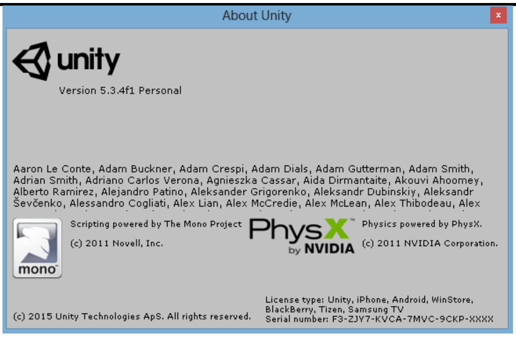
Cardboard VR Simple Sports

1. GENERAL INFORMATION

DATE OF DOCUMENT	16/07/2016
NAME OF THE PROJECT	Cardboard VR Simple Sports
AUTHOR	Michael Soler
UNITY VERSION	5.3.4.F1 PERSONAL (NEWER VERSIONS DO NOT WORK)
CONTACT	michael.soler.beatty@gmail.com



Version 5.3.4f1 Personal

Aaron Le Conte, Adam Buckner, Adam Crespi, Adam Dials, Adam Gutterman, Adam Smith, Adrian Smith, Adriano Carlos Verona, Agnieszka Cassar, Aida Dirmantaite, Akouvi Ahoomey, Alberto Ramirez, Alejandro Patino, Aleksander Grigorenko, Aleksandr Dubinskiy, Aleksandr Ševčenko, Alessandro Cogliati, Alex Lian, Alex McCredie, Alex McLean, Alex Thibodeau, Alex

Scripting powered by The Mono Project

(c) 2011 Novell, Inc.

Physics powered by PhysX.

(c) 2011 NVIDIA Corporation.

(c) 2015 Unity Technologies ApS. All rights reserved.

License type: Unity, iPhone, Android, WinStore, BlackBerry, Tizen, Samsung TV
Serial number: F3-ZJY7-KVCA-7MVC-9CKP-XXXX

Includes video tutorial at the end of the document

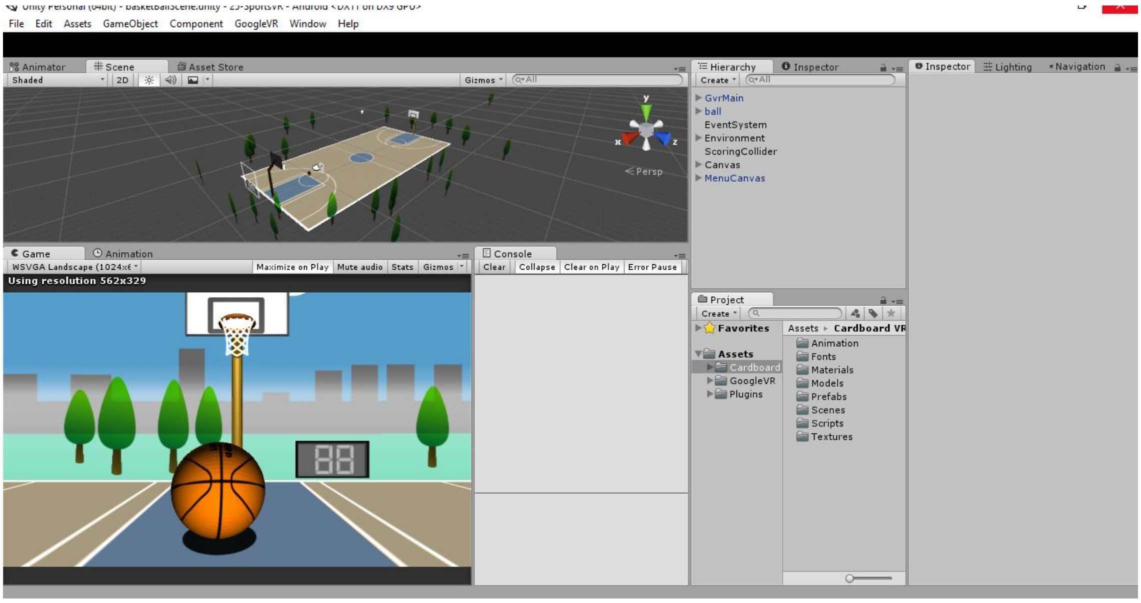
Index

2. IMPORTING INFORMATION

This package works with googleVR SDK=0.8 and can be downloaded from:

https://developers.google.com/vr/unity/download

Once googleVR is downloaded and imported to unity, import our package. Your final project should look like this:



3. PROJECT DESCRIPTION

This asset gives our customers the basic tools to build a VR sports game. It includes two different scenes, the first one is a basketball game, while the second one is a football game. The user is able to interact with the ball using the magnetic trigger (or other means) and he can score by pressing it again. The environment is prepared for VR (low poly count).

