

# MOBILE PROGRAMMING AND MULTIMEDIA

PROJECT REPORT

A.Y. 2023/24

---

## Archimede Tower Defense

---

Davide Spada n° 2113495  
Mirco Bisoffi n° 2113496

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Framework</b>	<b>2</b>
2.1	Introduction . . . . .	2
2.2	Unity Framework . . . . .	2
2.2.1	Unity Interface . . . . .	3
2.2.2	Scripting Language . . . . .	3
2.3	Motivations . . . . .	4
<b>3</b>	<b>Archimede Tower Defense</b>	<b>5</b>
3.1	A Short Story . . . . .	5
3.2	Player Actions . . . . .	5
3.2.1	Home Screen . . . . .	6
3.2.2	Shop Menu . . . . .	7
3.2.3	Map and Turrets placing . . . . .	8
3.2.4	Pause Menu . . . . .	9
3.2.5	Turret Sell . . . . .	9
3.2.6	Game Over . . . . .	10
<b>4</b>	<b>User Interface Design</b>	<b>11</b>
4.1	Home screen . . . . .	11
4.2	Left or Right Handed . . . . .	11
4.3	Pause Button . . . . .	11
4.4	Sound . . . . .	12
4.5	Shop Menu . . . . .	12
4.5.1	State Icons . . . . .	12
4.5.2	Turrets . . . . .	13
4.6	Turret Placing . . . . .	13
4.7	Tutorial . . . . .	13
4.8	Emotional design . . . . .	14
4.9	Game Icon . . . . .	14
<b>5</b>	<b>Battery consumption</b>	<b>15</b>
<b>6</b>	<b>Future Implementations</b>	<b>15</b>

# 1 Introduction

For the project of the course *Mobile Programming and Multimedia* we decided to develop a funny game based on the famous sub-genre of strategy games Tower Defense. The goal of the game is to avoid the enemies from reaching the end of the path, which is achieved by placing strategically defensive structures, called turrets, along their path to defeat them.

Moreover, the game is played in a 2D environment and it is designed for mobile platforms.

Also, we have decided to redesign it by referring to concepts that recall the Department of Mathematics. In particular, the map corresponds to the complex of buildings where the *Torre Archimede* is located, while turrets and enemies refer to concepts concerning school and study.

For these reasons we named this tower defense game as **Archimede Tower Defense**.

## 2 Framework

### 2.1 Introduction

*Archimede Tower Defense* game is developed in a Unity environment, specifically configured for 2D mobile platforms. Here we are going to present some information regarding the framework and the reasons we chose it.

### 2.2 Unity Framework

The game industry nowadays is expanding and growing continuously, on all sides considering consoles, developing frameworks, games, players and developers. According to the Unity Gaming Report 2023, Unity has a share of 43% of the global game engine market and the number of games made with Unity has grown by +93% over the last year. In particular, according to Unity Technologies, 71% of all mobile games are built with it, for a total of 2.7 billions users.

Created in 2005 by the Unity Technologies company, it has become the world's leading cross-platform development environment that provides a wide range of software solutions and tools to create real-time interactive 2D and 3D content. The Unity Engine supports building games for more than 19 different platforms, including mobile, desktop, consoles, and virtual reality, while the Unity Editor is supported on Windows, macOS, and the Linux platform, which uses C# as the primary programming language for the engine.

Among all the game engines, it is considered one of the best solutions for beginners that provides at the same time a complete suite of tools for experienced developers.

Unity development consist of 3 parts:

- **The game engine** : allows to create, test, and play games and experiences in various environments.
- **An application** : it provides a complete interface that combines design and user interaction with graphics with a wide range of controls and tools.

- **IDE** : an external Integrated Development Environment (VS Code for example) for writing code scripts in the C# programming language, for managing the logic and behaviour of different components in the interface.

### 2.2.1 Unity Interface

Figure 1 shows the Unity interface.

