



# TP RÉALITÉ AUGMENTÉE : UNITY + VUFORIA - CRÉER UNE IHM EN AR<sup>1</sup>

1. Introduction.....	3
2. Objectif.....	3
2.1 En détail.....	3
2.2 Matériel nécessaire.....	3
2.2.1 Logiciels.....	3
2.2.2 Matériel.....	4
2.2.3 Configuration de la caméra vuforia.....	4
2.3 Exemple de résultat attendu.....	4
3. Les logiciels.....	4
3.1 Prise en main de Unity.....	4
3.2 Prise en main Vuforia.....	5
3.2.1 Créer un compte :.....	5
3.2.2 Remplir le formulaire.....	5
3.2.3 Récupérer la key :.....	5
3.2.4 Créer des éléments associées.....	6
3.3 Principe.....	7
3.4 Exercice de base : Afficher un cube lorsque la caméra AR voit l'image target.....	9
3.4.1 Insérer une caméra Vuforia.....	9
3.4.2 Configurer la caméra.....	9
3.4.3 Insérer une image target.....	9
3.4.4 Configurer l'image target.....	9
3.4.5 Insérer un objet3D sur l'image target.....	10
3.4.6 Configurer le cube.....	10
3.4.7 Tester.....	10

---

1 By S.B. lycée sembat v20201209

4. Création d'une scène avec unity.....	13
4.1 Première scène.....	13
4.2 Création des scripts.....	14
4.2.1 Script du bouton 'quitter'.....	14
a) créer un script : .....	14
b) placer le script dans un gameobject : .....	15
c) associer le script au bouton 'quitter' : .....	15
d) Tester le bon fonctionnement de la fonction.....	16
4.3 Scripts slider et text.....	16
4.3.1 Scripts du slider.....	16
4.3.2 Association entre le slider et le text.....	16
4.3.3 Tester le bon fonctionnement : .....	17
5. Lien avec des données externes.....	18
5.1 Contexte du test.....	18
5.2 Fichier php.....	18
5.3 Script WebTest.cs dans unity.....	19
5.4 Configuration du bouton LireData dans unity : .....	19
5.5 Résultat : ok.....	19
6. Solution récente.....	21
7. Conclusion.....	22

# 1. Introduction

L'AR est la réalité augmentée : le principe est de superposer des informations utiles sur le monde réel.

L'usage en industrie commence à émerger afin de faire de la maintenance assistée. Un technicien équipé de lunettes AR peut être guidé afin de réaliser des tâches complexes nécessitant des aides externes.

# 2. Objectif

Etre capable de créer une IHM en AR en utilisant : unity3D (logiciel de création de jeu 3D) et vuforia (moteur de reconnaissance de forme)

## 2.1 En détail

L'application se compose d'éléments UI (user interface) qui se superposent à la réalité visualisée à travers une caméra.

Les données affichées proviennent d'une base de données.

Des scripts développés en C# permettent l'activation des éléments et les interactions : utilisateurs, réseau, BDD.

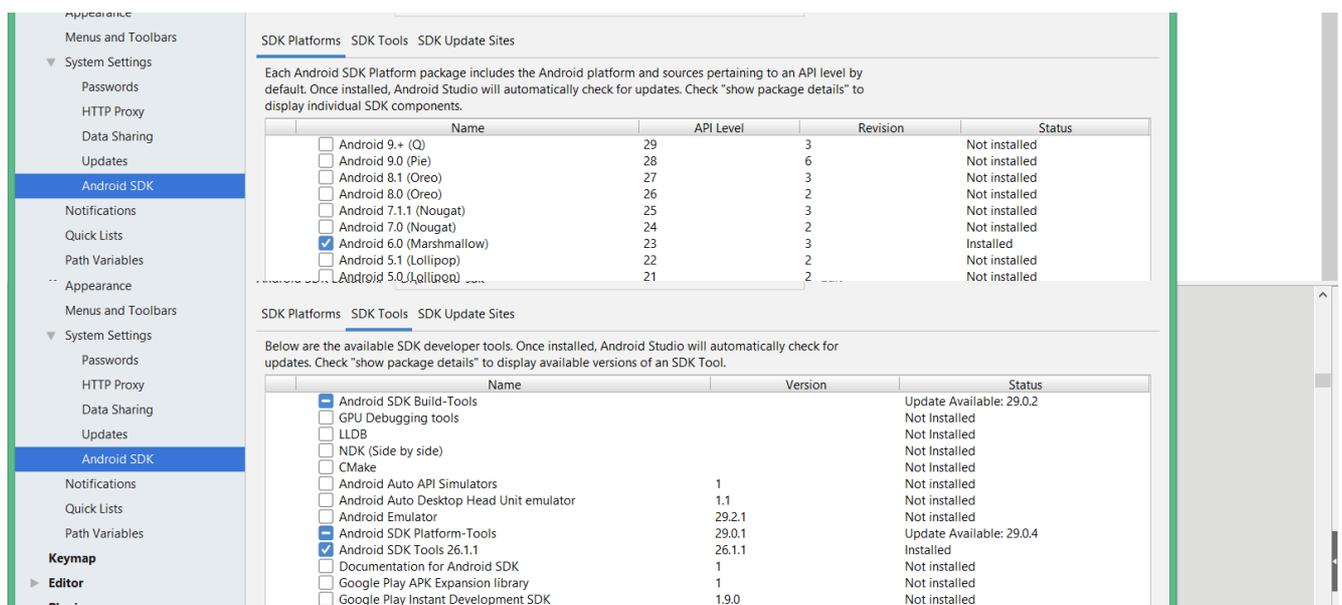
## 2.2 Matériel nécessaire

### 2.2.1 Logiciels

Installer Unity3D 2018 (pour l'IHM) + Visual Studio 2017 associé (pour les scripts en c# avec autocomplétion)

Installer Vuforia à partir d'unity. (ajouter un module à partir de UnityHub)

Installer le SDK android afin de pouvoir compiler pour android : SDK platform (Android6 API23) + SDK Tools 26.1.1



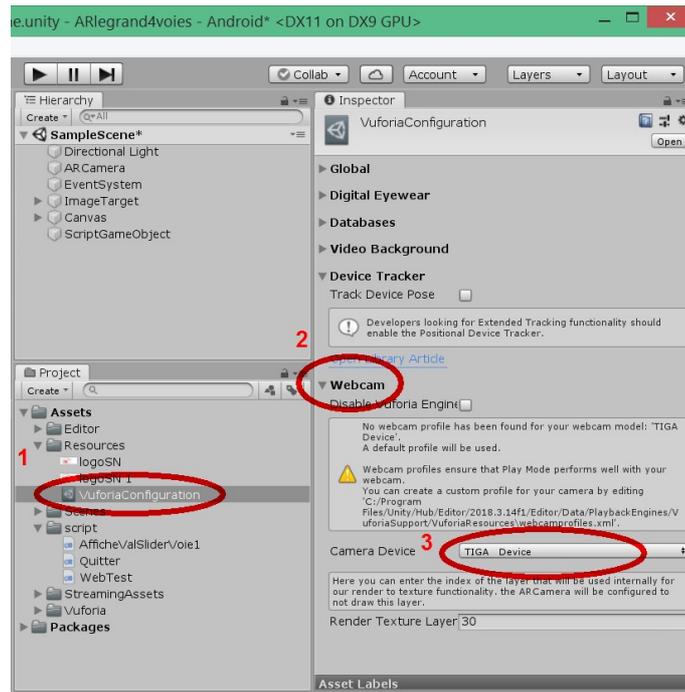
## 2.2.2 Matériel

Une webcam 720p si vous travaillez sur un pc sans caméra.

