

UNIVERSITE LE HAVRE

FACULTE DES SCIENCES ET TECHNIQUES

RAPPORT DE PROJET – Master 2 IWOCS

SUJET

« Réalisation d'une application Android introduisant de la réalité augmentée sur des cartes de visite »

Etudiants :

- Mouhamadou Moustapha LO
- IBRAHIM ELMI Youssouf

Encadré par :

- ◆ M. Jean Grieu
- ◆ Mme Florence LECROQ

Année universitaire : 2017 – 2018

REMERCIEMENTS

D'abord nous tenons à exprimer notre gratitude envers Monsieur Jean Grieu et Madame Florence LECROQ pour avoir proposé ce projet , pour leur disponibilité tout au long de sa réalisation et pour nous avoir offert l'opportunité de découvrir le monde de la 3D et de la réalité augmentée et ainsi pouvoir acquérir de nouvelles compétences dans des domaines qui nous étaient méconnus auparavant.

Ensuite, nous remercions le responsable de notre formation M. Laurent Amanton pour nous avoir permis d'effectuer ce projet dans le cadre du module Big Project de la formation Master 2 IWOCS (Ingénierie Web, Objets Communicants et Systèmes complexes).

SOMMAIRE

INTRODUCTION.....	5
I. Analyse du Sujet.....	6
1. Objectif du projet.....	6
2. Contexte.....	6
3. Enjeu.....	6
II. Outils Technologiques.....	7
1. Blender.....	7
a. L'univers de la 3D.....	7
b. Aperçu du logiciel.....	8
c. Un petit peu de vocabulaire.....	9
2. Unity.....	10
a. Installation et aperçu.....	10
b. Asset Store.....	12
c. moteur de jeu Unity.....	13
III. Réalisation.....	14
1. Modélisation du personnage sous Blender.....	14
a. L'add-on ManuelBastioni.....	14
b. Ajout des clés de référence (keys).....	16
c. Synchronisation audio et keys.....	18
2. Compilation sous Unity.....	19
a. le kit Vuforia.....	19
b. notion d'Image Target dans Vuforia.....	21
c. intégrer les fichiers de Blender.....	22
d. compiler pour Android.....	23
3. Difficultés rencontrées.....	25
IV. Résultats et Tests.....	26
CONCLUSION.....	27
Bibliographie et Webographie.....	28

INTRODUCTION

Dans le cadre du module Big Project du Master 2 IWOCs, nous avons choisi de travailler sur le projet proposé par M. Jean Grieu de l'ISEL et qui consiste en la réalisation d'une application Android introduisant le concept de réalité augmentée sur les cartes de visite.

Oui mais c'est quoi la réalité augmentée ?

La réalité augmentée est la superposition de la réalité et d'éléments (sons, images, 2D, 3D, etc.) calculés par un système informatique en temps réel. Souvent, elle désigne les différentes méthodes qui permettent d'incruster de façon réaliste des objets virtuels dans une séquence d'images.



L'exemple illustré ci-dessus du célèbre jeu Pokemon Go (100 millions de téléchargement rien que sur le play store de Google) dont le but est d'attraper des Pokémons montre à quel point la réalité augmentée devient de plus en plus émergent.

Cette émergence est permise par le développement et la démocratisation des technologies de l'information et de la communication avec des applications dans plusieurs domaines (éducation, santé, jeux vidéo, cinéma, l'industrie, etc.).

C'est donc ce principe que l'on veut intégrer dans la visualisation de cartes de visite en créant une application Android qui lance une animation 3D par dessus la carte.



Pour mener à bien ce projet, on a commencé d'abord par une phase de recherche et de documentation sur ce que c'est la réalité augmentée ainsi que la conception de modèles et d'animations 3D. Ensuite on a commencé à travailler et à prendre en main les outils avec lesquels il nous a été demandé de travailler à savoir **Blender** pour la conception de l'animation 3D et **Unity** pour l'intégration et la compilation pour mobiles Android.

I. Analyse du Sujet

1. Objectif du projet

L'objectif de ce projet est donc de pouvoir réaliser une application Android qui introduit le concept de la réalité augmentée sur la visualisation de cartes de visite en y ajoutant une animation 3D d'un personnage qui se présente.

L'application doit donc être capable de reconnaître la carte ou des informations présentes dessus (le nom, la photo, le mail) et en fonction de cela lancer l'animation 3D de la présentation de la personne concernée.

