Game Development Using Unity Game Engine

Submitted by

Muhammad Atif

Supervised by

Dr. Sara Shahzad

BS Computer Science
Session 2020 - 2021

Department of Computer Science
University of Peshawar

Contents

Chapter 1

Introduction

Proximity Assault is an action game in which the player takes out foes that stand in the way of advancement. The player must strategically battle the enemies, who are actual human kinds. The difficulty level rises as the player advances; adversaries attack when the player gets too close to them. The game has an AI engine that increases the difficulty and realism of enemy engagements, as well as a shop system where players may buy weaponry.

1.1   Story Line

The area was attacked by terrorists and they took control of most of, the nuclear plans, the role of the player is to terminate them and protect the world for another destructive war. The player plays as a skilled terminator who is charged with the mission of terminating all the enemies and securing the nuclear plans. As the player progresses in the game, they will face more stronger and skilled enemies trying to make the levels harder to complete. The player will uncover the hidden conspiracy and people behind this destructive plans.

1.2   Features

- **Uniqueness in environment:** The game provides a unique environment for every level and player will experience gameplay in different environment every times.

- **Diverse Bestiary:** As the player progresses in the game he will face enemies with different skillets and improved difficultly in each level.

- **Different Types of Arsenal:** The game provides a store where you can purchase different types of weapons to defeat your enemies in powerful way. The user can buy different types of weapons and abilities from store.

- **Puzzles:** The game provides an environment where the player will need to observe and use their abilities in more sensible ways as not to waste them.

- **Upgrade Abilities:** The store provides different types of weapons and magic abilities to help the player. These will be purchased with money earned during the completion of missions.

- **Learn Game Development** The design and implementation of this game will help me learn the basic interaction with unity environment and try the animation, scripting and level design of the game.

- **Target Audience**: The target audience for this game is players who enjoy action games with focus on exploring different conspiracies and dealing with combat and a light amount of puzzle solving and the developer can learn something new with a game development. The moderate difficulty level suits both casual and experienced users around.

1.3   Project Description

Many aspiring game developer lacks practical experience in the field of game development and they do not know how to fit their experience in practical projects. This project aims to provide a practical experience for learners to learn and develop a project which can enhance their skills and provide a good learning experience keeping the industrial trend in touch.

Aspiring game developers lack practical experience in game development, this project provides a platform for learning the basic interaction with unity environment as well as hand on experience on animation, scripting and level design. The main focus of this project is to gain industrial level experience in gaming and learn all the basic technologies used during game development journey.

1.4   Objectives

The objectives of this project are to:

- Develop a 3D mobile game called Proximity Assault using Unity Game Engine.

- Implement a variety of game play systems;

- including an inventory system,

- a health system,

- a damage system,

- a generic team ID system,

- an AI perception system,

- an AI behavior tree system,

- a projectile system,

- an ability system,

- a shop system,

- a UI management system and

4

- A level management system.

- Use the latest Unity technologies, including scriptable objects, Addressable, the new input system, and Timeline.

- Test the game on a variety of devices, including Android and iOS devices.

Unity Game Engine use C# (C-Sharp) language for programming. This Engine was developed by Unity Technologies, released in June of 2005 in Apple Developer Conference as a Mac OS X Game Engine. The Engine has gradually extended to support various platforms. Unity Game Engine is proprietary software but free for students, this project was developed using an educational version of the Unity Game Engine. Unity Engine targets the following API

- Direct3D on Windows and X-box 360: Used to render in three-dimensional graphics in applications where performance is very important. It uses hardware acceleration of the 3D rendering pipeline.

- OpenGL on Mac: Abstract API for drawing 2D and 3D graphics. It is designed to be implemented mostly or entirely in hardware instead of software.

- OpenGL ES on Android and iOS: A subset of the OpenGL API designed for embedded systems. It is a cross-language and multi-platform API that is why it is the most widely deployed 3D graphics API in history.

