Tampereen ammattikorkeakoulu

Video Game Difficulty and the Intended Player Experience

Lauri Rantala

OPINNÄYTETYÖ
Joulukuu 2021

Tietojenkäsittely
Pelituotanto

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittelyn tutkinto-ohjelma
Pelituotanto

RANTALA, LAURI:
Video Game Difficulty and the Intended Player Experience

Opinnäytetyö 33 sivua, joista liitteitä 1 sivua
Joulukuu 2021

Työn tavoitteena on kartoittaa, kuinka vaikeusasteita on lähestytty erilaisissa videopeleissä ja pelitutkimuksessa sekä selvittää, millainen suhde tällä on pelisuunnitteluun tavoitellun pelikokemuksen kanssa. Työn tarkoituksena on luoda 2D-toimintapeliin vaikeusaste, joka vastaa mahdollisimman monella pelaajalla tavoiteltua haastetasoa. Merkittävä osa videopelien vaikeusasteisiin keskittyvistä tutkimuksista hyödyntää Mihaly Csikszentmihalyin flow-teoriaa. Flow-teoria on yksinkertaistettuna teoria immersiosta, jonka ihmiset saavuttavat tehdessään heitä sopivalla tavalla haastavia toimintoja. Liian haastavat toimet aiheuttavat ärsyyntymistä ja liian helpot tuntuvat tylsiltä. Vaikka peleillä on paljon yleisiä lähestymistapoja vaikeusasteisiin (esim. pelaajan mahdollisuus valita vaikeusasteen taso), pelien erilaisuudesta johtuen jokaisen pelisuunnittelijan on lähestyttävä vaikeusasteita oman pelinsä tarkoitusperien kautta. Kun luotiin kehitteillä olevaan peliin vaikeusastetta, hyödynnettiin kolmea yleisintä vaikeusasteiden lähestymistapaa: manuaalista, dynaamista ja sulautettua tapaa. Näitä tapoja käsiteltiin pelin omien mekaniikkojen kautta ja yritettiin luoda mahdollisimman tasainen vaikeusaste pelaajan kokemustasosta riippumatta. Peli testattiin neljä kertaa, ja sitä muokattiin jokaisen testauksen jälkeen saadun palautteen perusteella.

Pelaajien pelikokemustasosta riippumatta peli kehittyi jokaisen testauksen jälkeen lähemmäksi tasapainoista kokemusta. Testaukseen ja niistä saatuihin tuloksiin vaikuttivat testaajien pieni määrä ja pelissä havaitut häiriötekijät. Nämä häiriötekijät aiheuttivat peliin haastetta tavoilla, joita ei ollut haluttu. Monet keskivaikeusasteella pelaavat kokivat pelin hieman helpoksi vielä kaikkien testausten ja muutosten jälkeen. Suuri osa kehityksestä olikin käytetty joko helpoimman vaikeusasteen joustavuuteen tai keskivaikean helpottamiseen. Jatkokehityksenä keskityttäisiin enemmän dynaamisten vaikeusasteratkaisuiden joustavuuteen haasteen kasvamisessa, jos pelaaja pärjää pelissä liiankin hyvin. On myös tärkeää huomioida, minkä pelissä halutaan olevan haasteellista ja minkä ei, sillä suunnittelemattomat haasteet aiheuttavat usein pelaajissa ärsyyntymistä.

Asiasanat: video pelit, haaste, flow, pelisuunnittelu

ABSTRACT

Tampere University of Applied Sciences
Degree Programme in Business Information Systems
Game Production

RANTALA, LAURI:
Video Game Difficulty and the Intended Player Experience

Bachelor's thesis 33 pages, appendices 1 pages
December 2021

The objective of this thesis was to map out different ways in which video game difficulty has been approached in game design and studies. After researching some of the most common forms of difficulty design, the purpose was to create a difficulty system for a 2D action game, which provides the desired challenge level regardless of the players gaming habits. Mihaly Csikszentmihaly's theory of flow and three of the most common difficulty design conventions (manual, dynamic and embedded) were used to design the difficulty system. These design conventions were used through the games own mechanics.

The game was tested four times with changes made after each testing based on the feedback. The tests utilized quantitative and qualitative methods for gathering feedback. After each testing the feedback from players got increasingly consistent regardless of their gaming habits. Some players who chose the medium difficulty still found the game to be easier than intended. Future development time should be spent developing dynamic systems that make the game harder if the player is performing better than expected. It is also important to test whether the challenge of the game comes from the designer's intended designs.

Key words: video games, difficulty, flow, game design

TABLE OF CONTENTS

GLOSSARY

DDA	Dynamic difficulty adjustment
Mod	User made modification to a game
Roguelike	Game genre influenced by the game Rogue (1980) with similar mechanics, including permadeath and turn-based combat.
HP	Health points
RPG	Roleplaying game

1 Introduction

Challenge is consistently considered one of the cornerstones of game design. It is so important that there are some players who do not consider something a game unless there is a challenge to overcome, and a failure state to accompany it. When talking about difficulty in video games, people are usually talk about ”mechanical difficulty” as Patrick Jagoda puts it (Jagoda 2018). Mechanical difficulty refers to the required hand-eye coordination and decision-making skills needed to over-come the challenges presented in the game (Jagoda 2018). Difficulty can be a delicate balance, and players often prefer winning, but making the game too easy can also make it boring and uninteresting for the player (Schmierbach, Chung, Wu and Kim 2014).

Video game industry has been steadily growing and diversifying every year. The majority of the player base is no longer represented by the so called ”hardcore-gamer” (ISFE 2020). There have been many different approaches over the years for tackling the game difficulty for a diverse group of players. Three most common ways are usually called: *manual*, *dynamic* and *embedded* difficulty (Smeddinck, Mandryk, Birk, Gerling, Barsilowski and Malaka 2016)

Usually, the discussion and research around game difficulty is centered around the theory of *flow*, which is a theory concerning immersion people have in the action presented (Csikszentmihalyi 1990). The action can be anything from work to artistic pursuit, but it is often used as the backbone when researching the effects of game difficulty and player enjoyment.

2 Flow

2.1 Background on flow theory

The concept of flow is taken from Mihaly Csikszentmihalyi's research into what he called "optimal experience" (Csikszentmihalyi 1990). This research has been used in many fields, with game design being one of them. Csikszentmihalyi describes the ways in which we can attain "total involvement". It is usually attained when a person has "a sense that one's skills are adequate to cope with the challenges at hand, in a goal-directed, rule-bound action system that provides clear clues as to how well one is performing." (Csikszentmihalyi 1990, 71). Usually, flow involves the person losing the sense of time by becoming deeply invested in the action presented.

Some of the most important elements for flow, based on Csikszentmihalyi's research, are challenge, clear goals, feedback, and a sense of control. Flow is achieved when the challenge and skill are in balance with clear goals and control. Too much challenge can turn to frustration and anxiety, but a lack of challenge can turn to boredom (Csikszentmihalyi 1990). These elements are also very important in game design, and this is one of the reasons flow has become an important factor when talking about game difficulty.

2.2 Flow in games

The most influential work on flow theory in games has been Jenova Chen's thesis Flow in games (Chen 2006). In the thesis, Chen implements Csikszentmihalyi's ideas on flow into game design and particularly into the design of the difficulty system of a game called *flow*. Chen's interpretation of flow, as it relates to game design, is that a game must be intrinsically rewarding, the game offers right amount of challenge to match the player's skill level, and the player must feel a sense of personal control over the game activity (Chen 2006).

