

2D Platform Game

Technical Design Document

Contents

Objective	3
MoSCoW	3
Game Research	4
First Game – Fireboy and Watergirl by Osla Albet	4
Second Game – Big Tower Tiny Square by Evil Objective	5
Third Game – Cat Mario (Shobon no Action) by Chiku	6
Level Design	7
Difficulty Manage	8
Mechanic	8
Character Control	8
Movement and Jump	9
Coyote Time	10
Variable Jump Strength	11
Jump Buffering	12
Climb	13
Environmental Interaction	15
Health and UI	15
Damage, Respawn and Checkpoint	16
Collectable and Display	18
Trap	20
Switch Interaction	22
Dialog	23
Portal and Teleport	24
UI Menu	25
Level Transfer	28
Post Process Effect	29
Audio, SFX and Particle Effect	29

Objective

The aim is to explore on forgiveness mechanic and implement into the game. Further development platform game to improve gameplay and game feeling. The project would develop in Unity game engine and mainly focus on player and game mechanics.

MoSCoW

Must Have

- A controllable character that can jump and walk.
- Level design for player to interact and move around.
- Some forgiveness mechanics.

Should Have

- Character health component so allow player to die in the game.
- Damage actors to damage character
- Pickup and collect count system.
- User Input on health
- Menu for player be able to pause game, quit game
- Dialog to guide the player through the level

Could Have

- Next level manager
- Background music in main menu and main game
- SFX when player walk, jump, damage, and other collision.
- Audio control on volume
- Camera and UI shake when player received damage.
- Camera zooms in and slow motion when player die
- Particle effect due to player interaction

Won't Have this time

- AI enemies will follow player.
- A player sprite that contains animation
- Giving player input on attack, increase movement speed, interaction.
- Set this game as a Multiplayer game.
- Colour system.

Game Research

Research on 3 types of 2D platform game and get idea on mechanic and map environment to create stages and add in interactive mechanic to improve game feeling for player.

First Game – Fireboy and Watergirl by Osla Albet



A free flash 2D platform 2 players cooperative game. Players will need to control two characters 'fireboy' and 'watergirl' to solve puzzles and colour diamond in each level and aim to clear all stage in the game.

<https://www.1001games.com/adventure/fireboy-and-watergirl-1-forest-temple>

Design Ideas:

UI Instruction – Only appear with player close to located field

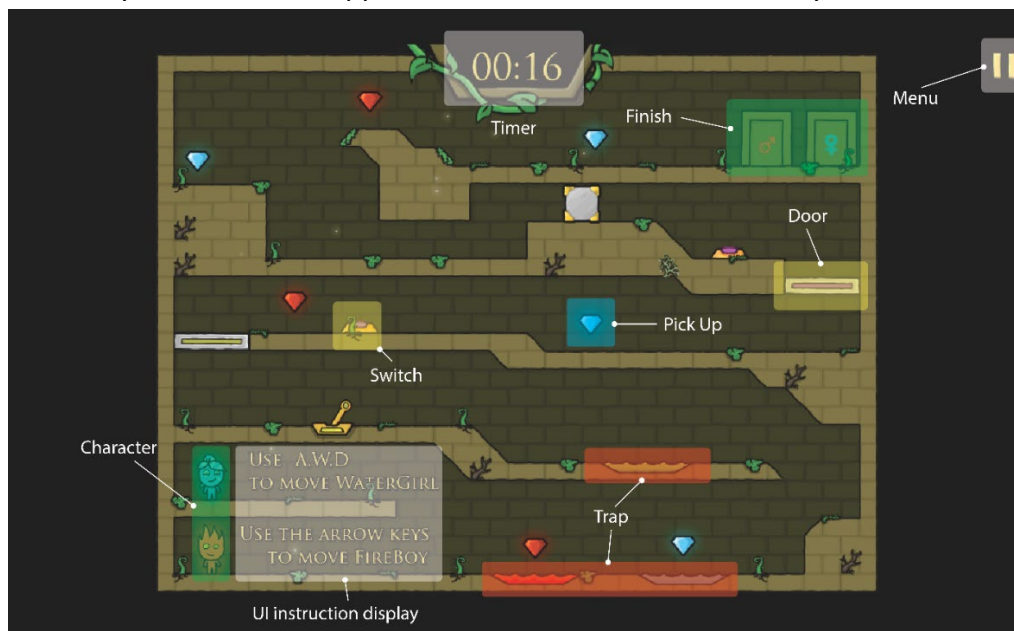
Trap – Character will respawn after one hit. Not life display or counted

Interactive puzzle – Door, platform and switches to interact with

Collectable – GEMs for player to collect and able to check collect state in pause menu

Timer – Counting display to tell player how long they took to clear the level

Game show heavy relationship on colours use and link to characters ability. Fireboy can't interact with any blue element, opposite Water can't interact with any red element.



Second Game – Big Tower Tiny Square by Evil Objective



A platform game require player to process level. Require player to jump most of the time. Simple and straight forward.

<https://www.crazygames.com/game/big-tower-tiny-square-2>

Design Ideas:

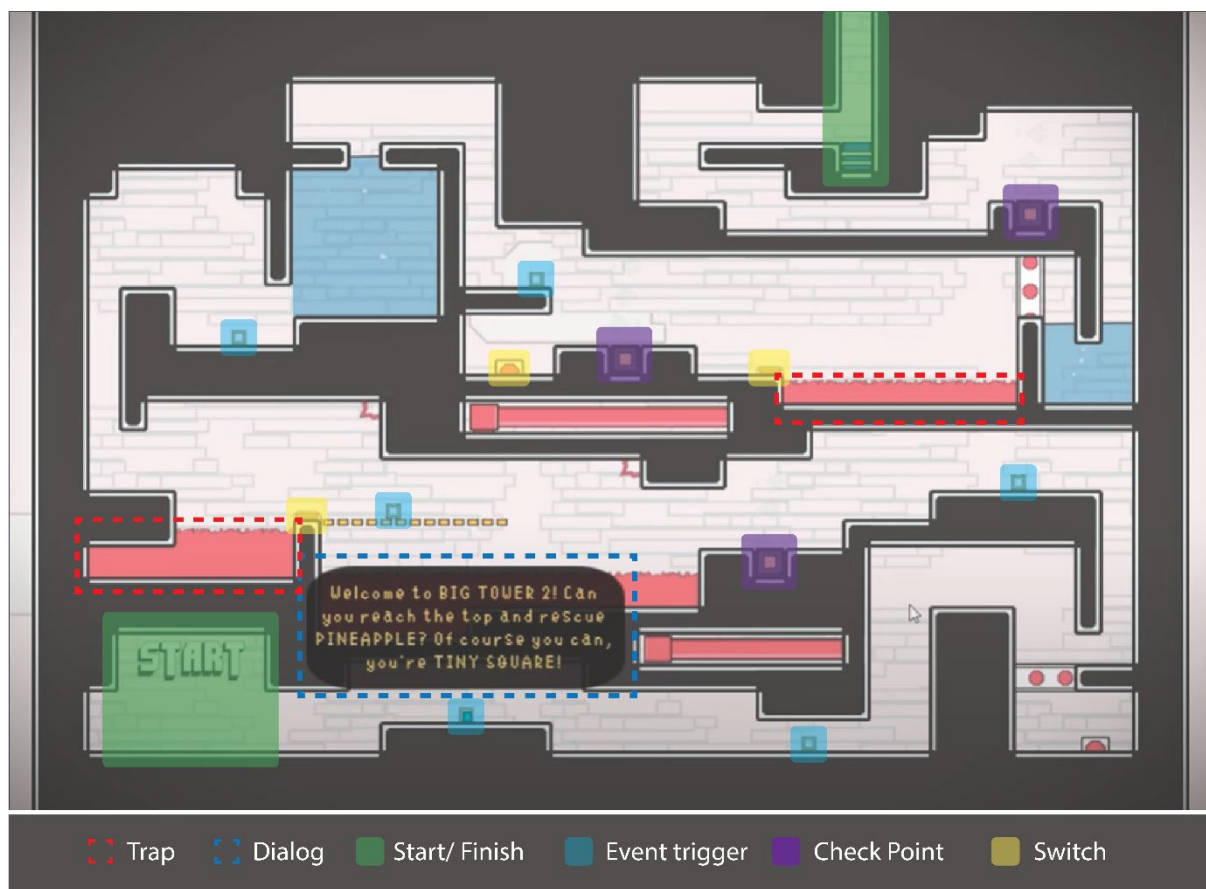
UI Instruction – Only appear with player close to located field

Trap – Character will respawn after one hit. Not life display or counted

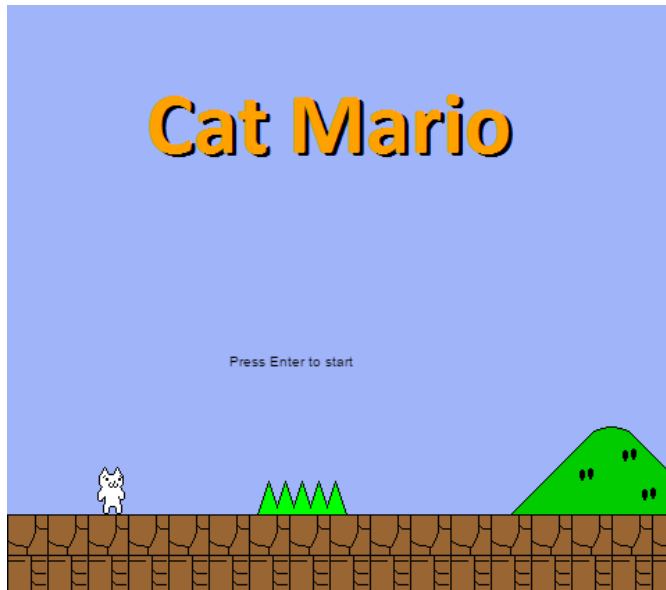
Interactive puzzle – Door, platform and switches to interact with

Checkpoint – Checkpoint for player to trigger during play

SFX and effect – Player movement sound effect, die particle display, player interaction effect
Game show heavy use on particle effect and sound effect to communicate with player, level mapping is simple and straight forward.



Third Game – Cat Mario (Shobon no Action) by Chiku



Platform game challenging player patient. An opposite logic with normal Mario game. Traps and enemies everywhere. One word to describe “Unfriendly”.

<https://catmario.eu/>

Design Ideas:

UI Instruction – Display player life remain, could when to negative point

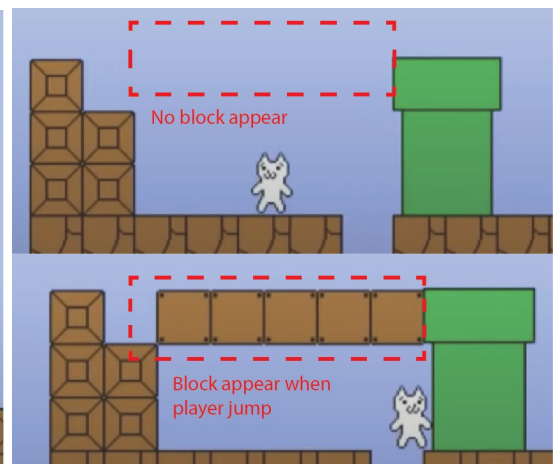
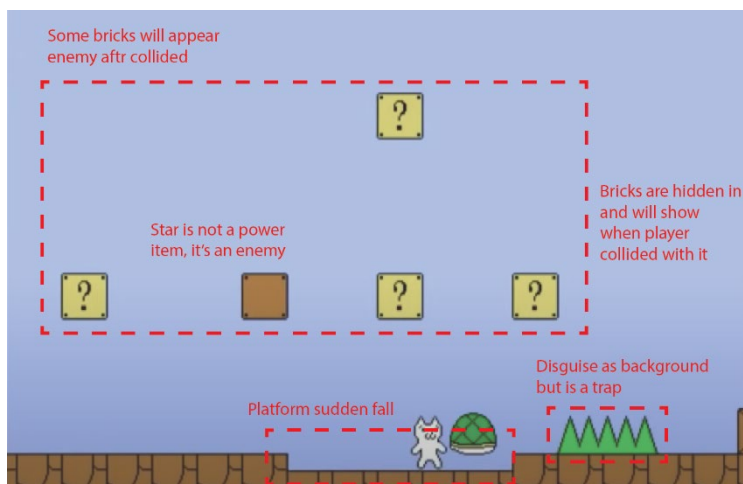
Trap – Any object in the game could be a trap

Death & Respawn – Respawn when even player received a hit from trap or enemy

Checkpoint – Checkpoint for player to trigger during play (only in 3D version)

