Swinburne University of Technology

COS20007 Object Oriented Programming

Research Project Initial Plan

PDF generated at 00:08 on Wednesday 17th May, 2023

HD Research Project Initial Plan

Name: Trung Kien Nguyen

Student ID 104053642

Overview of my project

My idea for the HD research project for the unit COS20007 is concentrated on the application of OOP principles in one of the most popular and powerful Game Engines in the world, which is Unity.

The aim of this project is to demonstrate, discuss, and evaluate the benefits (and potential constraints) of using OOP principles, including the four keyword: abstraction, inheritance, encapsulation, and polymorphism, as well as SOLID principles and other OO aspects, in a famous game engine of Unity. The project will not only involve in the OO default components of Unity, but also look at an example of my 2D top-down game (prototype) using OOP to in the codebase. After finishing this research project, I expect to understand more about the advantages, as well as improve the skills of using OOP principles in game development, especially in Unity projects.

Reflection in the unit's learing outcomes.

1. This project is to discuss how OOP's principles such as abstraction, inheritance, encapsulation, and polymorphism apply in game development, specially in Unity game engine, and also how they can be used to improve the codebase's readability, efficiency, reusability, and maintainability.

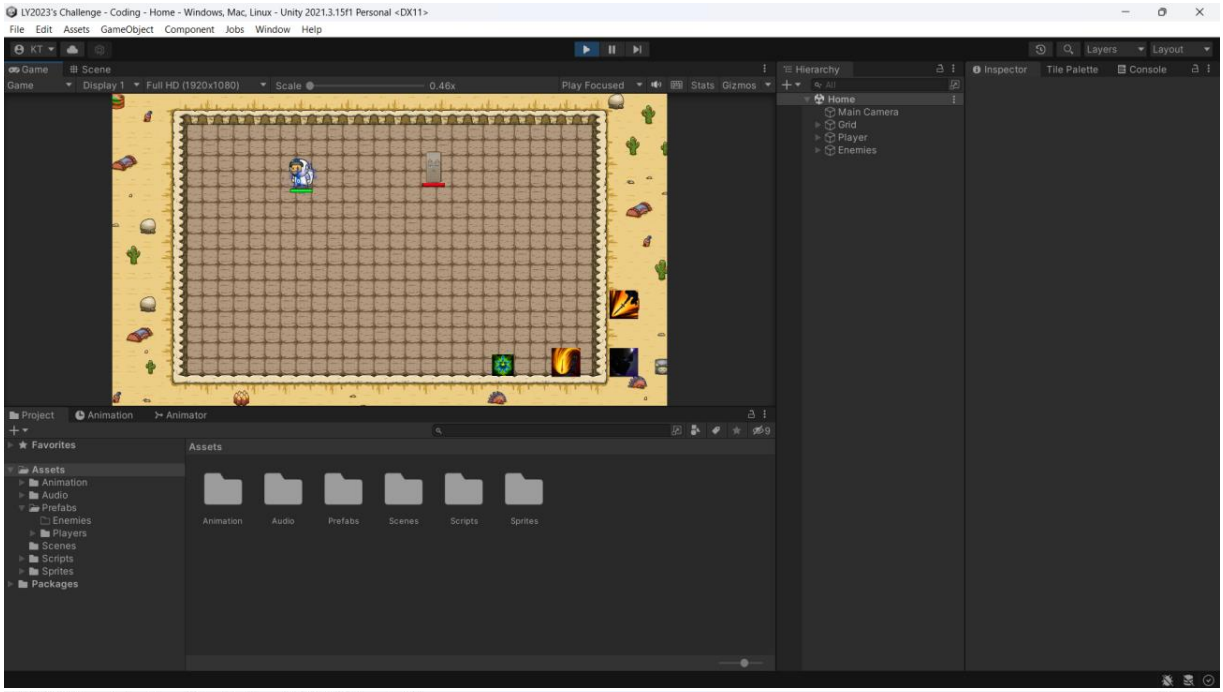
2. In my program used for this project, obviously I have used a high-level Object-oriented programming language – C#, which helps me to present OOP's principles in my codebase more easily.

