

DOCUMENTATION

UNLOCK SYSTEM
version 1.1.5

INTRODUCTION

This documentation was created in order to give the user an idea about Unlock System, next I will call it as Asset or US. What kind of Asset and its features will be discussed below. US consists of 80% of the code (C #). Users who know the basics of this programming language or similar languages (C, C++) will easily understand the logic of the Asset.

Most likely, the user will find errors both lexical and code. To simplify the task to the developer (me), I ask you to report any errors/bugs found to me at the email address (choco.16mail@mail.ru) or on the Asset Forum.

The US code is commented in those places where the meaning that this section (method, variable) of the code (script) carries is incomprehensible. I did not comment on the elementary code operations. Mostly the code is intuitive and clear, I did not use complex methods of calculation, but only used basic mathematical operations. The code does not claim to be “ready for play” and always requires improvement, which means that you can always improve it (increase speed, reduce the number of operations and etc.).

Contents

INTRODUCTION	2
WHAT IS IT AND FOR WHAT IT IS NECESSARY	4
What is it?	4
How will it help me?	5
MECHANICAL INTERFACE	6
3D models	6
Hacking tool	9
System interface	11
ELECTRONIC INTERFACE	12
3D models	12
Hacking tool	15
System interface	17
DEMONSTRATION LEVEL	21
Description	21
UPDATE 1.1.5	23
What is new?	23
Spin the drum	23
Interface	25

WHAT IS IT AND FOR WHAT IT IS NECESSARY

What is it?

Games won the hearts of many people, I turned out to be one of them. I love The Elder Scrolls V – Skyrim game, especially its world - the open world. The origin of the idea of creating the Asset began with this game.

The Unlock System is present in almost all games, be it a hacking box with valuable loot or a hacking door to the next level. In most cases, these systems are all primitive - “press the button and go” or “get the key and open the door”. This system is interactive, i.e. the player must interact with it, enter a random code or rotate the latchkey or even spin the drums, one way or another the player must be present and turn on the logic.

The system includes 3 "levels" – mechanical (latchkey and key hole), mechanical (spin the drums) and electronic (non-electrical) interfaces, I will describe them in more detail below.

So what is this Unlock System - is an interactive system of hacking objects, doors, locks, etc., in which the player must perform certain actions known in advance or not to win (open the door, the box, etc.).

How will it help me?

This question is an individual one, everyone has their own needs. If you are developing a mobile game, but want such an Unlock System, then you will just need to add it to your project and set it up. How to set it up - watch the video on my YouTube channel.

If you do not like the 3D models used in the interfaces, you can easily replace them with your own.

The system is easily integrated into any project, you just need to add it and set it up a bit. Also watch the video on my YouTube channel.

MECHANICAL INTERFACE

3D models

The door lock model has textures with a resolution of 2048x2048, they convey good detail of objects. The model is made with a modular principle, divided into 3 parts (see Figure 1), which can easily be replaced by others. For example, in the 2nd version of this lock, I replaced the case of the door lock and added objects that imitate the grid (see Figure 5). When replacing parts of the main thing to remember to follow the proportions.



Figure 1



Figure 2



Figure 3



Figure 4

The second version of the lock on the principle of action is no different from the first one, added grid, part of the door etc. I gave them a bit of dynamics, making them transparent.

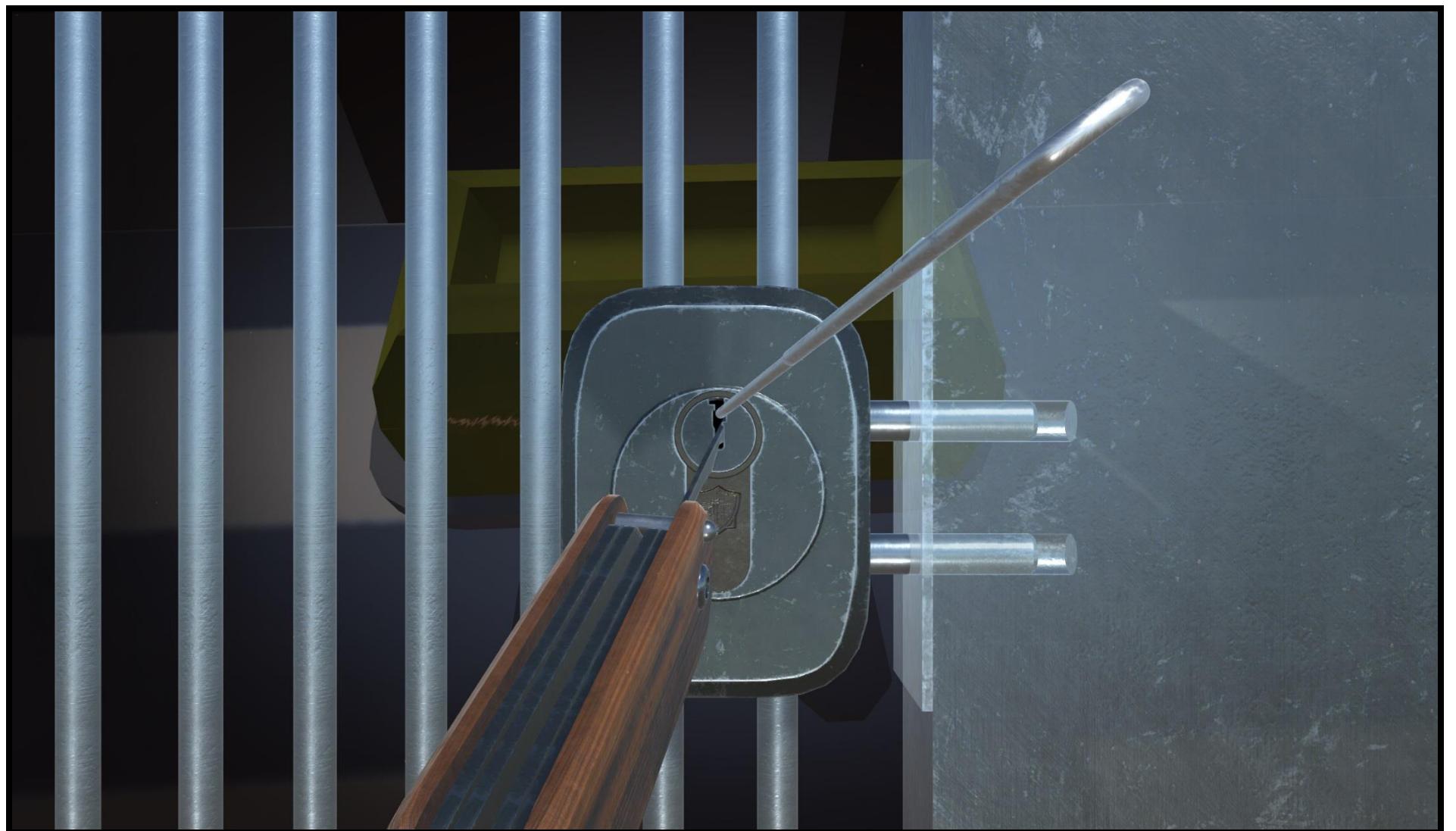


Figure 5

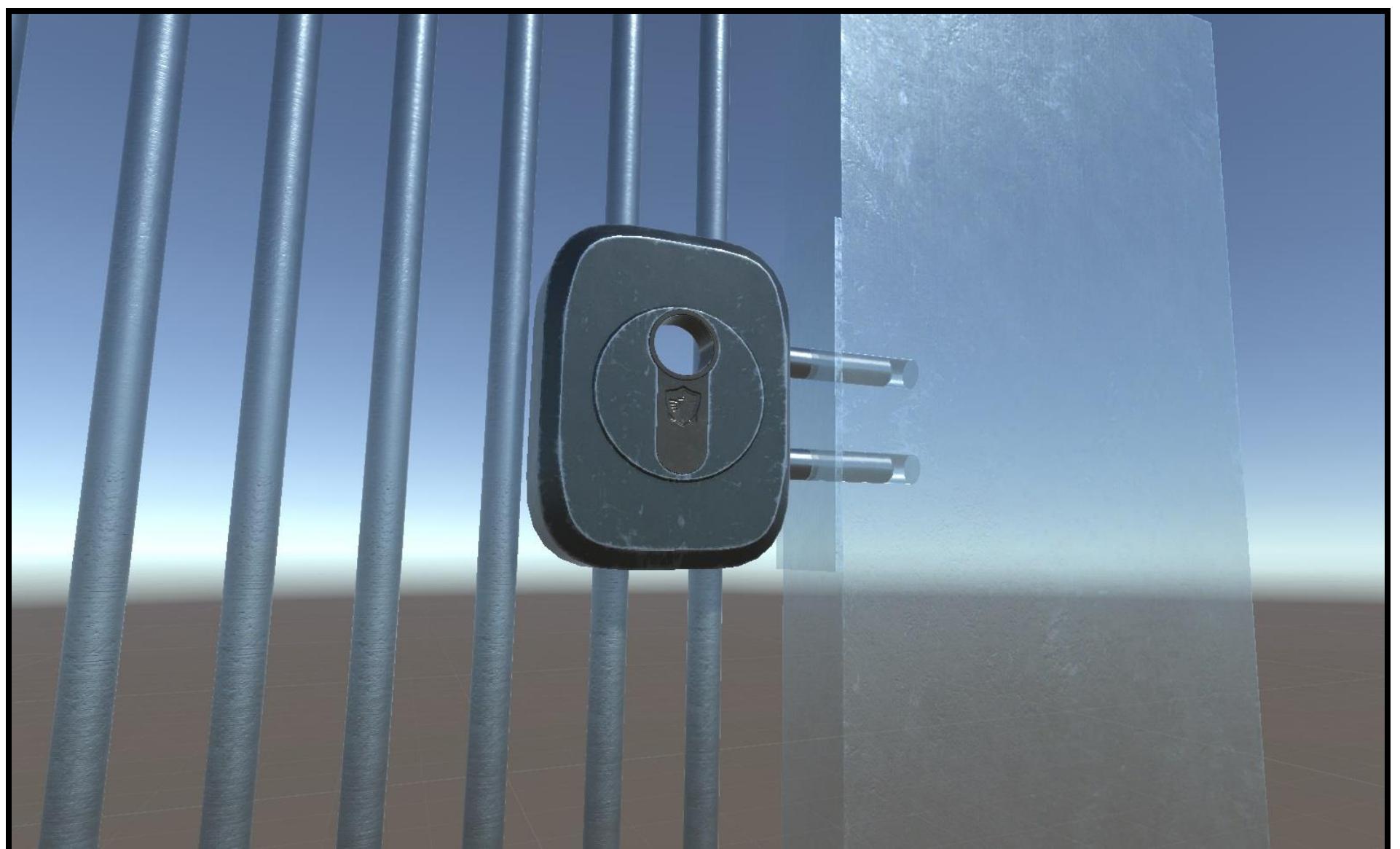


Figure 6

Hacking tool

As a tool was used multitool, a simplified version of it and a latchkey.