The Unity Game Engine supports the use of texture compression and resolution settings for all the platforms that the game engine supports. The Engine provides support to bump mapping, reflection mapping, parallax mapping, screen space ambient occlusion (SSAO), dynamic shadows using shadows maps, render-to-texture and full screen-post-processing effects. The Engine supports shader with many variants and declarative fallback specification which allows Unity to detect the best variant for the current hardware.

1.5 History of Unity Game Unity Engine

The engine made its first appearance in June of 2005 in Apple Worldwide Developer Conference, it was extended to 21 platforms and the latest version of Unity is currently called Tech Stream with version number 2023.1.5 released in July of 2023. The Unity version 5 allowed many inexperience developers to start producing games and many people started criticizing Unity. Games developed by Unity were downloaded more than 5 billion time and about 2.4 billion different mobile devices were used to develop them. That's the reason Unity is the leading software in Game Development.

1.6 Main Concept of Unity

The workflow of unity is built around the structure of component. A component is a smaller part of larger machine and in simple words its something that is complete on its own. Every component of a machine works independently and can accomplish its

task without the help of any outside sources. For example in a PlayStation controller, it has many buttons but each button has no idea that there are other buttons with different behavior. Every button function independently and the function controller is a one way street, and its task will never change due to what it is plugged into. This component can work as a standalone device and with multiple devices. Unity allow you to test your projects in separate windows and make edits to your code or game objects and these changes are reflected live in real time. These features are possible due to Unity component based architecture.

1.7   Project Motivation

This project aims to revolutionize the genre of action based games with modern design and that provide a fresh feel to the new generation. There were a lot of other motivations for the game including the creative environment of Unity which will allow the developer to experience new game mechanics, design quirky bug AI and craft captivating environments. It's a chance to provide the developer with a learning curve and improve their knowledge of game development. One of the reasons that motivate me to this project was the programming language used as a developer I like to program in C++ so I decided to give a try to game development with C# without switching to another language. We want to develop a game that can entertain and challenge users and also to inspire others to learn game development and think about the artistry efforts behind the game development.

1.8   Project Significance:

The significance for this project are as followed

- **Refreshing the Action Game Genre:** Many of the modern entries/players still adhere to this category of gaming and Proximity Assault on the other hand filled with strategic planning and humorous tone could attract wider range of audience. This could also inspire other developers to experience this genre and lead to a more innovative adventure games.

- **Show Casing the Power of Unity:** This game serve as a proof about the informing developers about the power of unity by making this game more visually appealing and engaging in term of graphics and 3D objects with a unique mechanics.

- **Encourage Developers:** The development of this game will be documented and shared with public, encouraging them to learn game development and make their own games using the power and features provided by Unity in a guided way. This could encourage the gaming community and lead to collaboration and knowledge sharing.

1.9   Project Uniqueness:

This game development project puts many aspects from real world in to this project. The most exciting features are the blending of genre, that is an action and adventure genre is mixed in this game providing player with an amazing experience of interaction with enemies and the provided environment in the game. Player navigating through different environment experiences different types of enemies, enemies with different type of difficulty levels. Player should expect them anywhere and every time in the game because the game is designed to add strategic depth and make it more engaging for users. The game features an encouraging perspective towards development, inviting everyone from the community to learn and collaborate in this development. The documentation will be written in such a style to allow more people to easily read it and find their way through game development.

1.10   Project Scope:

Project Scope is a map that guides the project keeping it on the track to the development journey, ensuring the team is on the track and in the end delivers an excellent product.

1.10.1   Imagine:

- You, the fearless hero navigating through the bustling areas of enemies.

- Fighting with a diverse set of enemies in which each carries a different set of abilities and power that can cause a lots of damage to you.

- Equipping you with huge arsenal of bug-bashing tools, along with some special abilities from store and some new experimental gadgets from shop/inventory.