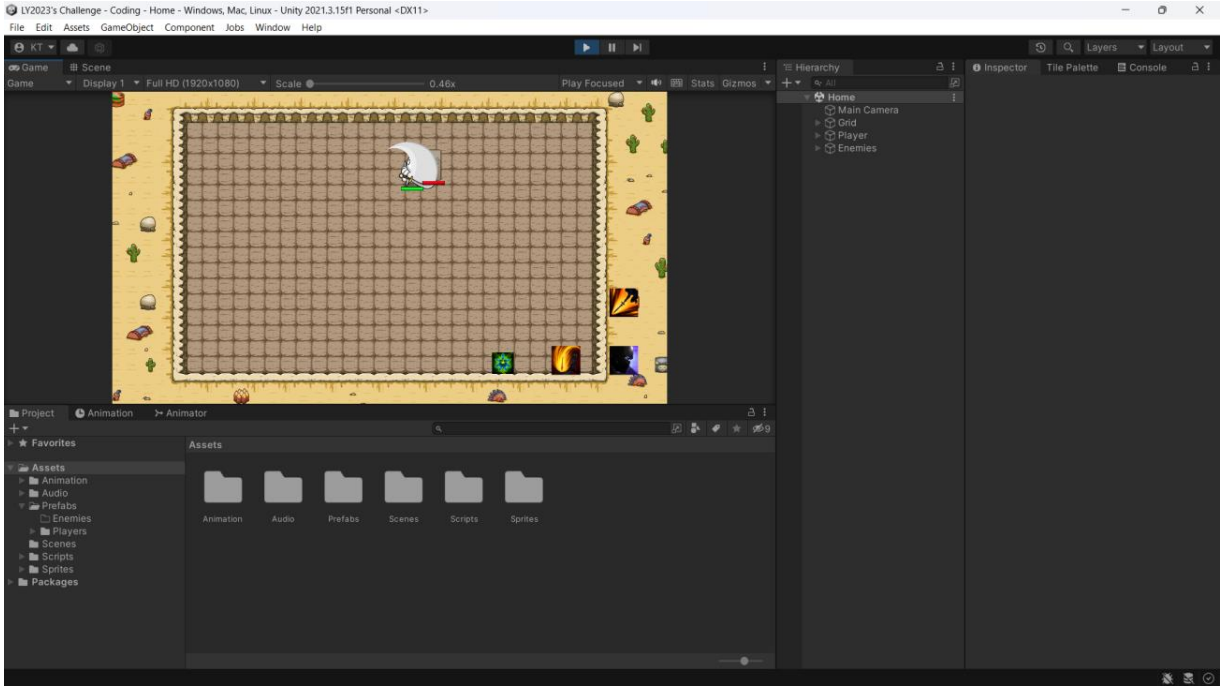
3. With the debug system of the Unity game engine, it is possible for me to design, develop, test programs using OOP principles.

