

Project

Rifters

Joan Dalmou Alacreu - al361638

Bachelor's Degree in

Video Game Design And Development

Universitat Jaume I, Castellón, Spain

Hanzehogeschool University of Applied Science, Groningen,

Netherlands

July 24, 2020

Advisors:

Antonio Morales

Abstract

This final degree project consists of the development of a competitive multiplayer game based on magic. This work has been done in the Game Lab program of Hanzehogeschool, being part of the Erasmus+ program 2019/2020. The project consists of the development of a product that could work at a commercial and innovative level.

Table of contents

Introduction	6
1.1. Work Motivation	6
1.2. Team	7
1.3. Objectives	7
Planning and resources evaluation	8
2.1. Planning	8
2.2. Resource Evaluation	11
2.2.1. Tools	11
System Analysis and Design	12
3.1. Game Design	12
3.1.1. Game Loop	12
3.1.2. Mechanics	12
3.1.2. Multiplayer System	14
3.2. Monetization Model	16
3.3. System Architecture	17
3.4. Interface Design	17
3.4.1. Main menu	18
3.4.2. Play menu	19
3.4.3. Join menu	20
3.4.4. Lobby menu	21
3.4.5. Character selection screen	22
3.4.6. In-Game UI	23
3.4.7. Pause menu	24
Work Development and Results	25
4.1. Work Development	25
4.1.1. Minimum Viable Product	25
4.1.1.1. MVP planning	25
4.1.1.2. MVP development	25
4.1.1.3. MVP testing	28
4.1.1.4. MVP results	30
4.1.2. Prototype	30
4.1.2.1. Prototype planning	31
4.1.2.2. Prototype development	31
4.1.2.3. Prototype results	34
4.1.3. Final product	35
4.1.3.1. Final product planning	35

4.1.3.2. Final product development	35
4.1.3.3. Final product results	36
Conclusions and Future Work	37
5.1. Conclusions	37
5.2. Future work	38
Bibliography	39
6.1. Bibliography	39
6.2. Source Code	39
6.3. Appendices	40
6.3.1 Test document	40
6.3.1.1. Introduction	41
6.3.1.2. Project Framework	42
6.3.1.3. Objectives & Research Questions	43
6.3.1.3.1. Research Question	43
6.3.1.3.2. Keywords	43
6.3.1.3.3. Sub questions	43
6.3.1.4. Possible Research Methods	44
6.3.1.4.1. Interview	44
6.3.1.4.2. Usability Testing	45
6.3.1.4.3. Critical Facet Playtest	45
6.3.1.5. Research Design	46
6.3.1.5.1. Operationalization	46
6.3.1.5.2. Evaluation Methods	47
6.3.1.5.3. Target group	48
6.3.1.5.4. Evaluation execution	48
6.3.1.6. Preparation and execution	50
6.3.1.6.1. Pre-Test	50
6.3.1.6.2. Participants	50
6.3.1.6.3. Execution	50
6.3.1.7. Results and Analysis	51
6.3.1.8. Conclusions & recommendations	53
6.3.1.8.1. How comfortable is the aiming system?	53
6.3.1.8.2. How precise is the aiming system?	53
6.3.1.8.3. How accurate is the input response of the aiming system?	53
6.3.1.8.4. Recommendations	54
6.3.1.9. Appendix	55
6.3.1.9.1. Design research canvas (first version)	55
6.3.1.9.2. Design research canvas (last version)	56
6.3.1.9.3. Test questions and instructions used by the researcher	57
6.3.1.9.4. Agreement survey	59
6.3.1.9.5. Critical facet playtest Survey	61

6.3.1.9.6. Critical facet playtest Results	62
6.3.1.9.7. Usability testing notes	64
6.3.1.9.7.1. Participant 1	64
6.3.1.9.7.2. Participant 2	64
6.3.1.9.7.3. Participant 3	64
6.3.1.9.8. Usability testing transcripts	65
6.3.1.9.8.1. Participant 1	65
6.3.1.9.8.2. Participant 2	67
6.3.1.9.8.3. Participant 3	70
6.3.2 First GDD	73
6.3.2.1. Project setup	74
6.3.2.1.1. Team name	74
6.3.2.1.2. Team composition	74
6.3.2.2. Game overview	75
6.3.2.2.1. Working name	75
6.3.2.2.2. Description	75
6.3.2.2.3. Target audience	75
6.3.2.2.4. Rating	75
6.3.2.2.5. Competitive products	75
6.3.2.3. The game world	76
6.3.2.3.1. Description	76
6.3.2.3.2. Elements	76
6.3.2.4. Game Rules	77
6.3.2.4.1. Objective	77
6.3.2.4.2. Scoring	77
6.3.2.4.3. Players	77
6.3.2.4.4. Equipment	77
6.3.2.4.5. The ball	77
6.3.2.4.6. Match	77
6.3.2.4.7. Field	77
6.3.2.5. Game mechanics	78
6.3.2.5.1. Player's abilities	78
6.3.2.5.2. Areal interactions	78
6.3.2.6. Characters	80
6.3.2.7. User Interface	80
6.3.2.8. Sound	80
6.3.2.9. Technical	80
6.3.2.9.1. Platform	80
6.3.2.9.2. Engine	80

1. Introduction

1.1. Work Motivation

This game originated in the first weeks of the Hanzehogeschool Game Lab program. In this program, students are required to develop a serious video game in groups of 5 to 6 people. Students are provided with a number of different options, being able to develop a video game for a specific client out of all the possible ones or to develop an independent video game (called Project X). This last option is considered to be the most difficult one since it does not follow rules determined by a client, but a video game has to be created from the very idea, having to meet the evaluators' expectations.

The team I joined had decided that they were going to carry out Project X and I had no problem with that. The initial plan was to make a multiplayer combat video game where two players would fight to knock down the opponent using magic to manipulate the terrain and attack. But this video game had a rather generic idea and since only a couple of Project X's pass the first cut a year we had to think of a more innovative idea with more market potential.

So after several modifications to the idea, we came up with the final idea that we worked on throughout the development of the game. A competitive multiplayer sports game where two players (with the possibility of creating new 2vs2 or 3vs3 modes) compete against each other to score more goals than the opponent before time runs out by using different spells to manipulate the field and control the ball. This project had more potential since there are no games in the market with this same formula (although there are similar products in which the project is inspired like Rocket League or the possibly already canceled Broomstick League, which can be seen on the Figure 1) and could be economically viable.



Figure 1: *References for the project Broomstick League (left) and Rocket League (right).*

1.2. Team

In the Game Lab program, a video game is developed during a semester in a team. The intention is that the students try to form the teams in a balanced way in the different fields of which the development is composed in order to bring out the best possible product.

In my case, the team was composed of 6 people. A designer, a researcher, an artist, a musician, and two programmers. My role in the team was that of programmer and artist, but since the students I developed the project with are sophomores their programming skills were not very advanced so most of the development I have worked only as a programmer, making only preliminary models and effects as a prototype so that the artist could make more detailed models that would fulfill the function that we programmers required.

1.3. Objectives

- To create a functional multiplayer system.
- To create a market plan that can convince investors.
- To create a set of spells that interact with the environment

2. Planning and resources evaluation

2.1. Planning

Because of my role in the team being mainly a programmer, that's where most of the hours were spent, but still, some time was dedicated to art, so the tasks have been divided into 5 blocks: *Research, Programming, Art, Documentation, Ideas / Group meetings*.

In Research (Table 1) the whole process of creating the first idea has been included. All the research on how the first idea of the game could be adapted to something functional on a commercial level and that could create its own identity. Also included are the hours spent following Unity tutorials in order to implement certain mechanics and the playtesting carried out during the game's creation process.

Programming (Table 2) includes everything related to Unity that is not art or VFX. All hours of programming, scripting, and prefab creation are included here. As the Game Lab program is divided into 2 phases (MVP and prototype) this section has been divided into three parts since the final product is also included, which is the one that results from the completion of the Game Lab to the delivery at the Universitat Jaume I.

Art (Table 3) includes hours spent making 3D models, VFX, and interfaces. Although many of the VFX have not been included in the prototype, they will be included in the final product to be viewed.

Documentation (Table 4) includes everything related to the documents that need to be created during the development of the project. This includes the memory and other documents that have been needed in Hanze.

Ideas / Group meetings (Table 5) will include everything related to the team, meetings, presentations, and feedback with the teachers.

Finally, there is a table (Table 6) showing the total number of hours planned for the development of the project.

Due to the deviations suffered during the project creation, the total number of hours spent in the project are 375 even when only 300 had been planned.

Research	
Task	Expected hours
Unity tutorials	10
Playtesting	10
Research about alternative sports games	5
Total	25

Table 1. *Research hours.*

Programming	
Task	Expected hours
Programming for the MVP	50
Programming for the prototype	100
Programming for the final product	30
Total	180

Table 2. *Programming hours.*

Art	
Task	Expected hours
VFX for spells	10
Base model for the arena	4
Total	14

Table 3. *Art hours.*

Documentation	
Task	Expected hours
Technical proposal	5
Memory	30
Final presentation + preparation	5
Total	40

Table 4. *Documentation hours.*

Ideas / Group meetings	
Task	Expected hours
First concept	8
Weekly group meetings	14
Meetings with coaches	5
Pitching the project to the coaches	4
Final Concept	10
Total	41

Table 5. *Ideas and Group meetings hours.*

General Planning	
Task	Expected hours
Research	25
Programming	180
Art	14
Documentation	40
Ideas/Group meetings	41
Total	300

Table 6. *General Planning.*

2.2. Resource Evaluation

2.2.1. Tools

- Unity 3D. The game engine we used to develop the game.
- Visual Studio. Code editor used along with the game engine.
- Github. Software used to share the project with the other developers and work simultaneously.
- Adobe Photoshop and Adobe Illustrator. Software used to create graphics for the UI and marketing of the game.
- Blender. Software used to create 3D models for the assets of the game.
- Substance Painter. Software used to create the textures of the 3D models for the game.
- Google Docs. Software used to create documents.
- Google Spreadsheets. Software used to create the breakdown of hours spent in the project and the business tables
- Trello. The web used to create the Scrum planning.
- Discord. Software used to communicate with teammates and do weekly meetings.

3. System Analysis and Design

3.1. Game Design

In this section, the game's functioning will be developed. The game has undergone many changes during development, as it was originally intended to be a multiplayer combat game using level manipulation and the final idea is a sports-based multiplayer game where different spells are used to score the ball in the opponent's goal.

3.1.1. Game Loop

The game loop is very simple. Players will access the main menu at the start of the game. From this screen, they will have access to different options, including starting a game. At this point, the players will choose whether they want to be the host of the game (a device that will play the role of the server as well as the client) or join another host. Once both players have entered the room (the player other than the host must have submitted the host's IP to join the game) the host can start the game.

Once a match has started, both players will have to dispute control of the ball to score in the opponent's goal, using the different spells in their favor. Every time a player scores the players will return to their starting position and the ball will appear in the center of the stadium. This process will be repeated until the time reaches 0, where the player with the most goals will win the match. In case there's a tie, the game will end and both players will finish with a tie result.

3.1.2. Mechanics

One of the first tips received after the first testing sessions was the reduction of mechanics that the game contained. During the first part of the development, the imagination was given free rein and eight spell mechanics were created. This

saturated the game so we proceeded to reduce the number of these trying to concentrate on perfecting a few mechanics.

