****

***Slimy Doodly***

**Technical Specification Outline**

**Team *Dislocation***

***Producer : Haneul Lee  
Game Designer : Hoseob Jeong***

***Technical Director : Yoonki Kim***

# 

# T**able of Contents**

[**Technical Specification Structure**](#_ayk2fhpzjpxa) **4**

[**Architecture Overview**](#_6ieo6u31ue05) **4**

[Graphics Implementation](#_jairja37jz7e) 4

[Sprite Sheet](#_koxqewelizn4) 4

[Animation](#_a4vcjfv02d8p) 4

[BackGround](#_nu928viss66g) 4

[Physics Implementation](#_pnz15yrsnkgs) 4

[Collision Check](#_2kljq2ihoc90) 4

[Map And Objects Collision](#_xyd965qrugco) 5

[Physics Object Behavior](#_g99epa51nubg) 5

[Player Controls Implementation](#_lhfpfjpzfl84) 5

[Behavior Implementation](#_mvo643bsnjja) 5

[Slime Movement Implementation](#_bx45uk8eqhrq) 5

[Naming Convention](#_v755g9ry70m) 5

[Extension Names](#_ndkq03xg524z) 6

[Debugging](#_k9d281995rwl) 6

[Technical Risks](#_skd6syz1getb) 6

[Map and Physics Object’s Collision](#_mqt71bstyls2) 6

[**Appendices**](#_bjpwl659lspo) **7**

[Appendix A: Interface Flow](#_xvipforlkqeq) 7

[Appendix B: World Class info and Hierarchy](#_prye0fvez5qb) 7

[Appendix C: GameObject Class info and Hierarchy](#_r0jce9bknybg) 8

[Appendix D: Tile (Map) Manager Class info and Hierarchy](#_10xoh9lw862n) 9

[Appendix E: Camera Movement Method](#_a5u6tw5h5o9s) 10

# Technical Specification Structure

## Architecture Overview



**<How StateManager and World Class communicate>**

**<Game Overflow>**

## Graphics Implementation

### Sprite Sheet

### Animation

### BackGround

## Physics Implementation

### Collision Check

#### AABB Collision

* Used for collision between object & object (Object can be player, physics and item or something)

#### Line and Rectangle Collision (Using Line and Line Intersect)

* Only used for wavegun and object collision

### Map And Objects Collision

#### Collision Points

Each Physics Object has multiple collision points to manage precise map and physics object collision.

### Physics Object Behavior

#### Gravity

1. All the physics objects are under gravity’s low when it is not on ground.

#### Drag

1. If Object is on the ground, drag acceleration will reduce the object's speed.

#### Wave Interaction Behavior

1. Physics Objects are all applied by wavegun. It uses line and line intersection to detect when an object is hit by a wave. If object is hit, wave’s vector are applied to hitten object’s velocity

## ***Player Controls Implementation***

### KeyBoard

#### In - Menu

1. Select Menus - W, A, S, D
2. Enter Menu - SpaceBar

#### In - Game

1. Player Basic Movement Keys - W, A, S, D
2. Jump Key - SpaceBar
3. Level Reload - ‘R’ Key

### Mouse

#### In - Game

1. Aim the wavegun - move mouse cursor
2. Shoot - RightButton

## ***Behavior Implementation***

### Slime Movement Implementation

#### *Script Based System*

1. Script Base Movement (use our own .scp file)

***Coding Methods***

### Naming Convention

File Name : SomeFile.xxx

Class Name : SomeClass

Function Name : SomeFunction

Variable Name : someVariable

Flag Name : isFlag, canFlag (only use is or can)

### Extension Names

#### Image

1. .png : Sprite Sheet Image File
2. .spt : Sprite Information script File
3. .anm : Animation Information script File

#### Sound

1. .wav : BackGround Music and sound effect sound file
2. .sef : Sound Information script File

* Information about how long sound will be played
* Which order for specific sound effect will be played

(sample)ShootEffect.sef

* ShootSound 0.5 second
* WaveGondSound 0.2 second
* ...

## ***Debugging***

* From CS230 Engine Logger
* Having 4 different severity level (Verbose, Debug, Error, Event)
* In debug mode output comes out to console, on release mode output will come out to .txt file.