Chen thinks that dynamic difficulty or DDA, is an important part for designing a game, which would have a flow state, but it is only a part and cannot reach flow by itself. Instead, Chen states that a form of embedded difficulty is needed for flow to adjust to differences in players' gaming experience. If possible, these embedded choices might be monitored by a form of DDA to detect when it is time to offer these choices for the player (Chen 2006).

Chen's design for a wide variety of players to reach flow is "a wide spectrum of gameplay with different difficulties and flavours", "DDA system to allow different players to play in their own paces" and "Embed DDA choices into core gameplay mechanics" (Chen 2006, 14). Pictured below is Chen's figure of the flow zone in games (image 1).

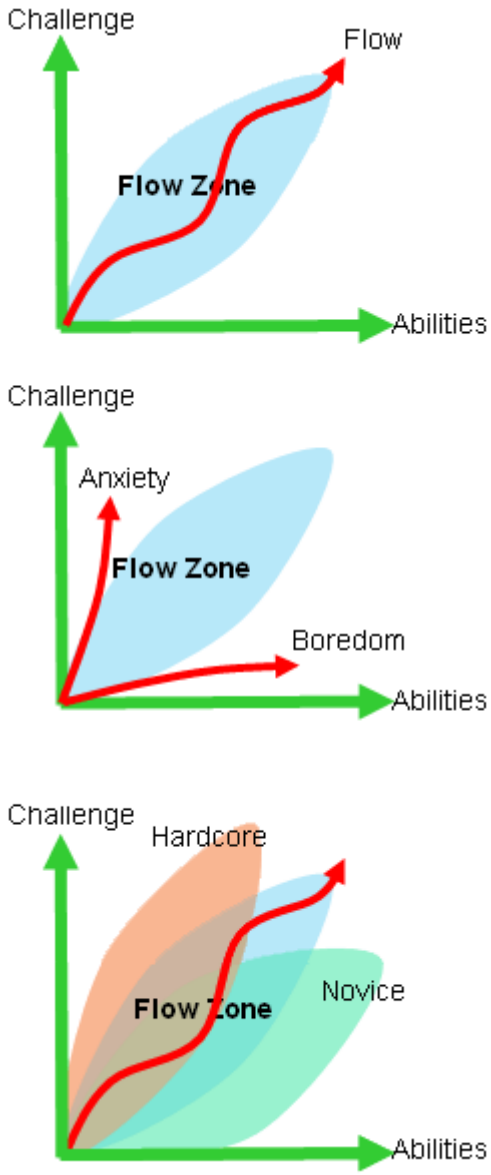

IMAGE 1. Flow in games (Chen, 2006).

3 Forms of difficulty design

3.1 Manual Difficulty

Manual difficulty adjustment is one of the ways to design difficulty. It is most often presented in the game UI through a selection menu. *Easy, normal,* and *hard* are the typical selections, with many games having different variations on these three and some games offering additional difficulties. The idea behind the manual difficulty adjustment is that the game designers can design an experience with a specific difficulty progression, often done in a linear way. The game usually starts relatively easy and gets progressively harder as the player gets through the game. There is a certain specificity to manual difficulty, which is very hard to achieve with the other forms of difficulty design. This form of design, however, can leave many blind spots when it comes to player experiences, since the skill and experience of players may vary a lot. Manual difficulty does not provide an optimal difficulty for many people, but with enough difficulty options it might come close to it.

There is some form of naming convention behind the difficulty levels, but this does not extend to their effects since games can be very different from each other. Easy mode in some games can make the player invulnerable and offer a way to enjoy the games story without having to worry about any challenge, and in some games even an easy mode can be a difficult challenge. Although there is some common way of naming difficulties: *easy, normal, standard, hard* and so on, there are games that have very different names for their difficulties. Pictured below is the difficulty selection screen of Doom for example (image 2). When it comes to easier difficulties some players may feel like they are shamed for choosing a specific difficulty if the difficulties are named in a more provocative way. This was a criticism leveled at Wolfenstein II: The New Colossus and its difficulty naming scheme (Gach 2017). The game's very easy difficulty is named "Can I Play Daddy" for example.

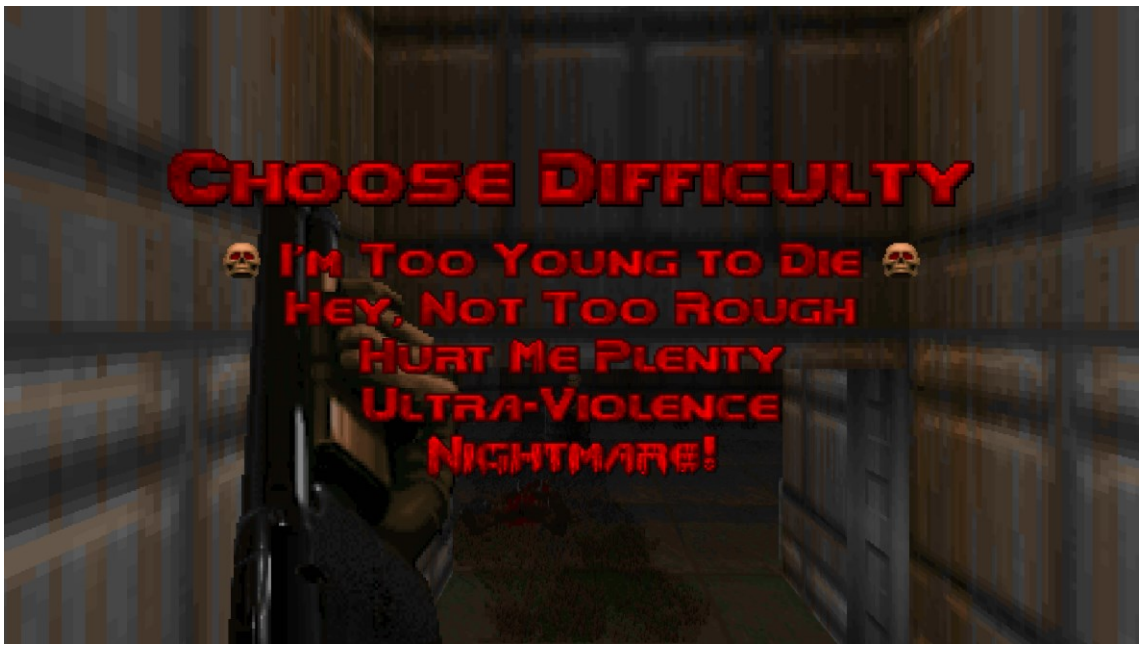

IMAGE 2. Doom (1993) id software.

Many games have different iterations on manual difficulty settings.

3.2 Dynamic difficulty

Dynamic difficulty is getting increasingly popular as time goes on. The idea is that through the data gathered from the user experience, the difficulty can be adjusted for the specific player's skill level (Zohaib 2018). Each game has its own different metrics, which are monitored and used for difficulty adjustment. These can range from player character deaths, health, resources, to time player spent playing the game. The designers need to decide what are the most useful metrics in their game for evaluating player performance.

Dynamic difficulty can hinder creating a sense of progression since the difficulty is adjusted on player behavior. A sense of accomplishment from overcoming a challenge can easily be robbed from players by a poorly designed dynamic difficulty. The game must monitor the player's skills improving or falling and keep the game difficulty in a balance. The game also must keep these changes from the player and to appear consistent (Zohaib 2018).