Ainsi la carte de visite devient un objet vivant tout en remplissant sa fonction de présentation et de contact.

2. Contexte

La carte de visite est un outil de communication efficace lorsqu'il s'agit d'échanger des coordonnées ou de diffuser l'image de son entreprise auprès de son réseau professionnel.

En effet il est d'usage de remettre des cartes de visite à plusieurs occasions : dans des salons professionnels, à l'issue d'un rendez-vous, etc.

Les informations présentes sur la carte vont permettre à une personne de reprendre facilement contact avec vous ou avec l'entreprise.

Non seulement la carte de visite doit indiquer votre identité, le poste que vous occupez dans une entreprise, votre secteur d'activité, ou un tas d'autres informations qui permettent de mieux cerner votre personnalité.

« *Le problème c'est qu'il n'y a pas suffisamment de place sur la carte pour tout y mettre.* »

Faudrait donc penser à comment améliorer cet outil de communication bien que efficace.

3. Enjeu

La plupart des cartes de visite sont similaires sur le fond et diffère sur le design ou la forme. En effet on trouve souvent sur les cartes : le prénom, le nom, le téléphone, l'e-mail.

L'efficacité de cet outil peut être améliorée en se démarquant des autres.

C'est là qu'intervient le principe de la réalité augmentée. Avec ce phénomène, non seulement on a les informations de contact de la personne sous les yeux mais aussi avec l'application Android adéquat on peut visualiser en scannant la carte, une animation 3D de la personne qui se présente de manière plus explicite et plus détaillée.

Cet aspect ludique et magique qu'offre la réalité augmentée va nous permettre de communiquer et d'échanger de plus en plus efficace.

Le marché de la réalité augmentée pourrait atteindre 120 milliards de dollars de revenus en 2020, selon les prévisions de [Digi-Capital](#). On pourrait largement dépasser ce chiffre quand on sait que la réalité augmentée s'invite dans presque tous les domaines de notre société (santé, éducation, industrie, entreprise, tourisme, etc.).

Et comme dans toute révolution technologique, de nombreux acteurs à commencer par les plus grosses entreprises (Google, Amazon, Apple, Microsoft) veulent tirer leur épingle du jeu en essayant de trouver la solution qui fera basculer l'immersif dans l'usage grand public.

II. Outils Technologiques

Dans cette partie, nous allons présenter les différents outils utilisés pour mener à bien ce projet.

Les outils utilisés dans le cadre de ce projet sont **Blender** et **Unity** avec pour chacun une utilisation particulière. Nous allons donc présenter ces deux outils afin de justifier leur utilisation dans la résolution du problème posé.

1. Blender

Blender est un logiciel libre et gratuit de modélisation, d'animation et de rendu en 3D. Il est disponible pour Windows, Mac et Linux, etc. Ce logiciel a su s'imposer sur le marché ces dernières années, supporté par une communauté particulièrement active.

Le logiciel Blender permet entre autres de :

- ✓ créer de manière réaliste des objets en 3D grâce à un moteur 3D
- ✓ concevoir des objets en 3D rapidement et facilement
- ✓ personnaliser vos objets en 3D
- ✓ éditer des animations pour vos créations
- ✓ sculpter vos modèles
- ✓ simuler du liquide, des cheveux, de la fumée, des vêtements, des particules, etc.
- ✓ créer vos propres jeux vidéo grâce à Blender Game directement dans le logiciel
- ✓ retoucher vos images
- ✓ etc.

Voici quelques réalisations faites avec Blender :



On voit tout de suite que les possibilités sont nombreuses avec ce puissant logiciel qu'est Blender.

a. L'univers de la 3D

3D est l'abréviation de trois dimensions : la longueur, la largeur et la profondeur. Il fait allusion à une façon de produire des graphismes en utilisant les 3 dimensions.

Ainsi la 3D permet à son utilisateur de créer diverses formes dans l'espace, mais également de leur donner une couleur, des propriétés, etc.

La 3D fonctionne sur un principe simple : il existe des points dans l'espace, dont certains sont reliés par des arêtes, et le tout forme parfois des faces, lorsque 3 arêtes ou plus sont reliés. Par convention, mais également pour des raisons de propreté au niveau du travail, une face ne doit pas avoir plus de 4 côtés. De plus, il faut savoir que, suivant les logiciels utilisés, on utilise soit des faces à 3 côtés, soit des faces à 4 côtés. Blender fait partie des nombreux logiciels qui favorisent les faces à 4 côtés. Il faudra donc s'y atteler dans la mesure du possible.