Figure 7



Figure 8



Figure 9

System interface

Let's talk about the interface of the mechanical Unlocking System. It is simple and intuitive. The player needs to move the mouse to the side (left, right), while the latchkey will rotate in the keyhole, when the player stops the mouse and presses (and holds it) a certain key the multitool starts to rotate. If the player gets into a certain area (which is set randomly, depending on the selected difficulty level), the multitool will rotate until a “click” sound is played, if the player does not get into the area, then the latchkey will start to twitch and eventually break. In figures 10 and 11, the blue color shows the total number of latchkeys, after which you cannot open the interface. The red area shows the “help” area, if you suddenly forgot how to manage it or how to exit. Green color shows the area of moving the latchkey and multitool.

After the “click” sound is played, the window will automatically close and transmit a signal that the door is open (or a box, etc.). For a visual representation of the system, just run the demo scene and spend some time trying to complete all levels (this will be difficult to do).

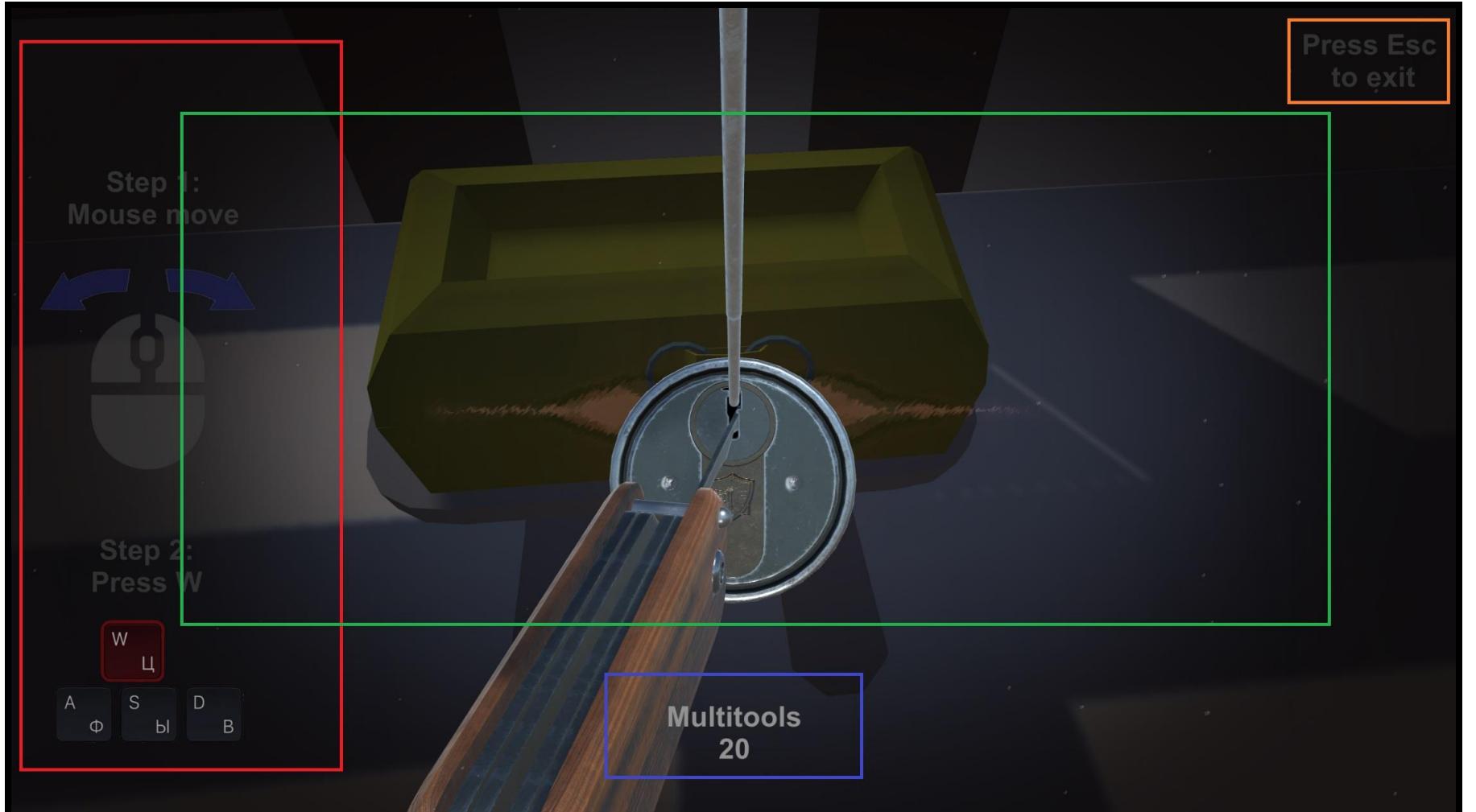


Figure 10 – The first version of the lock

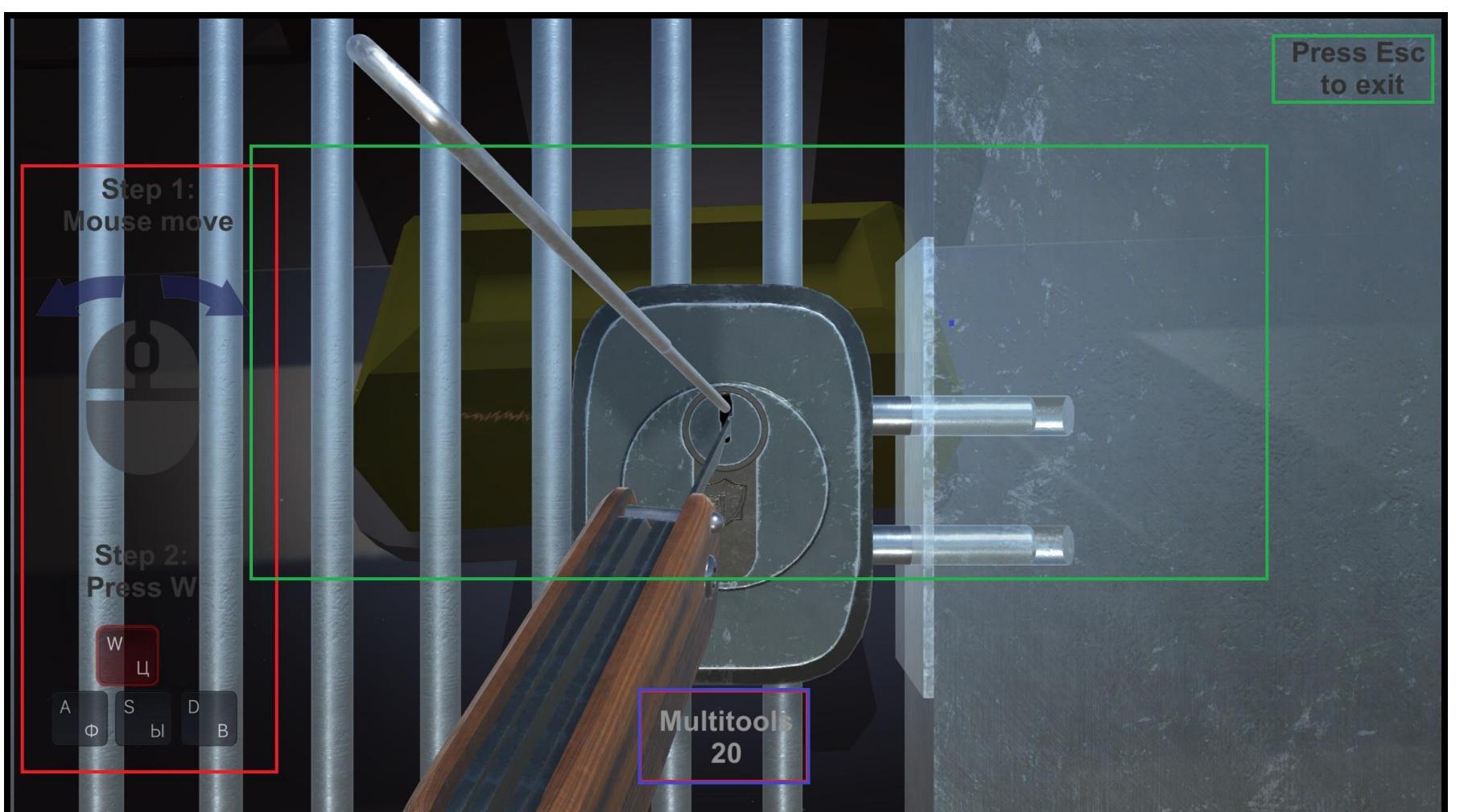


Figure 11 – The second version of the lock

ELECTRONIC INTERFACE

3D models

Electronic box models have textures with a resolution of 2048x2048, as was written early, they convey good detail of objects. For better performance, you need to reduce the size of the textures in any available editor (not necessary). The model is made with a modular principle, divided into many parts, which can easily be replaced by others or used in other assemblies.