Dynamic difficulty can also be used to create tense situations, since the difficulty can be changed if the players experience is getting too easy (Zohaib 2018). This can be seen for example in Nintendo's series of racing games: *Mario Kart* as the non-player drivers get their speed adjusted if the player gets too far from them. This is often called rubber-banding in racing games. Nick Melder writes that rubber-banding is "when an AI-controlled vehicle gets too far in front of the player, it will slow down to allow the player to catch up and, similarly, AI-controlled vehicles behind the player will gain a boost to their speed to help them catch up to the player" (Melder 2013, 507). Sensible parameters should be used when creating a rubber-band system, and it should be avoided when player is close to the computer-controlled drivers, to avoid making the effect too noticeable (Melder 2013). DDA is also used in Capcom's survival horror series *Resident Evil* to create a tense experience by adjusting the available resources based on how well the player is doing. DDA also ensures that no matter how badly the player is doing, they will not be locked out of beating the game. DDA often works together with some form of manual difficulty system and in *Resident Evil 2* (image 3), the game will keep some DDA systems on the most difficult level all the time if the player chooses *hardcore* difficulty when starting the game (Saunders 2019).

IMAGE 3. Resident Evil 2 (2019) Capcom.

Some of the Resident Evil games have dynamic difficulty systems to change how much resources are available to the player.

3.3 Embedded difficulty

Embedded difficulty means building the difficulty options into the core experience of the game rather than having a menu full of options. The idea is to provide immersive decision making rather than explicit meta level options for the player. Many games have these kinds of decisions for players to make, for example upgrade trees and leveling up to make the player character stronger. This form of levelling up to make the game easier is usually referred to as "*grinding*" and is popular in many Japanese role-playing games (image 4). Embedded form of difficulty design can give the player a higher sense of control and authority for their own experience. There is also a possibility for higher sense of frustration from the player (Smeddinck, Mandryk, Birk, Gerling, Barsilowski and Malaka 2016).

Embedded difficulty is one of the pillars for the flow state in games according to Chen. His game *flow,* that was a part of his thesis had an embedded difficulty system implemented in it. In the game the player controls an organism in a surreal biosphere and consumes other organisms to evolve. "By swimming closer to or farther away from other organisms, and eating different types of food, players subconsciously balanced their Flow experience." (Chen 2006, 16).

After the release of the action-adventure game *Sekiro* there was a wide discussion in the gaming industry on the game's difficulty, and one of the reasons cited for *Sekiro*'s extreme difficulty was that the developer From Software had left out many mechanics present in their previous games, which might make the experience much easier. Their previous games did not have manual or dynamic difficulty options, but instead they had embedded options for making the player's journey easier. For example, help could be asked from other players. Removing these options made the game too hard for many people, who were fans of the company's previous work (Thier 2019).

IMAGE 4. Yakuza 7 (2019) Sega.

Yakuza 7 moved towards a more roleplaying game approach in its design than was typical for the series up to that point. Now the player could make the game easier by leveling up their character, as they were beating opponents on the streets.

4 Intended experience and difficulty

4.1 Player and developer fixes

Challenge has always been a part of video games and many developers have made different forms of difficulty, part of their intended game experience. Difficulty has also proven to be a challenge for many developers, as there is no way of knowing what the player will find a good challenge and what will cause more frustration. Some games have gotten player made difficulty mods to “fix” their difficulty options. One example of this is the tactical game *XCOM 2* which originally had turn limits to encourage more aggressive play, but many players found that to be off-putting, so a mod was made for the game to remove this feature as the game did not include an option to turn it off (Shaska 2017).

Cheat codes were originally used during development to help test out the game, but some developers have found other uses for them (Silverberg 2020). Some developers have left these cheat codes in their games as a way of preserving the intended difficulty level, but also to offer players a clear unintended way of playing the game. On PC some players have made *trainers* for different games. These are application that can alter the games code to give the player benefits such as invulnerability, unlock progression items and so on.

4.2 Player encouragement

Some forms of behavioral psychology are often used to view and design systems that try to influence player behavior (Hopson 2001). Encouraging certain type of play from the player can sometimes be in opposition to making the game as welcoming as possible for many kinds of players, but some developers have found ways around this problem. Using negative reinforcements (punishments) should often be avoided (Stout 2015). Making difficulty affect the score of the player has been used by developers to avoid making the game too inaccessible to some people, because the game can still encourage certain type of play but allow the player the freedom of choosing a less challenging difficulty. The Capcom action-

game series *Devil May Cry* gives players items which can revive the player if they get defeated during a level but using these items will lower your score at the end of the mission. Arcade type games can have score multipliers based on difficulty, so that players can beat the game but if they want a better score, they will have to up the difficulty. “Bullet hell” *Touhou* games usually have their “good ending” locked behind a certain difficulty but otherwise the player can choose easy, and they can even select unlimited lives. These types of options can also encourage the player to replay and learn the game better.

Hitman is a series of games where the player must sneak around big environments full of armed guards and civilians. The player can wear disguises, hide, or even engage in combat with the guards. Since the game is designed around the player being stealthier and the developers feel like that is the experience, they want the player to have, their difficulty levels mostly affect the sneaking aspects of the game. For example, how easily the player is spotted, or after being spotted how quickly the player can escape their pursuers. These difficulty options have almost no effect on how fragile the player is in combat situations, since the developers want to encourage a stealthier approach from the player (Brown 2016). There is even a ranking system at the end of the level based on how stealthy the player was.

4.3 Challenge as a core experience

There are certain games that have some amount of player frustration as the intended experience. In these cases, the balance of when the game is frustrating enough and when it gets too frustrating, can be very difficult. *Darkest Dungeon* and *Pathologic 2* are two examples of games that have frustrating amounts of difficulty as part of their core design. Both games however offer difficulty options to make them easier, they just warn the player that changing the difficulty is against the developer intent and certain amounts of frustration is part of the design (image 5). *Pathologic 2* developer Ice-Pick Lodge wrote a post on steam about their difficulty system and how they wanted the players to balance their own experience within the framework of what they would call the intended experience. In the post they talk about how turning the game too easy would rob the

player of the meaningful choices that they wanted to explore with the experience (Ice-Pick Lodge 2019).

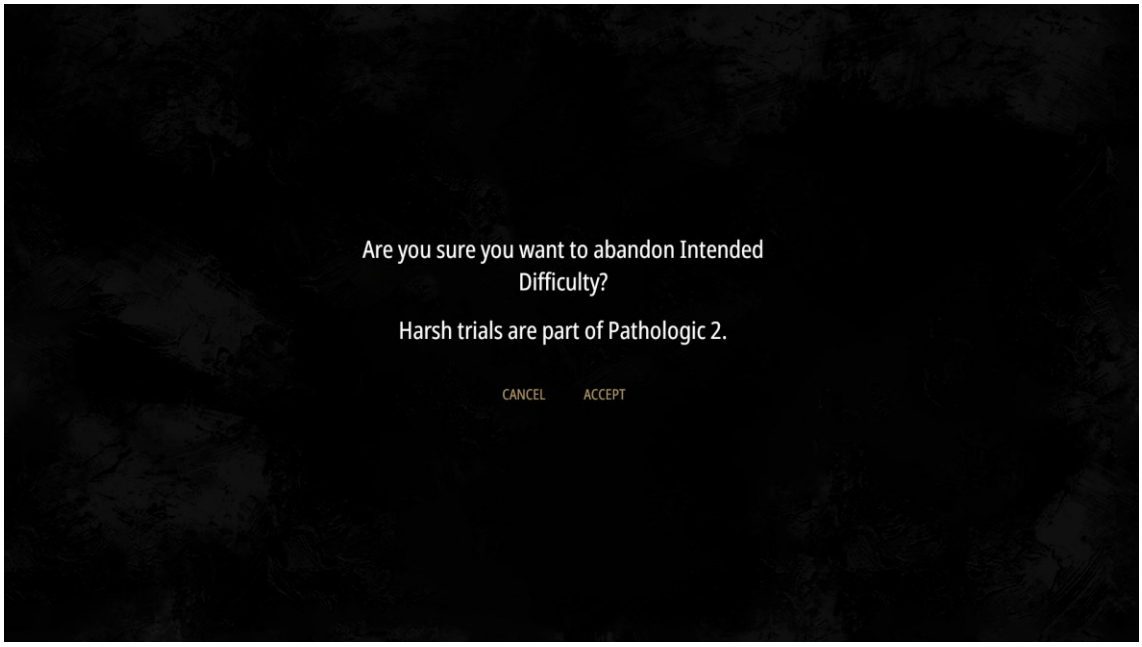

IMAGE 5. Pathologic 2 (2019) Ice-Pick Lodge.