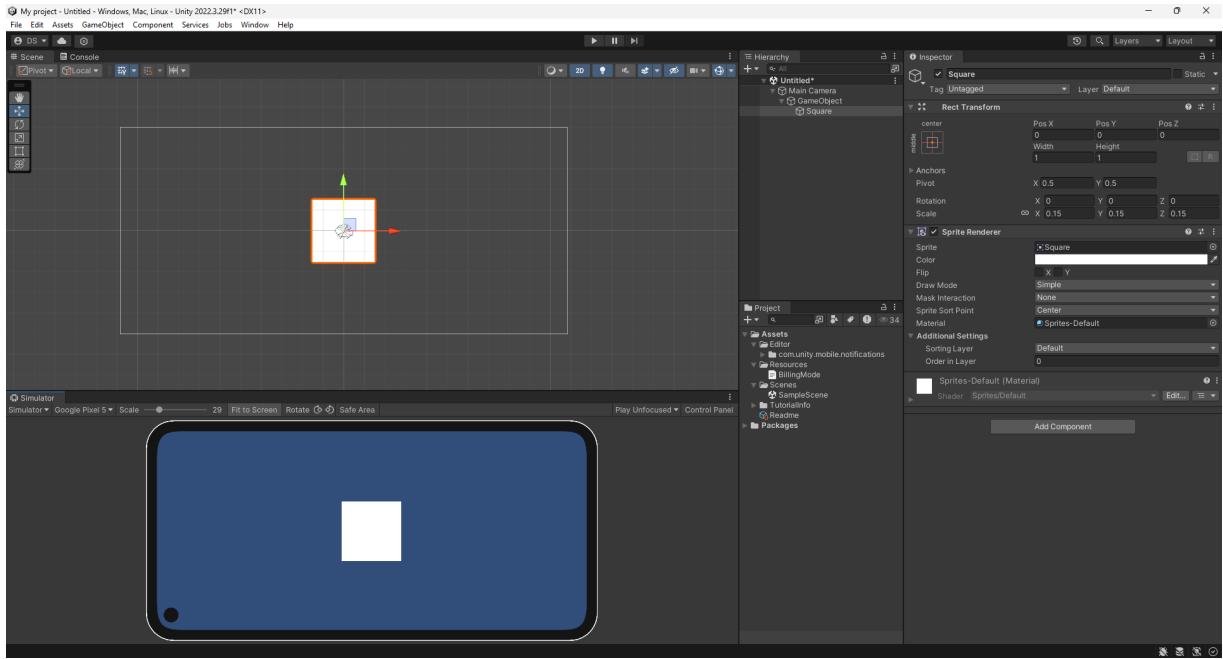


Figure 1: Unity interface

- **Scene**: This is where all the game objects will be placed, viewed and manipulated.
- **Simulator**: This is where results can be seen for testing purposes and it allows a wide range of different built-in mobile devices of major companies. In the figure it is shown the Google Pixel 5 simulator.
- **Hierarchy**: This window displays all game objects placed directly on the scene. Basically everything needs to be listed here, which include visual and non-visual game objects.
- **Project**: This shows what's inside folders on the disk. Everything from game objects, scripts, textures, folders, models, audio, video etc.
- **Inspector**: This panel displays different attributes and properties of the selected game object.

### 2.2.2 Scripting Language

Scripting is an essential part in all applications made in Unity. Most applications need them to respond to input from the player and to arrange for events

in the gameplay to happen when they should. Beyond that, scripts can be used to create graphical effects or to control the physical behaviour of objects. Each script can be attached to one or more game objects.

Here it is a brief overview of how it works.

As introduced, C# is the programming language and all scripts are derived from the base class *MonoBehaviour*.

*MonoBehaviour* class provides a framework of various methods, that can be overridden and extended to implement specific behaviors.

Some of the most used methods include *Start()*, called before the first frame update and used for initialization; *Update()*, called once per frame and used for regular updates; *FixedUpdate()*, called at a fixed time interval and used for physics calculations; *Awake()*, called each time the object is invoked; *OnCollisionEnter()* called when a collision occurs.

Also, it is possible to interact with other game objects using methods like *Find()* or *GetComponent()* and invoke their public methods for managing their behaviour, for example when a collision happens.

## 2.3 Motivations

Now we are going to present some consideration about pros and cons about the framework and the motivations that lead us to chose it.

As explained, Unity Engine is one of the leading framework in the game development that supports a wide range of different devices, making available to adapt the game for different platforms. This make it easy to maximize exposure to players and increase income and visibility. For this reason it is ideal for mobile games supporting building for major OSs like iOS and Android from a single codebase. This is crucial since it reduces development time and effort making it a cross-platform development framework. Also it provides out-of-the-box tools for 2D games, like advertising and in-game payments as well as built-in integrations to take advantage of technologies already present on every smartphone like GPS, gyroscopes, and accelerometers.