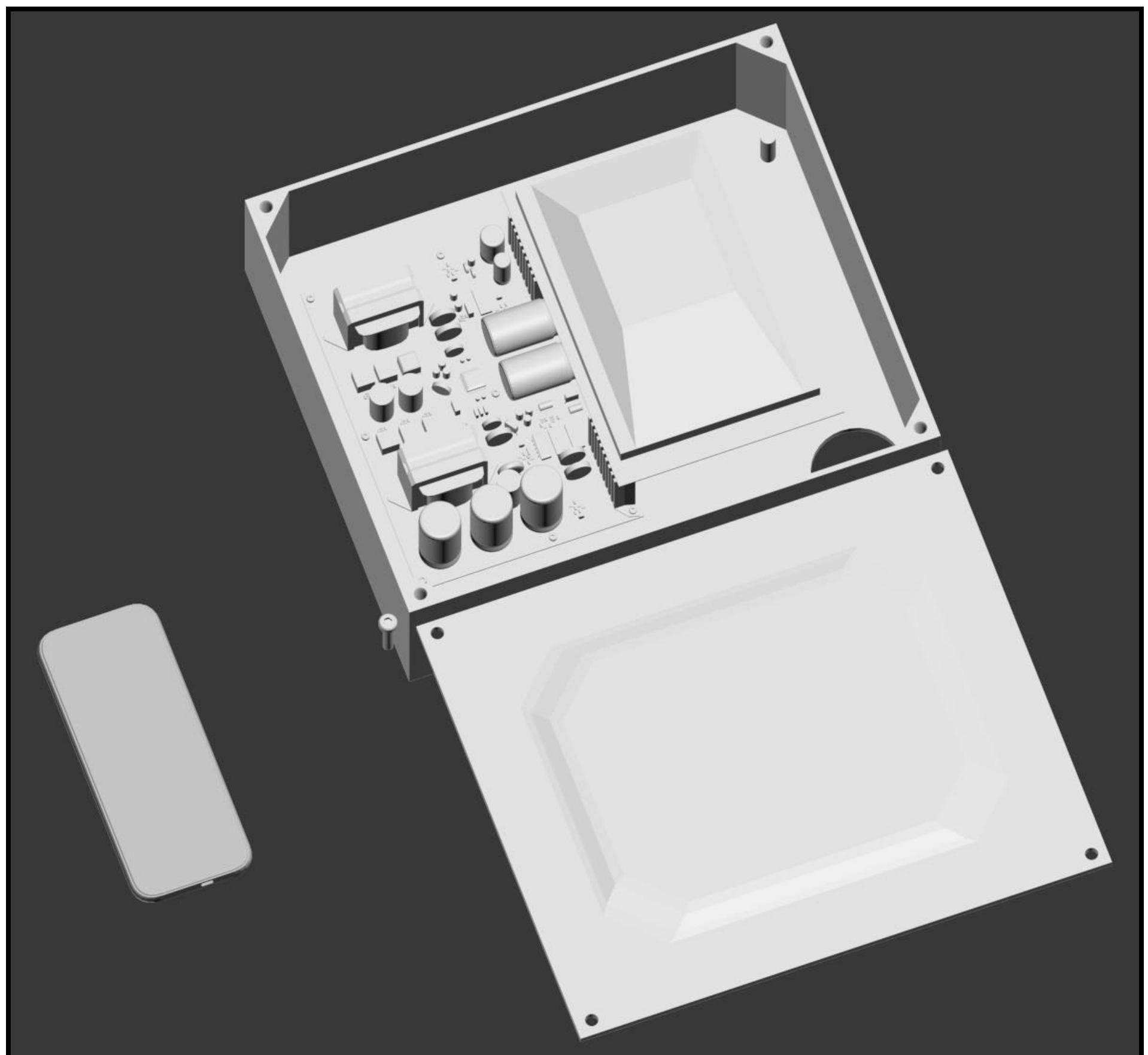


Figure 12

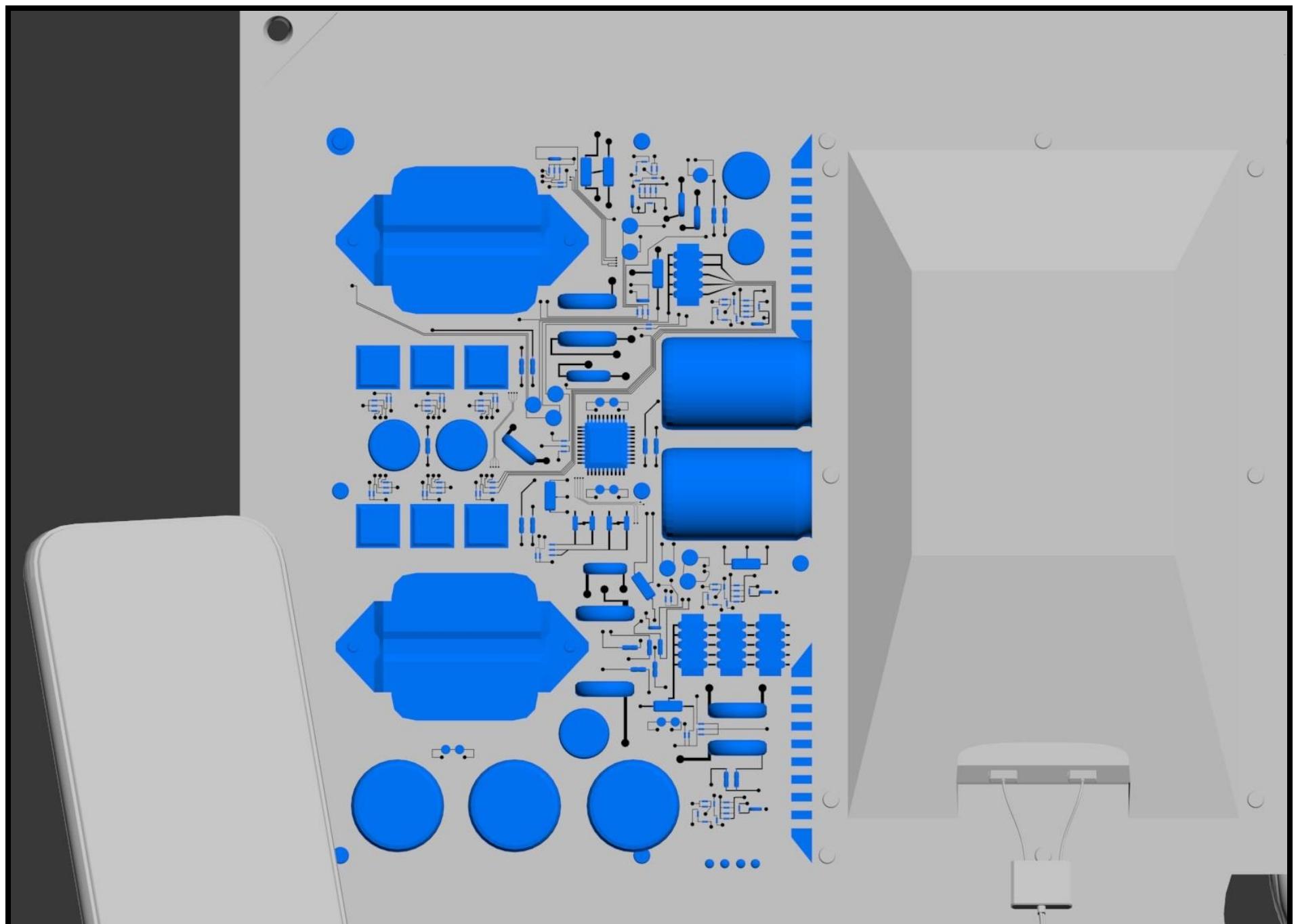


Figure 13

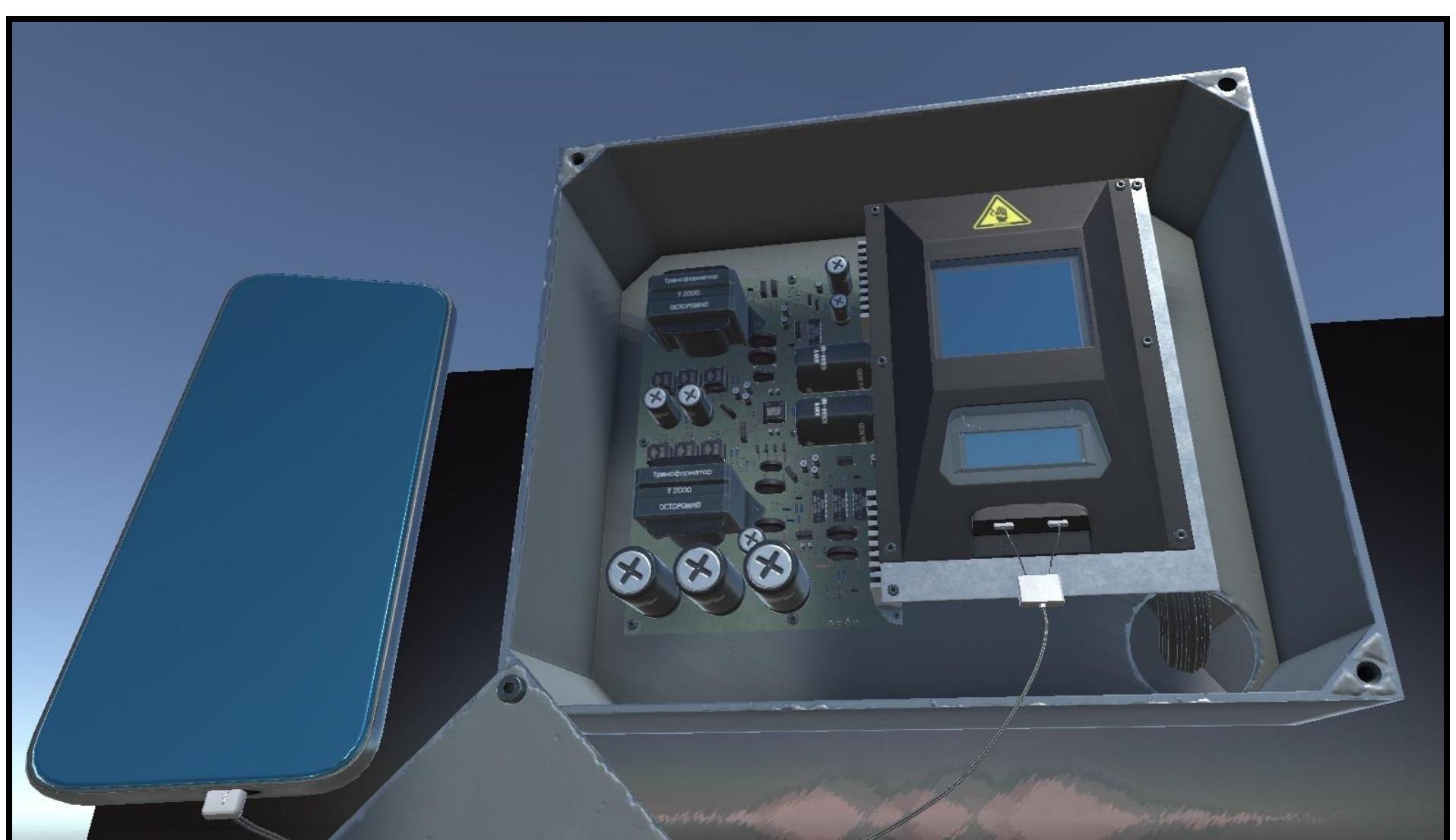


Figure 14

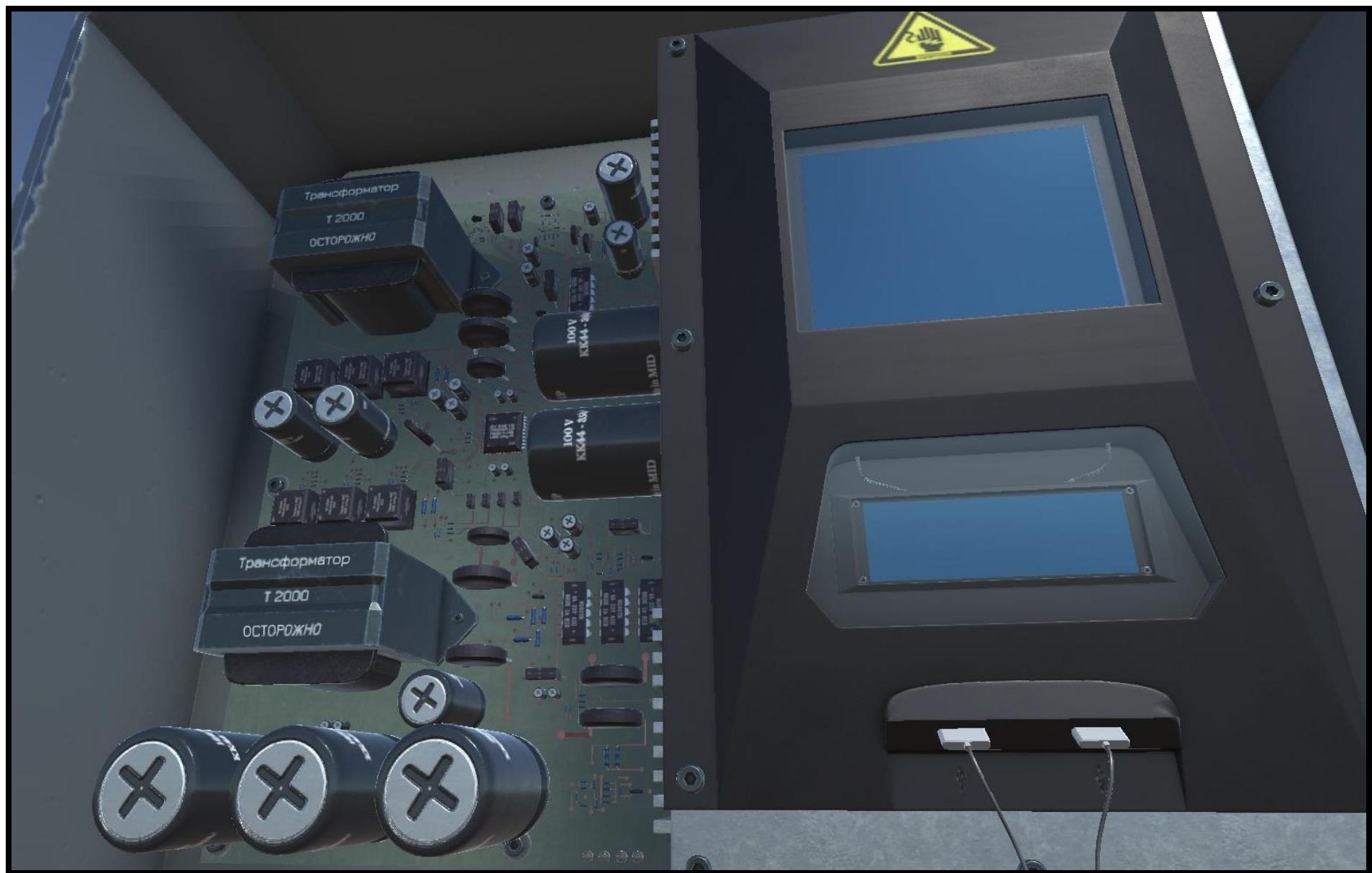


Figure 15