## ***Technical Risks***

### Map and Physics Object Collision

How can we solve tunneling effect

How to know what tile the object will be collided

What if object collides with multiple objects

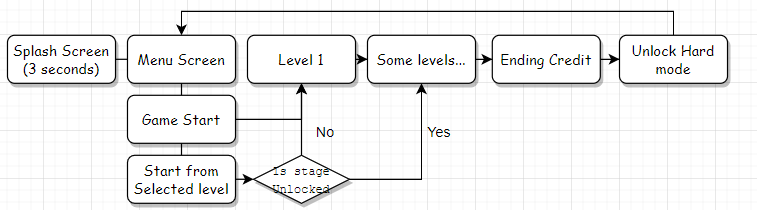
### Script Based Slime Movement

How to move the slime by timer.

How to make the slime move everytime.

# **Appendices**

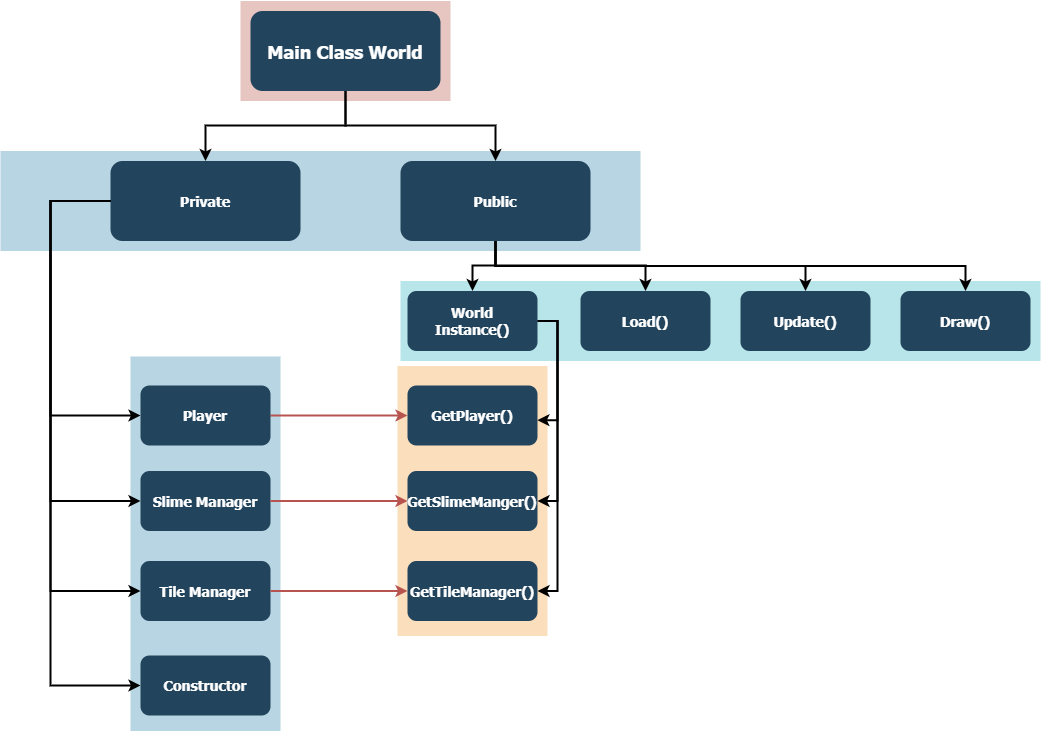
## ***Appendix A: Interface Flow***



## ***Appendix B: World Class info and Hierarchy***

#### What is World Class Means?

World class is using singleton methods, so World class can manage every class or some object managers(like tile manager for making maps by tile or the slime manager for store slimes which were used for prototype).



In World class, there are Player class, SlimeManager class, TileManager class by its own private member variable. In addition, to apply singleton methods, it has an instance of World class by Instance() function by returning a World reference for its return type. And load instance’s member variables by GetPlayer() function, GetSlimeManager() function, GetTileManager() function. By loading member variables like this method, we can change, update, delete these classes.

## ***Appendix C: GameObject Class info and Hierarchy***



**<GameObject Class Overview>**

#### What Is GameObject Class?

GameObject Class is a base class for all the game objects, it is managed by a simple manager class called GameObjectManager has simple add, update all, draw all and unload all functions.