So, in the game, you can find the following mechanics:

- Movement

Using the WASD keys the player will be able to move the character around the environment. It will collide with walls and other players but not with the ball, as it can only be manipulated with magic.

- Jumping

Using the spacebar players can jump over players. This move can be combined with the *blink* spell to advance great distances in a short time.

- Aiming

Using the movement of the mouse the player will move the camera to look in different directions. Just as looking up or down will not affect the character and will only move the camera, in sideways rotation the character will be affected by rotating next to the camera.

- Fireball shooting

By pressing the left-click the character will throw a fireball that will explode on impact with any surface. This impact will create an explosion that will add force to the ball if it is close. This way the players can control the ball and add speed to it.

- Ice spell casting

By right-clicking on the mouse the character will perform a spell that will freeze the ball by removing all movement. This mechanic is the counterpart of the previous one because as well as adding speed to the gameplay it serves to reduce the speed and keep the ball manageable.

- Blinking

If the player presses shift he will perform a scrolling spell. This mechanism has been called *blink* but is also well known as *dash*. It doesn't matter if the character is on the ground or in the air, he will advance very quickly.

All spell related mechanics have a cooldown that doesn't allow spamming of different skills without any limits.

3.1.2. Multiplayer System

The multiplayer system is one of the key pieces that make up this product. The entire game design is built around the multiplayer game, so without a multiplayer system, the project makes no sense.

The multiplayer system has been designed to fit the resources we as students had for this project. At first, we used Photon, a Unity tool that offers servers of up to 20 people in its free version, but even the size of the server was not convincing once the development was more advanced and also there were problems creating the system, so after analyzing it with the other programmer of the team we decided to change to Mirror, an asset of Unity based on the old multiplayer system of the engine with a great documentation that in my point of view was much more comfortable to use, and since the development of this system was in my charge I had the freedom of decision. Ultimately we chose to make a host-client system because we eliminated the need to create the whole server and we could get players to connect to each other.

The system had to meet the following requirements:

- Allow players to create lobbies
- Allow players to join lobbies
- The host must be able to start a match
- Players in the same lobby should join the same match
- Players must be able to interact with the other players during the match.

For the first and second objectives, the lobbying system currently used in the game has been created. From the moment the game starts there is a Network Manager (the entity within the game that manages all network-related events) that will be activated when the player enters the Play Menu (see Figure 3). Once in this menu, the player must decide if he wants to be the host of a game or join another host. In case of choosing the first option, the Network Manager will execute the order to create a room and be the host of this one (i. e. to exercise as server and client at the same time) and a new GameObject will be created in each client called RoomPlayer which

will be the representation of the player in the server. If instead of wanting to be the host, the player decides to join another player's room, he will be taken to the Join Menu (see Figure 4) where he will have to submit the host's IP and press "Join". Once you join, a RoomPlayer will be created in each client for the new player. There will be a RoomPlayer for each player that joins the lobby, and this will be in charge of showing the interface that is observed in the Lobby Menu (see Figure 5) and the Network Manager will update the RoomPlayer of each player every time someone makes a change (the only changes that can be made right now in the lobby is to change the status from Ready to Not Ready and vice versa). Finally, when both players are in Ready status, the host can press the Start button (a button that only appears to the host, the other players will not see this option to start the game).

When the host decides to start the game, the RoomPlayers will be replaced by other GameObjects called GamePlayers. They have a very similar function to the RoomPlayers, but instead of managing the lobby interface, they manage the game interface and the player data in it (score, etc). This change of GameObject is possible because every element of the game that intervenes with the multiplayer system has a NetworkID associated with a ConnectionID, so when the game starts the Network Manager replaces the RoomPlayers by GamePlayers assigning them the same ConnectionID to their NetworkID.

Once the game has started, it will be the Mirror himself who will synchronize the players and the ball through the components he has already created in his library. In order for the players and the ball to move in all instances of the game, these GameObjects have a component called Network Transform that synchronizes the movement of the elements in the instances. This also applies to spells cast by players.

And to make the animations created by the artist synchronize Mirror provides a component called Network Animator that allows the animations made by a character to synchronize in all instances of the game.

This is the basic design and operation of the multiplayer system, there are more layers of programming that make possible the operation of this system but serve to improve its performance and do not contribute anything to know what is its operation, more information can be found on at the end of this document where the Source Code is available.

3.2. Monetization Model

When carrying out this project at Hanzehogeschool, one of the requirements to move forward with the project was to complement it with other subjects related to marketing and building a business. For this purpose, during the development of the product, in addition to the feedback from the teachers, different market and audience studies have been carried out. In the appendices, you can find a link to a Google Drive folder where you can find all the documents related to this subject.

In the beginning, the business model proposed by the team was the traditional system of paying for the license. But both the discontent of a few of the team members and the feedback from the teachers led to the conclusion that the way the game was going to be sold needed to change. The different studies carried out can be found in the link to the appendices, but the conclusions we draw from them are that the game should be oriented to a free-to-play model with cosmetic micropayments.

The free-to-play model has been chosen because the trend shown by the video game market in recent years indicates that free-to-play games such as *Fortnite* or *Call of Duty: Warzone* are having much more impact. This is because not having an economic barrier that prevents the player from trying the product reaches a much wider range of players.

In addition, this free-to-play model has been shown to make players more willing to invest money in different parts of the game. It was decided that the micro-payments should only be cosmetic because none of the components were in agreement with

the current functioning of the loot boxes and it is also a current issue in the Netherlands, as the loot boxes have been studied lately to determine whether they were considered gambling or not. Following a process carried out by the authorities, it was determined that certain types of loot boxes can be considered gambling and therefore products containing these loot boxes cannot be marketed in the Netherlands. For all this, the general opinion of the team was to stay away from them.

3.3. System Architecture

In order to play the game, you do not need any special component except an internet connection to be able to connect to your opponents. In addition, a screen with a 16:9 aspect ratio is recommended as this is the resolution at which the entire design has been devised, and if a different ratio is used, the game will be scaled so that it can give way to loss of quality.

3.4. Interface Design

All the interfaces in the game have been designed at the organizational level by me. Both programmer were in charge of the art of the interfaces and some problems with the artist already arose that led to this situation. In the following section, we will explain the different interfaces and menus created for the game and what function they have, since it is something necessary to understand the operation of the game both at a multiplayer level and at monetization level.

3.4.1. Main menu



Figure 2: Main menu description image

The main menu of the game (*Figure 2 that can be seen above*). This is what the player will see when opening the game.

- Settings Button. This button leads the player to the settings menu.
- Character Selection Button. Through this button, the player will go to the character selection screen.
- Play Button. When pressing this button, the player will go to the *play menu*.

3.4.2. Play menu

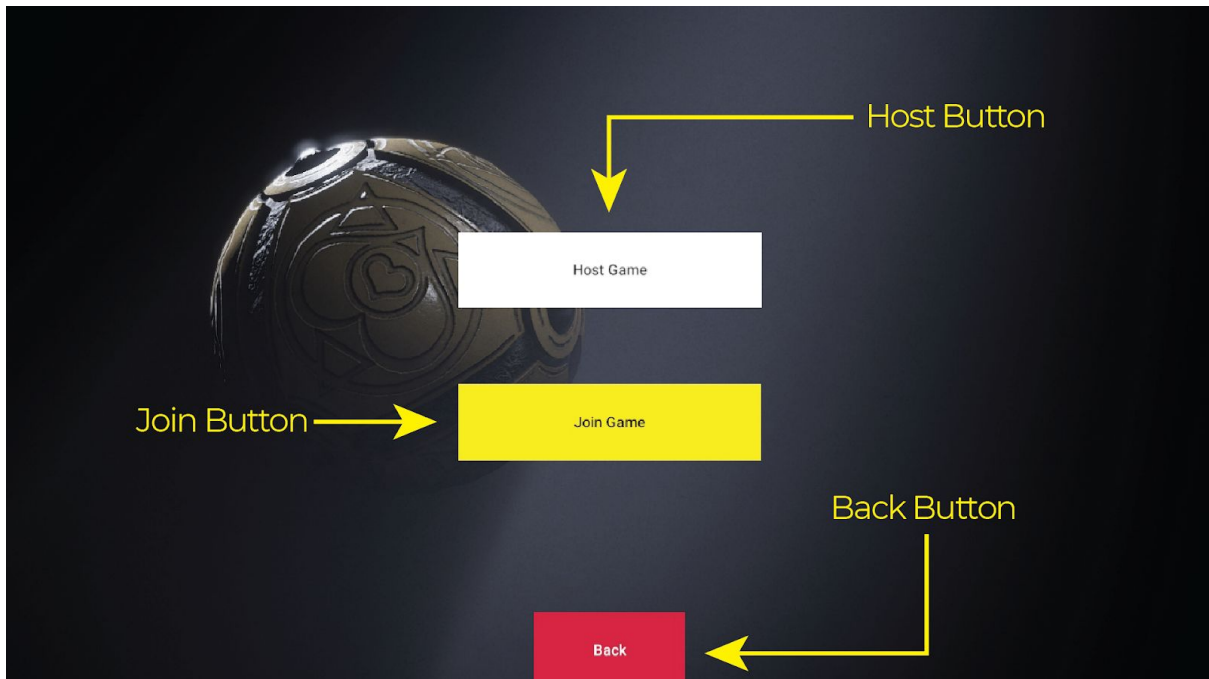


Figure 3: Play menu description image

When players arrive at this menu (*Figure 3*) they will choose between hosting a game and joining a game.

- Host Button. If the players press this button, they will join the lobby of a game and will wait for more players (it depends on the mode they are playing, in this moment there is only one mode that is 1v1).
- Join Button. By pressing this button, players will access the *Join menu* where they will join another player's lobby.

3.4.3. Join menu

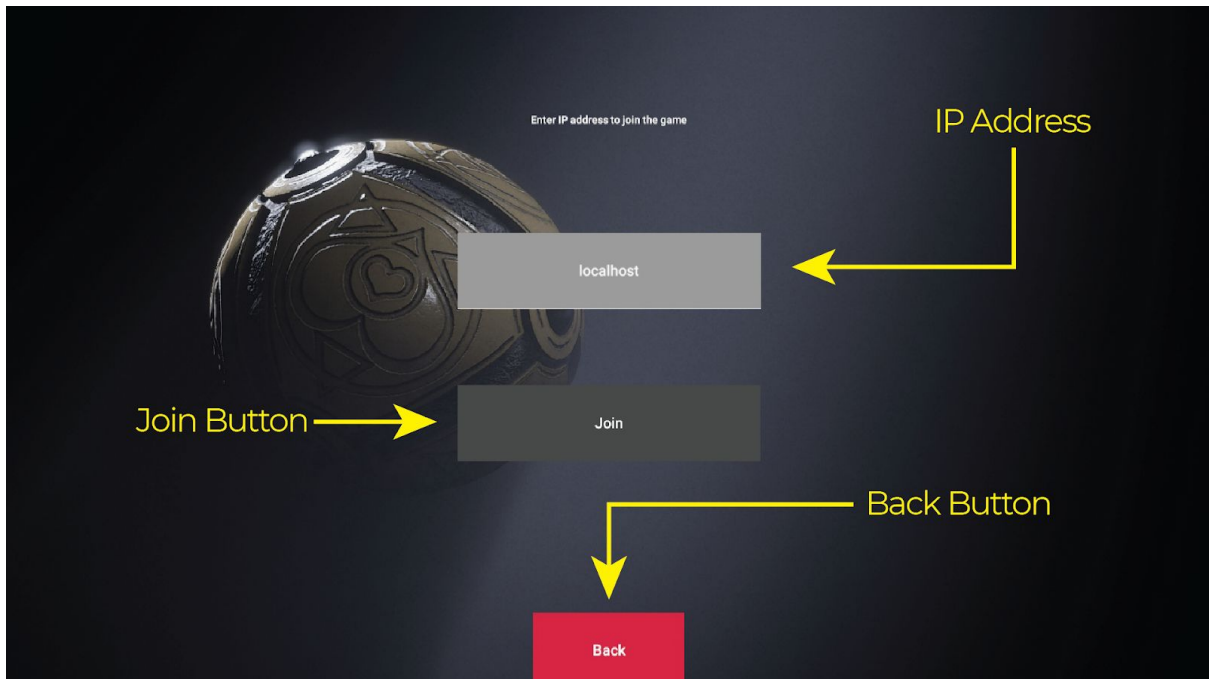


Figure 4: Join menu description image

In this menu (*Figure 4*), players will be able to join a session where another player will be waiting. To do this they will use the IP address of the host.