Théoriquement, il est possible de tout faire.

En vérité, la 3D et Blender ne sont que des outils, et bien souvent, le facteur limitant réside en les compétences de l'utilisateur. Ainsi, il y a très peu de choses qu'il est impossible de faire en 3D, il suffit juste souvent de beaucoup de patience et de savoir-faire.

b. Aperçu du logiciel

Pour utiliser Blender, il faut le télécharger. Et pour cela rien de plus simple, il faut se rendre sur le site officiel de Blender et cliquer sur l'onglet **Download** :

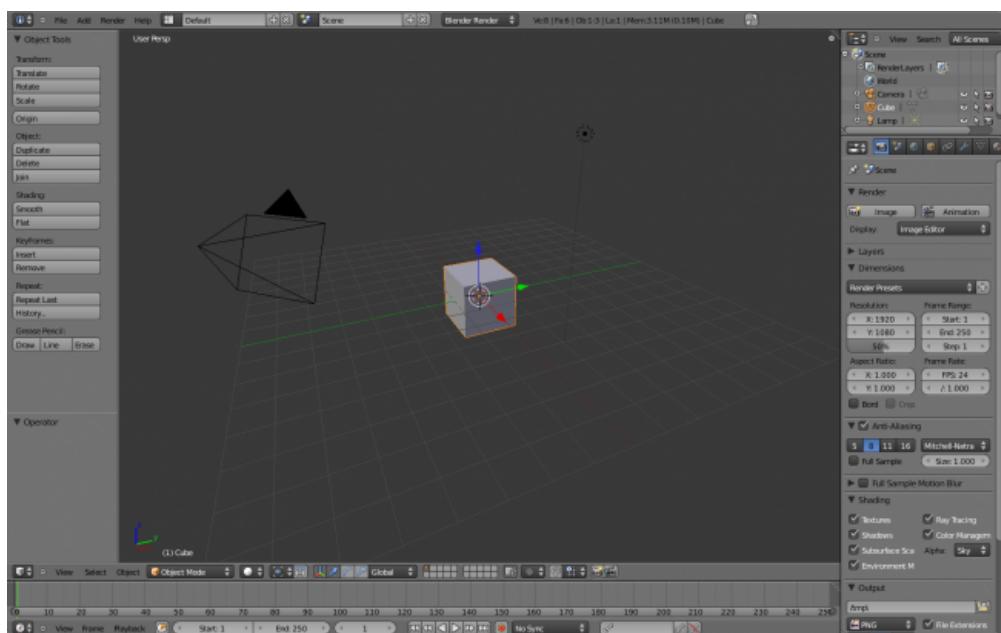
<https://www.blender.org/>

Choisissez la version qui correspond à votre système d'exploitation et cliquez sur le lien.

Pour l'installer, ne vous prenez pas la tête, faites toujours Suivant.

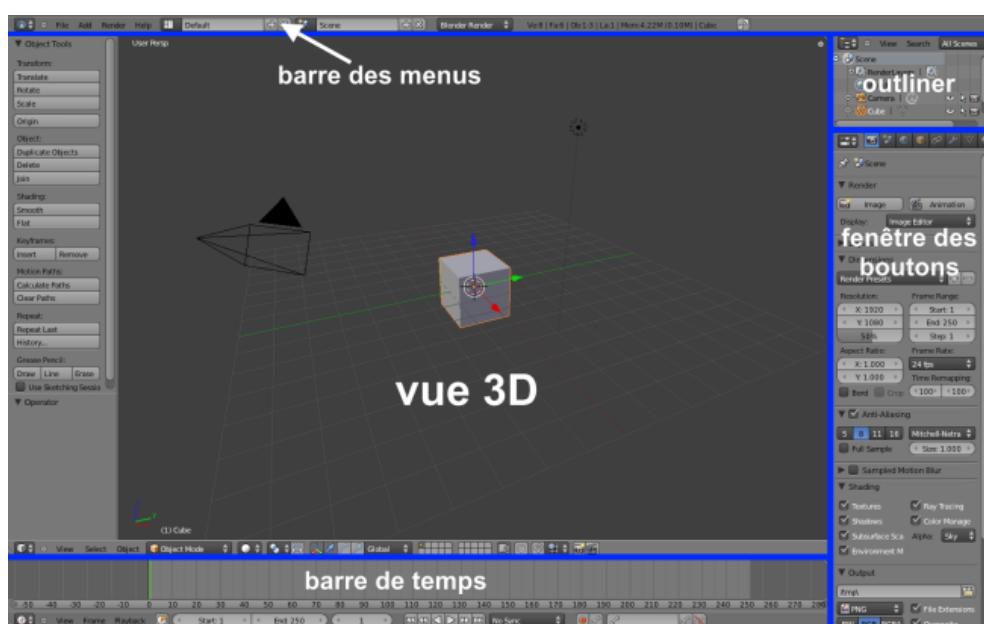
Une fois installé, vous devriez voir l'icône Blender dans vos menus.