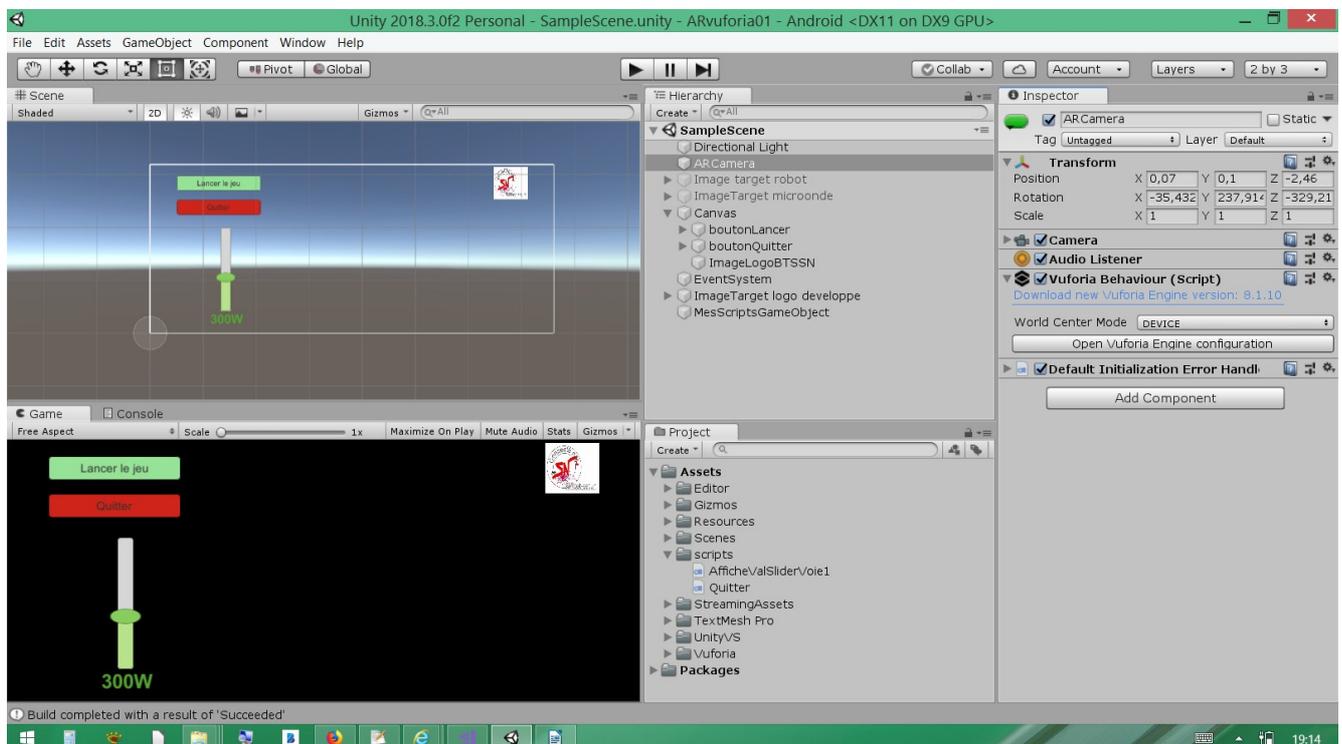
Un téléphone android avec écran 2880x1440 pour la 3D et au moins android 5 (fonction de l'api android que vous installez.)

## 2.2.3 Configuration de la caméra vuforia

Le choix de la camera de test (lors du développement sous windows) se fait dans le menu : vuforia configuration + webcam + camera device (choisir votre caméra)



## 2.3 Exemple de résultat attendu.



## 3.2 Prise en main Vuforia

Vuforia permet de gérer une caméra AR au lieu d'une caméra classique utilisée dans les jeux vidéo : FPS (first person shoot). Cette outil caméra est capable de faire de la reconnaissance d'éléments.

Le document : getting start with vuforia permet de comprendre le principe et l'installation avec Unity.

L'utilisation de Vuforia nécessite la création d'un compte afin de bénéficier de la base de données d'éléments que Vuforia peut reconnaître.

### 3.2.1 Créer un compte :

Aller sur : <https://unity-landing.vuforia.com/> + Register

### 3.2.2 Remplir le formulaire

Créer un licence key (gratuite au 15 mai 2019 mais il semble que ça change !! à voir)

The screenshot shows the Vuforia developer portal pricing page. The page is titled "Develop for Free" and "Ready to Deploy?". It offers three pricing options: Classic (\$499 one-time), Cloud\* (\$99 / mo), and Pro (Contact Us). The Classic and Cloud options are for companies with revenue under \$10 Million/year, while Pro has no revenue restriction.

	Classic	Cloud*	Pro
Get access to standard functionality	Yes	Yes	Yes
Create large and/or dynamic set of targets	No	Yes	Yes
Get access to advanced functionality	No	No	Yes
Price	\$499 one-time	\$99 / mo	Contact Us
License Eligibility	Apps built for companies with revenue under \$10 Million/year	Apps built for companies with revenue under \$10 Million/year	No revenue restriction

### 3.2.3 Récupérer la key :

Exemple de key fournie :

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR

AUjGoEL/////AAABmXwuZgZ3SkP9uoSi1m+wgnttP8P9yfPC8Bn03imHlbY4YqM9gAnmXYZ4h9mDG51XBguKtBPonvvln9JQmejWZCOAcIGO06ZsSV6W5IFe8/CLVp934Fs3sYVCIqA oBJ7OX4YsskvlWgtFAiHTAkZl1qCZZrxCa0Xt0oF1wNu3e1wWBwrKcyJv1qC6K5StgVS97o O6y6MXWGx97dL5LlkobCiEBEbr+ibsiB+f3bKoZuOtBCiqNgQBx/KDIBmcs+FNLkhdJCT5Q gTWTf3LIS3Rbe4mbfhZyMK3HrxIUxQuaFN8pQTiCtivdpqIoeluiY5Mqn6dCOg5g1WH7A NbY2pAsxMe9wwArzQ7jRWVmJvKvM3p

Cette key servira dans unity.

License Manager > **essai01**

**essai01** [Edit Name](#) [Delete License Key](#)

License Key Usage

Please copy the license key below into your app

```
AUjGoEL/////AAABmXwuZgZ3SkP9uoSi1m+wgnttP8P9yfPC8Bn03imHlbY4YqM9gAnmXYZ4h9mDG51XBguKtBPonvvln9JQmejWZCOAcIGO06ZsSV6W5IFe8/CLVp934Fs3sYVCIqA oBJ7OX4YsskvlWgtFAiHTAkZl1qCZZrxCa0Xt0oF1wNu3e1wWBwrKcyJv1qC6K5StgVS97o O6y6MXWGx97dL5LlkobCiEBEbr+ibsiB+f3bKoZuOtBCiqNgQBx/KDIBmcs+FNLkhdJCT5QgTWTf3LIS3Rbe4mbfhZyMK3HrxIUxQuaFN8pQTiCtivdpqIoeluiY5Mqn6dCOg5g1WH7ANbY2pAsxMe9wwArzQ7jRWVmJvKvM3p
```

Plan Type: Develop  
Status: Active  
Created: Dec 31, 2018 17:51

### 3.2.4 Créer des éléments associées

L'onglet TARGET MANAGER permet de gérer les éléments réunis dans des database.

Target Manager > **dbessai01**

**dbessai01** [Edit Name](#)  
Type: Device

Targets (5)

[Add Target](#) [Download Database \(All\)](#)

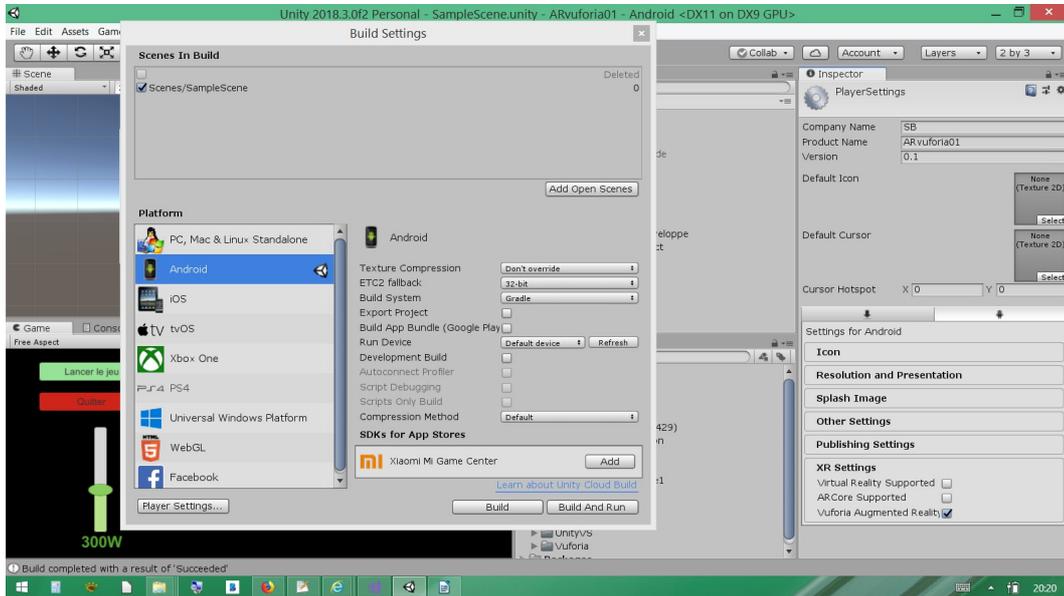
<input type="checkbox"/>	Target Name	Type	Rating	Status	Date Modified
<input type="checkbox"/>	robot	Single Image	★★★★★	Active	Jan 16, 2019 21:43
<input type="checkbox"/>	microonde	Single Image	★★★★★	Active	Jan 16, 2019 21:42
<input type="checkbox"/>	grillepain	Single Image	★★★★★	Active	Jan 16, 2019 21:41
<input type="checkbox"/>	logosn_fabriquepar	Single Image	★★★★★	Active	Dec 31, 2018 17:55
<input type="checkbox"/>	logosn_developpepar	Single Image	★★★★★	Active	Dec 31, 2018 17:55

Last updated: Today 08:14 PM [Refresh](#)

