

Unroll

Game Development Methodology

Final Report

Filipe Sousa 90714

João Galveias 90737

Pedro Vilela 90762

Instituto Superior Técnico

1. Playtesting report

1.1. Playtest procedure

We collected the playtest results through Unity Analytics events, to gather information on the players' performance, and a Google Forms questionnaire, to better understand our players' profile and to get some suggestions and opinions on what we did right and what we should improve.

The data we collected from the events was:

- Number of collectibles collected from each type (copper, silver, gold)
- Time the player takes to complete each level
- Number of deaths per level (in levels with water or lava)
- Number of times player presses "E" (ignoring cases where the player is near a helmet or door)
- Number of times player successfully grabs the ball

1.2. Gameplay questionnaire results

From the Google Forms questionnaire, we gathered information about the players' age, gender, favourite game genre and time spent per week playing video games. We also asked the players to give a rating expressing their general opinion about the game, and asked them some open questions.

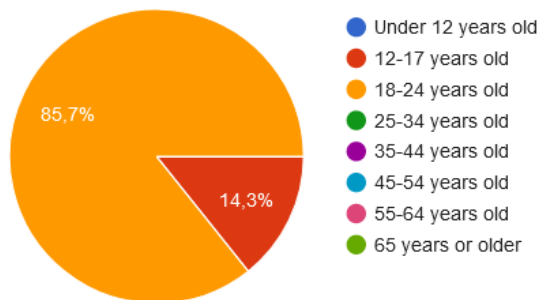


Fig. 1 - Age range distribution

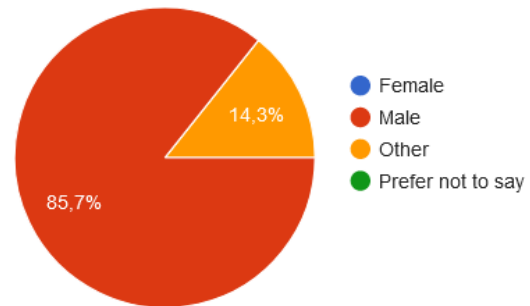


Fig. 2 - Gender distribution

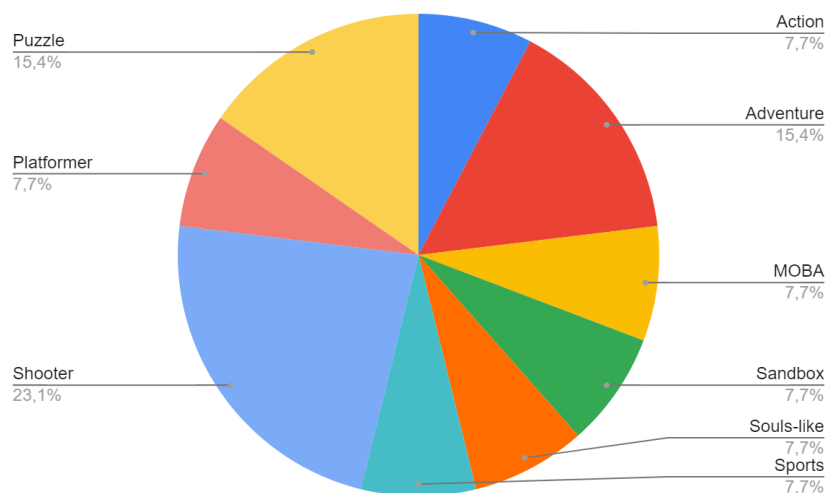


Fig. 3 - Favourite game genre distribution

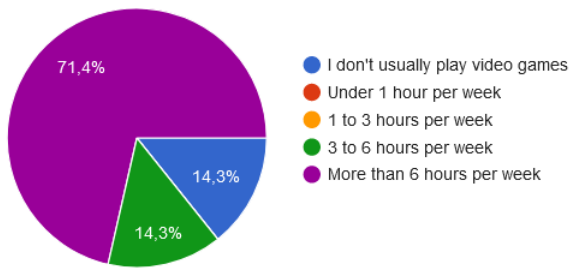


Fig. 4 - Weekly play time distribution

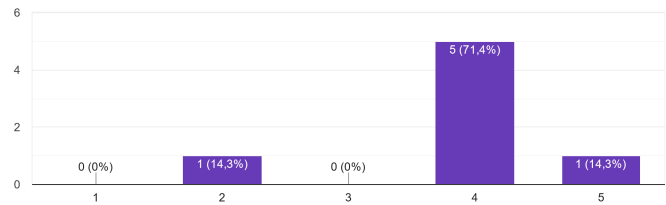


Fig. 5 - Rating distribution

The open questions we asked, and the answers we got are resumed next.

- **What is your opinion on the levels' difficulty progression?**

The majority's opinion was that the last level is much more difficult than the rest. This was expected, as the last level was purposely made to be more complex so we could see what a difficult level would look like. In the final version of the game, the player would have the necessary knowledge to understand what to do and would not feel a sudden jump in difficulty.

- **Which features did you enjoy the most and why?**

Here, the answers included the elemental powers of the ball, specifically the light power up, the "throwing ball" mechanic, the maze, adding that it was at a good difficulty for an introductory level, and the puzzles themselves.

- **Which features did you dislike the most and why?**

The answers for this section were a bit more diverse; one person commented on a recurring problem, which is the speed of the boy, who is still too slow. Another person also mentioned a problem of which we were aware, which was the restart button being an instant restart and any misclick would reset the level. Two people talked about the difficulty of the last level in comparison with the others. One person mentioned the fact