

Minxu Huang 101074668
Puraj Desai 101093244
John Paul Valdez 101055854

All Members must in agreement to:

The coding style:

- Bracing style
- Naming style
- Comment types

Project structure:

-Docs :

- TDD(technical design document),
- GDD(GAme design document),
- Test plans.

-Assets:

- All arts
- Sound engineer
- Designer files

-Source:

- Codes and subfolders with all .cpp and .h files.

-Temp:

- All temp files.

-Test:

- Test script
- Files to unlock cheats.
- Log files for warning, error, diagnostics.
- release notes for builds.

-Game:

- Executable and DLL files.
- Game data that required to run the game.
- Game data save files.

CPU: Any

GPU: Intel graphic hd 520.

RAM: 500MB or higher.

Storage: 100 MB or more.

Minxu Haung: "Okay."

John Paul Valdez: "I'm fine with this."

Puraj Desai: "I agree."

Minxu Huang 101074668
Puraj Desai 101093244
John Paul Valdez 101055854

Component Architecture description:

In the diagram below we have the actors such as player, enemy, cubeWall, hide area, steel, and UI.

The player has components such as projectiles which will act as bullets the player will shoot at enemies. Collider is for detecting if it's hit by an enemy bullet or collides with the walls. Mesh renders for texture or color. Transform for position and rotation. HP is the health of the player which will be 1 so if the player is hit even once, then it is gameover.

Enemies have the same components as the players but differs since it will eventually include an Ai in oppose to being controlled manually like the player.

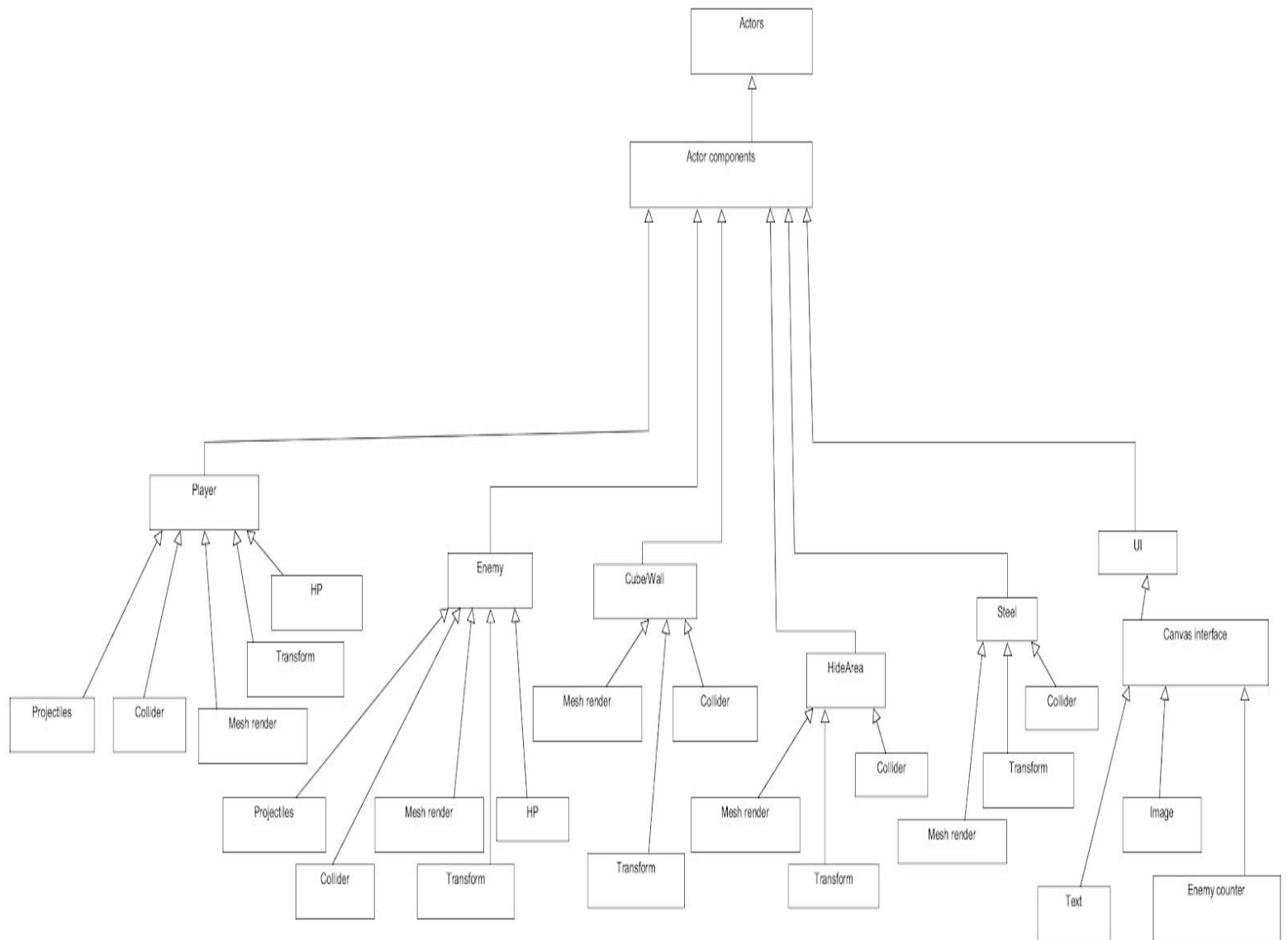
CubeWalls have colliders to detect the player, enemy, and projectiles. If the wall is hit by projectiles, part of the wall will be destroyed. Mesh renders for texture and color, transform for position.

