

Final Project

Preliminary Report

Student Id: 210168615

Introduction

I have chosen the Game Development - Arcade Game template for this project. I will develop a video game using Unity, in a similar style to the arcade games of the 80s and 90s. The game will be a vertical scrolling “shoot-em-up” game, like *Space Invaders* and *1942*, but also incorporating elements from the “bullet hell” subgenre that appeared in the 90s, with seminal games like *Batsugun* and *DonPachi*.

The elevator pitch for a game like this would be: *Get ready to test your skills in this adrenaline-fueled bullet hell space shooter inspired by iconic games like 1942 and DonPachi.*

There are a number of characteristics that define vertical scrolling “shoot'em up” games. The first of all is the orientation of the game, the player's character points toward the upper section of the screen and most of the enemies appear from there as well, moving downwards towards the player. A moving background creates the illusion of moving forward and adds dynamism to the scene. The main gameplay mechanics are moving and firing, destroying the enemies that attack the player while also trying to avoid being hit by projectiles, enemies or other bodies. There are other possible expansions that make the genre more diverse: powerups, upgrades, special movements, special attacks, sidekicks or weapons.

But what is the bullet-hell genre? “Bullet hell” or “bullet curtain” games are a subgenre of shoot'em up games, and are characterized for having a multitude of bullets being fired at the player, in complex and visually impactful patterns, such as helixes, spirals, circles, hexagons and clouds. To balance the difficulty, the player's hitbox is smaller than in other shoot'em up games, often being a small part of the aircraft sprite. This genre is very popular in Asia, and especially in Japan, but does not have the same recognition in other countries.

The main reason I have chosen this template is due to my own personal interest in game development and in this game genre. I grew up in the 90s and have many fond memories of playing video games as a young kid. At that time a career in game development seemed impossible, but now the technology and know-how are widely available.

Another aspect I consider relevant is that these games have mechanics that can accommodate the requirements of the project and still offer many possibilities. These limitations may be useful to keep the scope of the project within the time constraints.

The input is reduced, so only a small set of actions is available, which fits a space shooter comfortably. In these games, besides moving, the player can shoot, use some items (like bombs) or special actions.

The game must be easy to pick up. To overcome this limitation I am leveraging the familiarity of a known genre. Controls will be easy to understand, as will be based on the two main standards used for playing with a keyboard: the directional arrows and the WASD keys.

Very short initial gameplay sessions due to high difficulty: Bullet hell games are inherently difficult, due to the number of moving elements on the screen, and complex patterns.

And it also proposes some interesting technical challenges: the bullets and enemies move following complex patterns, which can be solved programmatically, sometimes involving mathematical functions. There is also the difficulty of keeping a good performance when there are many elements on the screen, and making the game's components modular and extendable. Regarding game design, the biggest challenge is adding a novel mechanic to make the game interesting and engaging.

Of course, there is no sense in developing a game if there is no market for it, and while this game won't be released commercially, the public has looked for independent and small studios or even solo developers as an alternative to the expensive AAA games released by the largest companies of the industry¹. The popularity of platforms like Itch.io and Steam for PC supports this idea, as they enable small studios to have a distribution chain without the need for physical copies. The invention and evolution of portable devices like cell phones and portable consoles was instrumental to the popularity of "hyper casual" games: games that are focused on one or two game mechanics, with very simple input, and aimed at massive non-expert audiences. The proliferation of this type of games contributed to spark the interest in vintage games, which are simpler and more focused, very different from the present day standard of open worlds, multiplayer oriented and convoluted games. There is also an increasing interest in games with higher levels of difficulty, like the *Dark Souls* series, evidenced by their success and the release of sequels and new titles.

As extensions to the project I will incorporate a 2 players option, a time dilation game mechanic (that will also help beginners to get used to the game), and a variety of enemies as wide as the time limitation allows.

Literature Review

In this section, I'll address the following topics:

What was the Golden Age of Arcade games? Which games from this period were most relevant and influential? And what was innovative or disruptive about them? I have taken the The Golden Age of Arcade Games.first section of the book "*The Golden Age of Video Games - The Birth of a Multibillion Dollar Industry*"² as reference.