On the graphic side it doesn't compete with other photo-realistic engines like Unreal however it provides an excellent levels of visuals even for advanced players. Another advantage is that it provides a basic pricing plan which is free for any individuals or small teams, making it one of the most affordable and most used in the game developing market. This motivate the Unity developer community, which is the largest among other engines. As beginners we found answers on forums, threads and video channels. Also huge number of training are available, which makes the learning curve "smoother" and simple.

However, the learning curve, especially in the beginning when facing for the first time the Unity framework, is hard since the learning can take time and various experiments to get familiar with the environment. We considered this eventuality and a certain amount of time was spent studying the framework also to understand what we could and could not achieve with our capabilities. As explained, the huge community online helped us on this side. Regarding the performances, in some use cases it can be high resource consuming, especially for 3D game development. For the project, developed in a 2D environment, our personal computer were sufficient since the resource demand was limited hence completely usable. Finally, large size of scenes that includes a lot of game

objects and components may require, as explained in the previous point, high computing resources. This was not our case since it is a simple game and the resources created never created computing problems.

### 3 Archimede Tower Defense

In the following section we are going present an overview of the mobile application. The source code is in the GitHub repository at the following [link](#). The application works on the following minimum versions of Android: Android 11 (API level 30) and requires about 48 MB of space.

*Archimede Tower Defense* is a 2D mobile game application in the style of a Tower Defense game. The game consist of only one level and one difficulty and, for time constraints, we didn't consider a progress saving options. The purpose of the game is not to allow enemies to reach the Torre Archimede on their path, which will otherwise take away one life point for each enemy that manages to enter. This is done by placing turrets along the way, that will hopefully defeat them. The player can play indefinitely as long as the number of lives is greater than zero. The aim is therefore to survive as long as possible.

#### 3.1 A Short Story

The Mathematics Department, housed in the famous and unconquerable structure known as Torre Archimede, was a sanctuary of knowledge, numbers and source codes. Its classes were busy with lessons, its computers ran day and night source codes designed to be the fastest in the world and its study hall was full of mathematicians and computer scientists working on secret algorithms. However, this serene environment was about to face an unusual threat.

The Medicine, Literature, and Psychology departments had always been strong rivals to Mathematics department, each planning their own strategies and strengths to defeat them secretly. Recently, something strange had begun to happen. Syringes, books, and brains, the representations of these departments, had started to unite together and march toward Torre Archimede, threatening to disrupt the mathematical peace within.

To defend their sanctuary, the brilliant minds of the Mathematics Department developed a unique mobile app called *Archimede Tower Defense*. Now, we call upon all mathematicians and computer scientists! The time has come to defend our domain. Play *Archimede Tower Defense* and join the fight. The future of Torre Archimede depends on you!

#### 3.2 Player Actions

Here we provide a list of screenshots that shows the different actions a player can face playing the game. The screenshot provided are taken from a Google Pixel 5 simulator.

### 3.2.1 Home Screen

As soon as the user opens the application, the Home appears as the first screen, as shown in Figure 2.

Here the user can select one of the three options provided:

- **Play:** allows to start the game.
- **Options:** allows to open the option panel, described later.
- **Quit:** Exit the game.



Figure 2: Home Screen

If the user select *Options* from the list, an option panel will appear in the same screen over the list, as shown in Figure 3. Here, the user can select one of the two options provided through a toggle that acts like a switch. In particular:

- **Sound:** The default value is *On*. It allows to control the sound of all the game by simply switching it.
- **Hand:** The default value is *Right*. It allows to select the hand they use, that is whether is right or left-handed. This is useful for the game, explained later in Section 4.2.
- **Cancel:** To close the option panel there is a cancel button on the right-up corner.

After selecting the preference options and closing the panel, the user, by clicking Play, can start the game, which will open a new screen, as shown in Figure 4.

The player here can see the first screen of the game that will start the Tutorial, better explained in Section 4.7, which teaches in four simple steps the basic instructions to play the game.

Regarding the game, the screen is divided into two main components: the **Shop Menu** and the **Map** as shown in Figure 5.