that it wasn't very "intuitive about where to go or what to do", this was due to us not having to implement the contextual tooltips, which would clear up most of the confusion. One other person mentioned a problem we have had for a while, the "goofy physics on ramps", that we intend to fix. One person talked about various features, such as: "there should be a map of the mazes outside", "the lighting up of the street lamps was confusing", "the teleporters could have added much more to the game" (this was our intention, but this was a last-minute feature implementation so it ended up not having a major role in this version of the game) and "the interaction between the lava ball and ice", where the latter should melt.

- **Do you have any suggestions you consider would improve the game?**

In this question, most players just gave suggestions for the problems they mentioned in the previous question, such as: a way for the boy to move faster, fixing the ramps, having a more gradual pace of introducing features, having a map for the maze outside and tooltips to help explain the mechanics the first time the player sees them. For the other suggestions, they include: having more distinct levels, to give a sense of progressing through the narrative, having more ways to close text boxes, as the mouse cursor sometimes wouldn't become enabled, add music and sound effects (something we were planning on leaving for a later stage of development) and being able to pick up the ball with mouse controls.

1.3. Game instrumentation results

We were sending Unity Analytics events once every minute, so we were able to estimate playtime by counting the number of events triggered.

We found out each player spends around **28.65 min playing per day** (not including days with no playtime). This value will be similar to the **average session playtime**, as most players only played the game once. This is about half of what we had estimated (1 hour), but we must take into account that this is not the full game, and many players didn't play for more time because they beat the full game. Also, the fact that the difficulty curve was too steep between the 4th and 5th levels, made some players get stuck and give up, which in the full game would probably not happen, as there would be more easy and intermediate levels for the player to get used to the mechanics.

The total daily playtime was higher in the days after MOJO, as expected. Then, after June 3rd, it was always below 40 minutes, and after June 13th, we had no more plays until now (June 22nd). We think one of our main problems was that we didn't advertise our game apart from during the MOJO event. We also tried to create a release announcement in itch.io and add the gameplay trailer to the game's webpage, but still with no success in increasing the number of players.

