Paradox Reflection - Technical Design Document

Paradox Interactive *Advanced Diploma of Professional Game Development Major Project*

Software

- Windows 7 Enterprise (64 Bit)
- Unreal Engine 4 (64 Bit)
- Source Tree

Third Party Libraries

No Third party libraries or code will be used for the development of the game.

Audio

Music and sounds will be provided by one of the game designers.

File Formats: wav

Graphics

Scale Ratio: 1 : 1
Resolutions: 1024 x 1024p
Bit depth: 32 bit
File formats: fbx, obj
Compression: tiff, png

Game Mechanics

Core Mechanics

Time Travel

The player can travel between time using static portals

- Time Portal Mirror
  These are used to switch between the two maps representing the present and past, they show the same room from the other time in the mirror's reflection

- Time Period Feedback
  The player can see the other area in the mirrors reflection which rotates to follow the player to make it look more like a mirror

- Paradoxes

Inventory

The player has an inventory that contains 5 slots for usable objects and a journal for 5 notes

- Class contains a static array for items and another for notes and each stores uint8's (bytes)
- UE4 blueprint Enums are used for switching between the images for the inventory screen and also used for setting what an item specifically is:
  - InventoryItems
  - JournalNotes
  - ImageSlots

Using Objects

The player can pick up and interact with objects in the game. Raycasting is used to find an object the player is looking at and calls a function for usableobjects for its interaction.

Other Mechanics

Puzzles

Foyer Present Mirror Repair

The portal to the past is broken and the player needs to repair it

- Remove the last of the mirror glass using a hammer
- Place a fresh mirror glass
- Use chalk to power the portal

Foyer Past Braziers

The player needs to set the colour of braziers to the correct combination to get a key to unlock the study door in the present.

- Enums used to define colors in code
- The player needs to get the brazier combination from a safe and a fuse
- The safe holds the combination and the entered numbers, which are stored as int arrays and using a UI screen the player can change the entered numbers
- To turn on the braziers a power box near the power switch needs to be fixed with the fuse
- The player gets the key from a strongbox that opens when the puzzle is complete

Study Piano

The player needs to find piano keys that are missing from the piano and then tune it. The Player then needs to play a combination of music pieces which are found in the bookcases in the past which then open a bookcase.

Playing notes on the piano is done by clicking buttons in a UI that calls functions from the piano which is found based on its class as it should be the only piano in game. The piano in the past of the game should be just an art asset.

The state of tuning is stored as an enum for which note the player needs to press, a switch statement then checks if the player played the correct note and then acts accordingly.

For playing, the piano stores the notes for the song in 3 seperate arrays and uses variables to store which notes the player should be playing next. The function checks the note the player played against the note it needs to play then checks if they are at the next phase of the song.

User Interface

UI widgets use enums to set the textures of image widgets. They all check the players inventory for their items and notes. The player selects an item and reads notes by clicking an invisible button on top of the item icon. If the inventory screen is open when the player enters the pause screen the inventory screen will close and the pause menu will appear.

Monster

The game has a monster that is impaled to the study door in the past of the foyer

- The monster slowly breaks free as the player progresses through the brazier puzzle
    - Getting the hint from the safe
    - Powering the braziers
    - Setting the braziers to the right colors

When the player has finished the puzzle a timer starts and the player must complete the piano puzzle before the monster breaks through the past study door, if the player doesn't complete the puzzle they die, restarting the game.