What methods or frameworks can we use to analyze these games? I took the article "*MDA: A Formal Approach to Game Design and Game Research*" as reference for this topic, and analyzed the game DonPachi using this framework.

¹ Anonymous, Accessed 2024-01-08. Wikipedia: Indie Game
[https://en.wikipedia.org/wiki/Indie_game#Fears_regarding_saturation_and_discoverability_\(2015%E2%88%92present\)](https://en.wikipedia.org/wiki/Indie_game#Fears_regarding_saturation_and_discoverability_(2015%E2%88%92present))

² Dillon, Roberto, 2011. *The Golden Age of Video games - The Birth of a Multibillion Dollar Industry*. A K Peters/CRC Press, USA.

What is the bullet hell genre? I have read Andrew Fan's guide³ to have a better understanding of the genre.

The Golden Age: When? Where? What games?

It's not easy to determine when the Golden Age started, authors don't agree about the initial milestone, so the year can vary in the 70s, and some of the events that are considered as the most relevant are: the release of *Computer Space*, the first arcade video game machine (1971), the release of *Space Invaders* (1978) and the release of *Asteroids* (1979) one of the first games using vector graphics. There is more consensus about the end of the Golden Age, and it is usually associated with the crisis of the video game industry that took place between 1983 and 1985.

It took place all around the world, but the main centers of research and development were the United States and Japan. Other countries, depending on their economic and academic power, were either secondary centers (as many countries in Europe) or simply consumers of these new technologies, as Latin American countries.

Many influential games were developed and released during this period, some of them are still very popular and are still references for many game designers and companies (especially in the mobile gaming industry).

Some of the most important games developed in the USA were: *Asteroids* (Atari, 1979), *Centipede* (Atari, 1981), *Defender* (Williams Electronics, 1981). And in Japan: *Space Invaders* (Taito, 1978), *Donkey Kong* (Nintendo, 1981) and *Pac Man* (Namco, 1980).

Among the most innovative aspects we can mention: *Space Invaders* opened up the themes and game mechanics, most previous games were based on sports like *Pong* and *Gran Track 10*, it presented a feedback loop (the multiple lives per play), an evolving environment, and a high score, which greatly enhanced competitiveness. *Donkey Kong* introduced characters that were clearly individualized and a very simple and easily understandable narrative, and was one of the first platforming games. *Pac Man* introduced a new type of interaction, where the player needed to avoid enemies in a maze and could obtain the power to temporarily eliminate them. Pac-man was also a game where the enemies were clearly individualized by their color and name, and even they had different behaviors.

The hardware that powered the arcade machines also evolved fast in these years: vector and color graphics allowed users to visualize more complex designs, generating the illusion of 3D movement and 2D graphics with a higher level of detail, characters could have a clearly defined image and more personality (like the ghosts of Pac-Man). This also influenced what kind of games could be developed. The evolution of home systems, both computers and video game consoles, gave birth to third party software development, with Activision being one of the first companies dedicated exclusively to game development.

But we could argue that besides the technological aspects that fueled these video games, the creative minds behind them installed video games in popular culture and set the foundations of what would be one of the biggest industries in entertainment.

³ Andrew Fan, accessed 2023-12-01, Andre Fan's Code Dump. <https://sparen.github.io/>

How can we analyze these games?

Video game studies are a controversial area because there is not a unified definition of what a game⁴ is (as seen in A book of Lenses, chapter 4 and Game Design Workshop chapter 2), its scope and method are not clearly defined yet, and there is not even a consensus on where this study should be located, if it's part of Humanities, as multimedial works of art, in social sciences as a cultural phenomenon, or as a multi-disciplinary study that is not scientific in itself. For a long time, the only widely available articles talking about video games were the critical reviews that some experienced players wrote for magazines, which were mostly subjective, even if they tried to adhere to internal standards, like pointing systems and common areas to review (graphics, sound and music, replayability, game modes, etc).