The project scope will clarify:

- **What's In:** We will have 10-15 hour of level design, 10-15 different enemies to fight with along with at least 8 upgradable weapons/tools, Leaderboards, achievements, and an optional humorous touches add spice to the mix.

- **What's Out:** VR support, complex multiplayer features and extensive character customization won't be part of the initial launch. The main focus will be on delivering the core gameplay first.

- **Why these Choices:** Budget, timeline and team size play a key role in these deliverables. We aim to allocate all the resources with great care and completely utilize them without compromising the project.

The project scope keeps the project in a fence. We can always adjust and adopt the boundaries, ensuring a fun and achievable development process. The table below summarize these features in a more understandable way.

Elements	What's Included	What's Excluded
Platforms	PC & Mobile	VR
Engine	Unity	None
Game Play Hour	10-15	None
Environment	Diverse City Locations	None
Enemies	Increased with level	None
Extermination Tools	8+ Up gradable Options	None

Chapter 2

Feasibility and Analysis

2.1 Existing Systems

Call of Duty, often abbreviated as CoD, is a first-person shooter video game series centered on humanity's intense battles against various threats, including terrorists, military factions, and supernatural forces. Developed by multiple studios and published by Activision, the franchise is known for its cinematic action, diverse gameplay modes, and competitive multiplayer experience.

2.1.1 Variety of Enemies

- **Modern Warfare:** Engage in intense firefights against ruthless terrorist organizations, heavily armed insurgents, and tactical military units. Expect high-stakes missions that test your skills in real-world conflict scenarios.

- **Zombies Mode:** Prepare for waves of undead hordes, featuring a variety of monstrous foes. Survive increasingly difficult rounds while utilizing an arsenal of weapons and traps to fend off the relentless onslaught.

- **Advanced Warfare:** Battle futuristic enemies equipped with exosuits and advanced technology. Utilize enhanced mobility and advanced weaponry to outmaneuver and defeat these formidable opponents.

2.1.2 Weapons

- **Diverse Arsenal:** Call of Duty offers an extensive range of weapons, from classic rifles and shotguns to modern assault weapons and high-tech gadgets, allowing players to customize their loadouts to fit their playstyle.

- **Vehicles and Equipment:** Take control of armored vehicles, drones, and helicopters to dominate the battlefield, providing tactical advantages and unleashing devastating firepower.

2.1.3 Intense Action

- **Explosive Moments:** Call of Duty thrives on fast-paced, action-packed gameplay. Expect dramatic set pieces, high-octane firefights, and cinematic explosions that elevate the adrenaline levels.

- **Co-op and Multiplayer Chaos:** Team up with friends in cooperative modes or compete against players worldwide. The shared excitement and strategic coordination in multiplayer battles are what make Call of Duty an unforgettable experience.

2.1.4 A Series Full of Innovation

- **Multiple Titles, Multiple Eras:** With numerous main entries and spin-offs, the Call of Duty series spans various historical periods and future settings. Each title introduces new gameplay mechanics, enemy types, and immersive narratives, keeping the experience fresh.

- **Different Styles:** From the gritty realism of "Call of Duty: WWII" to the futuristic warfare of "Call of Duty: Infinite Warfare," there's a Call of Duty game for every preference.

2.1.5 Advantages

- **Diverse Enemy Types:** Call of Duty offers a wide variety of adversaries, from ruthless terrorist factions to advanced robotic enemies. Each encounter feels unique and engaging, keeping players on their toes.

- **Extensive Arsenal:** The game features a vast array of weapons, from classic firearms to cutting-edge technology like drones and tactical gear. Players can experiment with different loadouts to find the most effective combinations for their playstyle.

- **Co-op Mayhem:** Team up with friends in cooperative modes for chaotic and thrilling gameplay. Coordinating tactics, reviving teammates, and sharing the thrill of victory create an unforgettable multiplayer experience.

- **Variety of Game Modes:** The Call of Duty series caters to diverse preferences. From the realistic combat of "Call of Duty: Modern Warfare" to the arcade-style fun of "Call of Duty: Warzone," there's something for everyone.