- IP address. Input field where the players will write the IP address of the host.
- Join Button. Button to join the lobby.
- Back Button. By pressing this button, players will return to the play menu.

3.4.4. Lobby menu

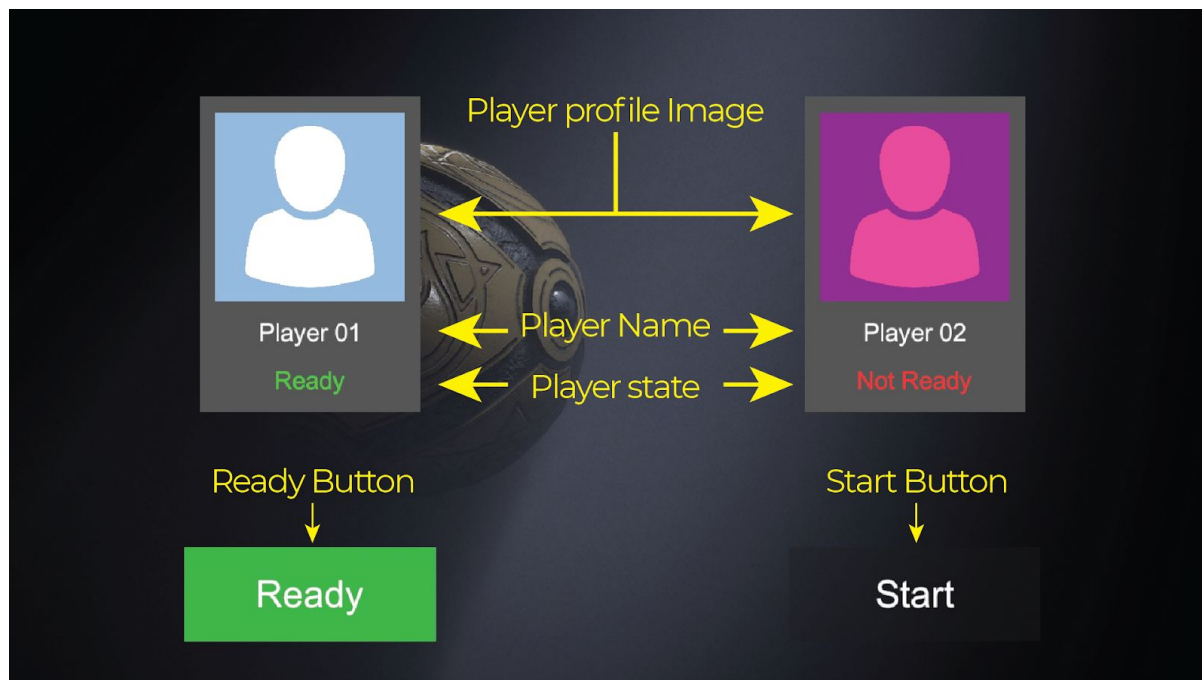


Figure 5: Lobby description image

In the lobby (Figure 5), players can change their status to Ready by using the Ready button. Here players can see each other's avatars and usernames, which are customizable aspects of the game. Once both players are ready, the host can press start and the game can begin.

- Player profile image. Player's avatar image. Players can choose between different avatars.
- Player name. Player's username.
- Player state. This label indicates if the player is ready to play or not.
- Ready Button. Both players have this button available and can press it to change their ready state.
- Start Button. This button is only available for the host. It will be interactable when both players are ready to start.

3.4.5. Character selection screen

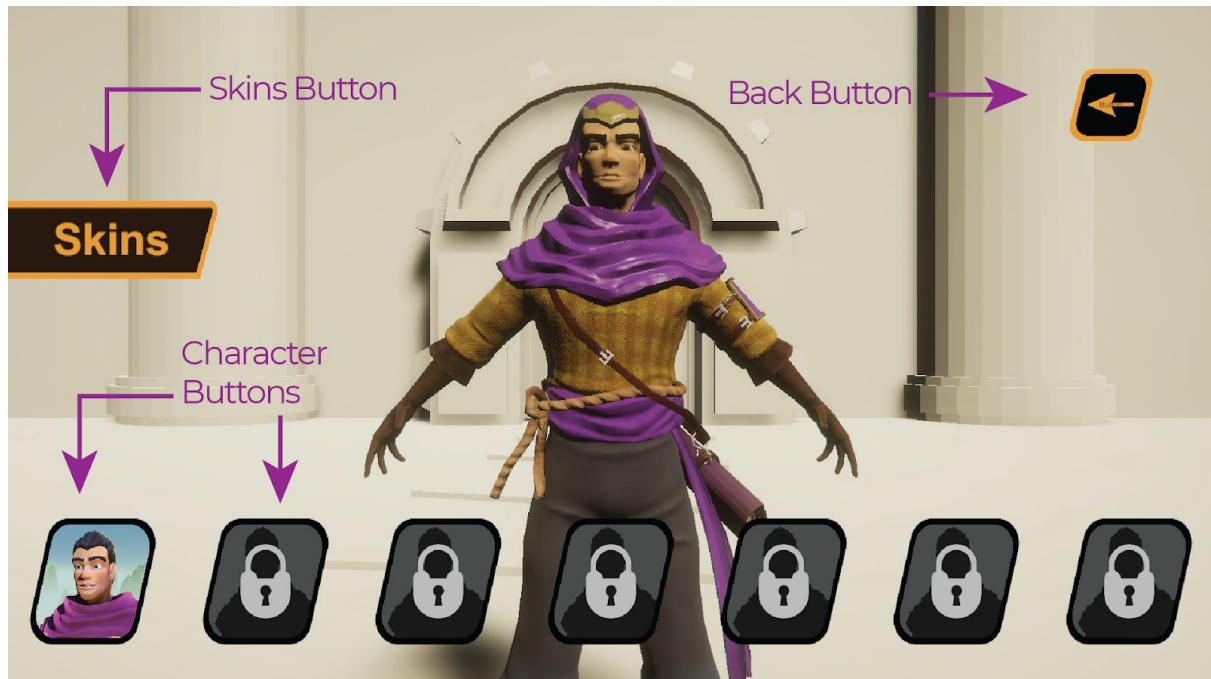


Figure 6: Character selection menu description image

This menu (*Figure 6*) is where the players will choose between the different characters available and their available skins and look at how they look on the in-game model.

- **Skins Button.** By pressing this button, players can choose the skin they want to use from the character they have selected at that time. This screen is where players can buy new skins and where they will find the new releases.
- **Character Buttons.** Each character will have its own button on the scene to select it. By pressing it the players will select this character to play.
- **Back Button.** If the players press this button, they will return to the main menu.

3.4.6. In-Game UI

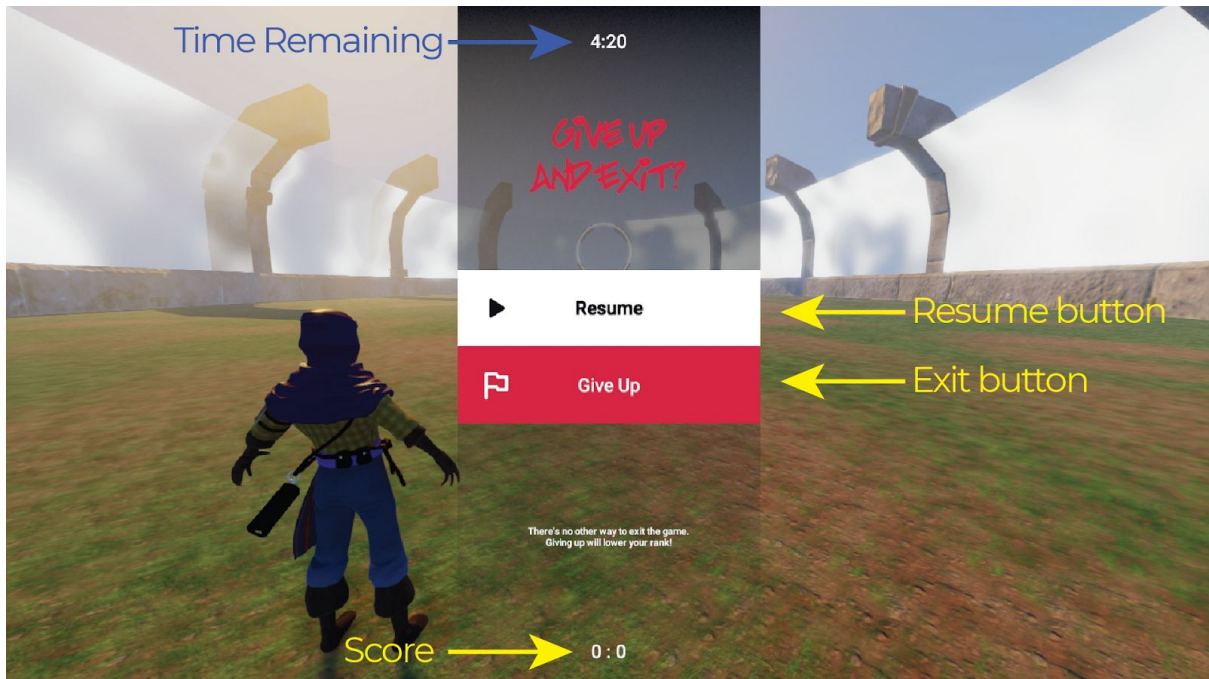


Figure 7: In-game UI description image

In this image (*Figure 7*), you can see the main interface of the game where the player can get all the information needed for the development of the game.

- Timer. Remaining time of the match. When it reaches 0 the game will be over.
- Score. The score indicates how the game is going. Your score will always be on the left and your opponent's score will always be on the right.
- Fire Spell. Each time the fire spell is used, the icon will become transparent and will become more opaque as the cooldown time passes.
- Ice Spell. It has the same function as a fire spell but when you use ice spell.
- Blink Spell. Same as the previous two with the *blink*.

3.4.7. Pause menu

Figure 8: *Pause menu description image*

Although if the player opens the pause menu (*Figure 8*) during a game, the game will not stop.

