UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Grete Hollo

Take a Leap – Hyper-Casual Mobile Game

Bachelor's Thesis (9 ECTS)

Supervisor: Daniel Nael, MSc

Tartu 2023

Take a Leap – Hyper-Casual Mobile Game

Abstract:

This bachelor thesis describes the development and design of a hyper-casual mobile game called *Take a Leap*. The game was made to offer a new but similar experience to the corresponding game genre. Unity game engine was used to develop the game. The game was later tested on potential players to find issues and evaluate the quality of the game. The results of the testing were rather positive, but in the testing process, several problems were revealed that need further development and improvement.

Keywords:

Mobile game, game design, software development, hyper-casual, Unity, usability testing, playtesting

CERCS: P170 Computer science, numerical analysis, systems, control

Take a Leap – *Hyper-Casual* Mobiilimäng

Lühikokkuvõte:

See bakalaureusetöö kirjeldab *hyper-casual* mobiilimängu *Take a Leap* arendamist ja disainimist. Mäng loodi selleks, et pakkuda uut, kuid vastavale mängužanrile sarnast kogemust. Mängu arendamiseks kasutati Unity mängumootorit. Hiljem testiti mängu potentsiaalsete mängijate peal, et leida probleeme ja hinnata mängu kvaliteeti. Testimise tulemused olid pigem positiivsed, kuid selle käigus selgus ka mitmeid probleeme, mis vajavad edasiarendamist ja täiustamist.

Võtmesõnad:

Mobiilimäng, mängudisain, tarkvaraarendus, *hyper-casual*, Unity, kasutatavuse testimine, mängu testimine

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Table of Contents

1 Introduction 5

2 Background 6

2.1 Defining Hyper-Casual Game 6

2.2 History 7

2.3 Inspiration 8

2.4 Similar Games 8

3 Game Design 11

3.1 Main Gameplay 11

3.2 Game Mechanics 11

3.3 Level Design 12

4 Implementation 13

4.1 Unity 13

4.2 Assets 13

4.3 Programming Patterns 17

4.4 Math 17

5 Testing 19

5.1 Methodology 19

5.2 Results 19

5.3 Improvements 24

6 Conclusion 26

References 27

Appendix 30

I. Glossary 30

II. Testing videos 31

III. Source Code 32

IV. Accompanying Files 33

V. License 34

1 Introduction

According to the findings of Newzoo, the global gaming market produced a revenue of $184.4 billion in 2022, with mobile gaming accounting for half of it, $92.2 billion [1]. Hyper-casual mobile games' popularity has significantly increased in recent years [2]. This might be because they have a large target audience and do not require much attention and time to play [3].

This bachelor thesis describes developing and designing a hyper-casual mobile game called *Take a Leap*, where the player has to jump from one platform to another. Unity game engine was used to develop the game. The game was later tested on potential players to find issues and evaluate the quality of the game.

Chapter 2 defines a hyper-casual mobile game and talks about the background and history of the genre. It also discusses the inspiration behind *Take a Leap*, as well as similar games and their comparisons. Chapter 3 explores the design process of *Take a Leap*, covering its concept, mechanics, and level design. Chapter 4 discusses various important aspects of game implementation. It explains why Unity was chosen as the game engine and covers the creation of visual and audio assets, including character design, animations, and sound effects. Additionally, the programming patterns implemented in the game, Object Pool and Singleton, are explored. Lastly, the significance of the Bézier curve is examined. Chapter 5 focuses on the testing process of *Take a Leap*. It describes how testing for the game was conducted and what the results were. Finally, it discusses future improvements planned for the game. Appendix I has some defined terms that are used in this thesis. Screen recordings made from testing the game are in Appendix II. A link to the game's source code is in Appendix III and Accompanying Files (Appendix IV) have the Unity project, build files of *Take a Leap* and testing questions and answers provided by the testers.

ChatGPT[1] was utilized to enhance the readability and aesthetic appeal of the text. The following prompts were employed: "Rewrite this sentence: *text*" and "Rephrase/rewrite this text so it is suitable for a bachelor's thesis: *text*." Grammarly[2] was used to ensure grammatical accuracy and to suggest alternative vocabulary.

[1] https://openai.com/blog/chatgpt

[2] https://app.grammarly.com/

2 Background

This chapter presents the background of the thesis, which focuses on developing a hyper-casual mobile game. To understand the game of this thesis and its design, it is also essential to understand the game's and its corresponding genre's background. Subchapter 2.1 explains what hyper-casual games are and how they are defined in this thesis. The use of AI in game development is also discussed in that subchapter. A brief history of hyper-casual games is presented in subchapter 2.2. Subchapter 2.3 discusses where the author got the inspiration for this thesis's game, *Take a Leap*. Subchapter 2.4 gives examples of similar games and compares them with *Take a Leap*.

2.1 Defining Hyper-Casual Game

The following paragraph is quoted from Michal Wagner's blog post [4]. Hyper-casual games are self-explanatory – the player does not need a separate tutorial. Such games' user interface is minimalistic, making it relatively easy for the user to start the game. The game mechanics are also instantly understandable, which makes the user want to play it again. People may quickly become disinterested or unengaged in these games, but developers can use this to their advantage. If developers see that their game is not being played as much anymore, they can direct players to their new game. The development of hyper-casual games is fast – a game of the corresponding genre can be announced in just a few days because there is no need to invent a story, and trivial game mechanics are used.

To make game development even faster, AI can be used. According to an article by Reza Hassanzadeh[3], ChatGPT has the capability to produce various elements of a story, such as plotlines, conversations, and descriptions of characters, as well as generating code. This tool also has the potential to be utilized in customer service, profit-making, and customized gameplay, ultimately leading to increased player involvement. ChatGPT could also be used to give new ideas of what kind of game mechanics to develop for a game.

[3] Reza Hassanzadeh. How AI is Changing the Game! Joyixir. Feb. 20, 2023. https://www.joyixir.com/post/how-ai-is-changing-the-game

Tom Kinniburgh writes in his article that the ten most popular mobile game mechanics are: tap/timing, stacking, turning, dexterity, rising/falling, swerve, merging, idle, growing, and puzzle mechanics [5]. This thesis uses timing, turning, and rising/falling mechanics.

In this thesis, to avoid confusion, a hyper-casual mobile game definition is that hyper-casual mobile games are self-explanatory, minimalistic, instantly understandable, and have a single action or goal.

Having established the definition of hyper-casual mobile games, it is relevant to look into this game category's historical background.

2.2 History

The first hyper-casual game, *Flappy Bird*, was released in May 2013 [6]. Matija Hanicar writes that the game was downloaded 50 million times in nine months and earned about $50,000 daily from in-game ads. The author removed the game after a couple of years because it received much negative criticism [6]. Florian Zandt writes in his article that hyper-casual games have become increasingly popular. In 2018 only 14 games made it into the top 100 downloaded apps, but in 2021 it had increased by 157 percent (Figure 1) [2]. In 2022, the hyper-casual genre dominated the download charts in the US with 1.6 billion downloads, while the simulation genre came in second with 617 million downloads [7, 8]. Johannes Heinze compares today's hyper-casual games to the arcade games of the 70s and 80s, saying that such a genre has always existed, but has only recently been turned into a major industry [9]. The rise and success of hyper-casual games in recent years have undoubtedly influenced the author's inspiration for developing a competitive and entertaining mobile game.