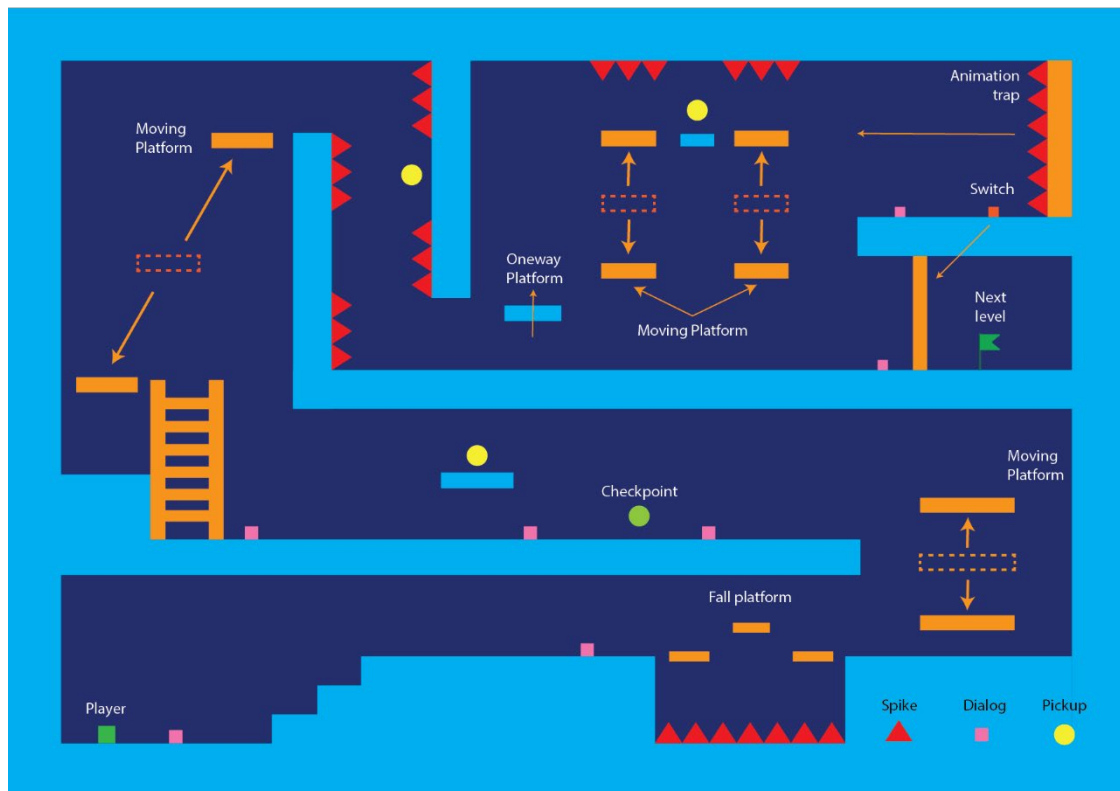
SFX and effect – Player movement sound effect, die particle display, player interaction effect

Game mainly focus on interaction between player and mechanics, traps and enemies everywhere unexpected. A sudden block appear when player processing jump in air. Require player to try and error and get use to the map.

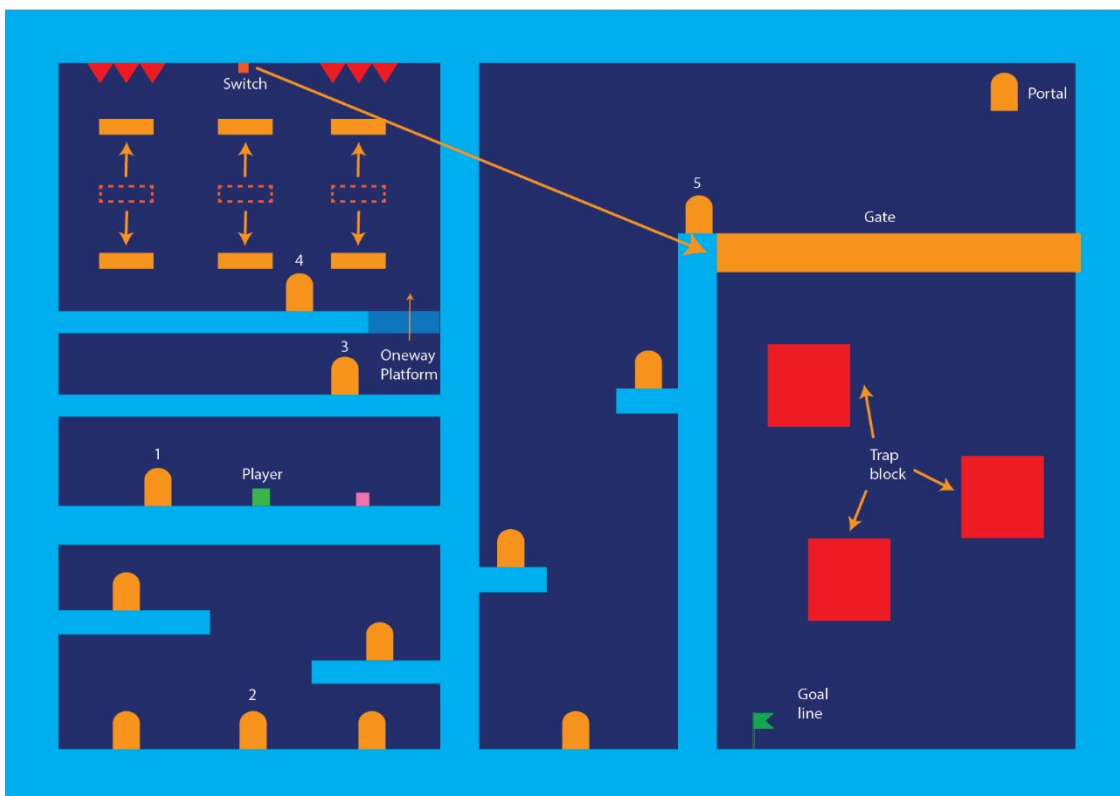


Level Design

Stage 1:



Stage 2:



Difficulty Manage

- There will be a stage to notify player the control and some mechanic in this game.
- After player get use to the control, the main game will begin when player travel through the blue portal.
- Stage 1 contain some mechanic like pickup, platform trap and instruction to guide player through the stage.
- Stage 2 is a simple puzzle game to go through portal to solve the puzzle and reach the goal.
- Idea of stage manage is begin with instruction and guiding, then player could start go on their own afterward, slowly increase the difficulty which require player thinking and imagination.

Mechanic

This project will contain mechanic for player to control character, some forgiveness mechanic that to understand player mistake and give forgiveness in some control in the game (Coster, n.d.), some external mechanics display character interaction and improve game feeling.

Character Control

Move Action – Keyboard A & D

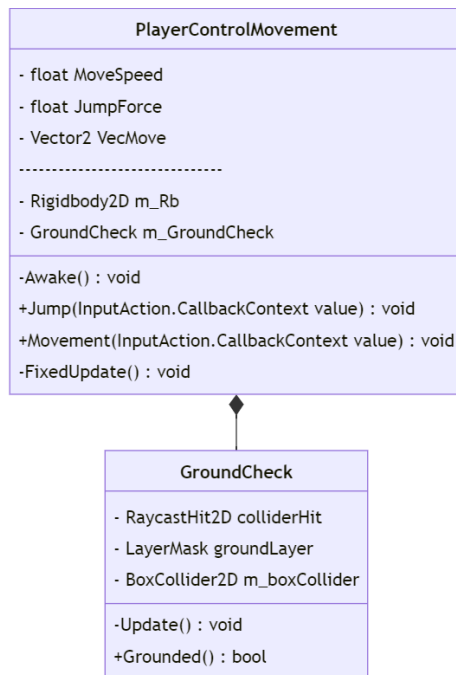
Jump Action – Keyboard Space

Climb Action – Keyboard W & S (Only available on ladder)

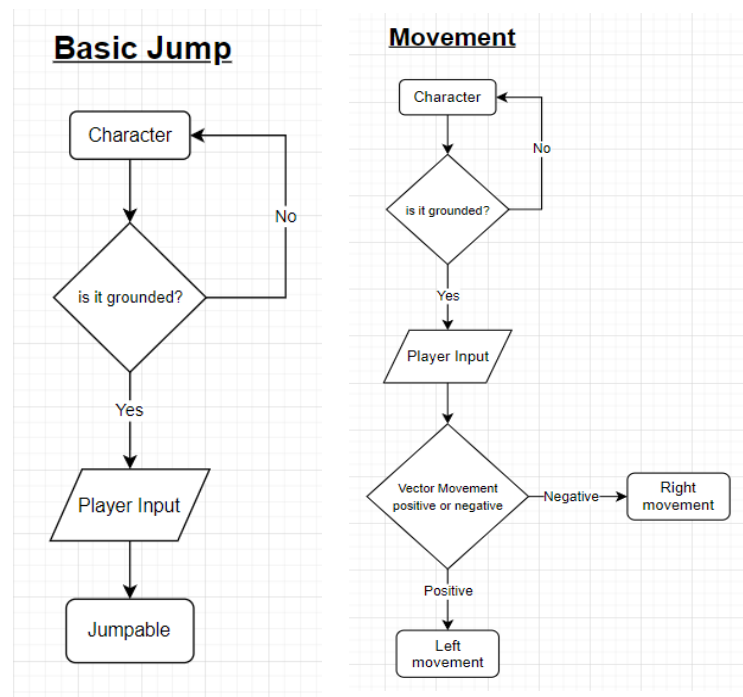
Movement and Jump

Unity new input system – input action mapping and unity event invoke.

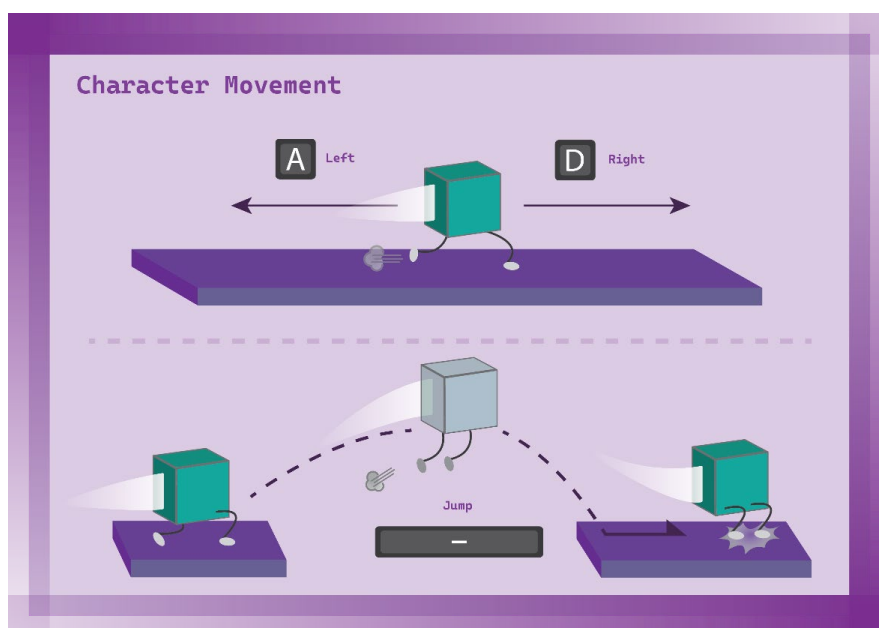
- Assign input and related action per function instead using update function.
- Require a Ground check script with Boolean function and Layer mask check player landing condition. Reference ground check script in player control script from start to perform jump function, else would cause character alibies of endless jump.
- Separate ground check and player control script, assign both scripts to character game object. Set and assign collision that use for ground checking and assign any game object that the character would land on layer to ground.



(Class Diagram)



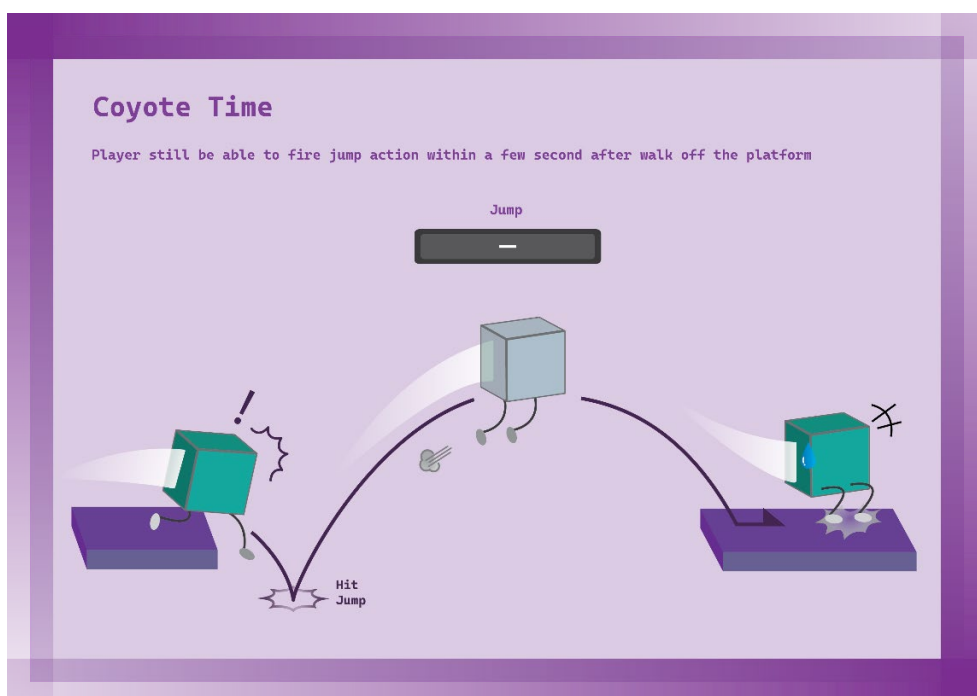
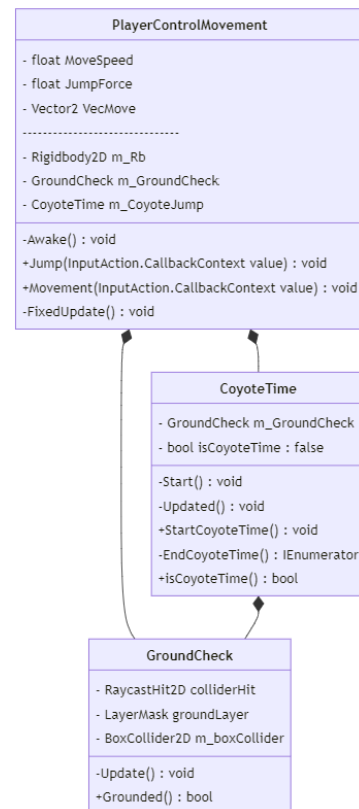
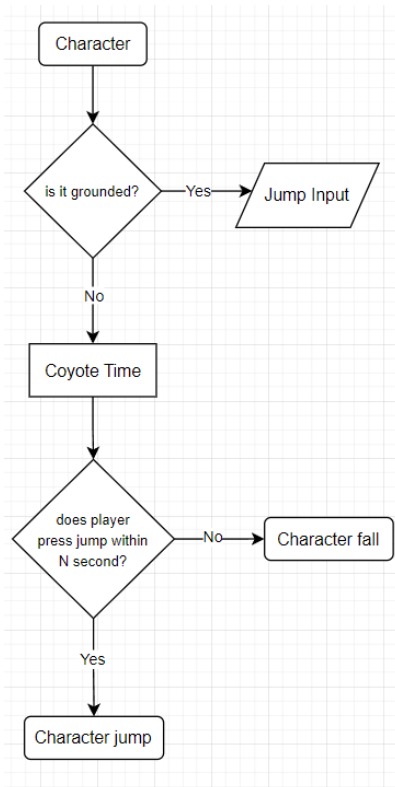
(Flow Diagram)