Figure 3: option Panel

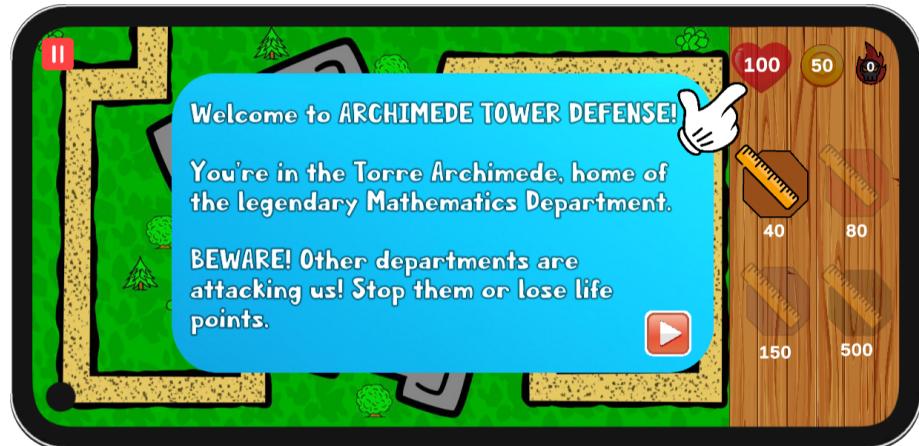


Figure 4: The Game - Start

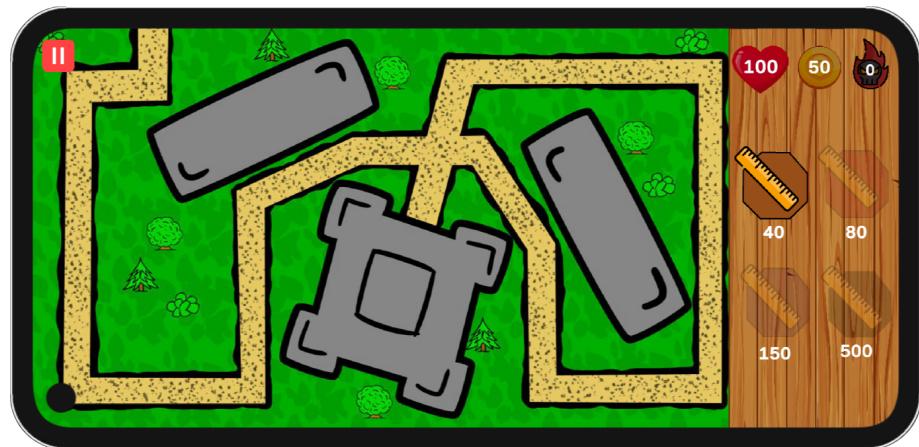


Figure 5: Game Screen

### 3.2.2 Shop Menu

The **Shop Menu**, in Figure 5 shown on the right, provides the necessary elements to play the game and check the status.

More in detail, on above there are three different icons:

- **Lives:** Shows the number of available lives and cannot be increased. It decreases of -1 every time an enemy enters Torre Archimede. It is represented by an heart.
- **Coins:** Shows the number of available coins, to place the turrets. The value is increased of +1 every time an enemy is killed and decreased of -1 when a turret is placed. It is represented by a coin.
- **Waves:** Shows the number of waves of enemies, that will spawn in increasing number, making the game more and more difficult. It is represented by an enemy.

Under the icon, in the center, there are four different type of **Turrets**, all of different colors (brown, red, purple and black). They have different prices, indicated by the value below them, and according to the price, more the value is bigger, more the shooting power, hence the ability to kill an enemy, is higher. Also, if no coins are available to buy a turret, it will be shown as faded, meaning that it cannot be placed.

### 3.2.3 Map and Turrets placing

The **Map**, in Figure 5 shown on the left, shows the game arena. The enemies will spawn from the top left corner of the screen and will end in Torre Archimede, in the center, following the yellow path.

To place a turret, the player must have sufficient coins. At that point the player can simply drag-and drop it on the screen. To help placing the turrets, while dragging, the icon will appear red if it is not possible to place it on the current spot. Otherwise, it will appear green, meaning that dropping it will place the turret where the finger is on.

This is shown in Figures 6 and 7.

