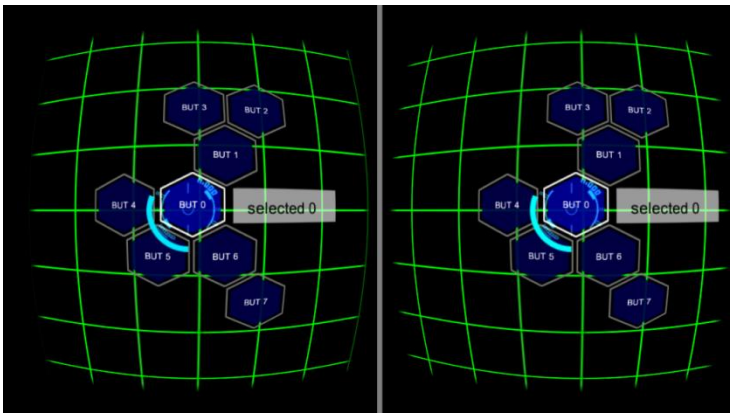
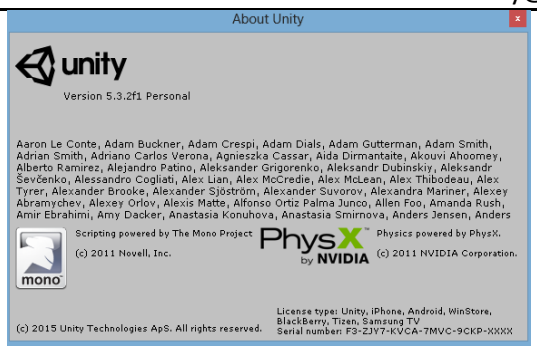
Cardboard VR TouchLess Menu Trigger



1. GENERAL INFORMATION

DATE OF DOCUMENT	22/03/2016
NAME OF THE PROJECT	Cardboard VR TouchLess Menu Trigger
AUTHOR	Michael Soler
UNITY VERSION	5.3.2.F1 PERSONAL
CONTACT	michael.soler.beatty@gmail.com



