Capstone Project

Software Requirements Specification

Roberto E. Acevedo Morales
Adrian Gonzalez
Dan Jones
Rajat Singh

Contents

1 Introduction

1.1 Purpose: Mission Statement

The goal of our team was to bring the excitement and uncertainty of distant galaxies to a browser window with a fast-paced arcade style game containing infinitely generating obstacles, enemies, and adventure.

1.2 Scope

The capstone project completed by Group 4, herein referred to as "The System," shall consist software, audio, and graphical assets combined to provide an entertaining gaming experience harkening back to traditional arcade titles such as Galaga and Space Invaders, but with some modern twists.

1.3 Definitions, Acronyms, and Abbreviations

1.3.1 **Application Programming Interface (API)**
A set of commands, functions, protocols, and objects that programmers can use to create software or interact with an external system.

1.3.2 **Directional Keys**
Keys on a user's keyboard including the arrow keys or "WASD" keys that are used for basic player movement.

1.3.3 **Frame Rate**
The rate at which consecutive images called "frames" are displayed while rendering film or computer graphics.

1.3.4 **Game Engine**
A framework for game development that provides inherent support for various common components of a game such as physics, animation, audio, and lighting.

1.3.5 **Graphical User Interface (GUI)**
A user interface that includes graphical elements, such as windows, icons and buttons. Interactions with the GUI are typically completed with a mouse and keyboard.

1.3.6 **HTML5**
A W3C specification that defines the fifth major revision of the Hypertext Markup Language (HTML).

1.3.7 **Leaderboard**
A table displaying the names of users with the top-10 highest scores in the game.

1.3.8 **Pickup**
Any generated item that can be collected by the user by colliding with it. They offer positive effects for the player such as replenishing health or energy or adding to the user's in-game currency.

1.3.9 **Player**
The virtual protagonist in the game which is controlled by the user during gameplay. This is differentiated from the "user" role in this document.

1.3.10 **Round**
One round of gameplay consists of the time they are able to control the player until exhausting

all of the player's lives.

1.3.11 **Score**
Within the context of The System, the uesr's score is determined by the amount of time they were able to survive during a single round of gameplay.

1.3.12 **Session**
The timeframe between when the user first launches The System and when they exit The System.

1.3.13 **Side Scroller**
A type of game in which a side-view camera angle is used to view a character travelling from left to right on the screen, involving movement in one continuous direction in many cases.

1.3.14 **Terrain**
A combination of randomly generated obstacles and enemies that make up the game world through which the user must navigate in order to survive for the longest time possible.

1.3.15 **User**
The human individual interacting with The System. This is differentiated from the "player" role in this document.

1.3.16 **User Interface (UI)**
See "Graphical User Interface (GUI)." The terms may be used interchangeably in this document.

1.3.17 **WebGL**
A JavaScript API for rendering interactive 2D and 3D graphics within any compatible web browser without the use of plug-ins.

1.3.18 **World Wide Web Consortium (W3C)**
An international community where Member organizations, a full-time staff, and the public work together to develop web standards.

1.4 References

1.4.1 https://techterms.com/definition/api
1.4.2 https://en.wikipedia.org/wiki/Frame_rate
1.4.3 https://unity3d.com/what-is-a-game-engine
1.4.4 https://techterms.com/definition/gui
1.4.5 https://www.techopedia.com/definition/27153/side-scroller
1.4.6 https://www.webopedia.com/TERM/H/HTML5.html
1.4.7 https://www.w3.org/Consortium/

1.5 Overview

The following document outlines the software requirements and specificatiosn for The System, including the fucntional, nonfunctional, domain, hazard, and system requirements. Requirements are organized into sections based on their applicability to more general goals such as resources used on the player's end, the features incorporated within the game, and how the game performs under appropriate conditions.

2 Overall Description

This section is intended to provide an overview of The System as a whole. It will explain the basic functionality of The System and how its various components interact with each other. It will also provide some insight regarding potential users and stakeholders, and list the constraints and dependencies involved in the development and use of The System.