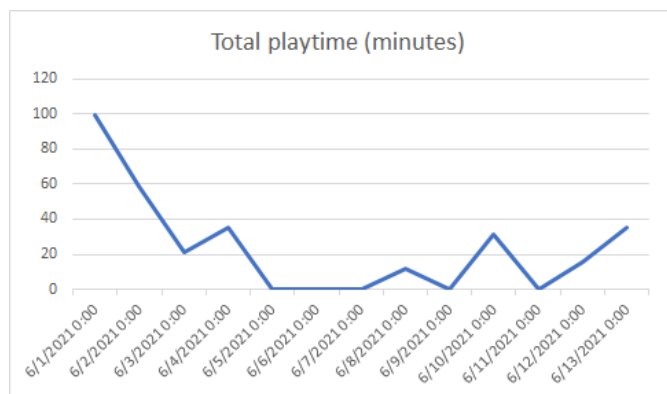


Fig. 6

We also found that around **59% of “E” presses** when close to the ball, resulted in the boy grabbing the ball. This may be due to the players not being used to the grab distance and preemptively pressing “E” when approaching the ball. Because we were not able to extract the raw data, we couldn't see if this success rate would increase with the player progressing in the game.

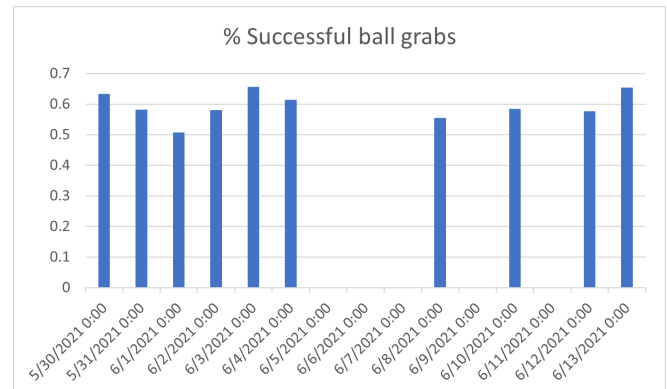


Fig. 7

Regarding the **collectibles**, we found out that the silver and copper collectibles seem to be collected with roughly the same frequency, and the gold collectibles are the most difficult to obtain. From May 31st (the day of MOJO) until today (June 22nd), **11 gold, 20 silver** and **21 copper** collectibles were collected. This yields a fraction of 1 gold versus 2 silver/copper collectibles, which makes sense, because the gold collectible in the intermediate level was indeed the most difficult one to get, and so the remaining 5 collectibles (1 gold, 2 silver

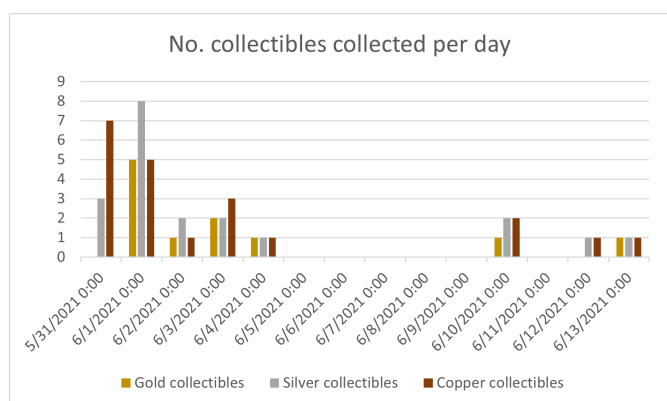


Fig. 8

and 2 copper) were obtained with higher frequency.

For “deaths” and “time to complete level”, we were not able to reach any meaningful conclusions due to the problem of Unity Analytics not allowing us to export raw data unless we have a Pro account, and so we can only see the number of deaths per day instead of per level (and the same for “time to complete level”).

1.4. Answers to our questions

We had several questions we intended to get the answers for using the data from playtesting.

One of them was if the value of each collectible (represented by its metal - copper, silver or gold) matched the difficulty of obtaining it. As we discussed before, this was not always the case, and it is something we need to tune.

Another one was if the difficulty curve was appropriate, which is also not the case, due to the difficulty spike between the intermediate and hard levels.