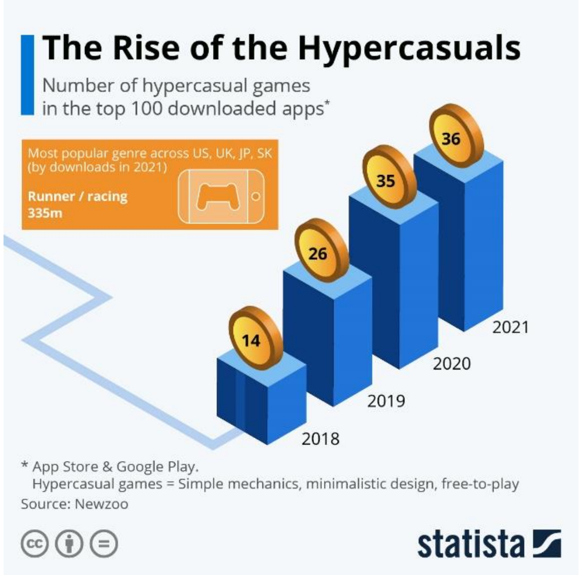

Figure 1. The growth of hyper-casual games [2]

2.3 Inspiration

The author of this thesis was inspired during a team meeting of MITS (Student Society of Mathematics and Informatics[4]). The idea came from devising a competitive game for an event. The invented game's idea featured cups that were upside down on the table and a ping-pong ball that had to be thrown on the first cup and let it bounce on as many cups as possible. Points were scored based on how far the ball reached. It resembled the existing game *Stick Hero*, which will be written about in the next chapter. This meeting inspired the author to come up with the idea of making a hyper-casual mobile game where the player has to jump from platform to platform. In order to start developing the idea, it was immediately sketched on paper, and further processing began. One of the first tasks was finding and researching similar games in the mobile gaming market.

2.4 Similar Games

One similar game is *Flappy Bird* (Figure 2), where the player has to press the screen to keep the bird in the air and not fly into the pipes [10]. A game mechanic similar to this thesis's game is timing. In the game of this thesis, the player has to let go at the right time to reach the next platform. The timing is even more crucial when the platforms move.

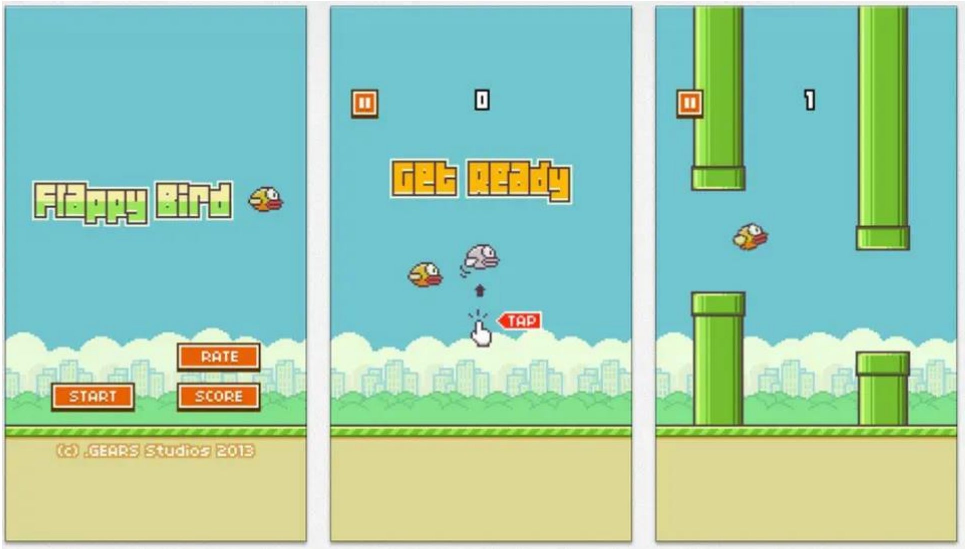

Figure 2. Screenshots of *Flappy Bird*[5] gameplay

[4] https://mits.ee/

[5] https://www.forbes.com/sites/anthonykosner/2014/02/03/flappy-bird-and-the-power-of-simplicity-scaled/?sh=3f7a30b47339

Another game is *Stick Hero*[6] (Figure 3), where the player has to hold their finger on the screen to grow a branch that reaches the following platform. In *Stick Hero*, and also in the game of this thesis, the player has to move from platform to platform by keeping their finger on the screen, which determines the length of further movement.

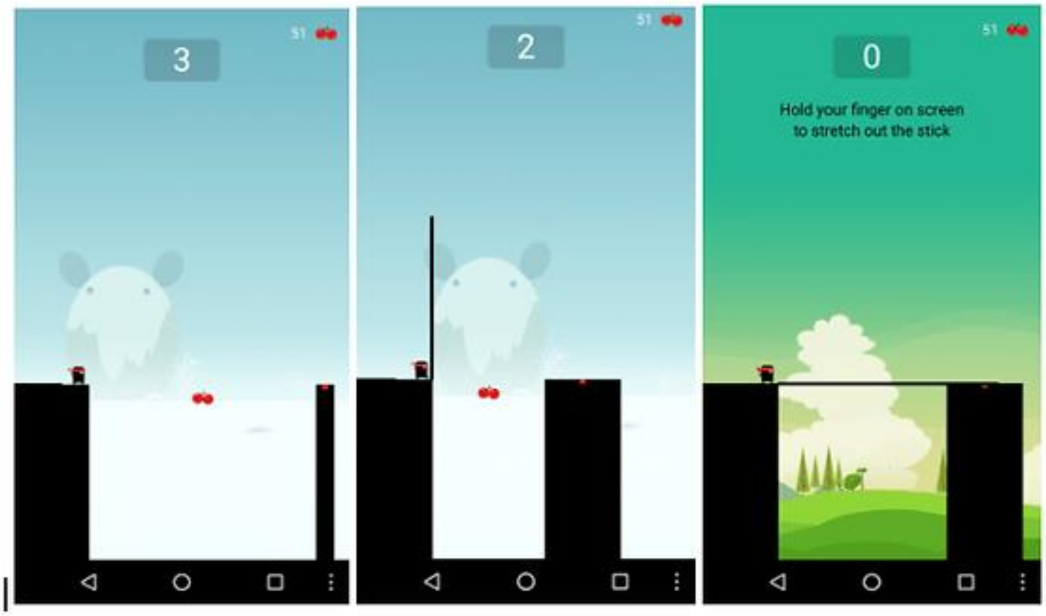

Figure 3. Screenshots of *Stick Hero*[7] gameplay

[6] https://apps.apple.com/us/app/stick-hero/id918338898

[7] https://roboticssocietymnn.wixsite.com/roboticsclubmnnit/single-post/2018/04/01/stick-hero-game-automation

A game named *Lyto in Wonderland* (Figure 4) has similar gameplay and uses similar mechanics. It is a 2D game developed by GameN[8]. The game is played by holding down a finger and jumping from platform to platform. Arrow moves only forward, and if the player holds it too long, they will definitely lose the game because they will miss the platform. In this thesis's game, *Take a Leap*, the arrow reverses its moving direction when the player holds down too long. Also, in *Take a Leap*, the game has been made more complex by having to aim in the right direction, but in *Lyto in Wonderland*, there is no aiming. As the levels progress, sometimes the platforms are ice cubes where the player cannot stay too long because they will melt. Furthermore, the platforms are sometimes smaller, so jumping on them is more challenging.