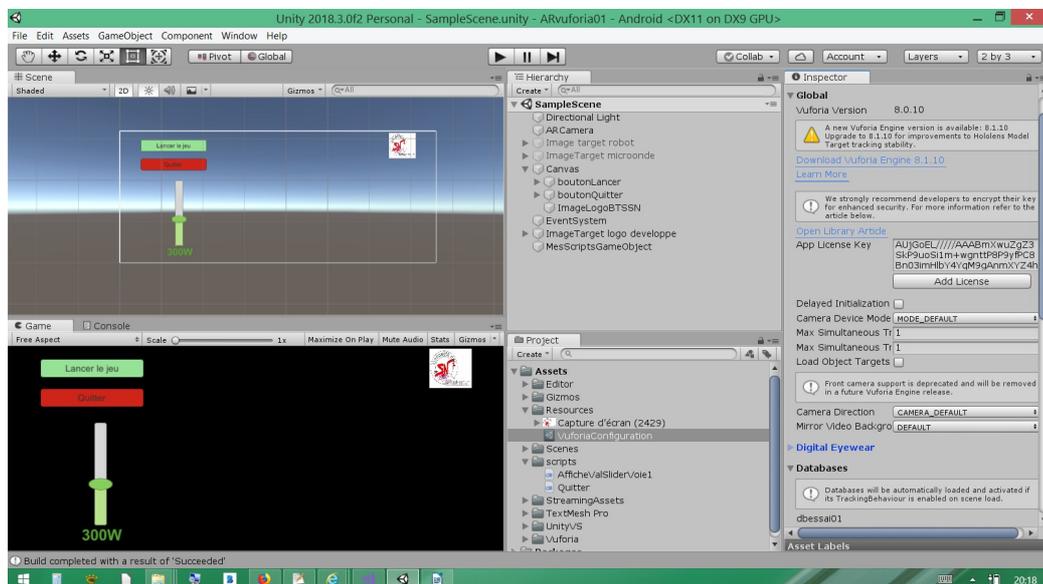
## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR

Cette base sera référencée dans Unity.

Dans unity le lien avec Vuforia se fait en activant XR setting : Vuforia dans Build setting + Player setting + cocher Vuforia augmented reality :



et en configurant les éléments de vuforiaconfiguration :



### 3.3 Principe

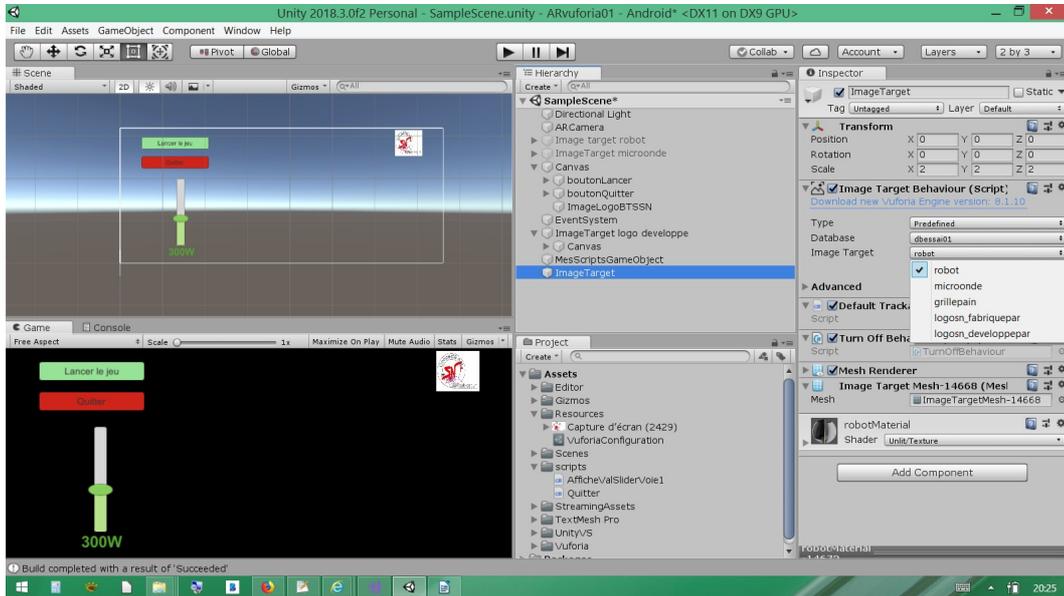
Dans unity la caméra standard est remplacée par la caméra Vuforia :

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR

### GameObject + Vuforia engine + AR camera

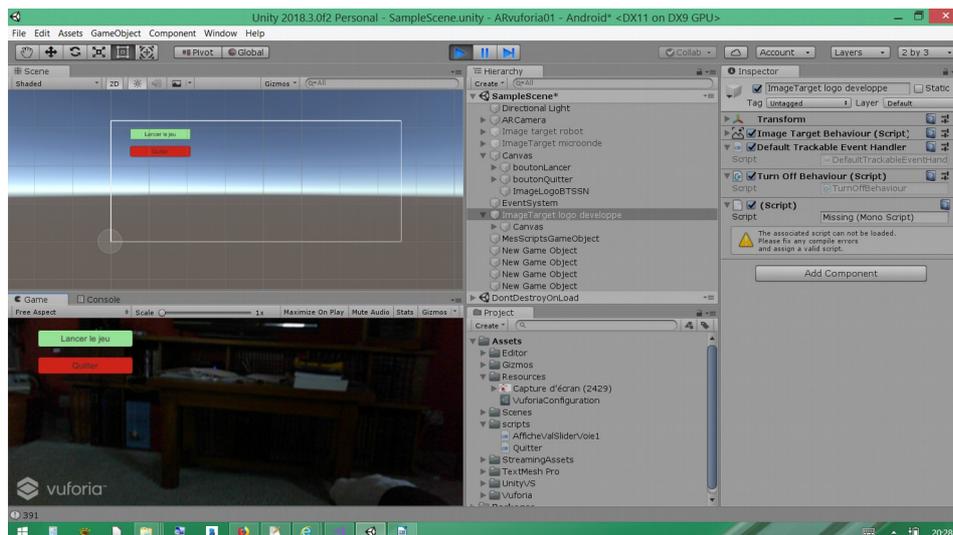
Placer une image provenant de la base Vuforia que vous avez crée : c'est la IMAGE TARGET provenant de GameObject + Vuforia engine + Image

Vous pouvez choisir une des images que vous avez mis dans la database de Vuforia :

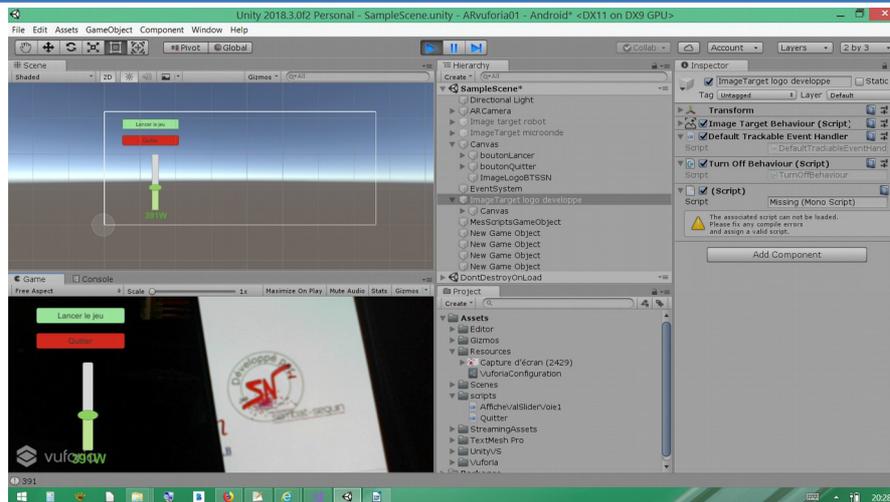


En fonctionnement, lorsque la caméra verra l'image TARGET les objets qui y sont associés apparaîtront en AR sur la camera :

Sans image reconnue :



avec image reconnue :



Le slider vert apparaît avec sa valeur.

## **3.4 Exercice de base : Afficher un cube lorsque la caméra AR voit l'image target**

Faire un nouveau projet

Effacer la caméra installée de base

### **3.4.1 Insérer une caméra Vuforia**

GameObject + Vuforia engine + AR camera

### **3.4.2 Configurer la caméra**

la position 0,0,0

Configurer Vuforia engine :

- 1.Vérifier que la KEY est mise (sinon copier là à l'emplacement prévu)
- 2.Vérifier que la base de donnée est installée (sinon : double cliquer sur le fichier .unitypackage)
- 3.Choisir la caméra.

