

# Report

Yifei Wang/yifeiw3/yifeiw3@student.unimelb.edu.au

Zhuolun Lang/zlang/zlang@student.unimelb.edu.au

Yujing Jiang/yujingj/yujingj@student.unimelb.edu.au

Our game is basically a third person game that allows the player to control the ball to pass through all the obstacles and overcome the harsh situation that the environment bring to the ball and finally achieve the goal which is get to the destination. Regarding to instruction for this game, we have a very simple and user-friendly UI for the player that can tap 'Start' on the Menu Scene, which is the first scene the player will see. The scene will be changed to the game scene as soon as the player taps the 'Start' button. It is very straightforward to play and control the ball. We have 'W'/UpArrow for rolling forward, 'A'/LeftArrow for rolling left, 'S'/DownArrow for rolling back and 'D'/RightArrow for rolling right. Since we can only control the ball and our game is designed to be a three-dimensional game, we have 'Q' and 'E' to help player to adjust the perspective from the ball. By pressing 'Q', the perspective will be turned right hand side and the perspective will be turned to the left by pressing 'E'. We have designed different types of ball, which has been made up of different varieties of material such as wooden, iron, bronze, fire and so on to make the player have more choice to play the game. Different balls have different attribute, which significantly affects their speed and friction. To change the ball during the game, just simply press 'C'. There are some harsh situation during the gameplay, for instance, climbing an uphill road or go over the traps. Then what we need to do is accelerate the ball by keeping pressing 'SHIFT' and 'W'/UpArrow.

Talking about the objects and entities have been modeled, those different types of balls are stored as many prefabs and it is easier to utilize when making the function of changing the ball type. The game map is generated by a CSV file. We read the data in the file and using our map Generator script to generate a map in the unity scene as a single game object. In addition, we have a fireball, which has no real objects but a branch of fire, which has zero in weights. Every ball have different attributes according to what they have been made up of.

When we handle the graphics pipeline and camera motion, we make our camera stuck with the ball in a constant distance and by pressing 'Q' or 'E' button, the camera will rotate around the ball as the central 90 degrees clockwise and anti-clockwise respectively. We can make player more comfortable and easier to get to the destination. When the ball has a collision to other objects, we use discrete method to minimize the computation so that we have relatively high FPS.

There are four different types of shaders have been utilized in our game. They are Phong shader, Disappear shader, Vibrate shader and Gradually show shader.

Phong shader is used to help adjust the ambient, diffuse and specular to make the game much more realistic. Disappear shader is created for some special mode which means that the player can only see the environment several seconds

and the whole environment will disappear for 1-2 seconds iteratively so that we make the game a bit harder to play and bring some fresh idea to the player. Talking about the Vibrate shader used in this game, we used it on the words and the button since the whole environment we make is in the universe and it all happens in some different gravity environment. Therefore, we prefer that the words and the button is sometimes not stable and even shaking and falling apart to make it much more real and interesting. The Gradually show shader is used when starting the game. When the game starts, the scene will gradually show up which makes visual experience softer.

The participants are all Melbourne University students and we found 6 participants to play our game. We conduct a survey by paper to make their easier to understand and lead them to give us some useful suggestions.

- 1) There are 4 participants saying that they prefer to have some more effects to make it more interesting and not that quick to get bored from playing the game. Then we add more particle effects to make it more fancy.
- 2) Two Participants mentioned that they would like to play game with some background music. Based on that, background musics for menu and gameplay has been added.
- 3) Most of the participants are not a big fan of vibrate shader in the Menu scene so that the vibrate shader has been eliminated from the Menu scene.