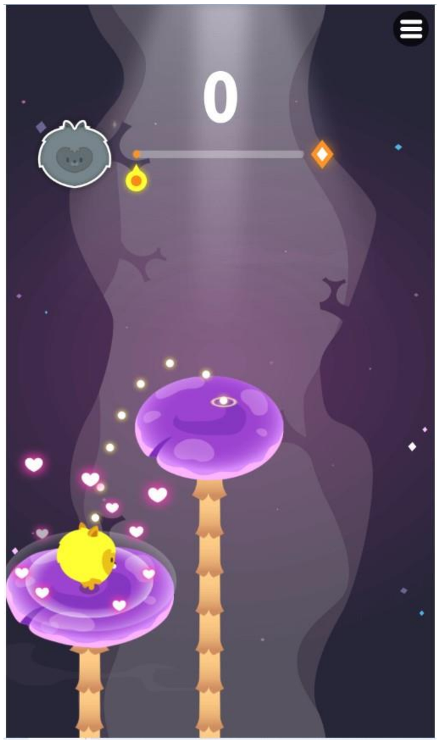

Figure 4. *Lyto in Wonderland* gameplay

By studying and analyzing the features of the games mentioned in this chapter, a better understanding can be gained of what makes a game successful and how these principles can be applied to the game design of *Take a Leap*.

[8] https://en.gamen.com/?categame=new&subcate=genre#fromNo_3

3 Game Design

The success of a mobile game largely depends on its design. A well-designed game can create an immersive experience that keeps players coming back for more [11]. This chapter examines the game design process for *Take a Leap*. Subchapter 3.1 discusses the concept, and subchapter 3.2 the mechanics of a mobile game developed for this thesis. The level design of *Take a Leap* is explained in subchapter 3.3.

3.1 Main Gameplay

The main gameplay is the core of the game experience [11]. The idea of the game developed during this thesis is that the player has to jump from one platform to another. If the player holds their finger on the screen, an arrow appears that shows how far the character can jump. The arrow moves back and forth until the player lifts their finger from the screen. Also, the player has to move their finger, either left or right, depending on which side the next platform is. When the finger is lifted from the screen, the player moves by arrow distance in the selected direction. On reaching a platform, the player gets a point. Crossing a certain number of platforms, the player reaches the next level where the platforms move. The levels become increasingly difficult. The goal is to collect as many points as possible, i.e., to reach as far as possible. Before diving into the details of the game design, it is important to understand the core mechanics that will drive the player's experience.

3.2 Game Mechanics

Game mechanics are the rules and systems that govern how the game is played [12]. The hyper-casual mobile game developed during this thesis mainly uses three game mechanics: timing, turning, and rising/falling mechanics. The timing mechanic is used when the player holds their finger on the screen and has to wait for the proper distance to jump to the next platform. The distance is indicated by an arrow that moves back and forth and appears when the player holds their finger on the screen. The turning mechanic is used to determine the direction of the jump. To do this, the user has to move their finger, either left or right, while holding it on the screen. The rising/falling mechanic is taken into use from the third level when the platforms start moving up and down.

With a basic understanding of the game mechanics established, the next focus will be on designing levels that challenge the player's skills and maintain their engagement with the game.

3.3 Level Design

Level design is the process of creating the game's challenges and obstacles [13]. In the final prototype, *Take a Leap* has three levels, and the only challenges are the moving platforms and timing the jump. In the first level, the platforms are not moving. The player has time to get used to the arrow movement and train their timing accuracy. In the second level, some platforms start moving left and right. This provides more challenge to the player because now they have to be more careful. When they target the arrow in the platform's center and jump, they most likely will miss because the platform will move away beneath them. The third level introduces up-and-down moving platforms. They provide a new movement to the game so the player finds something new and does not get bored. With the level design completed, the next step is to implement it into the game in the development phase.

4 Implementation

The implementation of the game is crucial to its success, and in this chapter, various aspects of it will be discussed. In subchapter 4.1, the author explains why Unity was chosen as the game engine. Subchapter 4.2 discusses the creation of the game's visual and audio assets, including the arrow, the character's design and animations, the platforms, and sound effects. The impact of sound effects and background music on the player's gameplay experience is also examined. Subchapter 4.3 explores two programming patterns used in the game: Object Pool and Singleton. The author explains why these patterns were selected and how they were implemented in the game. Lastly, in subchapter 4.4, the reader will learn about the Bézier curve and its significance in computer graphics and design systems. The author explains how this curve was used in the game.

4.1 Unity

The game, *Take a Leap* has been made in Unity[9]. Unity is a popular 2D and 3D game engine that has been made for game development[10]. It was taken into use because the author had previous experiences with this game engine. Furthermore, it is used a lot for hyper-casual game development. The hyper-casual genre has shown remarkable growth in the number of Unity games published from 2019 to 2021, with a 137% increase [14]. Popular mobile games made with Unity are *Crossy Road*[11] and *Subway Surfers*[12].

4.2 Assets

An asset can be anything that is part of the game [15]. It can be an audio file, 3D model, script, image, or something else. The following subchapters discuss the specific assets created for the game *Take a Leap*.

[9] https://unity.com/

[10] Lindsay Schardon. "What is Unity? – A Guide for One of the Top Game Engines". GameDev Academy. 2023. https://gamedevacademy.org/what-is-unity/#What_is_Unity

[11] https://mainleaf.com/mobile-games-made-with-unity/

[12] https://web.archive.org/web/20190324154937/https://unity3d.com/showcase/case-stories/sybokiloo-subwaysurfers

4.2.1 Arrow

The first arrow was a blue linear line (Figure 5) that showed the jump length. It was made in Unity, using its built-in 3D game object – cube. It moved forwards and back to the player. It was challenging to interpret in the 3D game world where the character will land.