2.1.6 Drawbacks

- **Repetitive Gameplay:** Despite the diverse enemies, the core gameplay loop can become monotonous. Engaging in similar combat scenarios repeatedly, even with different weapons, may lose its excitement for some players.

- **Cinematic Clichés:** Call of Duty often embraces its blockbuster action movie roots. If you're not a fan of over-the-top scenarios and scripted sequences, the experience might feel predictable.

- **Technical Issues:** The series occasionally suffers from bugs, server issues, and imbalances, which, while not game-breaking, can detract from the overall experience at times.

2.1.7 Problems

- **Steep Learning Curve:** The fast-paced nature of the game can be overwhelming for newcomers. Without adequate tutorials or guidance, new players might find themselves struggling to keep up.

- **Narrative Focus:** The story often takes a backseat to the action. While some titles attempt a deeper narrative, most prioritize explosive gameplay over storytelling, which might leave players wanting more depth.

- **Limited Replayability:** Depending on the specific game mode, the lack of substantial endgame content or varied experiences may result in diminished replay value once the campaign is completed.

2.1.8 Overall

Call of Duty offers a thrilling and dynamic experience for players who enjoy its high-octane action, cinematic gameplay, and competitive multiplayer. However, its repetitive mission structure, occasional technical issues, and sometimes shallow narrative may not resonate with everyone. If you're seeking a deeply immersive and polished shooter, Call of Duty might not be your ideal choice. But if you're in for intense co-op action and fast-paced combat, Call of Duty is definitely worth diving into.

2.2 Proximity Assault

Proximity Assault is an action based game, the player is aimed with the objectives of eliminating all the terrorist in the specific area. The area was occupied by some terrorist organization and they have occupied some nuclear plans from the government. The player will recover all these objective and move forward in the game.

2.2.1 Twist

Enemies Apocalypse: In a world overrun by a terrorist organization and they took controll of some nuclear plans from the government, The player is aimed with the objectives to recover those plan and uncover the aims of occupying these nuclear plans.

2.2.2 Dream world Assault

Enter the surreal realm of dreams where nightmares manifest as monstrous skilled terrorist. As a oneiric hero, you must navigate the ever-shifting landscape of the subconscious, solving puzzles and using dream-warping tools to vanquish these fantastical skilled enemies.

2.2.3   Target Audience

Proximity Assault targets a more casual market with simpler controls and less intense action. Or maybe it focuses on educating players about real-world situations of controlling the an organization which rebelled and went on this way.

2.3   Positive Impact

2.3.1   Educational Value

Exterminator could raise awareness about responsible pest control practices and the importance of protecting ecosystems. Educational elements could be subtly woven into the gameplay or story, making it both entertaining and informative.

2.3.2   Co-op Fun

Teaming up with friends to tackle infestations could foster teamwork and communication, making Exterminator a fun and social experience.

2.4   Detail Feasibility Report

Below is a detail feasibility report for Proximity Assault.

2.4.1   Technical Feasibility

The technical requirements for this project are a system that can support the game engine which in this case will be Unity, Unity requires a subscription but this project will be using the student plan, in which the team can use all the features for free. Unity works with C# which is a similar to C++. As a C++ Programmer we have a very good understanding of the concepts in C++ which we can easily apply to this scenario. The IDE the project will be coded with is Visual Studio Community which have a built in compiler with it. The design phase of the project will take a little longer because the development team will have to consult someone to deal with the design of the interfaces and design architecture of the system. The system can be easily completed with the above mention tools and techniques. However the system will also be tested and development will be continued in small phases as to have the room for maxi-mum improvements to interface, internal mechanisms and other features to make it very engaging. The development will also requires advance features like complex AI or detail environment, which will requires additional learning resources.

