

# Multiplayer Shooter Game

Harshal Narkhede<sup>1</sup>, Ganesh Kangade<sup>1</sup>, Prof. J. N. Shaikh<sup>2</sup>

<sup>1</sup>Student, Department of Computer Engineering, Zeal College of Engineering and Research Pune, Maharashtra, India

<sup>2</sup>Professor, Department of Computer Engineering, Zeal College of Engineering and Research Pune, Maharashtra, India

## ABSTRACT

### Article Info

Volume 8, Issue 3

Page Number : 524-527

### Publication Issue :

May-June-2022

### Article History

Accepted: 10 May 2022

Published: 30 May 2022

In this paper, we present the design and implementation of the game called "Multiplayer Shooter Game". It was developed keeping the Windows Operating System in mind. This is a shooter game with mechanics with player run, jump, shoot enemy, Enemy path finding etc. We implemented three different modes to play in this game. First one is single player mode where you need to kill all the enemies in path and finish the level, then we have multiplayer mode where you can connect with anyone from all over the world and challenge them and finally the zombie mode, which is a classic zombie mode with wave system implementation.

**Keywords:** Unity Game Engine, Game Modes, Modules.

## I. INTRODUCTION

Game improvement has forever been a dubious scholastic theme. The PC gaming industry has developed huge amounts at a time, turning into a standard programming improvement area, and procuring billions of dollars in income every year. This game contains three different modules to play where you can explore the different maps and have fun while playing. Unity3D Game Engine is an incorporated improvement apparatus used to foster intelligent items like computer games, architectural representation and continuous 3D activities.

## II. FUNCTIONAL REQUIREMENTS

The game is developed with Unity engine and C# script using visual studio code. The game can be controlled by key-board and mouse. Display resolution must be 1920\*1080. The game also contains music and sound effects, not crucial for gameplay. We are using Photon engine for database. Photon Unity Network (PUN) is designed for unity indie game developers. It is a great server for simple setup for multiplayer.

### A. Unity 3D

The Unity game engine was presented by Unity Technologies in 2005, and has since become quite possibly the most famous stages for creating 2D and

3D games. It has been embraced both by little, outsider engineers and by huge business game improvement organizations. The capacities that are upheld by Unity3D are exceptionally plentiful. Unity3D produces the applications in light of JavaScript as well as C. These are utilized to relegate the movement or constant progress of the Game-Objects characterized in the application. GUI of Unity3D assists another engineer with drawing closer electively and content and program the change of the Game-Object. The most recent form, Unity 5, is a completely useful game engine, with all cutting-edge highlights empowered and unreservedly accessible to designers. Unity is an integrated environment, which combines a number of sophisticated components such as the PhysX physics engine, the Mechanism animation system, a self-contained terrain editor and much more. It is also seamlessly integrated with the Monodevelop code editor, so that any changes made in Monodevelop are transparently compiled by Unity's C# or Javascript compilers, and inserted into game. Compilation errors are displayed in the Unity console window

## B. Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C, Java, Python, PHP, Go) and runtimes (such as .NET and Unity). Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python, C++ and Fortran. It is based on the Electron framework,[19] which is used to develop Node.js Web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

## C. Photon Unity Networking (PUN)

Photon Unity Networking (PUN) is a Unity bundle for mul- tiplayer games. Adaptable matchmaking gets your players into rooms where items can be synchronized over the organization. RPCs, Custom Properties or "low level" Photon occasions are only a portion of the highlights. The quick and (alternatively) dependable correspondence is done through committed Photon server(s), so clients don't have to associate coordinated.

PUN Structure: Generally, you don't need to mind the construction of the PUN bundle yet only for the outline, this is the way things stack up. The PUN bundle wraps up three layers of APIs: The most elevated level is the PUN code, which executes Unity-explicit highlights like organized objects, RPCs, etc. The subsequent level contains the rationale to work with Photon servers, do matchmaking, callbacks and such. This is the Realtime API. This can be utilized on it's own as of now. You will see a great deal of cross-over of points among PUN and the Realtime API (a.k.a. LoadBalancing API) however that is ne. The most reduced level is comprised of DLL records, which contain the de/serialization, conventions and such.

## III. SUMMARY OF THE GAME DEVELOPMENT

In this research paper:

- 1) Game Name: Multiplayer Shooter Game.
- 2) Game Genre: Multiplayer, Shooter.
- 3) Platform: Supports only Windows operating System.
- 4) Game Characteristics: Our game has three levels for the player to travel through. The various scenes depicted in those levels.