One framework designed to study video games is called MDA⁵ (Mechanics, Dynamics and Aesthetics), and was proposed by Hunicke, LeBlanc and Zubek, in 2004. It's not the only framework, and it's not accepted unanimously by the community, but it can help us understand some aspects of the games.

The framework breaks down the games into smaller components: Rules - System - "Fun", which are the ways the player can perceive the game, and their counterparts from a design perspective: Mechanics - Dynamics - Aesthetics. The mechanics describe the components of the game, in a very low level of detail, like data structures and algorithms. Mechanics can be described as the actions that the game allows for the player and AI. The dynamics describe the behavior of the mechanics on a real play session, how they interact with other systems and with the player's input, ie. how the actions can be used in real life scenarios (rocket jumping, for example, is a possible move, but it wasn't designed intentionally, it's just a natural consequence of the interaction between the game's mechanics and the players' creativity). Finally, aesthetics are the expected emotional response from the player, evoked when she or he plays the game, and the product of the dynamics.

The framework includes eight categories that describe how a game is perceived by the player, from the immediate "sense-pleasure" and "challenge" to the more abstract "fellowship" and "discovery". The game can evoke multiple different sensations in the player: graphics, sound and music appeal to the dimension of the game as they directly stimulate the player's senses, but also if the game is cooperative (either with other players, locally or online, or AI), it can also create a sense of community or fellowship, thus being a social framework.

How can we use the framework to analyze shooter games?

As an example, I'll try to apply the framework to analyze the game "DonPachi". The first step is finding out what are the most relevant categories in the aesthetics dimension that we have to consider.

⁴ Some examples of definitions can be found in Wikipedia's page for Game. Anonymous, accessed 2024-01-08. Wikipedia, Game. <https://en.wikipedia.org/wiki/Game#Definitions>.

⁵ Hunicke, Robin & Leblanc, Marc & Zubek, Robert. (2004). MDA: A Formal Approach to Game Design and Game Research. AAAI Workshop - Technical Report.



Images: In game screenshots from DonPachi.

Sensation. The game uses an abundance of color and rich textures to be visually appealing, the aircrafts, tanks and scenarios were carefully designed to keep a consistent futuristic style, there are multiple visual effects, different types of shots, bombs, explosions required by the genre. There are plenty of sounds, like gunshots, yells and explosions, that highlight the actions of the scene, and the fast paced music conveys a sense of urgency. Both the images and sound help to create an immersive experience.

Fantasy. The game presents a fictitious situation, where the player represents the only pilot able to face the challenges that the enemies present. While the plot is not very relevant to the game's continuity, the sense of danger is present throughout the whole game and the threat is constant, which is consistent with the back story.

Narrative. Most of the game is linear, with the missions ending and starting without any cutscene or dialog. This game is clearly not story driven, and the player doesn't have any say in the game's outcome.

Challenge. This is the focus of the game, the player's reflexes and skills are tested constantly and the difficulty of the game increases every mission.

Fellowship. The game allows 2 players to participate simultaneously, while it can be very helpful to have a comrade joining the fight, the sense of companionship admits only one additional person.

Discovery. The game's world is limited and doesn't admit any independent exploration.

Expression. The game doesn't present any opportunities for the player to affect the character's journey, or to customize the character in order to make it more adequate to his or her playing style.

Submission. The game is engaging, as it keeps the player's focus throughout the whole session. The challenge, the sound effects and the constant movement, make a very immersive experience.

How are the most relevant aesthetics of DonPachi modeled?