(Mechanic Diagram)

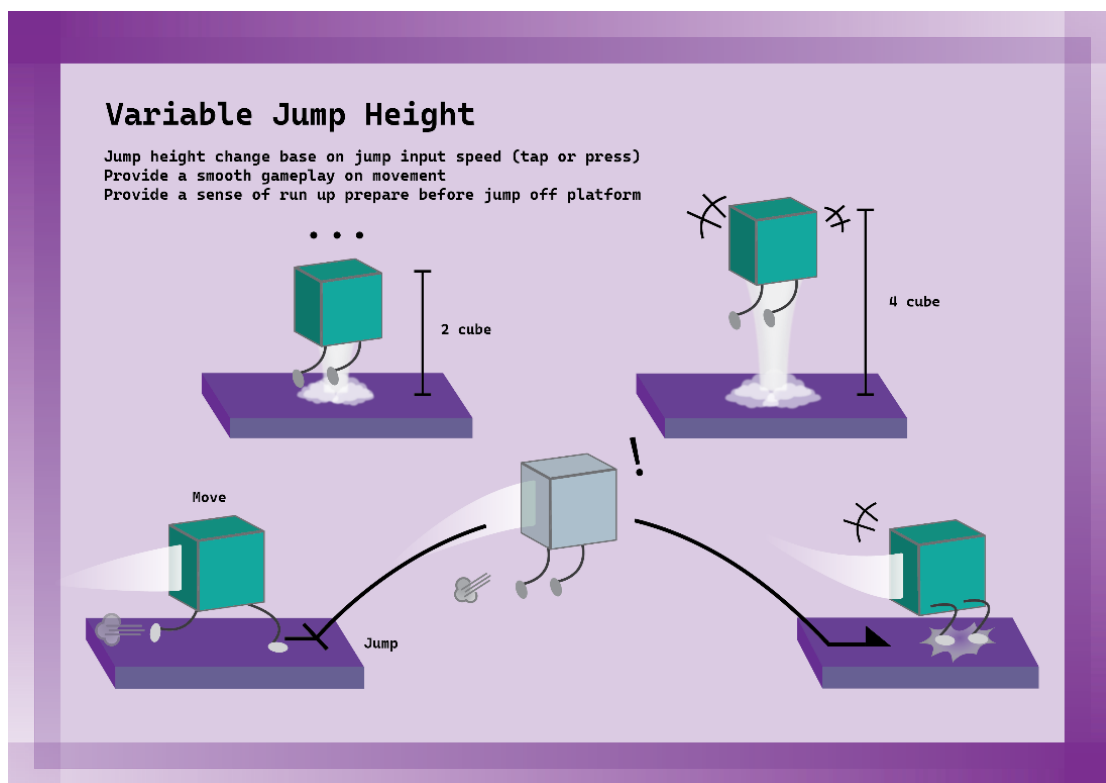
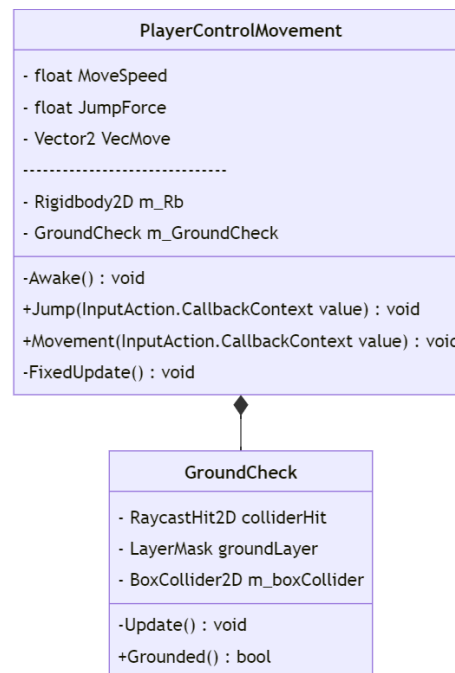
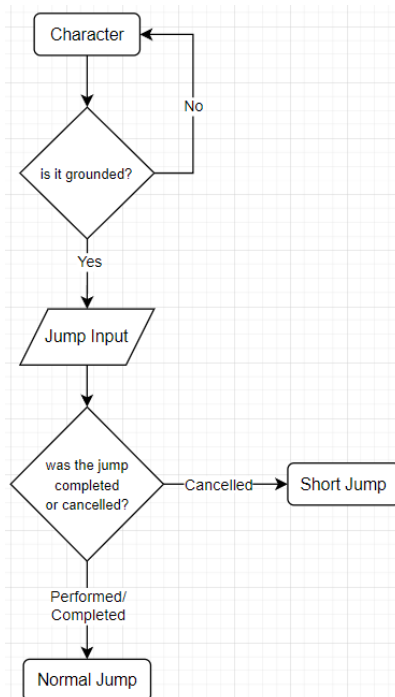
Coyote Time

- A type of forgiveness mechanic allow player still be able to trigger jump action within set time after they fall off the platform. Aims to forgive player slow reflection.
- Coyote time implement as an individual script and separate from player control script.
- Require a box collision on character game object to detect player off or on platform, a different collision from ground check script.
- Run coroutine when character off the platform and start timer count down, once waiting time reach jump act will be de-active.



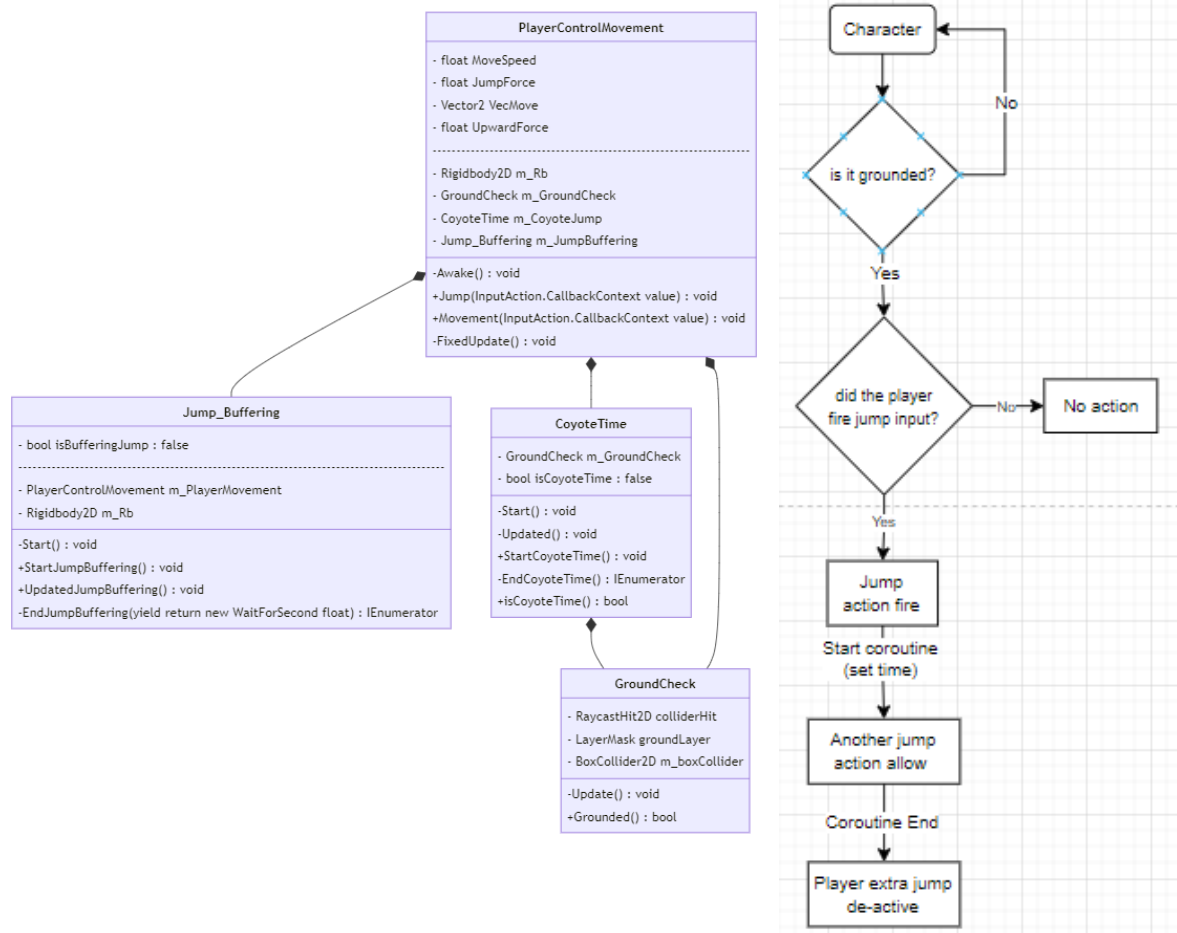
Variable Jump Strength

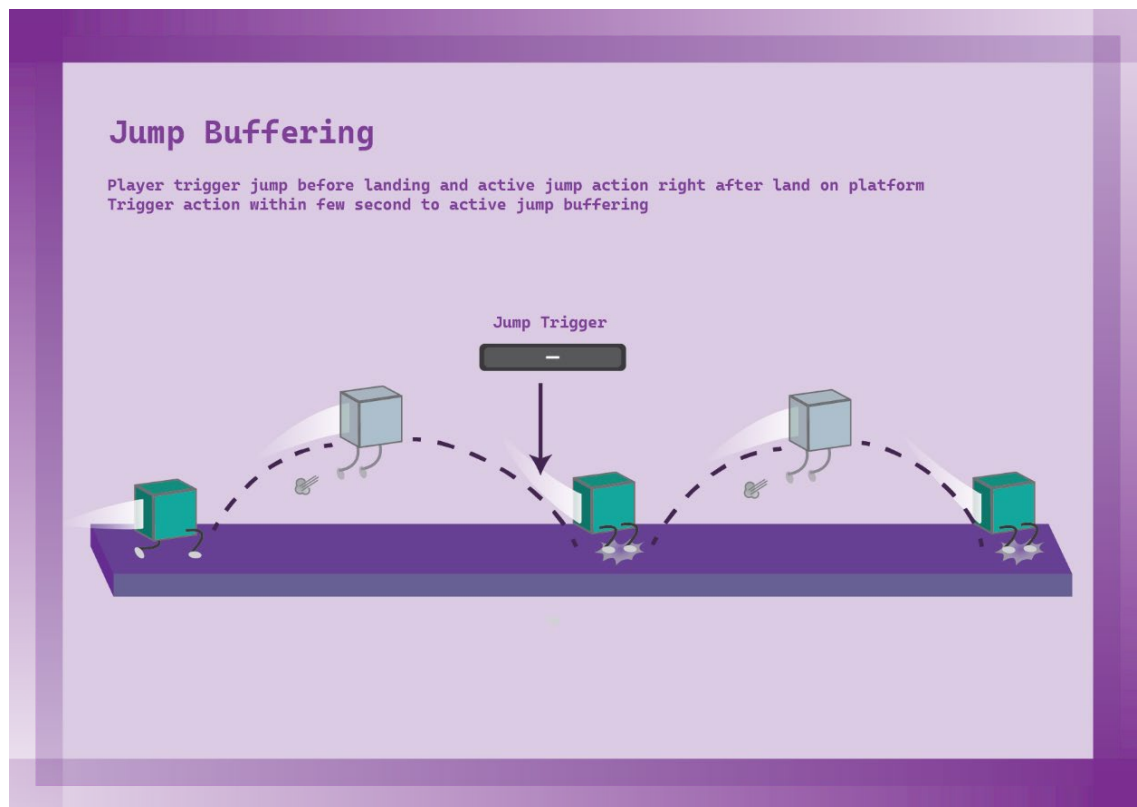
- A type of forgiveness mechanic player could trigger different jump height accord to player input.
- Variable Jump Strength require respond to action in Jump function, define dose player trigger jump input in a completed action or interaction been cancelled.
- Variable Jump Strength will need to be perform in Jump function, result player perform jump in high or low height.



Jump Buffering

- A type of forgiveness mechanic player could perform another jump action between before and after land on platform. Benefit to keep player movement and speed up while controlling the character.
- Jump buffering in this script also be separate from player control script, apply it as an extra element.
- Jump buffering script is not complete, its working but in a frankly way. Action triggered like double jump, but also work as jump buffering mechanic.