Lors du démarrage de Blender on tombe sur ce qui suit :



Il est essentiel de savoir se repérer dans l'interface de Blender.

Le schéma ci-dessous montre les différentes parties du logiciel avec chacune une fonctionnalité particulière.



La vue 3D : c'est notre zone de travail, c'est ici que nous allons créer les figures 3D

La barre des menus : Menu classique que l'on trouve sur presque tous les logiciels. Elle nous permet entre autres d'enregistrer notre travail, de faire des export, de changer de scène, d'ajouter des scènes, de faire des rendus de notre travail, etc.

L'outliner : C'est là où verra tous les objets présents dans la scène classés par hiérarchie

La fenêtre des boutons : C'est là où l'on pourra changer les propriétés de nos objets avec différents paramètres.

La barre des temps : Cette partie regroupe les outils d'animation les plus courants.

On va passer le plus de notre temps dans la vue 3D car c'est l'espace où nous pourrons visualiser, déplacer, redimensionner et tourner nos formes primaires comme le cube, la sphère, le plan, le cône ou le cylindre.

En dessous de la vue 3D, il y a une barre pleine de boutons :



- **en Rouge** : Type de fenêtre. Ici dans la capture le type est Vue 3D (3D View)
- **en Jaune foncé** : permet de cacher ce qui est encadré en vert (View Select Object)
- **en Vert** : Ce sont les menus qui proposent les options pour la vue 3D(View), pour la sélection(Select) et pour les objets sélectionnés(Object)
- **en Bleu** : C'est le bandeau déroulant qui permet d'avoir accès à plusieurs modes d'affichage de la scène. (Object Mode, Edit Mode, Sculpt Mode, etc.)
- **en Rose** : permet de changer le style de l'affichage (Solid, Wireframe, Textured, etc.)
- **en Violet** : permet de changer le centre de rotation
- **en Orange** : permet de choisir le mode de sélection (par points, par face, etc.)
- **en Bleu foncé** : permet de trier ou de déplacer les objets dans différents calques (20 calques en tout).
- **en Vert foncé** : permet d'activer le mode d'édition proportionnel.
- **en Jaune** : permet de faire un rendu du travail (type photo ou type animation).

c. Un petit peu de vocabulaire

En 3D, on note un certain nombre de vocabulaire, utilisé très couramment. Voici les principaux :

- ✗ **Maillage** : ensemble de points, d'arêtes et de faces
- ✗ **Mesh** : maillage formant un ensemble cohérent
- ✗ **Texture** : image apparaissant sur un objet
- ✗ **Matériel** : couleurs, propriétés liées à l'aspect final (effet de transparence, de miroir), et textures d'un ou de plusieurs objets.
- ✗ **Modificateur** : outil permettant une modification semi-automatique ou automatique d'un maillage afin d'effectuer une action longue et/ou répétitive.
- ✗ **Contrainte** : outil permettant de limiter les possibilités de modification d'un objet afin de les bloquer et faciliter l'édition.
- ✗ **Rendu** : calcul qui permettra de créer une image (ou animation) finalisée du projet.
- ✗ **Objet** : mesh ou entité(lumière, caméra, etc.) comportant des propriétés diverses.
- ✗ **Scène** : ensemble d'objets.

2. Unity

Unity est un moteur de jeu multi-plateforme (smartphone, Mac, PC, consoles de jeux vidéo) développé par Unity Technologies. Il est l'un des moteurs les plus répandus dans l'industrie du jeu vidéo, du fait de sa rapidité aux prototypages.

