

# Content:

- Short about application.
- Main menu.
- Levels.
- Saving data.

## Short about application.

I created an application in Unity 3D that is a game with a ball. Unity is a very powerful tool for designing games because it combines the ease of using a visual interface for simulation with the power of writing custom scripts for objects. In comparison with other tools that I've used before to create games this one is one of the highest level by far.

The game itself is still work in progress so it's pretty far from done, however the core of the game is working, the work that is left is on the art and bug fixing.

When entering the game, you have a menu where you have 2 options:

- Go to shop where you can select skins.
- Go to another menu where you can select the level.

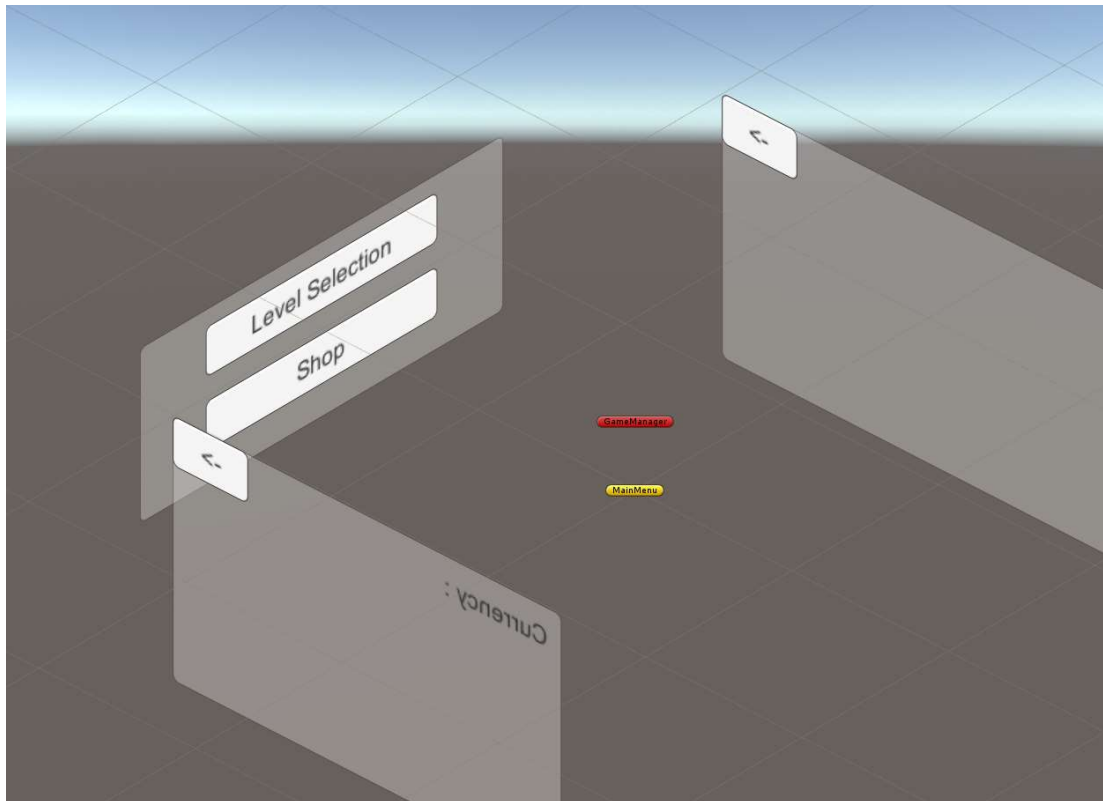
The application has 2 main parts:

- Main menu.
- Levels.

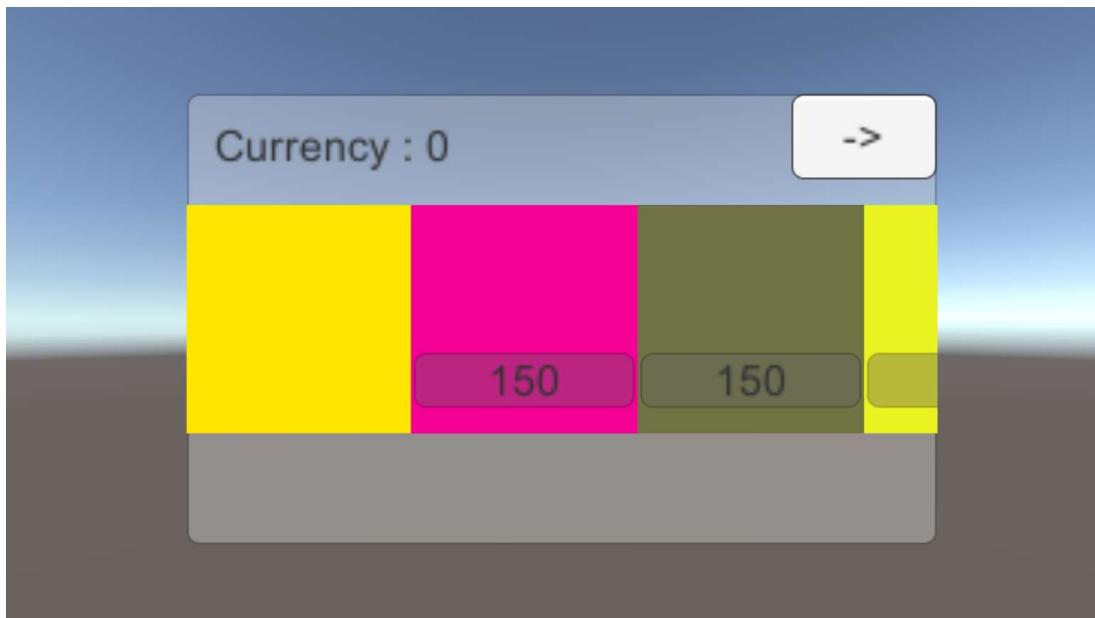
## Main menu

Main menu is a scene consisting of 3 panels. Each panel is a part of a menu. They are placed perpendicularly so when camera moves from one to another you get a nice transition effect.

