IZMIR UNIVERSITY OF ECONOMICS
FACULTY OF ENGINEERING
COMPUTER & SOFTWARE ENGINEERING

FENG 498 PROJECT REPORT

CANNON SMASH

&

FLEPTRIS

Authors:

Atahan Ekici

Egemen Ustaoğlu

Supervisor:

Prof. Dr. Hüseyin Akcan

CONTENTS

1) Abstract

We developed 2 games in our project; one is a 3D Hyper-Casual Runner game, which passes through obstacles of different colors by breaking the same color walls, and the other is a Hyper-Casual game that includes a man-throwing cannon to knock down dancing human towers, just like in old circuses.

Hyper casual games have become very popular in recent years and have spread all over the world, and our motivation is increasing day by day as we want to develop ourselves in this genre.

We completed a fast and dynamic development process on time with the cooperation of Happy Game Company, which was established by a team who graduated from the same university as us, and with the support of our supervisor Hüseyin Akcan.

At the time this document was written, our test results for our game "Fleptris" were not yet finalized, but our other game "Cannon Smash" was tested and we got the results.

We have 2 fields that we can trust, which are CPI and Retention values. The CPI for our “Cannon Smash” game is 10.42$ and the Retention value is 44.44%.

CPI is the average ad cost shown to a random IOS user in the United States to download our test game. These ads take place on Facebook family-affiliated platforms such as Facebook and Instagram. The lower the CPI, the more the game's potential payoff is seen. The CPI value, which was 10.42$ in our game, had to come under 1 to satisfy us.

The retention value, on the other hand, checks whether a person who downloaded a game re-enters the game the next day, and the fact that the game is a binding game played continuously increases the retention value. Based on these figures, our average retention was 44.44%. In a market where over 35% is considered good enough, it is pleasing that the retention value of our game is 44%.

2) Introduction

For this semester we agreed to release at least 2 Hyper-casual games for our graduation project. We will be working with Happy Game Company and Rollic Games throughout these projects and our instructor informed us about their educational videos and guides so we will be using these materials to enhance both our knowledge and products.

- Our first game is named “Cannon Smash” which is a 3D Hyper-Casual mobile destruction game where we shoot people from a cannon to destroy human towers and dancing giants.

- Our second game is named “Fleptris” which is a 3D Hyper-Casual mobile runner game where we try to control a falling ball by hitting blocks of the same color as our character.

2.1) Problem Statement

In fact, although it seems that all we have to do is to make games, it is not easy to do this in this competitive market. The Hyper-Casual market is very fast and competitive, and it takes rapid prototyping, rapid testing phases, critical fixes, flexible working process and preparedness to keep up with this competition.

It takes an additional effort to keep all this up to the standards of the Happy Game Company and Rollic Games we work with.

2.2) Why is The Project Worth Doing?

The biggest chance we got while carrying out this project is to test our game by Rollic games, a worldwide publisher, so we stepped into the professional sector during our school years and we have the opportunity to observe the sector from the inside. If the result from these tests is very successful, the possibility of a great financial gain should not be overlooked.

2.3) Related Works

After testing a hyper-casual game, we encounter 4 important values, they are CPC, CPI, Retention and Playtime.

If we explain the meanings of these values, we can start with CPC.

2.3.1) CPC

So, another value accompanying the CPC formula, what is CPI?

Cost per install (CPI) is a pricing model used in mobile user acquisition campaigns in which game or app advertisers pay each time a user installs their app from their ad. CPI is a very common pricing model, and it is specific for mobile apps and games only.

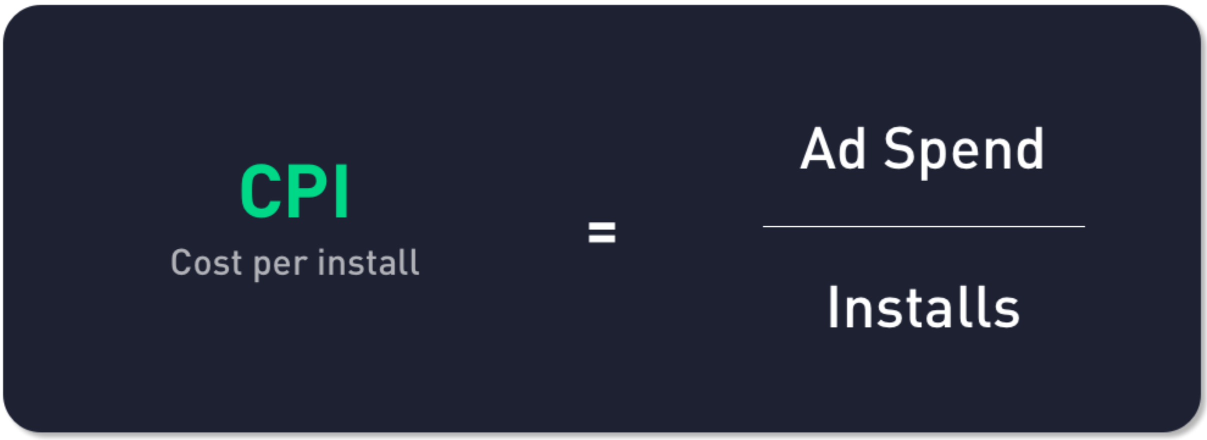


Figure 1

2.3.2) CPI

Cost per click (CPC) is also known as pay-per-click (PPC), CPC is a method that games use to bill based on the number of times a visitor clicks on an advertisement. CPC is often used when advertisers have a set daily budget. When the advertiser's budget is hit, the ad is removed from the rotation for the remainder of the billing period. [5]

For example, a game that has a CPC rate of $1 and provides 1,000 click-throughs would bill $1000 ($1 x 1000). The amount that an advertiser pays for a click is usually set either by a formula or through a bidding process. The formula used is often cost per impression (CPI) divided by percent click-through ratio (%CTR).

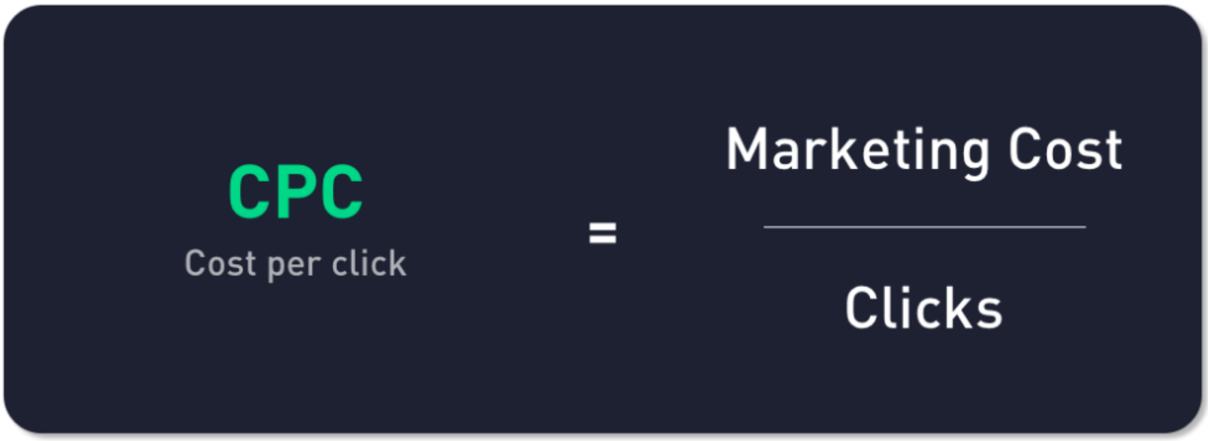


Figure 2

2.3.3) Retention & Playtime

If we leave the advertising balances aside and come to the gameplay values of the game, we will come across two words, Retention and Playtime. Playtime is the average time spent in the game for a day by the test group who started downloading and playing the game via ads, and this value is specified in seconds.