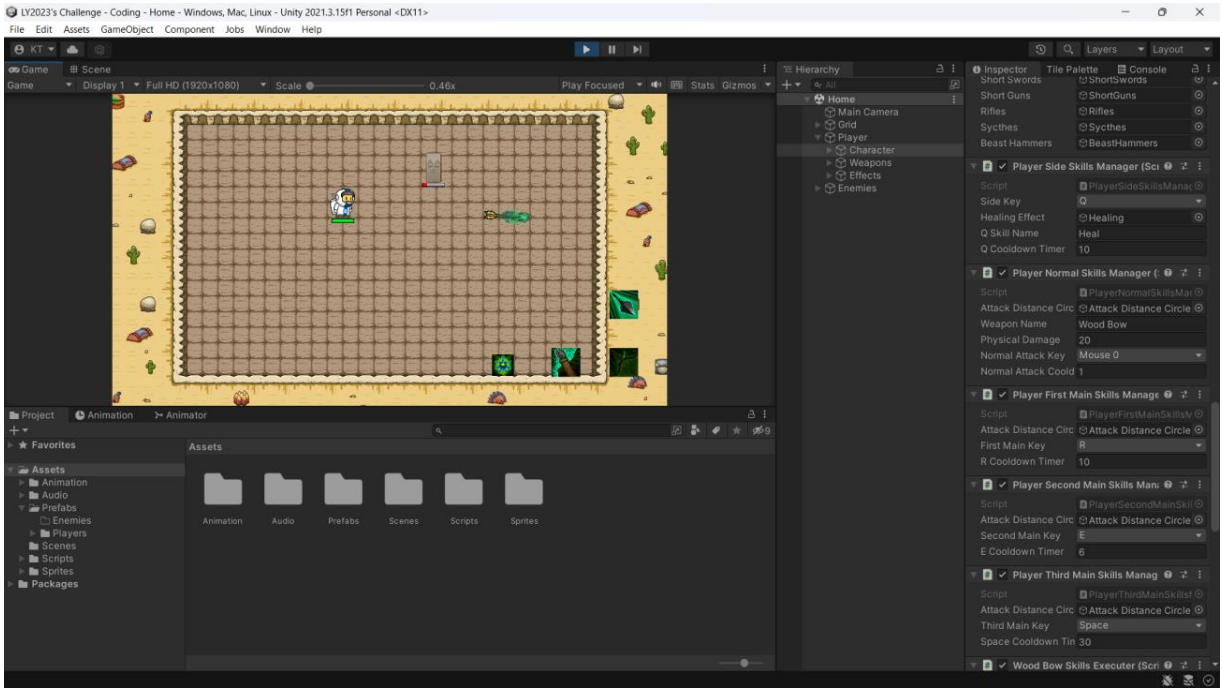
4. Of course, the diagrams and illustrations are necessary for all object-oriented programs or projects, including my example Unity programs, and will be presented in this research project

5. Conducting this project may requires me to describe and explain the factors that contribute to a good OOP solution, reflecting on my own experiences, skills and drawing upon accepted good practices, which are illustrated through my Unity program. This helps me meet the intended learning outcomes of the unit.

Some screenshots of my Unity prototype program:


LY2023's Challenge - Coding - Home - Windows, Mac, Linux - Unity 2021.3.15f1 Personal <DX11>
File Edit Assets GameObject Component Jobs Window Help
Game Scene
Display 1 Full HD (1920x1080) Scale 0.46x Play Focused Stats Gizmos
Hierarchy
Home
Main Camera
Grid
Player
Enemies
Inspector Tile Palette Console
Project Animation Animator
Favorites
Assets
Animation
Audio
Prefabs
Enemies
Players
Scenes
Scripts
Sprites
Packages


LY2023's Challenge - Coding - Home - Windows, Mac, Linux - Unity 2021.3.15f1 Personal <DX11>
File Edit Assets GameObject Component Jobs Window Help
Game Scene
Display 1 Full HD (1920x1080) Scale 0.46x Play Focused Stats Gizmos
Hierarchy
Home
Main Camera
Grid
Player
Enemies
Inspector Tile Palette Console
Project Animation Animator
Favorites
Assets
Animation
Audio
Prefabs
Enemies
Players
Scenes
Scripts
Sprites
Packages


LY2023's Challenge - Coding - Home - Windows, Mac, Linux - Unity 2021.3.15f1 Personal <DX11>
File Edit Assets GameObject Component Jobs Window Help
Hierarchy
Home
Main Camera
Grid
Player
Character
Weapons
Effects
Enemies
Inspector
Tile Palette
Console
Short Swords
Short Guns
Rifles
Sycthes
Beast Hammers
Player Side Skills Manager (Scr
Side Key
Healing Effect
Q Skill Name
Q Cooldown Timer
Player Normal Skills Manager (
Attack Distance Circ
Weapon Name
Physical Damage
Normal Attack Key
Normal Attack Coold
Player First Main Skills Manage
First Main Key
R Cooldown Timer
Player Second Main Skills Mana
Second Main Key
E Cooldown Timer
Player Third Main Skills Manag
Third Main Key
Space Cooldown Tin
Wood Bow Skills Executer (Scri
Project
Animation
Animator
Favorites
Assets
Animation
Audio
Prefabs
Enemies
Players
Scenes
Scripts
Sprites
Packages