We also wanted to know the player profile of players who enjoyed our game the most. We discovered that all players whose favourite genre was “shooting” gave Unroll a rating of 4 out of 5, and the only player whose favourite genre was “platformer” gave our game a rating of 5 out of 5. Our target audience was puzzle fans, and from the 2 players whose favourite genre was “puzzle”, one of them voted 2 out of 5 and the other 4 out of 5. Unfortunately, we think there is not enough data for us to reach any meaningful conclusions, but it makes sense for us that the player who enjoys platformer games also enjoyed our game. For the shooting fans, it also makes some sense, as the “hard push” action can be familiar to them, by being similar to firing a weapon.

Another question we had was which features were good as they were now, and which ones needed improvement, but this was answered by the open questions we discussed before.

For the remaining questions, which included, for instance “How long do players take on average to beat the full game?” and “How often do players die in each particular level?”, we couldn’t answer due to not being able to export raw data from Unity Analytics. There were also some answers that we couldn’t answer because we needed a heatmap, which we ended up not implementing, after discussing with the lab teacher its relevance to our project.

1.5. Implications of the results in our game’s design

The level progression issues will be easily solved with the addition of all the levels we intend to have in the game. Regarding the most enjoyed features, we will try to add more puzzles that require the “throwing ball” mechanic and the light power up. For the most disliked features and frequent issues, we will increase the speed of the boy, fix the ball’s physics, add contextual tooltips and fix the restart button issue by adding a confirmation box. About the elemental reaction between the lava ball and the ice, we think it would be a good interaction to add but we feel that it would add some confusion because we also have white blocks in some levels which aren’t ice and they would mislead the players into thinking they could break them, with the lava ball. We also want to add some background music and sound effects to make the gameplay more pleasant.

Regarding the game instrumentation results, the average playtime won’t really have an impact on the game design, as the results are not unexpected, as we discussed before. For the fraction of successful ball grabs, the value seems low, but as we couldn’t extract the raw data, it could be the case that the players improve their technique with time, and in that case, we would not need to make any changes. For the collectibles, it could be a good idea to make silver collectibles harder to obtain (so they are more challenging to obtain than the copper ones) and to also increase the difficulty of collecting the gold diamond in the hard level.

2. Development and exploitation plan

In this section, we describe our **plan** to continue the development of Unroll, along with our **marketing strategy**, and discuss our expected **costs** and **revenues** over a period of **1 year**.

2.1. Planning and scheduling

The first step would be to correct the current bugs, which would take us about one week. Then we would fine-tune the game mechanics and add some suggested features such as allowing the boy to run. We estimate this would take about 5 days.

After that we would keep adding more levels, until we reach the total 30-40 we had planned. In parallel, we would be working on the story along with the game writer. We estimate that these tasks, in parallel, would take about 5 months.

We would conduct two playtesting sessions (apart from the playtesting itch.io players do on their own) - one of them around the middle of the development schedule and the other near the end. The first one would be both to find bugs and to tune the gameplay experience while the second one would be mainly to allow us to find “hidden” bugs that rarely occur, right before the launch date.

About 1 month and a half before the main release, we would start the Early Access phase, which would allow players to have an experience of the game more similar to the final product, at a lower cost and before its release (but possibly with some bugs, and missing features). Finally, on December 1st, the game would be released on Steam and Epic Games.

Unroll - Project Schedule			DURATION (days)
START DATE	END DATE	DESCRIPTION	
22-Jun-21	29-Jun-21	Fix current bugs	7
29-Jun-21	4-Jul-21	Add suggested mechanics	5
30-Jun-21	15-Aug-21	Crowdfunding	46
4-Jul-21	1-Dec-21	Add remaining levels	150
4-Jul-21	1-Dec-21	Story development	150
15-Sep-21	16-Sep-21	Playtesting session	1
20-Nov-21	21-Nov-21	Playtesting session	1
15-Oct-21	1-Dec-21	Early Access	47
1-Dec-21	2-Dec-21	Full game release	1