Playtime and retention figures are of the kind that support each other. It indicates how much the player likes the game, how much he wants to play, and whether he is willing to play the game again.

As for the retention value, retention of the game is a measure of percentage of the people that continues to for example, among 1000 people who downloaded the game and entered the game, 125 people who re-entered the game the next day indicate that the retention value is 12.5 %.

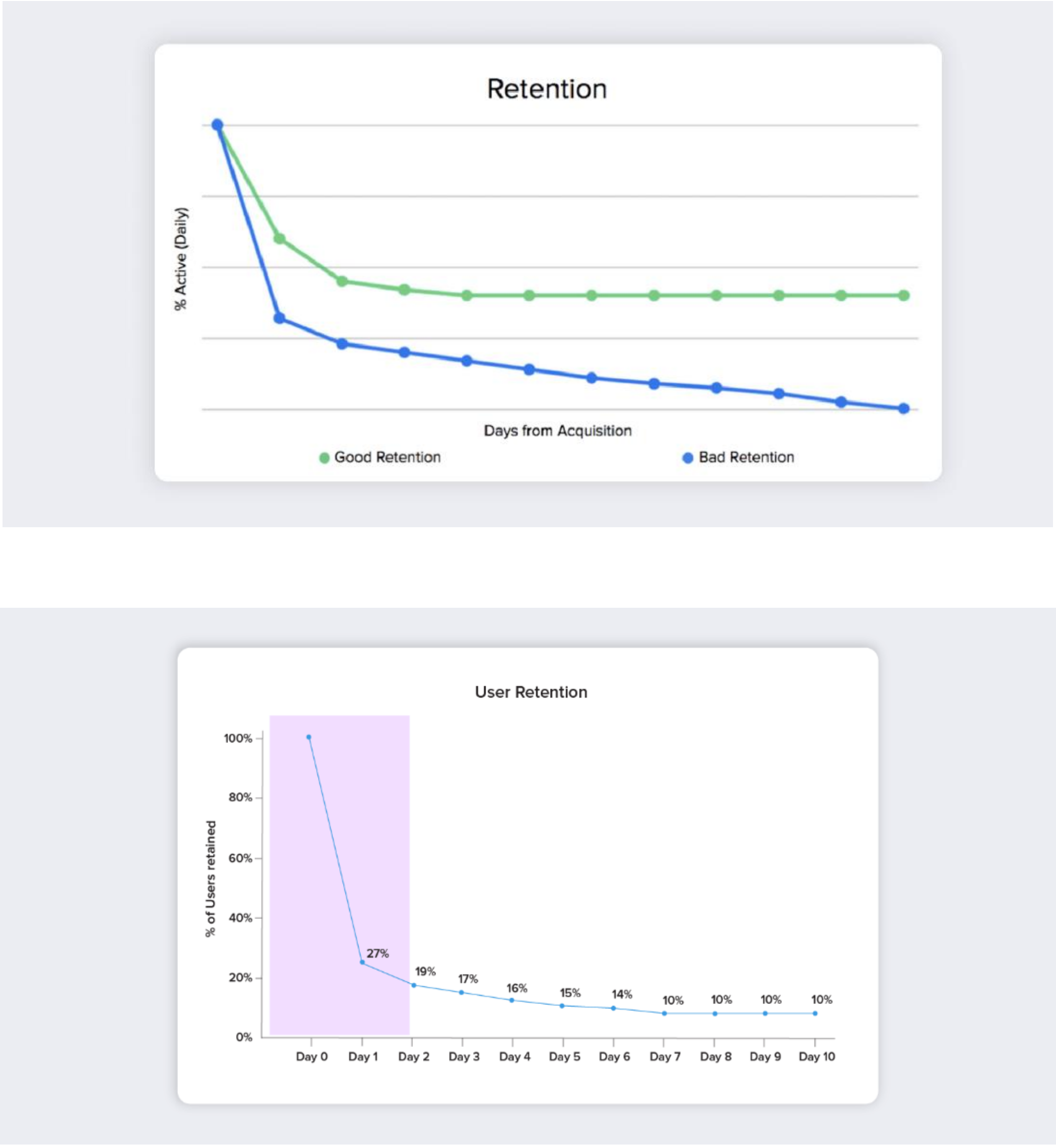


Figure 3

3) Literature Review

Since the very beginning of the AppStore and google play both published steadily mobile games. For the design pattern in the mobile games, the purpose is to extract visibility to the user.

Content and the research determine the strengths and the weakness of the absent literature. For mobile games, the background information is far beyond the general. Playing and understanding a simple pick up and play type of games, it is well suited for the device, like the everyday smartphone. Generally, the hand-driven graphics are be well understood. Players' requests are generally, the games should be simple but effective not only the mechanic but also the graphic part. The release update and the hot-fix patches help the innovate the game in the market. With the updates, the game sort of being promoted to the innovated to the player.

3.1) Demolish!

Inspiration for Cannon Smash

People like to destroy anything possible, especially children.

"Demolish!" A game released by the French Hyper-Casual giant "VooDoo" and downloaded tens of millions of times. There is only one logic in the game, to throw cannonballs at giant buildings with our catapult and destroy them.

After a while, the controls of this game, which became repetitive and boring, were not good enough in our opinion, so we developed our own game by adding a lot of things to the inspiration we got from this game. [1]

Figure 4

3.2) Flappy Bird

Inspiration for Fleptris

With a quote from the famous game review site IGN: “Despite being so needlessly cruel, Flappy Bird is also extraordinarily fair and exacting. You won’t find any randomly inescapable patterns, or run-ending collision detection issues here. Whatever the number is on the screen when your randomly colored bird’s flight inevitably ends, it’s a badge – an immutable quantification of your skill and focus. It is in this regard that Flappy Bird finds its one true success.”

In our "Fleptris" game, we tried to use the tap-tap mechanics of this game, which everyone has played at least once, as top-down instead of left-to-right. [2]

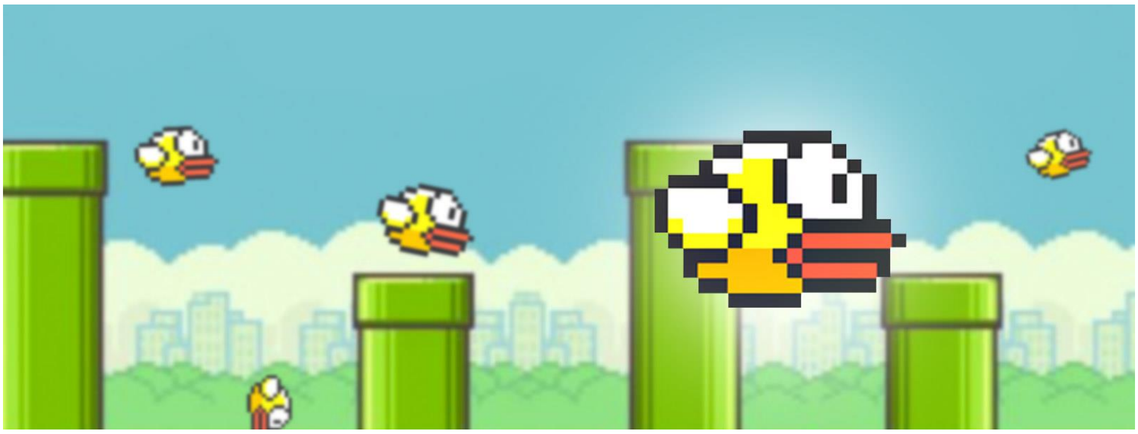


Figure 5

3.3) Tetris

Inspiration for Fleptris

We all know tetris, and it was perhaps the only game our parents played when we were our age, so we were inspired by both this familiar feeling and the color combinations in the current version of tetris. [4]

