**Requirement Specification:**

This part of the document outlines the requirement specification of the game Operation-Bio Purge 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 with reference to the game Operation-Bio Purge below.

**Functional Requirements:**

* **Main Gameplay:**

* Player controls a character.
* Levels filled with different enemies.
* Player uses different weapons and tools to eliminate enemies.
* Boss enemies will be present at the end of every level.
* **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).
* **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.
* **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.

**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 for the Operation-Bio Purge are defined below.

* **Performance:**
  + Maintain smooth frame rate and responsiveness on target platforms (iOS, Android).
  + Optimize every feature in order to avoid performance issues.
* **Graphics And Audio:**
  + 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.
* **Controls:**
  + Responsive and faster controls for aiming, and interacting with the environment.
  + Support for different input methods like keyboard, mouse (Optional).
  + Sufficient options for customizations.
  + Sensitivity features.
* **User Interface:**
  + New game, continue game, exit.
  + Links to social media platforms.
  + Setting button to adjust controls.

**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.

* **Main menus:** 
  + Start game, continue game, and option buttons.
  + Information about the game and social links.
* **Gameplay Screen:**
  + Health bar, Objectives status indicators, ammo/resources indicators.
  + Mini map or level overview (Optional).
  + Button to access inventory.
* **Inventory Screen:**
  + Display of the available resources.
  + Upgrade options with cost and description.
  + Special abilities management with usage indicators.
* **Pause menu:**
  + Options to save, resume, quit and adjust game settings.
  + Control layout and settings overview.
* **Other UI elements:**
  + Tutorials to instruct player how to play and how to navigate the UI.
  + Victory screen at the end of the level.

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 preproduction, 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.

**Persona:**

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

**Demographics:**

* Name: Ali
* Age: 23
* Occupation: Marketing Specialist
* Location: Peshawar City
* Device: IPhone 11

**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.
* Gets bored with complex controls and demanding mechanics.

**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.

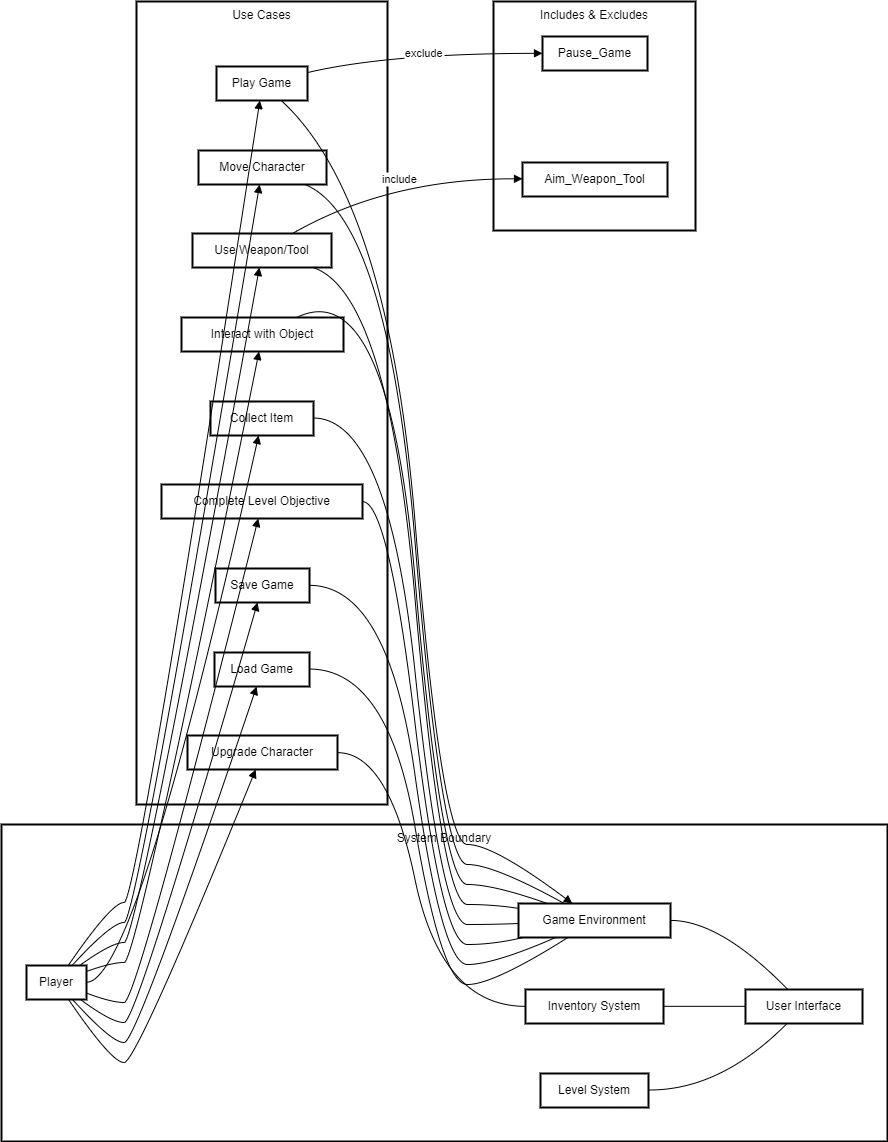
**Concerns:**

* 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.

**User Interactions with the System through Use Case Diagram:**

This section dives into the user interaction between the user and the Operation-Bio Purge 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 “Operation Bio-Purge” game system.

* **System Boundary:**
  + **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 gameplay 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.
* **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.
  + **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 or attributes.
  + **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.
* **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.
* **Actor Use Case Relationships:**
  + Arrows connecting actors (Player) to use cases indicate that players interact with those functionalities.

**Development Environment Requirements:**

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

**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.

**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.

**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
* **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.

**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.

* **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).
* **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.