Some games warn players that difficulty and struggle is part of their core experience as shown on the picture above.

From Software's *Souls*-series of videogames which includes *Dark Souls,* is known for being difficult. These games have difficulty as a core part of their experience. The games director Hidetaka Miyazaki has said though that the difficulty by itself has never been the intention of these games but more the feeling that comes from overcoming tremendous odds (Kamen 2016). The games have no manual or dynamic difficulty settings, but they do have options for the player to make them easier. For example, player can summon another player to help them with a boss battle if they are having difficult, or the player can spend time leveling up their character. There is the possibility that many players will abandon these games when they are first facing difficulties since many of these embedded difficulty options will only come clear after having some familiarity with the game's different systems.

5 Game Project

5.1 Game Introduction

The purpose of this study was to create a difficulty system that would meet the wanted results from player experience, which was a challenge that could be overcome by any player but would still provide enough of a challenge not to be boring. The game that this difficulty system is created for is a 2D action game called Another Roguelike (image 6). The game has “roguelike” elements such as randomly generated levels and the loss of all progress on character death. The levels are not completely random but generated from different room pre-sets. The rooms are premade with enemies placed on them and then they are pulled from a pool randomly when the level is created. Player’s health level persists between levels and the only way to get health back, is to use an upgrade opportunity between levels to heal.

The player character can fire arrows in all directions and has the ability to dash to avoid enemies. The dash mechanic is on a cooldown of a few seconds, to prevent the player from abusing the invulnerability it provides. Player can take five damage before they are defeated, and enemies deal one damage per hit, mostly from projectiles but some enemies deal damage with contact to player character. There are also spike traps that deal one damage if the player walks on them.

The game has four levels with sections between that allow the player to upgrade their character to make things easier. These upgrades include giving the player plus one health, improving the player’s arrows to deal more damage and upgrading the player’s movement speed. The final level consists of only two rooms with the latter of these containing a boss encounter. The boss is a significantly more challenging enemy with more complex attack patterns and more health.

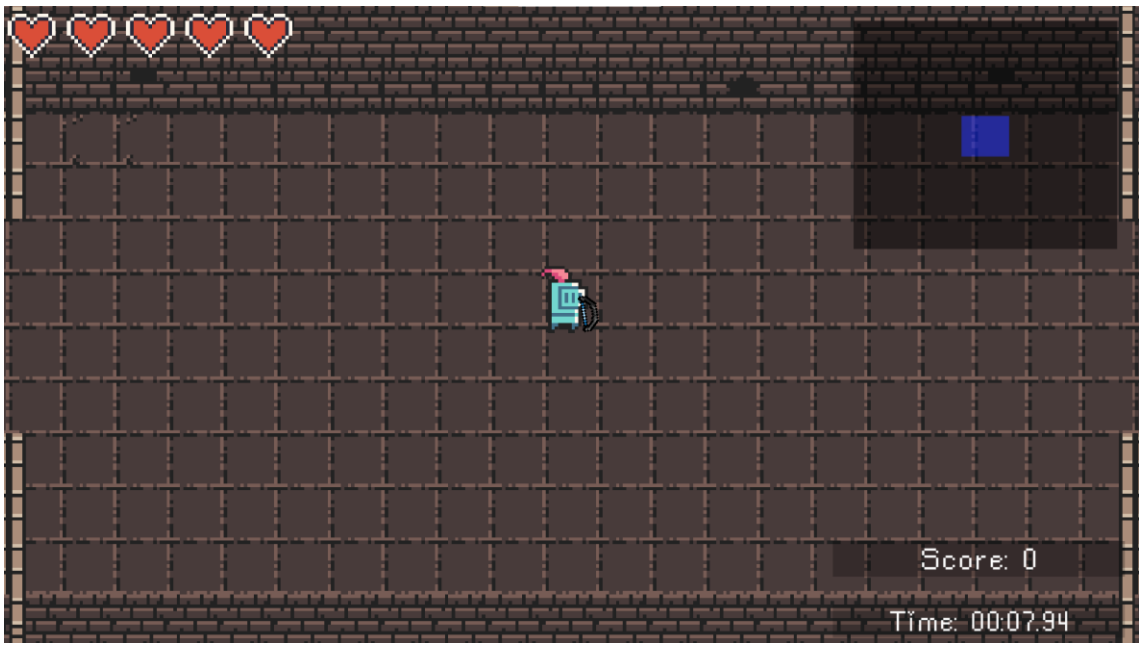

IMAGE 6. Another Roguelike (2021).

5.2 Creating the difficulty system

5.2.1 Difficulty premise

The plan was to create a difficulty system that takes player's skill level into consideration and provides a challenging experience for all players, while avoiding making the game too hard for some players. This possibly means that most players would not die but got through the game with only one or two health remaining.

The difficulty system was created as a combination of the different forms of difficulty design. *Manual, dynamic* and *embedded* designs were implemented to the game. The idea behind this design was to give the players the autonomy of choosing their own skill level with the manual difficulty system, but to also provide possible adjustments, if the difficulty proved to be too much or too easy. These adjustments would be done using automated or embedded design choices. Manual difficulty was added so that players could choose their starting skill levels and to increase player autonomy (Smeddinck, et al. 2016). The game would then follow a specific pattern designed for each difficulty. Enemy stats and the pools of possible rooms are different between difficulties.

5.2.2 Room pools

The room difficulties are based on the number of enemies, the number of different kinds of enemies, traps, and the different combinations of these elements. For example, difficult rooms might have traps aligned in such a manner that the player is forced to use the dash to get through the room without taking damage (image 7). These kinds of rooms would not appear on easy difficulty.

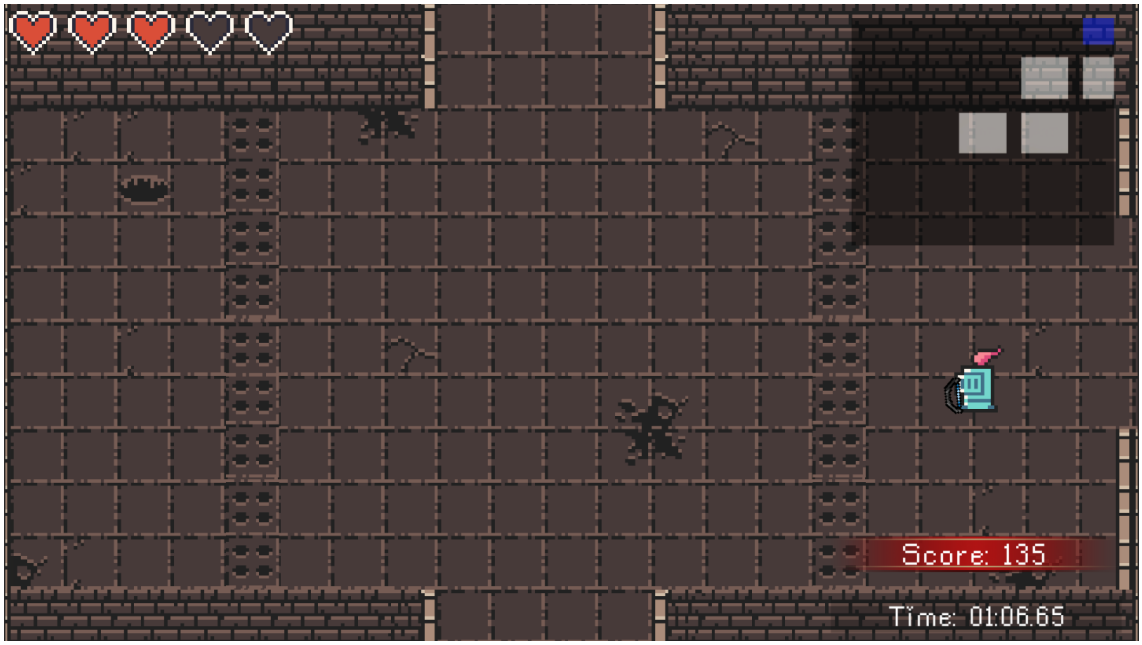

IMAGE 7. Another Roguelike (2021)

Some difficult rooms might contain traps in ways that will force the player to dash through them to avoid damage as seen in the image above.

