UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Kaarel Rüüsak

Bomb Golf – A Precision-Based Mobile Game

Bachelor's Thesis (9 ECTS credits)

Supervisor: Raimond-Hendrik Tunnel, MSc

Tartu 2020

Bomb Golf – A Precision-Based Mobile Game

Abstract:

This thesis is about creating a casual mobile game. There have been very few examples of good physics-based 3D mobile games that reward the accuracy and timing of the player. This thesis will help fill that gap by creating a game that has a unique combination of game mechanics. Most of the game was made by a team of two people, the author of the thesis focuses on the front-end development of the game. In the scope of this thesis, the game was tested at an expo. Based on the results of this test, the game was improved. Later, the game was uploaded on the Google Play store and tested online. The data gathered from the second test was used to propose future improvements for the development of the game.

Keywords:

Mobile game, mobile game development, mobile game design, puzzle game, arcade game, 3D, mobile, physics-based game, Unity, Android, Google Play, usability testing

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

Bomb Golf – Täpsuspõhine Mobiilimäng

Lühikokkuvõte:

Selles bakalaureusetöös on käsitletud mobiilimängu *Bomb Golf* loomist. Väga vähesed arendajad on loonud häid 3D mobiilimänge, mis põhinevad füüsika simulatsiooni kasutamisel ning tasustavad mängija täpsust ning ajastust. Töö aitab täita seda auku turul, luues mängu, millel on unikaalne mängumehaanikate kombinatsioon. Selle bakalaureusetöö käigus testiti mängu avalikul näitusel, et leida probleeme mängu disainis ning mängu uuendada. Hiljem laeti see mäng Google Play poodi üles ning seda testiti üle veebi. Teisest testist kogutud andmeid kasutati selleks, et pakkuda välja ettepanekuid tulevikus mängu edasi arendamiseks. Enamus mängu arendust toimus kaheliikmelises meeskonnas.

Võtmesõnad:

Mobiilimäng, mobiilimängude arendus, mobiilimängude disain, mõistatusmäng, arkaadmäng, 3D, mobiilne, füüsika-põhine mäng, Unity, Android, Google Play

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimis-teooria)

Table of Contents

1 Introduction

Video games are one of the newest forms of media. The video game market is quickly growing, and mobile games are at the forefront of this growth [1]. This kind of growing market supports new developers and even Estonian businesses have been successful in this field recently [2]. That means the development of mobile games can be profitable, and Estonia is a suitable place to start.

Mobile games differ quite drastically from the rest of the video game market. Adler *et al.* [3] found in their work that users prefer the use of mobile phones due to ease-of-use and comfort. The work also discovered that mobile users perform poorly at video games compared to desktop computer users. From this, we can deduce that mobile games have more potential users, but good mobile games should be less difficult than games on desktop computers. The market for mobile games is described further in Chapter 3.

In this thesis a mobile game called *Bomb Golf* was created. The game is based on timing and aiming bombs to destroy boxes. *Bomb Golf* is intended to be a game that could be popular in the mobile game market and can be used as a basis for developing a commercial product. Chapter 6 deals with testing and validating whether or not the created game achieves these goals.

Most of the development process was done in a team of two people, one of whom is the author of this thesis. The main responsibility of the thesis' author in the team was creating the visuals and front-end systems. However, after the first round of testing concluded, the second member of the team left. Beyond that point, all development of *Bomb Golf* is conducted only by the author of this thesis. The game is created using the Unity game engine. This choice is analyzed and explained in Chapter 2.

The gameplay of *Bomb Golf* consists of trying to progress from level to level by destroying all the boxes in the level. The player is limited by the number of bombs they have for completing the levels, which raises the stakes and motivates players to strategize. The players also gain gems by destroying boxes, which they can use to directly or indirectly purchase more bombs. Chapters 4 and 5 provide a more detailed overview of the game's design.

The Appendix includes an overview of the Accompanying Files, some early Development Screenshots and a Glossary of terms. Screenshots of the game at the end of the work for the current thesis are displayed in Figures 1 and 2. *Bomb Golf* is available at the following link: https://play.google.com/store/apps/details?id=com.bombgolf.bombgolf.

Figure 1. Screenshot of the first level in *Bomb Golf*

Figure 2. Screenshot of the factory in *Bomb Golf*

2 Technologies Used

In software development, it is important to pick the tools carefully in order to avoid wasting time, effort or money. For this reason, the software chosen to be used for development and some notable alternatives are described in this chapter. Subchapter 2.1 explains the choices and alternatives for various graphics software. Subchapter 2.2 explains what a game engine is and why we decided to use one. In Subchapters 2.2.1 to 2.2.3 various game engines are compared and one is chosen to be used for the final product.

2.1 Graphics Software

The 3D computer graphics software Blender[1] was used to create all the 3D graphics required for the game. It was chosen over other 3D modelling software like Maya or 3DS Max because we were experienced with the software and it was free for commercial use. The version of the game created within the scope of the thesis was not monetized in any way, but since the goal was to use it as a basis for creating a commercial product, we wanted to avoid using software that required a costly license for commercial use, such as Maya or 3DS Max.

The vector graphics editor Inkscape[2] was used to create the 2D vector graphics for the game and the raster graphics editor GIMP[3] was used for editing the raster images. Much like Blender, these image editors were chosen because the author of the thesis was experienced with them. The alternative tools for image editing would have been Adobe Photoshop and Adobe Illustrator. Each of these Adobe tools cost 23.99€[4] every month for normal use or sixty percent less for students, which was another reason why the development team decided against using them. Inkscape and GIMP are both free for commercial use.

However, as professional artist Saporito [4] has stated, Adobe Illustrator provides a more efficient workflow than Inkscape for creating logos and icons. Furthermore, compared to GIMP, Photoshop has more features and is more powerful[5]. Because of this, Adobe tools could have helped the development team work faster and more efficiently.

---

[1] https://www.blender.org/
[2] https://inkscape.org/learn/faq/
[3] https://www.gimp.org/docs/userfaq.html
[4] https://www.adobe.com/ee/creativecloud/
[5] https://expertphotography.com/gimp-vs-photoshop/

Considering the price of using both Adobe tools (19.2 € every month including the 60% student discount) and adding up the costs of using them throughout the entire development period (11 months), this software would cost the team 211.2 €. For comparison, the net revenue per download for the Google Play Store in the first quarter of 2018 was 0.10 $[6] or 0.09 €. This means that in order for the Adobe software to be the better choice, their use should help us reach over 2346 additional users. This did not seem like a reasonable assumption, which is why we decided to use GIMP and Inkscape for the development of *Bomb Golf*.

2.2 Game Engines

A game engine[7] is a tool that automates much of the work required to create a video game by making the process more abstract. This can allow developers to make games quickly without the need to explicitly use libraries or write complex code for physics, rendering and input. Instead, game engines provide visual tools and a runtime for faster and easier development. While it is possible to make a game without the use of game engines, in order to speed up the process of game development, our team decided to use one.

2.2.1 Unreal Engine

Unreal Engine[8] is a video game engine that can be used for many platforms. In addition, it offers a lot of rendering tools and options for the developers. However, most of these tools are intended for graphically intensive games with high performance requirements. Because the goal for this work is to create a game that is competitive on the mobile game market, these additional options are not useful for the development of the game.

There are also some problems with transferring games from Unreal Engine to the Android OS and compressing the game, which can cause performance problems on devices with a low amount of memory [5]. Because of this and the development team's lack of experience with creating mobile games using this software, Unreal Engine was not chosen for use in the creation of the game.

[6] https://www.ped30.com/2018/05/28/apple-app-store-clobbers-google-play-now-ever/
[7] https://www.gamecareerguide.com/features/529/what_is_a_game_.php
[8] https://www.unrealengine.com/en-US/features

2.2.2 Playcanvas and Facebook Instant Games

Playcanvas[9] is a relatively new game engine, which is created for the development of 3D browser games that are powered by HTML5. Due to this HTML5 support, it is easy to play games created in Playcanvas through a web browser and it is also possible to port these games over to other mobile platforms using free tools[10]. Video games developed with Playcanvas can also be published on the Facebook Instant Games store.

Facebook Instant Games[11] is an application store, which opened for public use in March of 2018. The store is integrated into the popular social networking service Facebook and all of the games there are playable both through the web browser or using a mobile device. Compared to older internet stores (e.g. Google Play), less mobile game developers have had the chance to develop games for Facebook, which is why the competition on this platform is lower. That is why development for this platform seemed like a good opportunity.

Because the Playcanvas engine seemed interesting, the initial prototype of *Bomb Golf* (displayed in Figure 30 in the Appendix) was created using Playcanvas. During the creation of this thesis, Facebook closed their platform to games which are not published by officially registered businesses. This was a problem for our team, because we wanted to test our game on a wide audience before committing to registering a co-owned company. Also, the development was slower while using this engine due to a lack of community support compared to competitors (e.g. Unity). For these reasons, we decided to use a different game engine.

2.2.3 Unity

Unity[12] is a versatile video game engine, which can be used to develop games for over 15 platforms, including mobile. Because of that variety, games made in Unity can be published on many digital distribution platforms, like the Google Play Store, the iOS App Store and web browsers (through HTML5). Unfortunately, Unity games compiled in the HTML5 language do not support mobile devices, which is required to publish games on the Facebook Instant Games platform.