The problems faced by implementing the system will requires an expertise in testing for which the system must be test be a person from out of the team, as the a solo developer we will have to manage the time and available resources with great efficiency in order to complete all the phases of the project. The testing and designing will require some extra time which can be solve either by hiring a person from out of the team or we will develop them ourself but the at least one will requires time.

The system will probably take months to be completed because it requires the process

of learning along with development of the system. The team will try to adjust everything accordingly in order to achieve the development of the system in the limited time available.

2.4.2 Economic Feasibility

The resources required to learn the basic interaction with Unity environment will requires the team to have some skills in becoming familiar with the environment. The team will purchase some courses from Udemy in order to achieve their objective in faster and efficient way. The team will use assets which are probably free or low cost but once the testing is done the team will invest in the purchasing of assets but if there was sufficient time most of the assets will be developed for the system by the team which at some stage of development will include a designers and testers. As development is completed the team will be testing and finding bugs in the system before publishing it online, once everything is good it will be uploaded at some platforms like steam and mobile app stores with a revenue model like paid download and in-app purchases. The game market is very competitive; it would require some time to adjust itself to stakeholders out there. The team will also keep a backup plan to have budget for the marketing of the system. If the system requires some developers to be hired for some parts of the project, the team will make sure there is enough budget to get the best developers out there.

2.4.3 Social Feasibility

The most important of the social feasibility is the ethical considerations in the game, as discussed be-fore the game aims to highlight a point to the nation which is responsibility; Responsibility is one of the important aspect of the humans. This will provide a message to the nation to be aware and be awake to take action before its too late. The system will be designed in such a way that no culture, race, age or gender is targeted. The player will be able to experience a very friendly environment in which they will have no idea of the they being discriminated in anyway instead the team will design strategies in order to remove anything that will contribute to discrimination of that particular group. Instead the game target audience globally. The development of the game will be an educational asset for other students, developers and anyone who wants to start with game development but requires a clear road map for them. All the resources, documentation and any assets that are required for redesigning of the game will be available to the everyone from around the globe. The last point is basically highlighted to give the project and educational value. Accessibility features will be given more diverse touch in order to make the game playable for people with different abilities like adjusting the difficulties level and color blind options along with subtitle options to be incorporated in to the game.

Chapter 3

Requirement Specification

3.1 Introduction

This part of the document outlines the requirement specification of the game Proximity Assault which will serve as the blueprint or the base for the features, functionality and quality of the game. The aim of clearly defining these requirements is to keep everyone including developers, designers, testers and even the lovely audience to be on the same page, leading to a successful development journey. Requirement specification is extremely important for the success of a system. Vague instruction set can lead to confusions among the team and the promised end product will not be up to the mark. The requirement specification is categorized into functional and non-functional requirements which are discussed in details.

3.2 Functional Requirements

3.2.1 Main Gameplay

- Player controls a character.

- Levels filled with different enemies.

- Player uses different weapons and tools to eliminate enemies

3.2.2 Progression

- Unlock new game level with experience or in-game currency.

- Difficulty increases with every new level.

- Optional side objectives to make the game more engaging (Optional).

3.2.3 Inventory and Upgrades

- Player can use different weapons and tools with different attributes.

- Option to upgrade weapons for improving power and efficiency.

- Able to use some special abilities in the game.

3.2.4 Menus and UI

- Feature an extremely smooth and easy User interface for the players.

- Pause menus with options to save, quite or adjust setting of the game.

- Health bar, objective status indicator and other elements for feedback.

3.3 Non-Functional

The non-functional requirements of the system are about how a system should behave instead of what specific function to perform. The non-functional requirements are defined below.

3.3.1 Performance

- Maintain smooth frame rate and responsiveness on target platforms (iOS, Android).

- ' Optimize every feature in order to avoid performance issues.

3.3.2 Graphics And Audio

- Make sure the frame rate does not fall and is stable [2]

- Graphics should be visually appealing and according to the theme and tone.

- Good sounds and background music.

- Sounds that inform the user about the certain actions like low health and enemy response fire.