Other ways that some rooms are considerate more difficult than others, are the way in which enemies are positioned in relation to the entrance. The player can only see the layout of the room when they enter it, so if an enemy is close to the entrance, the player would have to react very quickly to avoid damage. The level is generated completely at the start, so the pools of rooms that are used cannot be changed during the level. At the end of each level the game will check whether it would be best to keep the same pool of rooms or to increase to a more difficult pool. The game will avoid decreasing the difficulty of room pools since it would be a more noticeable and immersion breaking adjustment from the game and

DDA systems should try to be as discreate as possible (Zohaib 2018). *Normal* difficulty has the intended progression from *easy* –> *normal* -> *hard* room pools (image 8). If the player is struggling, the third level might be a normal pool of rooms. *Easy* difficulty has only rooms from the easy pool on default and *hard* uses the hard pool of rooms. Hard difficulty will not lower the room pool at any point since players choosing that difficulty will usually be more open to defeat and want to be tested (Boutros 2018).

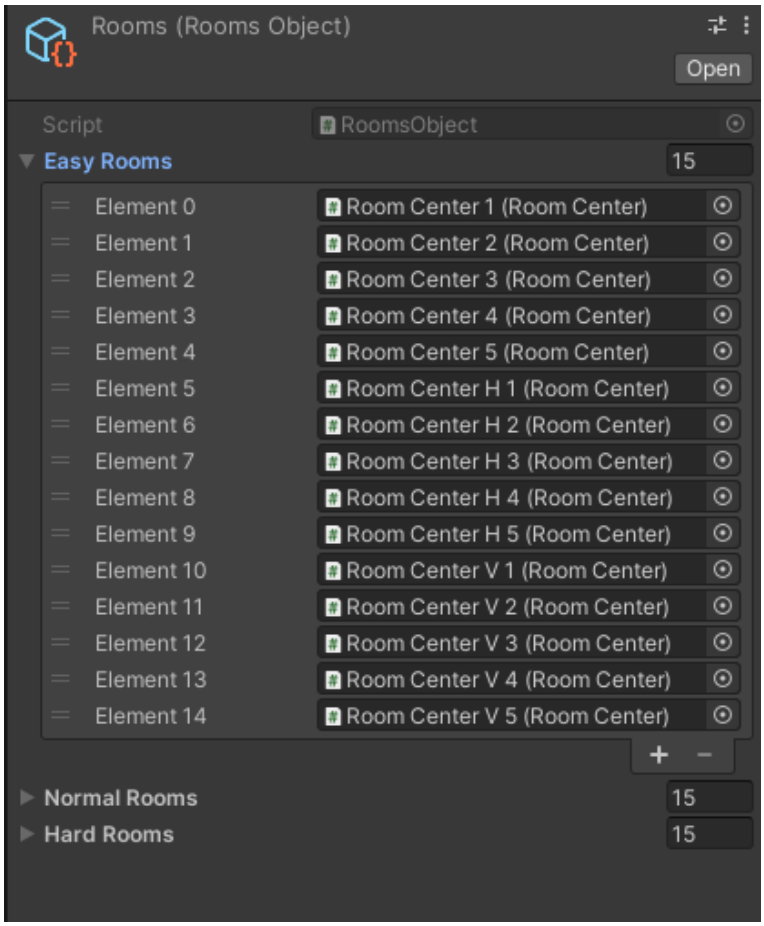

IMAGE 8. Another Roguelike unity-project (2021).

Each difficulty has their own pools of rooms. They are arrays that store every possible room for each difficulty as seen on the image above.

The boss fight level will have two variants. The easier version that will be used for the *easy* and *normal* difficulties, will have box objects placed on the level that the player can use to hide behind when avoiding the boss's attacks. On *hard* mode these boxes will be removed, and the player will most likely have to keep using the dash ability to avoid taking damage.

5.2.3 Enemy stats

Enemy stats are also used to control the difficulty (image 9). The stationary enemies that shoot fire or lasers at the player, have a fire rate stat that increases with difficulty. The bigger moving enemies that damage player through touch, have a speed stat that increases with difficulty. All the enemies have a health stat that also increases with difficulty. Stat increases can be used more discreetly so they are usually preferred to the more dramatic change of the room pools. They can also be adjusted during the level when the player changes the room.

These adjustments are made based on the player's performance. Player's current health and progress is considered, and what level they are on can affect the difficulty, for example if the player has chosen normal but is struggling on the first level already, the enemy difficulty can be changed to easy. Likewise, if the player has chosen easy difficulty but has had no problem getting to the last level with full health, then the enemy difficulty can be raised. Other adjustments include when the player has taken three or more damage within a single level, the enemy difficulty can drop for the duration of that level.

```
9 references
public void SetEnemyDifficulty()
{
    switch(enemyDifficulty)
    {
        case EnemyDifficulty.Easy:
            foreach(GameObject enemy in GameManager.instance.stageEnemies)
            {
                if (enemy != null)
                {
                    EnemyController currentEnemy = enemy.GetComponent<EnemyController>();
                    currentEnemy.fireRate = 2f;
                    currentEnemy.moveSpeed = 2f;
                    currentEnemy.laserRate = 5f;
                    currentEnemy.healthDropChance = healthDropRate;
                }
            }
            break;
        case EnemyDifficulty.Normal:
            foreach (GameObject enemy in GameManager.instance.stageEnemies)
            {
                if (enemy != null)
                {
                    EnemyController currentEnemy = enemy.GetComponent<EnemyController>();
                    currentEnemy.fireRate = 1f;
                    currentEnemy.moveSpeed = 3f;
                    currentEnemy.laserRate = 4f;
                    currentEnemy.healthDropChance = healthDropRate;
                }
            }
            break;
        case EnemyDifficulty.Hard:
            foreach (GameObject enemy in GameManager.instance.stageEnemies)
            {
                if(enemy != null)
                {
                    EnemyController currentEnemy = enemy.GetComponent<EnemyController>();
                    currentEnemy.fireRate = 0.5f;
                    currentEnemy.moveSpeed = 4f;
                    currentEnemy.laserRate = 2f;
                    currentEnemy.laserOnTime = 0.5f;
                }
    }
            break;
    }
}
```

IMAGE 9. Another Roguelike unity-project (2021).

Enemy stats have three different permutations that are based on difficulty. Some of the stats can be seen in the image above.