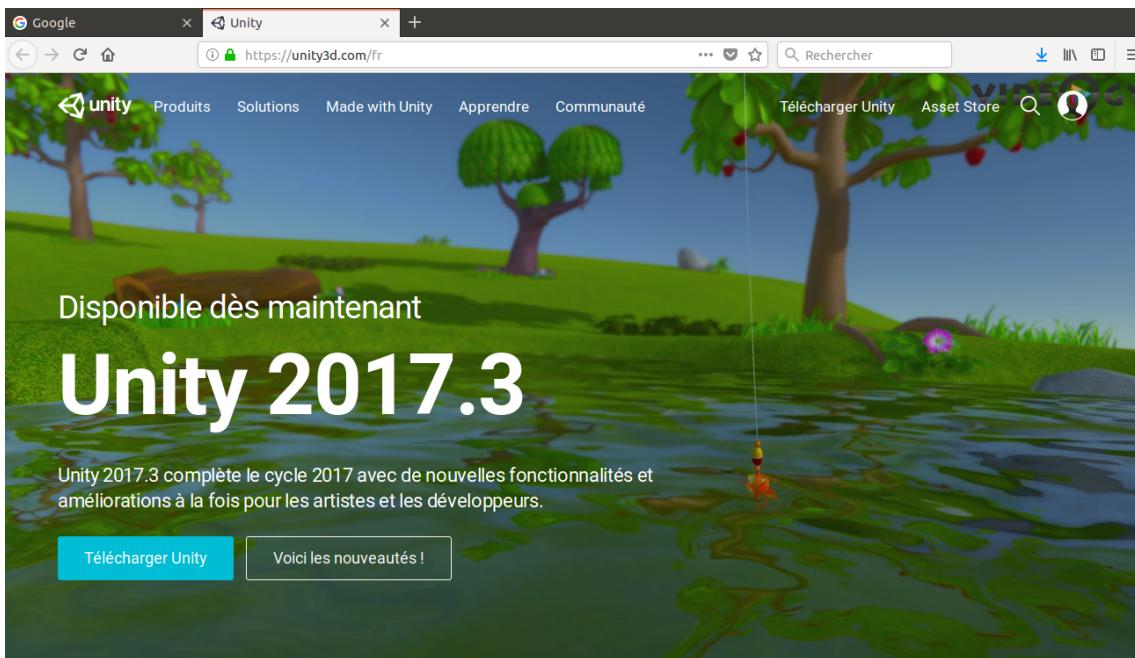


Le logiciel Unity 3D met aujourd'hui la création de contenus interactifs à la portée d'un grand nombre de métiers liés à la création. Il gère les objets animés 2D, 3D, le son, l'image, la vidéo et permet d'exporter son travail sous de nombreux plateformes.

Le logiciel est accessible en licence libre de droits pour une grande partie de ses fonctionnalités et bénéficie du support d'une importante communauté d'utilisateurs et de développeurs.

a. Installation et aperçu

Pour pouvoir utiliser unity, il faudra d'abord l'installer. Le package d'installation est disponible sur le site officiel de unity : <https://unity3d.com/fr> dans l'onglet **Télécharger unity**.