---

[9] https://blog.playcanvas.com/faq/
[10] https://cordova.apache.org/
[11] https://developers.facebook.com/blog/post/2018/03/14/instant-games-platform-open/
[12] https://unity3d.com/unity/features/multiplatform

Unity was one of the first game engines to implement a marketplace to sell or publish different tools created by their community. Because of that, Unity has a wide library of tools available, many of them free for commercial use. Also, the development of mobile games for the Android OS is simpler and easier with Unity than with some of their competitors [5]. This is useful, because *Bomb Golf* is developed primarily with Android devices in mind, since Android smartphones currently have the most market share compared to any competitors[13].

For the above reasons and because access to the Facebook Instant Games platform is now somewhat limited, the development team decided to create the game in the Unity engine. In addition, both of the developers were experienced with this game engine.

[13] https://www.idc.com/promo/smartphone-market-share/os

3 Competitors

Market research is very important when developing a game, because it allows the developer to find the overlap between what the players want to play and what the developers want to create [6]. This chapter analyses a few mobile games similar to *Bomb Golf*, in order to draw inspiration from them and to make sure *Bomb Golf* stands out from the competition.

3.1 Angry Birds 2

Angry Birds 2 (2015) is a puzzle game created for the Android and iOS platforms by *Rovio Entertainment Corporation*. It belongs to a series of games with very similar gameplay. The *Angry Birds* series was the first mobile game series to reach a billion downloads [7].

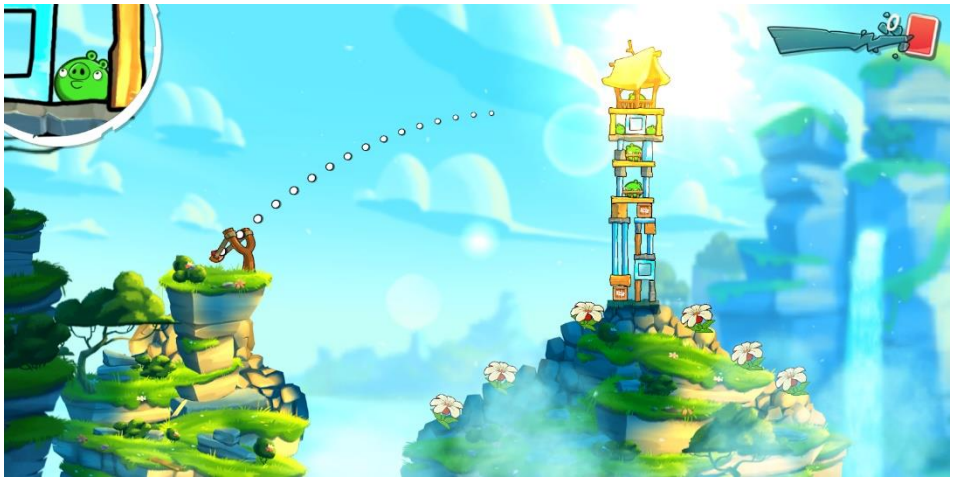

Figure 3. Screenshot from *Angry Birds 2*

The player's goal in *Angry Birds 2* is to destroy all the pigs on different structures in the level using a slingshot (see Figure 3). This means that the player usually needs to aim at the structure's weak points to make it collapse onto the pigs. Alternatively, the player can try shooting the pigs directly.

In some ways, the game mechanics of *Angry Birds 2* are very similar to *Bomb Golf*. Just like in *Bomb Golf*, the win condition of each level is achieved when the player destroys all the targets. Both games restrict the number of projectiles the player can use, and *Angry Birds 2* even has an exploding version of the usable projectile.

However, the gameplay of *Bomb Golf* differs significantly from *Angry Birds 2*. In *Bomb Golf*, the player aims projectiles horizontally and the levels are three-dimensional, whereas *Angry Birds* is a two-dimensional game viewed from the side. The aesthetics of *Angry Birds 2* are also detailed, cartoonish and use saturated colors, unlike what we wanted to design in *Bomb Golf*.

3.2 Boom Golf Park: 3D Bomber Mini Golf Fun Game

Boom Golf Park: 3D Bomber Mini Golf Fun Game or just *Boom Golf* is a game on the Google Play Store with approximately 501 to 1000 downloads[14]. It was last updated on the first of October, 2019. The game was developed by Space Tech Gamers.

The reception for *Boom Golf* is not very positive, however. The game has one review, which gives the game a score of 1/5 stars. This review critiques the game's aiming system, which uses a slider to select the speed at which the projectile's direction is changed. This system can cause players to easily overshoot or undershoot their intended angle, which can be irritating.

Steve Swink [8] explained in his book "Game Feel", that a similar indirect aiming system was not annoying when controlling the "warthogs" in the game *Halo*, because it created an interesting challenge and a sensation of control for the player. However, in *Boom Golf*, the aiming system does not provide enough feedback to create such a sensation. That is because instead of any tangible objects, it only rotates non-diegetic arrows next to the projectile. In addition, because *Boom Golf* has static targets and the players have a lot of time to aim, it does not create an interesting challenge either. For these reasons, *Boom Golf*'s control scheme is not suitable for a golf-based game, which is why we used a different control scheme for *Bomb Golf*.

The main goal of *Boom Golf* is to hit the ball until it reaches the hole. The player has infinite turns, but if they are not fast enough, the ball will explode, and the player will lose. As shown in Figures 4 and 5, the timer for the bomb is displayed on the top-left corner of the screen as a burning fuse.

[14] https://play.google.com/store/apps/details?id=com.spacetechgamers.boom.golf.park.bomber.mini.golf.fun.game

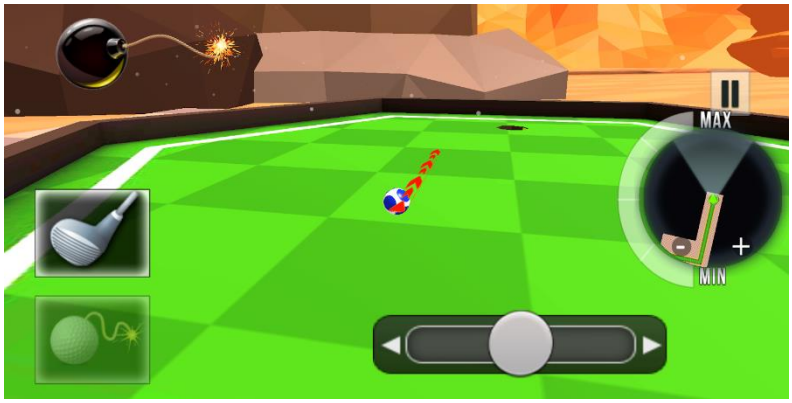

Figure 4. Screenshot of *Boom Golf*

Figure 5. Screenshot of *Boom Golf*

The greatest similarities between *Boom Golf* and *Bomb Golf* are in the theme of the two games. Both games allow the player to launch explosive projectiles and both are in 3D. However, the aesthetics of *Boom Golf* are much more colorful and saturated than *Bomb Golf*. Unlike *Bomb Golf*, *Boom Golf* also uses some detailed image textures in its levels.

Like most golf games, *Boom Golf* rewards the player's precision. Aside from this, however, its game mechanics are very different from *Bomb Golf*, which are described in detail in the next chapter. In *Boom Golf*, the controls for choosing the velocity of the projectile are separated from its direction and based on the player's timing, but that is not the case in *Bomb Golf*. Also, the goal in *Boom Golf* is to reach the hole with a single projectile, but the win condition in *Bomb Golf* is based on destroying multiple targets using a limited number of projectiles (usually more than one).

4 The Game Design

As Jesse Schell [9] explains in his book, "The Art of Game Design: A Book of Lenses", game design is a vast field, which focuses on the decisions needed to determine what a game should be. According to him, a game consists of mechanics, aesthetics, story, and technology, which all contribute to the final design of the game. This chapter describes the various parts of game design for *Bomb Golf*. The technology of the game is discussed in Chapter 2. The game's mechanics and some of the related aesthetics are discussed in Subchapter 4.1. The wider aesthetic style of the game is explained in Subchapter 4.2. The design of specific levels is covered in the next chapter, which is why it is not discussed in this chapter.

4.1 Game Mechanics

Game mechanics are the underlying rules and relationships which games consist of. They are the part of a game which cannot be categorized as its aesthetics, story or technology [9]. In this chapter, such parts of the game are explained. An overview of these mechanics in the final build of the game can be found in Figure 29 in the Appendix.

4.1.1 The Bomb and Timer

According to Schell [9], the skills required of the player are a very important part of determining the player's experience. One of such core skills in *Bomb Golf* is timing the bombs, which is why it is crucial for the bomb's timer to be as simple and convenient to the player as possible.