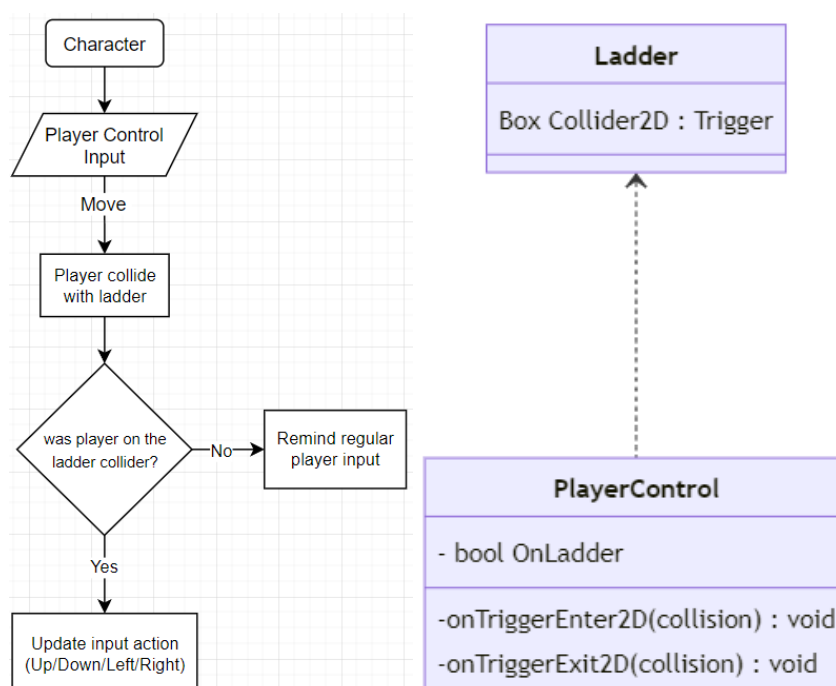


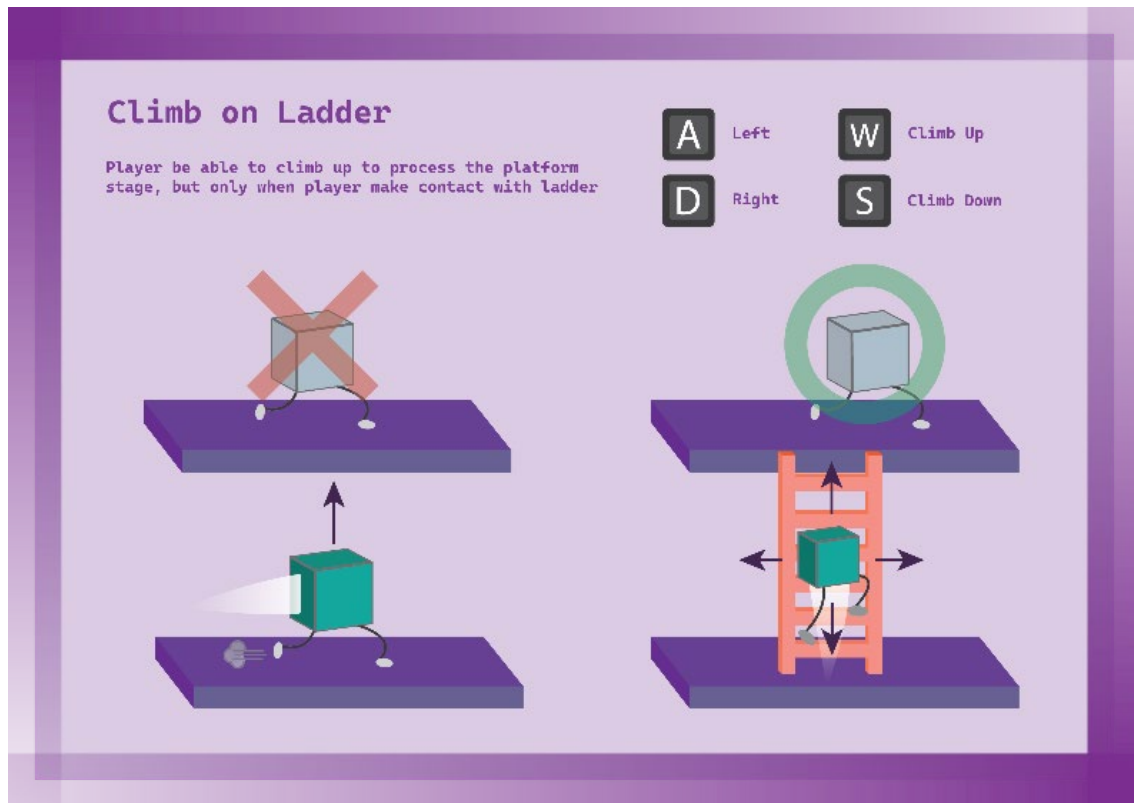


Climb

Unity new input system – input action mapping and unity event invoke.

- This is an input action extension. Allow player interact with an Object got a particular tag and perform extra input action.
- Climbing only allow to fire when player interact with ladder or climbable object in the game. Allow player gain extra action on up and down instead only left and right.
- Gain more movement input for player in the game instead left, right and jumping only.



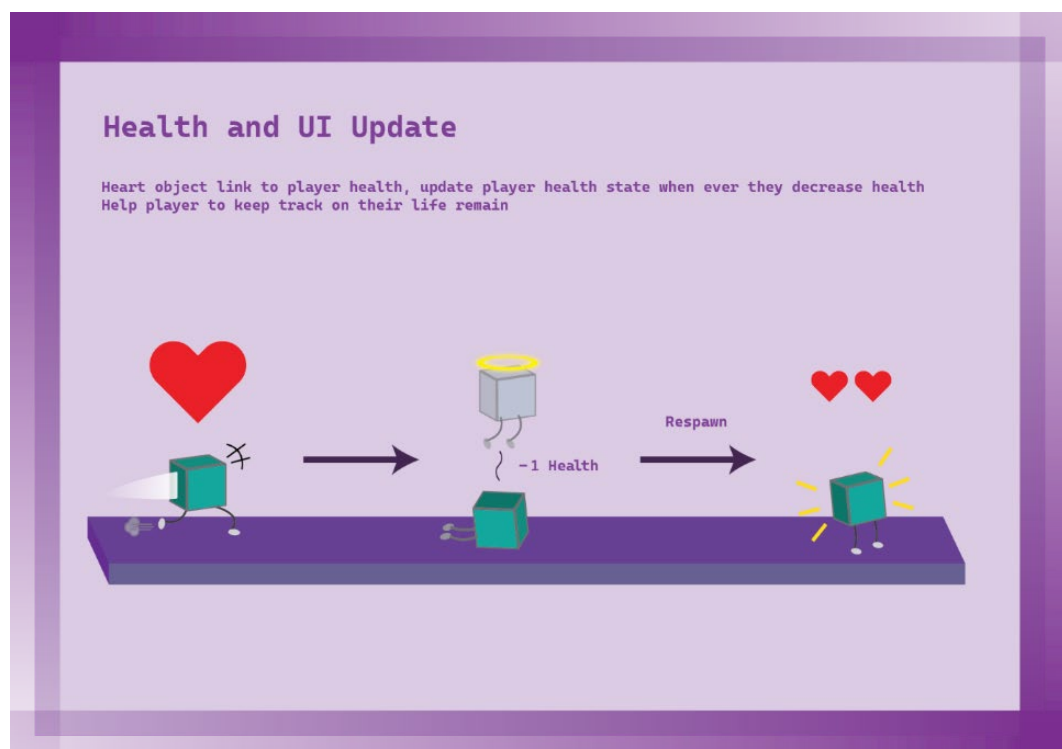
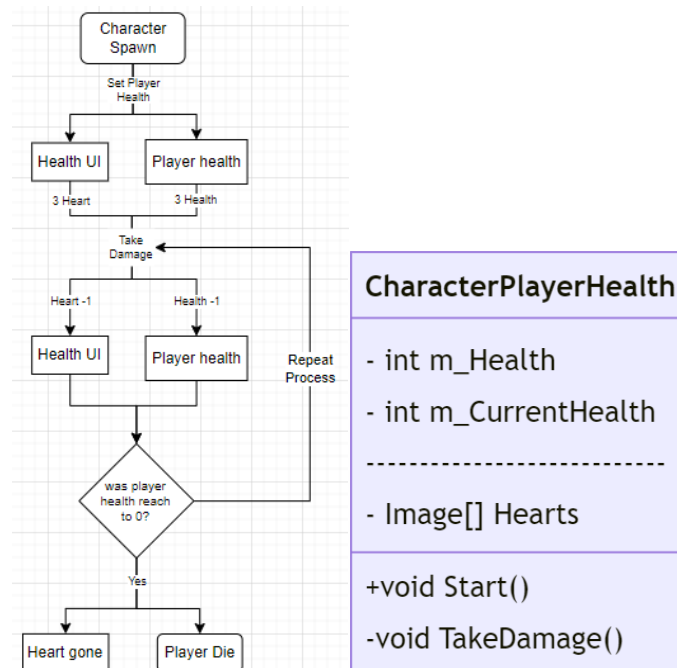


Environmental Interaction

This part related to the details of character interact with in the project, so it would be related to character interaction with game object, health and damage component, user interface and SFX aware player of their condition etc. These essential would give benefit to improve game feeling and game play.

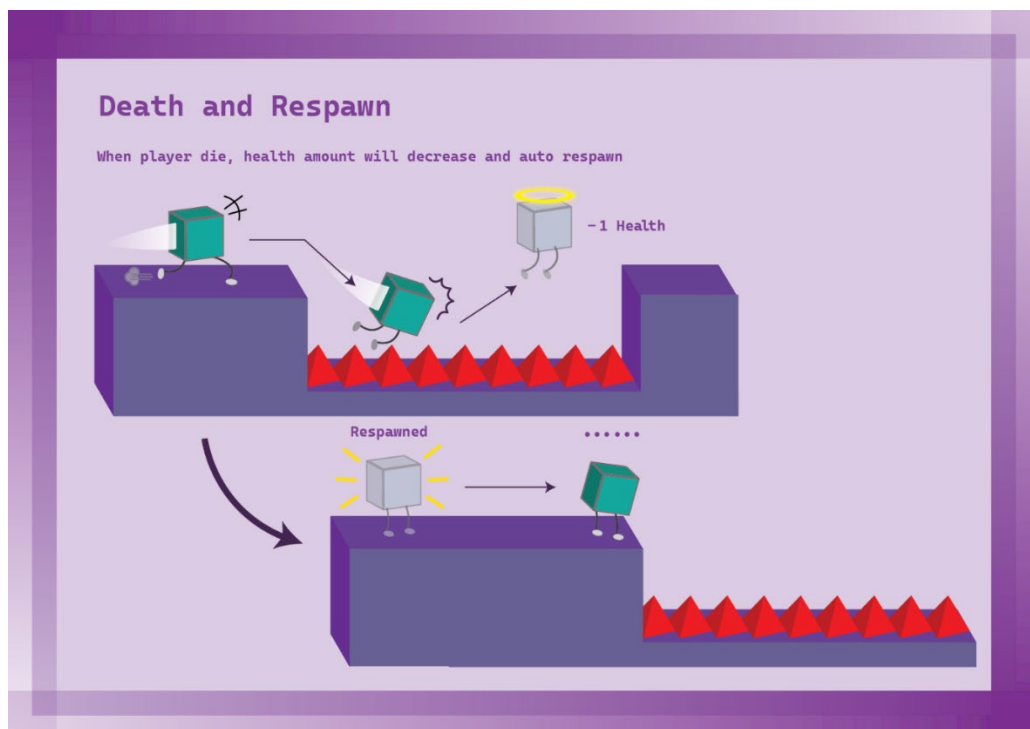
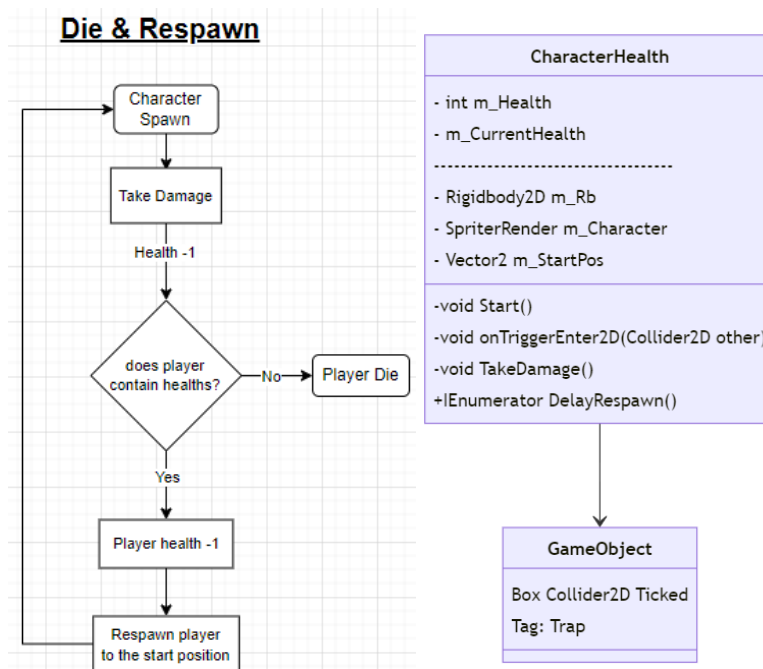
Health and UI