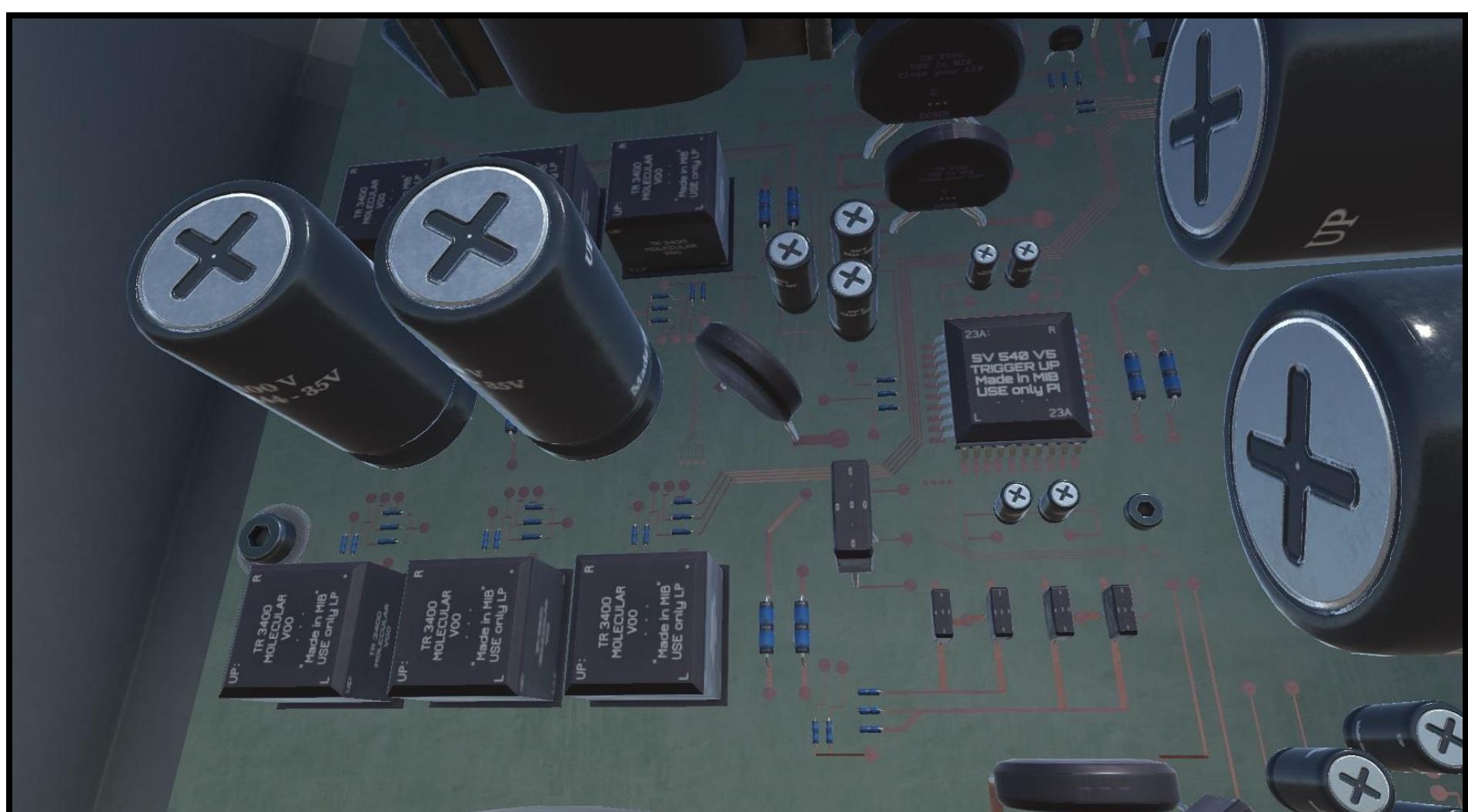


Figure 16

Hacking tool

The phone is used as a hacking tool, all actions will take place on the phone screen (Figure 18).

Background on the phone can be easily changed to the one you need.