2.1 Product Perspective

The system consists of a variety of different software modules, auditory, and visual assets all packaged within a singular final build. It can be hosted on an HTTP server to be accessed by any user with a link. Accessing and interacting with the system will all be performed within the web browser chosen by the user. Support for different browsers and adherence to web-based standards and protocols is left up to the build process supplied by the Unity Game Engine.

2.2 Product Functions

The functions of The System are divided into two major categories: gameplay and navigation. The gameplay category can be further subdivided into three additional subcategories: movement, offense, and collection. The movement subcategory involves using the directional keys to navigate through the randomly generated terrain. Movement can be used for defensive purposes to avoid obstacles or to position the player for an attack. The offense subcategory involves using the mouse to rotate the player and firing at obstacles or enemies. This can be done for the purpose of eliminating threats from enemies, or clearing up the terrain for safer movement. The collection subcategory involves the user strategically selecting randomly generated pickups to collect in order to recover health and energy, as well as add to their in-game currency.

The navigation category includes any interactions the user carries out with the various features present in the UI included with the game. Such interactions mainly take place outside of the gameplay – before the user has begun playing or after the player has died. Once The System is launched, the user is greeted with a main menu which provides them with control over when the gameplay begins. This main menu contains a settings page which allows the user to adjust their volume settings or delete their progress and other saved data. Upon exhausting all the player's lives during gameplay, a "game over" screen is displayed to the user. This provides them with four options: return to the main menu, begin another round of gameplay, visit the in-game store, or view the leaderboard. Inside the store, the user can trade their in-game currency for additional weapons and upgrades which improve the player's mechanics to facilitate their survival during gameplay. The leaderboard option directs the user to a UI panel prompting them to submit their name and score. Whether the user decides to submit their name or not, they are then shown a new UI panel with the names and scores of the top-10 highest scores submitted.

2.2.1 Use Cases

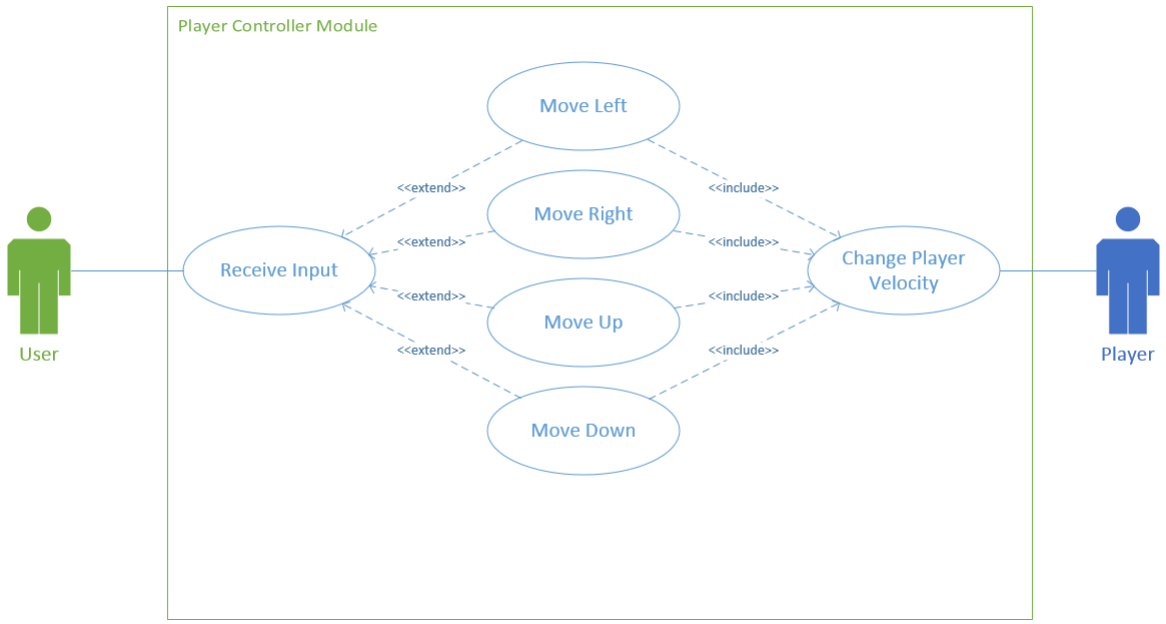

Figure 1: Move Player (UC-1)