The Boss's health and attacks will be based on the current difficulty, which is decided by the game at the start of the boss level. The game will not adjust the boss's difficulty during the fight to avoid the possible immersion breaking and noticeable DDA adjustments (Zohaib 2018). The boss's attacks include firing damaging fireballs in random directions around him. The number of fireballs increases as the boss loses health. Boss's secondary attack is firing laser beams in different directions that start rotating around him. The number of fireballs and laser beams, as well as the rotation speed of the lasers are going to be dependent on the player's difficulty (image 10).

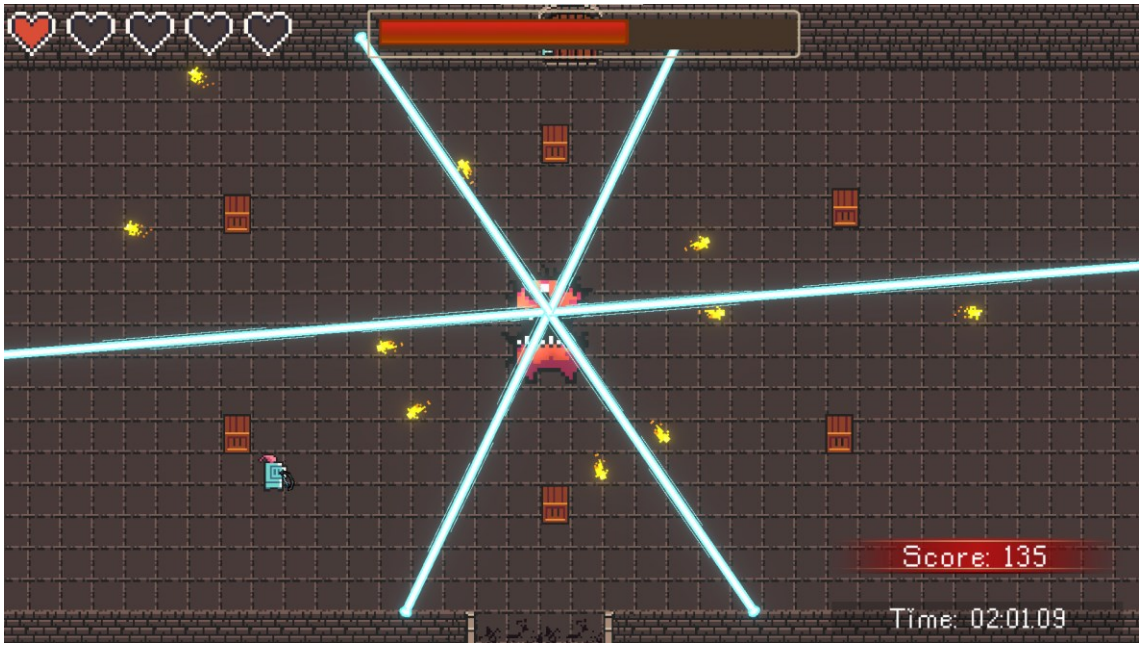

IMAGE 10. Another Roguelike (2021).

The Boss's attacks as well as the arena are different based on the players current difficulty level. Some of the boss's attack can be seen on the image above.

5.3 Testing

To test the difficulty settings and their effectiveness, a combination of unity analytics and google forms questionnaire was used (appendix 1). Unity analytics gathers data from the player, and you can make different custom events to keep track of what parameters you want to analyze (image 11). To test this game four custom events were created. The events were for when the player completes the game, takes damage, their character dies and an event for when the player finishes a level. Player damage event sends data about the room that the player was in when they took damage. This can help discover if any room is unnecessarily challenging for the selected difficulty. Level end event sends data that contains the player's current health, the difficulty they are on and the name of the level they just completed. Game completion event works similarly to the level end event but also sends data about the player's score and completion time. Player death event sends the name of the level.

```
public void SendDataGameComplete()
{
    AnalyticsResult result = AnalyticsEvent.Custom("GameComplete", new Dictionary<string, object> {
        { "Difficulty", DifficultyController.instance.selectedDifficulty.ToString()},
        {"Score", playerScore },
        {"Time", timerValue },
        {"Health", currentHealth }
    });
}
```

IMAGE 11. Another Roguelike unity-project (2021).

Example of unity analytics used in the games code above.

Google forms questionnaire asked the players about their playing habits and how they felt about the game's difficulty. The questions were: do they play often or rarely, did they complete the game or not, how they found the game's difficulty to be (from one to five with one being easy and five being hard), did the enemies or traps cause more difficulty or both equally. There was also a space for free comments.

5.3.1 First testing

First testing was done without the game having any DDA implementation. The game had three difficulty options to choose from and the difficulties decided what room pools are used and what the enemy stats are. The game would not change based on player performance.

Most of the testers had a lot of experience with games although a few people with less experience were also asked to test the game (image 12).

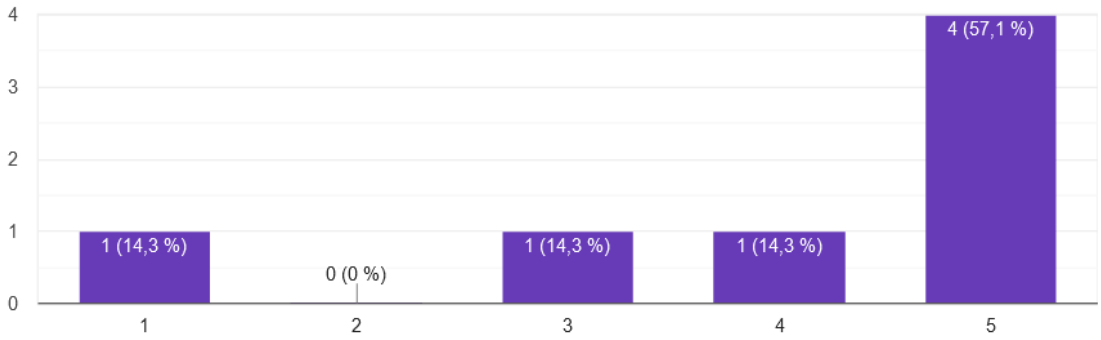

IMAGE 12. Questionnaire results.

Testers experience with games is pictured above, with five being “I play games often” and one being “I play games rarely”.

Most players with a lot of experience found the game to be too easy and the players with less experience found the game to be too hard. Only two testers could not complete the game: one experienced player who chose hard difficulty and the player with the least experience could not finish the game on easy. Average health players finished the game with was 3.14, which was a little more than was expected. There were two rooms that dealt more damage to players than average, these rooms had traps and/or enemies very close to the entrances so players would get damaged before they could react. There were no indications that players found traps to be more dangerous than the enemies nor the other way.

Some changes were made to the game based on the first testing. Health drops were added to enemies on easy and normal if the player was on low health. These

drops would heal the player with one hp, but they would still be random with 20% chance of dropping and only if the player had one or two hp left. The rooms that did the most damage to players were altered so that players would have more time to react. Traps now play a short animation when the player enter the room so that the traps are more clearly communicated to the player. Player now flashes red when hit with an enemy attack since before the player sprite would just have its alpha value lowered for a little bit when taking damage.

5.3.2 Second testing

Second testing was done after some DDA elements were added to the game. These elements include the health drops and adjustments to enemy stats during the game based on player performance. Also, room pools would change if the player was having trouble, or if they were performing better than expected. Testers were different from the first test, but the experience levels were very similar. This time the results for the difficulty experience was more balanced, but some people still found the game to be too easy as seen from the image below (image 13; image 14).