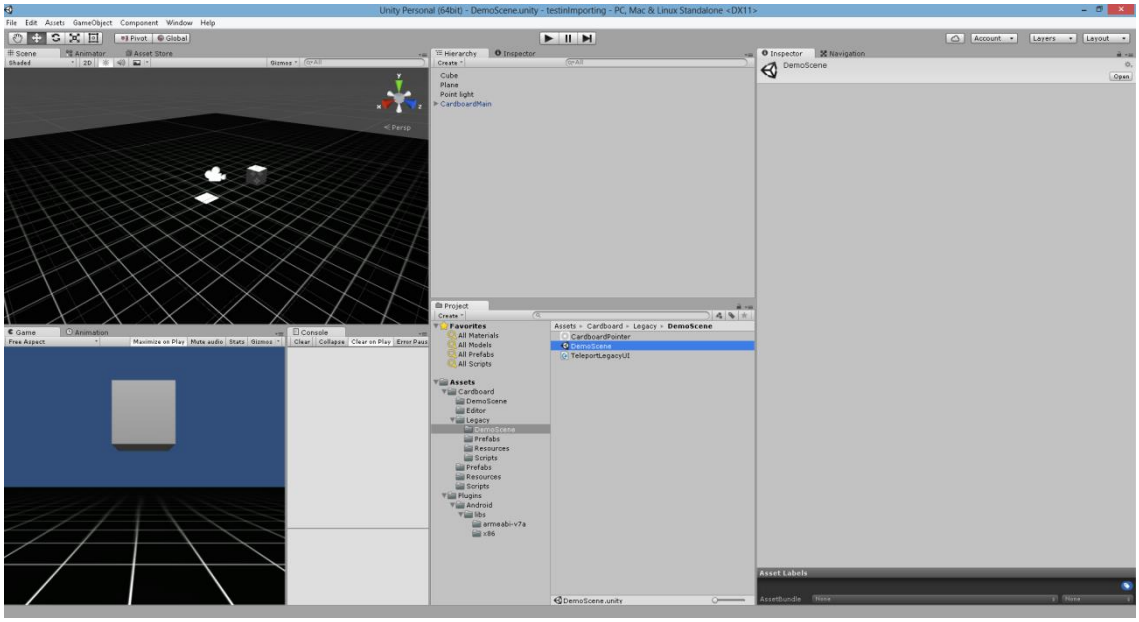
Index

2. IMPORTING INFORMATION

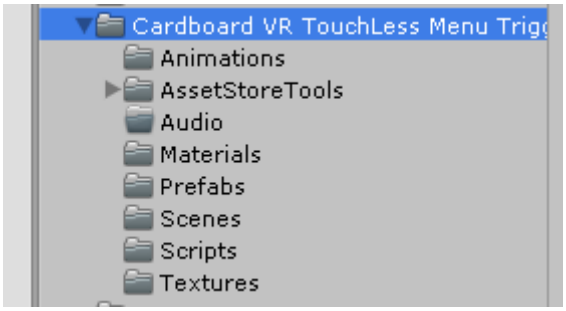
This package works with the "google cardboard" for UNITY that must be downloaded first using the following link:

https://developers.google.com/cardboard/unity/?hl=en

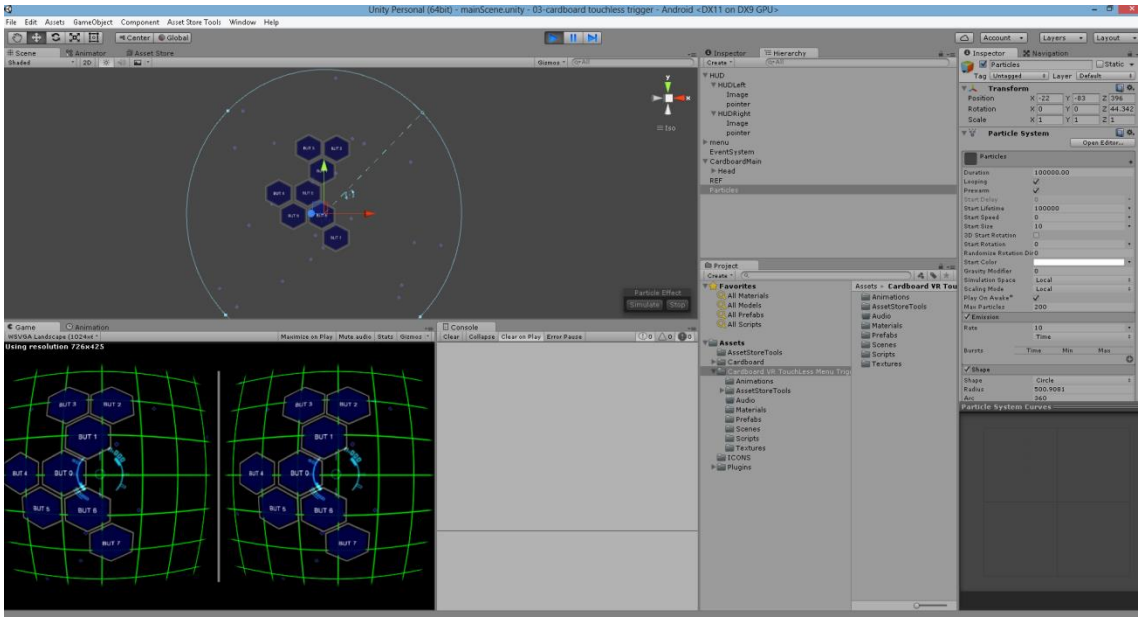
Once downloaded and imported to unity, your project should look like this:



Then, import our package to the project, which will leave you the following folder configuration:



The gaze input collider must be disabled. There is no need to change collider or other game objects.


Unity Personal (64bit) - mainScene.unity - 03-cardboard touchless trigger - Android <DX11 on DX9 GPU>
File Edit Assets GameObject Component Asset Store Tools Window Help
Using resolution 726x425

3. PROJECT DESCRIPTION

This package allows the user to use the VR cardboard headset to select different buttons without having to press the magnetic trigger (touchless). A loading circular bar is displayed before triggering the function inherent to the button. .This package includes:

- Hexagon button style.