### **3.4.3 Insérer une image target**

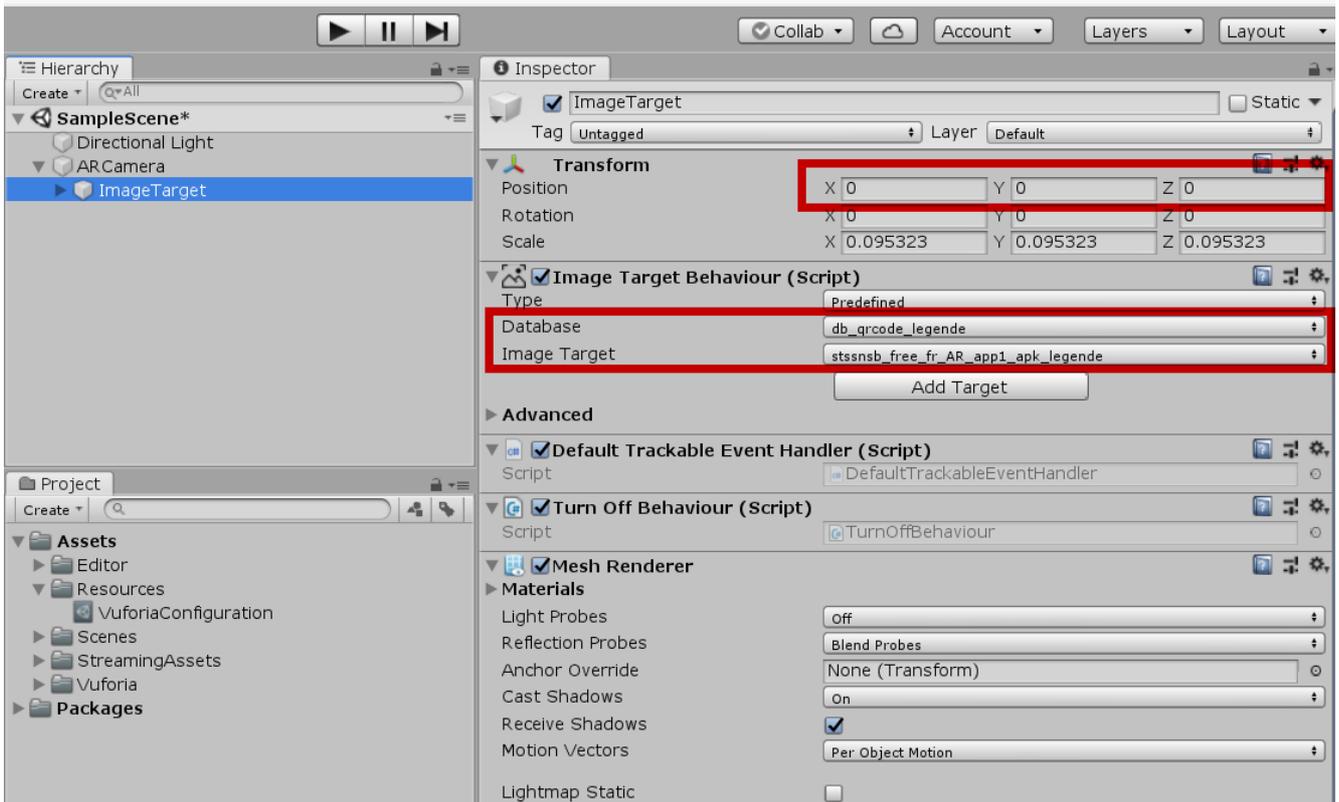
Clic droit sur ARcamera + Vuforia engine + Image

### **3.4.4 Configurer l'image target**

la position

la database

le nom de l'image

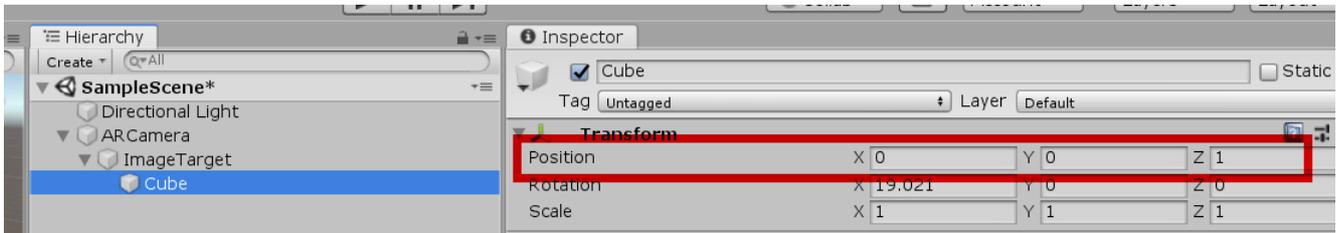


### 3.4.5 Insérer un objet3D sur l'image target

clic droit sur ImageTarget + 3D Object + Cube

un cube est placé

### 3.4.6 Configurer le cube



**IMPORTANT : l'arborescence est importante :**

**L'ImageTarget est sous l'ARcamera**

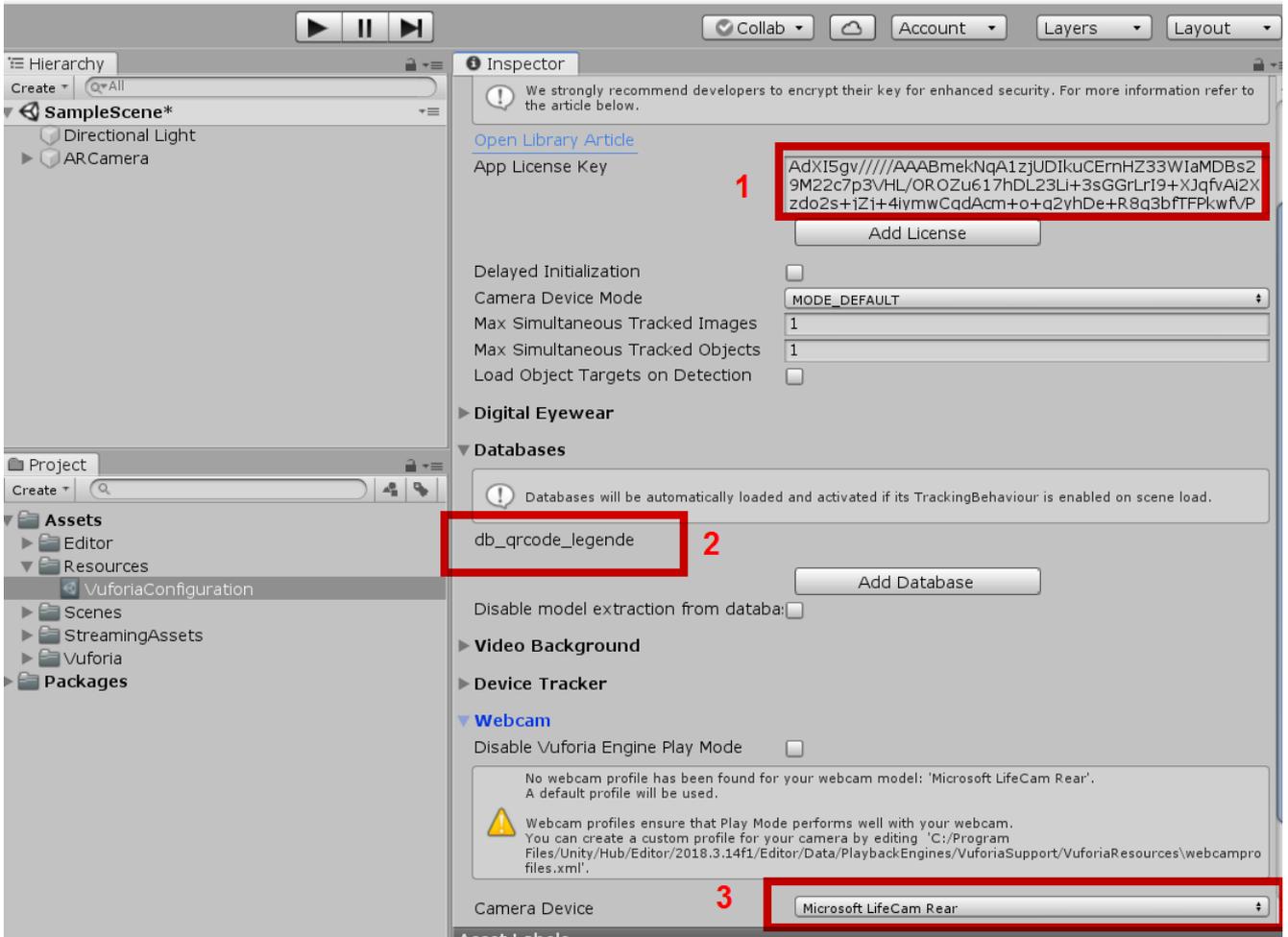
**Le Cube est sous l'ImageTarget**

C'est pour cela que le Cube apparaît si on détecte l'ImageTarget par l'ARcamera.