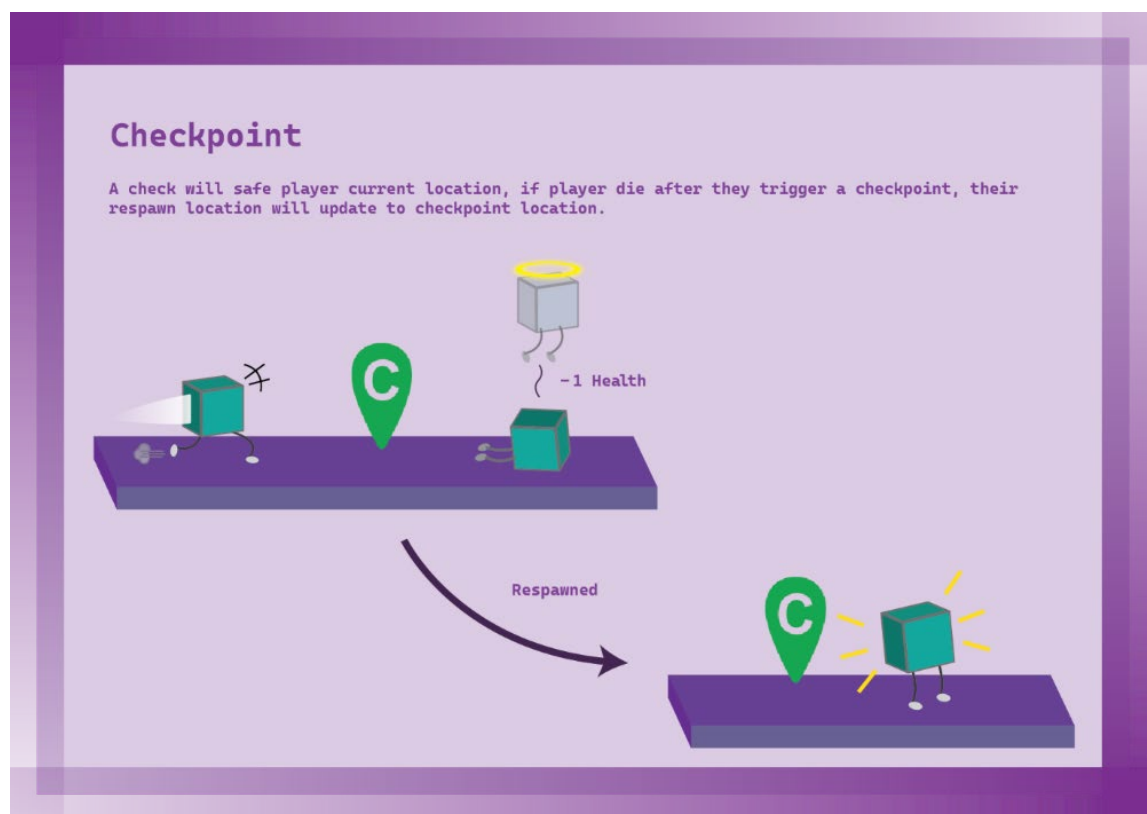
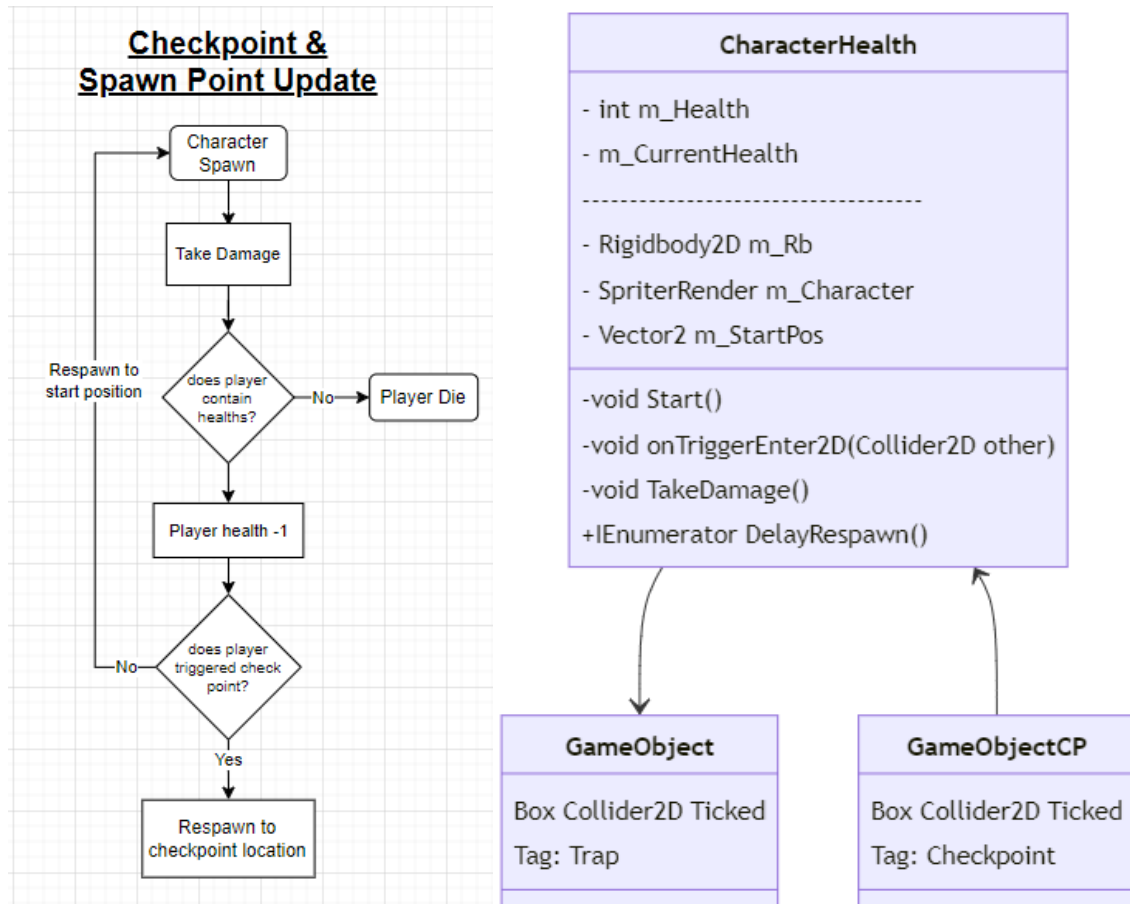
- Health script assigned to player character also contain Health UI reference. Assigned by array of image and include a code to check player current health state and link it to image array.
- Heart image will decrease per player receive damage.



Damage, Respawn and Checkpoint

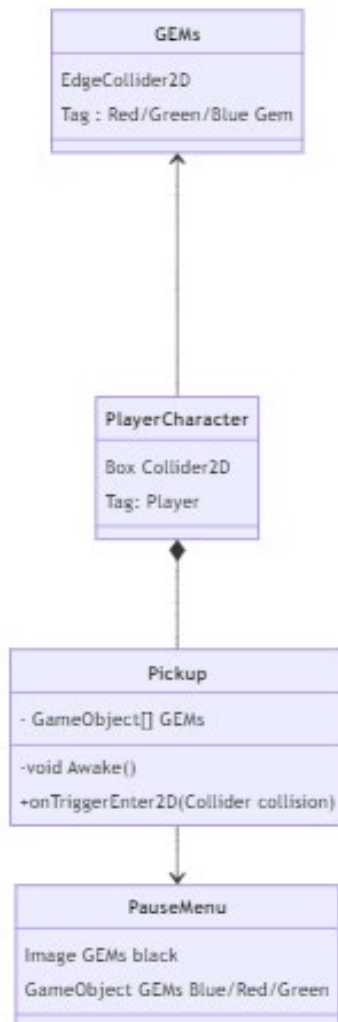
- A health script contain player health, take damage function and on trigger function to create player life, damage and respawn.
- Start function set player health and respawn location and on trigger function to check tag in game with tag assign as trap to collide with
- Run the take damage function once on trigger detect tag and run coroutine to respawn, coroutine will need to stop player movement and inactive sprite to show player was hit to some trap.
- On trigger will also have checkpoint tag to run code see if player trigger checkpoint object in the game, if yes update spawn point. Trigger checkpoint and damage effect will be list detail at (Effect section).



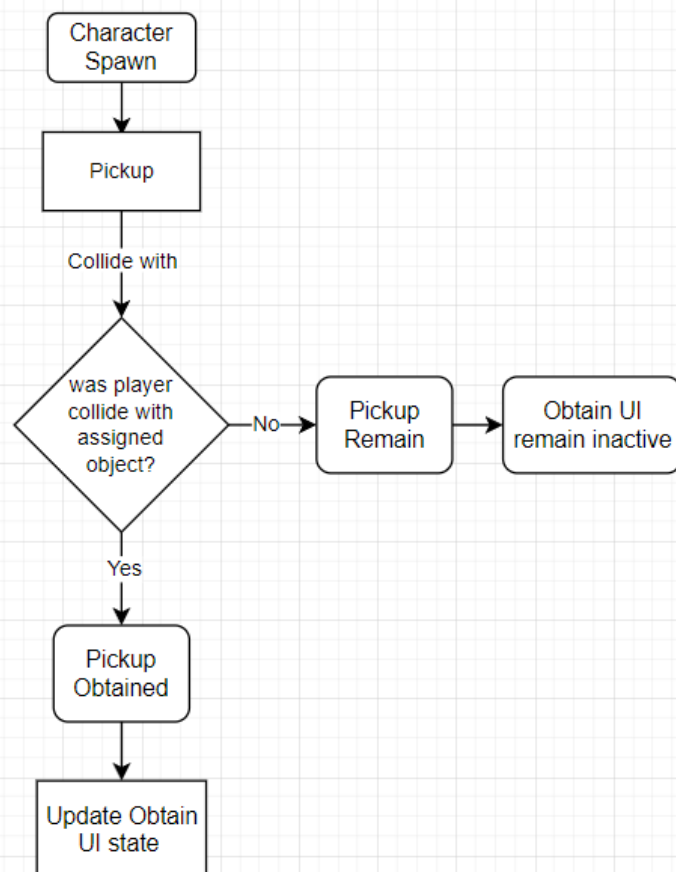


Collectable and Display

- Player character would be assigning a pickup script to allow pickup and save collected data for display in pause and checkable by player.
- A trigger function in pickup script and link game object in UI pause menu and set active or inactive base on player action.
- To prevent player could trigger multiple times, destroy the object once player trigger with the GEMs to prevent confusion.

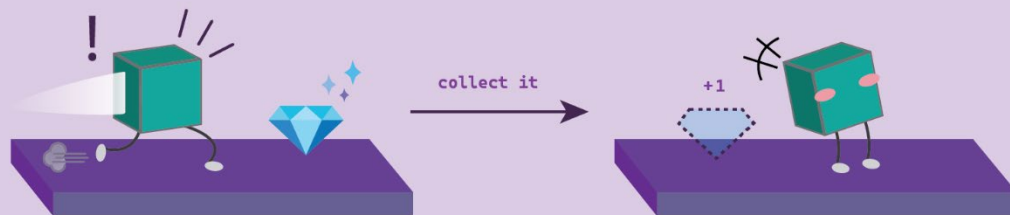


Collectable & UI Display



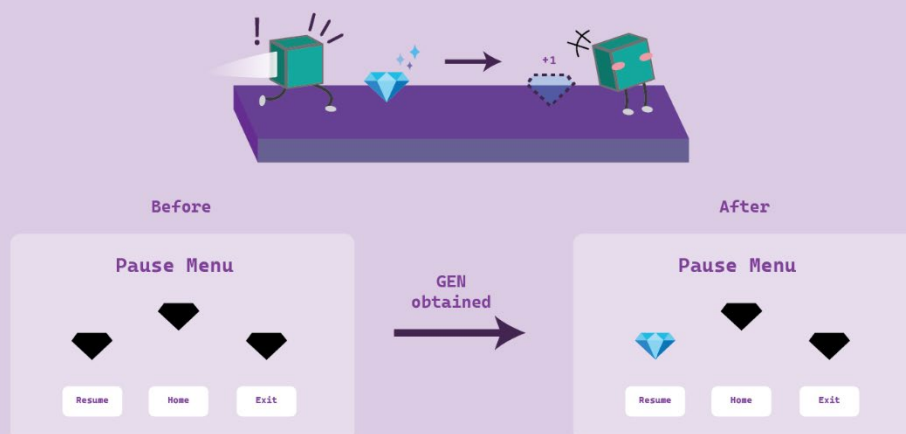
Pick Up Mechanic

Player be able to climb up to process the platform stage, but only when player make contact with ladder