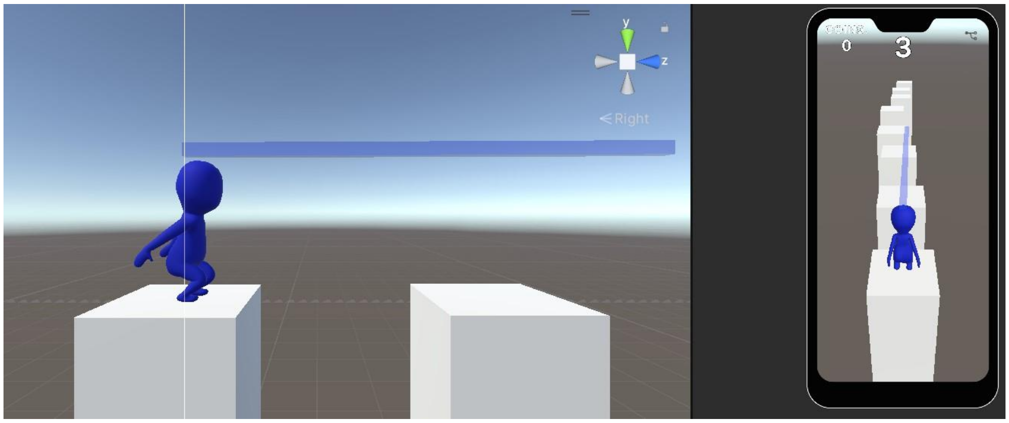

Figure 5. First arrow

The second arrow was made in Blender[13] (Figure 6). This arrow's length stayed the same, but this time it did not move forward. It only moved up and down to show the jumping angle. In this way, knowing where the character would land was also very hard.

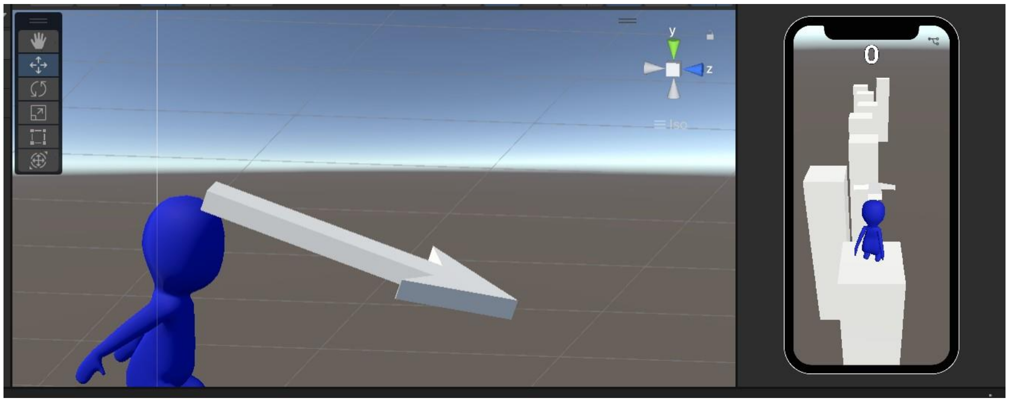

Figure 6. Second arrow

The last arrow used the Bézier curve and Unity's Line Renderer (Figure 7). The Line Renderer component is used to connect two or more points in 3D space by drawing a straight line between

[13] https://www.blender.org/

them, requiring an array of points as input[14]. It moved forwards and back to the player, indicating jumping length. Its endpoint showed where the character would land. This is the arrow that was used in the final prototype of the game and also tested.

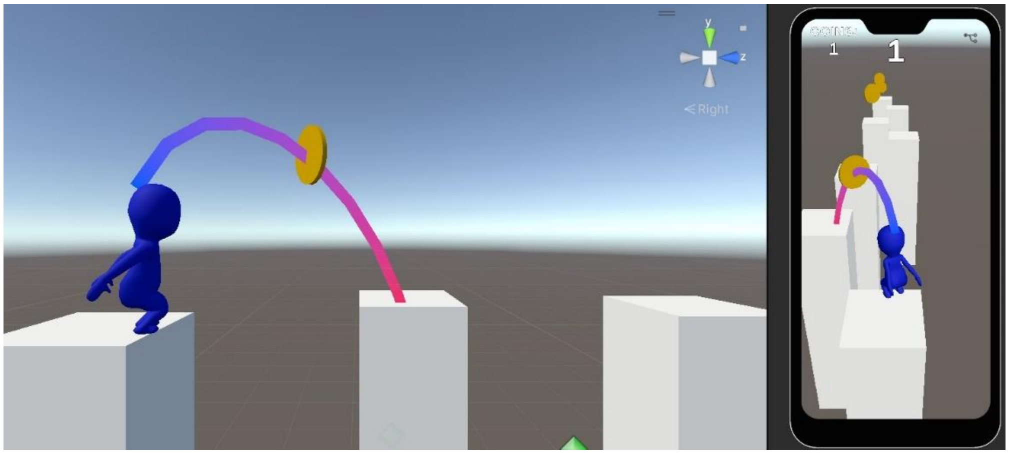

Figure 7. Third arrow

4.2.2 Character

The character and its animations have been made in Blender. The author watched YouTube tutorials[15] to learn about the topology of humanoid characters and how they are made. It is minimalistic and has only one color because that was the most common style when looking through other hyper-casual mobile games. It has five animations: idle, jump, jump, start falling, and falling.

4.2.3 Platforms

Platforms are created using Unity’s built-in game objects, cubes, that have been modified. These platforms can either move left and right (Figure 8), up and down (Figure 9), or remain still.

[14] https://docs.unity3d.com/Manual/class-LineRenderer.html

[15] https://youtu.be/ftSMD447O1Y, https://youtu.be/4fICQmBEt4Y, https://youtu.be/4z7G4TyKE9g, https://youtu.be/ogz-3r0EHKM

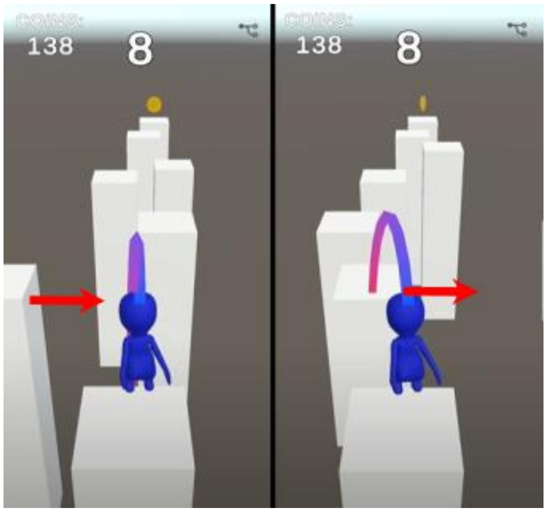

Figure 8. Left-right moving platforms

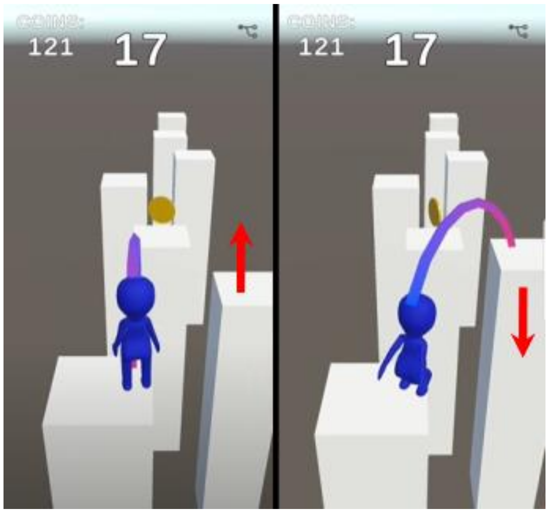

Figure 9. Up-down moving platforms

4.2.4 Coins

Coins are also created using Unity's built-in game objects, cylinders. They have been modified, and metallic gold material has been added. They also have a spinning animation. Coins can spawn as a single coin on one platform or multiple coins from one platform to the next (Figure 10). The Bézier curve is used to construct multiple coin trajectories.

