Tank Shooting Based on Unity3D

G. Wang

Supervisor: Dr. J. Zhao

Computer Science and Technology Program, Division of Science and Technology, BNU-HKBU United International College

** Corresponding student author. Tel: +86-131-6961-4284, E-mail: j430003024@mail.uic.edu.hk*

Abstract

In this project, we created a multi-player 3D shooting game based on Unity. We provide two means for players to be connected and start a game: through an online game server or through direct connection based on TCP-IP. In the game, each player controls a tank. The goal is to aim and shoot the opponent's tank while moving their own tank to evade bullets. Health points will be deducted if a tank receives damage and if a tank's health points drop to zero, the other player claims victory.

There are two main scenes, one is lobby scene, which is used to connect two users, setup the game as requested by the players. The other is main game scene, which is used to play the game. Unity 3D is used as the engine of the project. With its powerful deployment platform, this project can run on computers as well as mobile phones.

Introduction

Unity3D is an award-winning tool for creating Interactive 3D applications on multiple platforms. Unity3D is combined by game engine and the editor. The engine contains software components that are the most common and recurring tasks in game research and development. It covers the topic like sound, graphics, physical and network functions, and it supports C#, Boo, and JavaScript scripting.

Figure 1.Unity3D LOGO

Lobby Scene

There are two different ways to go into Game Scene from lobby scene.

One is designed for developer user, It can directly link by manual connection, which is based on TCP/IP mechanism protocol. When one player goes into the manual connection, it has to create a room. After the player create a room. The other player can only join the room by typing the physical address. Also, after 2 players join the room and ready for the game. Two players are going to the same game scene in the same time. In addition, player can select the name and color in the game. Once player select the color, the main character rigidbody can directly changes into selected color. Lobby scene can use round robin to let the main character spawn randomly in the spawnable point. Currently, main game scene has four different spawnable point. In case we want to add more player, we just need to add some spawnable point and set the maximum number as we wish.

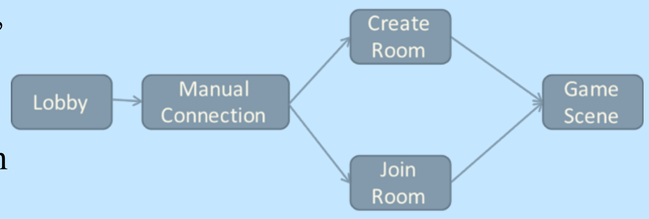

Figure 2. A Flow of Manual Connection

The other one is designed For normal user, The best connection is matchmaker. This connection is based on UNET mechanism protocol. When one player goes into the matchmaker, it can create a room. After the player create a room. The other player can list the room and select which room player is going to connect. After 2 players join the room and ready for the game. Two player are going to the same game scene in the same time. Also, this scene can select the color and types with the player name and room name as player wish.

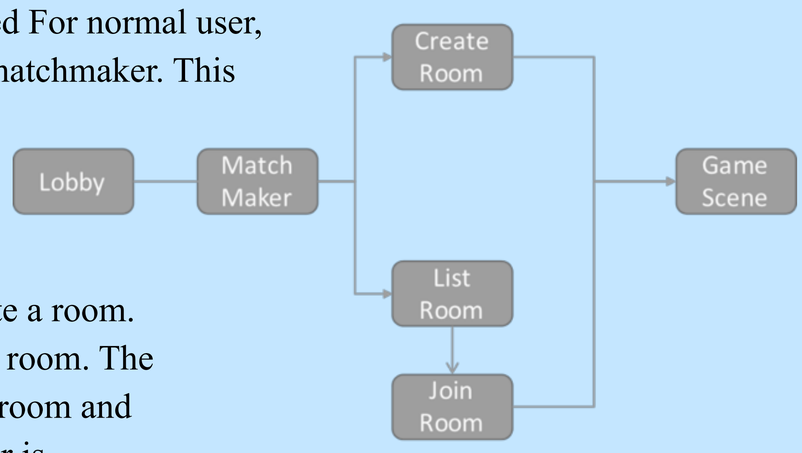

Figure 3. A Flow of Match Maker

The structure of our distributed search engine is as follows:

There are actually two TCP connections between query processors. One for sending/receiving query and the other one for sending/receiving results. The searching process (not including preparing data part) is:

Then we need to discuss the procedure including preparing data part in three steps: distributed crawling, indexing and distributed searching.