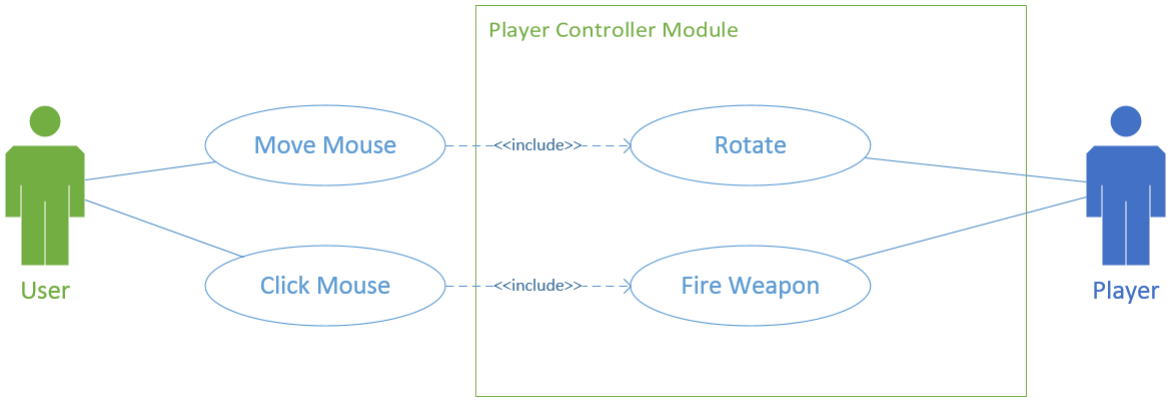

Figure 2: Fire Weapon (UC-2)

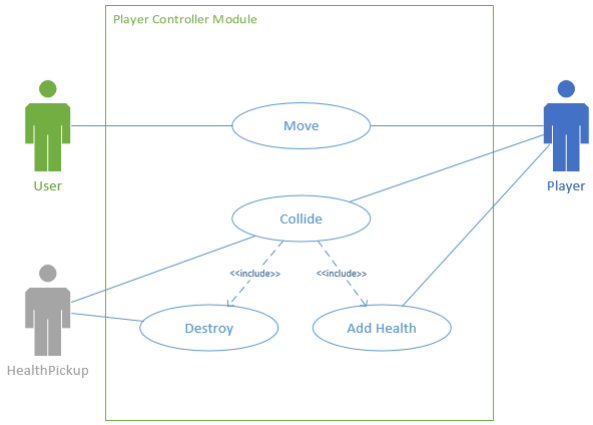

Figure 3: Collect Pickup (UC-3)