3.3.3 Controls

- Responsive and faster controls for aiming, and interacting with the environment..

- Sufficient options for customization.

- Sensitivity features.

3.3.4 User Interface

- New game, continue game, exit.

- Links to social media platforms.

- Setting button to adjust controls.

3.4 UI Requirements

This includes the guidelines for how a game user interface should look and function when the player is interacting with the system. It includes menus, buttons, maps, inventory screen and health bar.

3.4.1 Main menus

- Start game, continue game, and option buttons.

- Information about the game and social links.

3.4.2 Gameplay Screen

- Health bar, Objectives status indicators, ammo/resources indicators.

- Mini map or level overview (Optional).

- Button to access inventory.

3.4.3 Inventory Screen

- Display of the available resources.

- Upgrade options with cost and description.

- Special abilities management with usage indicators.

3.4.4 Pause menu

- Options to save, resume, quit and adjust game settings.

- Control layout and settings overview.

The main and most important requirement of the game will be learning how the process of designing and developing a video game works. To extract useful results from all the phases involved in game development like pre-production, production and post-production. More requirements include learning the art of how to draw graphics and music for game, scripting, artificial intelligence, level creation and user interface.

3.5 Persona

The Operation Bio-Purge has a very simple persona in which a user named Ali, the Casual Gamer.

3.5.1 Demographics

- Name: Ali

- Age: 23

- Occupation: Marketing Specialist

- Location: Peshawar City

- Device: IPhone 11

3.5.2 Gaming Habits

- Play mobile game in short breaks or during free time.

- Prefer casual, puzzle like games with simple mechanics and clear goals.

- Play games with light heart themes and great visuals.

3.5.3 Motivation

- For relaxing himself and enjoying in free time.

- Enjoys the satisfaction of completing short objectives.

- Appreciates game with quick to do actions.

- Like good game designs.

3.5.4 Concerns

- Limited time to play the game

- Dislike ads and in-game pop-ups for purchases or any sort of disturbance.

- Do not like repetitive game play.