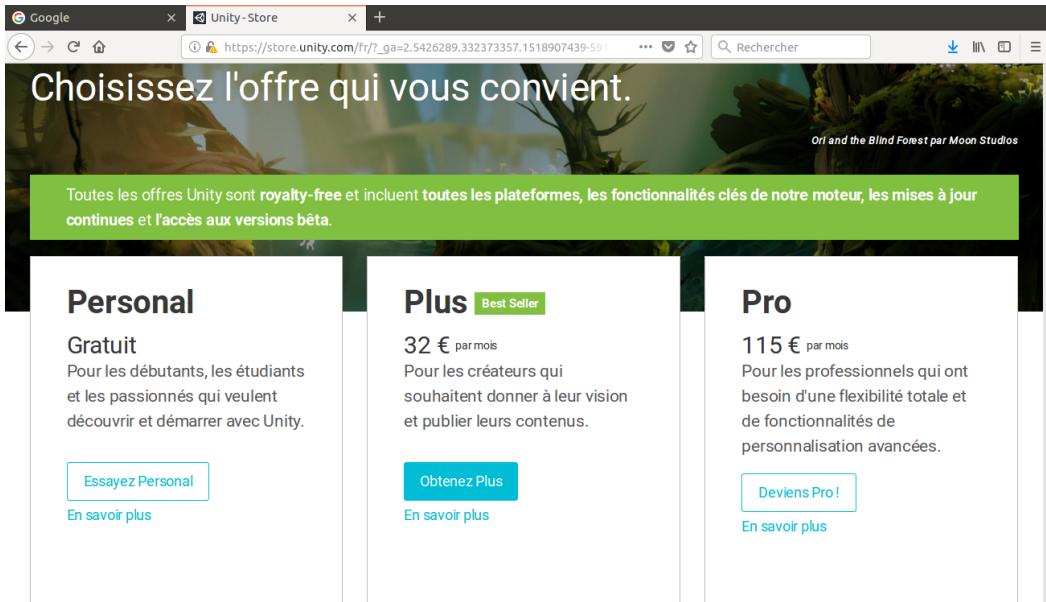


Avant de télécharger le pack correspondant à votre système, il faut d'abord créer un ID unity (un compte) sur ce site. Cet identifiant va vous servir plus tard pour vos projets.

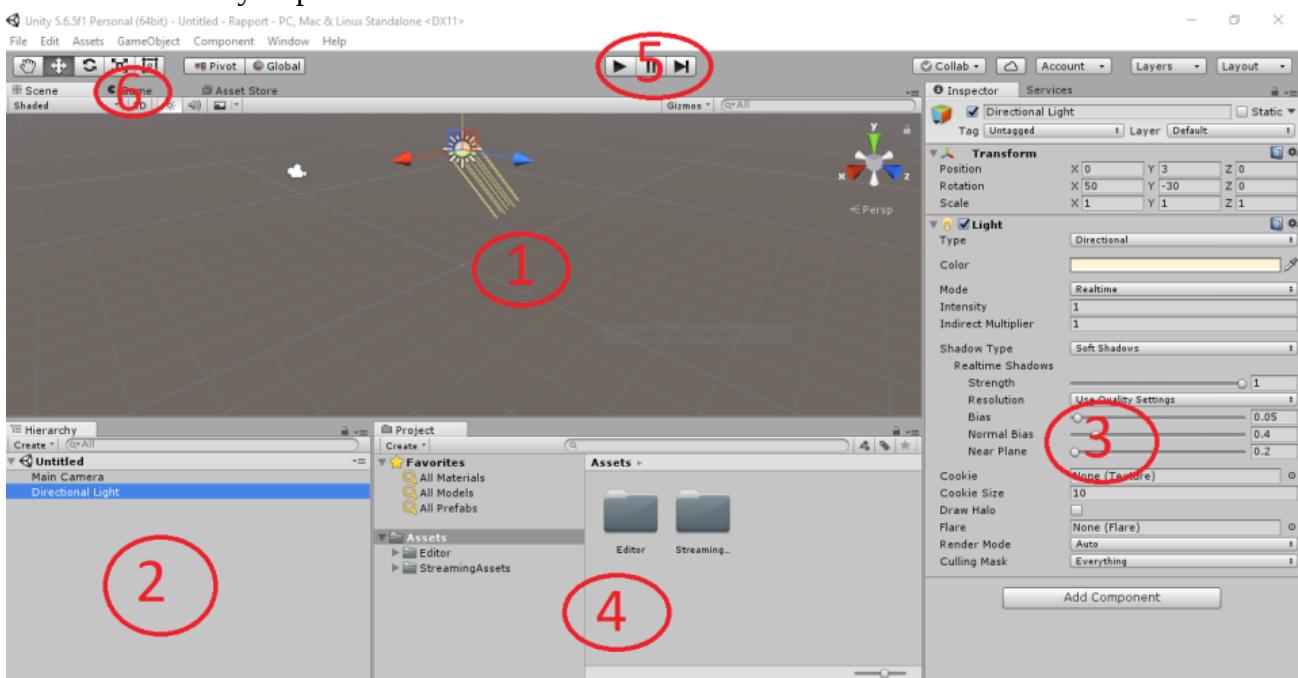
Il existe 3 versions possibles de unity :

- ✓ **Personal** : la version gratuite de unity pour les débutants, les étudiants et les passionnés qui veulent découvrir et démarrer avec unity.
- ✓ **Plus** : Version payante (32 euros/mois) pour les créateurs qui souhaitent publier leurs contenus.
- ✓ **Pro** : Version payante (115 euros par mois) pour les professionnels qui ont besoin d'une flexibilité totale et de fonctionnalités avancées.