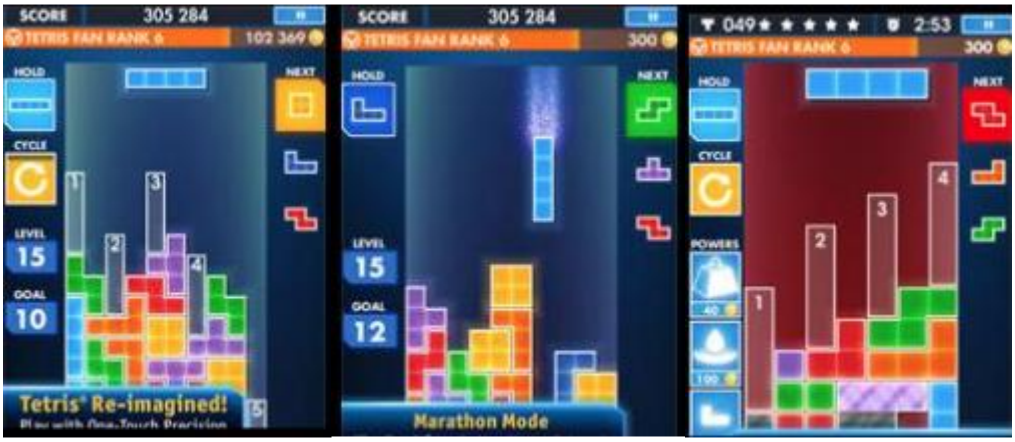


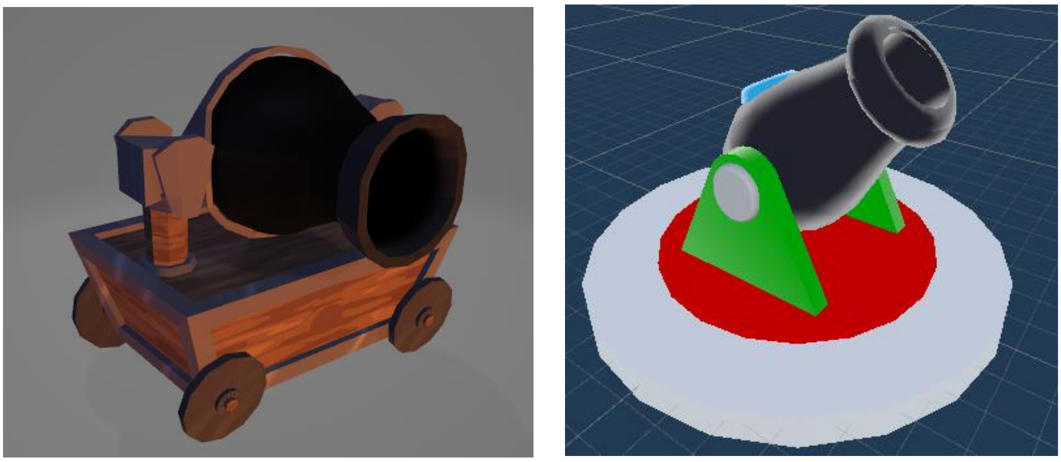
Figure 6

4) Methodology

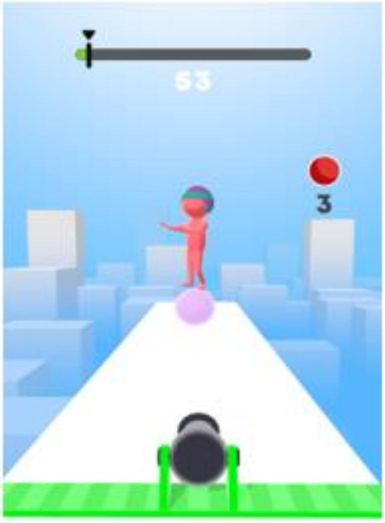
There are actually a few key factors when developing a Hyper-Casual mobile game.
These are modelling, design, theme, mechanics and dynamics of the game. Each of these key factors are indispensable for making a successful game, every design factor should appeal to the eye and an advertisement that comes across while surfing the internet should attract the attention of the user within the first 3 seconds. It is also very important that the game is addictive.

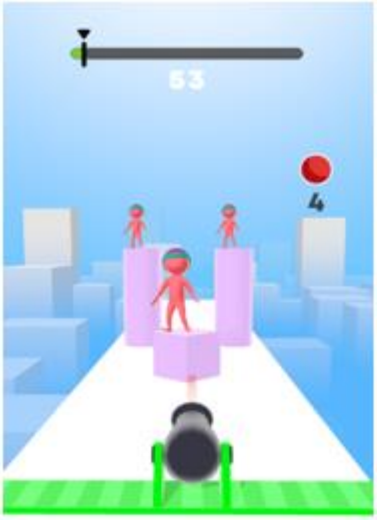
4.1.1) Modelling (Cannon Smash)

In Hyper-Casual games, the lines are soft and the models are superficial, but this does not mean that they can be prepared randomly, on the contrary, it is difficult to do something simple differently.



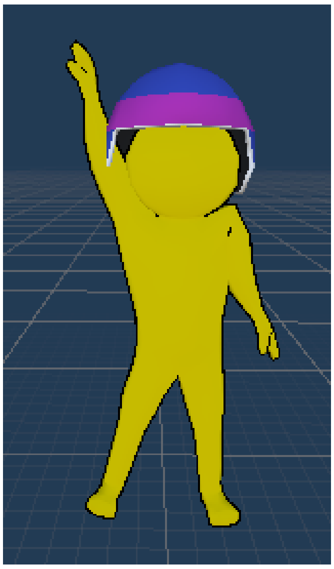
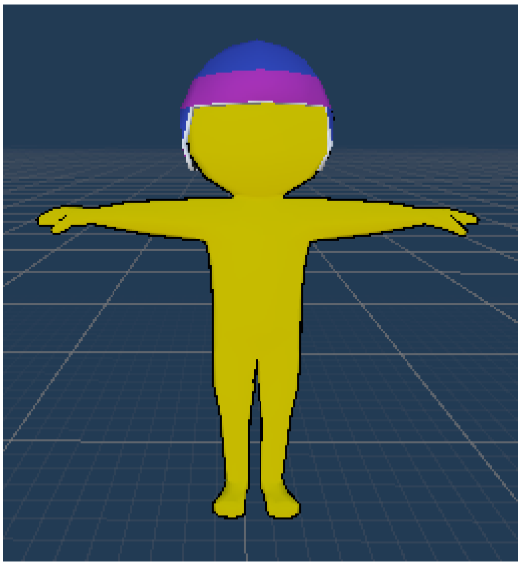
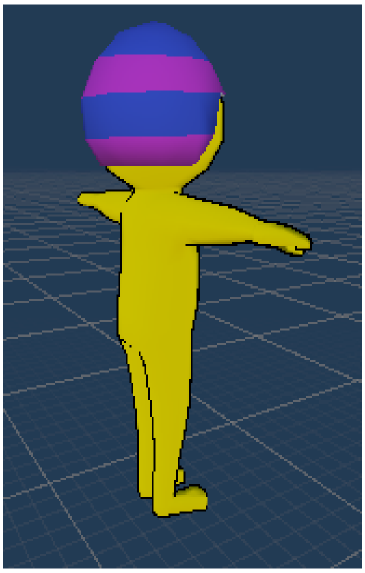
An ordinary cannonball that you can find in a thematic game and a "cannon smash" cannonball.