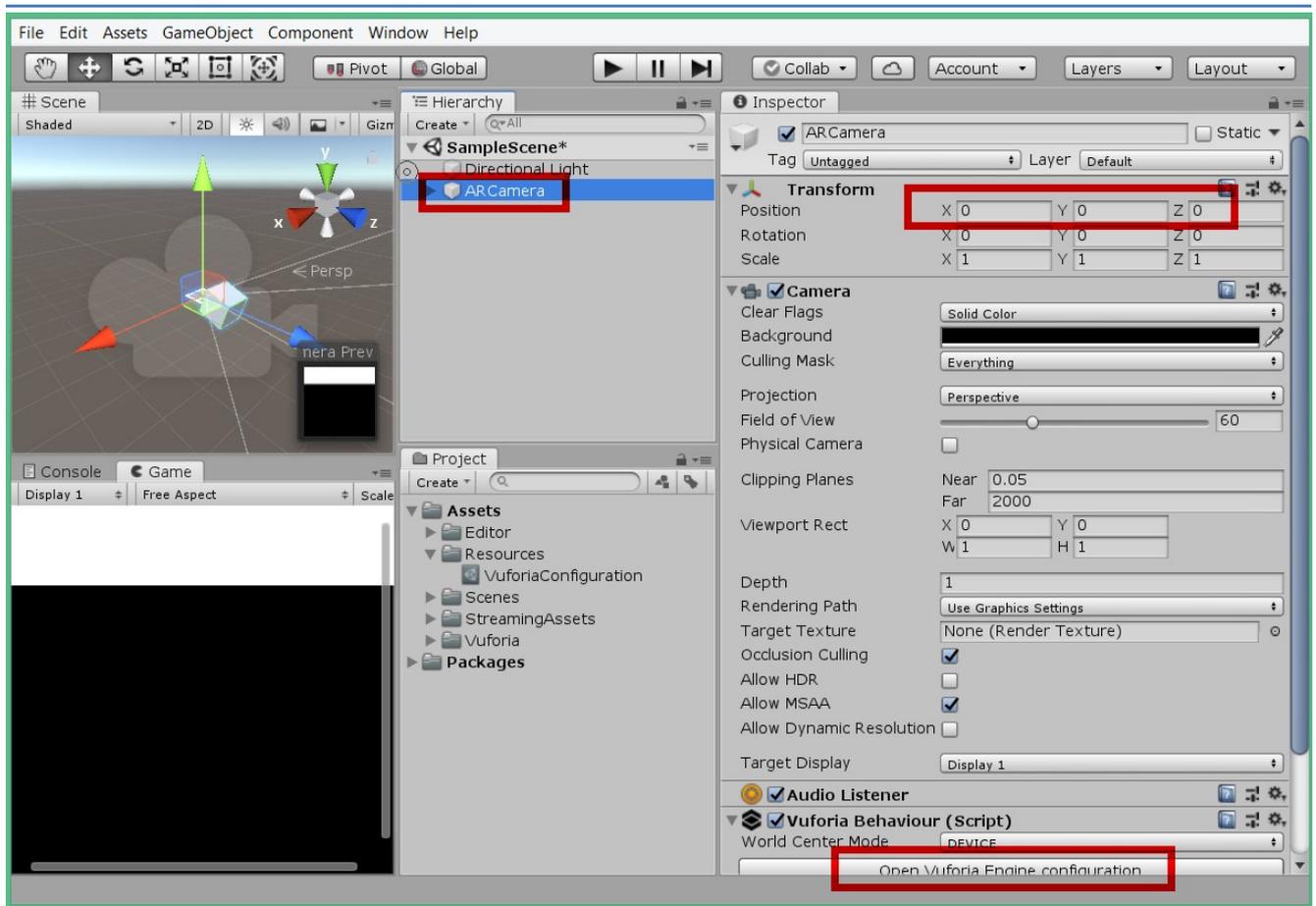
### 3.4.7 Tester

Passé en mode Run

Présenter l'imagetarget devant la caméra le cubedoit apparaître.



## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR



## 4. Création d'une scène avec unity

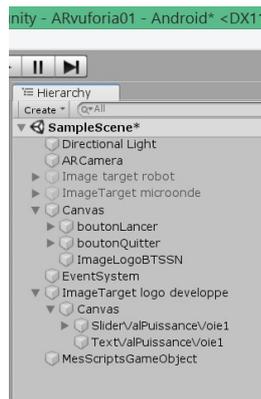
### 4.1 Première scène

A l'aide du tuto unity les bases, créer une scène avec :

- une caméra Vuforia (effacer celle installée au démarrage de unity).
- une imagetarget Vuforia
- un slider et un texte sous imagetarget.
- un bouton « quitter » directement sur la scène
- une image normale directement sur la scène

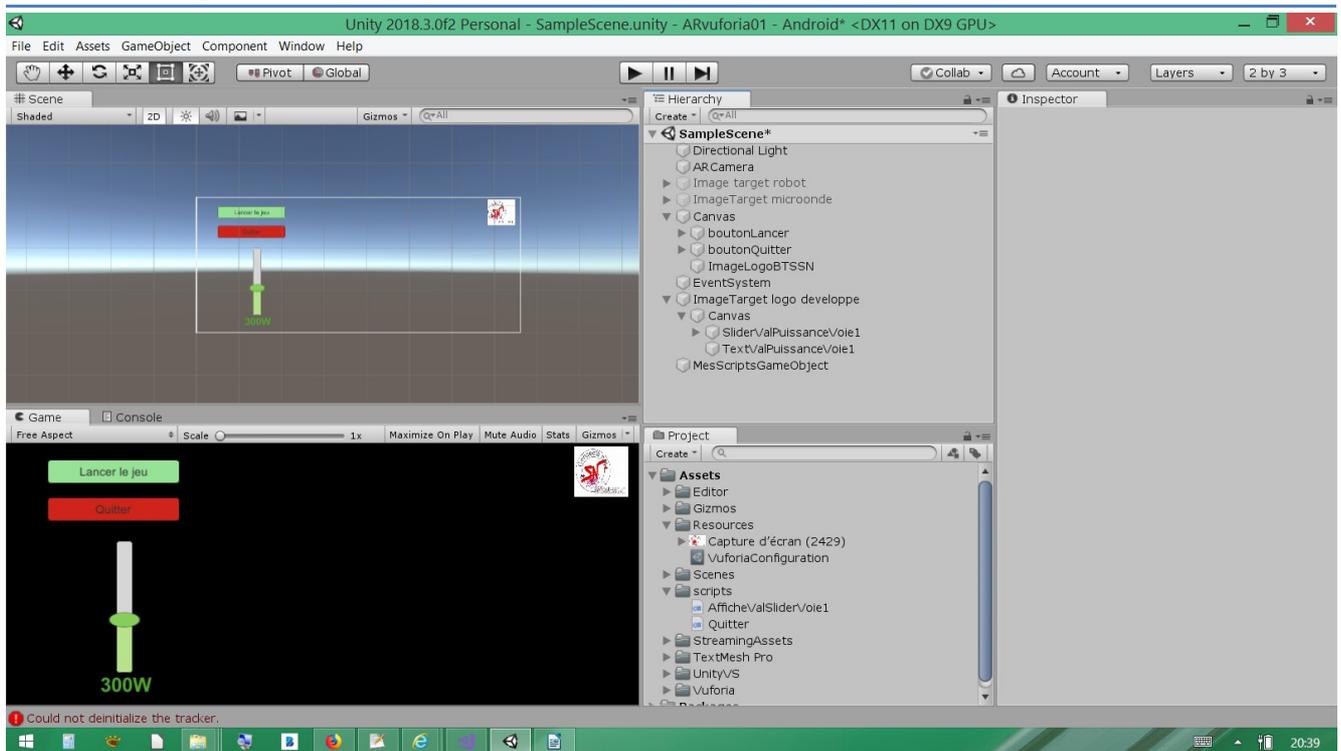
Remarque : une zone de dessin 'canvas' est créée automatiquement lors de l'ajout d'éléments UI.

Voici un exemple d'organisation :



qui provient de la scène suivante :

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR



### 4.2 Création des scripts

Dans assets, créer un dossier pour mettre les scripts : 'MesScriptsGameObject'  
Assets regroupe tous les éléments présents dans le projet.

#### 4.2.1 Script du bouton 'quitter'

Il faut :