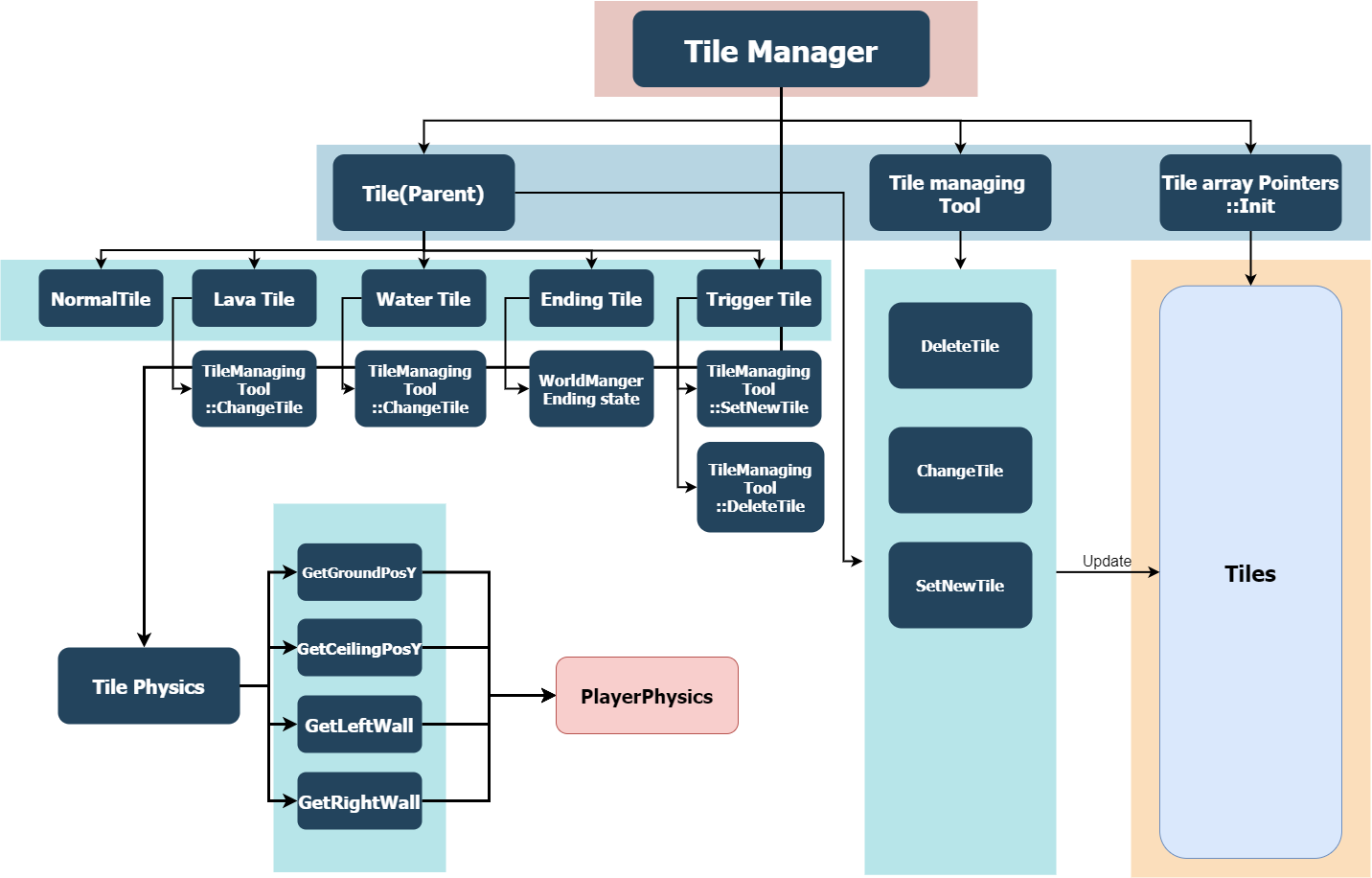


**<GameObject and its’ child classes hierarchy relationship>**

Player class has wavegun class, wavegun class, player update controls wavegun’s update and draw function too.

