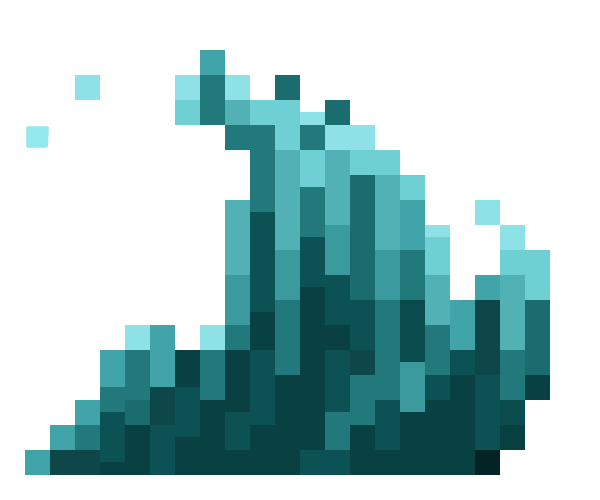
Beta Milestone Individual Report

Looter

by

Neoul



GAM 250

6.12.2021

Spring

UI Designer & Programmer & Test Manager: Junseok Yang

MILESTONE : BETA MILESTONE

TEAM NAME : NEOUL

GAME NAME : LOOTER

NAME : YANG JUNSEOK (JUNSEOK.YANG)

JOBS : UI PROGRAMMER & TEST MANAGER

DEGREE PROGRAM : RTIS

GAME DESCRIPTION : It's a top-down pixel game like a common 2D game. The player is a shooting game that can attack close and far.

Use live bullets to hit various enemies, level clear, and use various weapons.

Lastly, as you can see from the image next to it, the map is big and the user can adventure through the map and get treasure.

LINKS :

\* Discord: <https://discord.com/channels/750662266185187369/>

 \* GitHub Repository: <https://github.com/Rudy-Castan-DigiPen-Teaching/game-project-neoul>

 \* Neoul Check: <https://docs.google.com/spreadsheets/d/1mKUEGqaMvN1M3C0BoOvyjI7-iCze5fzsIkBg-NevsJY/edit#gid=1862655681>

\* Neoul Trailer:

<https://drive.google.com/file/d/1PvBDdR4mIOVm956Cm0aN8yH4E5bx1rlA/view?usp=sharing>