Fig. 9 - Project Schedule

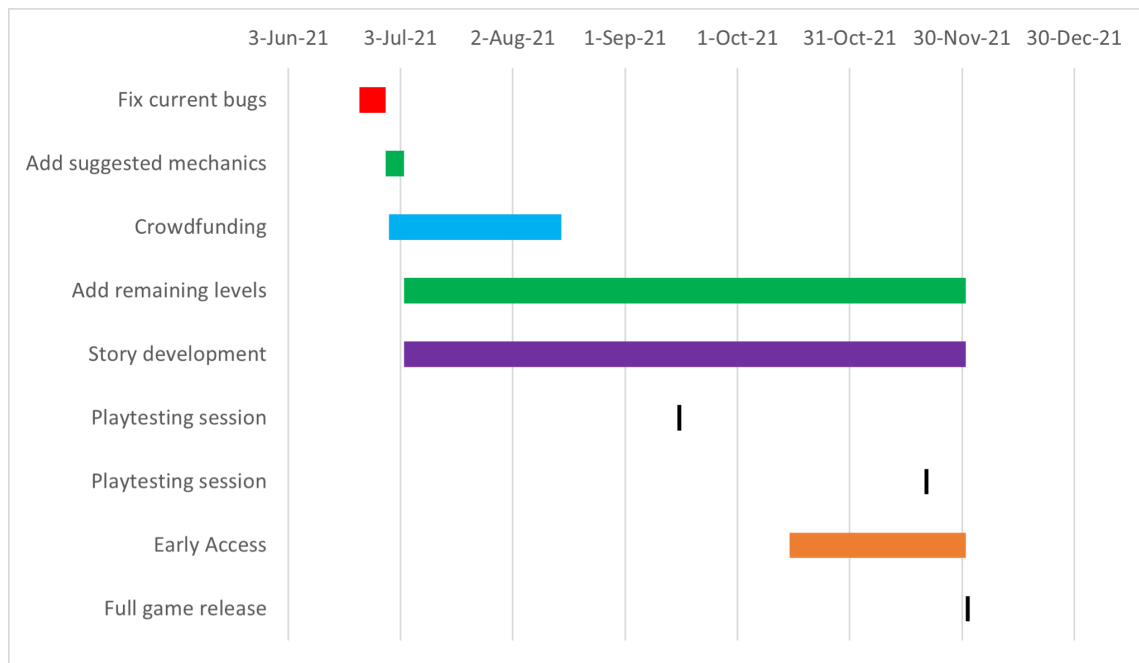


Fig. 10 - Schedule chart

2.2. Distribution channels

For the distribution, we will keep our game available on **itch.io**, and publish it on **Epic Games Store** and **Steam Store**.

2.3. Marketing strategy

To handle marketing, we are going to hire an external marketing agency. The activities we will request the agency to do include branding, creating a video trailer, social media management, public relations (contacting other people or institutions that might be interested in showcasing our game) and merchandising.

For this last one, we decided to create some products to help spread the brand and the game by using the people who buy to advertise the game for us, as well as giving them cool items they could use in their day-to-day lives.



Fig. 11 - Merchandise examples

2.4. Market Analysis

After researching statistics of games of similar genre and dimension (in terms of investment), we found out that on Steam, “the average indie game sells 1.500 copies” ^[1] during its first year (data from 2019). We decided to do a worst case scenario prediction, and so, we predicted to sell **1.200 copies in the first year on Steam**. We also discovered that the number of daily active users on Steam store during the 2020 is about **twice** the number of daily active users on Epic Games store (62.6 million vs 31.3 million) ^[2], so we predicted to sell half as much games on **Epic Games store**, i.e. **600 copies**. We couldn't find much reliable information about itch.io's sales, so we estimated a low value (**200 copies**). We chose a price of **20 € per game license**.

2.5. Costs

For the costs, we need to take into account the salaries for the developers (us), the game writer and an artist which we would hire to create some specific graphical elements, such as cutscenes. We also need to consider the game assets we would like to buy (mainly in the Unity Asset Store). Regarding marketing, we are taking into consideration the aforementioned activities (section 2.3).