3.6 User Interactions with the System through Use Case Diagram

This section dives into the user interaction between the user and the game using a use case diagram. A use case diagram describes all the functionalities of the system along with external users aiming to describe how an actor (user) interacts with the system. A use case diagram also specifies the main functionalities of the system that a user can perform while interacting with the system. This diagram also communicates the system behavior to the stakeholder in an easy and understandable way. The testing phase also uses the use case diagram as a base.
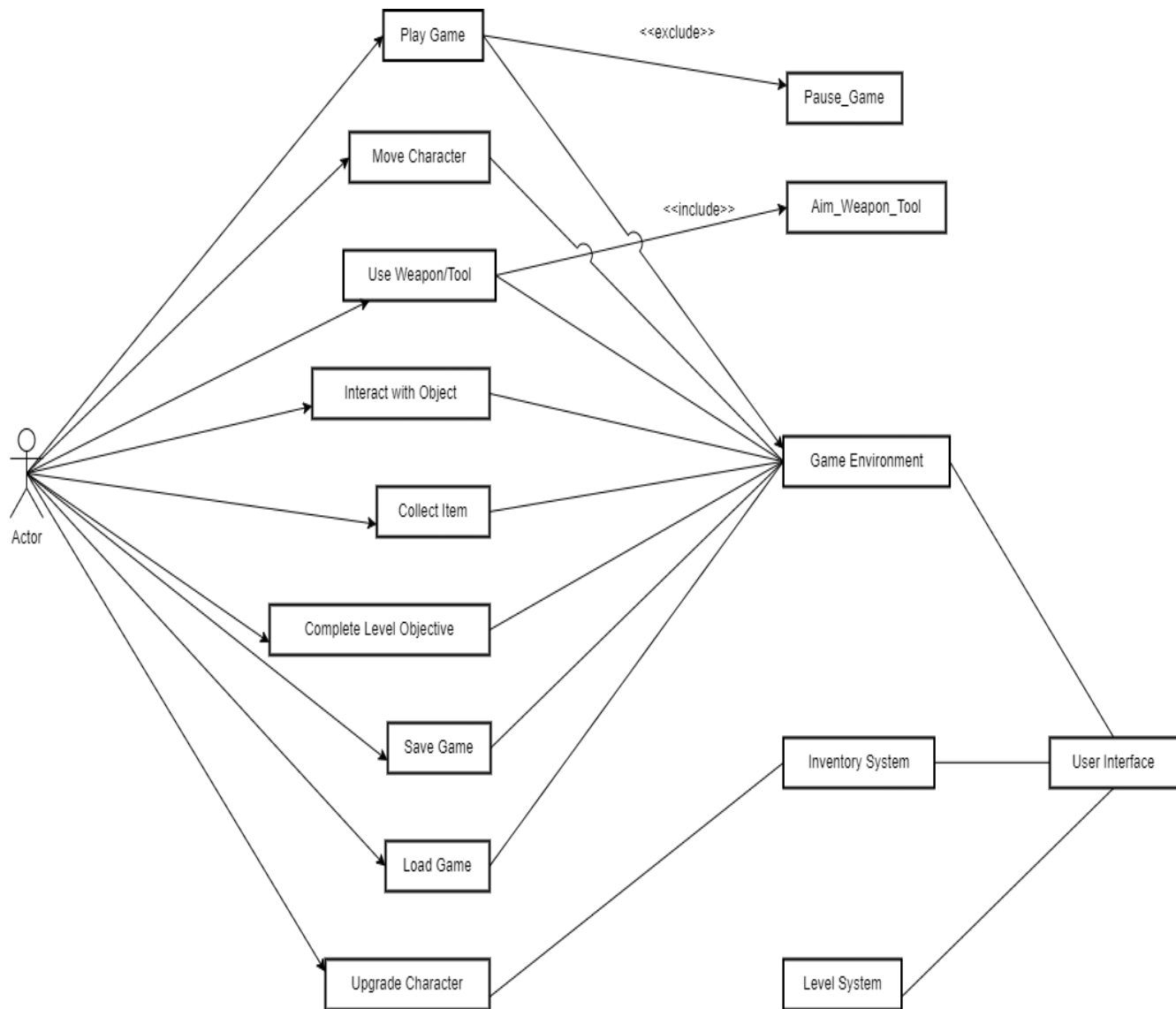
The UML diagram describes the user interaction and functionalities of the "Proximity Assault" game system.

3.7   System Boundary

- **Actor(Player):**The player represents the user interacting with the system.

- **Game Environment:**This represents the virtual environment where the game takes place including elements like terrains, enemies and objects.

- **Inventory System:** Manages the player inventory, the items that a player collects during the game play or purchases them from a store.

- **Level System:** This functionality manages the progression and objectives with in each level.

- **User Interface:** Provides the visual and interactive way through which a user interacts with the game.

3.7.1 Use Case

The use case describes various actions or functionalities that a player can perform while interacting with the system.

- **Play Game:** Begins the game play with the game environment.

- **Move Character:** Allows the player to move their character with in the game environment.

- **Interact with Object:** Enables the player to interact with objects present in the game environment.

- **Use Weapons/Tools:** Allows the player to utilize weapons or tools with in the game.

- **Collect Item:** Enables the player to collect game items found with in the game environment.

- **Upgrade Character:** Allow the user to upgrade their character abilities.

- **Complete Level Objectives:** Indicates the accomplishment of objectives specific to a level.

- **Save Game:** Allow the players to save their current progress within the game.

- **Load Game:** Allow to load the game from previously saved state to provide more engaging feeling the users.

3.7.2 Include and exclude

Indicate the relationships between use cases

- **Play game exclude pause game:** This means that when the player is playing the game, the pause functionality is not available.

- **Use Weapons/Tools includes Aim Weapons/Tools:** This signifies that aiming is a part of the process of using weapons within the game.