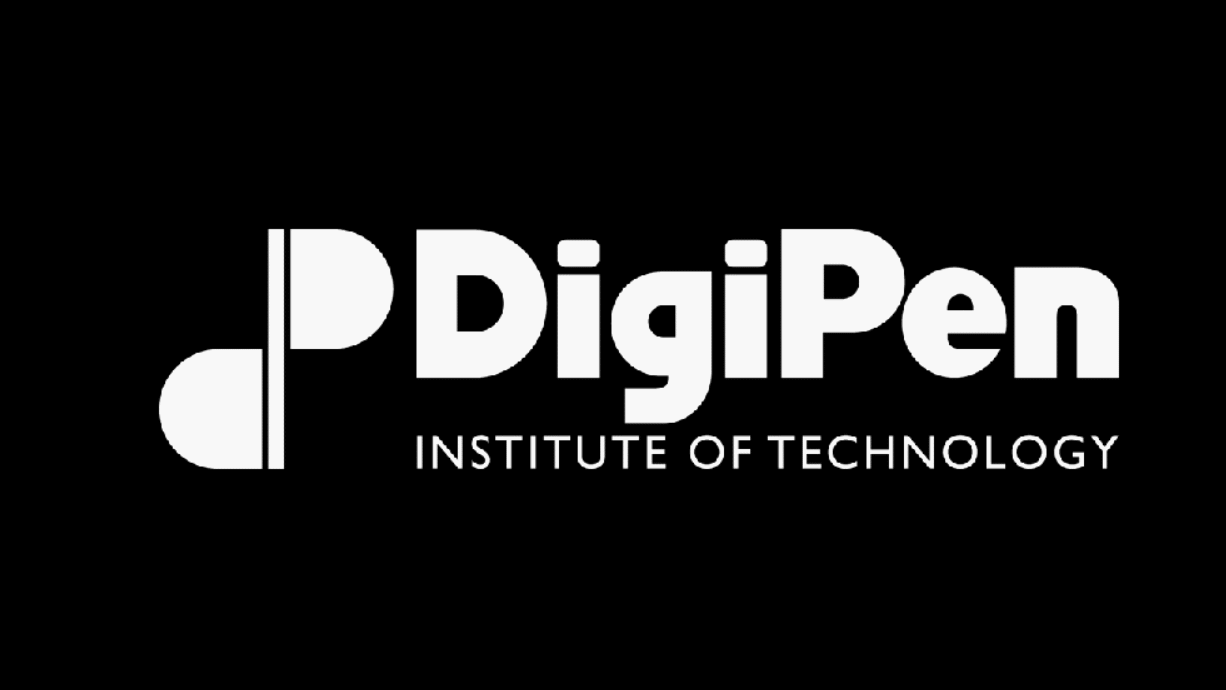
WORK DONE

CODING

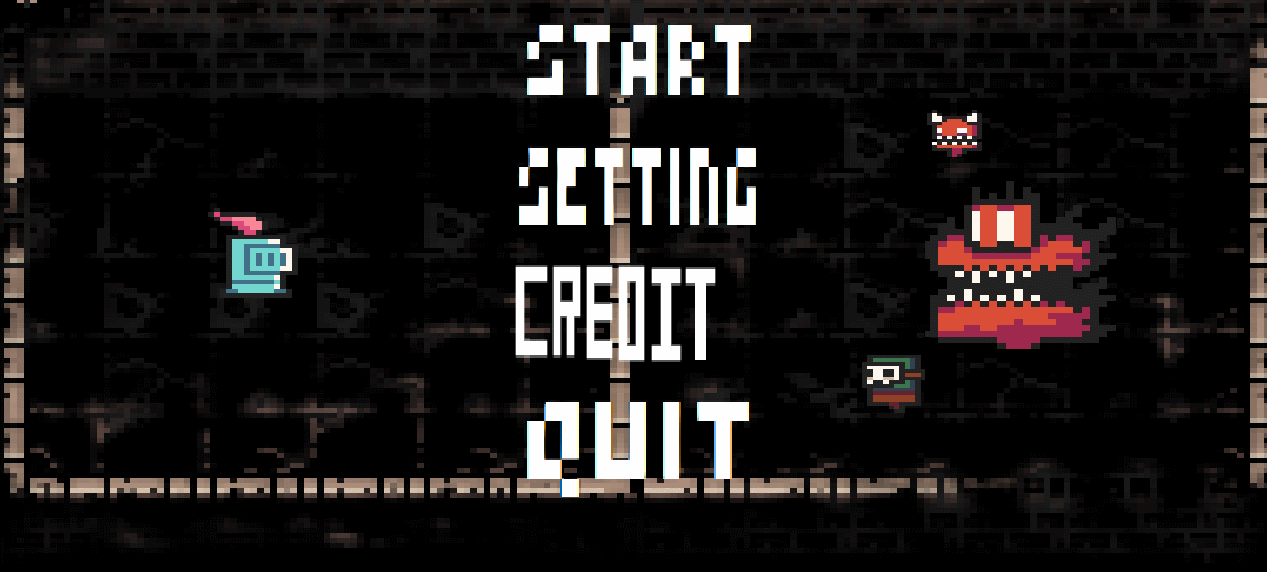
I was in charge of SCENE, UI, and HUD. First of all, we focused on menus, options, credits, splashes, level optimization, and game balance. The menu implements animation centric start, credit, quit, and options include advanced window screen window size control and sound control. You can also press the Start button to enter the game and restart it. And Splash puts various logos on the menu. The credit screen shows all the information required by Digipen. The UI has multiple UIs, including bullets, weapons, level board, and health. There are several HUDs in the HUD, including operation. I also did some emotions. Status and bullet points are UI changes due to player properties. Key HUD describes keys that can be used instead of tutorials. The Weapon UI tells you what kind of weapon the player currently has. The Level board UI also tells you the level of the current player.