Hide area is very similar to cubeWalls but the only difference is that it is an onTrigger. So if an enemy or player is inside of it then they can't see or detect each other. Also, projectiles pass through it.

Steel have the same thing as the cubewall but the only difference is that no matter how many times the projectiles hit the steel it will not be destroyed.

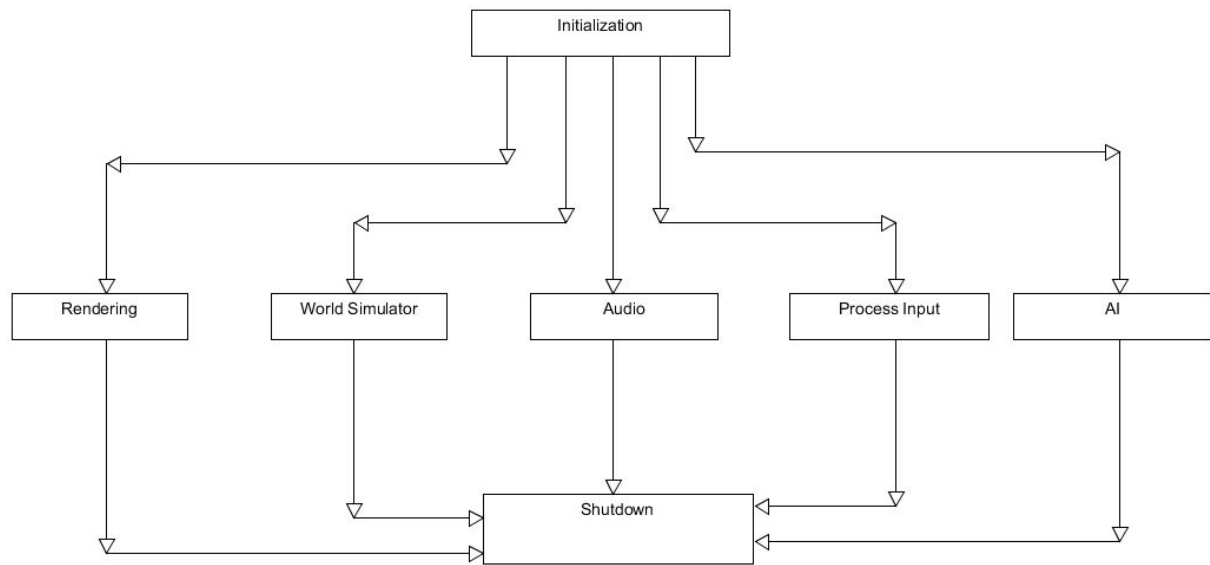
UI has a canvas component. Inside the canvas component is a text/log component to display texts on the screen, image component for icons and other visuals, and finally an enemy counter component to let the player know how many enemies are left to destroy.

Class Structure:



Minxu Huang 101074668
Puraj Desai 101093244
John Paul Valdez 101055854

Main loop Structure:



Pseudocode:

MainLoop:

```
begin ();  
InitialStage()  
{  
    Checksystemresource(){  
        hardDriveSpace();  
        memory();  
        inAndOutput();};  
    CPUspeed();  
    memorycache();  
    Createwindow();  
    audioSystem();  
    Loadsavedfiles();  
    CreateSurface();  
    physicsAIAndMore();  
};
```

```
render()  
{  
    PlayAudio();  
    GetInputs();AI();
```

Minxu Huang 101074668
Puraj Desai 101093244
John Paul Valdez 101055854

```
};  
LoadScene()  
{  
    Player.Draw;  
    Enemies.Draw;  
    World.Draw;  
};  
PlayAudio()  
{  
    BackgroundMusic.play;  
};  
GetInputs()  
{  
    if(input.GetKey(...))  
        Do something;  
};  
AI()  
{  
    enemy.FindPath;  
    enemy.ShootProjectile;  
};  
ShutDown();
```