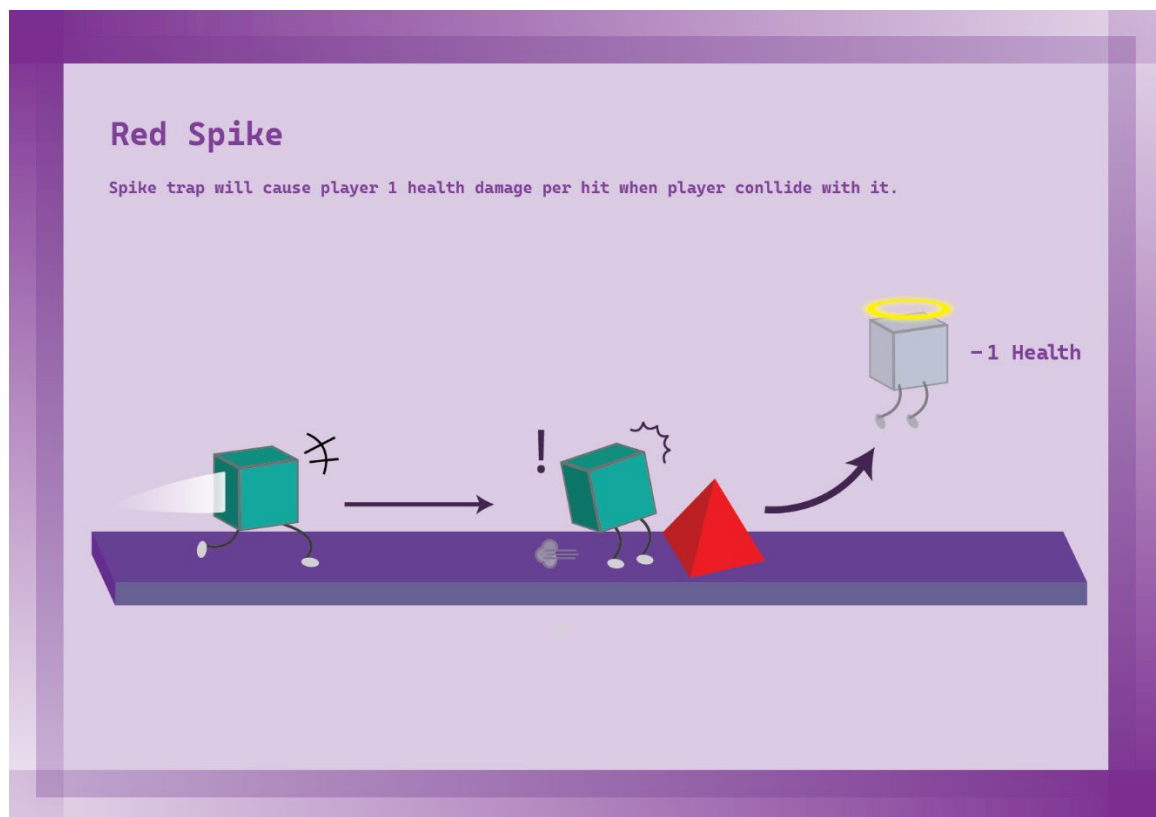
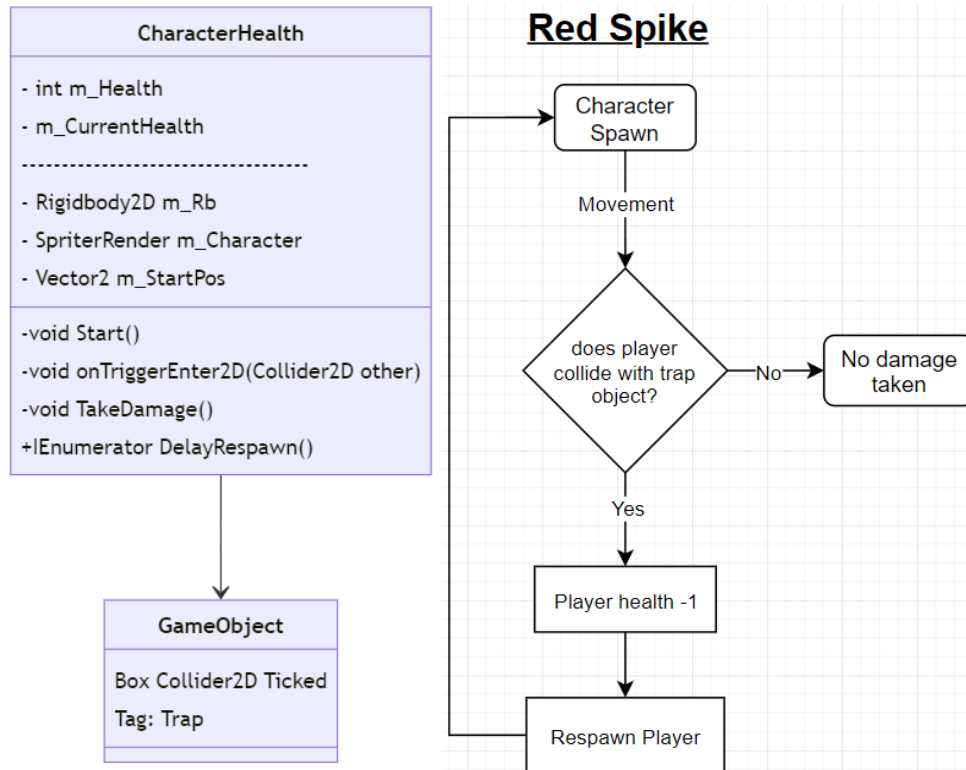
Collectable State

When player obtained a GEM, the pause menu will update the state of GEM.

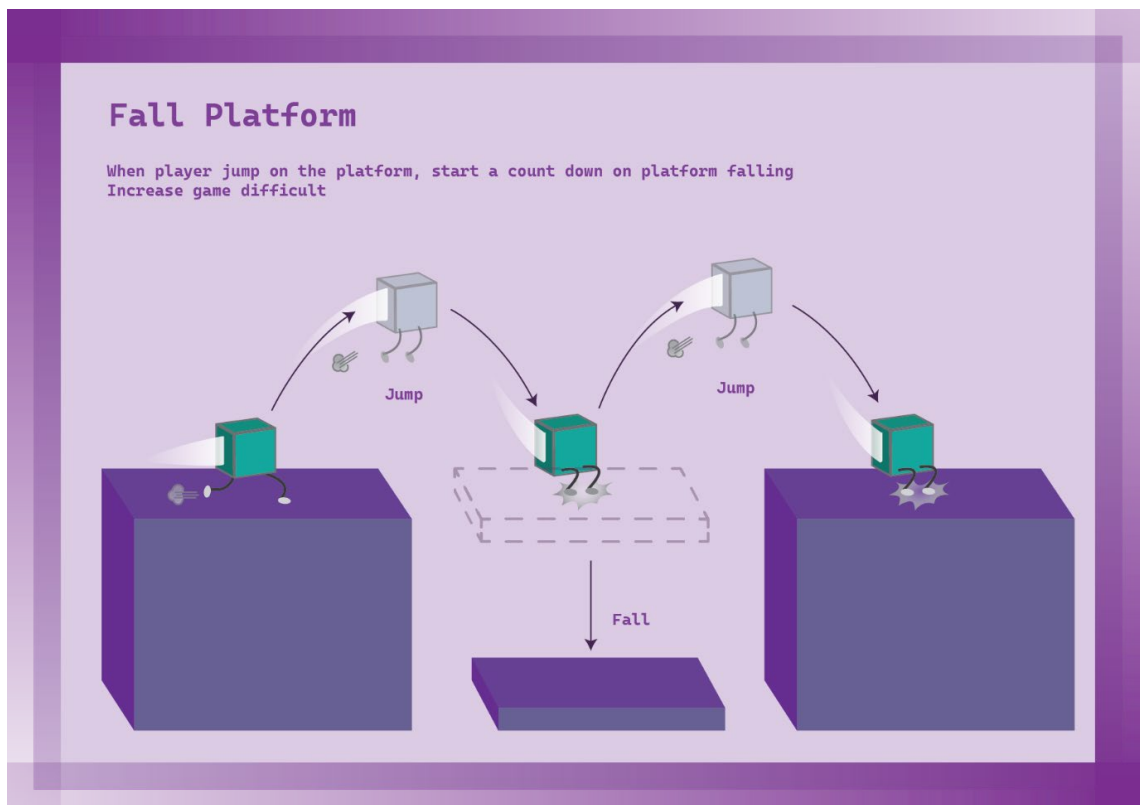
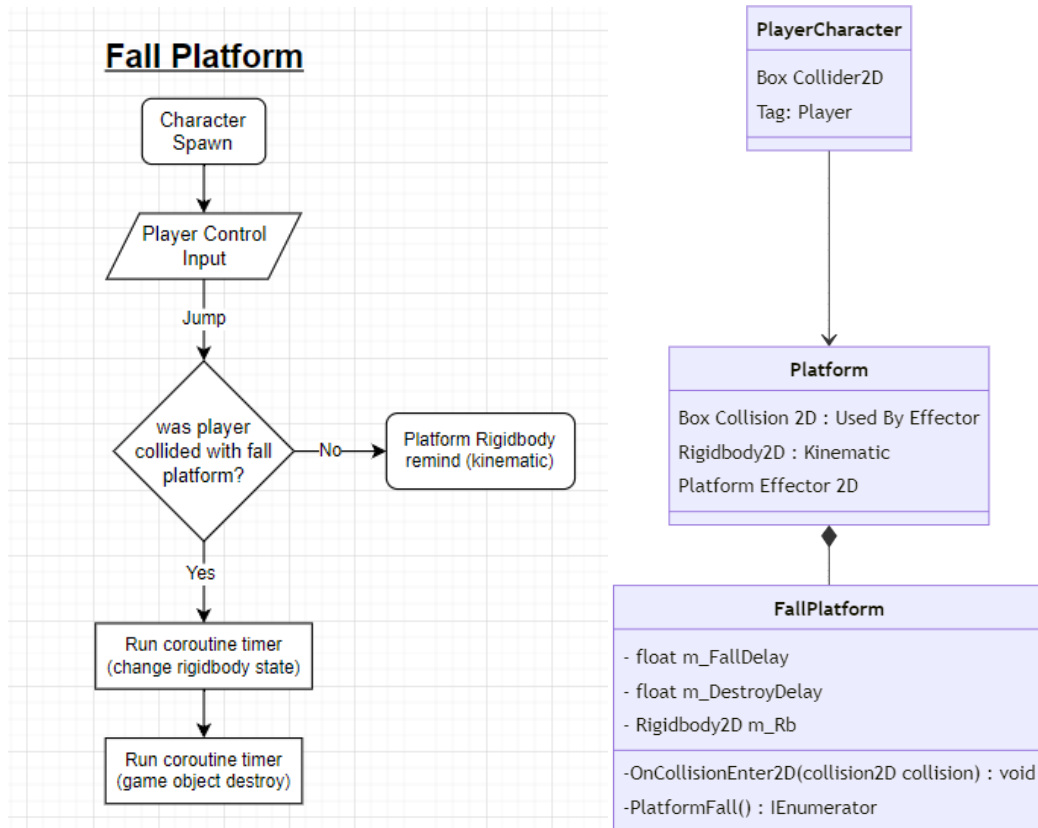


Trap

- Game Object in the game with trap tag and collider box then run through player's health script on trigger enter function to run damage function.
- Red spike doesn't require individual script but got reference from health script, this wasn't the best solution but it's functioning the game.

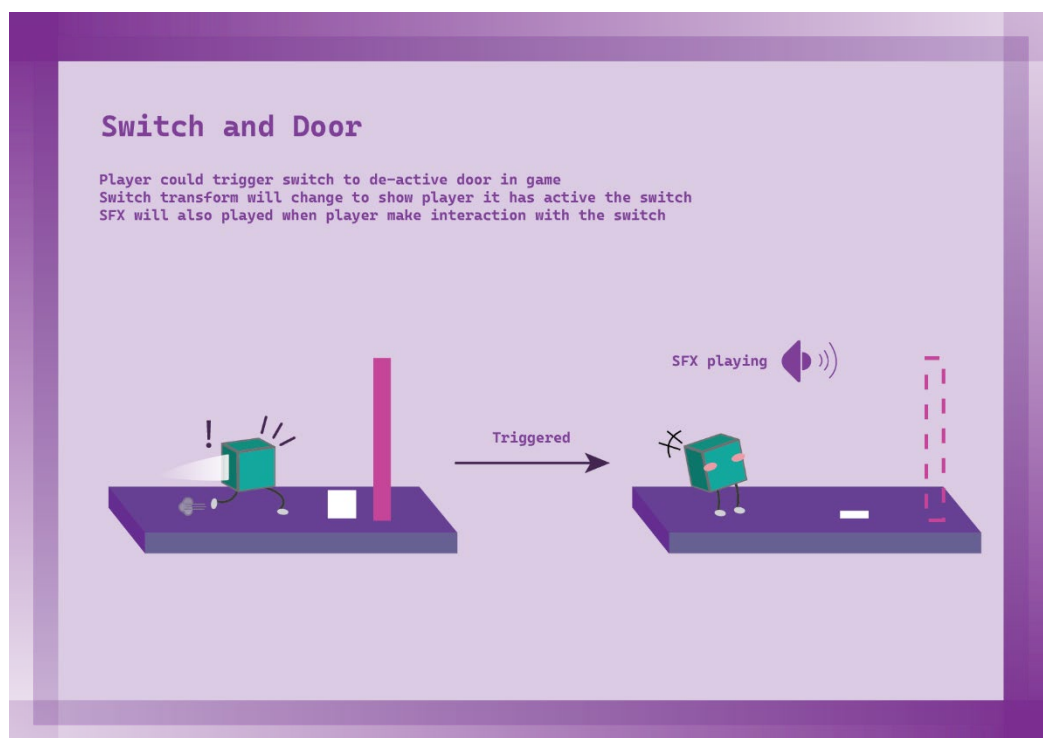
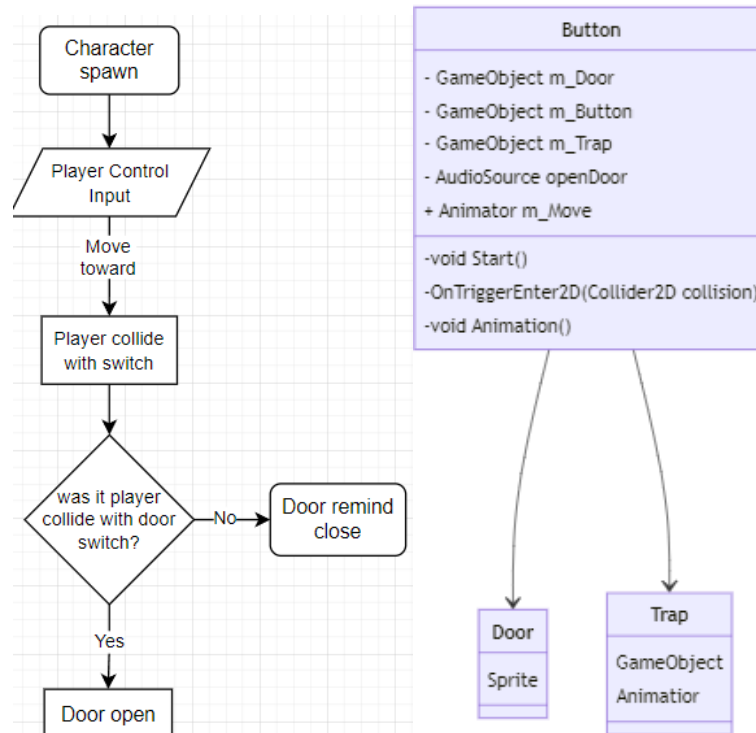


- Fall platform will need a regular platform object and attach fall platform script to the it.
- Trigger from on collision function and run a coroutine of count down time of the platform fall action, also destroy object after few seconds.
- Set the platform rigidbody from dynamic to kinematic then change it opposite once coroutine run.