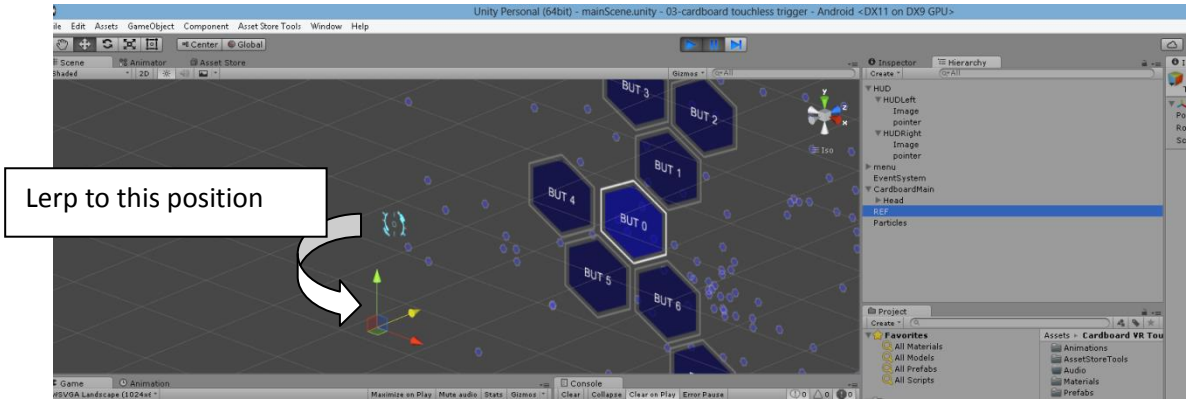
- Scripts to trigger the buttons without touching the magnetic trigger.

- Scripts two smoothly move the player to face the buttons.

-Futuristic environment.

-Simple Text animation.

The program works as follows:



There is a “REF” gameobject that moves in front of the button when the “OnTriggerEnter” event is called. Once the reference has changed, the cardboard object is moved smoothly using “Vector3.lerp()”.

4. LAYERS, TAGS AND COLLIDERS

LAYERS

All objects are placed on the default layer.

TAGS:

All gameobjects are untagged.

COLLIDERS

Check if the “gaze pointer cursor” collider is disabled:

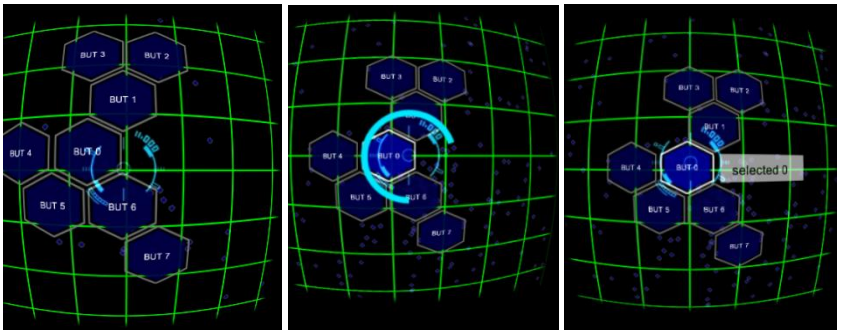
It is important to import the cardboard package correctly, and check if these colliders are working properly.

5. SCRIPTING INFORMATION

We explain each script with some detail in the following table:

- **CircleTrigger.cs:**

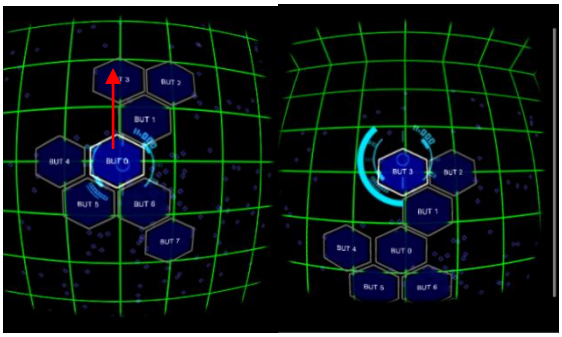
  It is used to trigger the pointer events.



IMPORTANT VARIABLES
public Color start,end, current → these are the colors of the circular crosshair. **public Image CircleImage,CircleImage2** → these are the two images used for the crosshair. **public PointerEventData pointer** → this is the event system's pointer events that is generated when the timer is up. **public bool fixedColor** → sets fixed color to the crosshair.
IMPORTANT FUNCTIONS
public void prepareToClick ()→it is used to set the timer to zero and start the event and shows the loading circular bar. **public void undoClick()**→ stops the timer.

- **SimpleCamFollow.cs:**

  It is used to move the cardboard to the Reference position.



IMPORTANT VARIABLES
public float distance→ the distance between the panel and the user. **public Transform REF**→ the reference transform whose position is changed in the following function.
IMPORTANT FUNCTIONS
MoveREF() → sets the position of the REF transform.