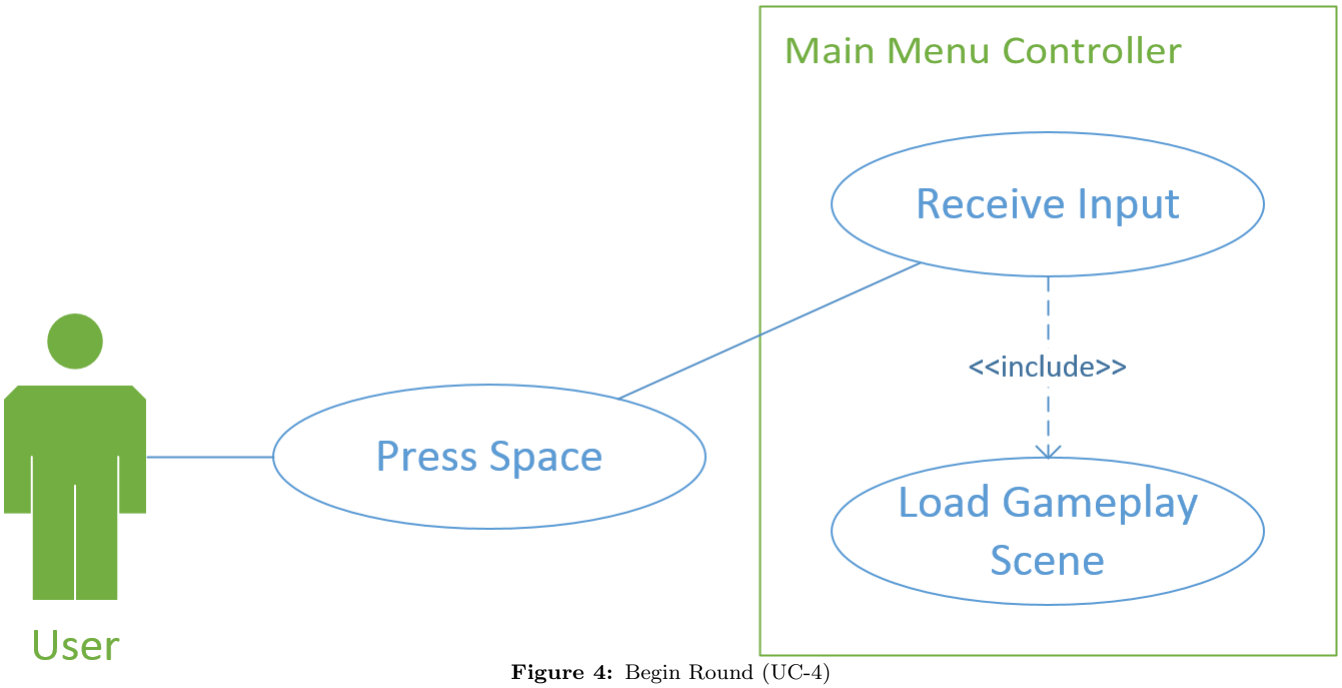

Figure 4: Begin Round (UC-4)

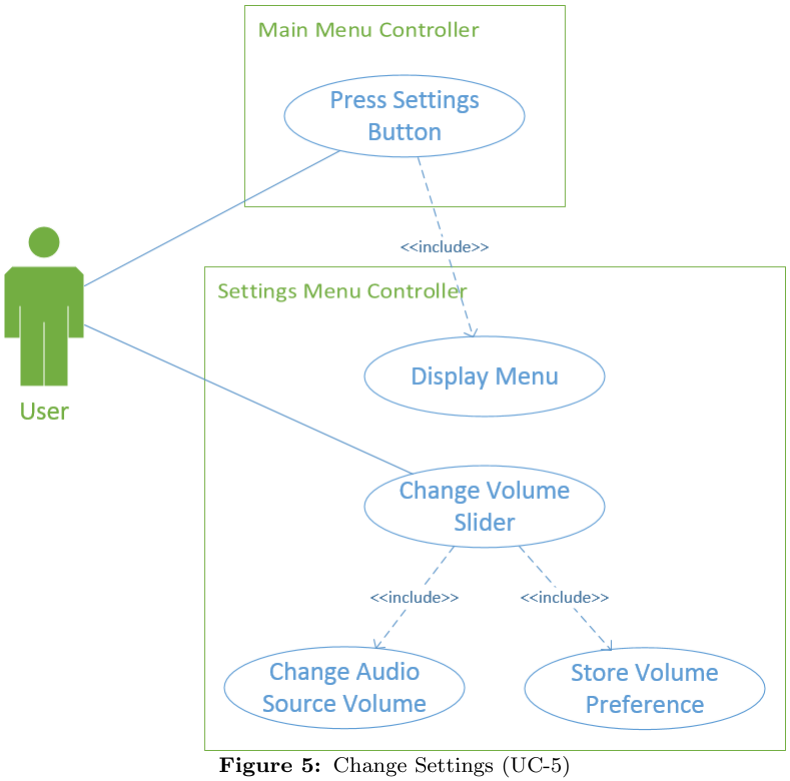

Figure 5: Change Settings (UC-5)

