**Lab Week 5**

*50.033 Game Design and Development*

1004147 --- Riley Riemann Chin

**Are you participating in the Weekly Lab competition?** ~~Yes~~/No

**Provide the YouTube/other platform link to your screen recording:**

<https://youtu.be/y7zTKtf1AXw>

**Provide the link to your lab repository:**

<https://github.com/rileychin/50.033-Game-Dev-Labs/tree/lab5>

**Describe what you have done to achieve the desired checkoff requirement for this lab:**

[Your **high level** description here]

* You don't need to be too specific. This is just to assist us when we check your repository
* Scripts added, Assets added if any
* General modifications that you have done: eg animating the enemies, implementing FSM for the NPCs, etc.

Implement a Powerup Cast feature using the ScriptableObject Event system.

* When key Z is pressed, attempt to cast powerup in the first slot. Nothing should happen if you haven’t collected anything there. Else, it must affect Mario’s max or jump speed for a specific duration as dictated in the Powerup ScriptableObject instance.
* Similarly with key X, for the powerup in the second slot.
* You need to have at least two different powerups as per previous lab.

To implement the checkoff for the mario powerup, first I had to create CastEvent and CastEventListener.cs so that they will act as a new event for when a powerup is pressed.

Text

Description automatically generated

CastEventListener

Text

Description automatically generated

CastEvent

The raise and OnEventRaise accepts a keycode K argument for which key is pressed. Then, on PlayerController, I added an invoke for when the player pressed z or x so as to invoke the OnPlayerCast event.

Text

Description automatically generated

OnPlayerCast is a new scriptableObject instance

Graphical user interface, application

Description automatically generated

Used to handle all the cast event when players click z or x.

Graphical user interface, text, application, chat or text message

Description automatically generated

We subscribe playercontroller EV to the onplayer cast when he clicks z or x, which will then be listened to by the casteventlistener in powerupmanager gameobject.

Graphical user interface, text, application

Description automatically generated

Then it will call the AttemptConsumePowerup in powerupmanagerEV.

Text

Description automatically generated

Then the powerup will be disable after a few seconds

Text

Description automatically generated

Which will revert the changes for IntVariable for MarioJumpSpeed and MarioMaxSpeed!