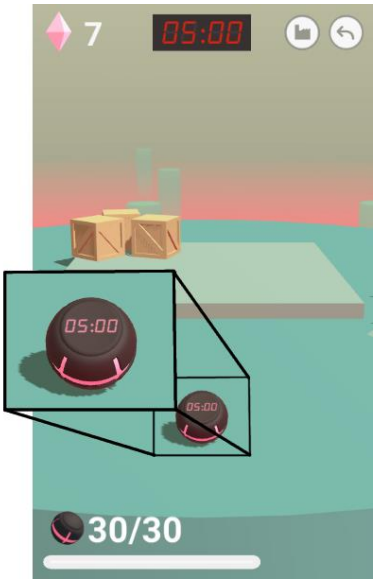

Figure 6. The timer

Bomb Golf displays the timer in two locations: on the two-dimensional user interface and as a part of the bomb. This may seem redundant but having two timers is designed to solve many problems. For new players, the timer on the bomb could help them associate the two and figure out the timer's purpose. This design also draws the player's attention towards the timer, letting them know it is important. However, the bomb's timer is very small, and it can be difficult to read on some phones. Because of this, the game also has a large non-diegetic timer as part of the user interface. The two timers are visually identical so that players know they are used for the same purpose (see Figure 6).

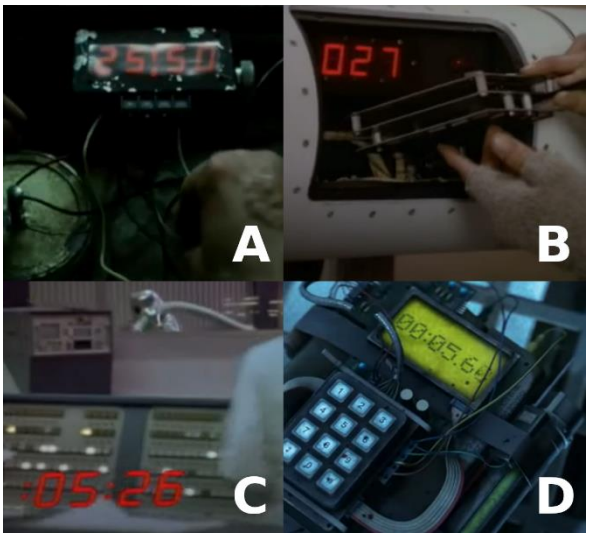

Figure 7. Digital bomb timers from media: Fight Club (A), MacGyver (B, C) and a trailer for Counter Strike (D)

To make the game easier to understand, the development team wanted the aesthetics of the timers in *Bomb Golf* to be easily recognized by the players and associated with bombs. For this reason, we took inspiration from popular media such as Fight Club, MacGyver and *Counter Strike*, all of which use digital clock displays to display remaining time on armed bombs (see Figure 7). The timer in *Bomb Golf* looks like the screen of a red digital alarm clock (displayed in Figure 6). In previous iterations, the team also experimented with some borders consisting of wires and a case around the display of the timer (displayed in Figure 31 in the Appendix). However, this border turned out to be too large to fit with the rest of the GUI and it seemed too detailed for the art style of the game.

4.1.2 The Arrow

The arrow (see Figure 8) is a vital element in *Bomb Golf*, because it provides the player with a visual indicator for what velocity and direction the bomb will be launched at. The camera rotates along with the arrow in order to give a better understanding of the world and the launch direction to the player. When playing the game, the player can hold and drag their finger on the screen in order to resize the arrow and/or change the direction of the arrow and the camera. This control scheme was chosen, because we hoped it would be familiar to people who had played games like *Angry Birds*, as it follows a similar logic. In both games, the players need to pull their finger back in order to launch the projectile forward, which is indicated by a partial visual representation of the motion the projectile would follow upon release.

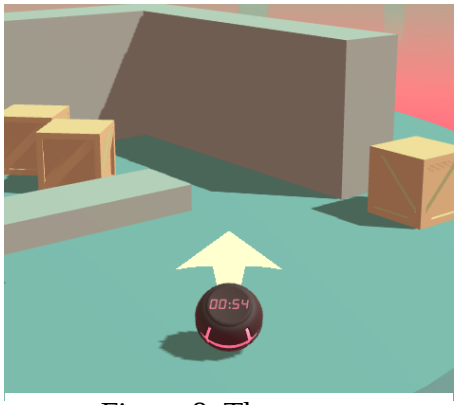

Figure 8. The arrow

4.1.3 Targets and Carbide

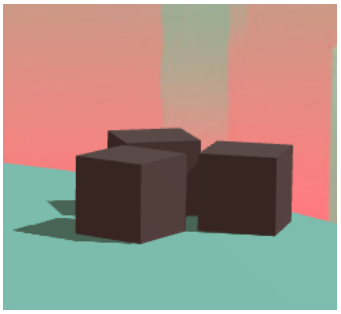

Figure 9. The targets

In order to progress through the levels of *Bomb Golf*, the player needs to destroy all the targets in the current level. When no targets remain, the player must move to the next level to continue playing. The only way to destroy these targets is to make a bomb explode near them. Aesthetically, these targets look like dark cubes (see Figure 9) and upon destruction they explode into 64 identical dark cubes before vanishing. These visuals were changed after the first testing phase (described in Chapter 6.1.3).

In order to reward players and provide them with a greater sense of progression, we also implemented carbide, which is a resource that looks like pink cubes in early versions of the game. This can be seen in the upper-left corner of Figure 32 or 33 in the Appendix. In later versions of the game, carbide was changed to look like pink diamonds (see Figure 10).

Players can acquire carbide by blowing up targets in the levels and they can use it to upgrade their factory (explained in Subchapter 4.1.3) or buy bombs to prolong their game sessions. As Schell [9] has mentioned, rewards which extend playtime are effective, because they tap into the natural human drive for survival.

The number of currently available carbide is always displayed in the upper-left corner of the screen. In order to make the origin of carbide clearer to players, each time a target is destroyed, it reveals a 3D representation of carbide. The carbide then moves towards the player's starting position before sinking into a hole in the ground (see Figure 10). At the end of that animation, the currently available carbide is increased appropriately.

Figure 10. Carbide animation

4.1.4 The Factory

Early on in the game's development process, every player had five bombs to beat each level and they could try again infinitely. This approach provided little punishment for losing and few rewards for succeeding, so players did not have much incentive to keep playing. As Salen *et al.* [10] mentioned in their book "Rules of Play - Game Design Fundamentals", punishments are important and rewards need to be balanced along with punishments in order to make the play experience pleasurable.

Additionally, it is important to keep the players coming back to a game. Industry experts generally agree that well-integrated energy systems which limit player progress can boost user retention [11]. The definition for retention can be found in the Glossary (Appendix I).

For these reasons, we made the bombs into a limited resource which is shared across all levels. When the players run out of bombs, they need to wait for more to be generated by an in-game factory, or they can buy bombs directly for the collected carbide. The time remaining until the next bomb is generated is displayed to the player as a bar filling up on the bottom of their screen. When the bar is full, a new bomb is made.

The bombs are created by the factory even while the player is outside the game, which incentivizes players to pace themselves when playing the game. Players can also purchase upgrades to the factory, which can either increase the speed of bomb generation or the maximum number of bombs stored by the factory. The menu for the factory (displayed in Figure 33 in the Appendix) also contains a visual representation of the factory, which was planned to have noticeable changes for each upgrade. However, due to time constraints, these changes were not implemented in this thesis.

The visual representation of the factory consists of a machine emitting smoke through a chimney. This factory has a pipe leading out through its front and into a bomb pen which functions much like a cattle pen. For each remaining bomb the player has, a bomb is rolling around in these pens (see Figure 11). This suggests to the players that all the bombs that are not in use are kept in the pens.

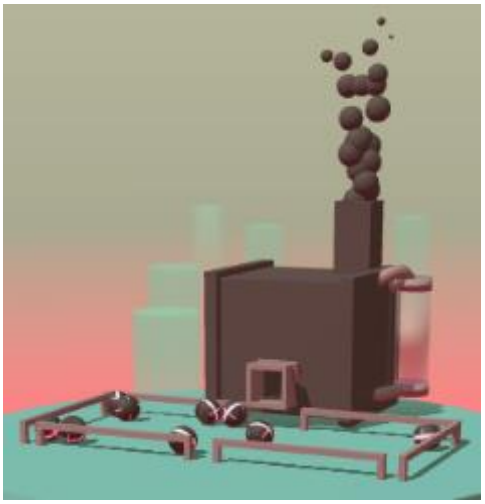

Figure 11. The factory

4.1.5 Obstacles

The obstacles are an essential part of the challenge in *Bomb Golf*. These are almost always in the shape of cuboids and the players are unable to move them. In order for the players to distinguish them from the targets, the obstacles use bright colors, much like the floor. Some obstacles have been tilted to create ramps so the players could use them to cross gaps.

Additionally, later levels include orange obstacles, which have altered physics properties. When a bomb touches one of these obstacles, it maintains most of its velocity. These obstacles are used to increase the variety in the game's levels and create different challenges.

4.2 Aesthetics

Figure 12. Race The Sun

A very important factor we took into consideration when deciding upon the general art style of the game was how it would affect the game's file size. Reducing the size of applications is crucial in order to attract users who have poor or expensive internet connection [12]. As high-quality image textures increase the application's file size drastically, we wanted to pick an art style that would not rely on such textures. Additionally, using smaller image textures in the application would benefit us by improving the game's CPU performance[15].

For this reason, we decided to use an artistic style for the game which utilizes simple shapes and solid pastel colors. For this, we took some inspiration from *Race The Sun*, a PC game from 2013 which similarly uses solid desaturated surfaces and simple abstract shapes (see Figure 12).