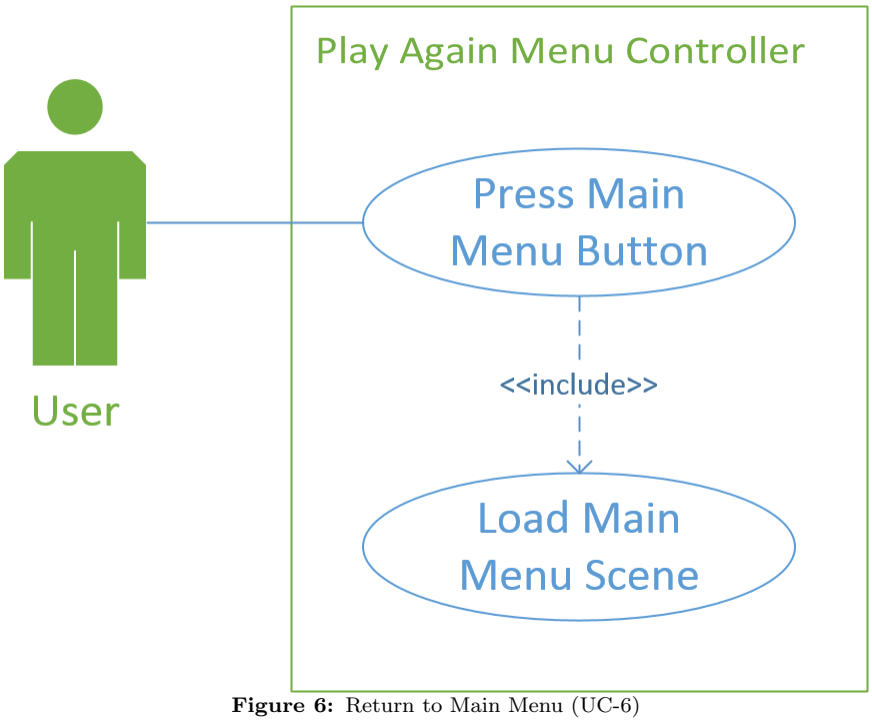

Figure 6: Return to Main Menu (UC-6)

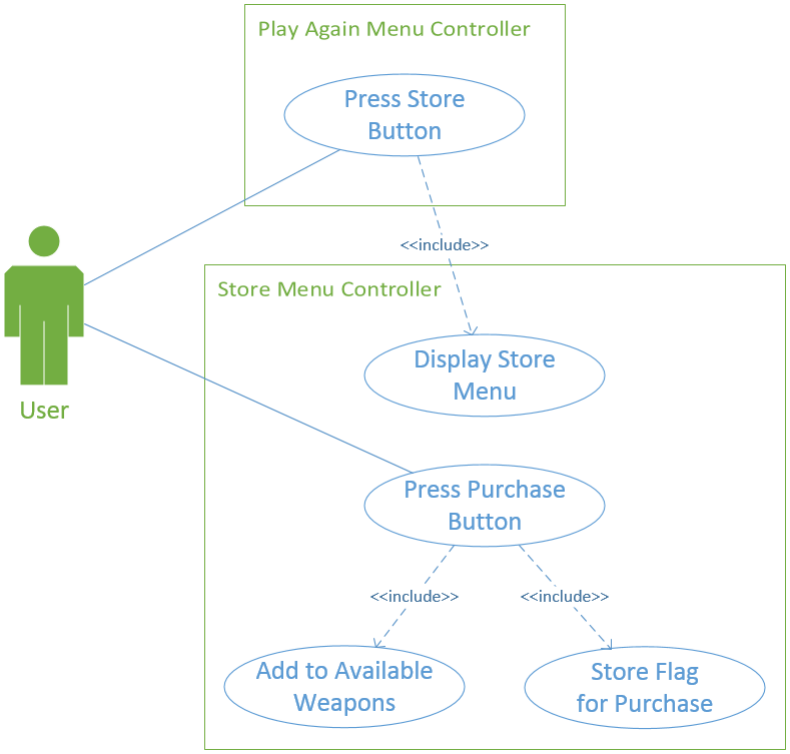

Figure 7: Purchase Weapon (UC-7)