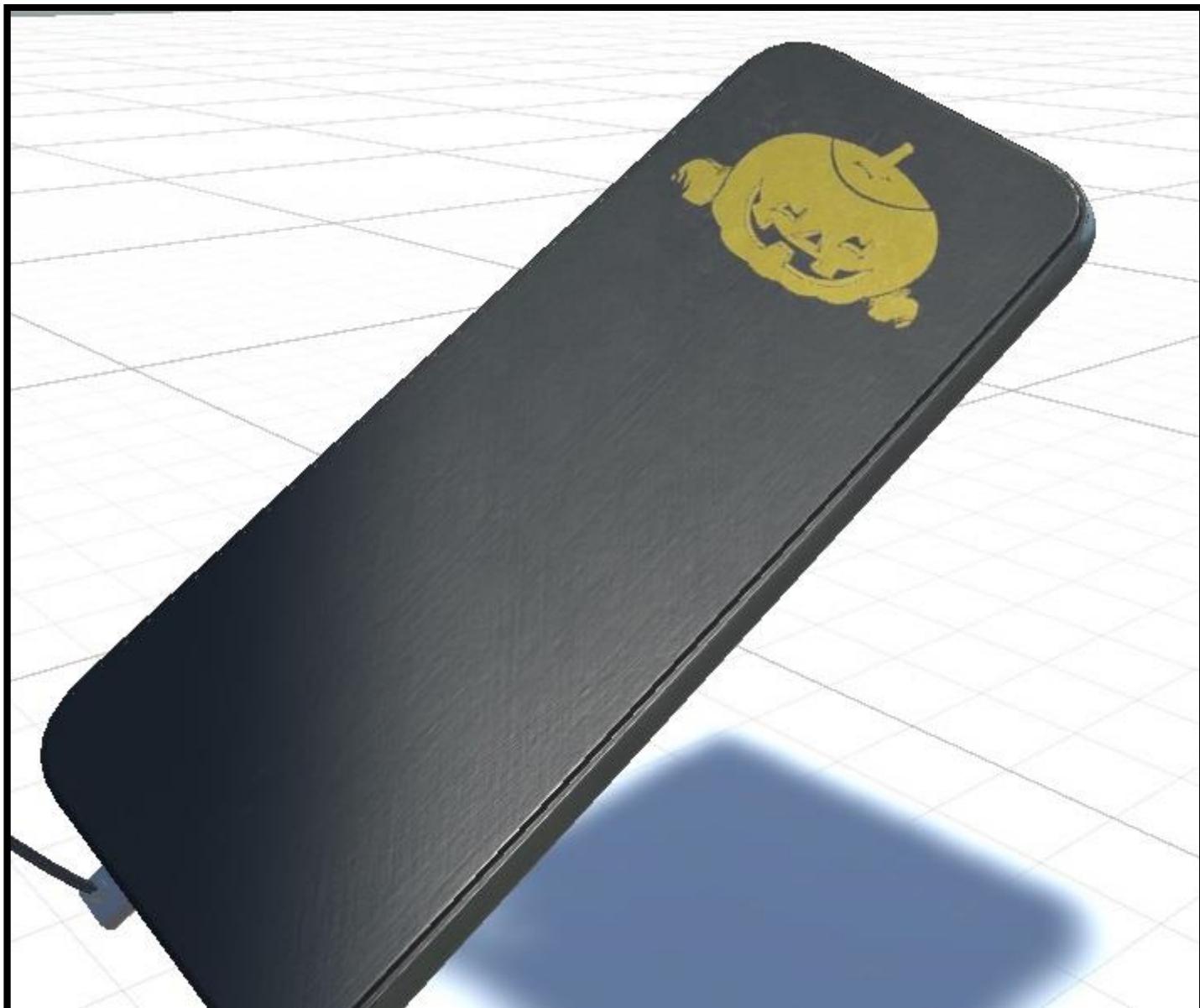


Figure 17 - PumpkinPhone

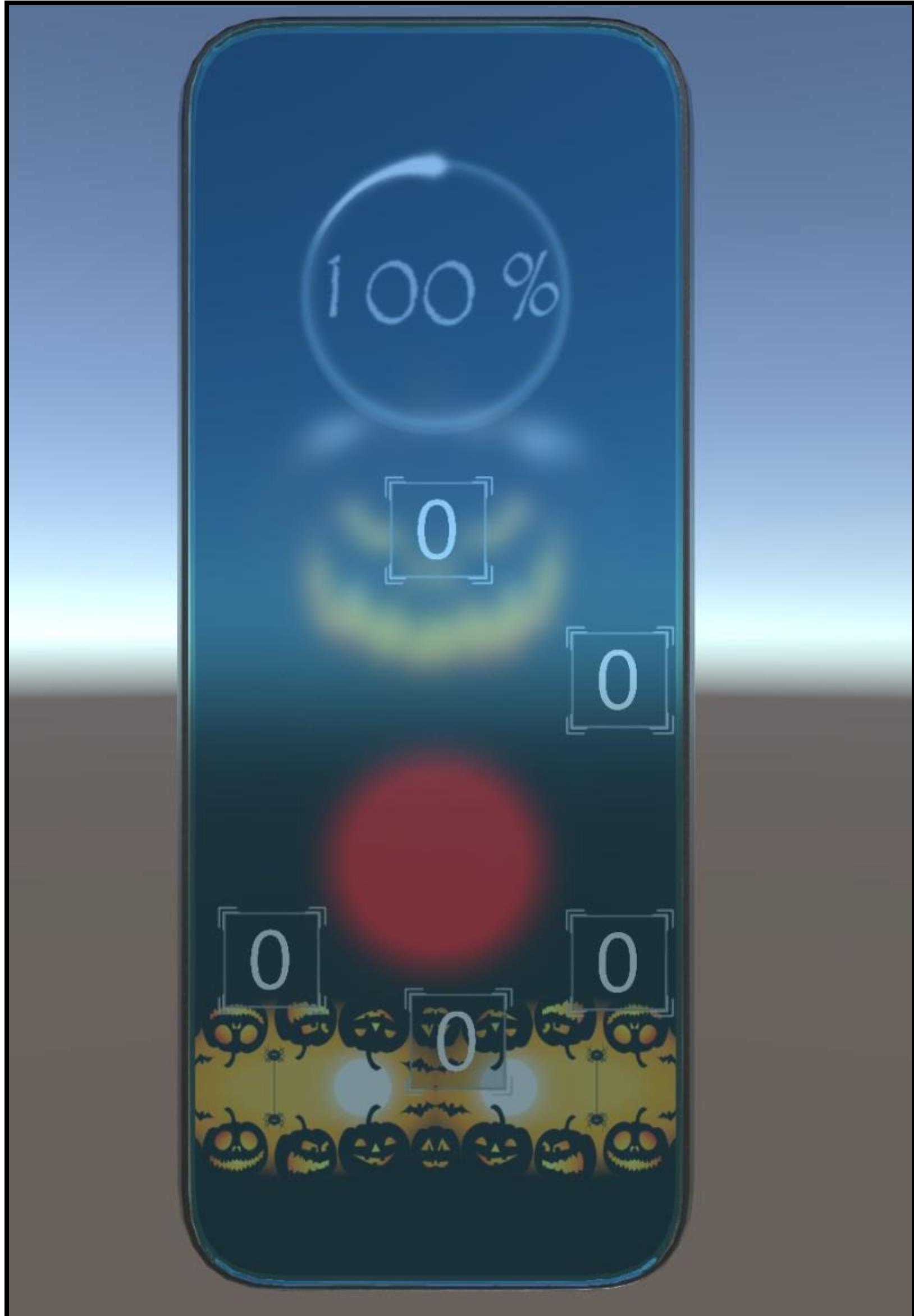


Figure 18 – Screen

System interface

The interface of this system is very specific. I have not seen such interfaces before. In Figure 19, we see on the left a field with tips, in the center is the “PumpkinPhone” hacking tool, a box on the right with various electrical devices. The box has a countdown timer, the time depends on the level of difficulty. In addition to the timer on the box there is an additional screen, it is not working (use it), and shows only the inscription “WARNING”.



Figure 19 – System interface

All of it starts with the download icon (Figure 20) on the phone screen. During loading all elements of the system are not available (except the “Esc” key - exit).



Figure 20

After the “system” is booted, the buttons with numbers will appear on the screen and the player must enter the code by successively pressing these buttons. After entering the code, the system will start checking (reconciliation) and if the code is not correct, then the player will have to enter the code again, but from a different combination of numbers.

For example, in Figure 21 we see 3 buttons (8, 2, 3). By successively clicking on them, they change color to green and become inactive until the player clicks on all the buttons. By sequentially entering the code (Figure 21), pressing the buttons 2, then 8, then 3 (yes, I knew the code beforehand by looking at the script), they all turned green (Figure 22), saying that the code was entered correctly. After that, a red button starts pulsing on the phone screen, and you need to click on it to activate the “scan system” and “virus injection” (Figure 23). Upon completion of the scan, the message “Open” will appear on the phone’s screen (Figure 24).



