Name	Mark Eldridge	Team	Green Sky Games	TL	3	Date	4/13/25	Time	

Fill in the underlined areas (and the boxes above), now but don't write on the remainder of this form.

Contribution: Briefly describe what your feature(s) is/are:

/10

In charge of Main Character Selection, Main Character Animation, Main Character Health Bar and Sanity Bar (adjusting them based on events)

Walk me through your Gantt chart. How long did this take? How long did you estimate it would take? What did you learn about your skill as an estimator?

I was able to complete my tasks below the time frame I gave myself.
I learned that it is best to get the basics tasks out of the way before moving onto the more complex tasks that integrate with everyone elses tasks.

Run your game and point out places where your code is called and run. (I will cycle through asking you this question and the next one until you either run out of interesting things to talk about or it is clear that you have made an above average contribution.)

Show the C++/C# code that was run. Walk me through the methods called from the time it enters your section of code.

Technical:

Walk me through your test plan. Give an example where a test case later found a bug in your code by things a teammate added later. (Or explain why you chose a test case specifically because you wanted to ensure that a teammate would know if they broke your code.) /4

In the Character Selector scene I have a boundary test that shuffles through the Character, ensuring that you can see all the character available to pick from. This broke multiple times due to prefabs edits and script edits.

Pick a Prefab you have created that is documented well in a separate readme file. (I will point to several places in your code documentation and ask) What question where you trying to answer here? Who do you anticipate would be asking that question? What other questions might this person need the answers to? /3

Prefab Name: Riding Hood

Show me a class in your code where there could be either static or dynamic binding. Write some mock code on this paper showing how you would set the static type and dynamic type of a variable.

Super Class: Character

Sub Class: PlayerHealth & EnemyHealth /3

Virtual Function: TakeDamage(int health)

Choose a dynamically bound method. What method gets called now?

Change the dynamic type. What method gets called now?

Pick a statically bound method. Which one would be called in each of the two previous cases?
Show me an example of reuse in your code where you violate copyright law.
How does it violate copyright? Never gave credit to the following website.
What did you have to do to integrate it with the code you wrote? What are the legal implications if you market your code with the re-used portion? Use fair use argue that you can use this anyway. /4

I made multiple sprite sheets of Red Riding Hood and integrated the blue print designed by bluecarrot16, JaidynReiman ,Benjamin K. Smith (BenCreating), Evert, Eliza Wyatt (ElizaWy), TheraHedwig, MuffinElZangano, Durrani, Johannes Sjölund (wulax), Stephen Challener (Redshrike) into the PlayerControl.cs script. I can be sued. I only used some of the animations provided in the blue print which is not enough to say I stole their entire blue print.

4. One big or two small, well-chosen patterns.
Small Patterns = {Singleton, Private Class Data}
Which patterns did you choose?
1. Singleton Pattern /4

2. Private Class Data

Why did you choose each pattern? (Justify your use of it).
We wanted to ensure there was only one global Audio Instance. Having more than that could lead to inconsistence behavior in the Game.

By making _selectedOption private, we have control how this variable is accessed in modified. This protects the CharacterManager from unintended modifications, ensuring integrity. We have a read function getSelectedOption() which helps to access the variable in different scripts.

Draw the class diagram for your pattern(s).

See Below

Would something else have worked as well or better than this pattern? When would be a bad time to use this pattern?

The only time I found Singleton being a problem is when I tried testing. Singleton caused tight coupling. In large complex projects where testing is crucial, maybe using a Factory method would work better for each scene.

```
public class Character : MonoBehaviour
{
    public virtual void TakeDamage(int amount)
    {
        Debug.Log("Character took " + amount + " damage.");
    }

    public void Heal(int amount)
    {
        Debug.Log("Character healed for " + amount + " health.");
    }
}

public class PlayerHealth : Character
{
    public int currentHealth = 10;
    public int maxHealth = 10;

    // ... (rest of your PlayerHealth code)

    public override void TakeDamage(int amount)
    {
        currentHealth = Mathf.Clamp(currentHealth - amount, 0, maxHealth);
        Debug.Log("Player took " + amount + " damage. Current health: " + currentHealth);
        // ... (rest of your damage handling logic)
    }
}

public class EnemyHealth : Character
{
    public override void TakeDamage(int amount)
    {
        Debug.Log("Enemy took " + amount + " damage. Enemy reacting differently.");
        // ... (enemy-specific damage logic)
    }
}

Character characterReference;
characterReference = GetComponent<PlayerHealth>();

characterReference = GetComponent<EnemyHealth>();
characterReference.TakeDamage(5);
```

Universal LPC Spritesheet Generator

Free/Libre pixel art from the Liberated Pixel Cup

You must credit the authors of this artwork.