Enemies' behavior has two components: movement and attack (mechanic). The combinations of different movement and attack patterns generate different "personalities" for the enemies (dynamic). And depending on how they behave and how many of them there are in a group, the challenge the player faces will increase or decrease (aesthetic). If the enemies fire constantly and move too fast, the player won't be able to evade them (dynamic), creating a feeling of powerlessness and frustration. On the contrary, if the enemies are slow and don't fire (dynamic), the player may get bored quickly (aesthetics). In the case of DonPachi, there's a balance: some enemies move slowly but shoot faster, bigger rounds or homing projectiles, while other enemies move faster, but have limited shooting capabilities. In general, the combination of different kinds of enemies makes the game fast and keeps the player's attention, and prevents it from becoming monotonous. Projectiles' physics (and the aircraft's too) is not intended to be realistic, bullets normally would move much faster than in the game, even considering the difference in scale, and the controls of even the most modern aircrafts are most likely not as responsive as shown in the game (dynamics). This is done to make the game accessible and to create diversity (aesthetics). Regarding sensations, the game features a colorful palette, stereo sound and background music (mechanics). The player can quickly see the bullets, because their color contrasts with the background (dynamic) and thus is able to avoid them, sounds like explosions and gunfire are signals that inform the players of the game's status, this makes the player's task a bit easier and enhances the immersiveness of the game, as the game involves different senses at the same time, and can also be used to call the potential players' attention when they walk through the arcade (aesthetics).

What are bullet hell games?

There are not many resources to design and develop videogames of specific genres. There are some examples of the most popular game genres, from official sources as Unity and Unreal and from indie developers, and tutorials that cover the basics of each video game engine in the market, but even tutorials that orient people to "develop" their own FPS or 3D platformer do not dive deep into how to create game mechanics that will make a game unique or how to design the levels to make it engaging, or how to balance the difficulty of a puzzle or strategy game, meaning that the most relevant topics of game design are not incorporated in these resources. Even though there are many bullet hell games, and it's a genre that started over twenty years ago, there is no authoritative body of knowledge or design guide or even a generally accepted definition of the genre, what components are required by it, how to use them. Other genres, even if they lack proper documentation, have at least examples so known that people immediately understand the reference (there are so many FPS and platformers that people understand how most mechanics work, and many movements and techniques have widely known names like double jump, wall jump or rocket jump).

In this website, Andrew Fan tried to create a simple and flexible vocabulary to describe different patterns in *danmaku* and enable designers to discuss and analyze games of this particular genre⁶, and also help beginners understand how to design a game that is properly bullet hell.

⁶ Andrew Fan, accessed 2023-12-01, Andre Fan's Code Dump. Sparen's Danmaku Design Studio. URL: <https://sparen.github.io/ph3tutorials/ddsg0.html>

He proposes: “The first thing to note is that the differentiating factor [between regular shoot’em ups and danmaku] is NOT the number of bullets. You can have a lot of bullets, but if they aren’t thoughtfully used, the pattern isn’t exactly a bullet hell.”⁷ There is a visual / aesthetic component that other games don’t have, and it makes a big difference in how these games are designed and developed.

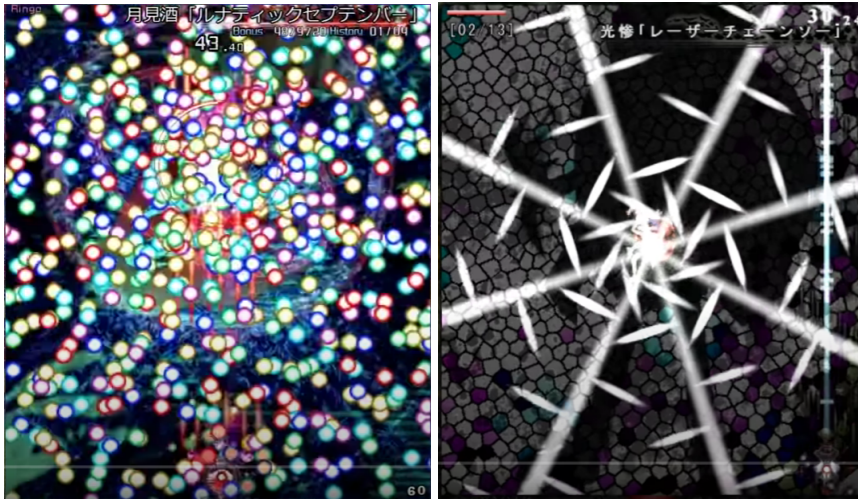
This guide starts from the most simple element, the bullet, and works its way up to complex patterns (bullet groups like rings and stacks) to slowly build an understanding of how small differences in parameters like speed, spacing in a group, angles and number of bullets can have a big impact in games as fast-paced and complex as bullet hell games, both from a gameplay and an aesthetic perspective. For example, one of the first elements is the form of the bullets, and Andrew makes the point that depending on the movement of the bullet, a bullet with an angled shape may be better than a round-shaped one, as the player can perceive the direction of the bullet faster when there’s a visual hint.