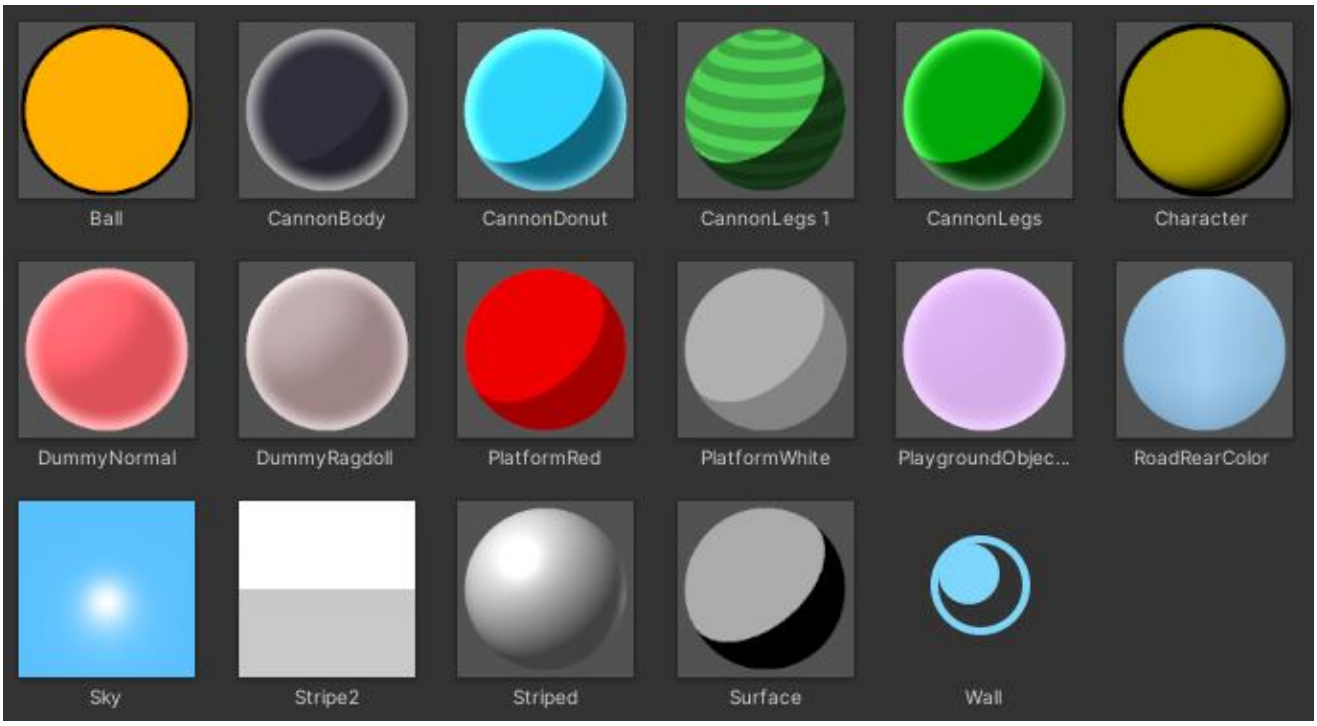


We decided to define 3 basic colors in the game, these are green, blue and pink tones.

We modeled our character the human cannonball helmet that many of us have seen in circus-themed movies. We've also made our character's color a phosphorescent yellow to help distinguish it from other people.

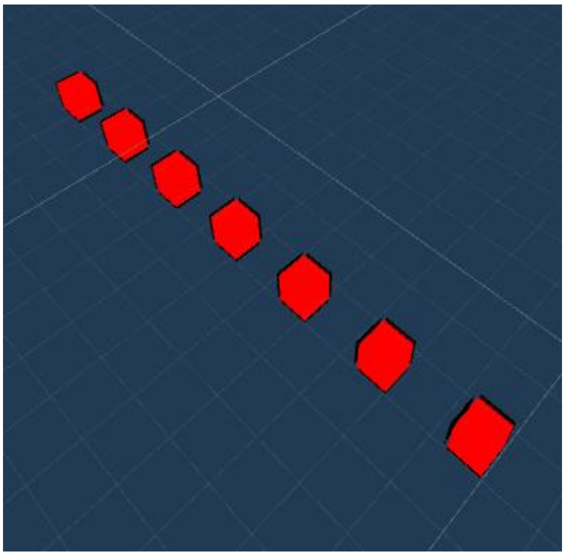




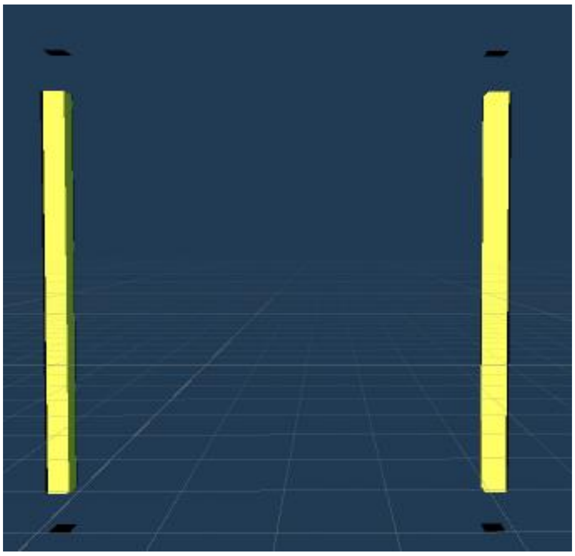
We carefully selected all the colors, shaders and materials in our game, we used MK Toon, Minimalist v2.4.3 and Epic Toon FX to reflect the Hyper-Casual look in the best possible way.

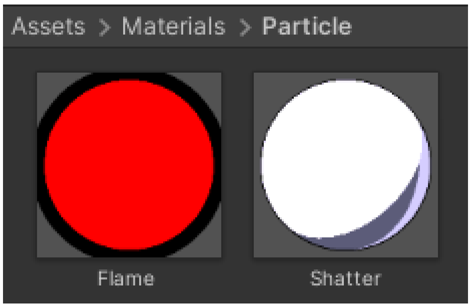


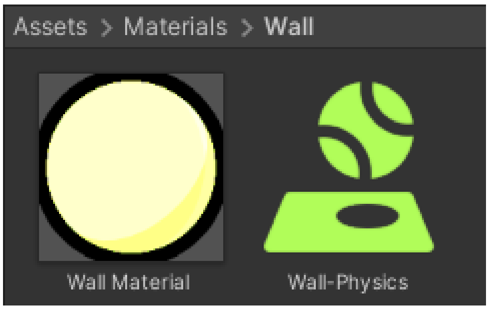
4.1.2) Modelling (Fleptris)

In "Fleptris" we tried to keep the models as simple as possible, one of the things we wanted to try in this game was to see what someone who is far from modeling can do with just code skills and unity's interface. With the custom shaders, special particle effects and post processing methods we used, we aimed to provide an eye-catching Hyper-Casual experience from just simple cubes and spheres.









4.2.1) Scripts (Cannon Smash)

For Scripting in our Games we have used C# as a programming language of choice and Visual Studio 2019 as the IDE.

They are:

- **CameraShake.cs:** This code allows to shake the camera briefly and with a certain intensity when our cannonball fires, thus giving a better sense of explosion.

- **CannonFire.cs:** It contains the smoke and flame effects created by our cannonball during fire, and also determines the starting position of the cannonball.