- Time remaining. Remaining time of the match.
- Resume button. This button will close the pause menu.
- Exit Button. By pressing this button, the player will leave the game and the opponent will be declared the winner.
- Score. The score of the match.

4. Work Development and Results

4.1. Work Development

Given the way the work is programmed at Hanze, the development of the product is divided into two parts, the creation of the MVP (Minimum Viable Product) and the creation of the Prototype. There is also a third phase, which is the work done by me individually from the delivery of the MVP in Hanze to the final delivery in Spain. The following sections describe the planning, development, results, and testing of the different parts.

4.1.1. Minimum Viable Product

The Minimum Viable Product is the minimum expression of video games that can express the mechanics and possibilities of the product. For this part of the project, it was necessary to create what were the core elements of the game, so the first spells that interact with the environment were devised, the character's movement was determined, and the first version of the level was designed.

4.1.1.1. MVP planning

- Create four spells that represent how the player can interact with the level
- Create a player's movement that's comfortable and precise
- Create the first version of the character of the game
- Study how the target audience feels about the game idea
- Create an arena that can fit the spells created and allow fast movement

4.1.1.2. MVP development

The most important thing during this phase was to get the character's movement to be fluid and to get spells that would integrate correctly with the speed of the gameplay.

The other programmer of the team was in charge of the movement so this section will not be given too much attention in the following explanation.

As for the spells, that was the part that took more work during this first phase of development. Together with the designer of the team, we were working on the design of the different spells (can be found in the first version of the GDD included in the appendices) and we managed to program and implement seven of them, which are described below:

- Pull Objects.

This is the first spell that was created. It's the simplest of all, next to the next one on the list. The mechanics consist in that certain objects distributed by the stage are interactive and by means of the press of a key, the player can attract them (only one at a time) and take them with him, until he decides to throw them.

This mechanics allows one to hinder the movement of the opponent and even to deviate it from its trajectory.

- Walls.

The *wall* is a mechanism that consists of certain objects on the stage that can be activated by pressing a key and its size grows vertically.

This mechanism allows cutting paths for both the opponent and the ball.

- Projectiles.

Projectiles are a mechanism by which a number of projectiles are fired and directed towards the ball, chasing it until it hits it.

This is because the ball was originally intended to have a not too complex AI, and if it didn't interact with players for a certain number of seconds, it would start to move independently. After several testing sessions, this did not convince the players as it was not predictable and made the game quite difficult for them so it was decided to leave the idea aside and the mechanics of the *projectiles* lost a little bit of its meaning.

- Dash.
A movement mechanic that allows the player to move a great distance in a short period of time.
- Push.
This mechanics consists of a strong push in front of the player. When the player presses the corresponding key all moving objects in front of the player (whether moving parts of the stage, players, or the ball) will be moved by a great force.
The mechanics are adorned with a whirlwind effect to give the impression that the player was using magic to summon the wind and create a great force.
- Roots.
Roots is a mechanism that has been created to control the rhythm of the gameplay and prevent speed from making the games unplayable.
Roots is a spell that the player can place wherever he sees fit (in its first version, in its second implementation the *roots* were made to be fixed elements of the scenario and the player had to activate them).
When the ball enters the field of action of the *roots* a force attracts it to a fixed position where it will remain until the lifetime of the spell ends.
This spell created many possibilities for combination with other spells created previously since if the player was close he could use the *push* to push the ball after fixing it or if he was far away he could attract a nearby object and throw it against the ball to move it to where it was most convenient.
- Geyser.
This is the latest mechanics created for the MVP. It contains quite a few similarities to the *roots* as it also has an area of effect and two implementations of it were made (the first being a spell that the player could place wherever it was most convenient and the second being a fixed element of the scenario).
The mechanics consist of an area that once placed creates an upward force that moves everything inside it (whether it is the ball, a moving

stage element, or a player). This can be used by players to interrupt the movement of their opponents, to move the ball over obstacles placed by the opponents or to make very fast movements (such as placing a *geyser* in front of you, using its force to shoot upwards and using the *dash* to advance a long-distance).

In addition to all the elements already mentioned for this first phase, the different menus were also created and the selection of characters began to be devised, although nothing was implemented since it was an element more related to the monetization of the product.

4.1.1.3. MVP testing

During this first phase of production, the work done on the project had to be combined with different subjects of Hanze, and one of them required each member of the team to do a test session. Each one tested a different mechanic of the product, but for this document, the aiming system is going to be explained.

The testing required the use of at least two Design Research methods. These methods are mechanisms that help developers find the feedback they are looking for for the tested elements. Each has its pros and cons, with some being good at understanding players' feelings, but bad at finding the specific mistakes that are being made, while others serve the opposite purpose.

In my case, Usability Testing and Critical Facet Playtest have been used. In the appendices, you can find the document developed during this session with the explanation of the methods in more detail, because they have been used and their pros and cons.

Once the methods had been chosen and the reasons for their use understood, the test was carried out with three different participants. The session was planned as explained in the document in the appendices and consists of the following two parts:

- Playing Part

In this first part, the test participants had to follow a series of instructions given by the analyst to complete a series of challenges. Since the pointing system was being tested, participants were first given a skill test to determine their level of accuracy in pointing at different items on a screen. Once this test was done, the next test was done where the players had to use the shooting mechanics to hit different targets that were located around the stage.

With the results of both tests, the calculations found in the tables of the test document found in the appendices were made to determine how the pointing system improved or worsened the players' skill.

In addition, during this phase, the test organizer observed the participants perform the different tests to record all their reactions and behaviors in order to analyze them in more detail and draw conclusions.

- Survey part

In this second part of the testing session, the participants had to fill in a survey, created specifically for this test, with questions about the pointing system. These questions consist of trying to understand the participants' feelings when using the pointing system in order to better understand their reactions during the first part and to be able to know if the development of this element was on the right track.

The survey can be found within the test document in the appendices.

The results of these sessions can be found in the document, but overall, users found the pointing system very accurate and comfortable to use. After analyzing the data obtained during the first part of the test, it was observed that the system hardly affects the players' ability when using it, so it was possible to determine that the work done was correct and that this path should be followed.

One last piece of information about the testing of the aiming system is that it allowed finding a bug when using the shots because the projectile fired had a slight deviation to the left of the screen that made some players miss some shots that should have been hit.

4.1.1.4. MVP results

Once the MVP was over, the professors were introduced and several testing sessions were held at events and with fellow university students. This provided the team with a lot of feedback to work with and generally fairly consistent feedback.

Both teachers and classmates commented in their feedback that the gameplay and the idea were very attractive, each person that knew about what was being created liked the idea and, when playing it, they had very good feelings but they all agreed on the same thing:

- The gameplay has too many mechanics and being such fast gameplay the players don't have time to understand what elements they have to use.
- The ball having its own artificial intelligence makes the players confused.
- The stage is very loaded with elements and makes the movement of the ball and the players difficult, emptying it and leaving space would help to make the gameplay clearer.

Therefore, certain changes were decided to be made to the prototype.

4.1.2. Prototype

Once the MVP was completed and with all the feedback received from colleagues and teachers, it was decided that we should make certain changes following the recommendations. To begin with, it was decided to reduce the mechanics of the game, as the 7 spells were too many and it has been reduced to 3, combining some of the previous ones. In addition, the number of elements on the stage has been reduced and it has been opened so that players can move more freely.

On the other hand, the most important element of the game has been implemented, which is the multiplayer system. This has meant many hours since none of the components had any experience programming a multiplayer system or using multiplayer in Unity, and knowing that I was the most experienced programmer on the team, that task became mine. For this, we have used a Unity extension called Mirror, which has great documentation to learn how to use it, it is based on the old Unity multiplayer system, and through Discord, they attend any doubt in moments where the project was stuck.

In addition, the first phases of character selection have been implemented, creating a selection system with its own menu. Finally, something to be done in this phase of development is to improve the graphics of the game and the menus.

4.1.2.1. Prototype planning

- Create 3 new spells based on the older ones to make the gameplay less complicated
- Upgrade the art of the game
- Create a multiplayer system
- Create a character selection screen
- Make an open arena to allow a more fluid gameplay

4.1.2.2. Prototype development

The initial step in this phase of development was to implement the new spells. Together with the designer of the team both programmers worked on how to create the new spells and came up with these three:

- Fire Spell.

This mechanics is the combination of the *push*, the *projectiles*, and the objects that could be thrown. These spells have been combined because players wanted a spell that would deflect the ball as if it were a push but wanted to have some control over the direction of the ball, so the new spell consists of a fireball that the player throws and explodes on impact against the first element he finds in his path (either ball, player or scenario).

With this spell, the team wanted to add a little bit of skill to the gameplay, since the intention with the first version of the game was to make the players master the different mechanics to be able to move around the stage while controlling the ball, but seeing that this didn't convince the people who tried our game, the team wanted to introduce the aim as a mechanic in this spell that allows differentiating the good players from the casual ones.

- Ice Spell.

This spell is the counterpart to the previous spell. Just as the first spell adds speed to the ball, this one is the opposite, it serves to stop the ball where it is at that moment (If the ball is in the air it will fall by gravity but lose all other movements). This spell will only affect the ball, not other players.

These mechanics have been designed inspired by the mechanics of MVP which was called roots. The difference with the mechanics that it is inspired by is that the previous spell placed the ball in a fixed place while this stops it where it is for an instant and then external forces such as gravity affect it again, so if players want to combine both spells they have to be able to predict the movement of the ball after stopping it, requiring some skill to master these mechanics.

- Blink.

This spell is almost exactly the same as the *dash*. Spell's name have changed and new VFX have been added but it does almost the same. The only difference now is that you can dash in any direction, and in the MVP you could only dash forward.

Once the basic mechanics are completed it is time to implement the multiplayer system. This is the other key element in the development of the video game. Without a multiplayer system nothing that has been done so far makes sense so it was something that we had to complete. Since no one of my teammates had much experience in programming, all the development of this system is done by me and most decisions have been taken unilaterally because no other teammate wanted

to add anything, due to the fact that they were not collaborating on the development, they didn't want to slow it.

The first attempts to develop the multiplayer system were made with Photon, software that provides a dedicated server for up to 20 users at a time, but the documentation was not very complete and Mirror appeared. It is an extension of Unity that was based on the old multiplayer system of the engine and with a very complete documentation, many examples included in the Unity asset and a Discord server where people can contact the developers quite directly.

For this multiplayer system the first requisite was to connect two players so they could play a game, and, since the equipment we have is quite limited and creating dedicated servers is out of our capabilities, right now the game works with a Host-Client relationship being one of the players the Host. This avoids having a dedicated server, but creates the problem that the quality of the game depends on the host.

The operation that has been implemented can be followed in the explanation of the different interfaces in the previous section of this report. Players have the option when entering the game of whether they want to host a game or join one. To become the host, players simply have to press the Host button and wait for the necessary players to join. Right now, only players are required for a game, but this system can be extended to 4 or 6 players at a time, organized by teams. In case of joining a game, players must enter the IP address of the host and press *join*. Once the players are in the lobby, both can press the ready button to change their status and once all the players are ready, the Host can press the start button and start the game.

To add more personalization to the players, the option of being able to select a username that will appear in the lobby and also an avatar image have been added . The username is geared towards players being able to identify each other while the avatar image opens up new possibilities for monetization of the game, as it is not just the character skins that users can buy.