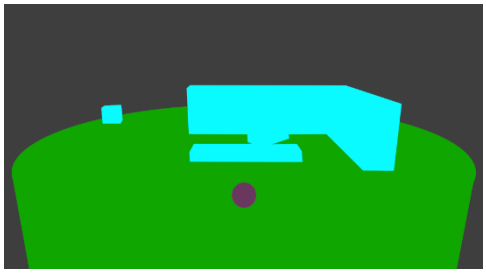
Figure 13. Shapes in the game's composition

Additionally, the choice of shapes in the game creates a visual narrative through dynamic composition. These shapes are illustrated in Figure 13. Both the bombs (purple) and the floor (green) are round, whereas the targets and most environmental objects (blue) are cuboids with sharp silhouettes. This difference creates dissonance between the player character and the environment, which suggests that the bomb (and by extension, the player) is being threatened or blocked by the obstacles and targets [13]. This helps players intuitively figure out that the bomb is supposed to overcome or destroy the cuboid objects in the environment.

[15] https://docs.unity3d.com/Manual/OptimizingGraphicsPerformance.html

5 The Levels

A level is the space in which the player needs to overcome a challenge or series of challenges [9]. By breaking gameplay down into levels and examining them separately, it is possible to see the smaller and more intricate parts of the game's design. This chapter is dedicated to analyzing and adjusting the game design in the levels of *Bomb Golf*. Subchapters 5.1 to 5.5 discuss and explain the design behind some of the game's levels in the order they appear in the game.

5.1 Level 1

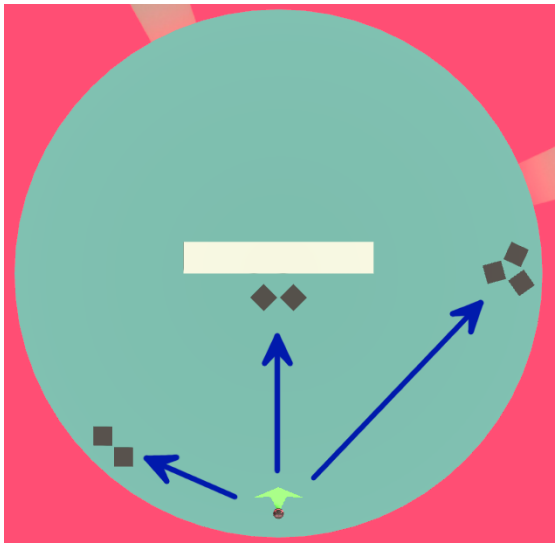

Figure 14. Level 1

The first level is a very important part of any game, as it creates the first impressions for the player and teaches them how to play. To avoid frustrating the player, this level should be very easy [9]. As shown in Figure 14, in *Bomb Golf*, the player does not need to overcome any obstacles to reach the targets in the first level. The green arrow at the bottom of Figure 14 represents the bomb's aiming indicator and the blue arrows indicate the intended routes for the bomb.

When first entering this level, new players are probably wondering how they can progress in the game. This simple level gives players an opportunity to try out the controls and figure out that the targets should be destroyed (there is little else in the level). Additionally, there is an obstacle near some of the targets, which teaches the players that the obstacles can not be destroyed.

5.2 Level 2

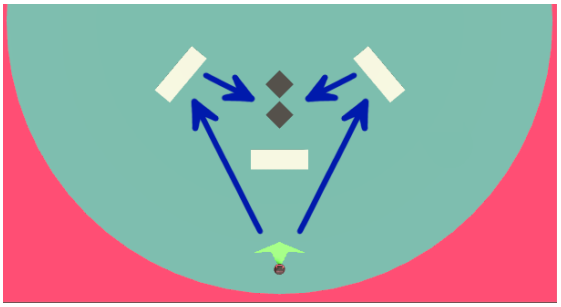

Figure 15. Level 2

As Dan Taylor [14] has mentioned in his 2013 article, good level design constantly teaches players something new. As discussed in the Introduction, we know that good mobile games should not be very difficult. For these reasons, level 2 was created to teach players a simple skill vital for completing most of the other levels in *Bomb Golf*: the use of the environment to bounce the bomb towards the targets. As shown in Figure 15, the players are intended to use the walls to the left or to the right of the level in order to bounce the bomb into the targets.

This level is quite forgiving to the players in terms of timing. If the bomb explodes too late, it is still likely to destroy the targets, as after colliding with the target the bomb has lost most of its velocity. This gives the players more room for error, which is useful in early levels such as this, because they can get used to the controls without being punished. That kind of design helps keep the players from feeling too anxious while playing [9].

5.3 Level 3 and Level 4

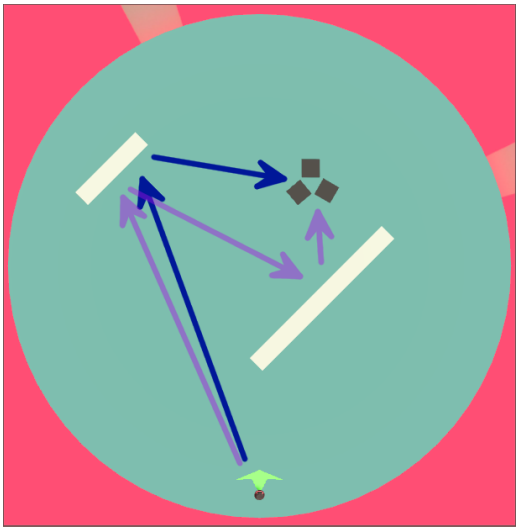

Figure 16. Level 4

Level 4 was created as an opportunity for players to get used to bouncing the bomb across long distances with precision (see Figure 16). The player is intended to use the wall at the back of the level to bounce the bomb into the targets, as illustrated in dark blue. The purple arrows indicate an alternate path which can be used by more confident players who launch the bomb at a high velocity.

According to Todd Howard [15], the executive producer and game director at Bethesda, one way to entice players to continue playing is to use a gameplay loop that consists of the following elements: learn, play, challenge, surprise.

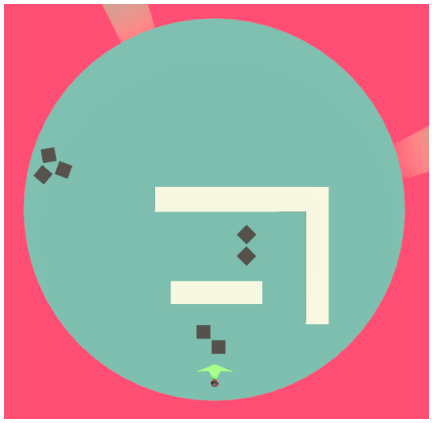

Figure 17. Level 3

Following this principle, the second level of *Bomb Golf* gives the players an opportunity to *learn* how to use their environment to bypass obstacles. Level 3 (see Figure 17) gives the player a more complex space, so they can *play* around with how the bomb bounces and level 4 *challenges* the players to destroy a distant target using their current knowledge. The last element of the gameplay loop, *surprise*, is addressed in the next subchapter which describes level 6.

5.4 Level 6

This level might be intimidating to new players at first glance. That is because it is the first level which blocks the direct path between the bomb and the targets while providing the player with no obvious method to get around this problem. As shown in Figure 18, the intended course of action for the player is to bounce the bomb against the back of the level twice and into the targets. In order to reach the targets using this method, players need to launch their bombs at a very high velocity.

Compared to other levels, level 6 does not require precise aim or timing and it takes more mental skill to solve than most other levels in *Bomb Golf*. This unexpected twist surprises the player, completing the game loop discussed in the previous subchapter. To reward players for finding the solution and overcoming this unusual challenge, both sides of the level have groups of three targets, which means the level contains a total of six carbide, an unusually high amount for a level that requires little mechanical skill.

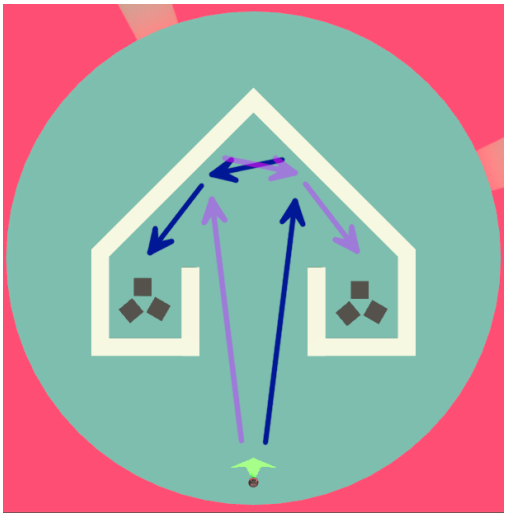

Figure 18. Level 6

5.5 Other Levels

Todd Howard's "learn, play, challenge, surprise" loop discussed in Subchapter 5.2 was used many times throughout the development of *Bomb Golf*'s levels. Another example of this is how ramps are used in the game. In level 5, players learn the basics of how to use ramps (see Figure 19). In level 7 and level 9, they have an opportunity to play around with various ramps at different angles. Level twelve is the next step in this loop, as it challenges the player to use a large ramp to destroy targets at different distances and heights.