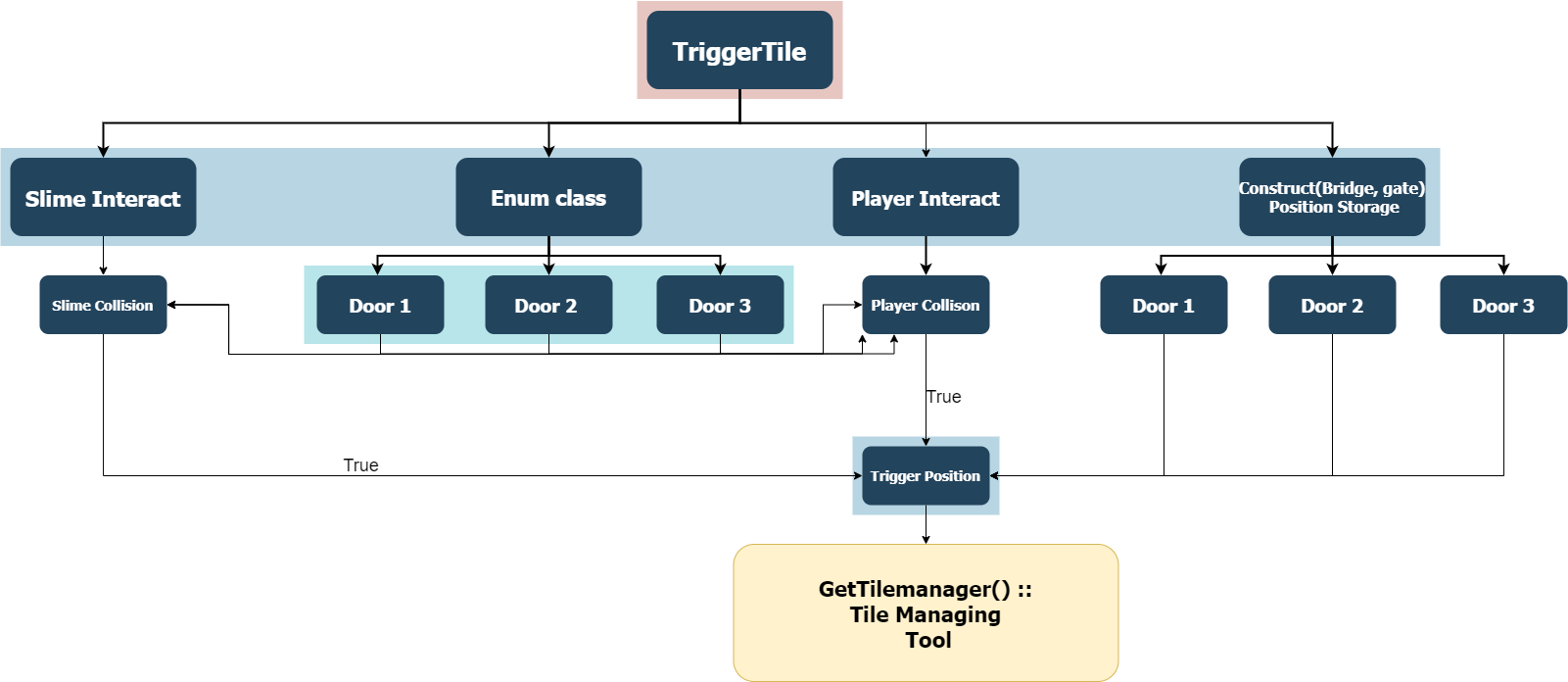
## 

## ***Appendix D: Tile*** (***Map) Manager Class info and Hierarchy***



**<Tile(Map) Manager Info>**

## Tile manager class is managing the tile array and interacting with tile and player. Tile manager included the Tile class array pointer of private members. Tile class is a parent class about Normal, Lava, Water, Ending, and Trigger tile. First, Init to put the array to make tiles. and using the Tilemanging tool to interact with the player and tile. and then, adapt(Update) to Tile array pointers.



**<TriggerTile Info>**

Trigger Tile class has Constructing(make a bridge or gate) storage, it will be script. So if slime or a player interact tile, Get TriggerPosition and build the bridge by using the Tile Managing tool. It is similar to the other Tiles.

## Appendix E: Camera Movement Method



**<Camera logic Info>**

Camera will have a frame if the player moves in the frame. Camera will not be moved. But if the player wants to move out of the frame. the camera will follow the area.