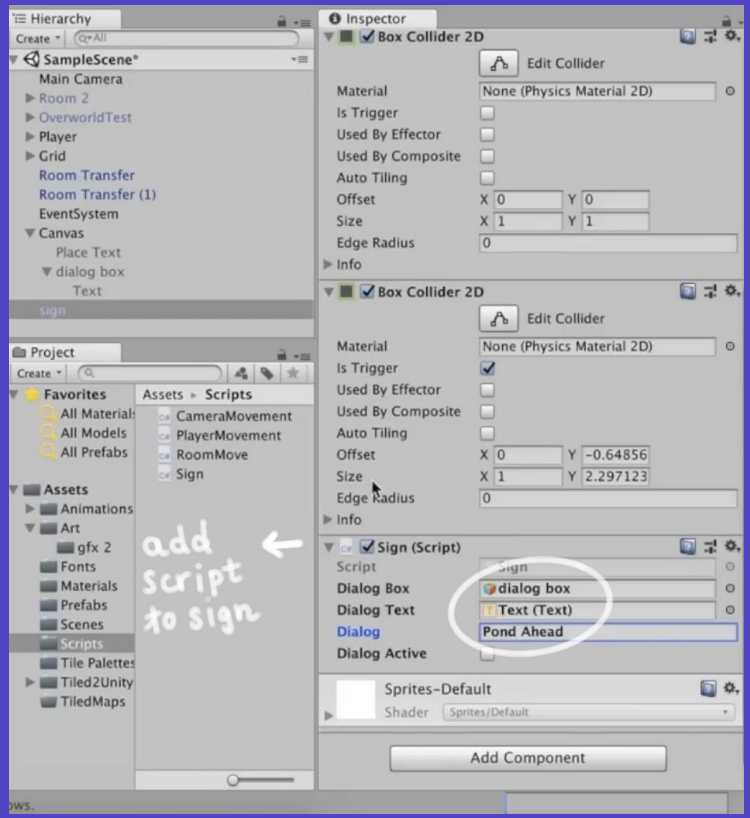
if DB is alr activate ⇛ deactivate <space bar> → DB gone