Si vous êtes un grand débutant sur unity, il faudrait donc cliquer sur Télécharger unity et choisir la version **Personal** qui est très bien fourni en termes de fonctionnalités.



Une fois le logiciel installé, vous pouvez le lancer et une fenêtre de dialogue va apparaître vous demandant de vous connecter avec votre compte utilisateur créé au préalable sur le site de unity. L'interface de unity se présente comme suit :



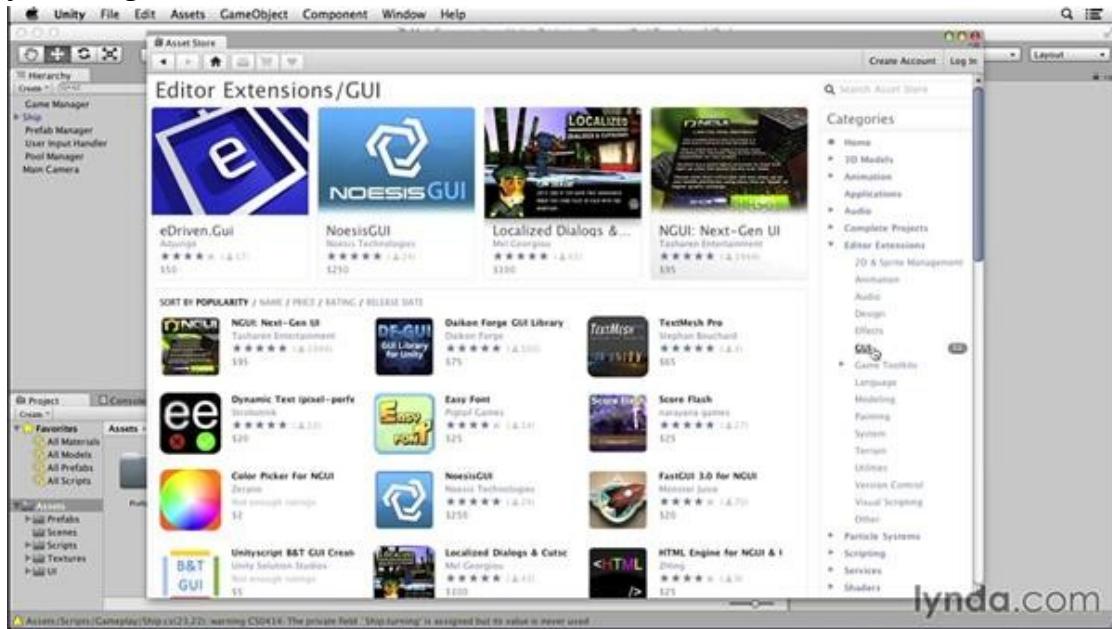
On peut voir sur l'illustration ci-dessus les différentes sections du logiciel :

1. Visualisation et manipulation des objets de la scène
2. Arborescence de la scène
3. L'inspecteur contenant les différents scripts liés aux objets ainsi que les propriétés modifiables de ces derniers.
4. L'explorateur du projet où l'on trouve les fichiers et module du projet.
5. Les boutons permettant de lancer l'exécution de la scène afin de voir le résultat.

6. Visualisation de la scène 3D en condition de Jeu (rendu final à la compilation).

b. Asset Store

L'un des plus grands intérêts d'Unity 3D réside dans ce qu'on appelle l'**Asset Store** (à l'image d'un Google Play store). Ce dernier est une sorte de boutique permettant d'accéder et de télécharger des jeux 3D ainsi que des modèles 3D créés par les différents utilisateurs. Ces jeux ou modèles peuvent être payants ou gratuits.



Avec le riche panel de produits qu'offre l'**Asset Store**, on pourra facilement y trouver des modèles d'objets déjà créés sur lesquels on pourrait se baser pour construire petit à petit notre scène.

Une fois que vous téléchargez un produit, le package apparaîtra dans votre dossier **Assets** et vous pourrez commencer à vous amuser.