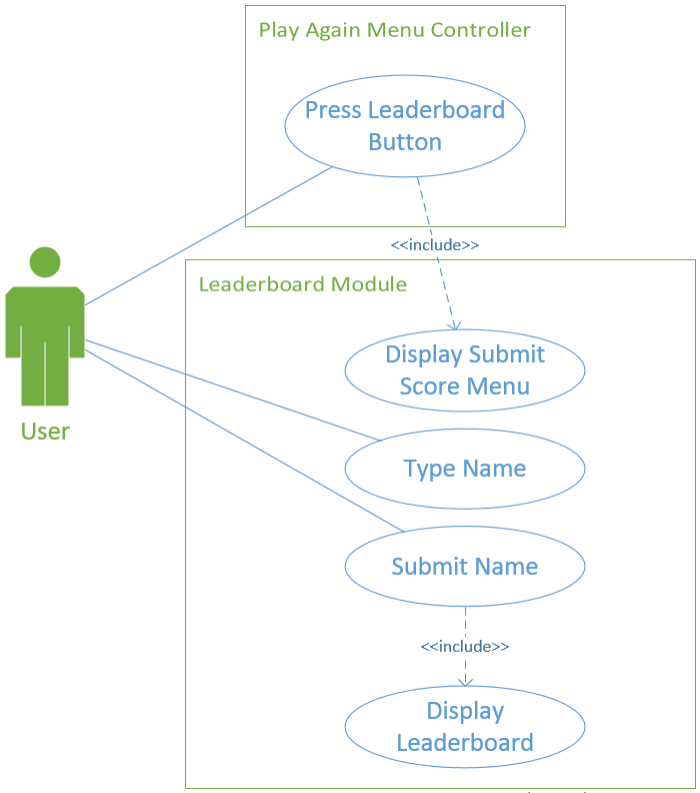

Figure 8: Submit Score to Leaderboard (UC-8)

2.3 User Characteristics

Due to the nature of video games, there is a broad and diverse pool of potential users of The System. There is no reason to exclude individuals as potential users based on any age constraints ranging from children to the elderly. As such, no special knowledge or skills should be assumed for any user with the exception of the basic use of a computer, graphical user interfaces, and web browsers.

2.4 Constraints

2.4.1 Browser

Old or outdated browsers may not offer the proper support for prerequisite technologies such as WebGL and HTML5 to properly run The System. Given the variety of browsers that are either currently or previously available to the public, providing support for every version of every browser is simply not feasible.

2.4.2 Hardware

Much like the issues that can arise with antiquated browsers, older computers may not be able to meet the demands to adequately run The System due to hardware limitations.

2.4.3 Type of Device

With the proliferation of tablets and smartphones, many games being produced today are intended for these platforms. However, The System is designed with a more traditional set of mouse and keyboard controls. So although users may possibly be able to access and load The System via the browser on their smart device, there is currently no support for the touchscreen

inputs used for these devices.

2.4.4 **Internet Connection**
Although The System itself does not depend on an internet connection or networking capabailities in any substantial way, some restrictions to the user's access to the internet may present constraints on The System as well. Most importantly, since The System is inherently web-based, the user will need some form of internet connection in order to access it. Additionally, the leaderboard feature present in the System will not be able to submit or fetch scores to and from the database without a stable connection to the internet. However, the remainder of The System will still function correctly once loaded even without internet access.

2.5 Assumptions and Dependencies

2.5.1 The user's browser is reasonably modern and able to adequately support contemporary software.

2.5.1.1 The browser was most recently updated within at least the last year.
2.5.1.2 The browser offers support for HTML5/WebGL.

2.5.2 The user is accessing The System on either a traditional desktop or laptop computer as opposed to a table or mobile device.

2.5.3 The user's device consists of adequate hardware to adequately support contemporary software.

2.5.3.1 The device has a CPU containing at least 4 cores.
2.5.3.2 The device has at least 4GB of RAM.

2.5.4 The user has a stable enough connection to the internet to load The System within a reasonable amount of time, and preferably to submit and fetch leaderboard scores as well.

3 User/Stakeholder Profiles

Stakeholder	Interests	Constraints
Player	Entertainment	Browser, Hardware, Device, and Connection limitations
System Administrator	Maintenence	Maintains system hardware and databases
Software Developer	Development	Runs in test environments

4 Core System Requirements

This section lists all of the core functional requirements for the System.