Finally, in this phase of development, it was necessary to start implementing a character selection screen, since this is where all the monetization of the video game

resides, there has been quite a lot of emphasis on it throughout the development and it is one of the key points why Hanze's teachers gave the green light to this project. Right now, the game only has one character so the other slots are blocked until the artist develops more characters for the game.

The character selection screen works quite simply. Players can enter this screen and choose from the different characters available, viewing a three-dimensional model of the character to see how it will look within a game. This has been chosen because many games such as *Overwatch* or *Apex Legends* include this possibility and after a survey among users of competitive games one of the biggest complaints we have gotten from players of games like League of Legends is that spending money based on a fixed image that does not accurately reflect the look of the skin within the game leads players to feel cheated.

4.1.2.3. Prototype results

Once the development of the prototype was completed during this second phase, a presentation of all the projects developed during the Game Lab was made. The teachers organized a Discord server with different voice and text channels for each group of students. Each group had a text channel where they could expose and promote their game, a text channel for download links, and another text channel so that both teachers and students could express their doubts and questions. In addition, all groups had a voice channel where they could speak more directly to anyone who wanted more direct communication with the developers.

For all the groups a schedule was organized for the presentation of the games. For one hour, the game's developers gave two presentations where anyone on the server could come in and see how the development had gone, watch some gameplay, and ask questions at the end of each presentation.

The results received from these presentations and the feedback given by students and teachers was that the game feels very fluid and is fun to play. Many hours had been spent playing with partners who came through our channels and wanted to play the game, and one of the main strengths the game had was that othe

teams' games had a beginning and an end, so it was not possible to enjoy them more times, while with our game you could play without limit and with very short games, so people were encouraged to come and play more or play with their partners.

4.1.3. Final product

For the final product, it is already a phase in development on its own that consists of improving certain aspects of the product made for its presentation in Spain. Most of the work done in this part of the development is on improving the multiplayer system, as by the end of the second phase the system had many bugs in terms of players disconnecting and adding the endgame scenes.

Finally, the work outside the multiplayer system is the polishing of details in menus and interfaces such as the selection of characters or the interface within the game.

4.1.3.1. Final product planning

- Handle disconnections from players
- Create a post-game screen
- Polish the game

4.1.3.2. Final product development

The development of this last phase has been focused on finishing the polishing of the multiplayer system to make it a functional system since for the prototype it was a system that allowed two people to play but didn't take into account things like disconnections or endgames. For this phase, the work has been on solving all those errors to have a playable system.

First of all, the disconnections of the players have been managed. In the previous phase, if the client disconnected from the game, the host stayed inside the game and could play, and if the host disconnected from the game, both players disappeared, but the client remained inside the game screen without being able to leave the game except by closing the application externally. Now, when one of the players is disconnected, the game is granted to the other player as a win.

Finally, some operational details have been polished in the main menu and some animations have been added to polish the final product.

4.1.3.3. Final product results

The changes made in this last phase make the product look much more finished avoiding errors such as disconnections or not having a screen at the end of a game.

5. Conclusions and Future Work

5.1. Conclusions

Carrying out this project in another country and with a large team has been a different experience to what we have lived so far during the years in the degree. On the one hand, having a bigger team and not being a single person has allowed me to be more ambitious and aim for things like a multiplayer system that I would not have dared to do in an ordinary situation. But he's also included his bad parts. Working with sophomores has meant that I have had to take the reins of the team, and it's something I'm comfortable with, but the workload this has entailed along with the situation I've been in that I haven't been able to work in person with my colleagues has made things a lot more difficult.

On one hand I'm satisfied with the work done. In this work I created a multiplayer system for the first time and I have expanded my limits to create things that until now I had not been able. Also I have been able to apply many design concepts and market research and target audience that until now had not been treated in-depth and are something essential in the development of a video game.

But on the other hand, working with inexperienced people has been a burden and I have had to take on a greater amount of work than I should have. This added to the fact that even though the group was made up of six people, the work was mostly done by three since three of the participants had practically no contribution to the project.

In general, this project has helped me to learn on my own and experience for the first time a real development of a video game and after that, I feel more qualified to undertake a journey in the working world.

5.2. Future work

After finishing the final product for presentation in Spain, the product will no longer be in development. None of the colleagues have decided to go ahead with this project so their last iterations are my improvements for their last presentation.

6. Bibliography

6.1. Bibliography

Mirror Networking. (s. f.). Mirror Networking. Recuperado 20 de marzo de 2020, de <https://mirror-networking.com/docs/>

Unity. (s. f.). Unity User Manual. Unity User Manual. Recuperado 10 de febrero de 2020, de <https://docs.unity3d.com/Manual/index.html>

Psyonix. (2015, 7 julio). Rocket League. Rocket League. <https://www.rocketleague.com/>

Photon. (s. f.). Photon Unity Network. Photon Engine. Recuperado 15 de abril de 2020, de <https://www.photonengine.com/pun>

6.2. Source Code

<https://github.com/joanely3000/TFG>

6.3. Appendices

6.3.1 Test document

This document is included in the appendices of the report because section *MVP Testing* explains in general terms the testing process carried out during the development of the product. Here you can find all the data on the analyses of the methods used, the results obtained, the conclusions drawn from them and the appendices with all the material obtained during the testing.

6.3.1.1. Introduction

This report describes the building, process, and results of the study of the conduct of a product in development. We have done Game User Research of a target component's user experience.

The Game User Research is something necessary for the development of a video game nowadays. The idea is to achieve the best possible design applying scientific methods and UX design principles and understanding the needs and requirements of the players. Also, the game research academic area tries to understand the players, their motivations, and how to predict and explain their actions.

The main idea of this project is to test the aiming system of the game we are developing. During this report, you will find an explanation of the context of the project to understand its functionality, including an explanation of the game in development, an explanation of what is being tested and why, an analysis of the different possible evaluation methods and a little brief on why I have chosen the ones I chose, the operationalization of the variables that need to be measured to answer the main research questions and the execution of the test.

All the data collected, the elements used, the recordings, and transcription of the testing sessions can be found in the appendix of the report.

6.3.1.2. Project Framework

Before starting the analysis we are going to do, the project we are making needs to be explained and the context it is being developed to be able to understand the problem. The game we are developing is a multiplayer competitive focused on fast gameplay, which is in the final part of its development. Some components and mechanics of the game need to be analyzed and tested to implement them correctly.

The game is a competitive multiplayer game where you play a new sport based on traditional football. The main concept of the game is to make the player feel in total control of the character and rhythm of the game and everything that happens around them at the same time that fast gameplay is happening so we have to make sure the mechanics are as polished as possible so the player does not feel he is not in control at any moment.

The target audience has the main characteristic that are players who enjoy playing competitive games where their skill is determining the fate of the match. Those players are used to precise aiming systems where they have all the control, and they even enjoy spending time improving their skill to be able to be the best in that game and have the highest rank. Also, short matches are something that can attract new players who don't play competitive games due to the time they take.

In the appendix of the report, the Design Research Canvas can be found. It has been used to guide the analysis of the aiming system, find the research questions that need to be answered, and find the evaluation methods that will help to find the answers.

6.3.1.3. Objectives & Research Questions

Knowing this is a complex game that is being designed and developed by different people, this analysis will focus only on the aiming system of the player. This system has 3 key concepts: Camera movement, aiming precisely, and the player's input response. It's already been said that the game tries to get the maximum out of player's skill, and that's why we want to make the player feel comfortable when playing and never feel like they are playing an unfair game. This is crucial for the project because all the competitive games already have this or are very close to getting the best aiming system possible. What we want with this analysis is find how good our system is and how we can improve it.

6.3.1.3.1. Research Question

Can the aiming system make the players that play competitive games based on aiming and precise feel like they are in control during all the playing time and never feel they aimings is unfair or is not behaving as expected?

6.3.1.3.2. Keywords

- Comfortable
- Precise
- Responsive

6.3.1.3.3. Sub questions

- How comfortable do the players feel the movement of the camera is?
- How precise do the players feel the aiming system is?
- How accurate is the camera movement when reacting to the player's input?

6.3.1.4. Possible Research Methods

Once we have done this, we have to choose which methods we are going to use during the testing session. After looking at different methods, we discarded those ones that due to technology limitations are not possible to implement or it would take too much time and resources. Due to this problem, everything related with telemetrics has been left over, even knowing that telemetrics would help a lot to this analysis.

After doing this, there are a few methods we can choose from, we are going to analyze and compare three of them that we think are the best ones for the mechanics we are trying to test on this project. After that, we will choose two of them.

All the methods described in the next paragraph and the information related with it are from "Games User Research" (OUP Oxford, 2018). More precisely, it is from chapter 7 written by Michael C. Medlock.

6.3.1.4.1. Interview

This method is about interviewing the participants after the testing session, asking them questions related to perception, opinion, thoughts, attitudes about the game, the gaming experience, and the mechanic that is being tested.

Advantages:

- It is good at understanding the individual point of view of the player about the game experience.
- It can help to better understand why the players do what they do

Disadvantages:

- It is not good at getting data related to the player's behavior when playing the game.
- It does not quantify the attitude of the players regarding the experience.

6.3.1.4.2. Usability Testing

This method is about watching the participants trying to complete different simple tasks of the prototype or the game in development. The researcher takes notes of all the comments the participant says and all the actions the participant does, while also being recorded to analyze it later.

Advantages:

- It is good at identifying problems that are not allowing the players to have to correct experience while playing the game.
- It is used to identify why the players are not able to complete the tasks that they have to do or enjoy the game as intended.

Disadvantages:

- It is not good at understanding and quantifying attitudes.

6.3.1.4.3. Critical Facet Playtest

A survey technique that evaluates attitudes and perceptions about very specific key experiences.

Advantages:

- It helps to find quantified perceptions about the attitudes of the users when experiencing the game.

Disadvantages:

- It doesn't show if the overall experience is being good.
- Because it is an auto evaluation it doesn't explain why players are feeling that way.

6.3.1.5. Research Design

In this section, we are going to see how the answers to the research questions have been found. We are going to test the aiming system, we know we have to test how comfortable the camera movement is, how precise it is and feels, and how the response to the player's input. We can get three key concepts out of here, and if we analyze them we can find a precise answer to the research questions:

- Camera movement comfort
- Aiming precision
- Input response

6.3.1.5.1. Operationalization

These concepts must be operationalized (make something measurable) to be able to get all the data required to elaborate on a better design of the aiming system.

Construct		Definition
Controls comfort		How comfortable the players feel when using the aiming system
Dimension	Definition	Indicator
Satisfaction	How satisfied the players are when using the aiming system	Evaluation of the players on the survey

Construct		Definition
System precision		Grade of precision the players perceive between where their shots are intended to go and where they really go
Dimension	Definition	Indicator
Precision	How precise are the shots of the player	Number of shots hit and missed by the player during the test
Perception	How precise the player perceives the system is	Evaluation during the survey

Construct		Definition
Input response		How good is the response of the system to the camera movements of the players
Dimension	Definition	Indicator
Correctness	How much the player can move the camera without feeling the system is not responding how the player wants	Evaluation during the survey

6.3.1.5.2. Evaluation Methods

To measure the dimensions that have been operationalized before two methods have been selected, those are Usability test and Critical facet playtest. We have chosen the first one because we can use it to find the exact problems that are not allowing the player to enjoy the game experience when using the aiming system and find the reason why this is happening. But even being able to provide qualitative data about all three questions, it will not help us to understand or quantify the attitudes, neither understand if the global experiences are being good. But to do this we choose the Critical facet playtest, this method succeeds where the Usability test fails and allows us to find the quantitative data about the attitudes of the players about certain aspects of the game. These two methods have been chosen because of how well they complement each other and also because they can be used with the same testers and that makes the testing easier.