### A. Game Modes

- 1) Single Player Mode: In the single player mode, we build the entire 100m\*700m map, where you need start at a point and end in the last of the maps. There are lot of Enemy AI in the map which will attack you if you get in their range or shoot them.

Also, we added ammo pickup system where there are lot of ammo lying around and you need to pickup ammo if you run out of it. There is also tab which has access of number of ammos we have and health bar. We also have three different guns with different attributes in the game, which are glock, shotgun and rifle.



**Fig. 1. Single Player Mode**

2) Multiplayer Mode: As you can see now this is our multiplayer map. We can play with player from over the world from any region in here. We have all the player ammo, ammo pickup, three different guns with different attributes etc. mechanics available here too. It is a simple PVP mode where you need to kill another guy to win the level. In top left corner, we also can see how many numbers of players are connected and in bottom middle, we can see the health bar, selected gun, and remaining ammo. Also, whatever name we put in lobby menu, are get shown over like that shown in picture.



**Fig. 2. Multiplayer Mode**

3) Zombie Mode: we also have a classic zombie mode in the game. This is just a side project we just wanted to add to our game. Is this also, all the mechanics for player are there. Only This game mode works with wave system. We generate zombies' wave at the beginning of the game. As we cleared the first wave of zombies, we instantiate another wave with more zombies. We have five waves are implemented in the game. After that you get a message that you win, or you can restart. In the top left corner, we also have access to how many zombies are remaining and the health bar is also there at bottom middle of the screen.



**Fig. 3. Zombie Mode**

## B. Modules

- 1) Game UI Screen:
- 2) Play: This is to select whatever mode you want to play.
- 3) Settings: This module can be use to change the game graphics and intensity of the background music.
- 4) Rules: This module shows the rules and controls required for the movement and exploration by the player in the game and constraints need to be followed by the player
- 5) Team Join: This is for Multiplayer Mode when you have to join the team.

### C. Data Flow Diagram

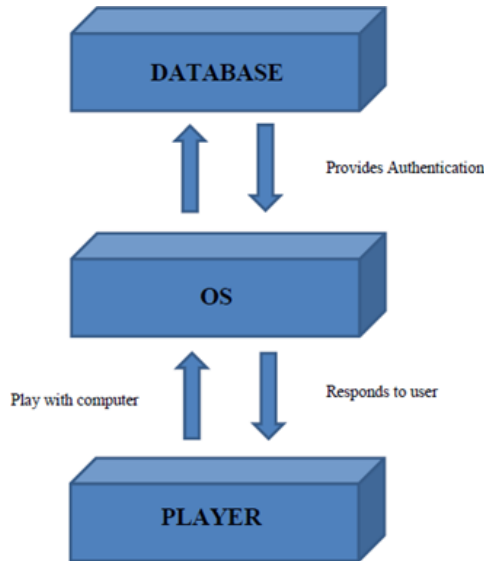


Fig. 4. Data Flow Diagram

### IV. FUTURE SCOPE

As our game going to be a multiplayer, maybe we will add chat option between user to interacts and more efficiently. Also in future, we can also add voice chat option, so while playing multiplayer with friends you can directly chat with you friend via voice. Maybe we will add profile system, where you can set up your profile, visit other profile, add someone by searching profile name function etc.

### V. CONCLUSION

This is fully working multiplayer, zombie shooter 3D game that really impact our core experience within game develop- ment field. We also overcome the problem within some of our research paper as we build a working UI for every game module/mode we have.

### VI. REFERENCES

[1]. A project-based guide to building 2D, 3D, augmented reality games from scratch <https://www.amazon.in/Unity-2020-Example-project-based-augmented/dp/1800203381>

- [2]. <https://gamedevacademy.org/category/unity-tutorials/unity-multiplayer-tutorials/>
- [3]. <https://link.springer.com/article/10.1007/s40869-019-00080-6>
- [4]. [https://www.researchgate.net/publication/312591645\\_Cross-platform\\_Application\\_Development\\_using\\_Unity\\_Game\\_Engine](https://www.researchgate.net/publication/312591645_Cross-platform_Application_Development_using_Unity_Game_Engine)
- [5]. <https://docs.unity3d.com/Manual/index.html>