- **CannonMan.cs:** Determines the crashing and flying situations of our main character that we throw.

- **CannonRoot.cs:** Determines the movement of our cannonball on the x-axis

- **GameManager.cs:** It keeps the entire project intact, running the interface, top indicators, level randomness and order, fail states, confetti, and more at the right time, correctly.

- **GroupController.cs:** There is 1 group in each level, this code determines how many characters there are in these groups and how many shots we have to destroy these characters.

- **LevelNumberManager.cs:** It keeps and saves our previous, next and current level status, and pulls this data from the records when the game is closed and opened.

- **RagdollSwitcher.cs:** They use this code as a bridge to switch to realistic physics (Ragdoll) when characters hit each other or a surface.

- **UIManager.cs:** It controls the game's shot counter, level status bar, level number and buttons, activates or deactivates them at the right time.

- **PredictionManager.cs:** Before our cannonball fires, it creates a translucent projectile that shows exactly where the fired character will land if it fires.

- **Shooter.cs:** It is the code that creates the character to be thrown in the right position at the time of launch and throws it to the right target at the desired acceleration.

- **Rotation.cs:** It allows the tip of our cannonball to move up and down continuously.

- **TypeController.cs:** Controls the dancing, posing, and hitability of the target people to be shot.

- **InputManager.cs:** Provided by Happy Game Company for touchscreen operation and calibration for mobile devices.

- **NaughtyAttribute.cs:** An open source code stack that pulls the core principles for InputManager to work.

4.2.2) Scripts (Fleptris)

- **CameraController.cs:** It is a camera controller with acceleration in direct proportion to the character as the main character falls down

- **CameraShake.cs:** This code allows to shake the camera briefly and with a certain intensity when our cannonball fires, thus giving a better sense of explosion.

- **Cube.cs:** Controls regular and synchronous color changes of long rectangles on the border.

- **GameManager.cs:** The main control code stack that spawns the main character at the desired coordinate at the beginning of the game, and also turns all the buttons of the game on and off.

- **LevelManager.cs**: When a level is created, it ensures that it can be produced regularly and without any problems, with endless continuity, it creates new walls as the character goes down the side walls, and destroys the walls behind the screen.

- **PlayerController.cs:** It is responsible for the color and particle changes of the character.

- **MovementController.cs:** Allows the character to be controlled on the x-axis.

- **OutlineController:** This code causes flame effects to appear on our character's surface when rubbed against the wall.

- **ParticleController:** The stack of code responsible for breaking down our character and obstacles upon impact.

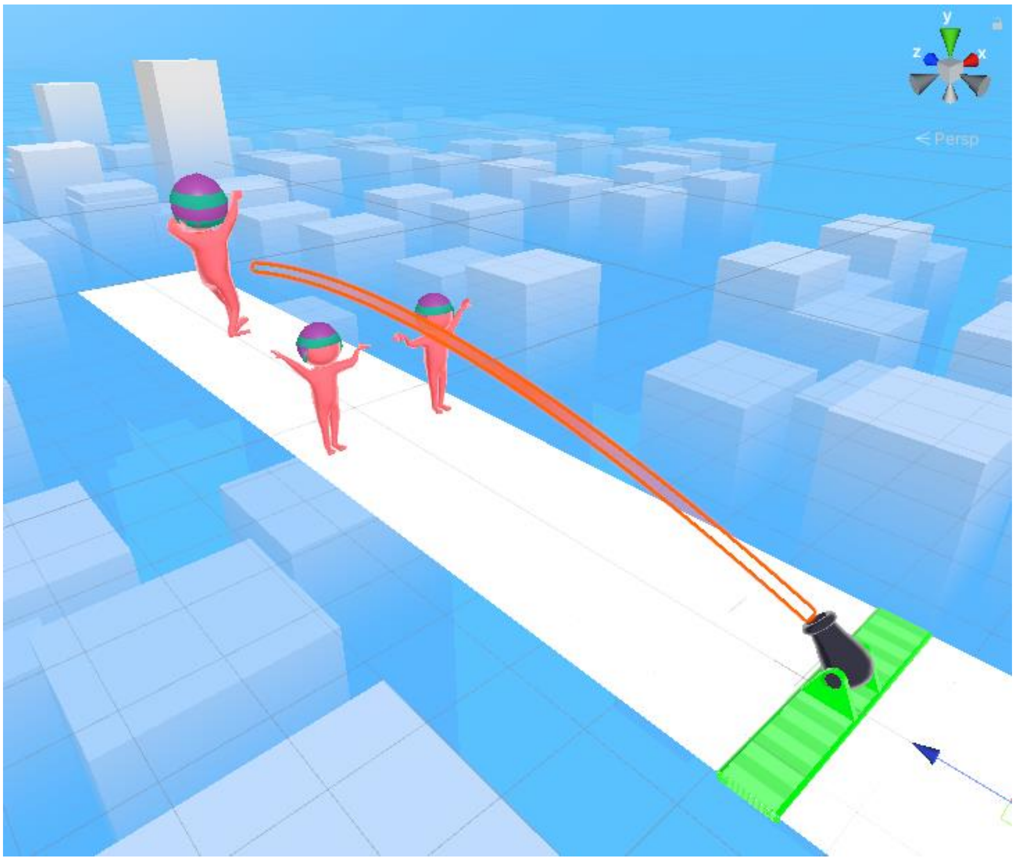
- **ScoreManager:** It is responsible for increasing the score as you destroy the walls and recording this score.

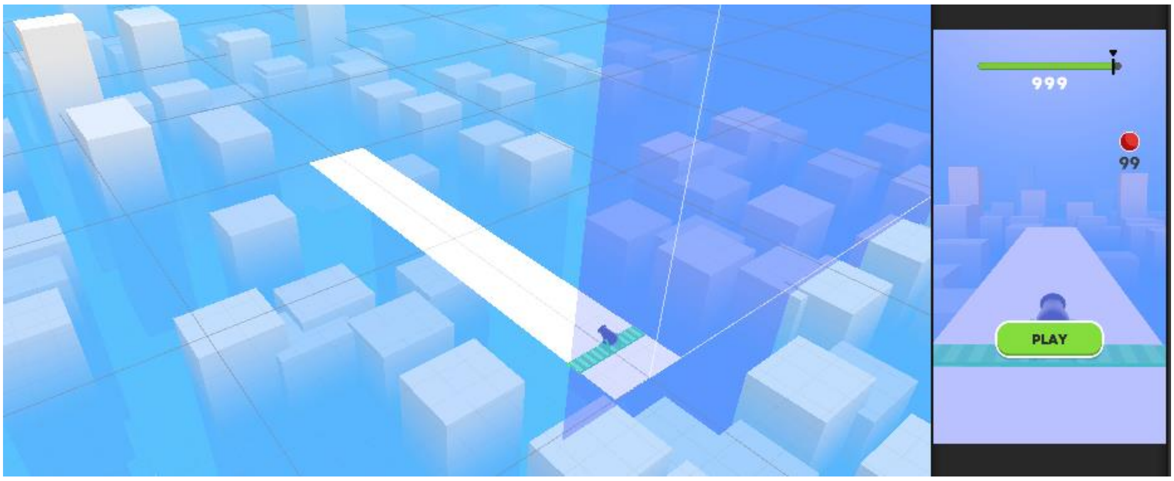
- **WallColor:** It is the code that makes the walls change color as the game progresses and the score increases.