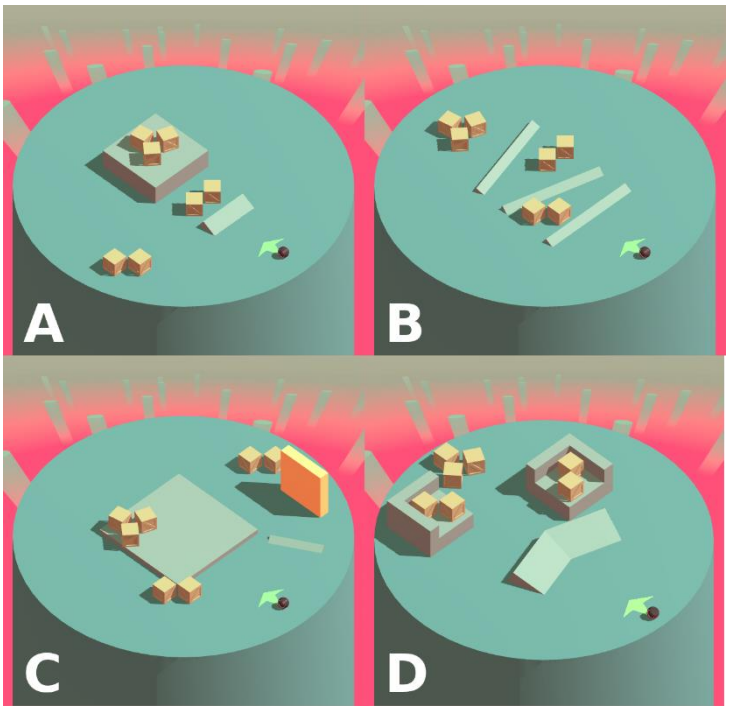

Figure 19. Levels using ramps:
Level 5 (A), Level 7 (B), Level 9 (C) and Level 12 (D)

Before the tests conducted at the Robotex 2019 expo (see Chapter 6.1 for details), the game only had 10 levels. During the online testing phase, 5 additional levels were created and added to the game, which means the final version of the game has 15 levels. The first 10 levels are designed to be completed in two to three days, whereas levels 11 to 15 would take an additional day or two to complete.

Additionally, challenge levels were added to the game after the Robotex expo, which gives players an opportunity to play any completed level with a limited number of bombs. Whenever a challenge level is completed for the first time, the player earns carbide. This gamemode was created so dedicated players can keep playing after they run out of bombs and earn some bonuses by doing so.

6 The Testing

Testing is an essential part of software development. When used well, usability testing creates data from potential users which can help identify problems in the game's design and reduce player frustration [16]. Because of this, the development team decided to first test the game at a public event. Subchapter 6.1 discusses these tests. Later in development, the game was released on Google Play for testing, which is discussed in Subchapter 6.2.

6.1 User Testing at the Robotex expo

Robotex International is an annual robotics festival held in Estonia[16]. The author of this thesis was invited to expo *Bomb Golf* at this event between 29.11 and 01.12 of 2019. This was a valuable opportunity for the team, because we were able to conduct live user testing on a wide audience. Along with *Bomb Golf*, games from other University of Tartu students were also on display[17] at the expo: *ShieldFormer*, *Planet Invasion* and *Warhammer Auto-Chess*.

The methods used for testing, the testing environment and other details about the testing itself are discussed in Subchapter 6.1.1. Based on the feedback acquired from the testing, a lot of conclusions were drawn, and the game was improved accordingly. This process is discussed in Subchapters 6.1.2 and 6.1.3.

6.1.1 Methodology

When *Bomb Golf* was tested at the Robotex 2019 expo, only the first ten levels had been created. Additionally, because some of the game's design does not work well in a public expo (such as bombs being generated over the course of hours), an expo build was created for *Bomb Golf*. An expo build[18] is a version of the game specifically made for an expo that highlights the most immersive parts of the game.

[16] https://robotex.international/
[17] https://cgvr.cs.ut.ee/wp/index.php/games-at-robotex-2019/
[18] https://cgvr.cs.ut.ee/wp/index.php/guide-to-expos/

The expo build for *Bomb Golf* has increased bomb generation speed and resets all player progress when the game is restarted. Up to two other games were shown alongside *Bomb Golf* while it was tested. Most of the testing was conducted in a small room (shown on Figure 20), but due to a lack of testers during the first day, some tests were also conducted at an open expo area. The testing equipment included two smartphones with *Bomb Golf* (LG Nexus 5 and Huawei Mate 20), a poster for the game, a stack of approximately 80 feedback forms, a few pens and a box which held the filled-out feedback forms. The feedback forms and poster can be found in the Accompanying Files (Appendix II).

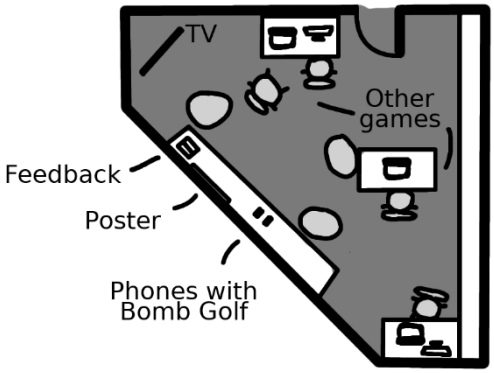

Figure 20. The room at Robotex 2019

The process of testing was simple. If a visitor showed interest in the game, the developer would give them a smartphone with a fresh version of the game and ask them to sit on a bean bag to play it. Beyond that, the developer would observe the visitor's progress and assist them only when it was absolutely necessary. This method was chosen, because any assistance offered by the developer would interfere with the results of the test [17]. When the visitor either completed the game or got tired of playing it, they were given a pen and an empty feedback form to fill out. Figure 21 depicts the final room used for most of the testing.

This method is not without its flaws. It is important to note that visitors choosing when to quit and whether to give feedback creates a survivorship bias in the test results. This means that the results contain less qualitative data about the users who were too tired or too bored to fill out the form.


Figure 21. Photo of the room at Robotex 2019 [18]

In order to partially counteract this problem and to get more feedback, the form was created to be as short and simple as possible. It contained only two short questions: “What was fun?” and “What to improve?” As these questions are very open-ended, the enthusiastic visitors who wanted to give more feedback still had the opportunity to do so.

6.1.2 Playtesting

During the event, a total of 39 responses were collected from visitors and approximately 45 people were observed playing the game. Due to the open-endedness of the questions in the feedback form, this feedback was really varied, but some similarities did occur. According to 6 people (15.8% of respondents), everything was fun and 8 people (21%) figured that there is nothing left in the game to improve. These respondents most likely enjoyed the game but were too tired or lazy to give thorough feedback (see Figure 22).

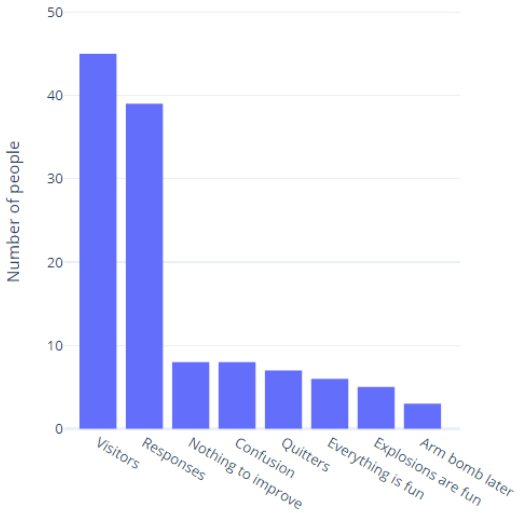

Figure 22. Feedback from Robotex 2019

According to 5 people (13.2%) blowing up the targets is fun. This is a good sign, because destroying targets is intended to be one of the most rewarding parts of the game. In addition, 3 people (7.9%) mentioned that they particularly enjoyed how the game challenges the player’s precision, which is also one of the intended results.

A total of 8 people (21%) mentioned being confused by either the rules of the game (the game design), the general graphical user interface (GUI) or the factory GUI. In addition, approximately 60% of the players who tried to use the factory were observed having difficulties with figuring out what part of the GUI is a button. This feedback was noted, and potential solutions will be discussed further in Subchapter 6.1.3.

Some of the visitors recognized pacing problems with the game. A total of 3 people (7.9% of respondents) mentioned that they would like to have some time to look around before the bomb is armed and the timer starts ticking down. Previously, we had thought that players would use the main menu to observe the level before playing, but this mechanic was rarely used, as players preferred to examine the level while zoomed in.

Only one visitor realized that they could turn their phone 90 degrees to make the game switch between landscape and portrait mode. All the menus and controls are functional both in landscape and portrait mode, though the game is designed foremost with portrait mode in mind. Every other player simply continued playing the game in whichever position the device was given to them. Approximately 65% of the visitors played the game in portrait mode.

One player mentioned while playing the game that they do not read the timer on the bomb, because it is too small. This is a small disadvantage, as using the bomb's timer allows players to aim while seeing the remaining time. Additionally, 2 players did not notice that the bomb even had a diegetic timer on it until it was pointed out to them.

In the feedback forms, most visitors briefly mentioned problems important to them or solutions that the team did not agree with. However, there were 2 clever and creative visitors who offered ideas for improvements, which the development team had not thought of, but realized they needed. These ideas and their implementation is discussed further in the next subchapter.

6.1.3 Improvements to the Game

After the expo, the development team immediately got to work on improving the game based on the feedback. One of the first improvements was to the interface of the factory. A very prominent problem found through testing was the confusing factory GUI, so it was completely revamped. Additionally, the size of the visualization for the factory was increased to look more appealing, which can be seen in Figure 23.