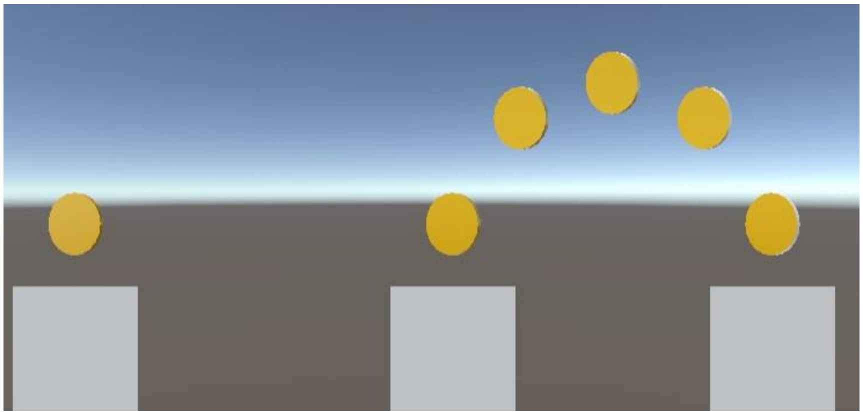

Figure 10. A single coin is on the left and Bezier curve coins are on the right

4.2.5 Audio

Sound effects and background music can influence the player's emotions [16]. Audio impacts the player's gameplay experience and overall satisfaction in performing an action [16]. For this thesis game, Jsfxr[16] was used to generate coin collecting, jumping, and falling sounds. The coin collection sound will change pitch to higher when two coins are collected less than 1.8 seconds apart. When a coin is collected more than 1.8 seconds from the last one, then the sound that is played will be with the lowest pitch. There is also an option in the settings menu to set the sound effects volume.

4.3 Programming Patterns

Programming patterns are general, reusable solutions to commonly occurring problems within a given context in software design [17]. Subchapters 4.3.1 and 4.3.2 explain how two patterns are used in *Take a Leap*.

4.3.1 Object pool

Object pool is an optimization pattern that reuses objects from a fixed pool instead of allocating and freeing them individually [17]. This thesis uses it for rendering platforms that the player jumps on and for rendering coins. This pattern was considered and used because the player sees a fixed number of platforms and coins at once.

4.3.2 Singleton

Singleton is a design pattern that allows only one instance of a class to be created and offers a universal means to access that instance [17]. Unity's built-in class PlayerPrefs[17] uses singleton. PlayerPrefs in Unity follows this pattern, meaning that only one instance of the class is available during gameplay, and all requests for PlayerPrefs refer to that same instance. This functionality provides a practical way to save and retrieve game data that needs to persist between gaming sessions, like high scores, player preferences, or other relevant game data. By adopting a singleton pattern, Unity guarantees that all code that interacts with PlayerPrefs shares the same instance, reducing potential conflicts or inconsistencies in data.

4.4 Math

The following paragraph is based on a part of the book written by Michael E. Mortenson [18]. The Bézier curve is a commonly used mathematical curve in computer graphics and computer-aided

[16] https://sfxr.me/

[17] https://docs.unity3d.com/ScriptReference/PlayerPrefs.html

design systems. It is used to define the shape of curves and surfaces, from car designs to type fonts. Due to its numerical stability, it is the ideal standard for representing complex piecewise polynomial curves. The curve is named after Peter Bézier, who developed it in the 1960s to provide design engineers with an intuitive way to alter and control shape. The construction of a Bézier curve is simple, and it can be derived algebraically using basis functions, control points, and degree elevation. Two curves can also be joined end-to-end to form a composite curve. In this thesis, a four-point Bézier curve has been used for multiple coins and arrow trajectories. The formula for the 4-point Bézier curve (Figure 11) is seen in Equation 1 where t is a parameter that varies between 0 and 1 and determines the position along the curve.

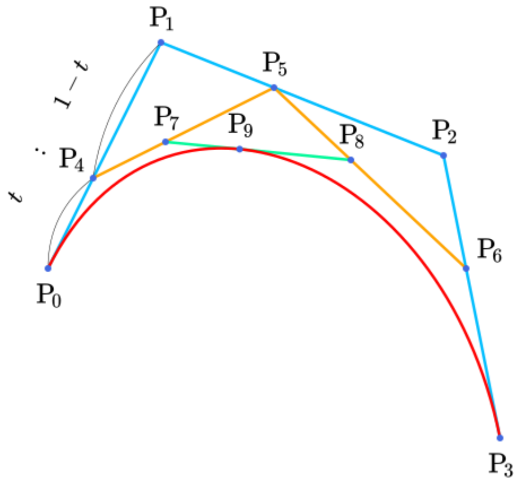

Figure 11. A cubic Bezier curve consisting of endpoints P0, P3 and control points P1, P2[18]

$$P(t) \ = \ (1-t)^3P_1 + 3(1-t)^2tP_2 + 3(1-t)t^2P_3 + t^3P_4 \qquad (1)$$

[18] https://commons.wikimedia.org/wiki/File:Cubic_B%C3%A9zier_Curve.svg

5 Testing

Testing is an important aspect of game development. It helps find bugs and areas that need improvement. Subchapter 5.1 describes how the testing of *Take a Leap* was conducted. The results are analyzed in subchapter 5.2. Subchapter 5.3 talks about improvements that are planned to be made to the game.

5.1 Methodology

The playtesting process is of utmost importance as it enables game developers to identify and address any issues that may be present in a computer game [11]. The testing was conducted on five individuals. Jakob Nielsen found in his study that five people were sufficient to identify the majority of usability concerns [19]. The testing was focused on evaluating the extent to which players understood the user interface of *Take a Leap* and the level of satisfaction derived from the gaming experience, as these factors are critical to the success of a computer game [20, 21]. Each tester participated in an individual session where they played *Take a Leap* without prior knowledge of the game while being observed. The testers were not provided with external assistance during gameplay and were required to navigate and get as high of a score as possible. If they had not pressed some buttons, they were told to press them at the end of the session. Their gameplay was recorded (Appendix II) for further analysis. Additionally, each tester was required to complete a form after the session, which included questions relating to their background, overall gaming experience, game mechanics, level design, user interface, and visual style. The questions and answers are in the Accompanying files (Appendix IV). The results obtained from the study were critical in determining the effectiveness of the design choices implemented in *Take a Leap* and identifying areas that required improvement.