**SCENES**

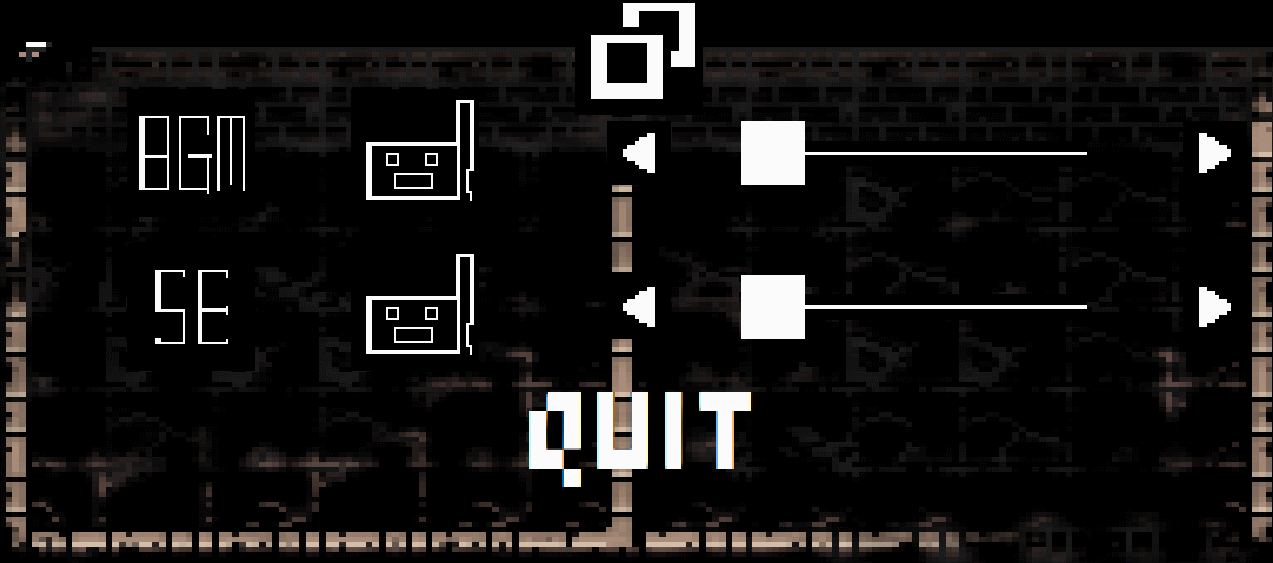
create splash scene (3hours)



Create and animate high quality menus (12hours)

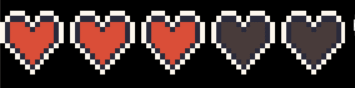


create setting scene (8 hours)



**UI**

Create advanced Health UI (10 hours)



Create advanced Bullet UI (15 hours)



Create advanced Weapon UI (5 hours)



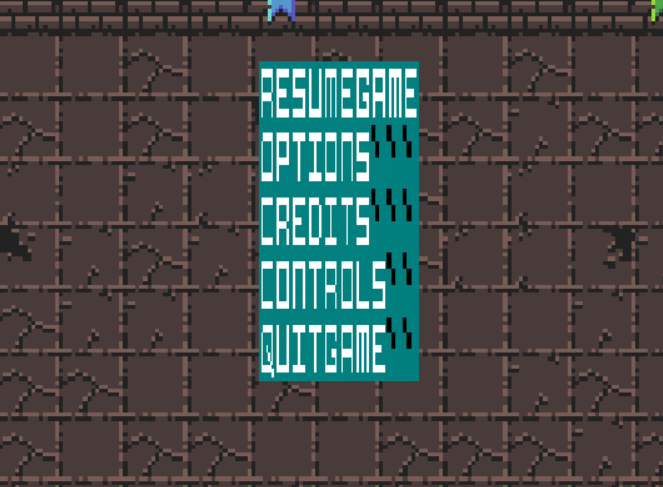
Create advanced Key HUD (8 hours)