To test and analyze the precision of the aiming system of the game we will execute a test during the usability test where the players will have to click on different circles when they appear on the screen. This first phase will allow us to know the skill of the player when aiming. When this first part is done we will move on to a real aiming system test. Players will have to aim and shoot different targets that will appear over the level. To calculate how the aiming system is working we will use these functions:

- Player skill results: $\frac{Hits - Mistakes}{Number\ of\ targets}$ on the first phase

- Aiming system results: $\frac{Hits - Mistakes}{Number\ of\ targets}$ on the first phase
- Aiming system behavior: *Aiming system results* – *player skill result*

This will give us a number between - 1 and 1. If the number is near 0 it means it is not affecting the player skill (which is what we want) and if the number goes over 0 it means it is making the player better (something not likely to happen).

During the critical facet playtest we will ask the participant how he perceived the precision of the aiming system. The answer can be between 1 and 5 and we will normalize that value.

To see how precise is the input response of the aiming system we will observe the players while they complete the simple tasks of the usability test and we will ask them to tell us whatever they think is not working on the aiming system.

Furthermore, we will ask the participants on the Critical facet playtest survey how they felt the input response. The answers will be between 1 and 4 and we will normalize that value.

And last but not least, to test how comfortable the aiming system is, the observer will take notes of the players' reactions when completing the tasks of the Usability test to see if he is comfortable using the system or he is struggling.

Also, when completing the Critical facet playtest survey they will find a question about the comfort of the system. The answers will be between 1 and 5 and we will normalize that value.

6.3.1.5.3. Target group

Regarding the participants there's no specific requirement if they are in the target audience we have described before. Due to the dimensions of this project with 3 participants will be enough.

6.3.1.5.4. Evaluation execution

The test is composed of two parts, the first one will be the Usability test. The participant will be asked to play a simple game that will determine his aiming skills

clicking targets that will appear on the screen. Once this is done, the test will move on to the second phase of this part. The player will have to aim and shoot to different targets that will appear around the level, and while he is doing this he will be asked to say anything he thinks is not working with the system. When the first part is done the participant will have to complete a survey for the Critical facet playtest.

The test will be done in a familiar environment for the participants. There's no need of using a lab for this and that will help the participants to feel comfortable and express their opinions openly. The test will be realized in the presence of the observer, who will guide the test that can be found in the appendix of the report. Lastly, the conversations between the observer and the participants and the screen of the players will be recorded and can also be found in the appendix.

6.3.1.6. Preparation and execution

6.3.1.6.1. Pre-Test

Before executing the real test a pre-test was done to check that everything worked fine, the instructions were clear and the participants could complete it without any errors. The pre-test was executed with people who knew the project and everything went as planned, with no errors, and the instructions were clear for the participants.

6.3.1.6.2. Participants

For this test, the participant members should have been part of the target audience and not related to the project or game development in general. Due to the dramatic situation that occurred with the COVID-19, this has not been possible, and the participants selected for the test have knowledge about game development.

6.3.1.6.3. Execution

The tests were programmed to happen during the same day. One of them happened during the morning through Discord. The participants received the demo of the game and a document with links to the agreement and critical facet playtest surveys. The dynamic of the test was explained to the participant and then we started the test. The other two participants followed the same process but they were in the same room as the observer, and the first participant was with his camera on (after agreeing they were comfortable doing that).

All tests were recorded and the recordings can be found in the appendix of the report. Also, the translation and transcription of them are there.

Apart from that, all the tests were executed without any big problem (some framerate problem during one test but nothing too problematic).

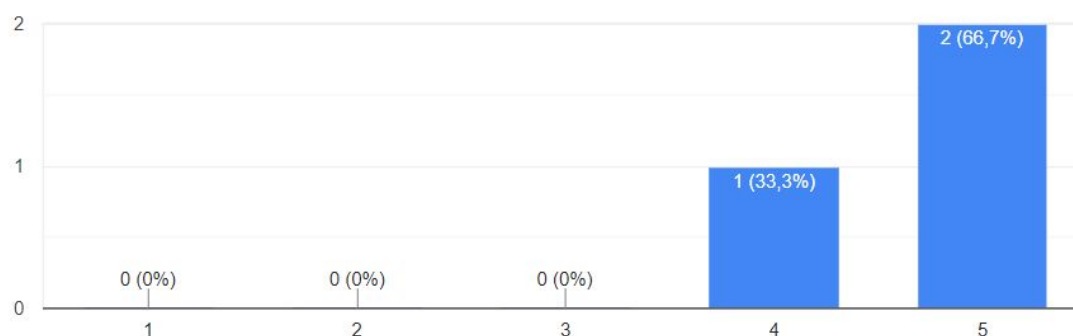
6.3.1.7. Results and Analysis

The test has been very productive in general and the objective that was to know if the aiming system was working properly has been reached. A lot of useful information about the system has been collected and can be found in the appendix with the usability test notes and the translation and transcription of the sessions.

The results of the test are quite interesting. The participants answered almost the same on the critical facet playtest survey. As can be seen in the results that can be found in the appendix, all the answers were the same except on the question where participants were asked how comfortable the aiming system was. Two participants answered it was perfect but one of them said that it was good but not perfect.

How fluid and comfortable was the aiming system

3 responses



On the Usability test, the results have been consistent too, all the participants said that the crosshair was too big and could make you miss some shots. Furthermore, all three participants have found a bug (not related to the aiming system, but creates the feeling that the aiming system is not working properly). Fixing this bug will help to make the product more competitive and accurate.

Something curious about this bug is that the players that were more used to play shooters fixed this deviation unconsciously and reported that something was not working on the system, but the participants that are not that used to play shooter games reported this bug while playing the demo and showed it so it would be recorded.

On this table the results of the measured dimensions can be found:

Dimension		Scale	Measurement
Controls comfort	Satisfaction	Normalized value	0.93
System precision	Precision	Scale value that goes from -1 to 1	-.22
	Perception	Normalized value	0.8
Input accuracy	Correctness	Normalized value	1

6.3.1.8. Conclusions & recommendations

6.3.1.8.1. How comfortable is the aiming system?

The participants found the camera movement very comfortable. We can see this on the table above where the value of comfortability is 0.93 over 1, which means that it is almost perfect.

6.3.1.8.2. How precise is the aiming system?

To measure the precision of the system we used two methods. The first one says how the system affects the players' skill and the other says how precise the system is perceived by the players.

Looking at the data we can see that the system is affecting the player's skill, but has a very small weight on them, and with practice and more use the players can end up having total control over the system without it affecting them. Furthermore, the players feel the system as very precise.

While the participants were playing the game we could see them enjoying the moment even when they found a bug. This says that the system is doing its job right now and the values point at this. A couple of things will need to be modified to get a better experience but in general, we are doing the right things.

6.3.1.8.3. How accurate is the input response of the aiming system?

All the participants have found the input response very pleasant. This is something very important for the project because if the players don't feel the game is acting as they want, they won't be able to improve their skills and can feel some unfair events are going on.

All this data says that we are going on the right way, there are some things to improve but it feels fast and accurate and if the players spend enough time they can end up controlling the game at a competitive level.

6.3.1.8.4. Recommendations

In the end, these are the changes we should make to achieve a better aiming system:

- The crosshair should be smaller so it does not disturb the aiming.
- The crosshair should change color when being over a target.
- We must correct the deviation on the fireball spell.

6.3.1.9. Appendix

6.3.1.9.1. Design research canvas (first version)

<p>Explore the problem (or opportunity)</p> <p>Aiming system</p>	<p>Explore solutions</p> <p>The movement needs to be precise and comfortable</p>	<p>Key Concepts</p> <p>The final product should:</p> <ul style="list-style-type: none"> - Be similar to other competitive games - Let the player explode his skills without interfering - Be precise <p>Some of the competitors are:</p> <ul style="list-style-type: none"> - Rocket League - Overwatch - Counter-Strike - Valorant 	
<p>Define problem and problem domain</p> <p>The player expects a precise and responsive aiming system</p>	<p>Define principle solution</p> <p>The aiming system needs to be:</p> <ul style="list-style-type: none"> - Precise - Responsive 		
<p>Define constraints and requirements</p> <p>The aiming system should be like other competitive games where there are no random elements that can interfere with players' precision</p>	<p>Ideate, specify & represent</p> <p>How can the aiming system be better?</p>	<p>Depict & prototype</p> <p>What can we make with the current aiming system?</p>	<p>Evaluate</p> <p>How good is our current aiming system?</p>

6.3.1.9.2. Design research canvas (last version)

<p>Explore the problem (or opportunity)</p> <p>The aiming system is a key component of the game. Having fast gameplay on this game will require us to create the best system as possible so the players can be in control all the time. They have to be doing many things in a moment and the systems needs to be something that works well enough so they don't have to think about it</p>	<p>Explore solutions</p> <p>The movement needs to be precise and comfortable similar to other competitors' systems.</p>	<p>Key Concepts</p> <p>The final product should:</p> <ul style="list-style-type: none"> - Be similar to other competitive games - Let the player explode his skills without interfering - Be precise <p>Some of the competitors are:</p> <ul style="list-style-type: none"> - Rocket League - Overwatch - Counter-Strike - Valorant 	
<p>Define problem and problem domain</p> <p>The project's target audience main characteristic is the love of competitive games. And the main characteristic of competitive games is they make you feel in control of every situation, so the system needs to fit those requirements and be as precise as possible and make the player feel comfortable.</p>	<p>Define principle solution</p> <p>The aiming system needs to be:</p> <ul style="list-style-type: none"> - Precise - Responsive to the player's input - Players must feel comfortable with it - Players must feel it is a fair system 		
<p>Define constraints and requirements</p> <p>The aiming system needs to :</p>	<p>Ideate, specify & represent</p> <p>The crosshair needs to be smaller, the camera</p>	<p>Depict & prototype</p> <p>The system needs improvements on</p>	<p>Evaluate</p> <p>Our current aiming system is working</p>

<ul style="list-style-type: none"> - Be considered as good as the competitors' systems - Be considered fair and make the player feel they are in control in every moment - Be as precise as possible so players can't feel imperfections - Be comfortable so players would want to play with it 	<p>movement needs to adapt it's speed to the player and there should not be deviation on projectiles</p>	<p>some points like the crosshair.</p>	<p>properly right now, the players are happy with how it works and enjoy playing the game when using it. It's not perfect but it is getting close to competence.</p>
---	--	--	--

6.3.1.9.3. Test questions and instructions used by the researcher

Describe the game and explain the nature of the study.

Ask to sign the consent form and clarify if any question are arise.

Ask de participant to share his screen and launch the game.

The first part of the test (Player skills)

- Explain the exercise to the player
- Tell the participant to play it
- Take note of the participant mistakes

Ask the participant to press the continue button.