5.2 Results

Testers were asked to fill out a form about the game’s design and feel. Subchapter 5.2.1 discusses testers’ backgrounds and gaming experiences. Testers’ comments and feedback about the game are analyzed in subchapters 5.2.2 to 5.2.7.

5.2.1 Testers

There were five testers in total. Two of them play mobile games once a month, and three do not play mobile games at all. This information is relevant because it affects their overall experience with the game and their ability to provide feedback based on their previous gaming experiences.

5.2.2 Overall Experience

The testers were asked to rate the game's fun score on a scale from 1 to 6, where 1 meant "awful" and 6 "perfect" (Figure 12). The average fun score was 4.0. Testers were also asked to explain their rating. It was said that the game got repetitive and, therefore, boring. Testers suggested adding more path choices and modifying gameplay speed to faster and slower. They were also asked to rate the game's difficulty from 1 to 6 (Figure 13), explain their rating, and share their favorite and least favorite moments during the gameplay. In the difficulty rating, 1 meant "very easy", and 6 meant "very difficult." *Take a Leap*'s difficulty was 3.6 on average. According to the opinions of testers moving platforms made the game more challenging. The exact landing spot or depth of the jump was hard to interpret. Testers' favorite moments during the gameplay were when the platforms started to move, collecting coins, squatting animation, and getting 21 points. Furthermore, the least favorite moments were when the tester missed the jump and the game ended, had to jump on the horizontally moving platform, and could not skip one platform making a long jump.

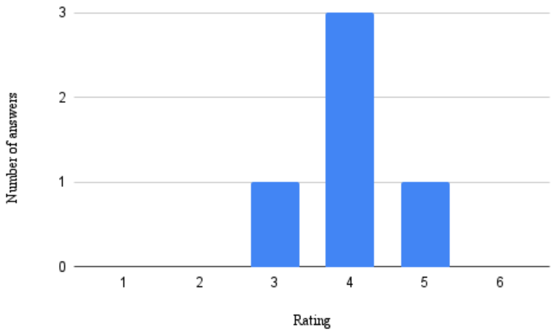

Figure 12. The ratings of overall gameplay experience in terms of fun

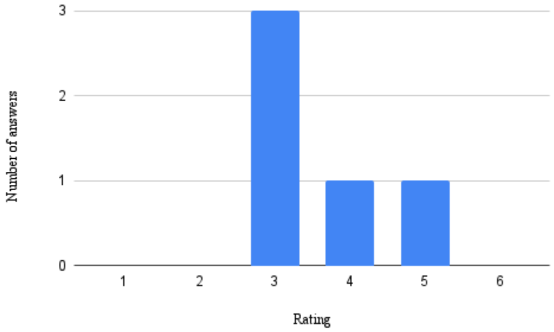

Figure 13. The ratings of the difficulty of the gameplay

The overall experience was relatively positive. The difficulty score landed almost in the middle, which is good. The testing showed shortcomings in the game's enjoyment factor, which can be enhanced by improving game mechanics, level design, and overall user experience. To understand which game mechanics needed improvement, the testers were asked to rate them individually.

5.2.3 Game Mechanics

Testers were asked to rate different game mechanics: moving the arrow, jumping, accuracy of the jump, and collecting coins. The scale it was rated was from 1 to 6, where 1 meant "awful" and 6 meant "perfect" (Figure 14). All mechanics got over 3.0 as the average score, meaning none were "awful" but rather good. The jumping mechanic got the highest average score of 4.8. Testers were also asked to name their favorite and least favorite gameplay mechanic. Jumping was mentioned the most as a favorite gameplay mechanic, and jumping accuracy as the least favorite. It was suggested that to make the arrow movement more smooth, faster, and accurate, add two mechanics for the arrow: fast swipe and slow precise jump, chain up jumps, and add a marker where the landing spot is supposed to be.

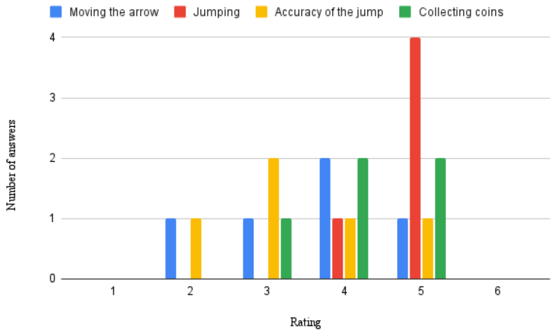

Figure 14. The ratings of game mechanics

Testers rated various game mechanics on a scale of 1 to 6 and found that all mechanics were good, with jumping getting the highest average score. The game mechanics analysis revealed important insights about the player's experience. The mechanics are implemented in level design, and the next chapter will discuss players' feedback on level design.

5.2.4 Level Design

Take a Leap's difficulty increases as the levels increase. Levels increase when the player has collected a certain number of points. Testers were asked to rate how the game's difficulty increased on a scale of 1 to 5, where 1 meant "too fast" and 5 meant "too slow" (Figure 15). Most testers rated 3, which means that the difficulty increased just right, and the average was 2.8, which

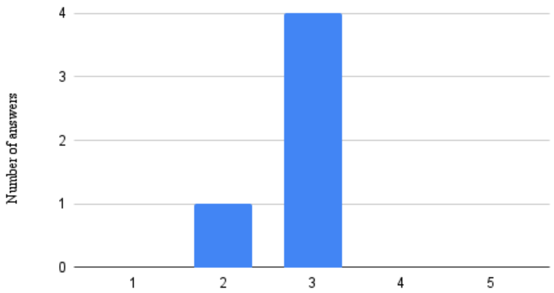

Figure 15. The ratings of the difficulty increase

indicates that the increasing speed can be made a tiny bit slower. Testers were also asked what platform movements were most and least challenging. The hardest ones mainly were left-right moving platforms, and for one tester, up-down moving also. The most effortless platforms for all testers were those that did not move at all. For improvements, many suggestions were made: making the platforms disappear when taking a long time to jump to the next, adding path choices and something like trees on the sides so it helps to measure depth, adding obstacles and power-ups, and correcting platform horizontal distance from one another. The excellent difficulty score might be because of the influence of the user interface.

5.2.5 User Interface

Controls in the game *Take a Leap* were holding down and moving the finger on the screen. Testers were asked to rate the controls in terms of comprehension on a scale of 1 to 6, where 1 meant "very confusing" and 6 meant "very intuitive" (Figure 16). Also, what was good about the controls and what could be improved. The average score was 5.4. Most testers liked that they were easy to use and understand. One tester said that the controls were "easy to figure out and complex enough for playing 30min." Sometimes the controls got stuck for a short time. For improvements, it was suggested to make arrow controls more sensitive and add two sets of controls: one for fast swiping and one for slow methodical gameplay.