Image: In game screenshot from DonPachi, displaying 4 different kinds of enemy projectiles, each with their own movement style.

There are many elements to consider when designing a video game, even in familiar genres, and they lead to different approaches. In this kind of game, these elements can be used to design games with very different playing styles: slower games with higher bullet density or more complex patterns that rely on precise movements, faster games with smaller bullet density that rely on reflexes and shooting. This guide’s last concepts of bullet density and pacing, and spatial density, can be very helpful for developers to understand what makes these games work, and how to experiment and create new challenges.

⁷ Andrew Fan, accessed 2023-12-01, Andre Fan’s Code Dump. Sparen’s Danmaku Design Studio. URL: <https://sparen.github.io/ph3tutorials/ddsga2.html>.



Images: Left - Screenshot from Touhou 15 - Legacy of Lunatic Kingdom.⁸ Right - Screenshot from Len'en 2: Earthen Miraculous Sword.⁹

Developing a good game usually requires a long time, a deep understanding of the genre's main mechanics, and several rounds of playtesting. Not all genres have this kind of material, but it is only an introduction and can only help to learn more from experience, make more insightful analysis and have a slightly better grip of how to tune the difficulty.

Design

Project's domain and users

The domain of the project is arcade action games, and the target users are video game players, from teenage to adults. As the game's difficulty level will be elevated, it's not likely to be played by casual players, most likely the audience will be gamers that are already interested in the genre.

Users and Domain's Requirements

One of the most important requirements of this game genre is to focus on the enemies and their attack. This is a part of the genre's definition, and to fulfill it, I decided to make both as configurable as possible, using the inheritance of the classes in C#. There's an abstract class with the most basic variables and functionality, and it's overridden by multiple implementations (movement, spread shots and enemies). And to store the configuration of these multiple elements, I'm using Unity's scriptable objects. Scriptable objects allow the creation of assets to store values and share them among multiple objects. Another useful resource for this requirement is Unity's modular and compositional structure; different

⁸ "Jaimers", accessed 2024-01-08. Youtube. "Touhou 15 東方紺珠伝 ~ Legacy of Lunatic Kingdom - Legacy Lunatic No-Bombs 1cc". URL: <https://www.youtube.com/watch?v=9D6fj5AUmnM&t=523s>

⁹ "MegamanOmega", accessed 2024-01-08. Youtube. "Len'en 2: Earthen Miraculous Sword - Extra Stage (No Commentary)". URL: <https://www.youtube.com/watch?v=vq2tlX-p9Sk&t=252s>

aspects of a unit's behavior can be developed independently and added as components of a game object.

To have a visual impact, different types of movement must be implemented, and they have to be used in forms that have an aesthetic value, present a challenge to the player, but are not of an impossible difficulty. Balancing the difficulty, while still making the enemies attractive will require some extra time.

Project's Structure

A Unity project has multiple elements: handling the player's input, the main character's health and score, a graphic interface, sounds and music, projectiles, enemies and their attacks. This is backed by C# scripts, art assets like sprites and sounds and prefabs. This is a list of some of the most important components of the game.

For the player:

Player Controller: receives the input and processes the corresponding actions.

Player Stats: Keeps the player's data: attributes, score, health points, and is used to inform the GUI.

Prefab: The player's ship with all the necessary components for it to move and interact with other objects.

Power ups: Healing and fire power increases.

Bullet controller: Collision logic.

Bullet prefabs: packages with different properties, to instantiate objects depending on the firing character's capabilities.