The second part of the test (Usability test)

Tell the participant: "Now you will have to shoot different objectives that will be appearing in front of the player, use the left click button of the mouse to shoot, each time you hit an objective it will disappear and the next one will appear. If you take too

much time to shoot the objective will disappear and it will count as a mistake. If at any moment you feel the system is not responding as you wanted please let me know, and also tell me everything you think about the system during the exercise.

Press the spacebar when you are ready”.

- Ask the player to move the camera
- The player will shoot all the objectives

Tell the participant to stop screen sharing so they don't feel uncomfortable

Ask to fill in the survey.

6.3.1.9.4. Agreement survey

Agreement Survey

Descripció del formulari

I voluntarily agree to participate in this research study

Yes

No

I understand that even if I agree to participate now, I can withdraw at any time or refuse to answer any question without any consequences

Yes

No

I can understand that I can withdraw permission to use data from my interview within two weeks after the interview, in which case the material will be deleted

Yes

No

I understand that if I inform the researcher that I or someone else is at risk of harm, they may have to report this to the relevant authorities - they will discuss this with me first but may be required to report with or without my permission

Yes

No

I understand that under freedom of information legalization I am entitled to access the information I have provided at any time while it is in storage as specified above


Yes

No

I have had the purpose and nature of the study explained to me and I have had the opportunity to ask questions about the study

Yes

No

<p>I understand that all information I provide for this study will be treated confidentially</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>I understand that disguised extracts from my interview may be quoted in design research report</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>I understand that signed consent forms and original audio recordings will be retained on Google Drive account of Joan Dalmau until the exam board confirms the grades for the project</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>I understand that I will not benefit directly from participating in the research</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>I understand that in any report on the results of this research my identity will remain anonymous. This will be done by changing my name and disguising any details which may reveal my identity or the identity of people I speak about</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>I understand that I am free to contact any of the people involved in the research to seek further clarification and information</p> <p><input type="radio"/> Yes</p> <p><input type="radio"/> No</p>
<p>Signature of the research participant</p> <p>Text d'una resposta breu</p>
<p>Date of signature</p> <p>Mes, dia, any </p>

6.3.1.9.5. Critical facet playtest Survey

Aiming system survey

How much did you like the aiming system overall?

1 2 3 4 5

I hate it I love it

How precise do you think the aiming system is?

1 2 3 4 5

I cant aim with it It is very precise

Did the system respond how you wanted to your input?

Not at all

It was lagging a lot

Just failed a couple of times

It was perfect

How fluid and comfortable was the aiming system

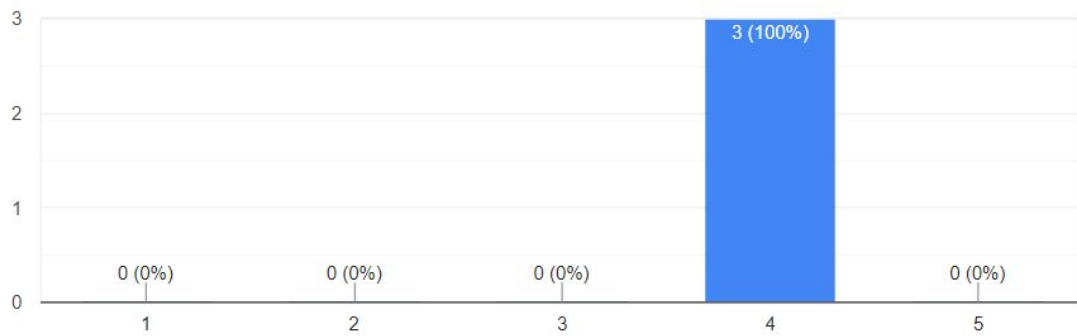
1 2 3 4 5

It felt horrible It felt nice

6.3.1.9.6. Critical facet playtest Results

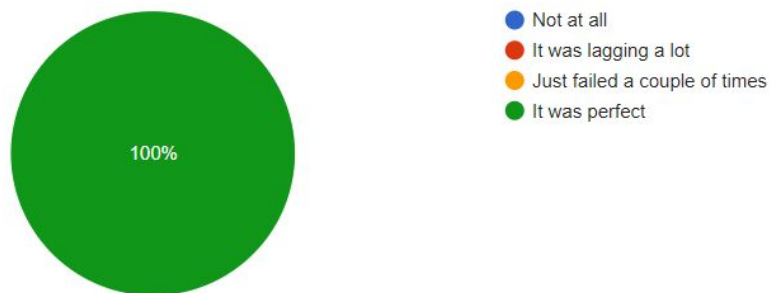
How much did you like the aiming system overall?

3 responses



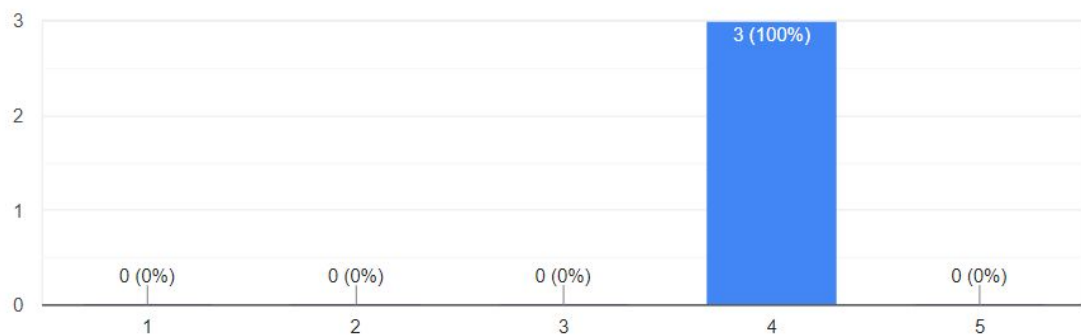
Did the system respond how you wanted to your input?

3 responses



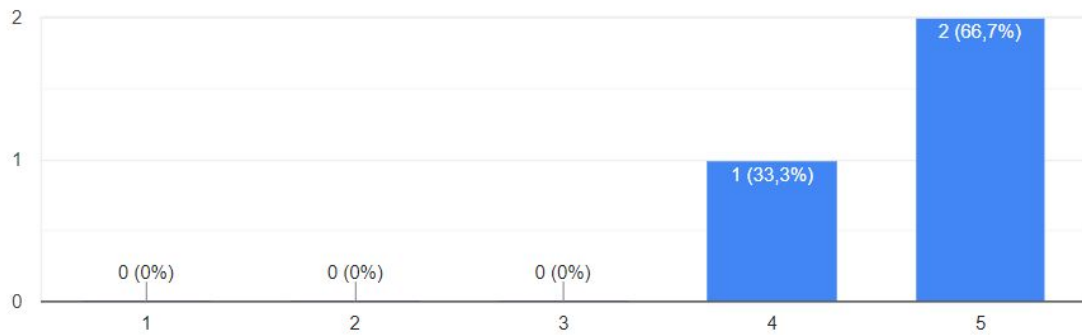
How precise do you think the aiming system is?

3 responses



How fluid and comfortable was the aiming system

3 respostes



Participant	Exercise	Made shots	Failed Shots	Percentage
Participant 01	First Exercise	70	4	94%
	Second Exercise	66	18	68%
Participant 02	First Exercise	70	1	98%
	Second Exercise	66	10	80%
Participant 03	First Exercise	70	3	95%
	Second Exercise	70	20	71%

6.3.1.9.7. Usability testing notes

6.3.1.9.7.1. Participant 1

1. This participant is a shooter player
2. He tried to move before saying anything
3. Moved the camera before he was asked to
4. The crosshair is too big
5. First shots missed going by the left side of the objective
6. Corrected the aiming deviation without noticing
7. After being asked if he felt any deviation he said no
8. The participant said that he was not feeling the aiming system as precise as he wanted

6.3.1.9.7.2. Participant 2

1. This participant is not used to aiming systems
2. The participant was worried about how much time he would have between objectives
3. The game lagged a moment when starting the test
4. He said the sound helped him aim better
5. The participant noticed a little deviation on the trajectory of the projectiles
6. The projectiles are being shot to the left side of the crosshair
7. People with more experience in shooter games do not notice the projectile deviation, they correct it automatically
8. The crosshair is too big.

6.3.1.9.7.3. Participant 3

1. The participant tried to move when the test started
2. The participant moved the camera before being asked to
3. The player feels the crosshair is too big
4. The participant feels the crosshair should change color when being on top of a target

6.3.1.9.8. Usability testing transcripts

R - Researcher

U - User / Participant

6.3.1.9.8.1. Participant 1

R - Open the game

U - I'm on it

R - I'm going to explain to you how this works. This test will have two parts, first...

The participant tries to press the start button

R- Wait to press it

U - I know I know

R- First we are going to test your aiming skills with a simple test...

U - It is bad

R - ...And see how the coordination between your eyes and hand is. So we will do this test and then move on to the next part. On the second part, you will be in the game and some targets will be appearing around the environment and you will have to shoot with the left click, like a normal shooter

U - What color are the targets?

R - They are red in both exercises. So any doubt? When we move on to the next phase I will explain it in more detail

U - Sounds logic

R - You should know you can't take the time of your life to shoot to the targets

U - They have a lifetime

R - Exactly, if the time ends and you haven't shot at them it will count as a mistake. If you press and don't hit the target it's also a mistake.

Starts the test

U - How much do I have to be doing this?

R - Until you finish

The recording software crashed the PC so we had to start recording again, a little cut can be noticeable

U - Do I press continue?

R - Yes

U - Ok

R - This is the second phase, targets will be appearing around the level and you will shoot them with the left click. Every time you hit one it will disappear and the next will appear, and the same, they have a lifetime, if it passes the target disappears. I am asking you to tell me whatever goes through your head, if the system is not responding as you want or whatever, ok? Tell me anything.

U - Ok do I have to do something so it starts?

R - Yes. Behind you won't appear any target, just in front of you and your sides. Whenever you want, press the spacebar and start.

U - Ok, let's go

U - Can I shoot as much as I want?

R - Yes, but there are mistakes, remember each failed shot is a mistake

U - Ok Ok

U - Maybe the crosshair is too big. Sometimes it seems you are going to hit the target but you don't

[...]

U - Gravity doesn't affect projectiles right?

R - No

[...]

U - It seems it is going to disappear before the projectile arrives

U - And if I do this?

Tries to control the projectile after shooting

[..]

Looking for the next target

U - Am I done? Oh no, there it is

Ends the exercise

U - Ok, do I press finish?

R - Wait for a second

R - Okay, press Finish, and now if you want to stop recording to be able to answer without feeling uncomfortable with me looking you can stop.

6.3.1.9.8.2. Participant 2

R - Targets will appear around the screen. Just click on them

U - Click whatever is on the screen?

R - Right, if you click out of the target it will count as a mistake, and if it takes you too long it will count as a mistake too.

U - Vamos com valorant / Like Valorant

R - More or less, so...

U - Do I have enough time or I gotta be fast?

R - The screen is big enough, you have 3 seconds per target, you should have enough

U - I just ask to know if I have to run

R - Don't worry, this is just to see your aiming skills

U - Ok

R - You can start whenever you want

U - The DPIs are too slow, but I guess I am more accurate this way

R - Adjust them as you wish

[..]

U - Missed one because the DPIs are a bit slow, but with the size of the targets is not easy to miss

R - you can adjust it

[..]

U - How many targets there are?

R - 70

U - Why?