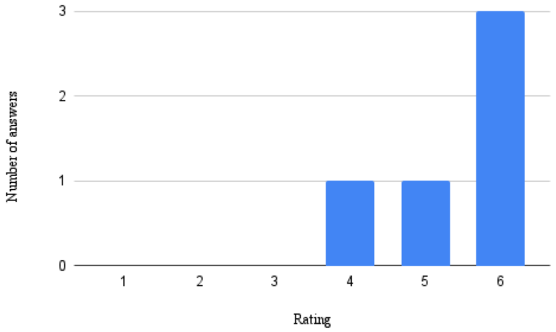

Figure 16. The ratings of the controls in terms of comprehension

They were also asked to rate the graphical user interface in terms of usability on a scale of 1 to 6, where 1 meant "awful" and 6 meant "perfect" (Figure 17), and the same thing in terms of comprehension on a scale of 1 to 6, where 1 meant "very confusing" and 5 meant "very intuitive" (Figure 18). The average score for usability was 5.4, and for comprehension was 5.0.

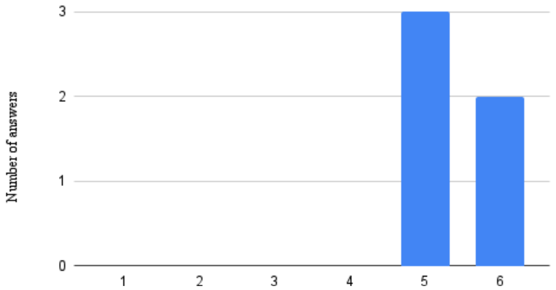

Figure 17. The ratings of the graphical user interface in terms of usability

How they understood how the player can jump was different for almost every tester. One read the text on the first screen, that said understood when they saw the arrow. They were asked if there was anything that can be improved. They suggested making the setting button more visible, renaming “MENU” to “REPLAY”, making graphics prettier, and adding instructions when playing for the first time.

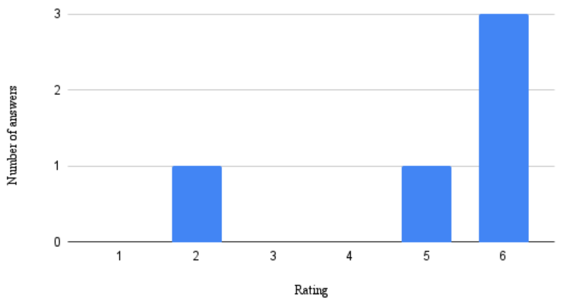

Figure 18. The ratings of the graphical user interface in terms of comprehension

5.2.6 Visual Style

The visual style includes everything that is visible to the player. The testers were asked to rate the appearance (Figure 19) and animations (Figure 20) on a scale of 1 to 6, where 1 meant “awful” and 6 meant “perfect”, and explain their rating. Appearance got the average score of 3.6, and for the explanation, it was said that the game was a bit too minimalistic, felt a bit raw, could be more polished, and they have seen games that are on a higher level. They suggested not using Unity’s default assets and having objects instead of platforms. Animations got 4.6 as the average score, and testers said that they could have been snappier, were jittery sometimes, and the arrow could be improved. Improvements based on testers’ suggestions: make the high score visible on the screen at all times, make the arrow change color relative to the distance, and add a separate death animation.

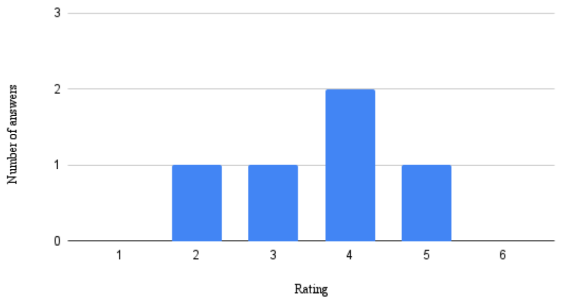

Figure 19. The ratings of the appearance

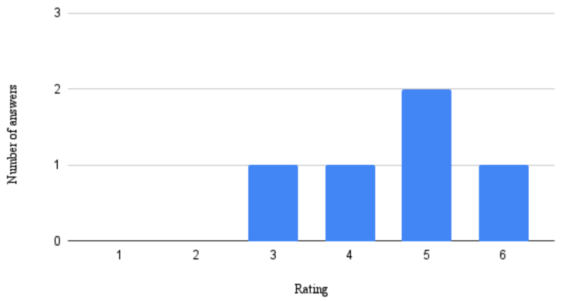

Figure 20. The ratings of the animations

5.2.7 Conclusion

Testers were asked how likely they would recommend *Take a Leap* to their friend on a scale of 1 to 6, where 1 meant "very unlikely" and 6 meant "very likely" (Figure 21). All testers chose different answers. The average score was 3.8 out of 6, meaning they are "rather likely" to recommend this to a friend. One tester rated 1 and explained that they do not recommend games to friends. Another, who rated 3, said this game has potential but would not recommend it right now.

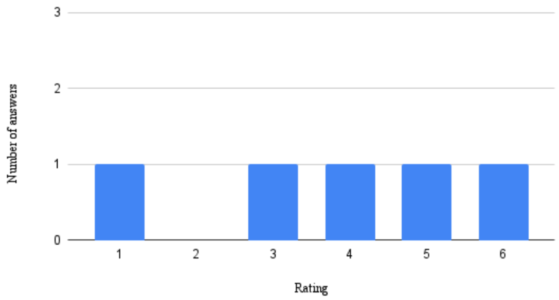

Figure 21. Ratings of how likely are testers to recommend *Take a Leap* to their friends

5.3 Improvements

No changes were made to the game after testing, but it is planned to fix the bugs found during the testing. Also, some suggestions made by the testers that are also brought out in chapter 5.2 are planned to be implemented.

Bugs that were found:

- The character goes inside the platform when landing on an up-down moving platform;
- After changing the volume, closing the game, and opening the game, the volume settings slider shows full volume, but it plays on the volume that was put before;
- Sometimes the next platform is too far on the side that it cannot be reached;
- Character jumping animation always takes the same time to complete, even when the jump is shorter or longer.

Ideas for future development:

1. When the player lands on the platform's edge, the character starts to fall and can be saved by tapping the screen multiple times;
2. Add achievements (reach score 100, 200, etc.);
3. Make the high score always visible when the game ends;

4. Rename the button “MENU” to “RESTART”;
5. Make the platforms disappear when taking a long time to jump to the next;
6. Add path choices;
7. Add something like trees on the sides;
8. Add obstacles;
9. Add a landing marker;
10. Add power-ups.

6 Conclusion

Hyper-casual mobile games have become more popular in recent years. They are self-explanatory, minimalistic, and instantly understandable. This thesis aimed to develop a hyper-casual mobile game *Take a Leap*. *Flappy Bird*, *Stick Hero,* and *Lyto in Wonderland* are similar to this thesis's game. *Take a Leap*'s main gameplay includes jumping from one platform to another. Game mechanics used are timing, turning, and rising/falling. The main challenges in this game are moving platforms and timing the jump.

The game was developed with the Unity game engine because the author had used it before and it is an excellent tool for making hyper-casual games. Many assets, like the character, animations, platforms, coins, scripts, and audio clips, were created for the game. Three versions of arrows were made, and the last one was selected for the final prototype. The Bézier curve has been used for multiple coins and the final arrow trajectories. Two programming patterns were used to optimize the code: singleton and object pool.