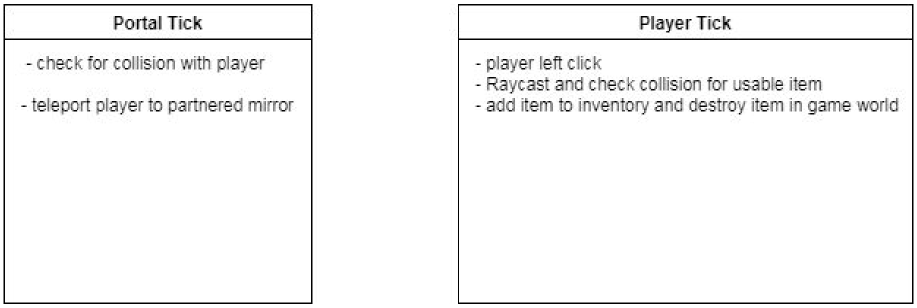
Technical Design

Platforms

- Microsoft Windows

Control Loop

- Update Loop



Portal Tick

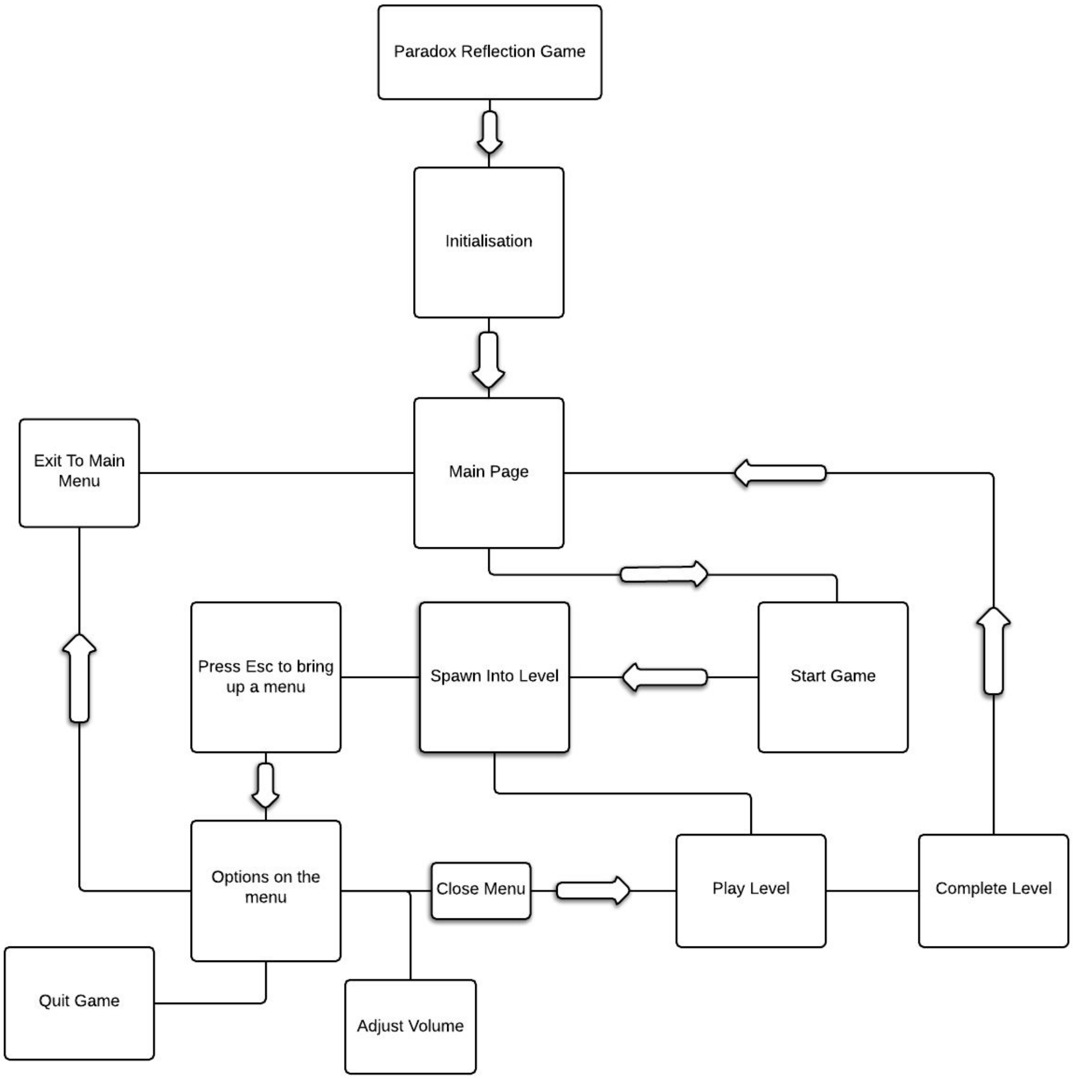
- check for collision with player

- teleport player to partnered mirror

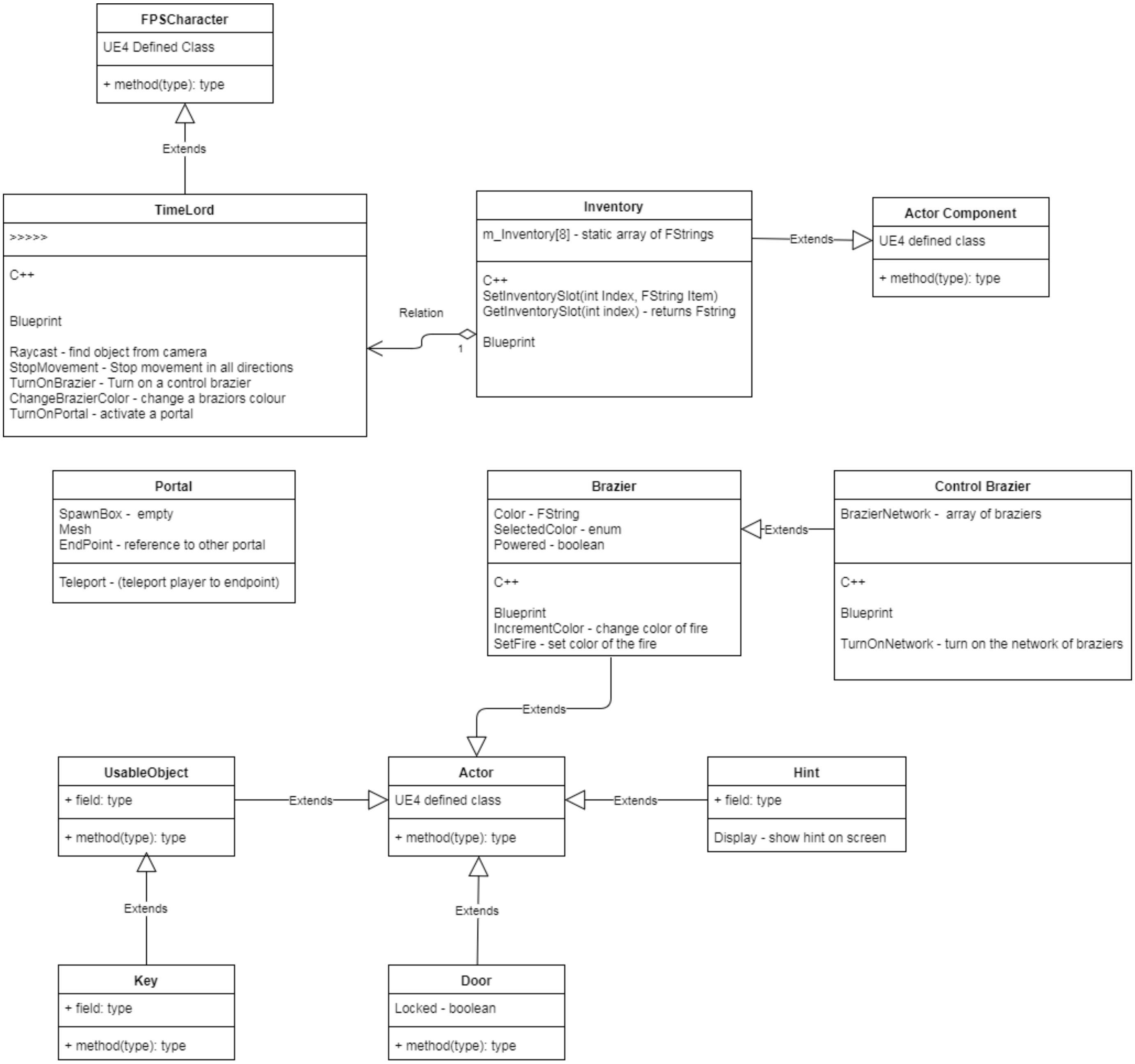
Player Tick

- player left click
- Raycast and check collision for usable item
- add item to inventory and destroy item in game world

- Screenflow


Paradox Reflection Game
Initialisation
Main Page
Exit To Main Menu
Press Esc to bring up a menu
Spawn Into Level
Start Game
Options on the menu
Close Menu
Play Level
Complete Level
Quit Game
Adjust Volume

Game Object Data



FPSCharacter

UE4 Defined Class

+ method(type): type

TimeLord

>>>>>

C++

Blueprint

Raycast - find object from camera
StopMovement - Stop movement in all directions
TurnOnBrazier - Turn on a control brazier
ChangeBrazierColor - change a braziors colour
TurnOnPortal - activate a portal

Inventory

m_Inventory[8] - static array of FStrings

C++
SetInventorySlot(int Index, FString Item)
GetInventorySlot(int index) - returns Fstring

Blueprint

Actor Component

UE4 defined class

+ method(type): type

Portal

SpawnBox - empty
Mesh
EndPoint - reference to other portal

Teleport - (teleport player to endpoint)

Brazier

Color - FString
SelectedColor - enum
Powered - boolean

C++

Blueprint
IncrementColor - change color of fire
SetFire - set color of the fire

Control Brazier

BrazierNetwork - array of braziers

C++

Blueprint

TurnOnNetwork - turn on the network of braziers

UsableObject

+ field: type

+ method(type): type

Actor

UE4 defined class

+ method(type): type

Hint

+ field: type

Display - show hint on screen

Key

+ field: type

+ method(type): type

Door

Locked - boolean

+ method(type): type

Data Flow

- Currently there is no data being stored in secondary memory.

Game Physics and Statistics

- Unreal Engine will provide all physics needed for the game
- The map will have two areas for representing the past and the present, so for each art asset there will be a second