R - To give you time to get used to the aiming system. The number in the next exercise will be the same

Finishes the exercise

R - Don't press the button

The participant is complaining about the cursor size and the light in the room

R - Now comes the important part

Keeps complaining about not important stuff like wires and light

R - Press continue and don't press anything else. It will change the scene

Changes scene

R - You see you have a crosshair, and now comes the aiming system test

U - Do I have to move the character?

R - No, just aim. Targets will be appearing around the level, not behind you but some of them on your sides

U - 180° more or less

R - Yes more or less. You have to shoot them, now you can't because it is disabled, and every projectile you miss it will count as a mistake, and if you take too long and the target disappears it will count as a mistake too.

U - Ok

R - So I think I'm not leaving anything... Wait, if anything on the system is not working as you want, just tell me, anything you want. And that's it, so whenever you want, press the spacebar and start

The game lags

U - Having sound helps to aim

[...]

U - Do I have the same time between targets that in the other test?

R - No you don't

U - Because the projectile takes longer to arrive at the target

Game lags again

U - Don't know what happened there

[...]

U - I didn't miss that one

[...]

U - I can't see some targets because of the light of the level

We should take into account the person testing this is middle-aged

U - I'm putting the crosshair on top of the target and I'm missing. The projectiles are not being shot straight. Look! I just pressed when the crosshair was on top of the target and I missed

U - I'm shooting a bit to the right and I hit the target on the center. There's a little deviation on the projectile

Finishes the exercise

R - Okay, press finish and open the second survey, answer the questions

Stops recording

6.3.1.9.8.3. Participant 3

R - Okay, when you press the start button the targets will appear, before pressing it, if you click out of the target it will count as a mistake. There is a certain amount of time between targets so you won't take all the time in the world to click the next target, and if you run out of time

U - It will disappear

R - Yes, and it's a mistake.

U - Is it long?

R - Yeah a little bit, don't worry.

U - Do I need to hear anything?

R - Not really, in the first exercise there isn't even sound, this one is just to test your skill

U - Okay, I'm going to start

Starts the test

Finishes the test

* R takes note of the results*

R - Okay now comes the second phase. When you press continue you will appear on the game. If you click the left mouse button you will shoot projectiles. Targets will appear around all the level (not behind you) and you have to shoot them. If you take too long the target will disappear and it will count as a mistake and every shot you miss will count as a mistake too.

U - I only have to use the mouse?

R - Yeah only the mouse.

R - And if you feel something is not working with the aiming system or it is not responding as you wanted, just tell me.

U - At the moment?

R - Yeah at the moment, Okay? So press the continue button, it won't start immediately. And whenever you want, press the spacebar and it will start.

U - Are you ready?

R - Yes I am

Starts the exercise

U - The game is lagging

R - Yeah, I think it's the computer's fault

[...]

U - The laterals of the targets count?

R - Yeah they do

U - A miss is a mistake right?

R - Yeah it is

[...]

U - That wasn't a miss!

Finishes the exercise

U - Hits: 70, mistakes: 20

R - Okay

U - I missed a lot, I was shooting double and I shouldn't

R - Okay press finish and fill the survey pls

6.3.2 First GDD

In paragraph *4.1.1.2. MVP Development* explains the development of the MVP and what decisions were made regarding the design. Here you can find the first GDD where these ideas and decisions were reflected and used as a starting point from which to move forward. For the topic covered in this document the most important section of the DGD is *6.3.2.5. Game Mechanics*, where you can see the first designs of the spells that were going to be implemented and where the ones explained in section *4.1.1.2. MVP Development*.

6.3.2.1. Project setup

6.3.2.1.1. Team name

Yellow Games

6.3.2.1.2. Team composition

Herman - programmer

Gok - programmer / designer / sound designer

Joan - programmer / artist

Cesar - artist

Mike - researcher / designer

Evgen - designer / publisher

Product owner - Mike

Scrum master - Herman

Genres: Sports, Competitive, Fantasy

6.3.2.2. Game overview

6.3.2.2.1. Working name

Project Rifiers

6.3.2.2.2. Description

It is a fantasy sport video game where 2 players compete with each other in order to chase a magical creature in portals by utilizing level manipulation spells.

Wrapped as performing a magical ritual with spectators.

6.3.2.2.3. Target audience

Teens and young adults

6.3.2.2.4. Rating

PEGI: 3; Online; In-Game Purchases

ESRB: E

6.3.2.2.5. Competitive products

Rocket League

Roller Champions (alpha)

6.3.2.3. The game world

6.3.2.3.1. Description

Playground is a sports arena with a complex structure, not just a flat play field.

It is placed on the mountain ledge. There are some ruins around, spectator seats on one side, view on the mountains on the other side.

There are some plants penetrating ruins and growing around.

Could create extra magic feeling with some anomalies around, fires in the sky, etc.

6.3.2.3.2. Elements

Arena:

- Mountain edge terrain
- Sports field layout
- Colored portals on each side
- Interactive objects
- An altar in the middle

Spectator seats

Plants

Game attributes:

- Gigantic flasks for each team to store scored balls;
- Some hourglass or progress bar-like magical device to reflect match time;

Background terrain

Skybox

6.3.2.4. Game Rules

6.3.2.4.1. Objective

Score more points than your opponent in the allotted time.

6.3.2.4.2. Scoring

To score a point you must chase a “ball” through a portal on one side of the field and it will be captured in the flask on the side of the field.

6.3.2.4.3. Players

Only 1 player on each side. Players are mages with simple spells which allow them to interact with a “ball” and manipulate specific parts of the environment.

Players can not do harm to each other and are protected by magic.

6.3.2.4.4. Equipment

Players have sturdy short robes with hoods, comfortable to run in. No extra protective gear is necessary as they do not have a high risk of physical injuries.

Players wield staff to perform magic (?).

6.3.2.4.5. The ball

Game is played with a living (or magically sentinel) ball with simple AI. It is always in motion, flying around the field. When it is not disturbed by players, it tends to stay around the center, drawn by magic force.

6.3.2.4.6. Match

Game lasts 5(?) minutes, has no breaks, no penalties, no timeouts.

6.3.2.4.7. Field

Field has multiple objects/areas which can be manipulated by magic in order to help the player chase a “ball” or interfere with the opponent.

6.3.2.5. Game mechanics

6.3.2.5.1. Player's abilities

1. Push – pushing power, cone shape in front of player. The closer to epicentre the target is, the stronger push it takes.

Ball, when pushed, will fly in direction of the push. Unexpected, huh?

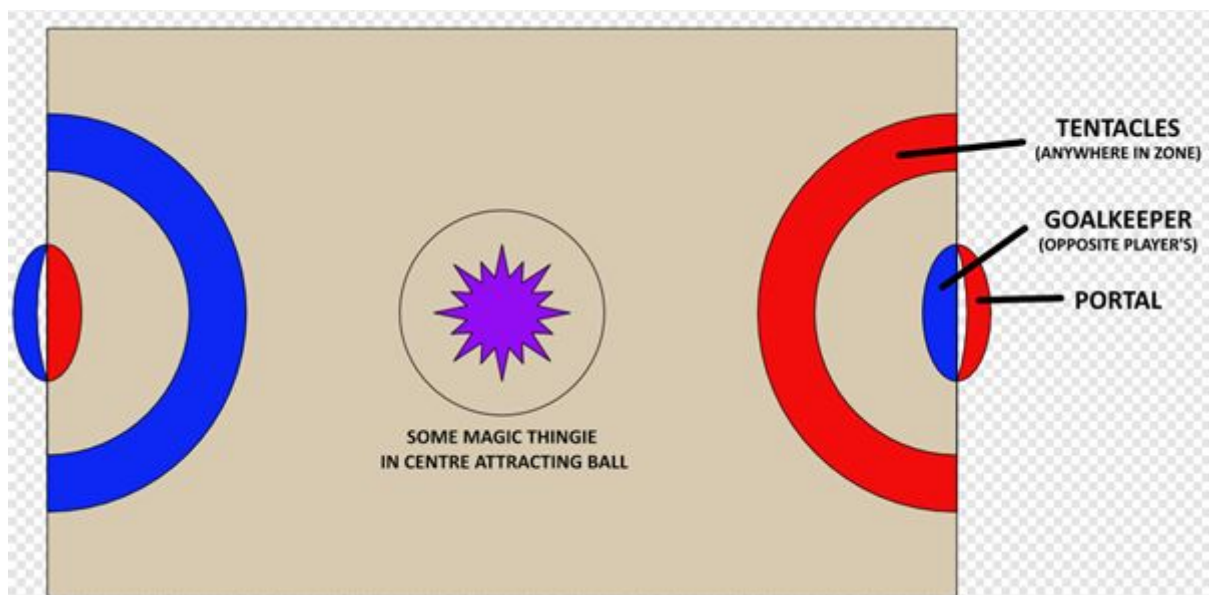
Short but existing cooldown (0.5 sec?), allowing to run and push ball like dribbling.

2. Areal interactions – described separately
3. Shoot – shoots small projectile (arcane bolt) which will fly to the ball (automatically) and explode near it, “scaring” it and changing its direction slightly.
4. Sprint/Dash/Blink – ability to move faster or shift forward to increase mobility and pace of a game.

6.3.2.5.2. Areal interactions

1. Temple wall – rise a brick temple ruin wall.
Player can't run through it. Player can break it with Push.
Ball tries to fly around maintaining direction but loses some time to do so. If collides on high speed – breaks the wall but loses speed greatly.
After destruction, needs T time to recharge.
2. Roots/vines – makes plants alive for a short time. If in that time ball is within the reach – it will catch it and hold in place for a moment.
3. Water geyser – explodes with water, pushing ball N meters back.
4. Ice something – explodes with cold wave, slowing down ball greatly.
5. Lava puddle – explodes with fire pillar, increasing ball's speed.
6. Metal rod – summons lightning on the metal rod. If ball is nearby, it gets hit by lightning and *greatly* increases speed for a short while.
7. Air pressure – sparkling area in the air, when activated creates the wind which blows in direction of caster's portal, making ball change direction there.
8. A lamp (jk, some magic totem thing) – for a few seconds, you control the ball with your camera, like psychokinesis.

9. Tentacle (puddle, void, swamp, whatever from where it can be summoned) – summons a single tentacle that swiftly catches the ball and throws it directly to the gate.
Channel spell, but also has cooldown.
10. Goalkeeper (release the Kraken) – summons a few magic tentacles in front of the goal's portal. Acts like a goalkeeper in any sports game – tries to catch the ball and throw it back, is quite effective but sometime fails.
No cooldown, but it is a “channel spell” – only works as long as you keep pressing the button. You can run around to prepare for your next move but can't cast other spells.



Will need a simple way to show when these things are ready to use – shining/rumbling/etc.

6.3.2.6. Characters

Players are mages with simple spells which allow them to interact with a “ball” and manipulate specific parts of the environment.

Players can not do harm to each other and are protected by magic.

Other characters: spectators, lower quality models with small differences to fill the crowd. If we have time.

6.3.2.7. User Interface

Main menu

Settings (can change controls)

In-game: spells icons, cooldowns, scores, timer, etc

6.3.2.8. Sound

Background music

Spells sounds

Object sounds

Running, jumping sounds

Score and other “UI” sound

6.3.2.9. Technical

6.3.2.9.1. Platform

Windows 10

6.3.2.9.2. Engine

Unity 3D