Game Scene

After we done some fussy lobby. We are going to create our major character, which is tank. Unity3d give us a chance to create major character by prefab, which is very convenient for us to develop game. For every prefabs have one or many actions, some of them has a model.

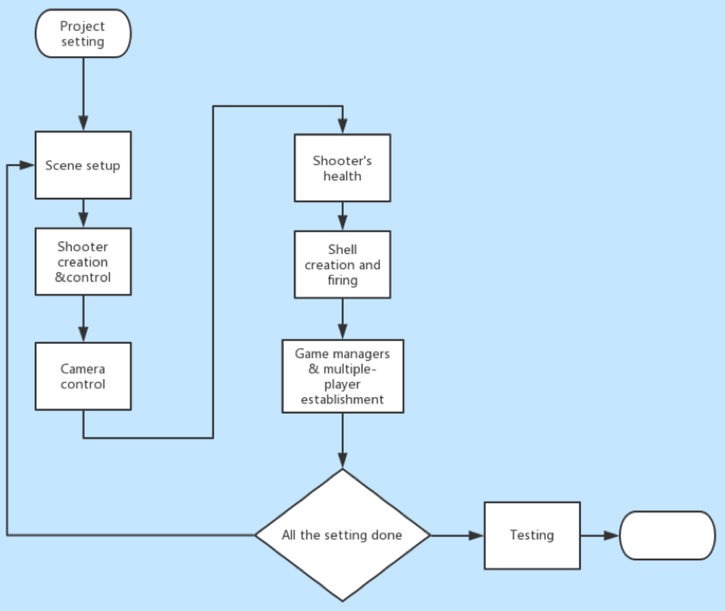

Figure 4. Game Scene Data Transition Diagram

Tank prefab

At the beginning, game scene creates 2 tanks on the level art. The tank can move because a tankmovement.cs script drag to tank prefab. The tank can shoot because a tankshooting.cs script drag to tank prefab. The tank can die because I create a tankhealth.cs script drag to tank prefab. The script tells the detail how the tank does and how to generate other prefabs. For example, tank shoots have to create a new shell prefab. The shell need to know, where am I going to instantiate and where should I shoot. For other example, after the tank dies, the prefab TankExplosion has to play immediately.

Figure 5. Tank Model

Shell prefab

After we done tank prefab, what we need to generate is shell prefab. Also, Shell prefab has a model and many actions. Once player hit on fire bottom, the shell automatically instantiates from the mouse of the tank.

If hit something not related rigidbody or do not hit something, the shell will die and play the prefab ShellExplosion. If the shell Hit something related rigidbody and has a tankHealth.cs associate with rigidbody, it will calculate the damage and play a physic force to tank. Finally, shell dies and play the prefab ShellExplosion. Then one shoot is over.

This project contains other 30 different prefabs (like tank and shell) to guarantee the mobile game to play. what you need to do is play my game and have fun.

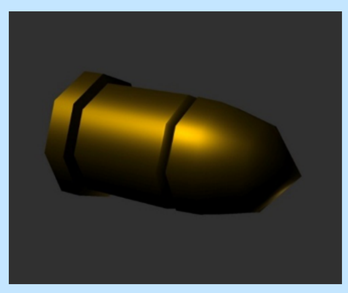

Figure 6. Shell Model

Figure 7. A screenshot of Tank health wheel

References

[1]"Unity - Unity - Overview", Unity, 2017. [Online]. Available: https://unity3d.com/unity. [Accessed: 10- Jun- 2017]

[2]"Lara Croft GO | Made with Unity", Made with Unity, 2017. [Online]. Available: https://madewith.unity.com/en/games/lara-croft-go?_ga=2.66045934.1926479982.1494921971-954352829.1491021222. [Accessed: 10- Jun- 2017].

[3]"Jenny LeClue - Detectivú | Made with Unity", Made with Unity, 2017. [Online]. Available: https://madewith.unity.com/en/games/jenny-leclue-detectivu?_ga=2.27652092.1926479982.1494921. [Accessed: 10- Jun- 2017].

[4]"TanksTutorialSlideDeck-v1", Oc.unity3d.com, 2017. [Online]. Available: https://oc.unity3d.com/index.php/s/n6P1VcTa4NWQbyn?_ga=2.175456482.1744239074.1494930459-954352829.1491021222. [Accessed: 10- Jun- 2017].