▼Controls

Download: Spritesheet single image (PNG) | Credits (text) | Credits (CSV)

Download pack: Spritesheet split animation images, credits, JSON (ZIP)

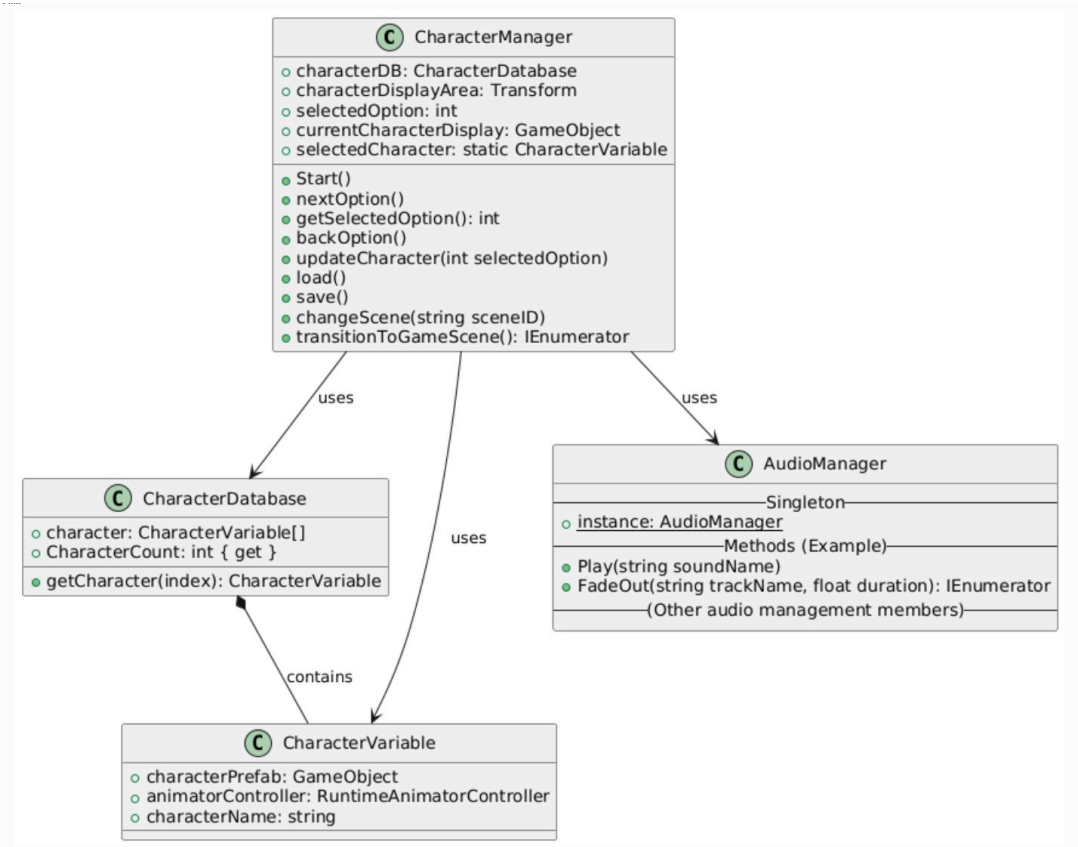
Import/Export: Export to Clipboard (JSON) | Import from Clipboard (JSON)

Edit: Reset all | Replace Mask (pink) | ☑ Match body color

View: Collapse All | Expand Selected | ☐ Compact Display

Search: Search

Preview Walk 1-2-3-4-5-6-7-8


CharacterManager
characterDB: CharacterDatabase
characterDisplayArea: Transform
selectedOption: int
currentCharacterDisplay: GameObject
selectedCharacter: static CharacterVariable
Start()
nextOption()
getSelectedOption(): int
backOption()
updateCharacter(int selectedOption)
load()
save()
changeScene(string sceneID)
transitionToGameScene(): IEnumerator
uses
uses
uses
CharacterDatabase
character: CharacterVariable[]
CharacterCount: int { get }
getCharacter(index): CharacterVariable
contains
AudioManager
Singleton
instance: AudioManager
Methods (Example)
Play(string soundName)
FadeOut(string trackName, float duration): IEnumerator
(Other audio management members)
CharacterVariable
characterPrefab: GameObject
animatorController: RuntimeAnimatorController
characterName: string