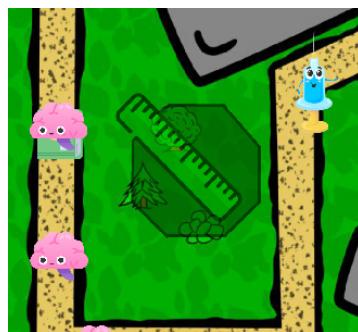


Figure 6: Safe placement

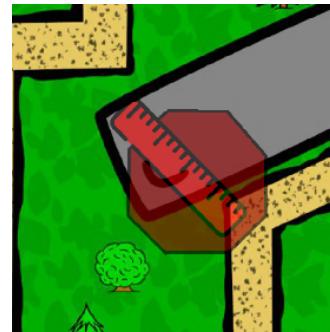


Figure 7: Wrong placement

At this point the player will see a screen like this, as shown in Figure 8. The turret will kill enemies, that will earn coins, to place new turrets and so on.

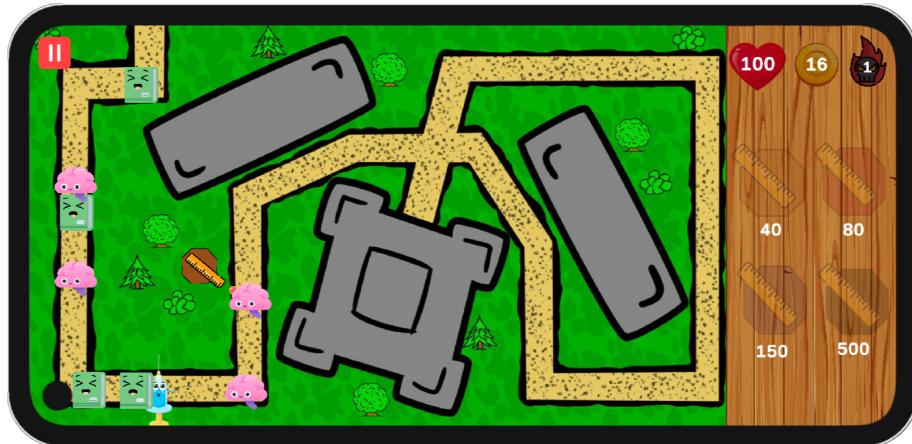


Figure 8: The Game - Turret placed

### 3.2.4 Pause Menu

Also, the player can pause the game. The button is on the top left of the map and it is discussed in Section 4.3. Tapping on it, it will display a **Pause Menu** as shown in Figure 9. It contains two buttons:

- **Home:** Showed on the left, allows to go to the Home screen, resetting the current game session.
- **Continue:** Showed on the right, allows to continue the game.

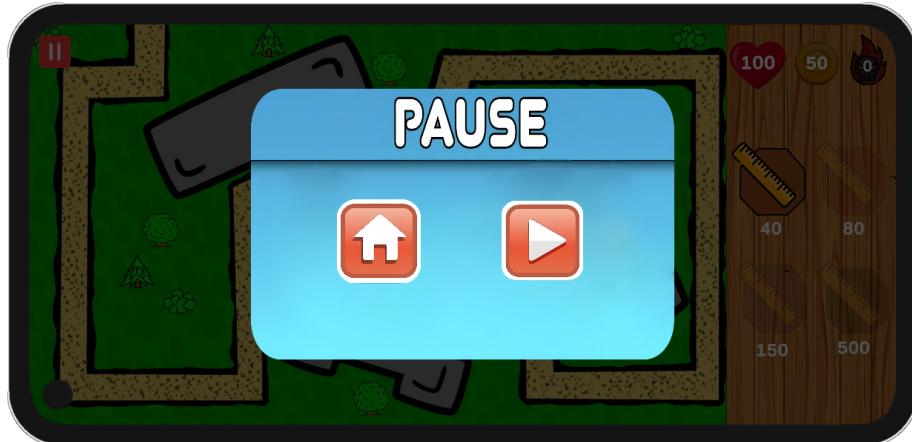


Figure 9: Pause Menu

### 3.2.5 Turret Sell

The player may also decide to sell one of the already placed turrets for half of the purchase price. It is only possible to do this after the tutorial is finished. This action is useful, for example, when coins are not enough to buy a higher-level tower or when a turret is no longer powerful enough. This action can be performed by a tap on the turret you want to sell. For the first time only, a pop-up will be shown to tell the user the unlocked selling functionality. At that

point it will be possible to continue. As shown in Figure 10, the sale pop-up will then be shown with the profit and a button to complete the sale. The amount will then be added to the total coins and the turret will be removed from the map. Otherwise, it is possible to cancel the operation.