if space bar is pressed + player is in range

DB appear sentence replace DB Text

Level 3: Dialog box

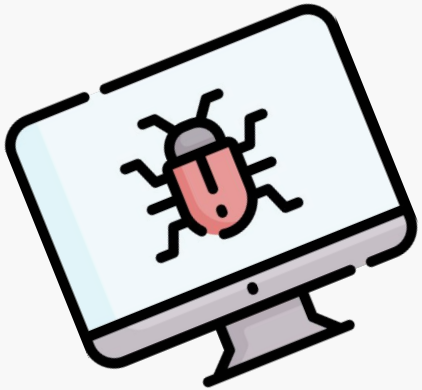
Unity

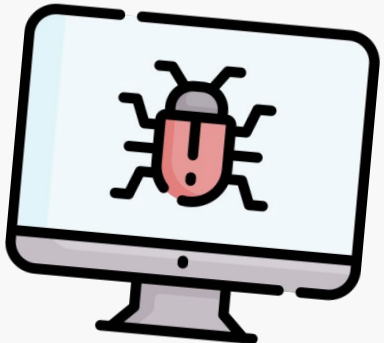


Level: 4

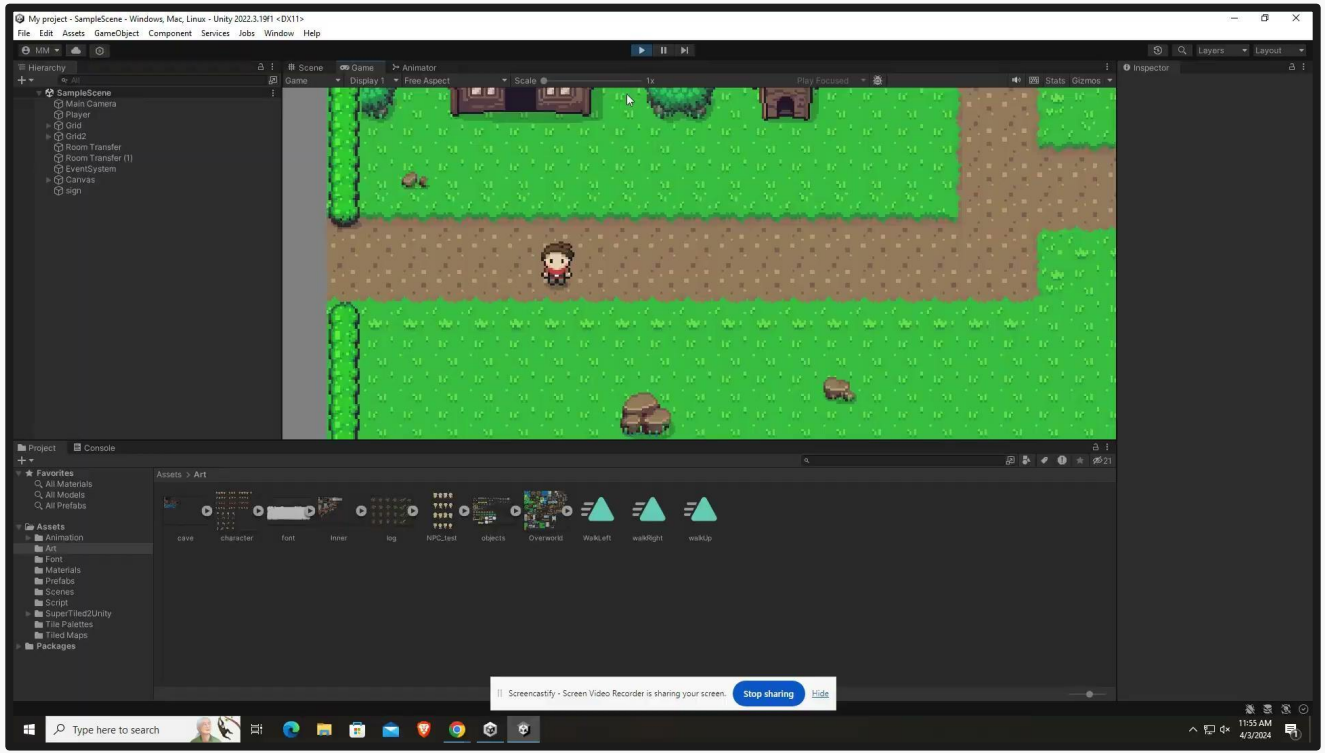
Level 4: Bugs and Errors

Definitely our hardest obstacle in making the project was facin challenges when it come to random bugs and errors in our game. It could be a code error, a tilemap error or even a small mistake we made that would change our game forever. Sadly, some of our projects still have these errors that make our game unplayable as of right now.





Level 4: Complex Bug Ex:



Level 4: Overcoming Challenges

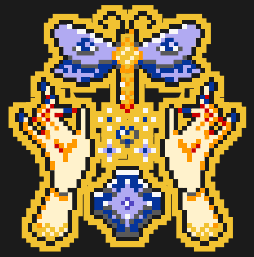
Excluding the errors that may have completely messed up someones game, we had many bugs that were difficult to fix, yet as a team we made sure to help each other with the problems that we were facing in order to have our projects stay on the right track. At one point that even meant changing and doing something different than what the tutorial was teaching us. We improvised many times which helped us use smart and critical thinking skills in order to avoid minor problems that may turn out big.

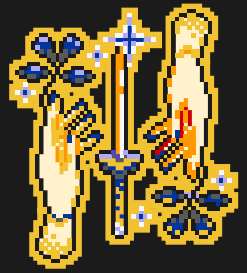
Level: X

Level X: Potential

Sadly the amount of time needed to really complete a majority of the game is longer than we had. Of course we knew that so our project was mainly about what takeaways we could gain from learning on how to make a game. Although we didn't get that far, the potential for the project surprised us much as we didnt even know how far it would come. Since we couldn't finish the project ourselves, these are videos and pictures of what the project should look like...

The End





ANY QUESTIONS?