Bullet spreads: groups of bullets created at the same time, settings for different implementations (radial and linear groups of bullets).

Attack patterns: settings for bursts of bullets fired by the enemies.

Movements: Abstract class and multiple implementations to manage the bullets and enemies' movements.

Game Controller: Controls some events of the game flow, like starting the game and ending it, player's respawn and GUI.

Time Controller: Controls the time dilation logic.

Level Controller: Controls the enemy spawning.

For the enemies:

Enemy prefabs: Packages with the enemies properties embedded.

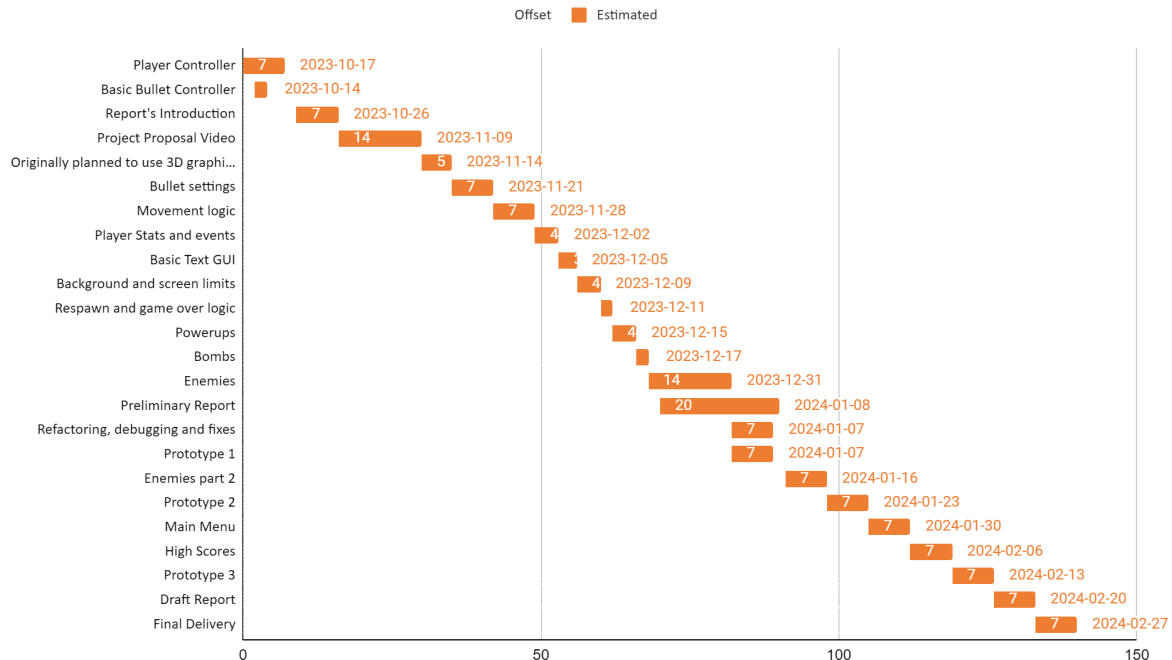
Enemy controllers: An abstract class with the base properties and methods, and different implementations for different enemies.

Work Plan

I have divided the work into smaller tasks and grouped them according to their content. I haven't set strict deadlines besides the milestones of the module, as in many cases I would have to adapt to accommodate both my other responsibilities and all the learning and

research done while advancing on the project. The main deadlines are the Project Proposal Video, Preliminary Report, Draft Report and Final Delivery.

Task estimation



Graph: Gantt chart with the groups of tasks and their estimated duration.

Project's testing and evaluation

I have planned the release of one prototype of the project before the delivery of the preliminary report to test the basic gameplay elements of the game. Then I'll release versions with smaller changes to iterate on the level and enemy design, and to add the remaining functionality. Feedback will be considered to make further improvements.

Feature Prototype

What was included in the prototype?

The first prototype has all the basic actions the player can take: moving, firing, using a bomb, and the time dilation special action, a player character with limited health points, and 3 types of enemies with their unique behaviors and firing patterns.