Figure 10: Shop Menu Swapped on the Left

### 3.2.6 Game Over

Finally, when the available lives in equal to zero, the player loses the game. In this game it appears a Game Over screen, as shown in Figure 11. From here the user can only go to the home screen, by tapping on the home button below.



Figure 11: Game Over

## 4 User Interface Design

In this section are presented the different design choices we made thinking of a mobile application environment, for a better usability and accessibility to the player, based on what we learned on the topics of the course.

### 4.1 Home screen

First of all, in the Home screen, as shown in Figure 2, the sorting of the various elements allows to see them from top to bottom in reading order. In fact, the title of the game appears at the top, while below it there are the three interactive buttons Play, Options and Quit.

We decided to create the buttons with a 3D style, with a rounded and smoothed rectangle shape, creating a visually appealing element. Moreover the colors are different, to distinguish them one from the other.

The Play button is bigger and positioned in the center (almost double the size of the others), since it is the most important of the Home screen, that effectively starts the game.

### 4.2 Left or Right Handed

To use all the screen available, the game can be played only in landscape mode, left or right (the choice is left to the user simply rotating the device). This implies that the game can be played while holding it in two different ways:

- With both hands holding the device. This implies that turrets will be placed with the thumb.
- With one hand holding the device and the other placing the turrets. This implies that turrets will be placed using the finger.

In both scenarios, the position of the Shop Menu is crucial since, a right or left-handed user (or a user who prefers one mode or the other), it will be easier if the menu is on the right or left side of the screen. This make the game more accessible, usable and customisable.

The setting is available in the Option panel, from the Home screen, as shown in Figure 3.

Figure 12 shows how it looks like when it is swapped for left-handed player.

### 4.3 Pause Button

The game can be paused using the red button on the top left of the map.

We had initially considered placing it below the turrets in the Shop Menu as it seemed a logical position but risked being covered by the hand and being pressed by mistake. We therefore placed it on the top left of the map.

In addition, the size of the button is small enough to not to clutter the map. These choices make it enough difficult to press unintentionally while playing but at the same time easy to notice and reach.

Moreover, the position does not obstruct the view of the other most important elements of the game.

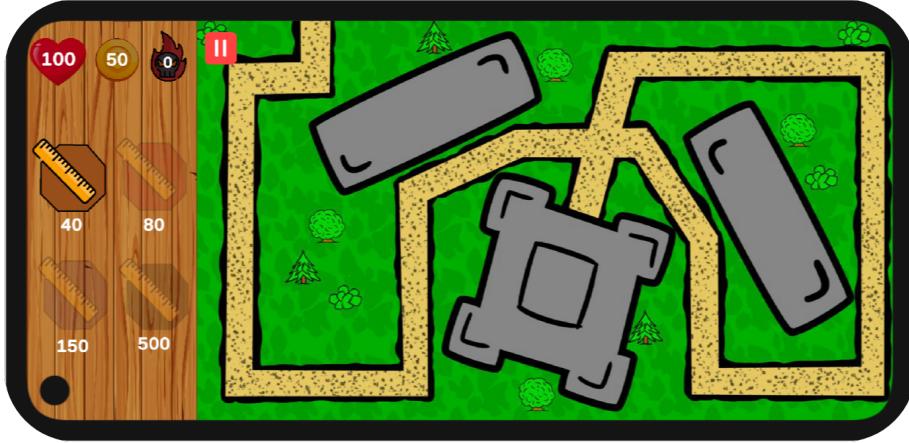


Figure 12: Shop Menu Swapped on the Left

#### 4.4 Sound

The game has a background melody that makes it nicer and more pleasant to play. The music is played throughout the game, from the home screen to the game.

In addition, each button in the application emits a sound to provide auditory feedback to the user.

A sound is also emitted when a bullet hits one of the enemies to again provide auditory feedback and enhance the gaming experience.

Finally, it is possible to disable all sounds in the game via the Option panel, setting the Sound value to Off as shown in Figure 3.

#### 4.5 Shop Menu

In this section are discussed the design choices of the Shop Menu.