4.1 Storage Requirements

4.1.1 The System shall not exceed a size of 20MB when exported by the Unity Game Engine.

4.1.2 The System shall maintain all persistent data within a database of JSON objects handled by the IndexedDB API.

4.1.3 The System shall not exceed 1MB of persistent data per browser.

4.2 Networking Requirements

4.2.1 The System shall be hosted in a web-based environment.
4.2.2 The System shall limit all network interactions to the use of an in-game leaderboard.
4.2.3 The System shall notify the user of any inability to fetch or submit scores to the leaderboard.
4.2.4 The System shall only attempt network interactions upon the user's request.

5 Feature Requirements

This section lists all of the features to be implemented in the System.

5.1 General Features

5.1.1 The System shall begin with a main menu screen.

5.1.1.1 The main menu shall include a title, a simple logo, and instructions for how to begin a round.
5.1.1.2 The main menu shall allow the user to begin a round by pressing the space bar.
5.1.1.3 The main menu shall give the user access to a settings menu by pressing a button in the bottom-right corner of the screen.

5.1.2 The System shall allow the user to change selected game settings.

5.1.2.1 The settings menu shall allow the user to delete all persistent data stored by The System.
5.1.2.2 The settings menu shall allow the user to set the volume of in-game music.
5.1.2.3 The settings menu shall allow the user to set the volume of in-game sound effects.
5.1.2.4 Volume shall be set on a scale of 0-1. For example, 0.5 would represent half the maximum volume.
5.1.2.5 Any changes to the settings shall be persistent between sessions.

5.1.3 The System shall include in-game currency.

5.1.3.1 The in-game currency shall made available for collection during each round of gameplay.
5.1.3.2 The System shall treat in-game currency as a pickup.
5.1.3.3 The amount of in-game currency collected shall be persistent between multiple sessions.
5.1.3.4 The amount of in-game currency shall be erased upon the user pressing the "delete data" button in the settings menu.

5.1.4 The System shall include an in-game store allowing the user to purchase weapons and upgrades.

5.1.4.1 The store shall be accessible to the user after each round of gameplay.
5.1.4.2 The store shall allow the user to make purchases by clicking a UI button.
5.1.4.3 The store shall display the name, price, and description for each available item upon hovering over their purchase buttons.
5.1.4.4 The store shall not allow the user to purchase an item already in their posession.
5.1.4.5 The store purchases shall be persistent between multiple sessions.
5.1.4.6 The store purchases shall be erased upon the user pressing the "delete data" button in the settings menu.

5.1.5 The System shall introduce a scoring system to rank players based on their performance for each round of gameplay.

5.1.5.1 The user's score shall be determined by the length of their round in hours, minutes, and seconds.

5.1.5.2 The user's score shall be constantly visible during gameplay.

5.1.6 The System shall contain a leaderboard interface which allows players to submit and compare their score with other users.

5.1.6.1 The leaderboard shall be hosted as a database on https://dreamlo.com/.

5.1.6.2 The leaderboard shall be accessible to the user after each round of gameplay.

5.1.6.3 The leaderboard shall display the 10 highest scores submitted.

5.1.6.4 The leaderboard shall accept multiple scores from users submitting an existing name, keeping the highest score.

5.1.6.5 The leaderboard shall not require the user to submit their score in order to view the top scores.

5.2 Gameplay Features

5.2.1 The player shall be controlled via mouse and keyboard inputs from the user.

5.2.1.1 Pressing the directional keys during gameplay shall result in the player moving on screen in that direction.

5.2.1.2 Moving the cursor on screen during gameplay shall result in the player rotating toward the direction of the cursor.

5.2.1.3 Clicking the mouse or pressing the space bar during gameplay shall result in the player firing the current weapon in the direction the player is facing.

5.2.2 The player will be provided with vitals which dictate their ability to survive and eliminate threats throughout a round.

