

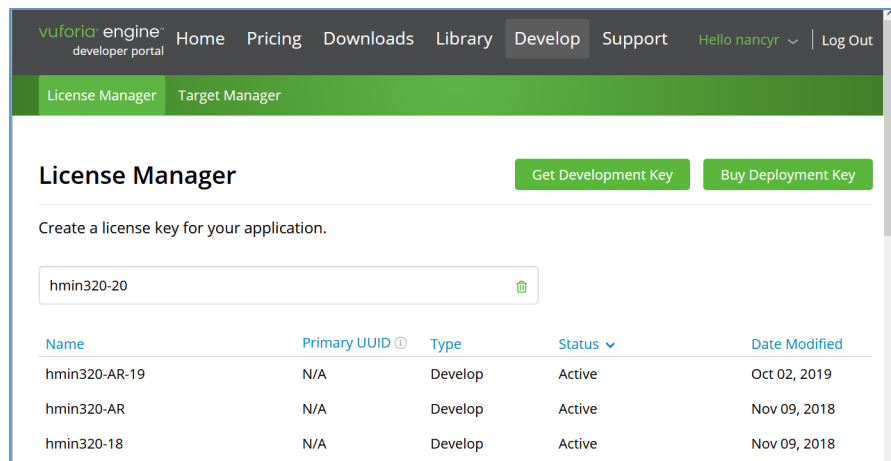
## TP4 - RA AVEC UNITY ET VUFORIA

Ce TP est le mode d'emploi pour construire des applications de réalité augmentée en utilisant Vuforia et Unity.

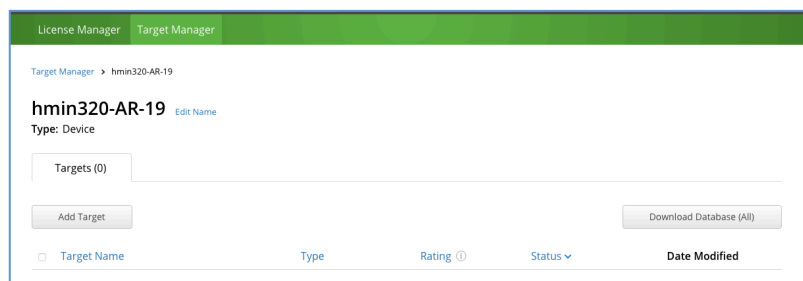
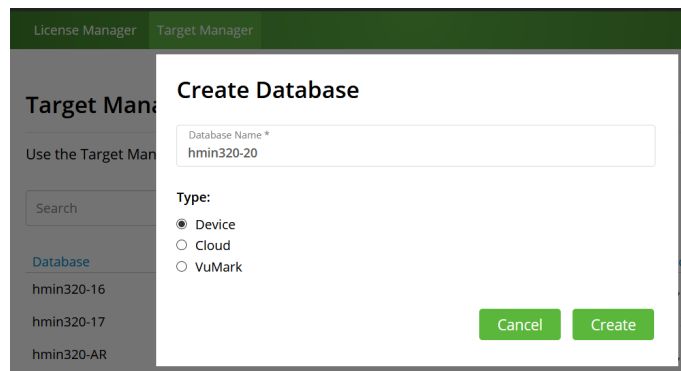
### CONFIGURATION VUFORIA

Commencez par vous inscrire sur Vuforia (<https://developer.vuforia.com/vui/auth/register>) pour obtenir une licence et définir vos marqueurs.

Une fois connecté, cliquez sur **Get Development Key** dans la rubrique License Manager. Créez une clé pour votre application.







Cliquez ensuite sur l'onglet **Target Manager**. Ajoutez une nouvelle database de type Device. Puis, cliquez sur votre base de données et ajoutez un target du type Image.



Sélectionnez *Single Image* et utilisez l'image du dossier du TP, billetBN.jpg. L'attribut Width, correspond à la taille physique du marqueur. Cette valeur peut être changée plus tard dans l'inspector d'Unity :

### Add Target

Type:

 Single Image
  Cuboid
  Cylinder
  3D Object

File:

.jpg or .png (max file 2mb)

Width:

Enter the width of your target in scene units. The size of the target should be on the same scale as your augmented virtual content. Vuforia uses meters as the default unit scale. The target's height will be calculated when you upload your image.


Name:

Name must be unique to a database. When a target is detected in your application, this will be reported in the API.