4.3.1) Environment (Cannon Smash)

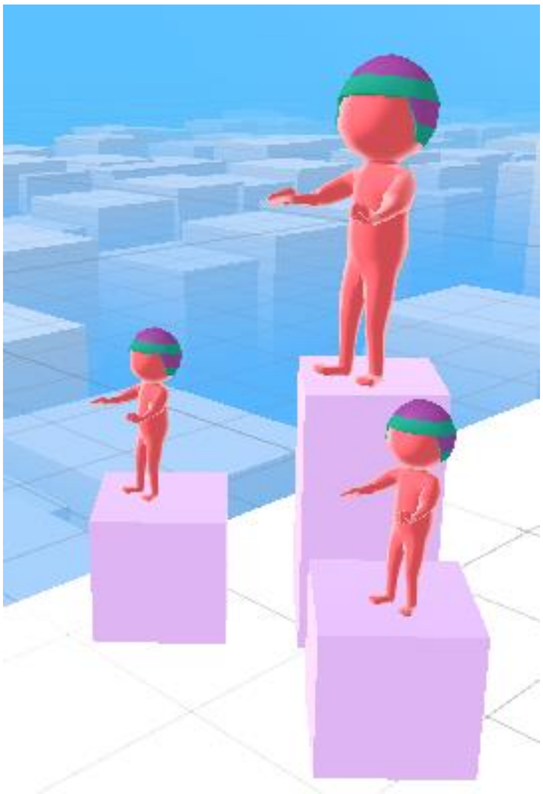
Coming to the environment of the game, we kept the ground very simple to avoid distractions, it was important for us to contrast with the characters and cannonball.

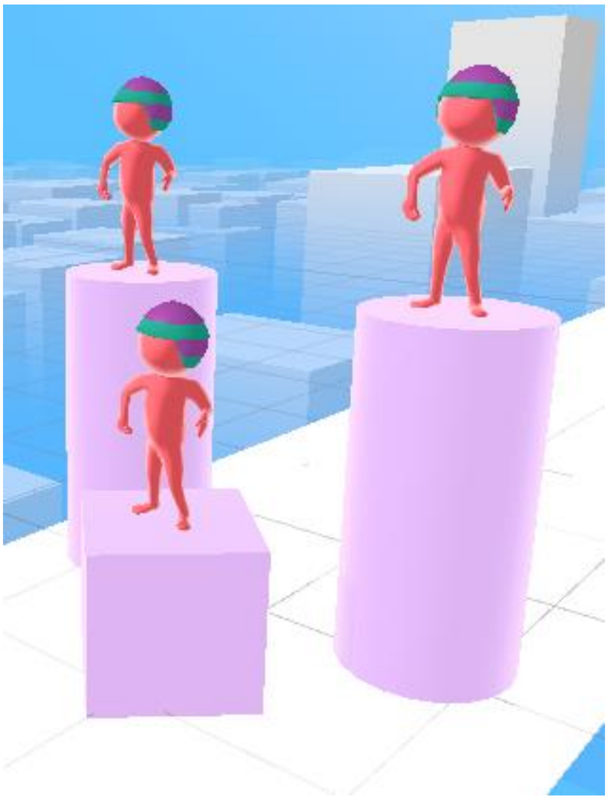
Outside of this surface, we have given the feeling of infinity with rectangular prisms accompanied by blue fog, so the user will not feel constrained.



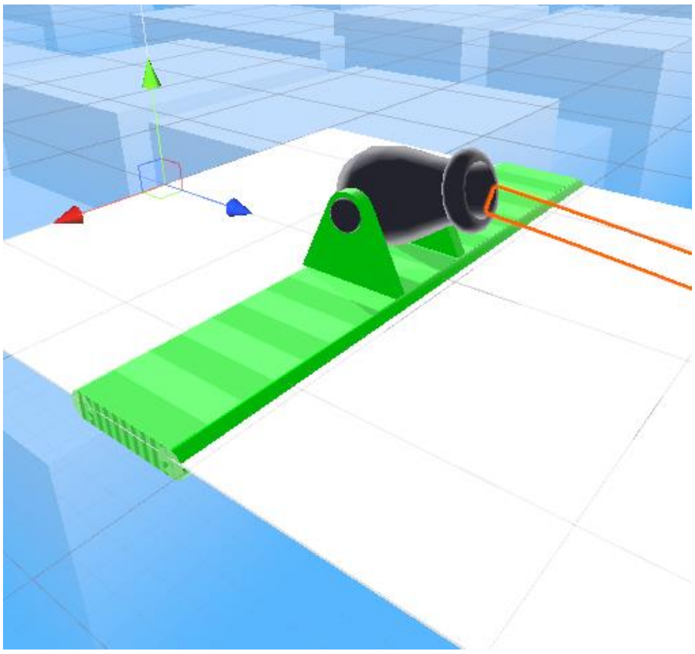


At the same time, to add variety to our sections, we added platforms of different sizes and shapes, as in the image below, and placed our characters on top of them, thus adding another dimension to the game on the Y-axis.