This is how main menu scene looks from the side:



We can't see the maps and items in shop simply because they are generated at runtime from the files. For example, the shop at runtime looks like this:

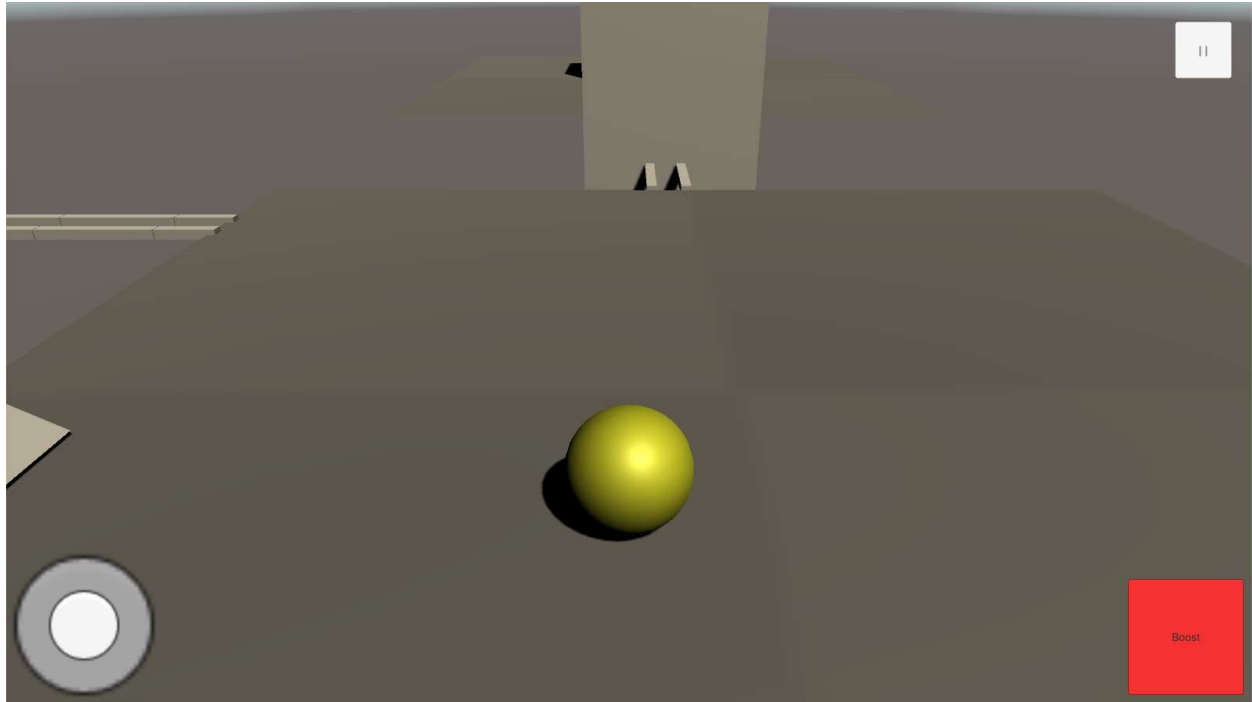


## Levels.

For the levels we create a thumbnail for each level first, then a scene with the exact same name as the thumbnail, because we access the scene by using thumbnail name.

Inside of the level we have the player, that is a sphere that has motoric and collision with every other object on the map.

Also we have a panel that is binded to our camera in that we can see a virtual joystick, a boost button and a pause button.



**Camera** is fixed on the ball, however we can change its position by swiping the screen to left or right.

**Boost** button boosts the speed of the ball by a flat amount (10f). It also has a cooldown of 2 seconds.

**Pause** button enables a panel where you can un-pause or go to the main menu.

We also have environment elements as:

- Breakable walls.
- Push buttons that break walls.
- Block that is the end of the level.

## Saving Data

So we want to save specific data so that we can use it the next time we enter the game or after we finish a level. We have a special class that is initiated at the start of the game and keeps the numbers and modifies them. We also are lucky that unity has built in functions for writing specific data to registry and

getting it by id, in this way when we exit the game the data is saved so that we can fetch it next time we enter.

Based on the saved data we can have specific options as unlocking a new level, or purchasing a new skin using the money we get from completing levels.

## Conclusion

What I liked the most in Unity is that you can create the prefabs of an object, for example of a ball that already has scripts for motorics and collision, and after that we can use the same ball in all of our levels without modifying any code. This is also applied to the entities that save our data. The only downside of making games is that you need a lot of time to complete a game, for example this game is just a result of watching a series of tutorials and it took me more than 12 hours to finish it, meaning that if we were to work with a lower level framework we would have needed twice the time for that.