Validez et cliquez ensuite sur le nom du target pour vérifier les points d'intérêt (features) détectés dans l'image. Le score (nombre des étoiles) vous donne une indication de la robustesse de l'augmentation par rapport à votre marqueur :

### billetNB

[Edit Name](#) [Remove](#)



Type: Single Image

Status: Active

Target ID: c9514fc03bf1449e946ed3afb95c8a7d

Augmentable: ★★★★★

Added: Oct 14, 2020 13:26

Modified: Oct 14, 2020 13:26

Ajoutez un deuxième ImageTarget en utilisant l'image grayscale.jpg. Revenez à l'écran base de données et téléchargez ensuite les informations de vos marqueurs pour Unity grâce au bouton **Download Database**.

Revenez à la page License Manager, cliquez sur le nom de votre clé et copiez sa valeur dans un fichier texte.

[License Manager](#) > hmin320-20

## hmin320-20

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

License Key

Usage

Please copy the license key below into your app

```
AVXc67D/////AAABmQhlnMvpVkk1gPuanNoCVMIX9eP1N00Y5FXe5hpc3+NuSi58Ey3GU/N/s86y/VXc9UDj77yj
goLLTj9zj009cwr0UdLLQjs2ywIbSurvYeyinzniPSVurtLE+j7wVDPKRPvnjBuOMI0WXCIEI6LXoi7slimIRG4I
T0rJZeeNeg5ZY99a1nxfObRjiOqvH3CygOJoJ2f/oP6Q0aXUoxZYQ0pUmZHPos5OhdskbApJ0uDKVSNKi2kVE6tp
PnLf8KyvJEodpEjgr92R4XVv47s5V+0kzFWKpStzWV1GY+fMr8UKWg9qYjuVntiK4ydZ2u+XgZBlehtur1gxHTt6x
15UFFxDnWD85wyag8sdsHymd
```

Plan Type: Develop

Status: Active

Created: Oct 14, 2020 13:23

License UUID: 1f475649558241e689b5ef65e62c9cc4

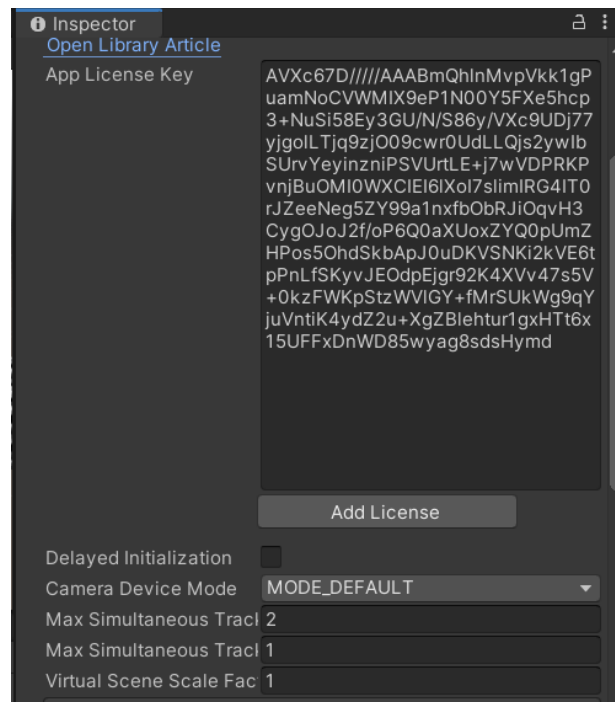
Permissions:

- Advanced Camera
- External Camera
- Model Targets

## CONFIGURATION UNITY

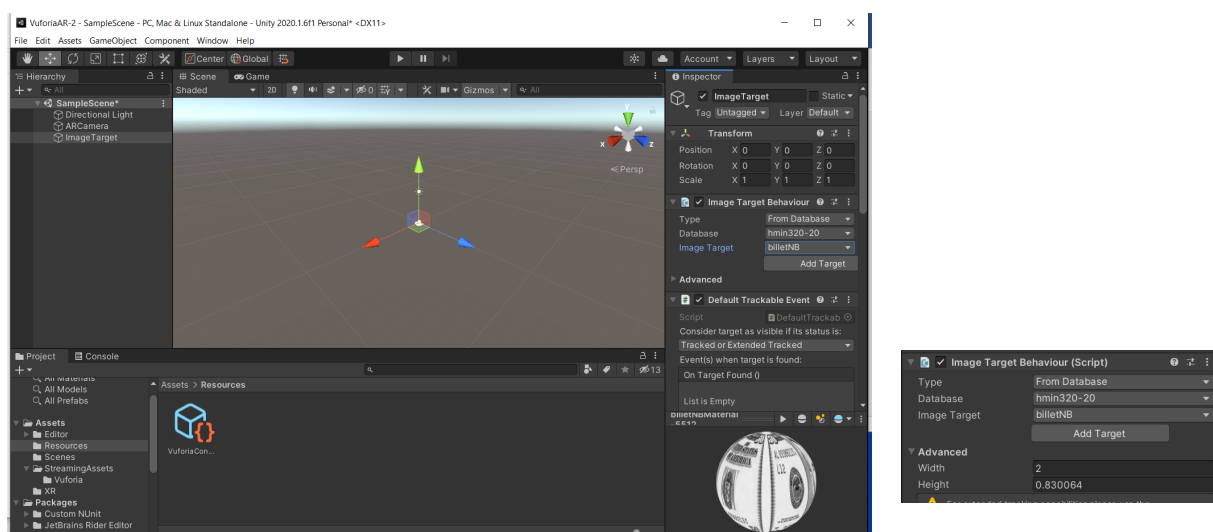
Créez un nouveau projet 3D Unity. Allez à la page <https://developer.vuforia.com/downloads/sdk>, et cliquez sur « Add Vuforia Engine to a Unity Project or upgrade to the latest version ». Cela va installer les composants nécessaires au fonctionnement de Vuforia.