Figure 21



Figure 22

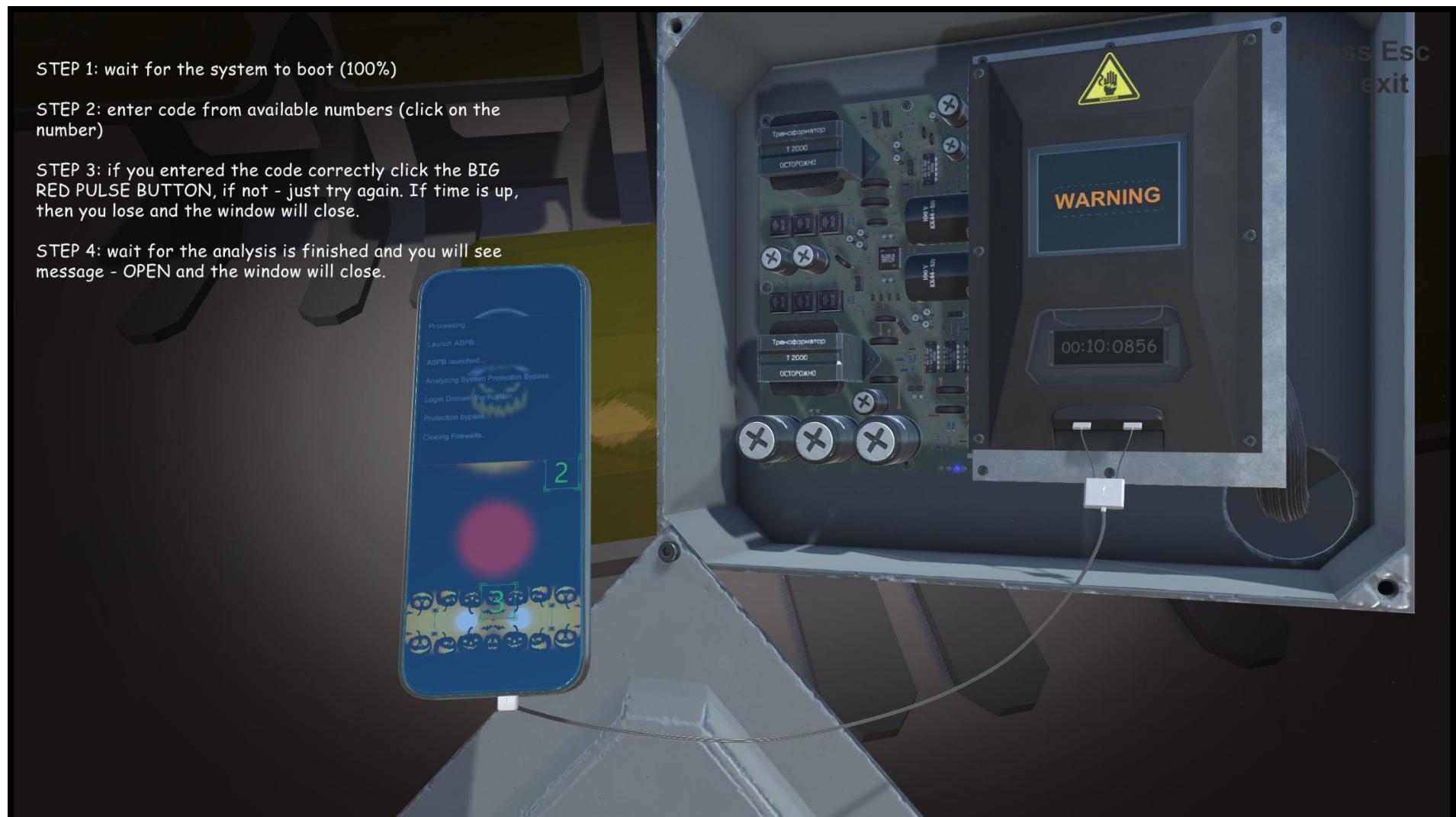


Figure 23

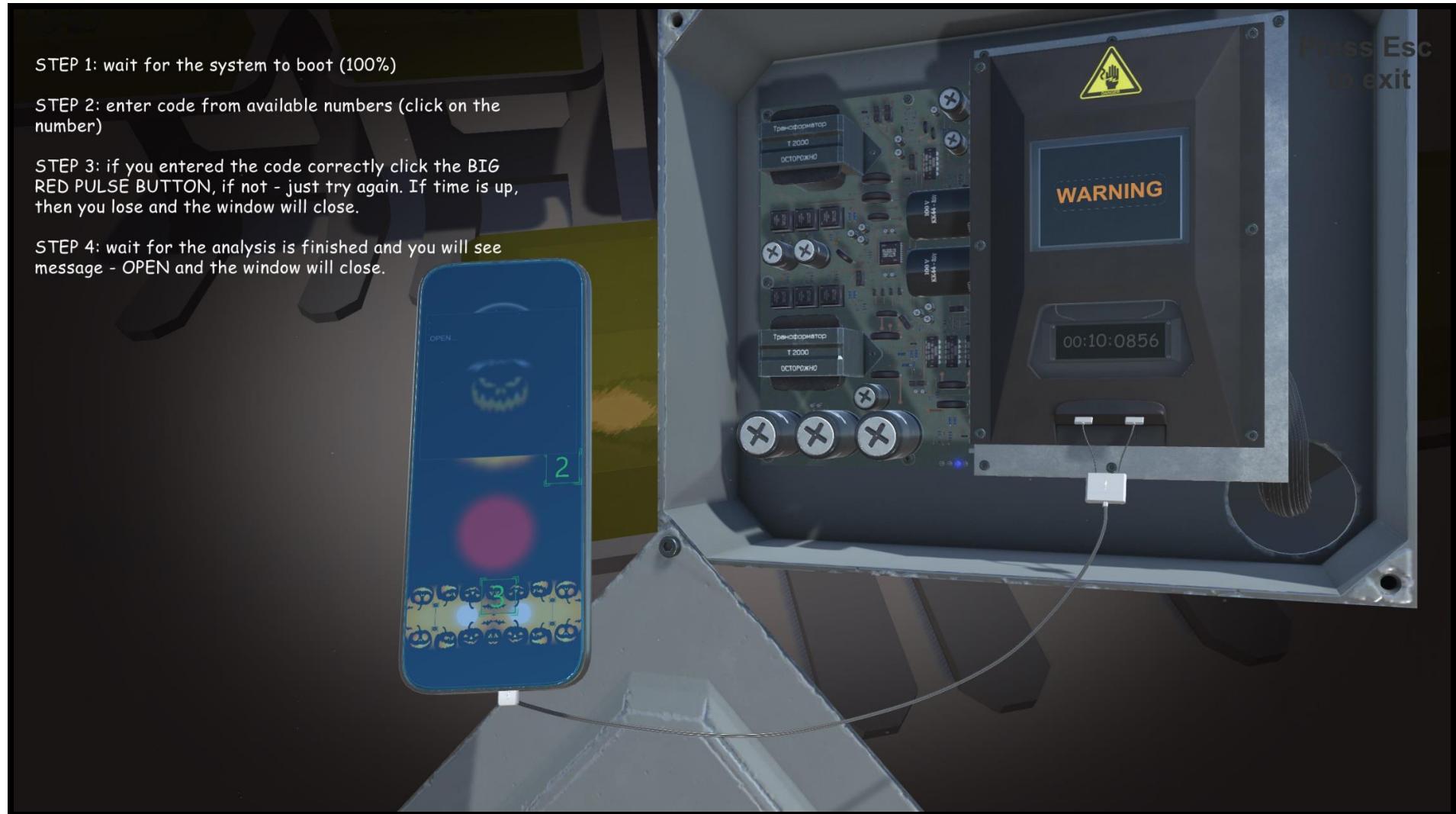


Figure 24

When entering the code, do not forget that the timer is working, and after the time has elapsed, the buttons will turn red, symbolizing that you have lost.

As already mentioned, the time on the timer depends on the level of complexity, as well as the number of buttons (numbers in the code), the higher the level, the more buttons and more time is needed to enter the code. Resorting to mathematics, you can count how many code combinations will be required for each level of difficulty.

So, the formula that calculates the number of combinations and n numbers:

$$P_n = n!$$

where is the sign ! - called factorial.

Accordingly, here is the table:

Table 1

Level of difficulty	Amount of numbers in the code	Amount of combinations
EASY	3	6
MEDIUM	4	24
HARD	5	120

For the level of “HARD” it will be hard to do, but Good luck!

Writing about the system interface, in this system, as in the mechanical one, there is some dynamics, for example:

- for both - a particle system imitating randomly flying dust in the background;
- for both - rocking the interface when moving the mouse in different directions, for the electronic system also rocking the box cap;
- for both - dark and transparent background;
- for the electronic system, this alternately turns on and off the light indicators on the electronic board in the box.

Writing of timer and levels. The timer count is set from the set level.:

Таблица 2

Level of difficulty	Time
EASY	13
MEDIUM	18
HARD	23

At the moment, the maximum time is 23 seconds, for the “HARD” level. If you wish, you can increase it; you just need to edit the texture in which the numbers are.

DEMONSTRATION LEVEL

Description

The demo-level consists of 8 closed sections. To get into each section you need to unlock the electronic door lock, after which the door can be opened. On the stage there are all levels of difficulty, saying beforehand to reach the level of "HARD" will be very difficult.