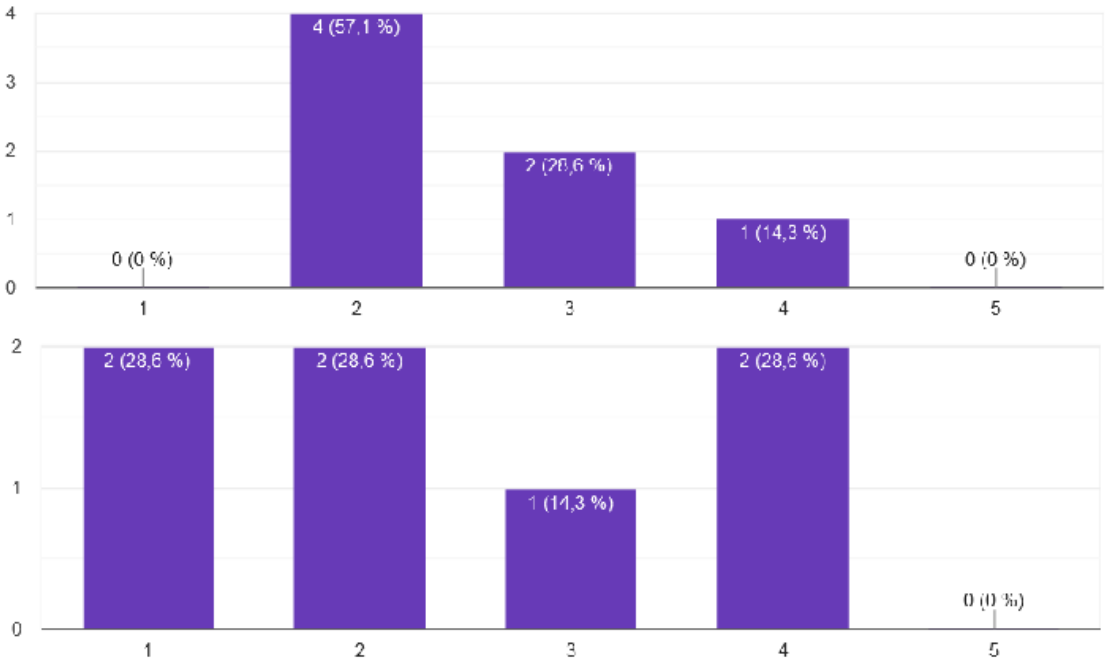

IMAGE 13. Questionnaire results.

In the above image first chart is the results from the second test and the second chart is the results from the first test, with one being too easy and five being too hard.

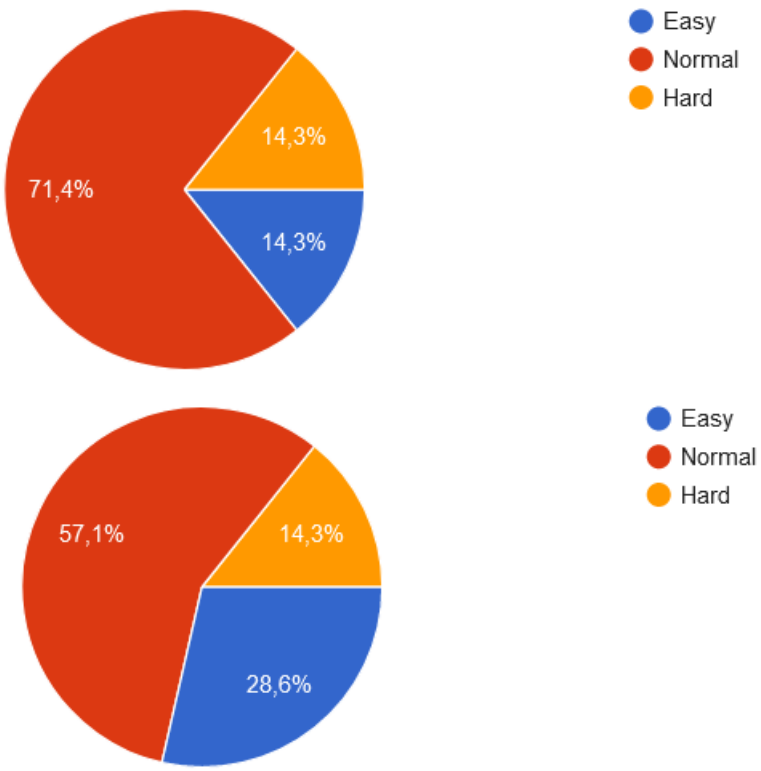

IMAGE 14. Questionnaire results.

Above pictured are the selected difficulties for the two tests. Top one is the second test and bottom one is the first test.

Less people were able to complete the game during the second test, however all the people who chose easy difficulty were able to complete it this time. There was no longer any specific room that had noticeably higher player damage rate. Average health for game completion was three this time. Many of the freeform comments were about getting damaged instantly when entering a room and not having time to react. Some rooms still had traps or enemies placed in positions that made them very hard to avoid. Also, some of the enemies or traps would get hidden behind the UI elements.

Changes were made to the game based on feedback. Enemy positions were changed, and enemies now waited until the camera had moved to reveal the

whole room before activating, also the camera movement speed was increased to avoid the player from running into traps while the camera was moving into place.

5.3.3 Third testing

Third testing was much smaller scale. Some of the earlier testers who had difficulty finishing the game were asked to replay the game with the changes made. Game completion rate was much higher this time around. All three people who were asked to replay the game now completed it. This could also be just because the testers now have more of a familiarity with the game, but the performances were a lot better than the first time they played. Easy difficulty testers did not find the game to be too easy this time.

Changes that were made after the third testing were mostly to make the game harder for people who found the normal difficulty too easy. The game would now start with *normal* room pool instead of *easy* on *normal* difficulty. The game would change enemy difficulty only if the player is really struggling. Health drops are still active when the player has low hp on *normal*. Questionnaires question on gaming habits was also changed to get a more accurate answers. The “I play games” one to five linear scale question was changed to similar question (“I play games on average?”) with more descriptive answers: *once a day*, *few times a week, once a week, once a month, less than once a month.*

5.3.4 Fourth testing

Fourth testing had similar player habits as the preceding tests. Although the question was more specific this time, the result was still very similar as seen from the image below (image 15). Some of the testers had played the game previously but not the second or the third version of it and some of the testers were playing the game for the first time.

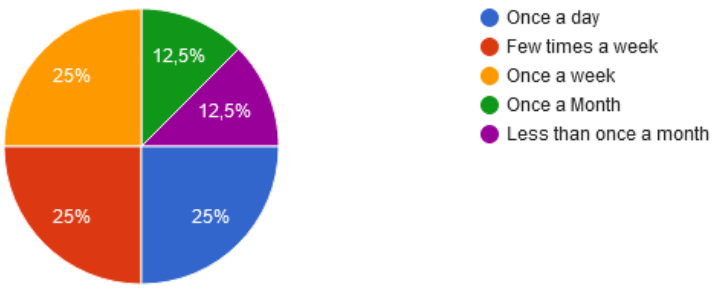

IMAGE 15. Questionnaire results.

Testers who said they played games once a week or less were more likely to choose *easy* difficulty when starting the game. Almost all of the players who chose *easy* found the game to be of average difficulty and most of the players who chose *normal* found the game to be a little bit below average on difficulty. The overall split on how players felt about the difficulty level was very similar to previous tests (image 16).

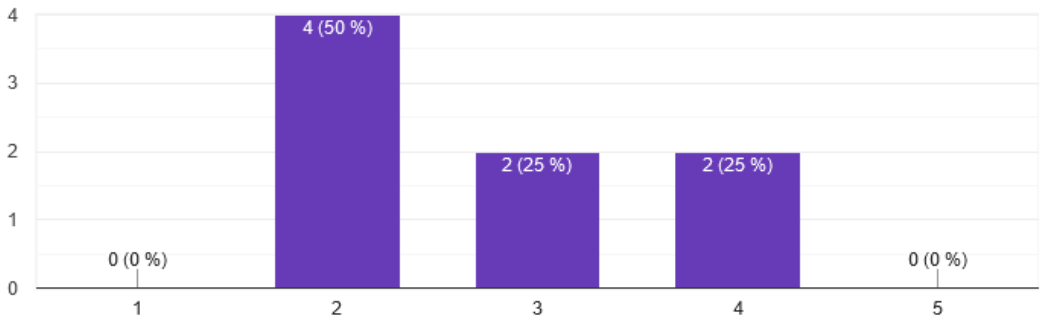

IMAGE 16. Questionnaire results.