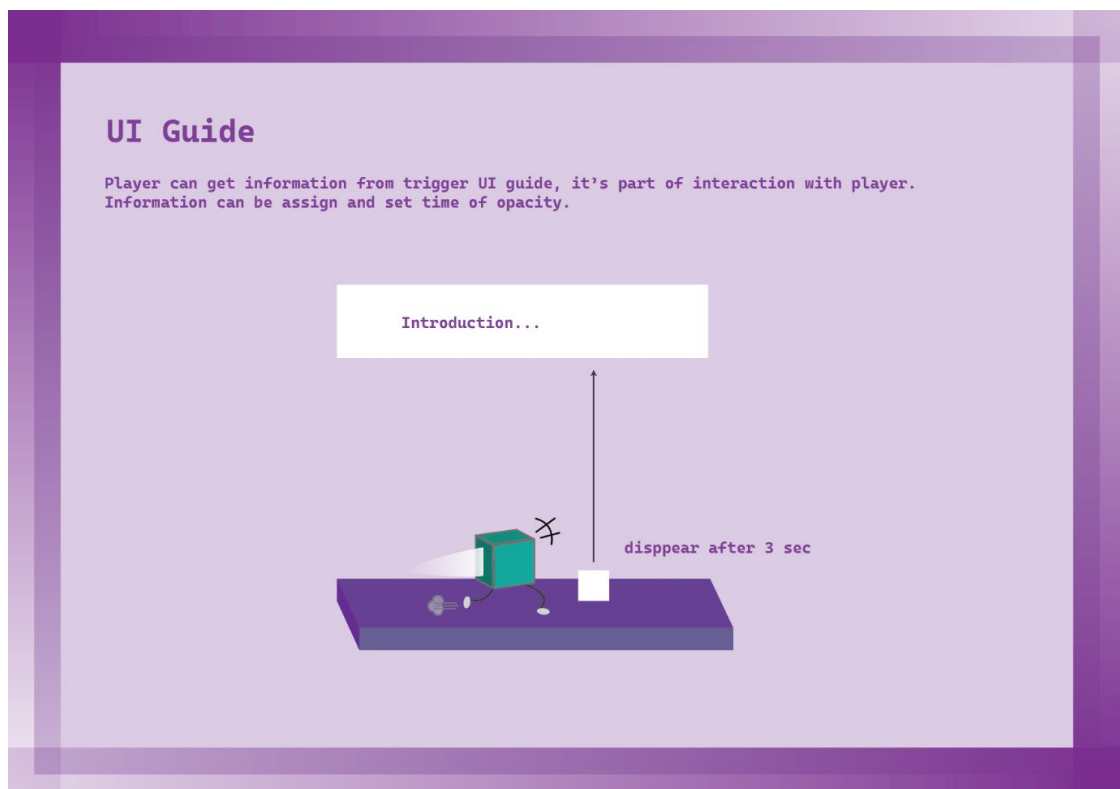
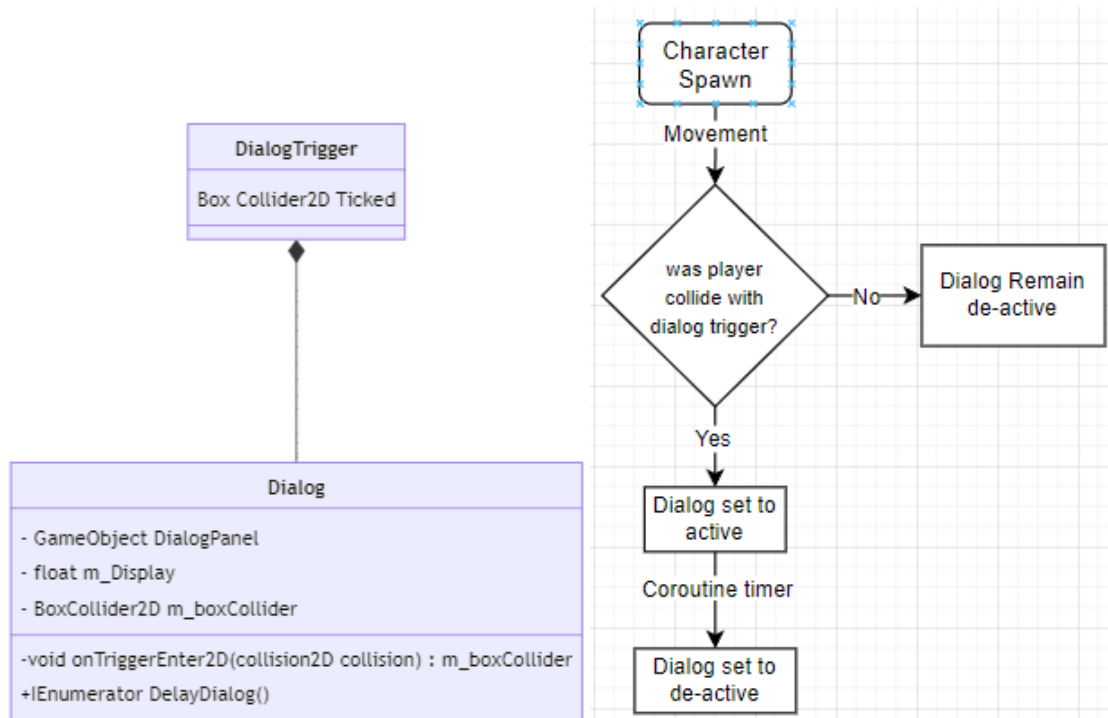
Switch Interaction

- Switch and door interaction. Player could trigger a switch to open a door.
- Switch and door will be assigned in the script and fire on trigger function when player interact with it, door will set de-active and button will change transform make it looks like switch has been triggered.
- SFX also include in the switch script to notice player they trigger the switch.
- This script also includes an extra action to play an animation to active a trap after player interacts with the switch.
- Create animation and get reference from the game object and fire play animation function.



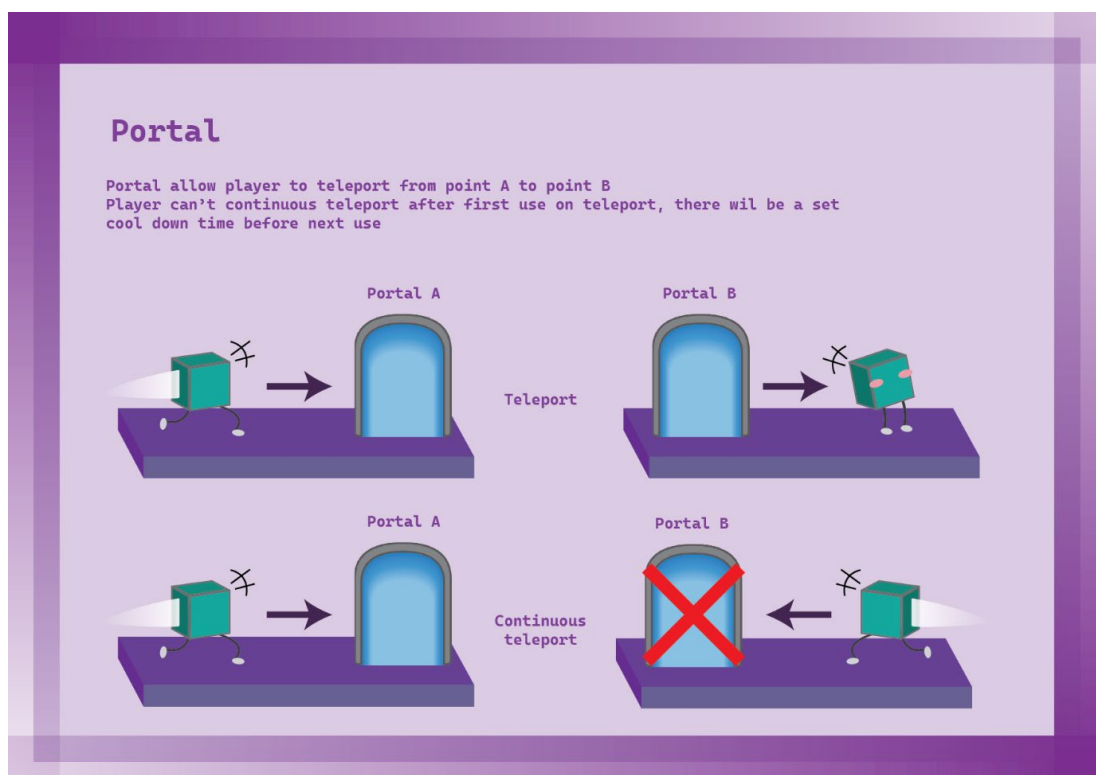
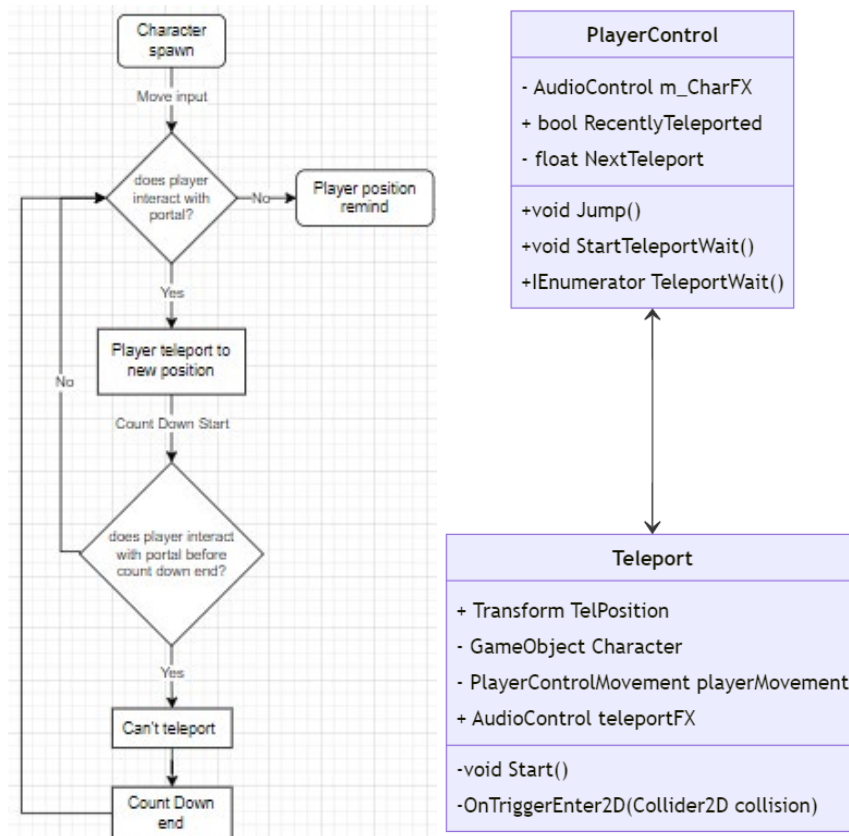
Dialog

- Player will get information by triggering dialog trigger in game and dialog would remain on screen for a set time.
- Require a script on the dialog trigger and set function to active dialog from canvas for set time assigned then could be change in editor.
- To be sure dialog does trigger properly, get reference of a collider box from player character, and assign to trigger the dialog with particular collider box from player character instead of player tag. Else it might receive collision error of some work and some not.



