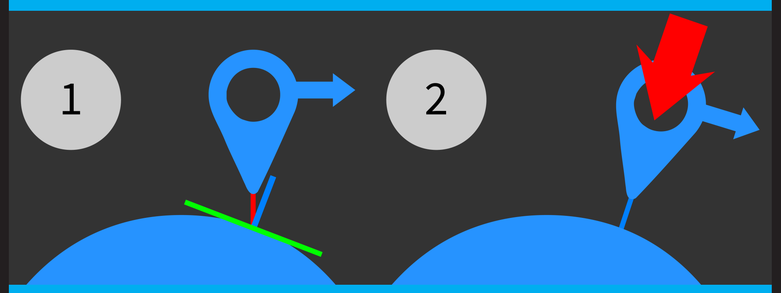
Software Created:

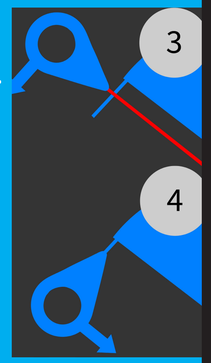
- A **player controller,** which handles keeping the player locked to the surface of geometries
- A **user interface controller**, which allows the user to select things by looking at them (a gaze pointer)
- A product with **8 shapes** using the two above packages



Problems in Gravity:

In order to allow users to walk along surface of shape, we need some way of keeping them stuck to the surface of the shape.

1. Fire a ray from center of player, in their relative down direction (red line in figure
2. Get surface normal (blue line in figure 1). **If ray doesn't hit anything, go to step 4**
3. Rotate player so that their Y axis matches that surface normal (figure 2)
4. Apply force to player in direction of this normal (figure 2). Go to step 1
5. Fire rays that originate below player in their local forwards and backwards direction (Figure 3 and 4). Do steps 2 and 3



Tools Used:

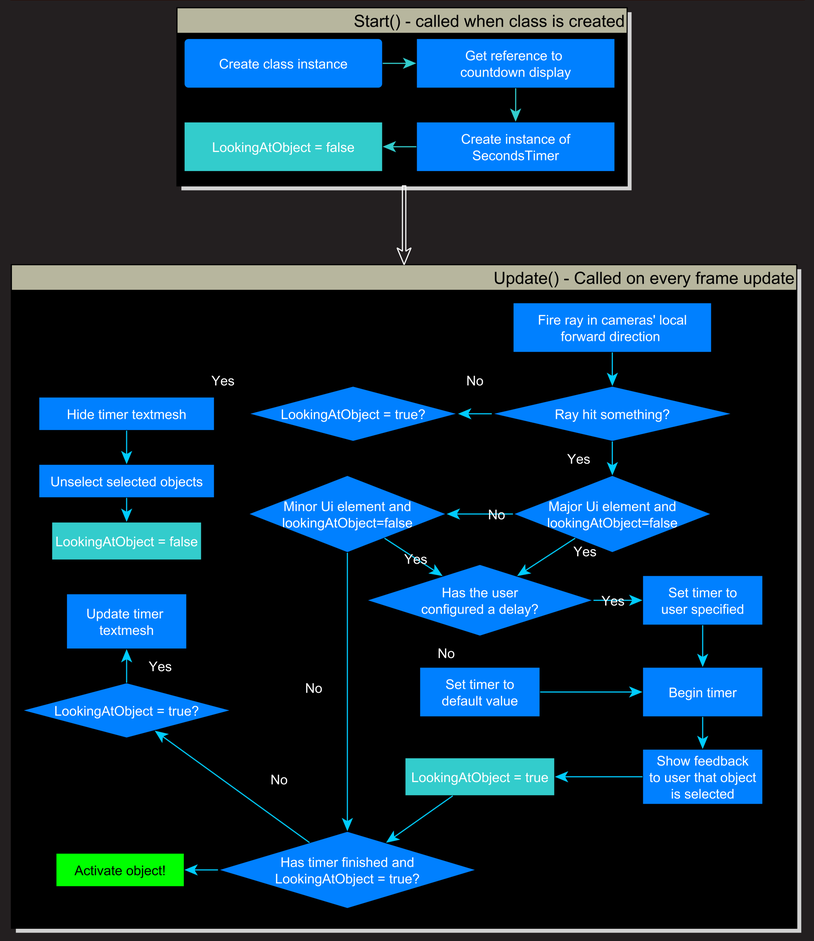
- Unity3D - Game engine and creation tool
- Blender - 3D Modelling application
- C# - Programming language
- Fireworks - Vector graphics designer
- Oculus Rift DK2 - HMD
- Oculus SDK 1.3 (latest)

What If the World Wasn't Round?

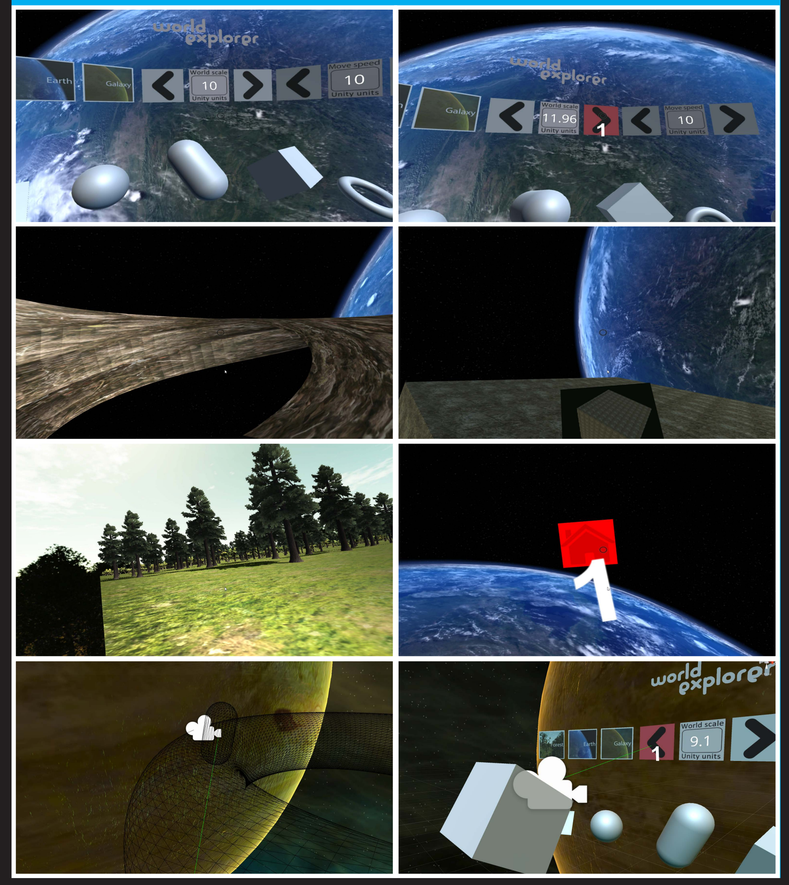
Goal:

Allow the user to explore worlds of arbitrary **3 dimensional shapes**, with the aid of a **Head Mounted Display**.

User Interface on HMD:



In Action:



Results:

1. Provides a unique experience that no other product can offer
2. Very efficient - runs at 75 frames per second on modest hardware
3. Showcases a unique movement controller, not often implemented
4. Showcases good UI design - follows guidelines set out by Google
5. Built so that the assets can be re-used: The UI framework is applicable to **any Unity project built for HMD**
6. User testing showed very positive results!

Try it for yourself!