In an attempt to keep the user interface as simple as possible, before the expo, the game only displayed the maximum number of bombs at the factory. However, this confused players who did not completely understand the factory, which is why it was changed to be always visible on the bomb counter. This change is also displayed on the bottom of Figure 23.

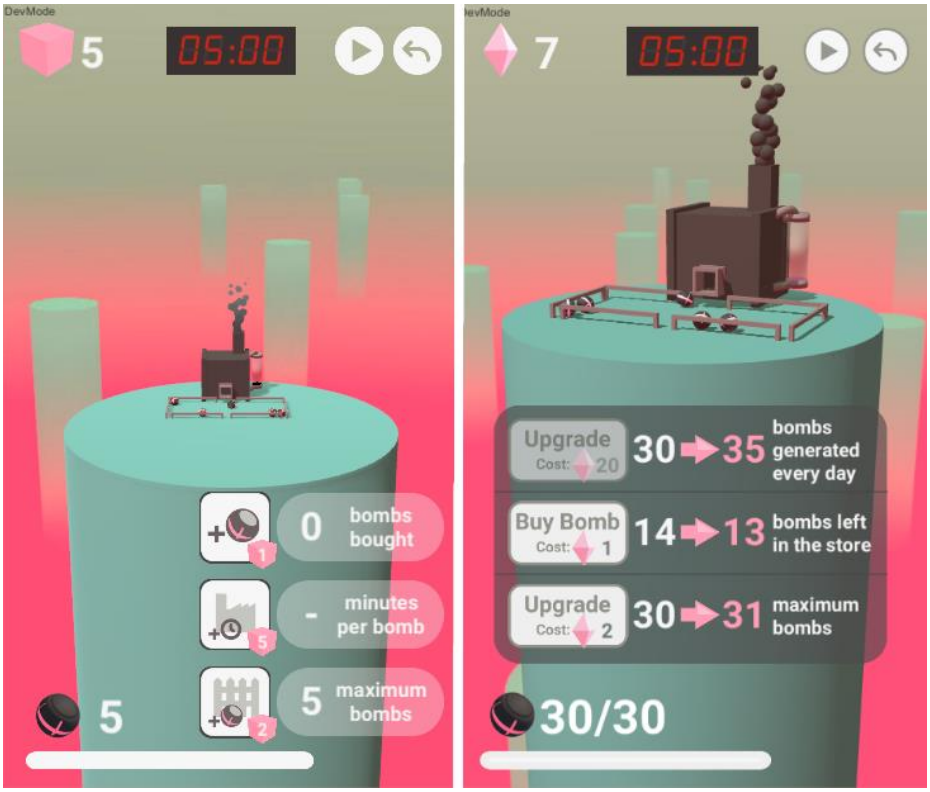

Figure 23. Old factory GUI (left) compared to the new factory GUI (right)

Because we wanted the game to be easy to try out, we were initially hesitant to implement tutorials. However, after some debate, we decided to implement a very short non-intrusive tutorial about the controls of the game in order to make it easier to understand. This tutorial is displayed only on the beginning of the first level to new players. It consists of an animated hand swiping down and aiming the bomb. The tutorial helps new players understand how they are supposed to play the game and it disappears after the player launches their first bomb.

Some players wanted a period of time before the bomb is launched to prepare and observe their surroundings without the bomb blowing up. Because of this, the timer for bombs no longer automatically starts ticking down when players enter the level. Instead, it starts ticking down the moment the player touches the screen while in the level.

When obstacles were first created for the game, they were white, because it fit the rest of the colors in the level. However, during live testing at Robotex, a few players said they were confused about the function of the obstacles. They did not know if they needed to blow the obstacles up along with the targets. One player even suggested a solution to this, which was to tint the obstacles the same color as the ground. Eventually, this solution was implemented, and the obstacles were tinted green to indicate they are functionally similar to the ground. (see Figure 24).

For similar reasons, targets were also tinted yellow and additional details were added to each side of the targets to make them look like wooden crates. This provides players with an explanation for why the targets contain carbides, which could help new players understand the game. This change can also be seen in Figure 24.

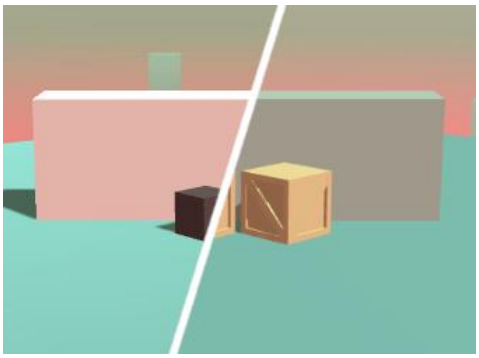

Figure 24. Old level colors (left) compared to new colors (right)

The game was very difficult and only a select few people were able to beat it. This was a problem, because as mentioned in the Introduction, good mobile games should not be very difficult. In addition to this, a couple of players had difficulty with seeing the small timer on the bomb. Because of this, bombs were made significantly larger. In order to not drastically alter the feel of the game, the explosion size and the scale of targets also had to be increased. The change in target size is illustrated in Figure 24. Because of this change, every level in the game is now less challenging.

One person mentioned in both verbal feedback and in their feedback form that it was difficult to see the diegetic bomb timer when shooting the bomb at an angle. This was because the bomb didn't rotate along with the camera when aiming. The change was quickly implemented and now the bomb rotates with the camera, so the timer is always facing the player.

6.2 Online Testing

After the testing of *Bomb Golf* at the Robotex 2019 expo and four additional months of development, the team wanted to test the game on a wider audience. For this reason, the game was released on Google Play in early access. It was free to download and available for all regions. Along with this version of the game, user testing forms were sent out to the public. These forms and the responses are covered in Subchapter 6.2.1. In addition to this, the version of *Bomb Golf* uploaded on Google Play automatically uploads retention metrics to the internet. These metrics are discussed and analyzed in Subchapter 6.2.2.

6.2.1 Feedback Form

The feedback form used in this test was created using Google Forms and distributed through e-mails and various text-based communication platforms like Discord or Slack. In additon, the form was sent to the APT GameGenerator[19] game developer community through social media. This form contains a link to the Google Play Store page for *Bomb Golf*, which players could use to download and install the application. The full form can be found in the Accompanying Files (Appendix II).

To make reliable comparisons with the tests at Robotex and give enthusiastic players an opportunity to express their opinions in detail, this feedback form contains two questions similar to the ones asked from visitors at Robotex: “What did you like about it?” and “What should we improve?”

Additionally, the form contains five more questions about user preferences, playtime, progress and satisfaction with the game. These questions were added to measure some components of usability which were covered by observation and verbal feedback at Robotex [17]. In addition, the form contains two more optional questions, where players can provide their email address or any extra comments they might have. These questions were added as an opportunity for interested players to provide additional details about the game or any problems they might have encountered.

[19] https://aptgg.ee/

6.2.1.1 Form Results

A total of 10 responses were collected over the course of two weeks. While this is a lot less feedback than was gathered at Robotex, the quality of the online feedback is significantly higher. Most of the respondents wrote down detailed comments about the game and suggestions for improvements. None of the respondents claimed there was nothing to improve or that everything is fun. Additionally, far more questions were asked from the respondents, which means more data was gathered.

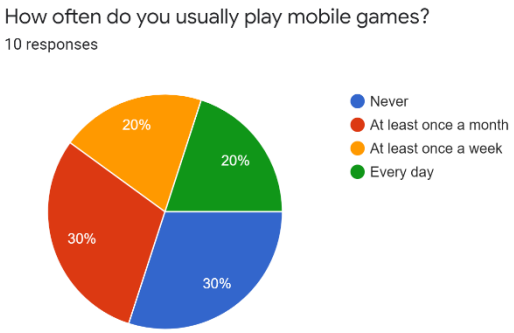

Figure 25. How often respondents play mobile games.

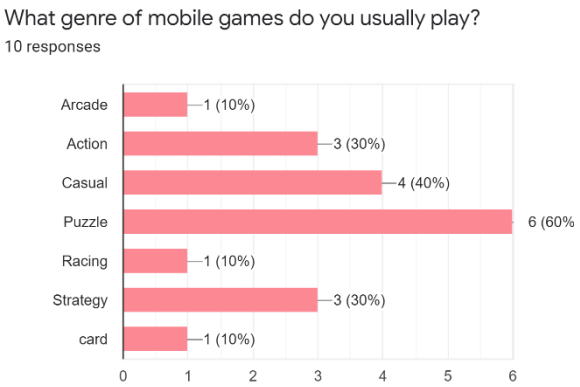

Figure 26. Respondents' preferred mobile game genres.

Of all respondents, 70% play mobile games at least once a month and 40% play at least once a week (see Figure 25). This is a good sign, because it means that if respondents enjoyed *Bomb Golf*, they might play it regularly. Additionally, three of the most popular mobile game genres among respondents were "Action" "Casual" and "Puzzle", all of which could be used to describe *Bomb Golf*. The respondents' preferred genres can be seen on Figure 26.

A figure that significantly stands out from the respondents' suggestions for improving the game is that 7 people (70% of respondents) found the aiming system for the bombs to be problematic. Most of these respondents simply said that the aiming was too sensitive, but one also mentioned that it is not fast enough to turn 180 degrees. Of these respondents, 3 wrote that they would like to have the ability to rotate the camera without aiming or arming the bomb.