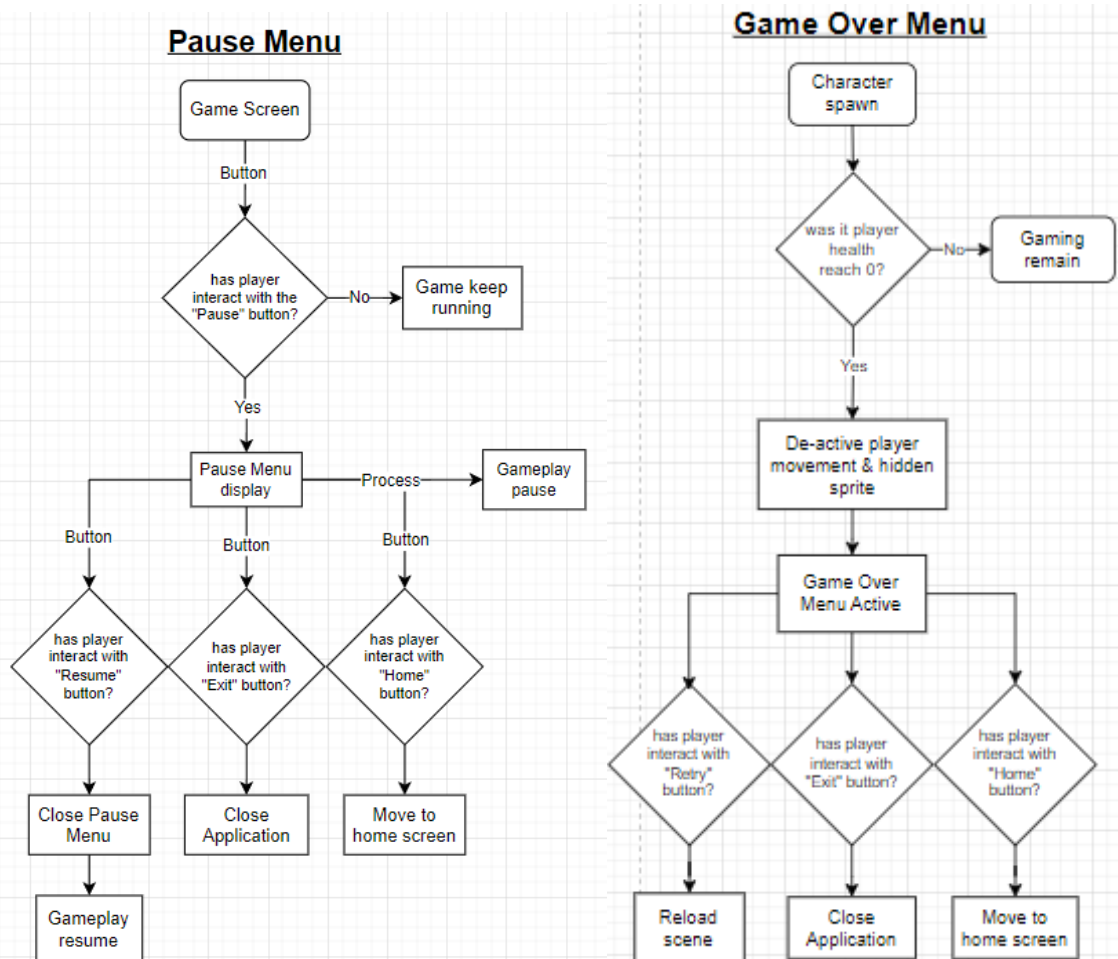
Portal and Teleport

- Player could teleport from one location to another by using portal.
- Setup transform position and code to allow player to move current location to a set location, it can be assign from unity editor to be handy to designer.
- Link between script of player control and portal to check player condition of teleportable, if teleported player will need to wait for certain set time before the next teleport action.

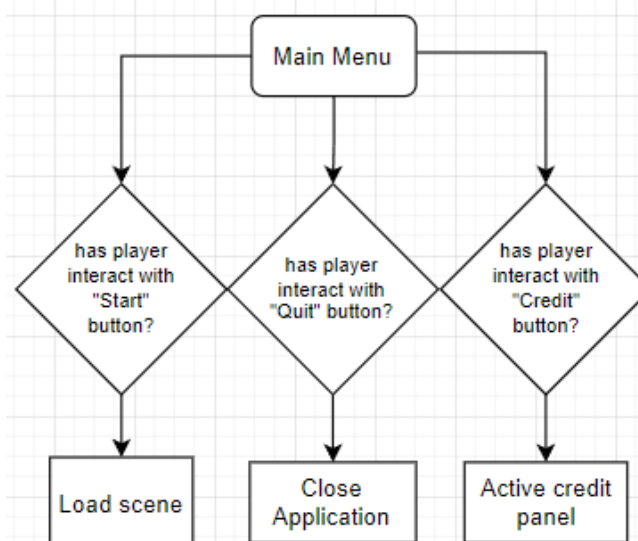


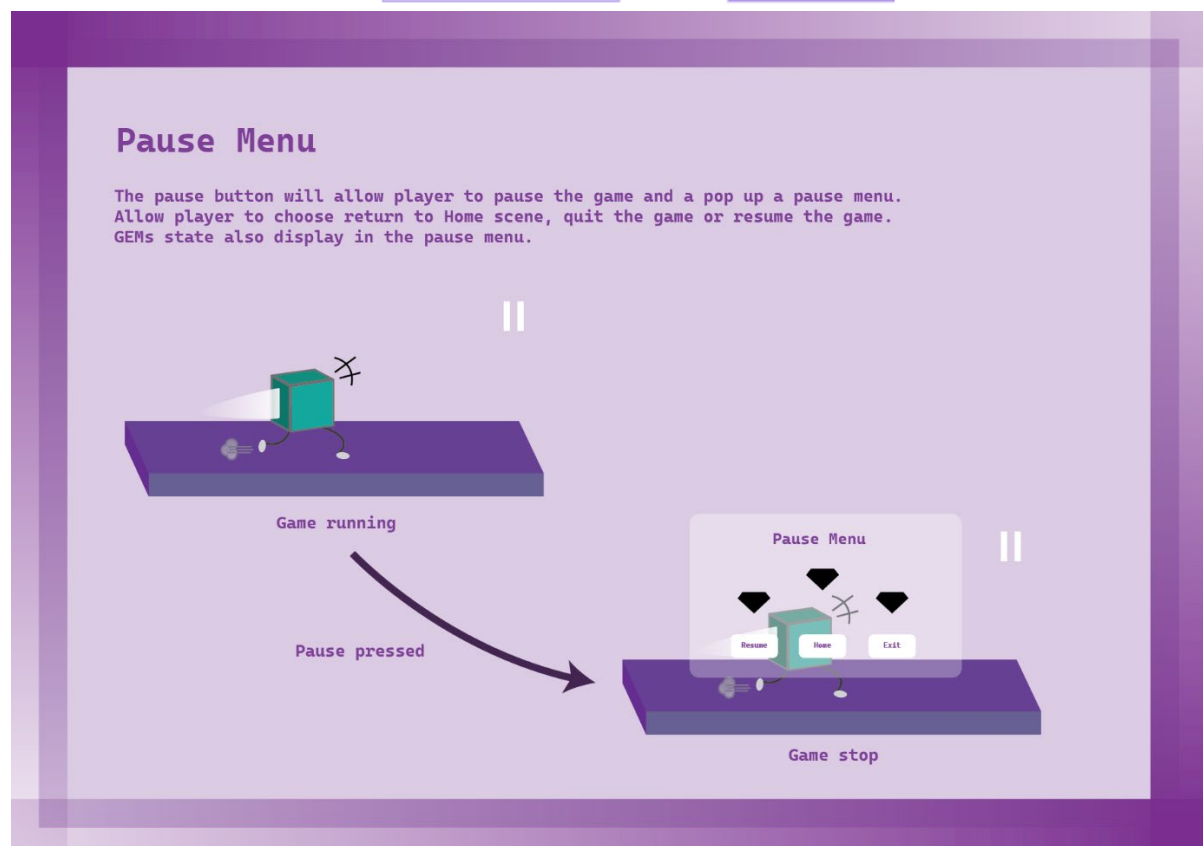
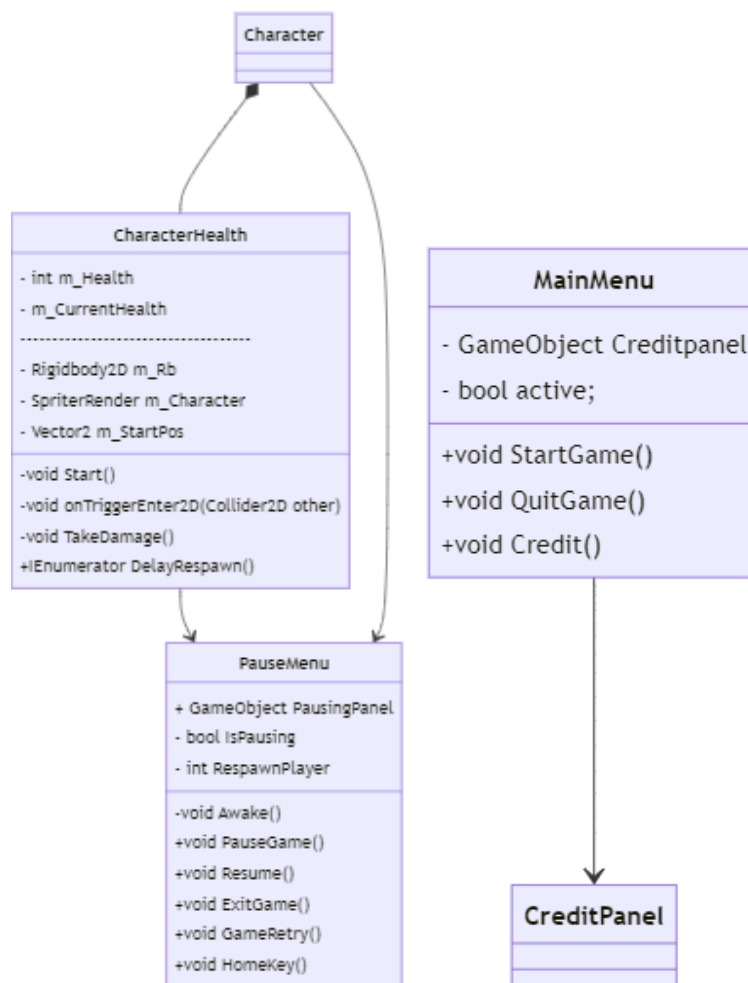
UI Menu

- Player will get 3 types of UI menu, main menu to start and quit game plus check on credit of this game. Pausing menu in scene to pause game and option quit or resume game. And Game over menu when player health reach 0 and choose to restart or not.
- 3 types of menus all require UI button related function to perform action. Load game scene, quit application and respawning.



Game Over Menu





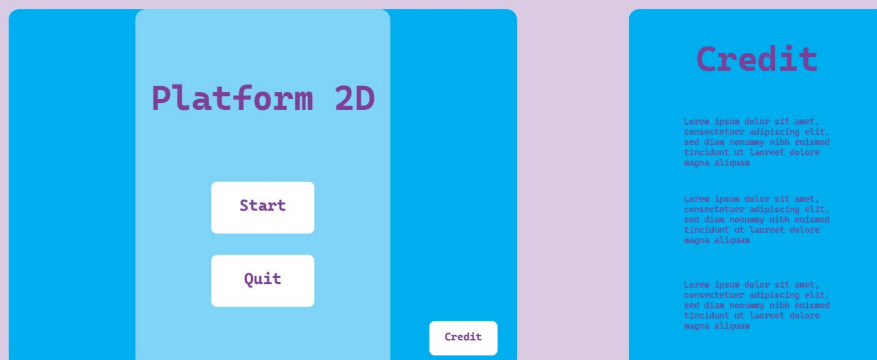
GameOver Menu

Once player health reach 0, game over menu will pop up to announce player
Allow player to choose return to Home scene, quit the game or restart the stage.



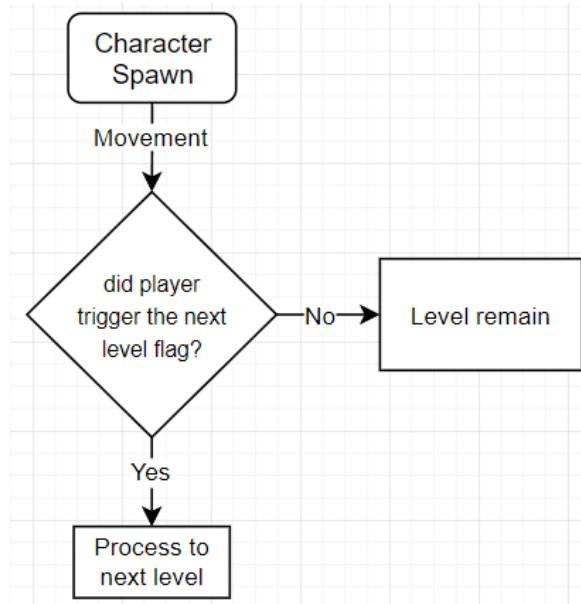
GameOver Menu

Main menu allow player to start game or quit game
Credit button will display the production detail of the game



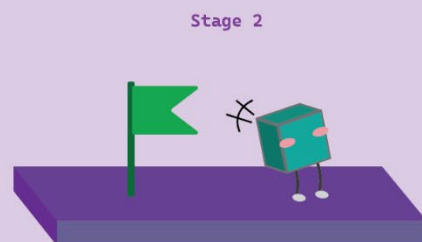
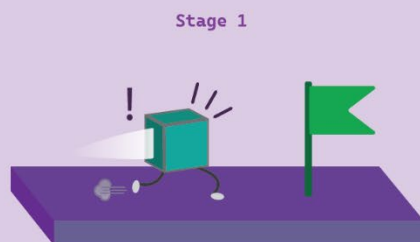
Level Transfer

- Once the player triggers a flag in the game, player will transfer to a new level.
- A Flag manage script to load a set scene from engine and call it by adding index to open new scene, this could be benefit if developer hope to add on new scene and no need to create a new script for loading purpose.



Next Level

When player walk to the flag, it will trigger the stage transfer mechanic
Player will transfer to a new scene of the game
Character will be a new character but contain the old mechanic and input



Post Process Effect

- The game will include post process effect to more the game look chromatic like an old 16 bit arcade pixel game.

Audio, SFX and Particle Effect

- This project will include background music in main menu and game level itself.
- Some SFX will include when action been approach, like GEM obtain sound, damage sound and game over sound.
- All audio and SFX will be control from an audio manager script and call out audio source from different scripts. This could benefit developer to view general audio and assign audio from Unity inspector.
- Particle effects are similar to the SFX, play when player trigger or make interaction.