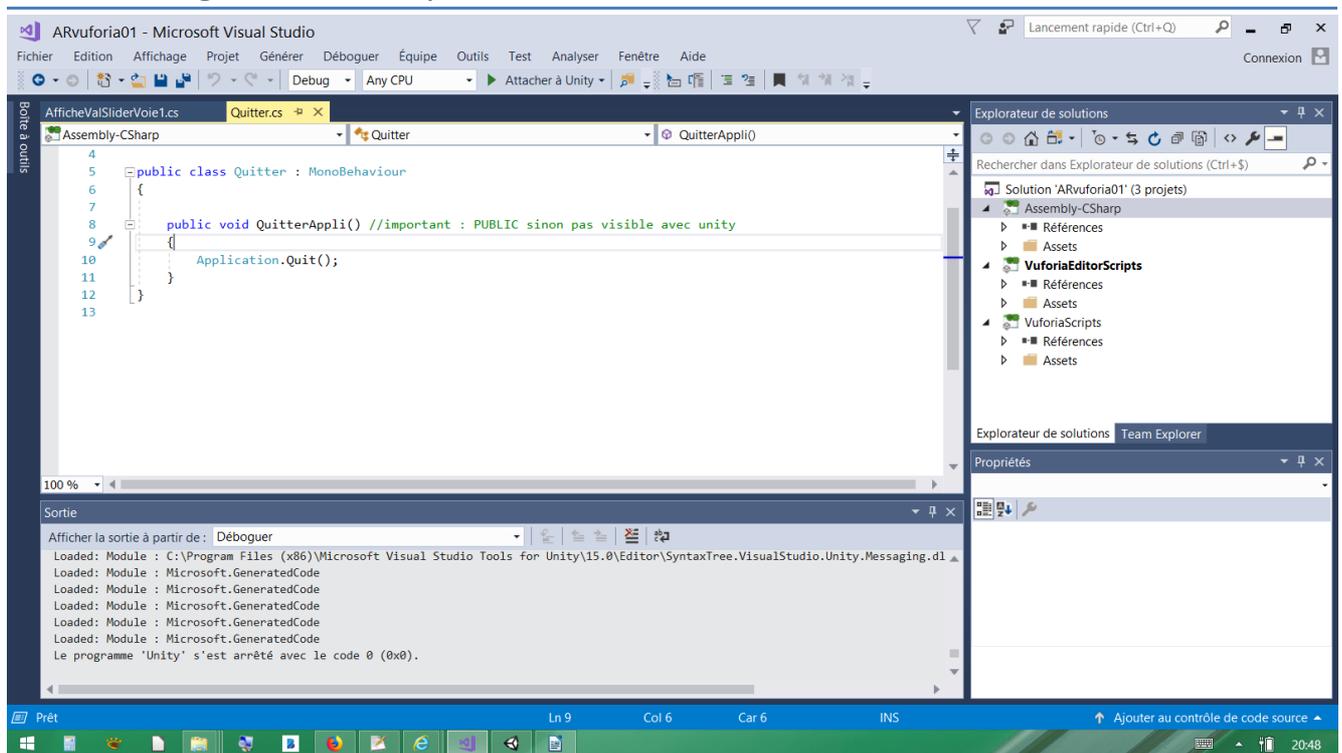
a) créer un script :

**Dans onglet 'project' cliquer gauche sur 'create' + 'c# script', le script apparaît.**

**Double cliquer sur le script : visual studio s'ouvre**

**Effacer les fonctions de base (start et update) et créer la votre :**

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR

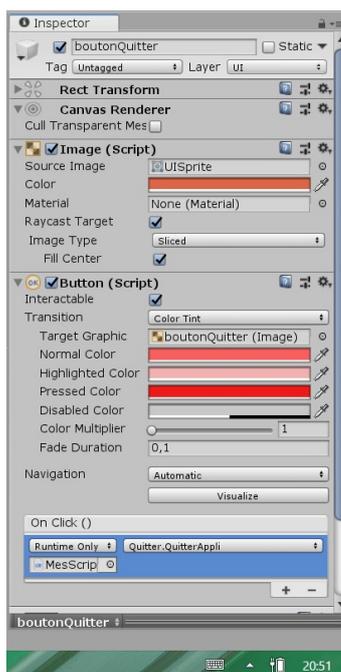


b) placer le script dans un gameobject :

**Dans unity, déplacez le dans le dossier 'MesScriptsGameObject' par un drag and drop.**

c) associer le script au bouton 'quitter' :

**dans unity, cliquer sur le bouton 'quitter', ajouter une action dans onclick(), déplacer 'MesScriptsGameObject' sous 'runtime only' et choisir la fonction QuitterAppli() dans fonction.**



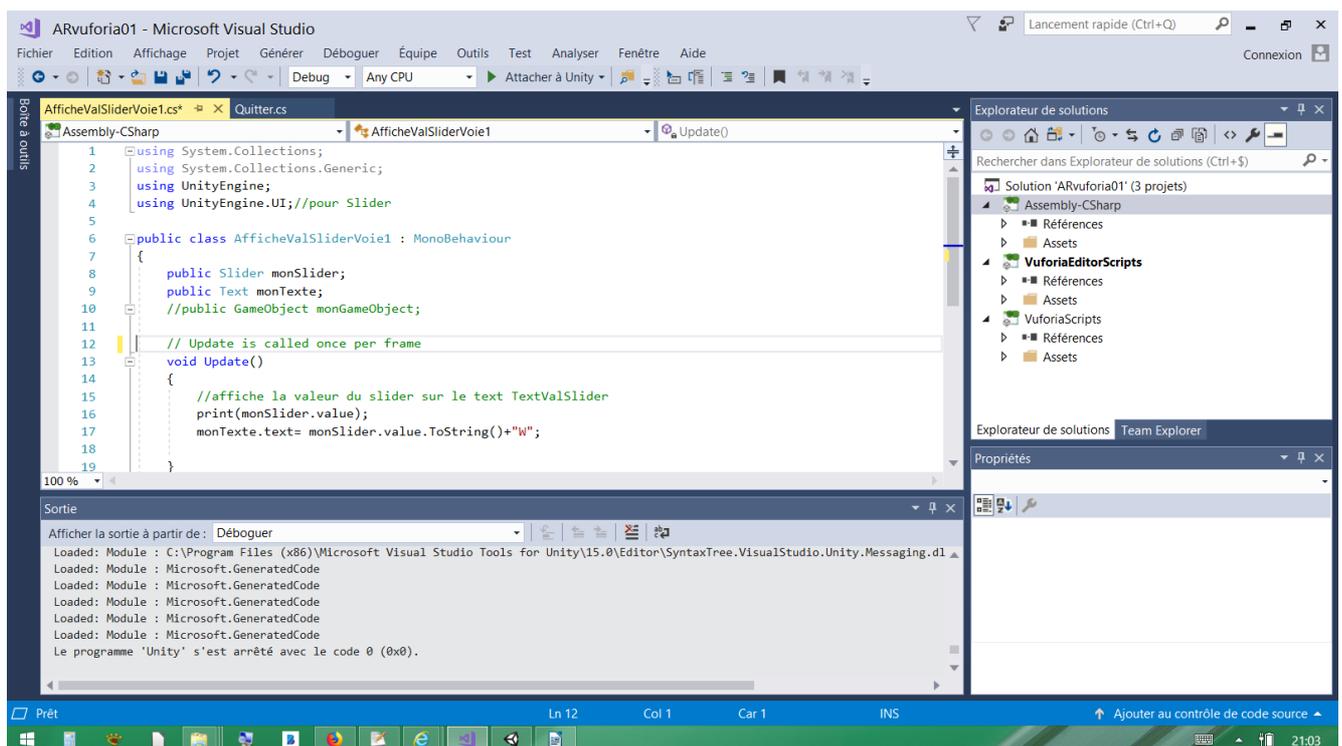
d) Tester le bon fonctionnement de la fonction.

Attention : la fonction Application.Quit() ne peut être testée qu'en exportant l'appli sur un smartphone car en mode développement elle est inefficace.

## 4.3 Scripts slider et text

On souhaite que le text sous le slider affiche la valeur réglée sur le slider.

### 4.3.1 Scripts du slider



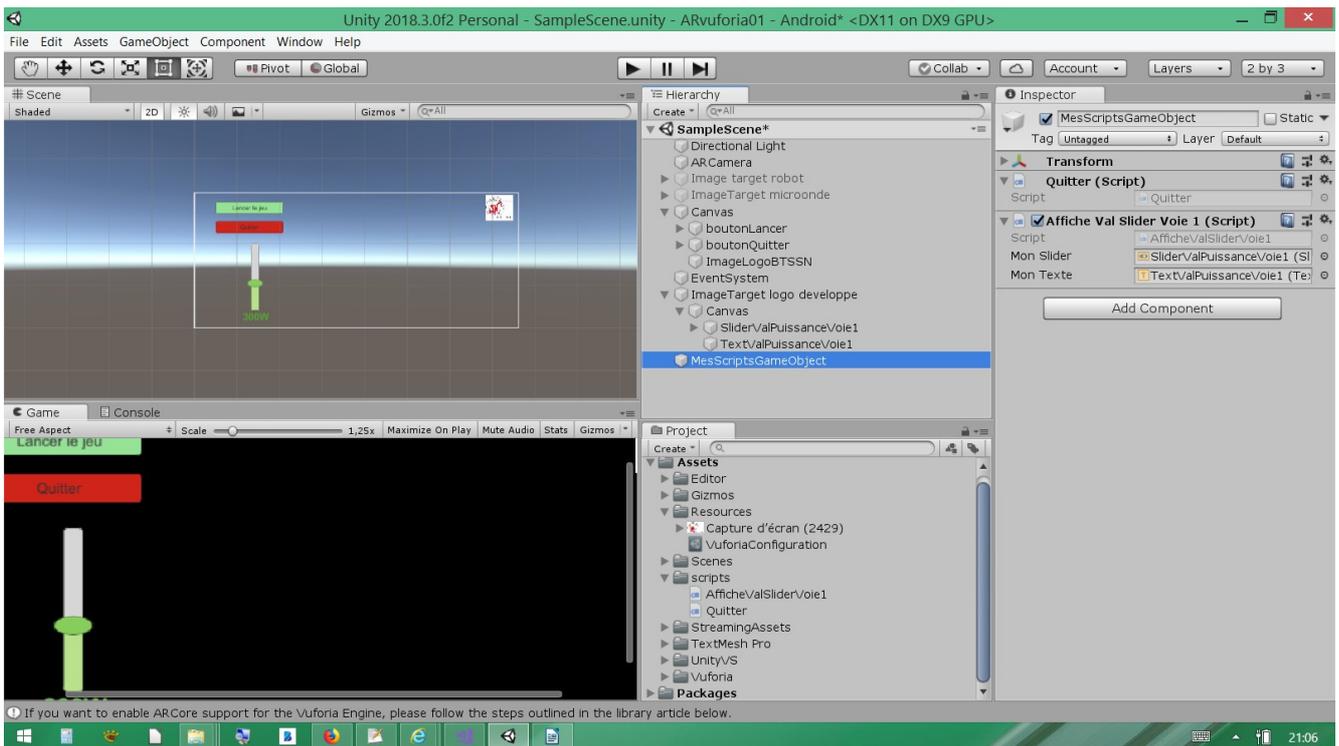
### 4.3.2 Association entre le slider et le text

Dans le script les variables 'public' sont visibles dans l'inspecteur afin d'y placer les paramètres que l'on veut.