Effacez la caméra par défaut, et ajoutez la caméra Vuforia : **GameObject > Vuforia Engine > ARCamera**. Sélectionnez l'objet **ARCamera** et cliquez sur le bouton **Open Vuforia Engine configuration** dans l'Inspector. Entrez votre clé de licence dans le champ App License Key. Changez Max Simultaneous Tracked Images à 2 (nos deux Image Target).

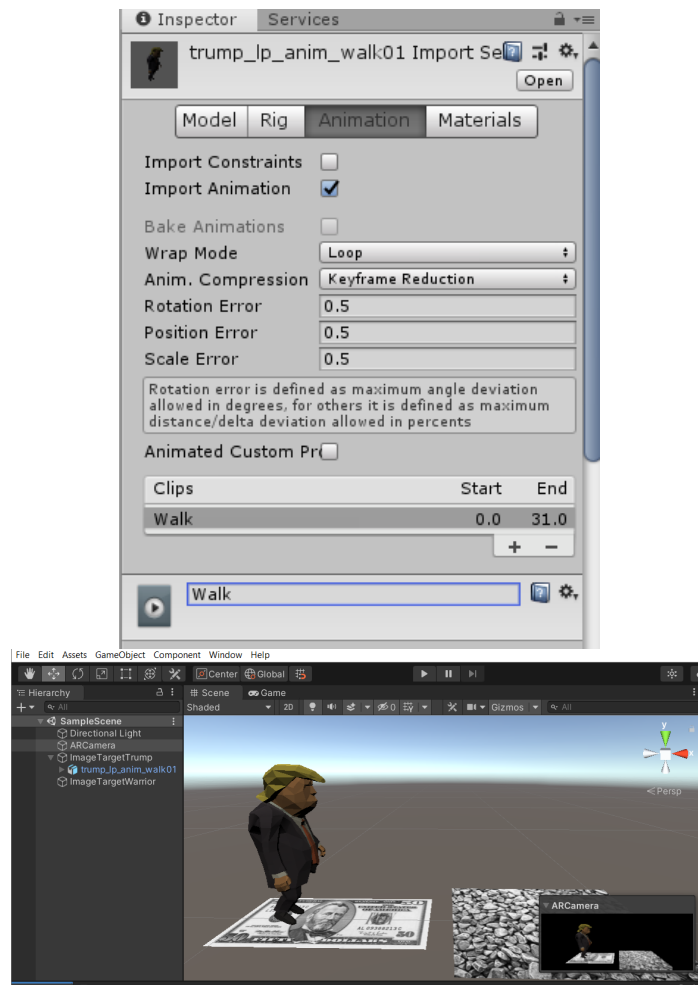


Maintenant, nous allons connecter les Image Target créés aux objet que l'on veut voir apparaître lorsque les marqueurs sont visibles par la caméra.

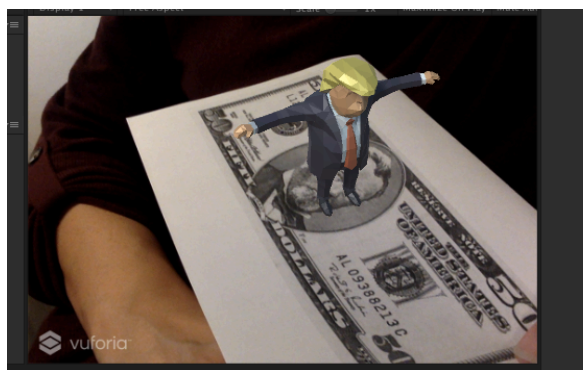
Créez l'Image Target dans votre scene grâce au menu **GameObject > Vuforia Engine > Image Target**. Dans l'Inspector sélectionnez le nom de votre database et le nom du target à utiliser. Une fois ces valeurs définies, vous verrez l'image du marqueur apparaître dans la scène. Modifiez sa taille si nécessaire dans la partie Advanced du Image Target Behaviour. Ajoutez le second ImageTarget.



