Planning and Documentation for Celestial

Pseudocode for Player Movement

```
Y direction = gravity
If “A” key is pressed
{
        X direction = -1
}
Else if “D” key is pressed
{
        X direction = 1
}
Else
{
        If player is grounded
        {
                X direction = 0
        }
}
If player is grounded
{
        Y direction = gravity
        Move(x direction * speed, 0)
}
Else if player is jumping
{
        If jump height >= 0 and jump height < maximum jump height
        {
                Jump height += jump height per frame
                Move(x direction * speed, jump height per frame)
```

```
        }
        Else
        {
        Player is falling
        }
}
Else if player is falling
{
        Y direction *= 1.005
        Move(x direction * speed, y direction)
}
```

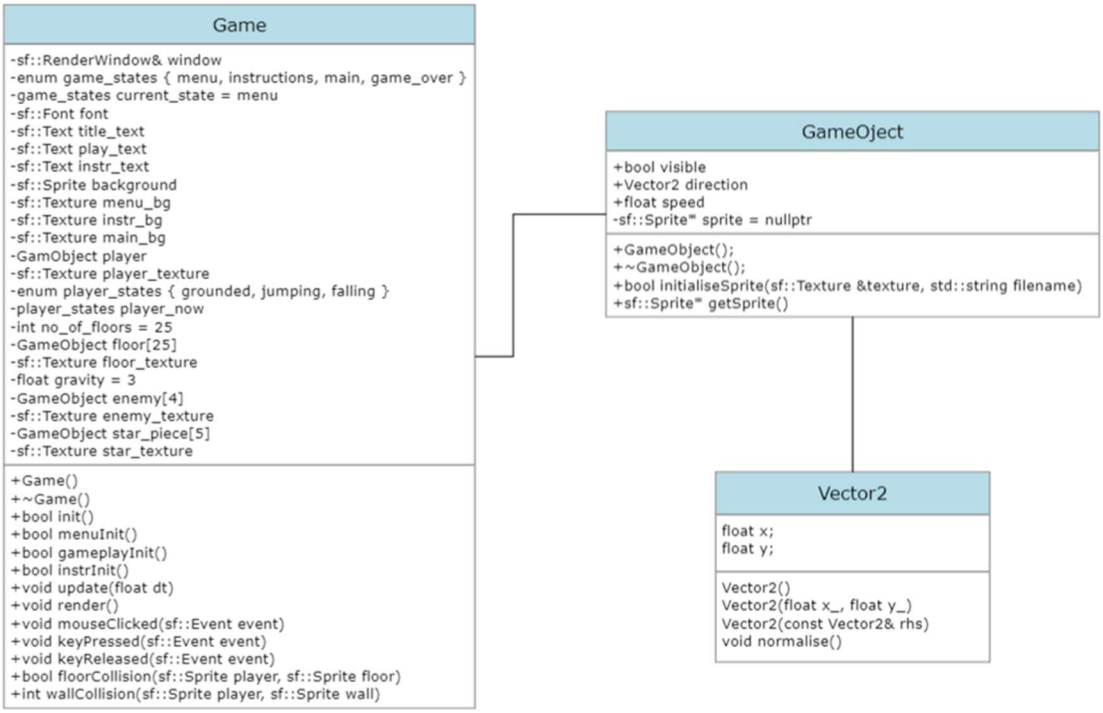
As can be seen the “A” and “D” keys dictate the x direction the player moves in. If the player is falling, their downwards force increases exponentially. They also continue in the x direction they were going, until they land. The combination of the exponential down force and the continued forward force creates a curved trajectory.

Pseudocode for Platform Collision

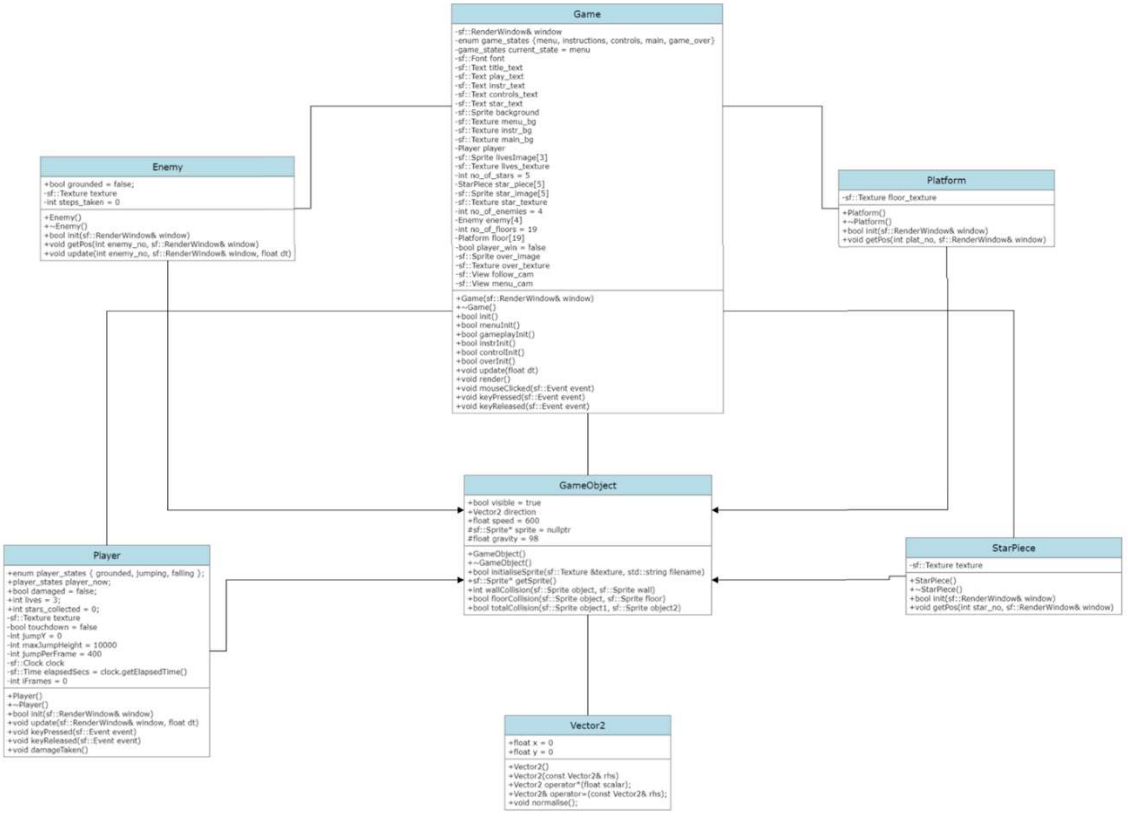
```
If
(player x position + player width >= platform x position &&
player x position <= floor x position + floor width &&
player y position + player height >= floor y position &&
player y position <= floor y position + floor height)
{
        Return true
}
Return false
```

UML Class Diagrams:

Before making the game:



After making the game:



As can be seen, the final product was a lot more complex than expected. This is because I decided to make the codebase more object-oriented than I initially had intended. Instead of having all of the elements as GameObjects in the main Game class, I decided to create separate classes for the Player, Enemies, Platforms, and StarPieces, and have instances of them as their own objects. This meant the Game class was more tidy. I also implemented 2 cameras, which was not originally planned.

PNGs for the class diagrams can also be found in the same folder as this document