Three respondents also mentioned their desire for additional tutorials. Specifically, one user felt dismayed, because they had wasted a lot of bombs in the first level while trying to figure out the controls. Solutions to this problem will be discussed in the next subchapter.

The average respondent beat 5.6 levels and spent 9.85 minutes in the game before answering the feedback form. All 15 levels were beaten by 2 respondents. Both of the respondents who beat the game had submitted two responses to the feedback form. In these cases, the second response was removed from the statistics and analyzed separately.

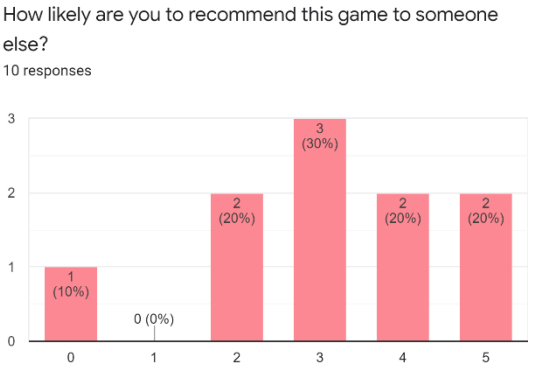

Figure 27. User satisfaction

On a scale from 0 (very unlikely) to 5 (very likely), the average response to the question "How likely are you to recommend this game to someone else" was a 3.1, which implies that there is a chance for the respondents to recommend it to others, but there is still room for improvement. Of note is that 2 people answered with a 5 (very likely) and one person answered with a 0 (very unlikely). This data can be seen in Figure 27. As a side note, one respondent changed their response from a 4 to a 5 after playing more of the game. This suggests that the beginning is not as enjoyable as the rest of the game.

6.2.1.2 Future Improvements

This round of user testing confirmed that there was a problem with the bomb aiming system used in *Bomb Golf*. This system will be modified to include rotating the camera without arming the bomb. Using this system, players would align their camera with the intended launch trajectory before beginning to launch a bomb. Because of this, it will also be possible to reduce the sensitivity of the aiming indicator (the arrow) for finer control over the bomb.

The first level will be modified to have infinite bombs and a clear indicator for this status in order to make the game less punishing to first-time players. Additionally, a tutorial will play whenever a player enters the factory for the first time.

A problem that one user mentioned was that even though they had disabled the option to rotate between landscape and portrait mode in their device's settings, the screen was still

rotating in the game. In the future, we will add a small options menu to the factory where players can enable or disable this feature. In addition, players will be able to use this menu to mute the music or the sound effects, which will be added to the next large release of the game.

6.2.2 Retention Testing

In parallel with the online feedback forms, more data was gathered from a wider audience. Whenever the version of *Bomb Golf* available on the Google Play Store is played with an internet connection, the game sends data online for analysis. If an internet connection is not available, the data is saved locally and sent at the next opportunity. The data is not affected by this process. This was achieved using a free tool called GameAnalytics[20], which was integrated into the game.

It is important to note that this data is likely somewhat biased, because the feedback form was sent out while it was being gathered (see Subchapter 6.2.1). The form did not explicitly ask users to complete any levels, but it could have provided players with additional motivation to play the game.

6.2.2.1 Results and Conclusions

Over the 20-day testing period, 42 new users played the game on 31 different device models, though the first level was completed only 27 times. This means that a large number of players got confused or quit before they were able to complete the first level.

The number of completions for each level can be seen in Figure 28. The figure shows a steady decrease in players who completed subsequent levels, which is to be expected. A sudden drop in the number of completions can be noticed on level 4, which suggests that the difficulty of this level might need to be adjusted.

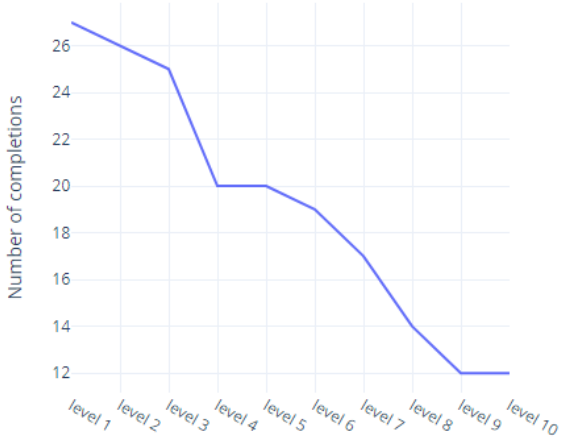

Figure 28. Level completions during testing period

[20] https://gameanalytics.com/

The game's day 3 retention during the testing period was 28.12%. This means that 28.12% of users returned 3 days after installing the game[21]. In comparison, last year's average day 1 retention for mobile games was 25% [19] and in most cases, retention rate tends to drop significantly after the first day. This means that *Bomb Golf*'s day 3 retention far exceeds the average expected day 3 retention, which shows that the game has notable potential.

However, *Bomb Golf*'s day 1 retention was 13.79% and its day 2 retention was 21.25%, which shows that the game has trouble with getting people invested after the first session. Many players gave up on the game after the first try and returned over a day later. The cause of this is likely the use of a limited, but ever-increasing pool of resources (bombs) as a way to regulate session length. This system causes subsequent sessions to become longer, which means the initial session is not long enough to get players invested.

[21] https://gameanalytics.com/docs/item/metric-and-dimensions-reference

7 Conclusion

Through the creation of this thesis, a precision-based mobile game called *Bomb Golf* was developed. This game was created to be popular in the mobile game market and to serve as a basis for developing a commercial product in the future. The game was compared to other similar games in order to confirm that it stands out from the competition.

Several game mechanics were designed, analyzed and implemented for the game, such as the timer, the targets and the obstacles. Additionally, to increase retention and incentivize players to pace themselves, bombs were made into a limited resource, which is shared across all levels. More bombs can be obtained using resources found in the game or by waiting for the factory to generate more.

The early levels were thoroughly analyzed to make sure they provide the players with a positive first impression and teach them valuable skills. Throughout the development process, a total of 15 levels were created for the game. Additionally, many of these levels were adjusted to create a loop of "learn, play, challenge, surprise" for the players.

The Robotex 2019 public expo gave the developers an opportunity to test the game on a wide audience, which was helpful for highlighting a variety of game design problems. The greatest problem highlighted in that round of testing was how the game and its systems were confusing to the players. A solution to this problem was implemented in the form of a tutorial, which introduces new players to the game's controls. In a similar fashion, many other problems found during the public expo were addressed in this phase of development.

Finally, a second round of testing was conducted online, as the game was uploaded to the Google Play Store and an online feedback form was created. The feedback from this form highlighted some problems with *Bomb Golf*'s control scheme and the game's lack of clarity. Additionally, retention testing from this time period revealed that the game has poor day 1 retention (13.79%), but high day 3 retention (28.12%), which suggests that the game has potential, but the initial user experience should be improved. In the future, development for the game will continue and solutions to these problems will be designed and implemented.

Thanks to all the testers for trying out the game and to Ivars Bergs, who was an important part of the development team for most of the process and helped significantly improve code quality. Special thanks to supervisor Raimond-Hendrik Tunnel who was an immense help with the writing process, answering many questions and offering useful feedback.

8 References

[1] GlobalData Plc. Video games market set to become a $300bn-plus industry by 2025. 2019. https://www.globaldata.com/video-games-market-set-to-become-a-300bn-plus-industry-by-2025/ (03.01.2020)

[2] Eensaar L. Eesti mobiilimängude tootjad rallivad edu poole. 2019. https://www.aripaev.ee/majandustulemused/2019/04/02/eesti-mobiilimangude-tootjad-rallivad-edu-poole (29.12.2019)

[3] Adepu S, Adler R. F. A comparison of performance and preference on mobile devices vs. desktop computers. Published in: 2016 IEEE 7th Annual Ubiquitous Computing, Electron-ics & Mobile Communication Conference (UEMCON). 2016. 10.1109/UEMCON.2016.7777808

[4] Saporito N. Inkscape Vs Illustrator: Can Inkscape Replace Illustrator? 2016. https://logosbynick.com/can-inkscape-replace-illustrator/ (09.03.2020)

[5] Christopoulou E, Xinogalos S. Overview and Comparative Analysis of Game Engines for Desktop and Mobile Devices. Published in: International Journal of Serious Games. 2017, Vol 4 No 4. 10.17083/ijsg.v4i4.194

[6] Elahee Y. Three ways to do market research for your video game. 2017. https://medium.com/@Tavrox/three-ways-to-do-market-research-for-your-video-game-a31597e79d3 (28.01.2020)

[7] Guinness World Records. Guinness World Records Gamer's Edition 2015. Guinness World Records Limited. 2014.

[8] Swink S. Game Feel: A Game Designer’s Guide to Virtual Sensation. United States of America, Burlington: Morgan Kaufmann Publishers. 2009.

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

[10] Salen K, Zimmerman E. Rules of Play - Game Design Fundamentals. England, Massachusetts: The MIT Press. 2003

[11] Jordan J, Cousins B, Levy E, Stolzenberg J, Robinson M, Katkoff M, Celinder M, Lovell N. Is there any place in the F2P designer's toolbox for energy-based economies? 2014. https://www.pocketgamer.biz/mobile-mavens/59247/is-there-any-place-in-the-f2p-designers-toolbox-for-energy-based-systems/ (19.04.2020).

