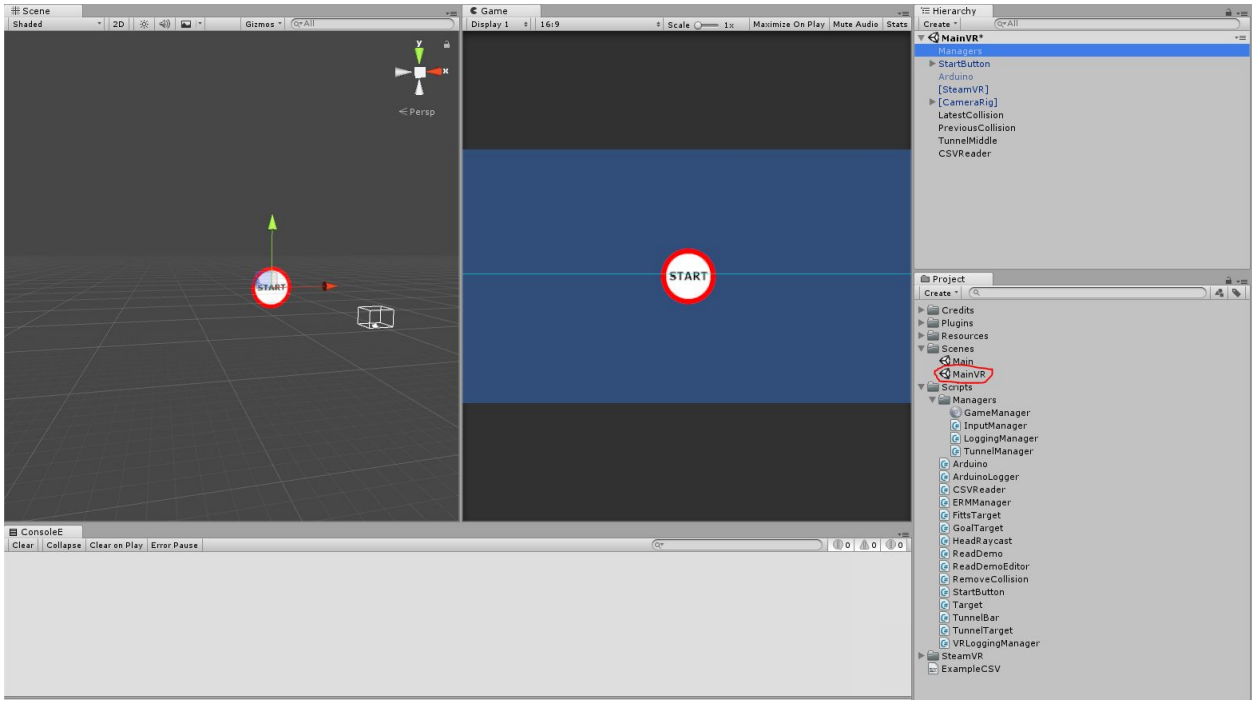
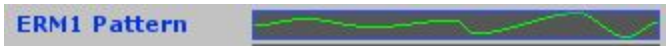


This documents the changes made to the RTII project in order to make it function in a virtual reality environment. First off, this is what objects should be present in the MainVR Unity scene.

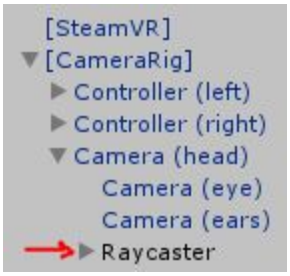
In the **Managers** object, the following two scripts have been added:

ERMManager is not a VR specific script, and may be present in the other unity scene as well. It can be used to play patterns with the ERMs attached to an arduino by creating a curve with the desired intensity, like so: 

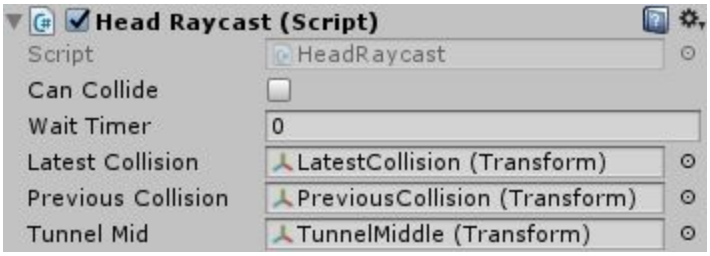
VRLoggingManager is a fork of the *LoggingManager* script, and logs to a subfolder of the folder where the other log files go.

The following data points are logged:
Date;Time;UserID;GameType;InputType;InputResponders;HitType;TargetNumber;TargetID;SessionTime;DeltaTime;TargetPosX;TargetPosY;HitPosX;HitPosY;xDistanceToExit;yDistanceToExit;distanceToPrevious;TargetWidth;TargetGap;Backtracking;ErrorTargetID";

[SteamVR] and [CameraRig] are both gameobjects required by Steam VR, but the **Raycaster** object within the camera is added specifically for this project.

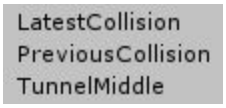


It holds this script, called *HeadRaycast.*

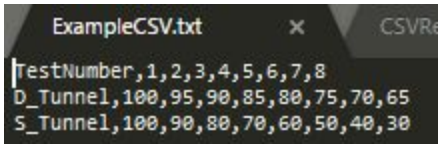


This script takes over for the mouse behavior in the non-VR version of the project. This casts a ray from the head of the person using the project, and whenever that ray intersects with a button or another game object, it triggers whatever code would trigger if the mouse cursor hit that object.

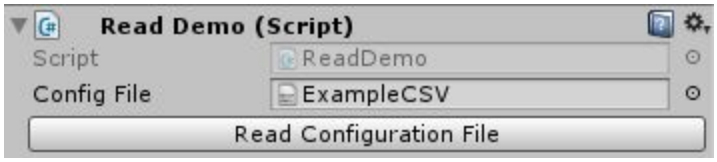
It also keeps track of the location of the gameobjects for the last couple of collisions with the gameobjects:



The final gameobject **CSVReader** allows the project to load information in from a text file. This text files uses the following format:



This example is uses for running 8 tests in a row, for test 1 D_Tunnel and S_Tunnel will be 100 and 100, for test 2 they will be 95 and 90 and so on. To make this work currently, the “Read Configuration File” button must be pressed after the project is started.



The code to make the project load this configuration when running the test is line 446-454 in the *GameManager* script, located in the **Managers** gameobject.

```
446         // Increment Test Number
447         testNumber++;
448
449         string tempDTunnel = GameObject.FindWithTag("CSVReader").GetComponent<ReadDemo>().configuration[testNumber,1];
450         string tempSTunnel = GameObject.FindWithTag("CSVReader").GetComponent<ReadDemo>().configuration[testNumber,2];
451
452         // Set Settings to Configuration Loaded from CSV
453         D_tunnel = int.Parse(tempDTunnel);
454         S_tunnel = int.Parse(tempSTunnel);
```