Some of the most important pieces of code are already written for this prototype: the player controller, which is responsible for handling the input, actions and the status of the player character; the game controller, which helps handle the game status; multiple additional classes that help with the configuration and instantiation of bullets and enemies, and their movement; and some of the enemies' controllers, which determine their movements and

their actions. The prefabs in this instance are based on primitives, circles, squares and some customized polygons.

The GUI is still very basic, with text indicators of the player's health points, lives and ECD status (ie, the special ability's readiness).

The enemies' movements and firing patterns are basic and will need tuning and playtesting to balance the difficulty of the game.

What was out of scope for the prototype?

The graphic interface will be improved, to have easier to read indicators, like a health bar and another one for the special ability showing both the cooldown time and the duration of the effect.

Visual and sound effects like animations, explosions and music will be added later.

At this point, there is no option for a second player yet, it will be developed in the next few weeks. The title screen and an options menu will also be developed for the second big iteration of the game.

The prefabs have placeholder sprites made from Unity's primitives and will be replaced with more elaborate ones from Unity's marketplace.

Did it work as it was expected from the planning and design?

Yes. Of course, this is a first prototype and only includes the most basic functionality. As discussed before, it doesn't have many elements that will contribute to its appeal: visual and sound effects, adequate balance of challenge presented by enemies and different types of enemies and attacking patterns.

What feedback was received from play testing?

There was positive and negative feedback. The things that the players noticed the most are the lack of sound effects and the low quality of the prefabs (since they are place holders for real graphic assets). The color palette is limited, and could be expanded to make objects more identifiable. The difficulty was hard to assess: some players found it too hard, some players found it adequate after a few tries, it's subjective and requires several tries to fix it. One remark that some people made is that the playing felt too small, this is one aspect that was overlooked but I'll fix in the next deliveries. There was also a bug reported, when the special skill is still active and the player loses a life, the slow motion effect doesn't get disabled automatically. I'll include this bug in the backlog. Another negative aspect was that the bullets that go off screen may reappear (if they have the ability to change direction) and could hit the player unexpectedly. This is unfair, since the player loses sight of those bullets and is incapable of detecting them early. Gameplay in general was ok, the speed felt adequate and both the bomb and the time dilation ability were said to be good resources to overcome the challenges the game presented.

Since this prototype was very short, there were no powerups in it, but they will be added as part of the level design. Also, the enemies do not display the "bullet hell" firing patterns, since they are very basic and new enemy types need to be added.

References:

Andrew Fan, accessed 2023-12-01, Andre Fan's Code Dump. URL: <https://sparen.github.io/>

Anonymous, accessed 2024-01-08. Wikipedia, Game.
<https://en.wikipedia.org/wiki/Game#Definitions>

Anonymous, Accessed 2024-01-08. Wikipedia: Indie Game. URL:
[https://en.wikipedia.org/wiki/Indie_game#Fears_regarding_saturation_and_discoverability_\(2015%E2%88%92present\)](https://en.wikipedia.org/wiki/Indie_game#Fears_regarding_saturation_and_discoverability_(2015%E2%88%92present))

Dillon, Roberto, 2011. The Golden Age of Video games - The Birth of a Multibillion Dollar Industry. A K Peters/CRC Press, USA.

Hunicke, Robin & Leblanc, Marc & Zubek, Robert. (2004). MDA: A Formal Approach to Game Design and Game Research. AAAI Workshop - Technical Report.

"Jaimers", accessed 2024-01-08. Youtube. "Touhou 15 東方紺珠伝 ~ Legacy of Lunatic Kingdom - Legacy Lunatic No-Bombs 1cc". URL:
<https://www.youtube.com/watch?v=9D6fj5AUmnM&t=523s>

"MegamanOmega", accessed 2024-01-08. Youtube. "Len'en 2: Earthen Miraculous Sword - Extra Stage (No Commentary)". URL:
<https://www.youtube.com/watch?v=vq2tIX-p9Sk&t=252s>