Ici grâce au script AfficheValPuissanceVoie1() et à sa fonction update() on veut régulièrement (d'où la fonction update()) mettre à jour le text.

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR

Les paramètres à mettre dans le script sous unity sont donc : le slider et le text comme sur la vue ci dessous :



### 4.3.3 Tester le bon fonctionnement :

Pas besoin de build : un appui sur 'play' suffit. Dans la zone 'Game' on peut voir la webcam du PC apparaître ainsi que les éléments directement dans la scène.

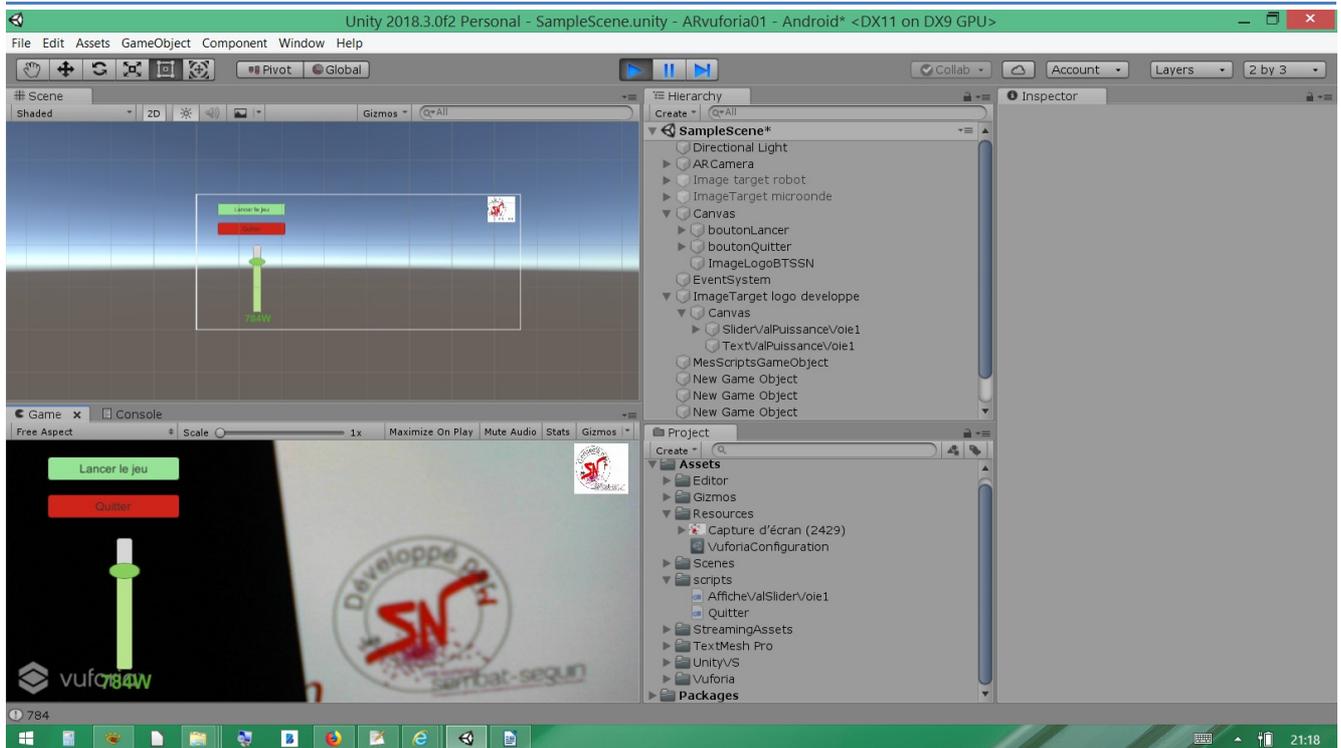
Les éléments sous 'imagetarget' nécessitent la présentation de l'image target.

Pour cela j'utilise mon téléphone avec l'image target en pleine écran :



A la reconnaissance de l'image le slider apparaît et sa variation modifie bien le texte associé.

## TP réalité augmentée : Unity + Vuforia - créer une IHM en AR



Remarque : les objets grisés sont inactifs.

## 5. Lien avec des données externes

### 5.1 Contexte du test

On veut que le slider prenne la valeur d'une donnée stockée sur une base mysql :  
Pour simuler la réponse à une requête, on crée un php qui envoie par 'echo' un nombre.  
L'envoi de la requête se fait par appui sur un bouton 'LireData' présent dans l'UI unity.  
La valeur lue configure la valeur du Slider.

### 5.2 Fichier php

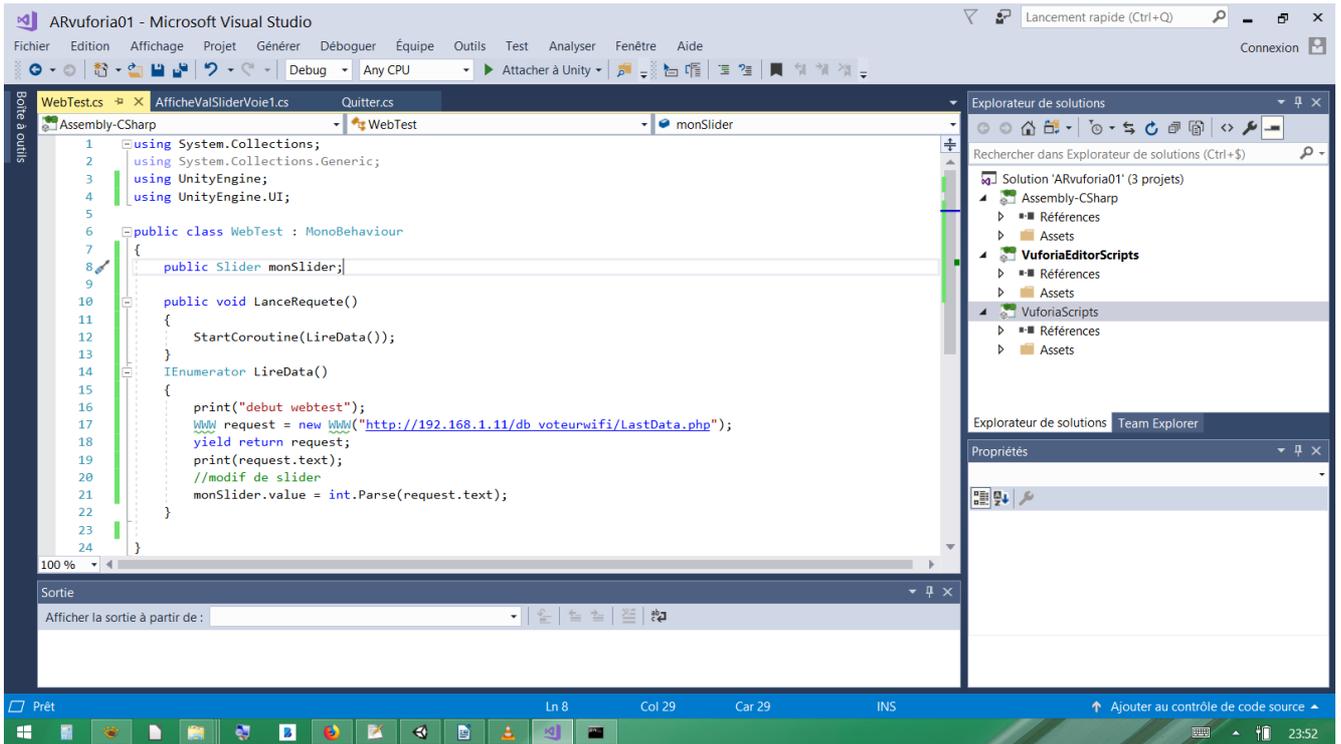
Fichier php à l'adresse 192.168.1.11 : (WAMP est en marche à cette adresse)

Un script php doit permettre de renvoyer la dernière donnée stockée : LastData.php