[12] Tolomei S. Shrinking APKs, growing installs. 2017. https://medium.com/googleplaydev/shrinking-apks-growing-installs-5d3fcba23ce2 (19.04.2020)

[13] Solarski C. The Aesthetics of Game Art and Game Design. 2013. https://www.gamasutra.com/view/feature/185676/the_aesthetics_of_game_art_and_.php (19.04.2020).

[14] Taylor D. Ten principles of Good Level Design. 2013. https://www.gamasutra.com/blogs/DanTaylor/20130929/196791/Ten_Principles_of_Good_Level_Design_Part_1.php (21.04.2020)

[15] Matos X de. Bethesda's Todd Howard on how games can make players 'proud'. 2012. https://www.engadget.com/2012-02-08-todd-howard-on-how-games-can-make-players-proud.html (19.04.2020)

[16] Rubin J. The Handbook of Usability Testing. New York: John Wiley and Sons, Inc. 1994.

[17] Nielsen J. Usability 101: Introduction to Usability. 2012. https://www.nngroup.com/articles/usability-101-introduction-to-usability/ (24.04.2020)

[18] Narits H. Robotex19-mängud. 2019. https://lightroom.adobe.com/shares/f069a9fd3c6e413586e3703c78ec9d56 (03.05.2020)

[19] GameAnalytics. Mobile Gaming Industry Analysis for H1 2019. 2019. https://gameanalytics.com/resources/item/mobile-gaming-industry-analysis-h1-2019 (02.05.2020)

[20] Naro G. Diegesis and designing for immersion. 2018. https://www.gamasutra.com/blogs/GabrielNaro/20180801/323232/Diegesis_and_designing_for_immersion.php#comments (04.05.2020)

Appendix

I. Glossary

Player Character	A character in a video game who is fictional and whose actions are controlled by the person playing the game (the player)[22].
Diegetic	Belonging to the universe within the game world, where the player character exists [20].
Non-diegetic	Not belonging to the game world [20]. This part of the game only exists for the player, not for the player character.
Image texture	In computer graphics, it is an image which is applied to a surface as a texture[23]. Usually this makes the image appear as if though it has been "painted" onto the surface.
User Interface (UI)	In industrial design, it is the space where interactions between machines and humans occurs[24].
Graphical User Interface (GUI)	This is a type of user interface which allows users to interact with the device primarily using graphical icons or audio, as opposed to text-based user interfaces or other alternatives[25].
Retention	In data analysis, retention (also known as user retention) is the continued use of a product or feature[26].

[22] https://en.wikipedia.org/wiki/Player_character
[23] http://math.hws.edu/graphicsbook/c4/s3.html
[24] https://en.wikipedia.org/wiki/User_interface
[25] https://en.wikipedia.org/wiki/Graphical_user_interface
[26] https://medium.com/@tsharon/measuring-user-retention-a2f937234b99

II. Accompanying Files

- */Robotex_feedback_responses* – the folder containing the responses for the feedback received at the Robotex 2019 public expo
- *robotex_feedback_form.pdf* – a close approximation of the feedback form used at the Robotex expo, the final version of this form was adjusted and printed by the organizers of the event
- *robotex_poster.pdf* – a poster with a short description of the game used at the Robotex expo
- *online_feedback_form.pdf* – the feedback form used for online testing
- *online_feedback_responses.csv* – the data gathered from responses to the online testing feedback form
- *retention_testing_data.xlsx* – the data gathered from retention testing
- *bombgolf_expo_build.apk* – the installation package for the expo build version of the game
- *expo_build_video.mp4* – a short video demonstration of the expo build for *Bomb Golf*
- *online_build_video.mp4* – a short video demonstration of the build of *Bomb Golf* uploaded to the Google Play store

III. Source code

Available via request on https://bitbucket.org/KaarelRyysak/bombgolf/src.

Please send an email to kaarelkaks@ryysak.com to request access.

IV. Development Screenshots and Graphics

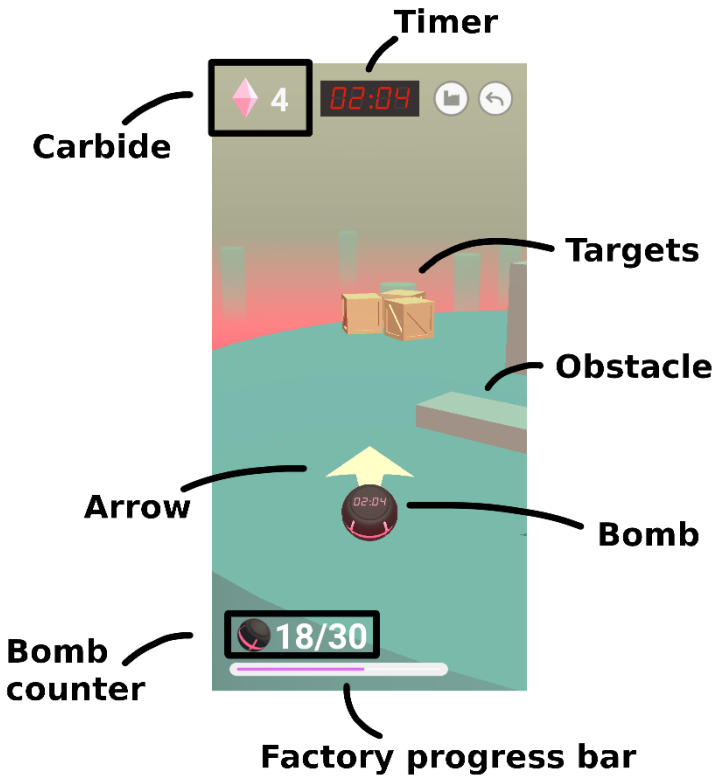

Figure 29. Overview of the game mechanics in *Bomb Golf*

Figure 30. Screenshot from a prototype of *Bomb Golf* created in Playcanvas

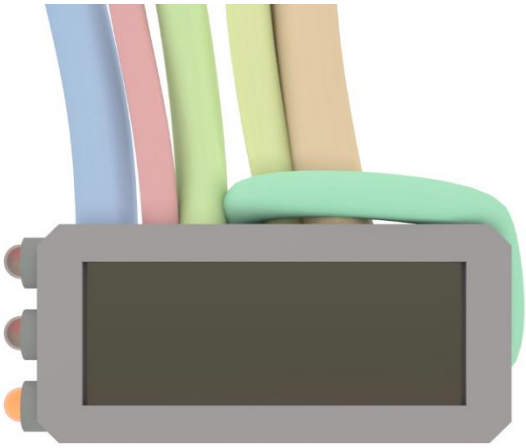

Figure 31. Prototype timer aesthetics

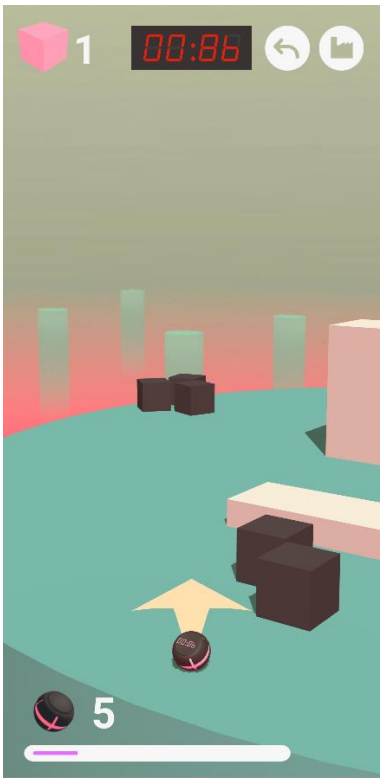

Figure 32. Screenshot from an early version of *Bomb Golf*

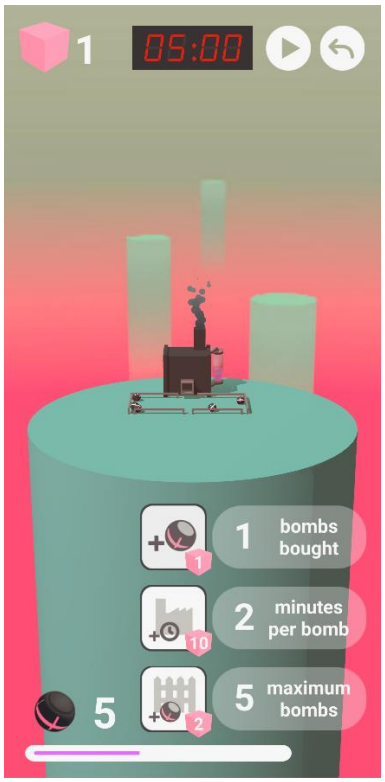

Figure 33. Factory in an early version of *Bomb Golf*

V. License

Non-exclusive license to reproduce thesis and make thesis public

I, Kaarel Rüüsak, (*author's name*)

1. herewith 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,
   Bomb Golf – A Precision-Based Mobile Game, (title of thesis)
   supervised by Raimond-Hendrik Tunnel (supervisor's name)
2. I grant the University of Tartu a permit to make the work specified in p. 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 3.0, 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 p. 1 and 2.
4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Kaarel Rüüsak

02/05/2020