The feedback received for the game was mostly positive, with players finding the game appropriately challenging. However, suggestions were made to improve the game's fun factor by refining the game mechanics, level design, and overall user experience. Additionally, the difficulty of the levels progressed at an appropriate pace. Many improvements can be made, and the testers also gave new ideas that can be added to the game.

The author would like to acknowledge and thank the supervisor, CGVR lab, and others who have supported completing this thesis. Special thanks to Daniel Nael for guidance, feedback, positivity, and patience throughout the thesis's development. The CGVR lab provided helpful materials and resources that aided in conducting research and developing the game. Additionally, the author would like to express their gratitude to the testers who provided feedback that helped improve the game's design and implementation.

References

[1] Wijman T. The Games Market in 2022: The Year in Numbers. *Newzoo*. 2022. https://newzoo.com/resources/blog/the-games-market-in-2022-the-year-in-numbers (05.05.2023)

[2] Zandt F. The Rise of the Hypercasuals. *Statista*. 2022. https://www.statista.com/chart/27525/number-of-hypercasual-games-in-the-top-100-downloaded-apps/ (05.05.2023)

[3] Eden M. Hyper-Casual Games: Why Are They So Popular? *Melior Games*. 2020. https://meliorgames.com/game-development/hyper-casual-games-why-are-they-so-popular/ (05.05.2023)

[4] Wagner M. How lightweight hyper casual games became the heavyweights of mobile gaming. *AppsFlyer*. 2021. https://www.appsflyer.com/blog/mobile-marketing/hyper-casual-games/ (02.12.2022)

[5] Kinniburgh T. Top 10 Game Mechanics for Hyper Casual Games. *Mobilefreetoplay*. 2018. https://mobilefreetoplay.com/top-10-game-mechanics-for-hyper-casual-games/ (03.12.2022)

[6] Hanicar M. Hyper Casual Games: Mobile Gaming's Most Profitable Genre? *Udonis*. 2022. https://www.blog.udonis.co/mobile-marketing/hyper-casual-game (02.12.2022)

[7] Knezovic A. 163 Mobile Gaming Statistics for 2022 That Will Blow Your Mind. *Udonis*. 2022. https://www.blog.udonis.co/mobile-marketing/mobile-games/mobile-gaming-statistics (01.12.2022)

[8] Data AI. State of Mobile 2022 Report. *Data AI*. 2022. https://www.data.ai/en/go/state-of-mobile-2022/ (01.12.2022)

[9] Heinze J. The ascendance of hyper-casual part two: What defines the genre? *Pocket Gamer Biz*. 2017. https://www.pocketgamer.biz/comment-and-opinion/65390/hyper-casual-part-two-what-defines-hyper-casual/ (01.12.2022)

[10] Williams R. What is Flappy Bird? The game taking the App Store by storm. *The Telegraph*. 2014. https://www.telegraph.co.uk/technology/news/10604366/What-is-Flappy-Bird-The-game-taking-the-App-Store-by-storm.html (02.12.2022)

[11] Schell J. The Art of Game Design – A Book of Lenses. United States of America, Burlington: *Morgan Kaufmann Publishers*. 2008

[12] Herger M., Kumar J. Gamification at Work: Designing Engaging Business Software. *Interaction Design Foundation*. 2013. https://www.interaction-design.org/literature/book/gamification-at-work-designing-engaging-business-software (07.05.2023)

[13] Karlsson T., Brusk J., Engström H. Level Design Processes and Challenges: A Cross Section of Game Development. *Games and Culture*. 2022. https://www.diva-portal.org/smash/get/diva2:1712739/FULLTEXT01.pdf (03.05.2023)

[14] Unity Gaming Report 2022. *Unity Technologies*. 2022. https://brand-news.it/wp-content/uploads/2022/03/Unity-Gaming-Report-2022.pdf (07.05.2023)

[15] Vionix. What is an asset in game development? *VionixStudio*. 2022. https://vionixstudio.com/2021/12/04/what-is-an-asset-in-game-development/ (04.05.2023)

[16] Andersen F., Danny, King C. L., Gunawan A. A. S. Audio Influence on Game Atmosphere during Various Game Events. *ScienceDirect*. 2021. https://www.sciencedirect.com/science/article/pii/S1877050921000016 (01.05.2023)

[17] Nystrom R. Game Programming Patterns. *Genever Benning*. 2011. https://gameprogrammingpatterns.com/contents.html (25.04.2023)

[18] Mortenson M. E. Mathematics for Computer Graphics Applications. Second edition. United States of America, New York: *Industrial Press Inc*. 1999.

[19] Nielsen J. Why You Only Need to Test With 5 Users. *Nielsen Norman Group*. 2000. https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ (20.04.2023)

[20] Flodman S., Fiona Stewart Karlsson. Success Factors in Profitable Mobile Games. *KTH Royal Institute of Technology*. Bachelor thesis in Media Technology. 2014. http://www.diva-portal.org/smash/get/diva2:721114/FULLTEXT01

[21] Northington A. The Psychology Behind Successful Mobile Game Design. *Braavo*. 2016. https://www.getbraavo.com/blog/psychology-successful-mobile-games/ (22.04.2023)

Appendix

I. Glossary

1. Hyper-casual game – a self-explanatory game with a minimalistic user interface, and instantly understandable game mechanics, it is designed to be easy to start playing and intended to be played in short sessions [4].
2. User interface – the visual and interactive part of a computer program or device that allows users to interact with it, including buttons, menus, screens, and other visual elements[19].
3. Programming pattern – general, reusable solution to a commonly occurring problem within a given context in software design [17].
4. Game mechanics – consist of rules and feedback loops, are designed to create a pleasurable gaming experience, and can be utilized to gamify various non-game situations [12].
5. Assets – anything that is part of the game [15].

[19] https://www.techtarget.com/searchapparchitecture/definition/user-interface-UI

II. Testing videos

Testing 1: https://youtu.be/t9hGoa-VJRU

Testing 2: https://youtu.be/AWrGIrYJRB8

Testing 3: https://youtu.be/UNWWN8sdfP0

Testing 4: https://youtu.be/RTNqilklglo

Testing 5: https://youtu.be/3czPbOrbmbI

III. Source Code

Take a Leap's source code and assets can be found at https://github.com/greteh/Take-a-Leap.

IV. Accompanying Files

Accompanying_files_Take_a_leap.zip contains the following folders and files:

- /Source folder contains Unity project files of *Take a Leap*.
- /Build folder contains the build files of *Take a Leap*.
- /Testing folder contains the questions and answers that the testers completed after testing the game.

V. License

I, Grete Hollo,

1. grant the University of Tartu a free permit (non-exclusive license) to

reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis

Take A Leap – Hyper-Casual Mobile Game

supervised by Daniel Nael.

2. I grant the University of Tartu a permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons license CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in points 1 and 2.

4. I confirm that granting the non-exclusive license does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Grete Hollo

08/05/2023