We also considered the cost of the hardware equipment we would need to develop the game, and of its maintenance. For the software, we wanted to buy a Unity Pro license to be able to export raw data from Unity Analytics (to have a better insight into the data) and to increase the maximum frequency of Analytics event calls we can make.

Finally, we considered the fraction of the revenue taken by each of the distribution platforms (on itch.io, we considered the default fraction of 10%).

So, we estimate our expected costs over **1 year** would be the following:

Items and activities		Expected Cost	
Salaries	3 developers - 6 months of work, with meals included	1.200€ per developer per month × 3 × 6 = 21.600 €	25.600 €
	Game writer - 3 months of work, with meals included	3.000 €	
	Artist - 1 month of work	1.000 €	
Game assets	Character models, level design assets (e.g., different city buildings and structures), music and sound effects	100 €	
Marketing	Branding	2.000 €	13.000 €
	Trailer	500 €	
	Social Media	3.500 €	
	Public Relations	2.000 €	
	Merchandise	5.000 €	
Equipment	Computers, peripherals and internet access	5.000 €	
	Maintenance of the computers	500 €	

Software	Unity Pro License	1.800 €	
Game distribution	Itch.io	2 € / unit × 200 units = 400 €	9.040 €
	Steam	6 € / unit × 1200 units = 7.200 €	
	Epic Games	2,40 € / unit × 600 units = 1.440 €	
Total		55.040 €	

2.6. Revenues

The main revenue will be the game sales which will be about 44.500 € summing up early access sales and full game sales. Then we expect to gain some revenue with in-game cosmetic items, which we estimate to earn 1.000 € from. Besides the game sales we also plan to double our investment in game and brand merchandise, gaining about 10.000 € in total. We plan on doing a crowdfunding campaign to raise up to 5.000 € to help in the game development and merchandise.

The expected revenue over **1 year** is summarized in the following table:

Items and activities		Expected Revenue	
Sales	Game licenses	20 € / unit × 2.000 units = 40.000 €	55.500 €
	Game licenses (early access)	15 € / unit × 300 units = 4.500 €	
	Cosmetic items	2 € / unit × 500 units = 1.000 €	
	Merchandise	10.000 €	
Crowdfunding		5.000 €	
Total		60.500 €	

2.7. Conclusion

According to our estimate, our net worth would be 60.500 € - 55.040 € = 5.460 €, not counting the salary each of us would receive, even if we don't count with the merchandise our net worth would be 50.500 € - 50.040 € = 460 € which would still be profit.

We think an outcome like this one would be a good result for our first experience in the video game market.

3. Postmortem report

In this section, we present an overview of all activities we performed since the beginning of the semester, and then, we reflect about the good and bad decisions we made during this time.

3.1. Overview of the course activities

Our first activity was the **concept pitch**, which we didn't feel very confident to do, since we had almost no time to think about our game idea and polish it a bit. However, we think we presented the main idea well. The next activity was **building low-fidelity prototypes**. During this activity, we had more time to brainstorm between us the main aspects of the game and also some particular features that could be tested in each prototype individually. We also tried to make our prototypes online-friendly so that it would be easy to test them remotely. The prototypes were really good to test the different features that ended up in the final game. However there was an aspect that we could have improved during the presentation of the prototypes, which was adding commentary to our prototypes' videos. Since they were really easy to use and test, getting feedback from the users was an easy task that went as planned. This ended the **first dev cycle**.

In the beginning of the **second dev cycle**, we did a **workshop** with our focus group. We did **zen voting** and **card sorting**. In general, it went well, because we got the opinion of the players about some of our ideas and also got some new ideas out of it. Then, we developed our **prototype**. There were some bugs that set us back a bit, but in the end we got it solved and the prototype itself presented our game idea really well. The **user feedback** was also great, because the players enjoyed playing it, but also pointed out aspects of the game that needed improvements. This was the last activity in the second dev cycle.