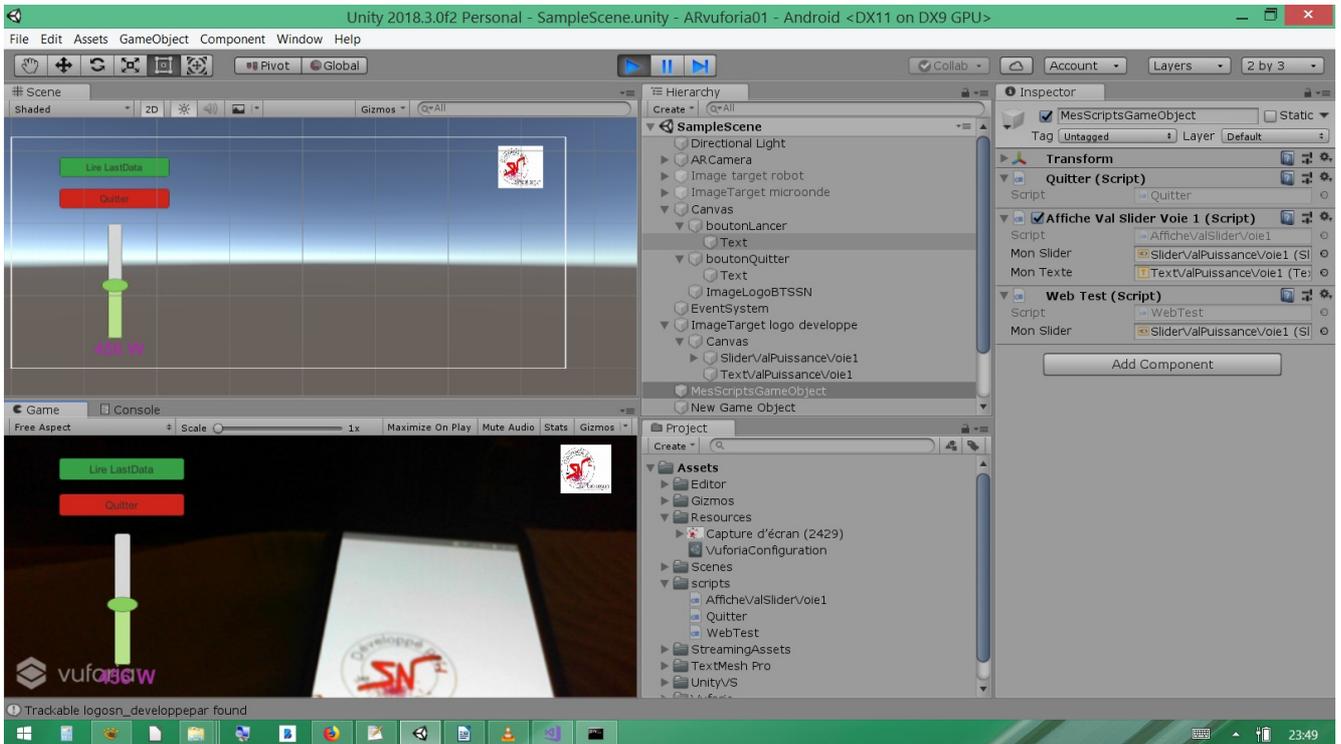
```
mysql et unity.cs | installation.bat | LastData.php
1 <?php
2 echo "456";
3 >
```

pour les essais on utilise un php simple envoyant un nombre.

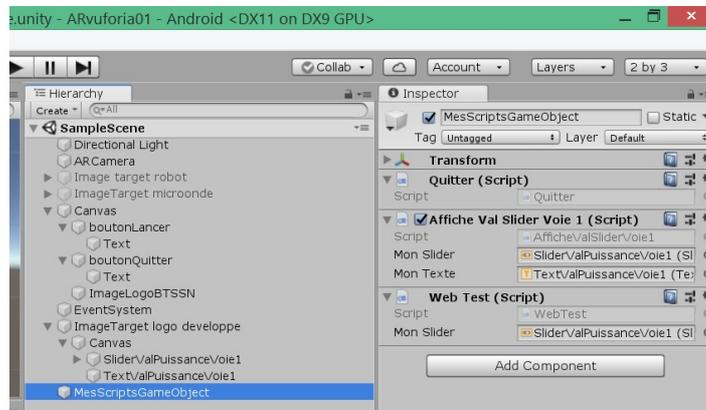
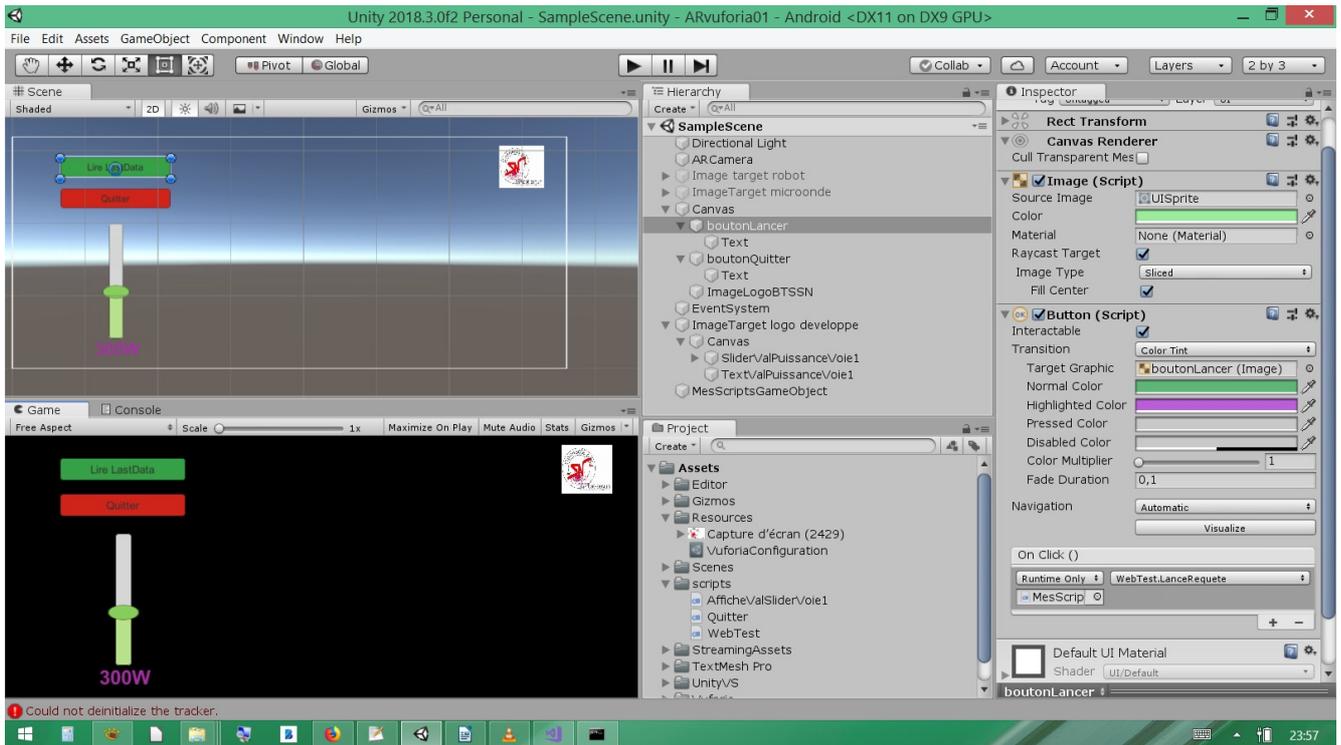
### 5.3 Script WebTest.cs dans unity



### 5.5 Résultat : ok



# TP réalité augmentée : Unity + Vuforia - créer une IHM en AR



Complément : solution avec URI modifiable :

En ajoutant un InputField on peut aller chercher le fichier LastData.php n'importe où !

## 6. Solution récente

La classe WWW de unity est dépréciée, une nouvelle classe est utilisable.

Voici le nouveau code :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Networking;
using UnityEngine.UI;

public class LectureDBupdate : MonoBehaviour
{
    public Text monText;

    // Update is called once per frame
    void Update()
    {
        print("début LanceRequete");
        StartCoroutine(GetRequest("http://192.168.0.181/voitureRC/LastData.php"));
    }

    IEnumerator GetRequest(string uri)
    {
        using (UnityWebRequest webRequest = UnityWebRequest.Get(uri))
        {
            // Request and wait for the desired page.
            yield return webRequest.SendWebRequest();

            string[] pages = uri.Split('/');
            int page = pages.Length - 1;

            if (webRequest.isNetworkError)
            {
                Debug.Log(pages[page] + ": Error: " + webRequest.error);
            }
            else
            {
                Debug.Log(pages[page] + ": \nReceived: " + webRequest.downloadHandler.text);
                monText.text = webRequest.downloadHandler.text;
            }
        }
    }
}
```

```
}  
} //GetRequest()  
}
```

En plaçant le code dans la fonction Update() cela permet une mise à jour automatique des données.

Résultat : ok testée.

## 7. Conclusion

Ceci est une première approche sur la création d'ihm en réalité augmentée.

A vous de jouer...