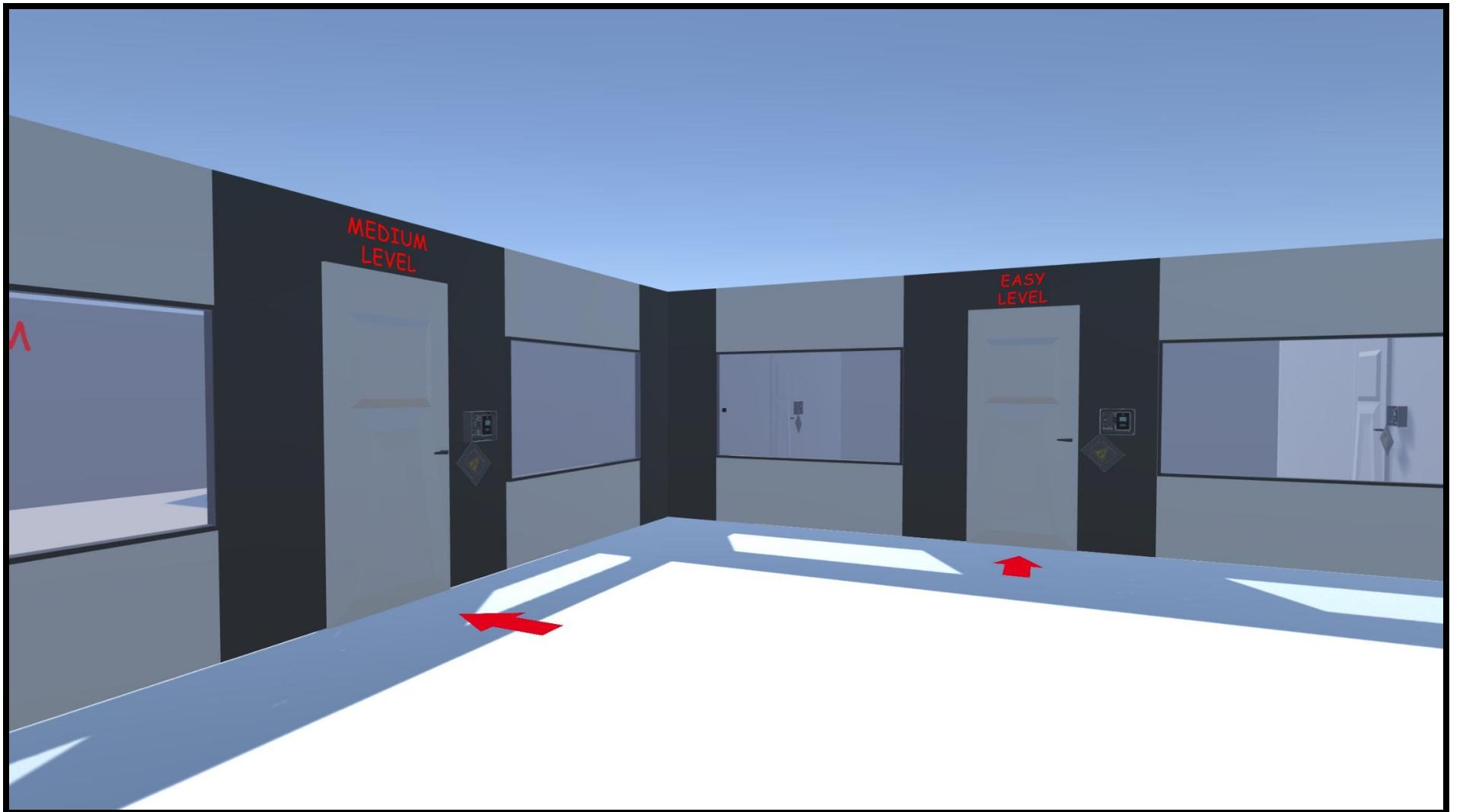


Figure 25

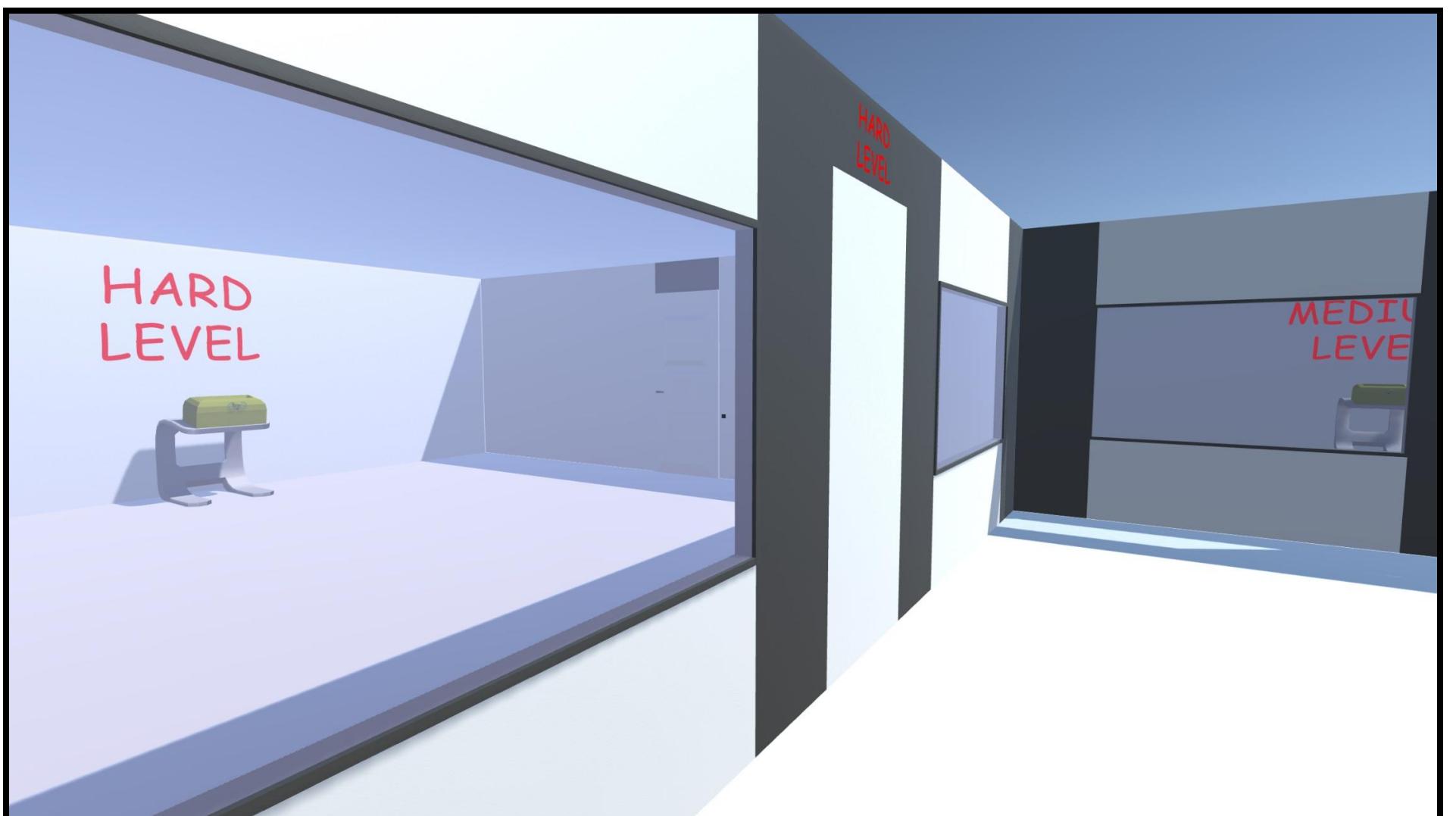


Figure 26

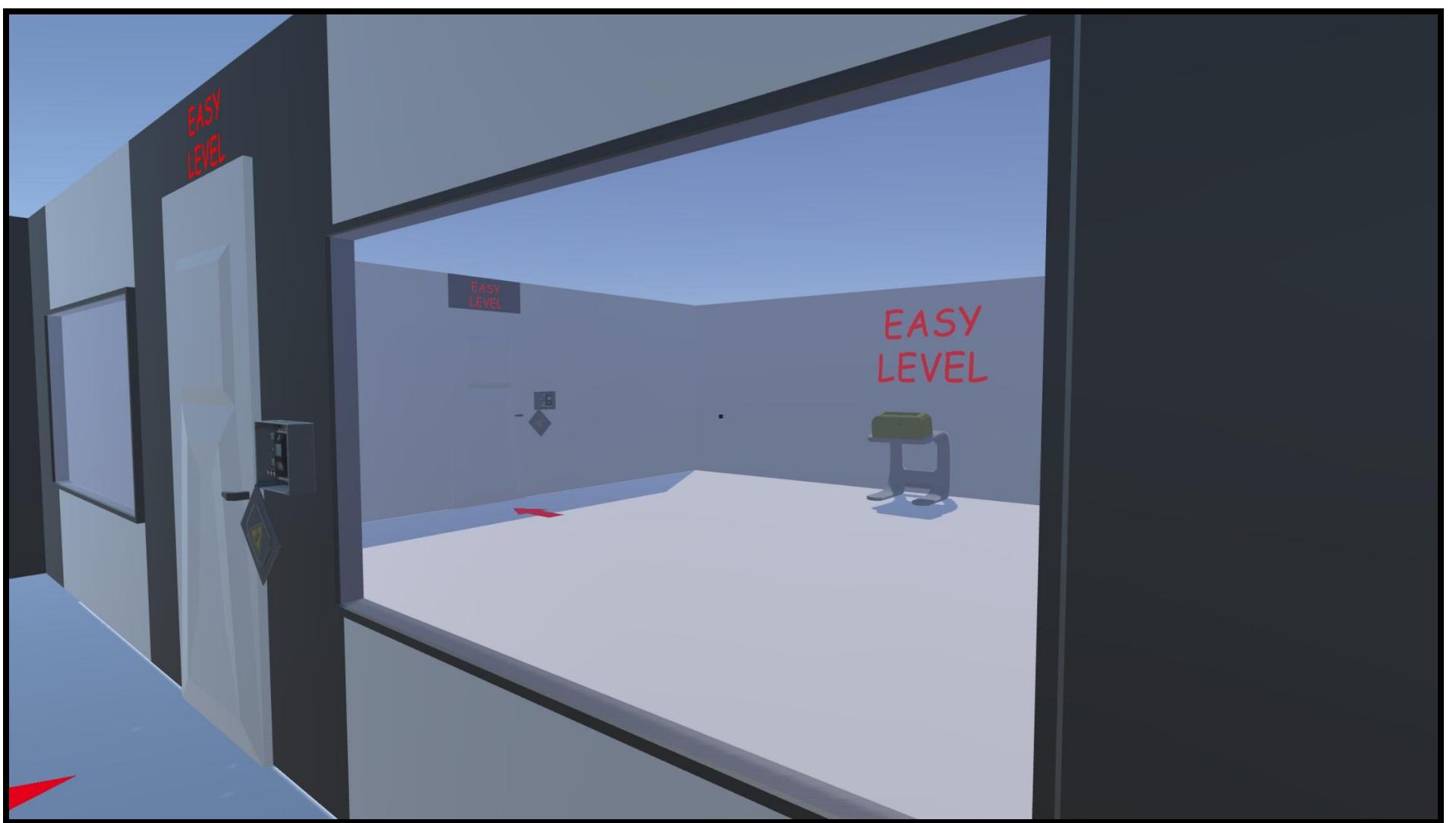


Figure 27

UPDATE 1.1.5

What is new?

In Update 1.1.5, a new interface was added (new lock model) - spin the drum. This is a combination lock in which you need to collect a combination of numbers or words.

In addition, some bugs were fixed.

Spin the drum

The combination lock includes 2 drums - numbers and words:

1. The numbers drum consists of 10 numbers (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).
2. The words drum consists of 10 words (A, B, C, D, E, F, G, H).