5.2.2.1 The health vital shall determine how much damage the player can endure without losing a life.

5.2.2.2 The default maximum health capacity shall be 100 units.

5.2.2.3 The lives vital shall determine how many times the player can exhaust their full health before ending the round.

5.2.2.4 The default maximum lives vital shall be 3 units.

5.2.2.5 The energy vital shall determine how many times the player can fire a weapon before needing to recharge.

5.2.2.6 The default maximum energy capacity shall be 100 units.

5.2.2.7 The energy vital shall be increased by 2 units every second until reaching the maximum energy capacity.

5.2.2.8 The vitals shall be constantly visible during gameplay.

5.2.3 The System shall allow the player to select multiple different weapons during gameplay.

5.2.3.1 The System shall initially supply the user with a standard laser weapon.

5.2.3.2 The System shall include a minimum of two alternative weapons in addition to the standard laser.

5.2.3.3 The System shall make the additional weapons available in the in-game store.

5.2.4 The System shall generate obstacles which damage the player upon impact.

5.2.4.1

5.2.5 The System shall generate enemies which target the player to deal intentional damage.

5.2.6 The System shall generate pickups to aid the player in their survival.

5.2.6.1 Pickups shall be collected when the player collides with them in-game.

5.2.6.2 Pickups shall have their colliders set as "triggers" within the unity engine.

5.2.6.3 The health pickup shall add 15 to the player's current health vital.

5.2.6.4 The energy pickup shall add 15 to the player's current energy vital.

5.2.6.5 The coin pickup shall add 10 to the user's available currency.

5.2.6.6 Each pickup shall play a unique sound effect when collected.

5.2.7 The System shall increase the difficulty of gameplay as the amount of time spent in a round progresses.

5.2.7.1 The progression of difficulty shall be managed by the Phase Manager module.

5.2.7.2

5.2.7.3 The time between each asteroid generation shall decrease by 20% of the current asteroid phase multiplier value of the PhaseManager each asteroid phase with a floor of an asteroid generating every second.

5.2.8 The System shall generate random events which introduce additional challenges to the player.

5.2.8.1 The events shall be managed by the Phase Manager module.

5.2.8.2 The System shall evaluate whether an event should take place every 60 seconds.

5.2.8.3 The event evaluation shall involve a 33% chance that no event will occur.

5.2.8.4 The events shall last 30 seconds.

5.2.8.5 The duration of an event shall not count toward the time until the next event evaluation.

5.2.8.6 The "asteroid belt" event shall temporarily disable enemy generation and set the Obstacle Generator module to generate an asterod every 0.6 seconds.

5.2.8.7 The "nebula" event shall temporarily reduce the player's maximum energy capacity by half.

5.2.8.8 The user shall be notified of an event taking place via a textual notification and sound effect.

6 Performance Requirements

This sections lists all performance requirements laid out for the System.

6.1 Frame Rate Requirements

6.1.1 The System shall at no time include spikes in frame rate that dip below 15 FPS.

6.1.2 The System shall limit the number of average performance spikes to a maximum of 1 spike per second.

7 Nonfunctional Requirements

This section lists the nonfunctional requirements pertaining to the System.

7.1 Software-Related

7.1.1 The System shall be developed using exclusively the C# language.
7.1.2 The System shall be designed using the Unity Game Engine.
7.1.3 The System shall be exported to the WebGL platform using Unity's included build options.

7.2 Graphics & Visuals

7.2.1 The System shall use a native resolution of 1400x700 pixels.
7.2.2 The System shall offer fullscreen support for a resolution of 1920x1080 pixels.

7.2.2.1 Any 16:9 aspect ratio display should also be supported as a result, at the possible expense of some minor graphical defects such as blurring at larger resolutions.

7.2.3 The System shall only use the .png format for images.
7.2.4 The System shall only use the .fbx format for 3D models.

7.3 Audio

7.3.1 The System shall only use the .mp3, .ogg, and .wav formats for all music and sound effects.
7.3.2 The System shall preload all audio assets prior to the beginning of gameplay.
7.3.3 No audio files shall contain greater than 1 second of silence.
7.3.4 No audio files used for music shall exceed 4 minutes in length.
7.3.5 No audio files used for music shall exceed 9MB in size.
7.3.6 No audio files used for sound effects shall exceed 4 seconds in length.
7.3.7 No audio files used for sound effects shall exceed 1.5MB in size.