3.7.3 Actor Use Case Relationships

Arrows connecting actors (Player) to use cases indicate that players interact with those functionalities.

3.8 Development Environment Requirements

Development of this game involves various tools and software with their own system requirements, below is a very detailed information of the development environment required for this game.

3.8.1   Development Environment

Operating System

- **Minimum:** Windows 10 64-bit (recommended latest stable version).

- **Alternative:** MacOs Mojave 10.14+ (recommended: latest stable version).

- **Justification:** Both Unity and MacOs are officially supported by Unity.

3.8.2   Hardware

Processor

- **Minimum:** x86 or x64 architecture with SSE2 instruction set support(e.g. Intel core i5-4460 or AMD equivalent).

- **Recommended:** Newer processor with multiple cores (e.g., Intel core i7 or AMD Ryzen) for smoother performance during development and testing.

RAM

- **Minimum:** 8GB

- **Recommended:** 16GB for multitasking and handling larger projects

Graphics Card

- **Minimum:** DX10, DX11, DX12 capable (e.g., NVidia GTX 970 or AMD Radeon R9 290)

Storages

- **Minimum:** 20GB available space for Unity and Project Files

- **Recommended:** More space depending on project size, additional tools and asset libraries.

3.8.3   Software

Unity Game Engine

- Download the latest stable version that is compatible with your system.

- Consider Unity Personal for small projects or Unity Plus for additional features.

Version Control

- Git (Recommended) with code hosted on GitHub

Graphics Designing Software

- Photoshop, GIMP for creating UI elements textures or logos.

3D Modeling Software

- Blender

Audio Editing

- Audacity

Asset Creations

- **Asset Creation:** Asset will be used from unity market place.

3.9 Minimum Hardware Requirements for Running the Game

These are the requirements for running the Operation Bio-Purge game on Android or iOS. These are the general requirements needed for running the system.

3.9.1 Android

- **Processor:** Dual-core 1.2 GHz processor (recommended: Quad-core 1.5 GHz or better)

- **RAM:** 1GB RAM (recommended 2GB or more)

- **Storage:** 500 MB free space

- **Operating System:** Android 5.0 or later (recommended Android 8.0 or later).

- **Graphics:** Adreno 305 GPU or equivalent (recommended: Adreno 405 or later).

3.9.2 iOS

- **Device:** iPhone 6 or later (recommended iPhone 7 or later).

- **Operating System:** iOS 10 or later (recommended: iOS 12 or later).

- **RAM:** 1GB RAM (recommended: 2GB or more)

- **Storage:** 500 MB free space.

Chapter 4

System Design

System design is the process of defining the [1] architecture, components, modules, interfaces, and data for a software system to meet specified requirements.

System design involves defining the overall architecture, breaking down the system into components and modules. Architectural design establishes the high-level structure, relationships between components, and communication mechanisms within the system.

4.1 Architecture Design

Architecture design is an important phase in system development that defines the structure and organization of a system. It defines the overall framework that indicates how various components interact, ensuring that the system meets functional requirements and performs efficiently. The architecture serves as a blueprint, guiding developers throughout the implementation process and enabling scaling and maintenance.

- **Component Identification:** Defining the individual components that make up the system, such as modules, libraries, and services.

- **Interaction Patterns:** Specifying how these components communicate and collaborate, including protocols, data flow, and interfaces.

- **Design Patterns:** Utilizing established design patterns to solve common problems in software architecture, promoting re-usability and clarity.

- **Scalability and Performance:** Considering how the architecture will support future growth and maintain performance standards under varying loads.

4.1.1 Architecture Design of Gaming System

Bibliography

[1] HOCKING, J. *Unity in action: multiplatform game development in C.* Simon and Schuster, 2022.

[2] SINGH, G., AND VANGAVOLU, V. V. S. S. R. Effect of resolution on player performance and experience in virtual reality low-fidelity first-person shooting games, 2023.