New! Try or apk for free!:

https://drive.google.com/file/d/0B5VocdPbQPqVNVF1R1lkOXF1N28/view?usp=sharing

This package contains the following:

-The necessary FBX, textures, models, scripts shown in the “apk”.

-Scripts manage the ball interaction and the scoring.

-Two demo scenes: football scene and basketball scene.

-Complete documentation to understand the principles of each package and full email support at: michael.soler.beatty@gmail.com.

This package needs the following:

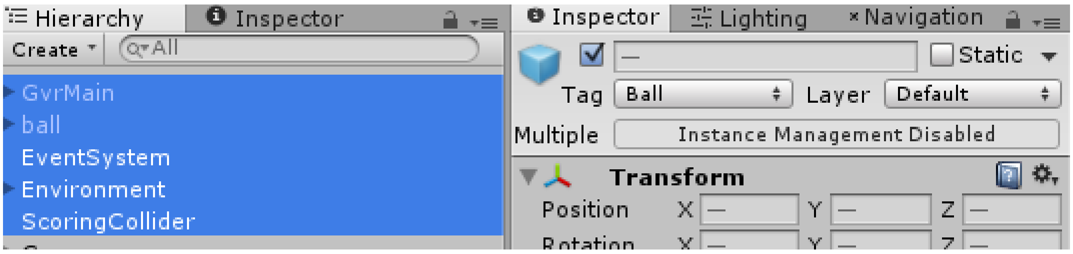
-Google VR SDK=0.8 **https://developers.google.com/vr/unity/download**

-Digital free use font from (placed at “/fonts” folder): http://dl.1001fonts.com/digital-7.zip

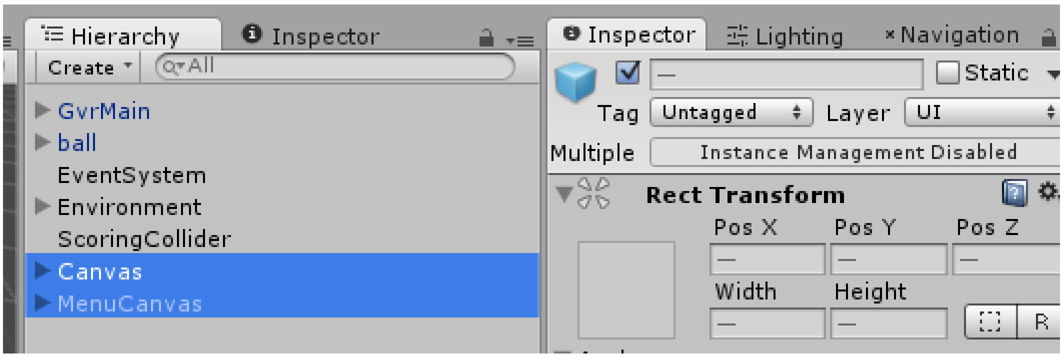
4. LAYERS, TAGS AND COLLIDERS

LAYERS

The only layer that is used is the default for regular gameobjects, and UI for the GUI objects.



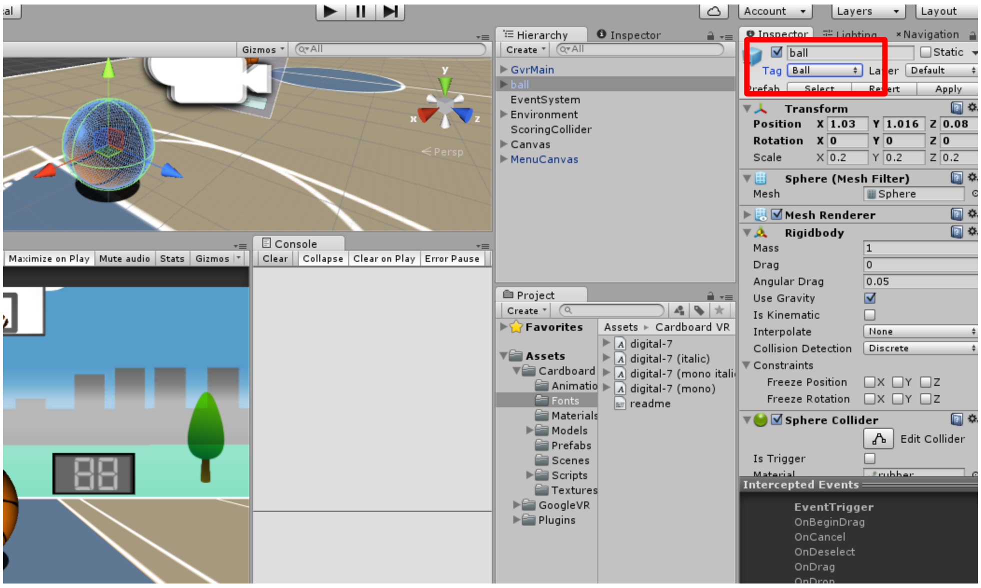
Tag: Ball
Layer: Default



Tag: Untagged
Layer: UI

TAGS:

Please verify that the “balls” on the basketball and football scenes are tagged as Ball. This tag is used on the script to verify if the user is “pointing” at the ball or the environment in order to perform different actions.



ball
Tag: Ball
Layer: Default

Transform
Position X 1.03 Y 1.016 Z 0.08
Rotation X 0 Y 0 Z 0
Scale X 0.2 Y 0.2 Z 0.2

Sphere (Mesh Filter)
Mesh: Sphere

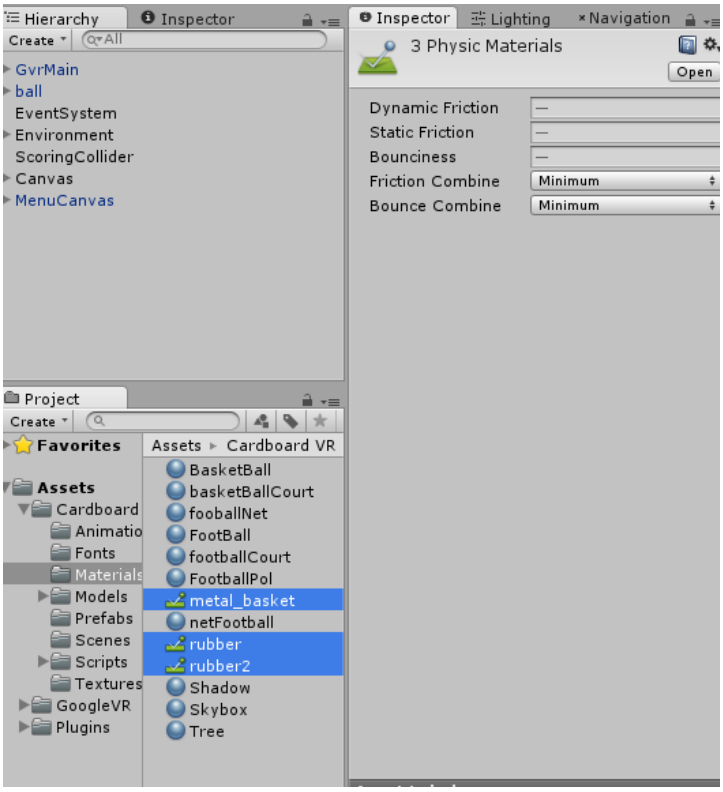
Mesh Renderer

Rigidbody
Mass: 1
Drag: 0
Angular Drag: 0.05
Use Gravity: ☑
Is Kinematic: □
Interpolate: None
Collision Detection: Discrete
Constraints
Freeze Position □X □Y □Z
Freeze Rotation □X □Y □Z

Sphere Collider
Is Trigger: □
Material: rubber

COLLIDERS

- Mesh colliders and box colliders are used for most of gameobjects that are placed as the environment. The balls are using sphere colliders and physic materials in order to create the bouncing effect. You can change the "bouncing restitution" by modifying the following parameters:



3 Physic Materials

Dynamic Friction —
Static Friction —
Bounciness —
Friction Combine Minimum
Bounce Combine Minimum

5. SCRIPTING INFORMATION

We always comment our script on the C# to make developers follow our code better. We have copied the main script’s variables and functions in the following table and we will explain how the package works in a schema:

BallInteraction.cs	
It is the script that allows the player to pick and throw the ball into the basket / or goal.	
Important variables	**Important functions**

// this is making reference to the cardboard object
public GvrViewer *CB*;
// this variable is used to know which ball is selected in case there is more than one
public GameObject *selectedBALL*;
// this is the head gameObject
public Transform *head*;
// this is the speed of the ball when folowing the head movement
public float *displacementSpeed*;
// this is the force value
public float *throwingForce*=20f;
// this is the rigid body attached to the ball
Rigidbody *rigBody*;
// this is the distance of the ball to the player, in case you want to create advanced interactions
float *distanceToPlayer*;
// this is the time parameter, that can be used to check the time between shots
float *elapsed*;
// chose Basketball or football
public bool *isBasket*=false;
	// THIS FUNCTION IS CALLED TO CHANGE THE STATE and CHOOSE A SELECTED OBJECT
	public void *pointerClick*(GameObject go)

Scoring.cs	
It is the script that updates the score	
Important variables	**Important functions**

//this is the public score for the game
public int *score*;
// this is the time variable used to prevent double-scoringpublic float *elapsed*;
// text scoring basket
public UnityEngine.UI.Text *textScore*;
// animation of the net of the basket
public Animator *animNet*;
	//this function updates the socore when ball enters the collider
	void *OnTriggerEnter*(Collider col)

6. VIDEO TUTORIAL FOR ADVANCED TECHNICAL EXPANATIONS

We are trying to give users a more advanced technical service by creating some videos that explain the main aspects of our packages and gives a detailed explanation about how the package works. You will find this tutorial at the following link:

https://youtu.be/ViXAQDfpOpc