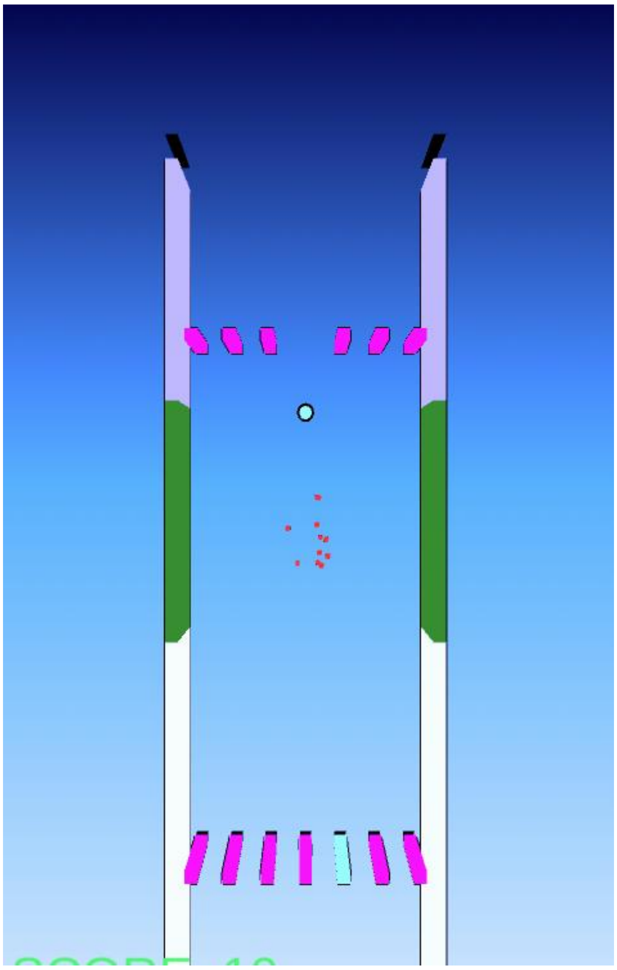


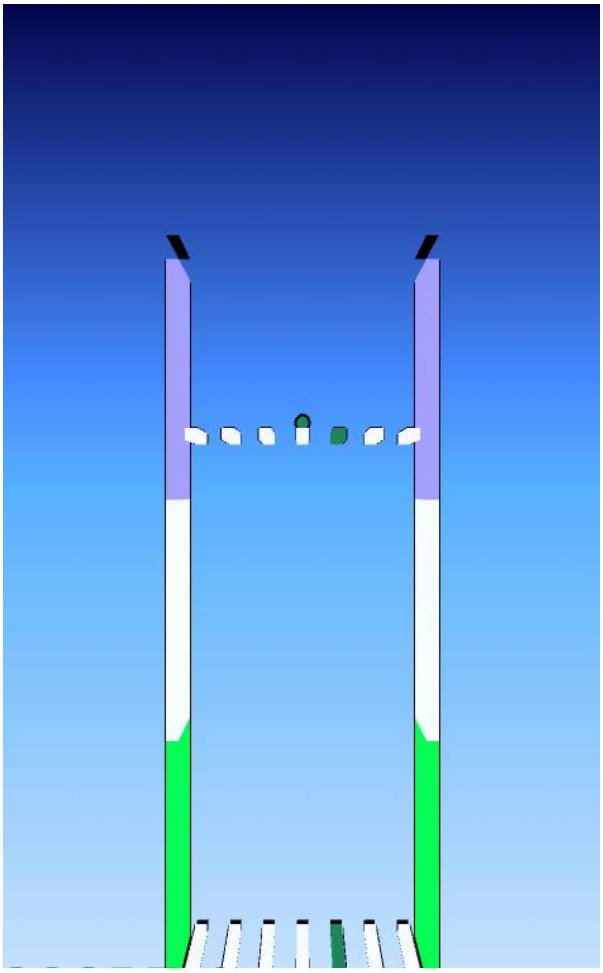
In addition to all that, we've added a theme-appropriate rail to the bottom of our cannonball to indicate its movement in the x-axis.



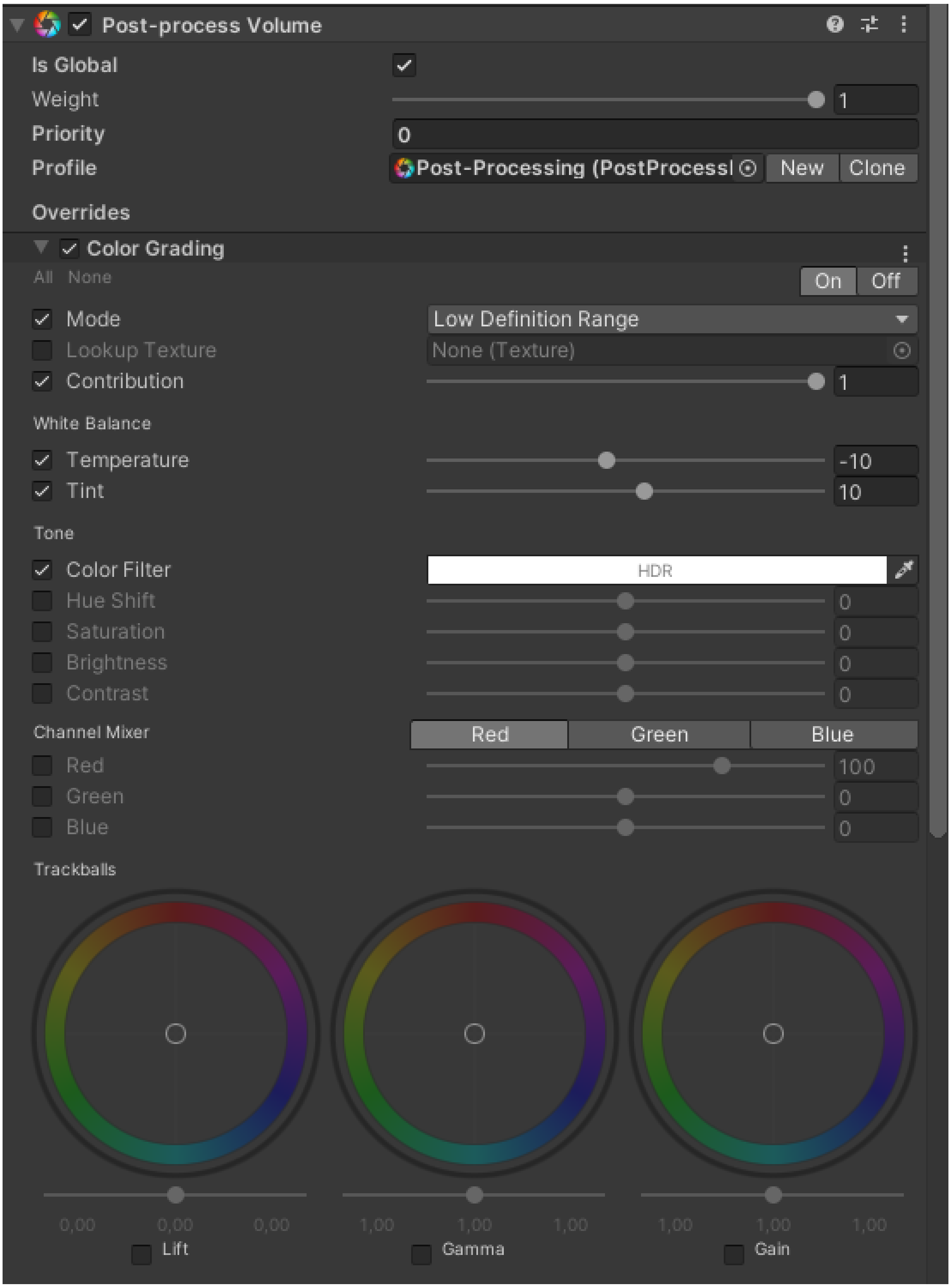
4.3.2) Environment (Fleptris)

In "Fleptris", we placed dashed lines on the right and left walls to show our speed, and these colors change with the game, so we provide both a dynamic image and color transitions for the active that does not bore the user.





We use our custom shaders and the post processing method to control the colors of the ball and the colors of the edges.



☑ Post-process Volume

Is Global ☑

Weight 1

Priority 0

Profile Post-Processing (PostProcessI New Clone

Overrides

☑ Color Grading

All None On Off

☑ Mode Low Definition Range

□ Lookup Texture None (Texture)