##### 4.5.1 State Icons

Regarding the icons of Lives, Money and Waves in the Shop Menu, we placed them at the top, to prevent the hand from covering them while placing the turrets.

The numerical value was initially placed below them, but we noticed that this made it more difficult and slower to read. Considering that during the game the user has the eyes on the map most of the time, placing the numbers inside the icons makes the information easier and faster to read, which is crucial for better player responsiveness and ensures a better gaming experience.

The Lives icon, represented by a heart, is bigger. This choice was made because it is the most important among the three since on it depends the game over and it needs to be checked most of the time.

The second bigger is the Money icon, since it is the second most important because it influences which turret can be bought and placed.

Finally, the Waves icon is the smaller and shows the number of the current wave of enemies.

#### 4.5.2 Turrets

Regarding the turrets, we have decided to make four types available that differ in their increasing price, which then determines their shooting power to kill enemies.

In particular, if the funds are not sufficient then the turret cannot be purchased and is shown in a faded colour. We made this choice because it is also quite common in other games of this type so it is intuitive for the player to understand. A possible extension, which we have not implemented for time reasons, is to use the concept of coaching, so that, if a user keeps trying to place an unavailable turret, is warned.

#### 4.6 Turret Placing

When placing turrets, the player is helped on the location. In fact, not all areas of the map are available for placement, in particular on the Shop Menu, the path, the Torre Archimede, the other buildings next to it (to prevent the map from being cluttered by too many turrets during the game) and on the turrets already placed.

The turret is placed with a drag-and-drop gesture. When dragging the turret for placement, it takes on two colours:

- Green: If the position is available; then when it is dropped, it is placed in that position.
- Red: If the position is not available; then when it is dropped, it is not placed there.

Finally, we have considered that during dragging, the finger and part of the hand may cover the turret. This is why we have ensured that during dragging it is increased in size so that it can be more visible.

#### 4.7 Tutorial

We decided to add an initial tutorial as soon as the game starts, to instruct the player on the main dynamics of the game. The first step is shown in Figure 4. The tutorial is divided into four total steps:

- **Introduction:** The welcome is given and the context introduced. Also it is explained the concept of lives, which are taken away if an enemy enters the Archimedes Tower.
- **Turret Placing:** It is explained how to place a turret.
- **Coins:** It is explained the concept of coins, which are earned when an enemy is defeated and allow the player to buy more turrets.
- **Waves** At the end of the first wave the user is informed about the concept of waves and that more powerful ones will come.

At each step there will be hands indicating the elements described to help the user. In addition, we have tried to emphasise the most important words by capitalising them.

Moreover, it is executed only once, that is, if the user returns to the home screen and restart the game it will no longer start the tutorial.

With each step, it will be possible to continue to the next one by pressing the button located at the bottom right of the box containing it (this does not apply to the second step, which instead requires the player to place the turrets and will continue when the player has completed it).

The tutorial, however, does not explain all the dynamics of the game, which would risk boredom and abandoning the game for the user, but lets the remaining rules to be discovered as the player proceeds.

#### 4.8 Emotional design

For the game interface we chose energetic and cartoonish colors, making it visually engaging, since it is a funny game. Also the design of the different elements are designed to catch the eye ensuring that the player does not lose interest during playing and that different game elements are easily distinguishable from each other.

#### 4.9 Game Icon

For the icon of the game we chose the following, shown in Figure 13. This icon tries to recall the Torre Archimede, where the Mathematics Department is located. We consider it funny and with a cartoonish style. The eyes add expressiveness and personality which give it a friendly look.



Figure 13: Icon

## 5 Battery consumption

We tested the energy consumption of the application on two Android devices, a smartphone and a tablet (Google Pixel 7 and Samsung Tab S9 FE). The tests were carried out using *Battery Guru*, an external application specifically designed to measure the consumption of device applications. The tests were carried out for a one-hour gaming session on both devices, revealing the following consumption:

- **Google Pixel 7:** 6.6% per-hour, for a total of 287 mAh
- **Samsung Tab S9 FE:** 4.9% per-hour, for a total of 382 mAh

## 6 Future Implementations

In this section we list a series of possible improvements of the game that we initially thought about.

- More levels available, more and more difficult.
- A way to save the progress of the game.
- In the Option panel the possibility to change the difficulty of the game.
- Different sounds effects.
- A more engaging tutorial, with a mascot of the Mathematics Department.