Now, all we had to do was to **complete the game** as much as we could. We focused on doing more levels, especially easier ones so the players could get used to some of the different puzzles that could be presented in the harder levels. We also added some assets to improve the aesthetics, but we couldn't add as many assets as we wanted because most of the good assets were paid. We added the game writer's short stories into the game. The development phase also didn't go as we planned because we were very occupied with other courses' projects. We wanted to add more levels, namely to introduce the concepts we've shown in the hard level, but at a slower pace, instead of all at once.

The **Mojo presentation** was something that we didn't practice much for, but when we did it all went well and smooth. In the first week, we got some players that came from Mojo, but after that, we didn't have that many players. We feel like we could have done a bit more advertising, but we weren't much comfortable about doing it. Nevertheless, we still had enough data to analyze the game's performance.

In the end, we feel proud of the game we developed and happy that most of the players also enjoyed playing it.

3.2. Bad decisions

- **Choosing to use Unity Analytics (free), as it does not allow to export raw data**
This was problematic, because, in this way, we can only know sums or averages of daily results, instead of each individual result, and so we cannot match each individual result with the questionnaire answers (using the random id we created), because we don't know which events were triggered by each player's id.
- **Giving too much freedom to the game writing student and a loose deadline for him to produce the content**

The game writing student ended up handing us the content he produced at a very late date, which forced us to use that text regardless of our opinion about its quality and adequacy to be used in our game.

- **Having committed to use Unity's physics engine for the ball's movement at an early stage in the development**

This forced us to find a way to keep the ball close to the player (when he was holding it) using Unity's physics. We couldn't just "warp" the ball to the front of the player in every iteration, because this would let the ball pass through walls. But maybe we could have used this approach if we didn't use Unity's physics for the ball, and did the collision detection manually (we are not sure how time consuming that would have been, though).

- **Not having a list of questions to ask the people who tested the low-fidelity prototypes**
Some people didn't know what kind of information we wanted to collect, and so, didn't know what to comment about, resulting in poor quality of the collected data.
- **Lack of advertisement**
We could have advertised our game a bit more to have more players to be included in our game analytics.

3.3. Good decisions

- **Making low-fidelity prototypes that were really easy to use in online testing**
Due to the pandemic, two of us decided to do low-fidelity prototypes adequate to be used in online testing, which allowed us to perform the testing in a different physical location from the testers, via video call. We also concluded that this could be a good idea even without the pandemic, as it widens the range of people who can test the prototype.
- **Using the 3 different low-fidelity prototypes to test 3 different features, which eventually were all included in the game**
This approach made each individual prototype different and useful, as different features were being tested.
- **Choosing a game with relatively easy features to code and easy level design**
Due to our previous experience from the "Game Design" course, we knew which ideas would be more difficult to implement than others. Because of this, we tried to choose a game genre and mechanics that would be easy to implement and to add functionality. This really was a good idea given the fact that in this course we develop the game while still having other course's projects and classes rather than during the exam preparation weeks.
- **Using Unity, which is a software that we already had some previous experience with**
Although having learned to use a novel engine could have been useful for our future, it would have been time consuming, so it probably wouldn't have been the best option given the limited time we have to develop the project. Furthermore, learning a new tool was not one of the main goals of this course.
- **Using free assets that could be reused in every scene**
The assets we used for our level design really improved the game's aesthetics and they made level designing faster and easier.

4. References

1. Valentine, R. (2019, September 12). *Mike Rose: Indie developers are pricing their Steam games too low*. GamesIndustry.Biz.
<https://www.gamesindustry.biz/articles/2019-09-10-mike-rose-indie-developers-are-pricing-their-steam-games-too-low>
2. Yin-Poole, W. (2021, January 28). *The Epic Games Store is getting a lot more popular*. Eurogamer.Net.
<https://www.eurogamer.net/articles/2021-01-28-the-epic-games-store-is-getting-a-lot-more-popular>