☑ Contribution 1

White Balance

☑ Temperature -10

☑ Tint 10

Tone

☑ Color Filter HDR

□ Hue Shift 0

□ Saturation 0

□ Brightness 0

□ Contrast 0

Channel Mixer Red Green Blue

□ Red 100

□ Green 0

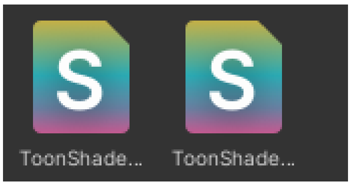
□ Blue 0

Trackballs

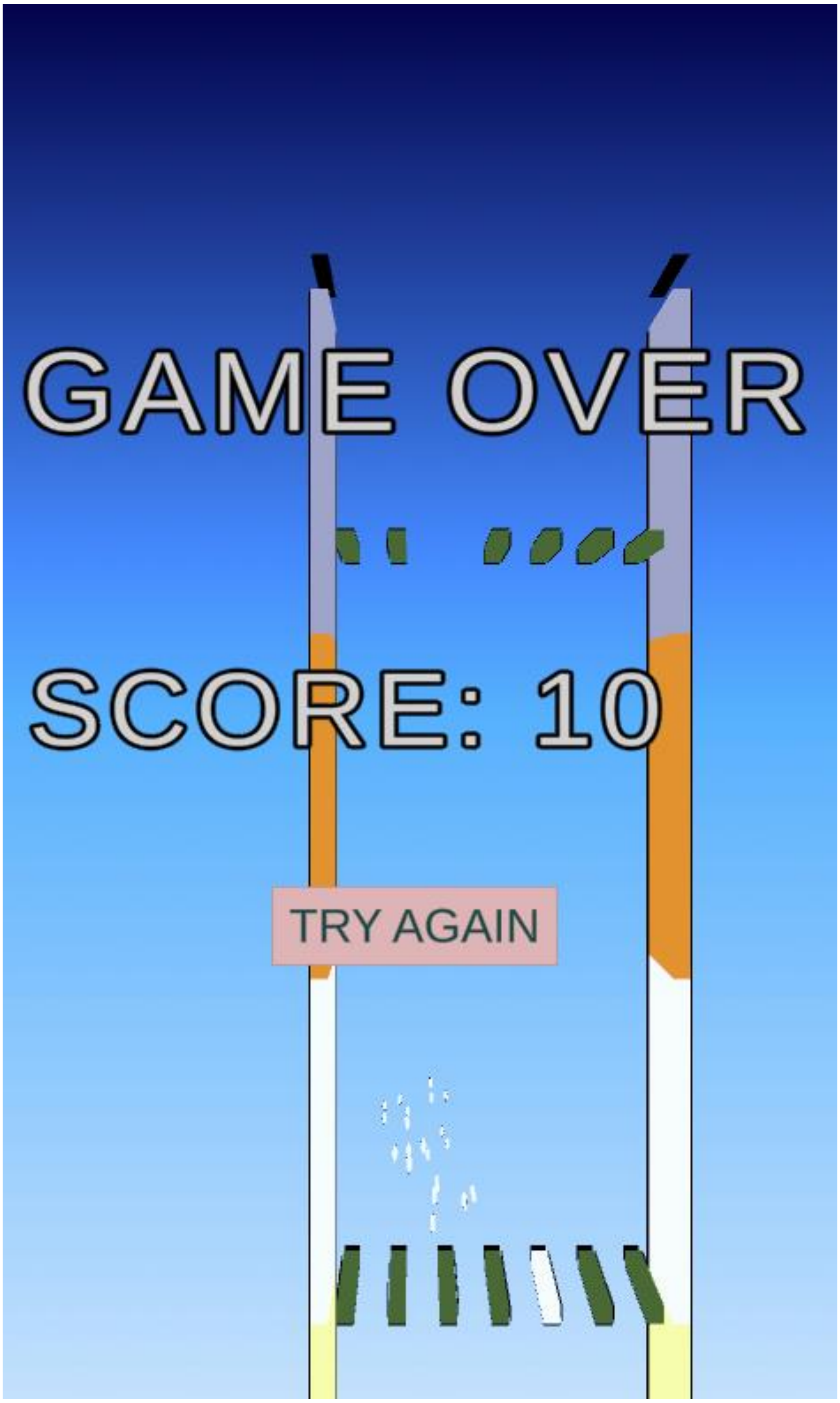
0,00 0,00 0,00 □ Lift

1,00 1,00 1,00 □ Gamma

1,00 1,00 1,00 □ Gain



In addition to these, when our character rubs against the wall, a red ring appears around our character to show the friction effect, if this friction continues, the screen shakes and our character breaks into pieces, the score we make is displayed on the screen and the "Try Again" button appears.



5) Results and Discussion

For now, our cannon smash game was tested on Facebook and Instagram platforms in cooperation with Happy Game Company and Rollic Games and we got 4 data as CPI, CPC, Retention and Playtime.

The CPC value of Cannon Smash that tested by Rollic Games came to $2.26, even though it's too high to be published, other values are important too.

Our Cannon-Smash's CPI value was tested by Rollic Games and result came as $10.42. If this value could be $1 and below, more tests and modifications could be made about the potential of the game.

The playtime value of our game as a result of the test is 270 seconds. According to the industry's experience, this value must be 240 seconds or more to know that a Hyper-Casual game locks the user on the screen.

The retention value we received as a result of the testing of our Cannon Smash game by Rollic Games was 44.44%. This value is enough for a Hyper-Casual game to be considered a very successful and can be published worldwide. But the CPI value should have accompanied this.

However, the game could not be published because the CPI value could not meet the criteria.

With the support of Happy Game Company and the testing phase of Rollic Games, we had the chance to learn by experiencing the keys to financial success in a Hyper-Casual game thanks to the feedback we got from the results mentioned above.

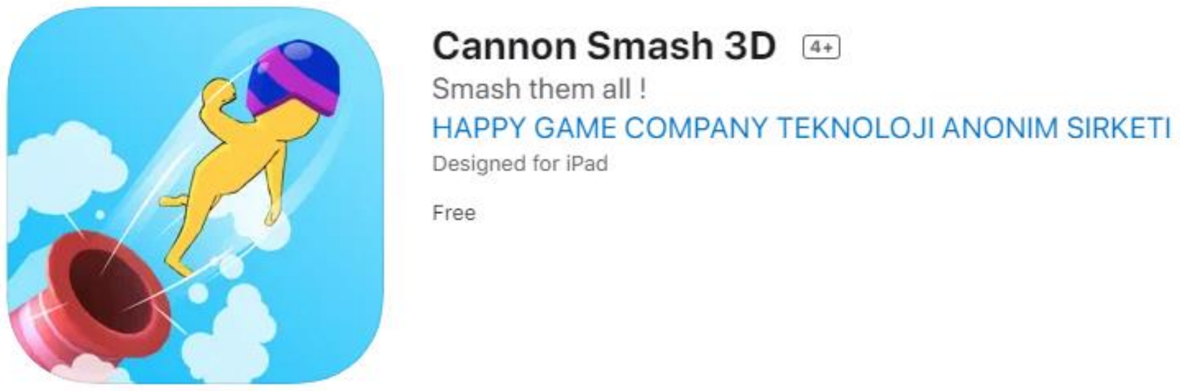
6) Conclusions

In conclusion, we have developed 2 Hyper-Casual games from scratch until they were released in Appstore.

In the process, we learned so many things about the feel of a game, the dynamic nature of Hyper-Casual, rapidly changing industry flows, and many technics about Unity game engine and C# programming language.

It made us proud to provide the industry standards of the games we developed and the standards of the world's giants of the industry.

As a result of the results we have obtained from these projects, we believe that we will move forward with more confident steps in the game industry, and we fully believe that these valuable experiences we have gained before we graduate will give us an advantage in our career.



Cannon Smash 3D 4+

Smash them all !

HAPPY GAME COMPANY TEKNOLOJI ANONIM SIRKETI

Designed for iPad

Free

Figure 7

We would like to thank Happy Game Company, Rollic Games and Prof Dr Hüseyin Akcan for their support in working with professionals and realizing tangible projects before we graduated.

7) References

1. Web Site: “ Demolish! Game review ” , as appears on https://sensortower.com/ios/us/voodoo/app/demolish/1499255972/overview - accessed on June 3rd, 2021.

2. Web Site: “Flappy Bird Game review” , as appears on https://www.ign.com/articles/2014/02/08/flappy-bird-review - accessed on June 3rd, 2021.

3. Web Site: “ Flappy Bird Arts” , as appears on https://fnoschese.wordpress.com/2014/01/30/flappy-bird-when-reality-seems-unrealistic/ - accessed on June 4th, 2021.

4. Web Site: “ Tetris Game review” , as appears on https://www.commonsensemedia.org/app-reviews/tetris - accessed on June 4th, 2021.

5. Web Site: “How to Calculate CPI, CPC, eCPM and More”, as appears on https://gameanalytics.com/blog/how-to-calculate-cpi-cpc-ecpm-and-more - accessed on June 4th, 2021.

6. Web Site: “The Mobile Marketer's Guide to Mastering User Retention”, as appears on https://clevertap.com/blog/guide-to-user-retention - accessed on June 6th, 2021

7. Web Site: “ Cannon Smash Market Page” , as appears on https://apps.apple.com/us/app/cannon-smash-3d/id1568058924 - accessed on June 6th, 2021