Create advanced Level board UI(3 hours)



Create advanced pause UI (15 hours)



**others**

advenced cam shake (3 hours) update level 1 2 3(20 hours)

connecting menu & credit(2 hours) make some emotion(5 hours)

play testing work (2 hours) make credit sprite(4 hours)

regulate game balance (3 hours) adding esc button (4 hours)

ACTIVE FILE

Player.cpp (10 hours) – Common file

Player.h

Level1.cpp (10 hours) – own activity file (1061 lines)

Level1.h

Level2.cpp (10 hours) – own activity file (845 lines)

Level2.h

Level3.cpp (10 hours) – own activity file (838 lines)

Level3.h

Menu.cpp (8 hours) – own activity file (291 lines)

Menu.h

Option.cpp (10 hours) – own activity file (532 lines)

Option.h

Pause.cpp (2 hours) – own activity file (201 lines)

Pause.h

Splash.cpp (3 hours) – own activity file (87 lines)

Splash.h

BulletUI.cpp (7 hours) – own activity file (30 lines)

BulletUI.h

ExitUI.cpp (3 hours) – own activity file (30lines)

ExitUI.h

HealthUI.cpp (8 hours) – own activity file (30 lines)

HealthUI.h

KeyUI.cpp (10 hours) – own activity file (110 lines)

KeyUI.h

LevelUI.cpp (2 hours) – own activity file (30 lines)

LevelUI.h

OptionUI.cpp (10 hours) – own activity file (66 lines)

OptionUI.h

WeaponUI.cpp (2 hours) – own activity file (33 lines)

WeaponUI.h

SUB ACTIVE FILE

Lasergun.cpp (5 hours) – sub activity file (90 lines)

Lasergun.h

Pistol.cpp (5 hours) – sub activity file (90 lines)

Pistol.h

Shotgun.cpp (5 hours) – sub activity file (135 lines)

Shotgun.h

MissileLauncher.cpp (1 hours) – sub activity file (85 lines)

MissileLauncher.h

UpgradeLaser.cpp (1 hours) – sub activity file (135 lines)

UpgradeLaser.h

Game.cpp (2 hours) – Common file

GameConfig.h – Common file

GameEngine.h – Common file

Types.h – Common file

GameMath.h – Common file

Run\_game.cpp (0.5 hours) – Common file

Win\_main.cpp (0.1 hours) – Common file

//\*\* These lines include annotated lines and blank lines

DEBUG

I had to go far several times while using the sauce tree.  
Whenever a link error or DLL LIB error occurred, it took a lot of time to fix it.

TEAM

After Alpha, our team was forced to cancel weekly school meetings due to Cvoid-19. So, little by little, the frequency of team meetings and smooth communication decreased, and of course, there were things that I couldn't do better than the planning and design I wanted. If the team members had smooth communication, they would have been able to play better. What the team felt was that they checked the list of submissions only after the last submission, and they first submitted it, but regret not preparing it in advance.

NOTES

During this game project, I worked hard and researched on the basic game level structure and how to give good and lots of feedback on state, UI, and HUD. Among them, I thought there were many better ways to do it when I changed the UI after receiving feedback on the play test. If I have more time, I am confident that I will be able to make the level cleaner and make the UI and HUD prettier. In the meantime, I only wanted to know about graphics, game status, and concepts of things. However, after this project, I realized the importance of UI, HUD, and the importance of play test & log. And I realized how important communication between team members is.