Figure 28 – The numbers drum

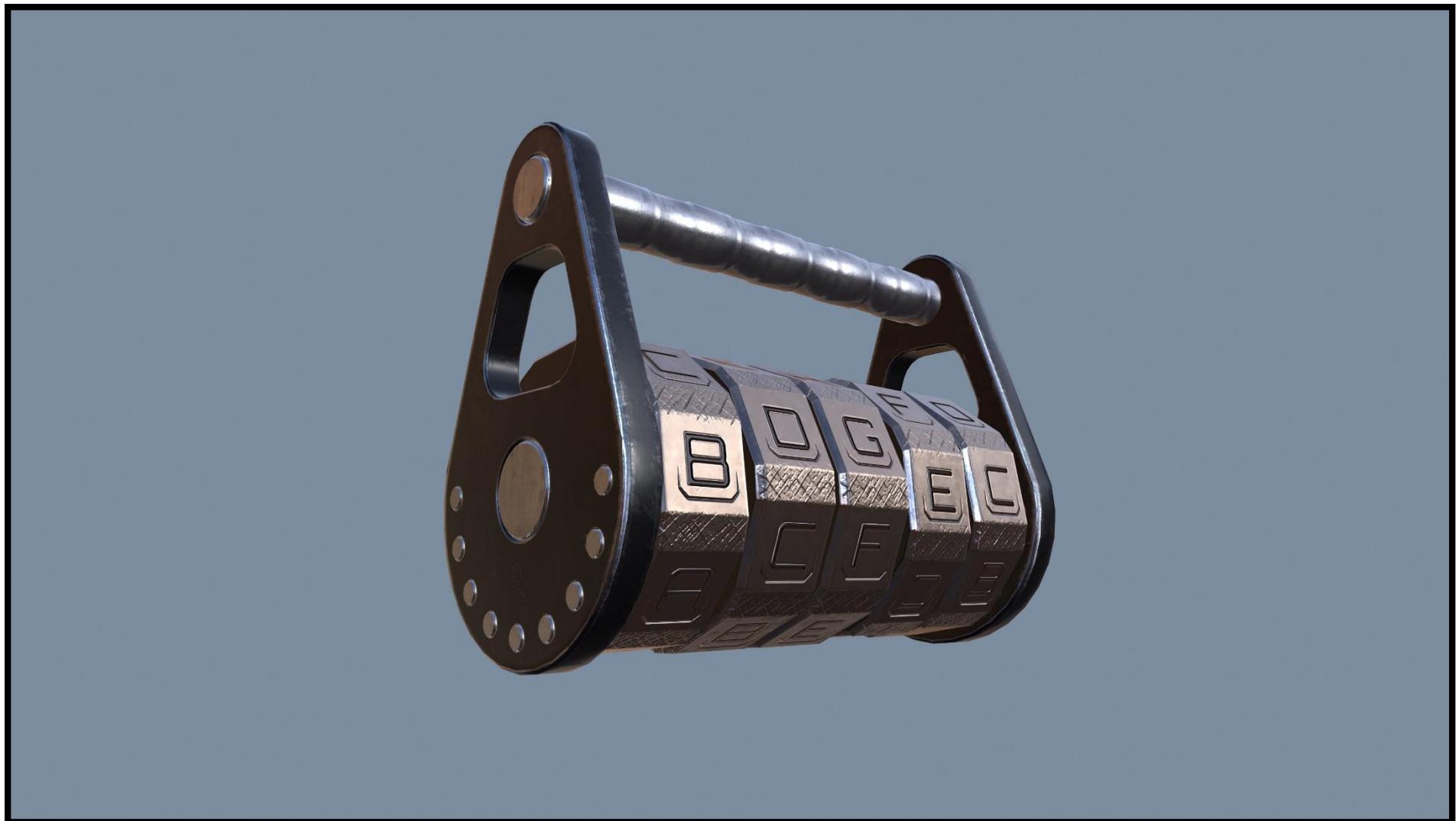


Figure 29 – The words drum

The combination lock has 2 textures with a resolution of 2K:

1. Base or clean texture with slight scratches;
2. Rust textures.



Figure 30 – The numbers drum

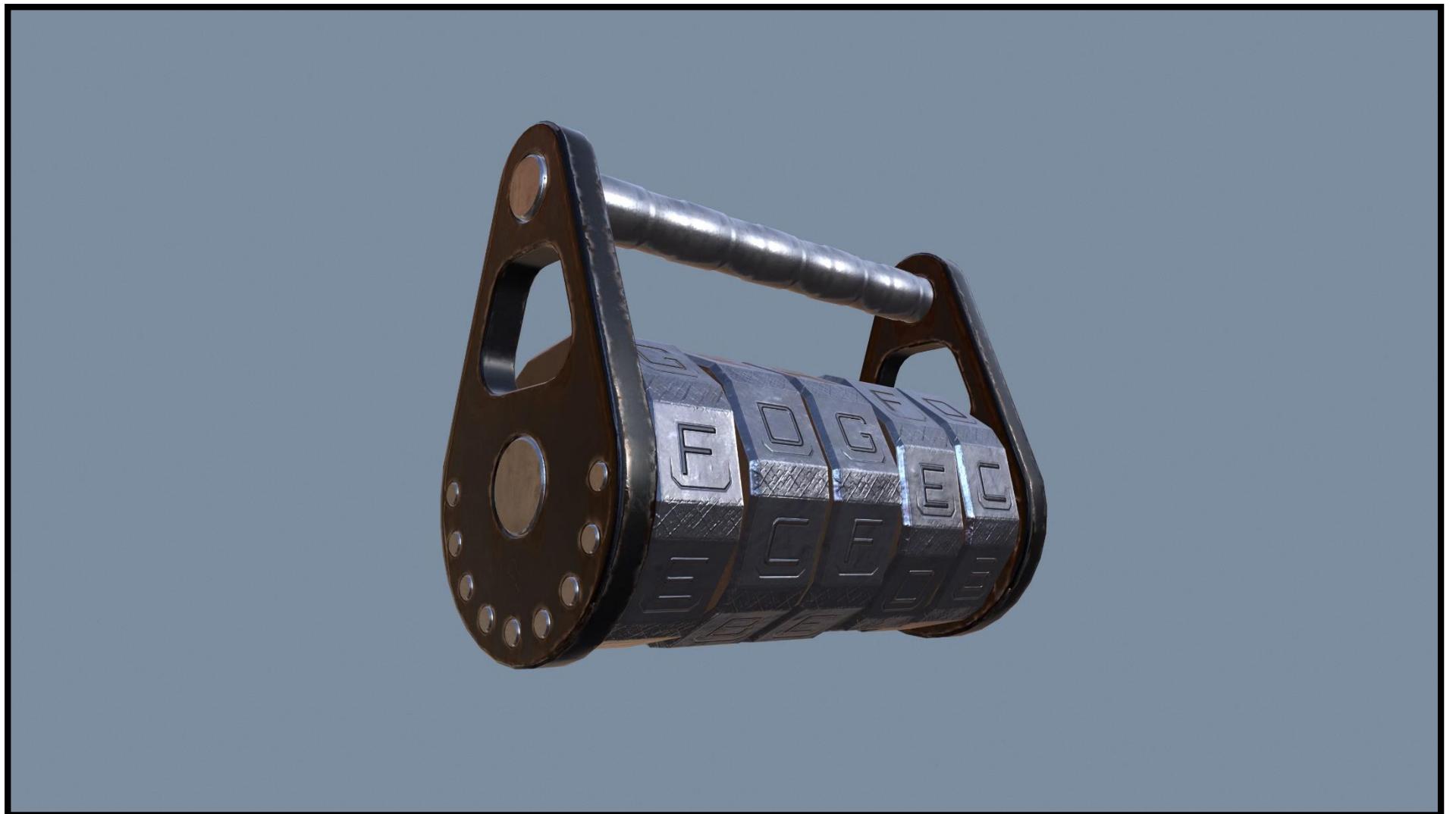


Figure 31 – The words drum

Interface

The code lock interface is very simple - lock and arrows (buttons) appear on the screen, which you need to click on. Amount of drum depends on the level of difficulty, the higher it is, the more drums. After the drum rotates, the scroll click sound is played, if the drum is in the right position, a special click sound is played. And so you need to put together a combination. This is the simplest lock implementation scheme.



Figure 32 – The numbers drum - Interface



Figure 33 – The numbers drum - Interface

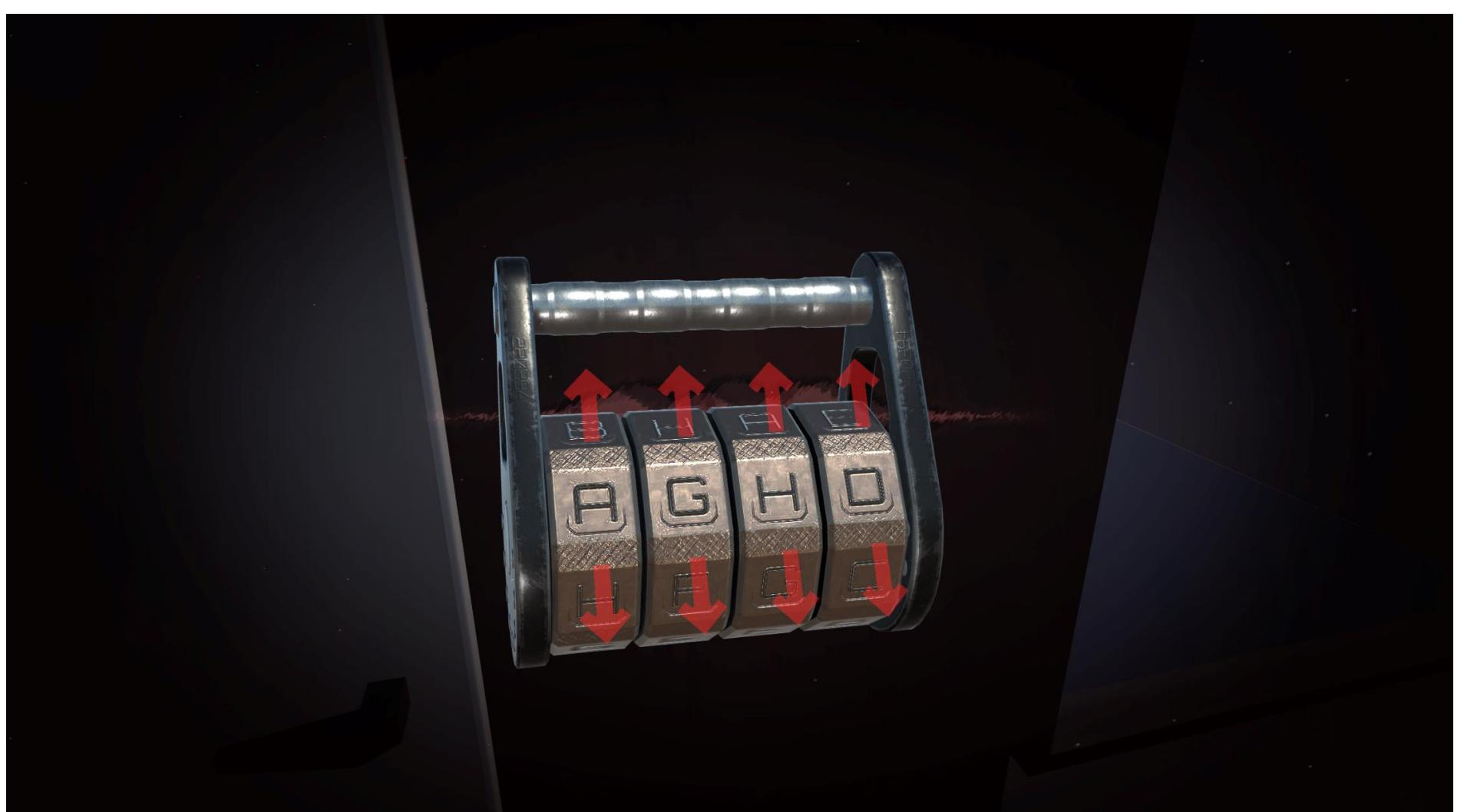


Figure 34 – The words drum - Interface