Above pictured is how players felt about the game's difficulty. Very similar to previous test results.

Unity analytics showed however, that the performance of players was more consistent throughout the game, as opposed to previous versions where the players would have difficulty on later levels only. This was clearer on *normal* than other difficulties. After the testing enemy base stats on *normal* were raised 25% and the bar for dropping the enemy difficulty based on performance was increased. This might hopefully make the game more challenging on *normal* difficulty.

6 Conclusion

The game got more balanced little by little after each testing but was still too easy for many experienced players on normal difficulty. The games easy difficulty saw the best improvements with the testing and iterations done to it based on the feedback. Testing difficulty is hard and every bug and design flaw affects testing in a very damaging way. It is difficult to get a clear view of the difficulty when unintentional challenges hurt player performance. It would take many more test sessions and iterations to get the difficulty to a satisfying level, but these tests already had some indications that the game was changing to a more balanced experience, where players with different experiences could enjoy the same kind of challenge.

In the future more focus should be but on the normal difficulty and possible changes to the DDA systems, to provide necessary challenges when the player faces little challenge from the game. There should also be more systems in place to encourage players who do well on *normal* to play the game again on hard. The game already has a simple scoring system but adding a letter ranking and making the score be affected by the difficulty, could be helpful in encouraging the use of higher difficulties. Making the best clear time also be individual for each difficulty, could also be used to encourage players to try out each difficulty. To give more value to these systems some form of online leaderboard could be implemented.

One of the main reasons for this thesis was my feeling that many games are designed around a single difficulty and even if there are some options provided to the player, these are still regarded as secondary to the one main way to play the game. It feels like to me that even games with a bigger budget and long development time still treat their easier difficulties as secondary and often just take all the challenge away from the game. I think that providing people who lack experience with games, a similarly engaging experience as is provided to players with lots of experience, is a great goal to have for every game. There was an article written when *Final Fantasy 7 Remake* was released that called the game's easy mode too easy and found that there was no fitting difficulty for the article's writer (Fahey 2020). This was a game with a massive budget and development time, but the developers still seemed to feel that the solution to players who

lacked experience, was to just take any challenge away from the game, instead of making a balanced difficulty level.

Every game is made with some form of core experience as a guide for the design. This core experience should inform the design of every difficulty level and not just the medium and possible harder difficulties. Just putting in a god mode that makes the player invulnerable is a poor way to design for a diverse player pool in my opinion. It makes the game feel like two different experiences, when it should feel like a cohesive whole. I feel like this is helpful avenue for many indie developers to increase their possible user bases. It does not need to be a complex dynamic difficulty system with AI learning algorithms but putting some form of effort into making each difficulty engaging can do a lot for a smaller game. Every game is different though and that is why every designer must approach difficulty design a little differently, but I find that putting in the effort is really important.

BIBLIOGRAPHY

Boutros, D. (2008) Difficulty is Difficult: Designing for Hard Modes in Games. Game Developer. Referenced 2021. https://www.gamedeveloper.com/design/difficulty-is-difficult-designing-for-hard-modes-in-games

Brown, M (2016) Should Dark Souls Have an Easy Mode? Youtube. Watched 2021. https://www.youtube.com/watch?v=K5tPJDZv_VE

Chen, J (2006) Flow in games [Online] https://www.jenovachen.com/flowingames/Flow_in_games_final.pdf.

Csikszentmihalyi, M. (1990) Flow: The Psychology of Optimal Experience. Harper & Row, New York.

Fahey, M (2020) The Difference Between Final Fantasy VII's Easy And Normal Modes Is Too Drastic. Kotaku. Referenced 2021. https://kotaku.com/final-fantasy-vii-remakes-easy-mode-is-way-too-easy-1842791313

Gach, E. (2017) It Might Be Time To Rethink Difficulty Menus. Kotaku. Referenced 2021. https://kotaku.com/it-might-be-time-to-rethink-difficulty-menus-1820961183

Hopson, J (2001) Behavioral Game Design. Game Developer. Referenced 2021. https://www.gamedeveloper.com/design/behavioral-game-design

Ice-Pick Lodge (2019) On Difficulty. Steam. Referenced 2021. https://store.steampowered.com/news/app/505230/view/5156014950378311484

ISFE Europe's Video Games Industry. ISFE. Referenced: 2021 https://www.isfe.eu/data-key-facts/key-facts-about-europe-s-video-games-sector/

Jagoda, P (2018) On Difficulty in Video Games: Mechanics, Interpretation, Affect. Critical inquiry. [Online] 45 (1), 199–233.

Kamen, M (2016) Dark Souls 3 director: it's about 'accomplishment by overcoming tremendous odds'. Wired. Referenced 2021. https://www.wired.co.uk/article/dark-souls-3-hidetaka-miyazaki-interview

Melder, N (2013) A Rubber-Banding System for Gameplay and Race Management. Game AI Pro. [Online] 507-512.

Sashka (2017) Disable Timers. Nexus Mods. Referenced 2021. https://www.nexusmods.com/xcom2/mods/61

Saunder, T (2019) Resident Evil 2 Adaptive Difficulty | How does it work? Game Revolution. Referenced 2021. https://www.gamerevolution.com/guides/491409-resident-evil-2-adaptive-difficulty-zombies-ammo

Schmierbach, M. et al. (2014) No One Likes to Lose: The Effect of Game Difficulty on Competency, Flow, and Enjoyment. Journal of media psychology. [Online] 26 (3), 105–110.

Silverberg D (2020) Game Breaking: How Cheat Codes Changed Video Games Forever. Popular Mechanics. Referenced 2021. https://www.popularmechanics.com/culture/gaming/a33650224/cheat-code-history/

Smeddinck, J. D. et al. (2016) 'How to present game difficulty choices? Exploring the impact on player experience', in Conference on Human Factors in Computing Systems - Proceedings. [Online]. 2016 pp. 5595–5607.

Stout, M (2015) Influencing Player Behavior. Game Developer. Referenced 2021. https://www.gamedeveloper.com/design/influencing-player-behavior

Their, D (2019) 'Sekiro: Shadows Dies Twice' Needs To Respect Its Players And Add An Easy Mode. Forbes. Referenced 2021. https://www.forbes.com/sites/davidthier/2019/03/28/sekiro-shadows-dies-twice-needs-to-respect-its-players-and-add-an-easy-mode/?sh=2e2d0df11639

Zohaib, M. (2018) Dynamic Difficulty Adjustment (DDA) in Computer Games: A Review. Advances in human-computer interaction. [Online] 20181–12.

APPENDIX

APPENDIX 1. Difficulty Questionnaire

1. Selected Difficulty *
   - ◯ Easy
   - ◯ Normal
   - ◯ Hard
2. I completed the game *
   - ◯ Yes
   - ◯ No
3. I play games *

	1	2	3	4	5	
Rarely	◯	◯	◯	◯	◯	Often

4. I found the game to be *

	1	2	3	4	5	
Easy	◯	◯	◯	◯	◯	Hard

5. Most of the challenge came from *
   - ◯ Enemies
   - ◯ Traps
   - ◯ Both equally
6. Free comment