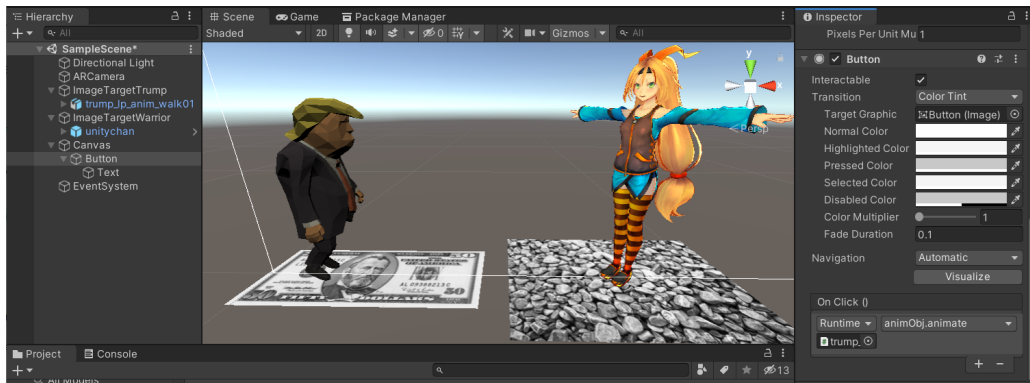
Ajoutez le modèle qui sera posé sur le marqueur en tant qu'enfant de l'Image Target. Dans le TP vous avez un dossier Trump que vous pouvez importer dans votre projet, j'ai choisi trump\_lp\_anim\_walk01 du dossier Anims en contrôlant l'importation : Dans l'onglet **Rig** de l'inspector cliquez sur **Animation Type > Legacy**. Puis, dans l'onglet **Animation**, cliquez sur **WrapMode > Loop**. Changez le nom de l'animation à « Walk ».



Testez votre scène en mettant le marqueur bien en vue de votre caméra.



Pour le second Image Target, importez un personnage de votre choix (le mien provient de Chan <https://assetstore.unity.com/packages/3d/characters/unity-chan-model-18705?assetID=71033040>).



Créez maintenant un nouveau script à associer à votre objet Trump :

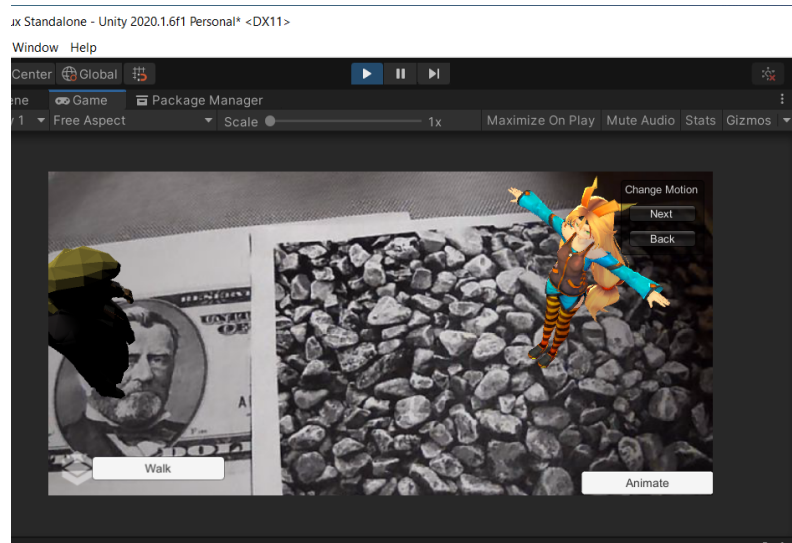
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class animObj : MonoBehaviour
{
    private Animation anim;
    public string animName;
    // Start is called before the first frame update
    void Start()
    {
        anim = GetComponent<Animation>();
    }

    // Update is called once per frame
    void Update()
    {
    }

    public void animate()
    {
        anim.Play(animName);
    }
}
```

Maintenant, on va ajouter un bouton permettant de lancer l'animation. Créez un objet Canvas, en mode *Scale with screen size* puis un objet Button. Associez le script au clic du bouton avec un événement *OnClick*.



Testez votre scène !

Créez un second bouton pour animer le deuxième personnage ou pour lancer un système de particules. Envoyez votre code et votre compte rendu à [nancy.rodriquez@lirmm.fr](mailto:nancy.rodriquez@lirmm.fr)