Il est également possible que vous soyez un créateur, et que vous souhaitez rendre vos œuvres disponibles (gratuits ou payants), qu'il s'agisse de personnages, de musique, ou même de code.

L'**Asset Store** reste cadre : le contenu qui s'y trouve a été validé au préalable, et le risque d'escroquerie est donc faible. Bien sûr, si vous envisagez d'investir une somme non négligeable sur un package, mieux vaut se renseigner avant, voire télécharger les versions d'essai (Sample) parfois disponibles.

En cas de déception ou d'erreur lors de votre achat, vous pouvez contacter le support Unity, ou directement le créateur du produit.

Les possibilités avec Unity et son Asset Store semblent donc infinies et la seule limite sera votre imagination.



c. moteur de jeu Unity

Le principe fondateur d'Unity est que l'on développe son jeu une seule fois et qu'on l'exporte ensuite vers toutes les plateformes du marché : « Author once, deploy anywhere ».

Ainsi à l'heure actuelle avec la version gratuite de Unity (version Personal), nous pouvons exporter notre jeu vers iOS, Android, Windows et OS X.

Pour le reste des plateformes faudrait basculer à une des versions payantes disponibles.

Unity est défini comme étant un moteur de jeu.

Qu'est ce qu'un moteur de jeu ?

Selon wikipédia, un moteur de jeu est un ensemble de composants logiciels qui effectuent des calculs de géométrie et de physique utilisés dans les jeux vidéo et dont l'ensemble forme un simulateur en temps réel souple qui reproduit les caractéristiques des mondes imaginaires dans lesquels se déroulent les jeux.

Ce moteur de jeu est donc un ensemble d'outils pour faire un ou plusieurs jeux :

- ✗ Editeur de scène
- ✗ Moteur de rendu 3D
- ✗ Moteur physique
- ✗ Rendu audio
- ✗ Réseau
- ✗ Intelligence artificielle
- ✗ Scripts : comportements, événements, ...
- ✗ Animation
- ✗ ...

Il existe d'autres moteur de jeu sur le marché comme : CryEngine, Unreal Engine, Torque 3D, etc.

III. Réalisation

1. Modélisation du personnage sous Blender

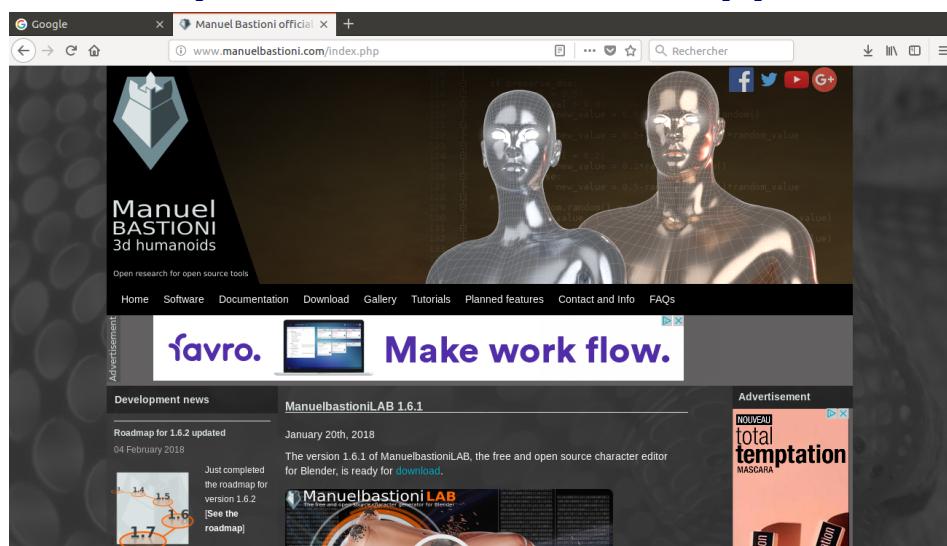
Après avoir montré un peu l'interface de Blender, il est temps pour nous d'attaquer la modélisation du personnage 3D qui va se présenter en réalité augmentée lorsqu'on va lancer notre application. La modélisation d'une forme aussi complexe qu'un être humain prend beaucoup de temps sur les formes et le perfectionnement des contours. Dans ce souci de précision, nous avons téléchargé un add-on (module) à Blender qui nous permet de créer des personnages à partir d'une base solide et de le modifier à notre guise.

a. L'add-on ManuelBastioni

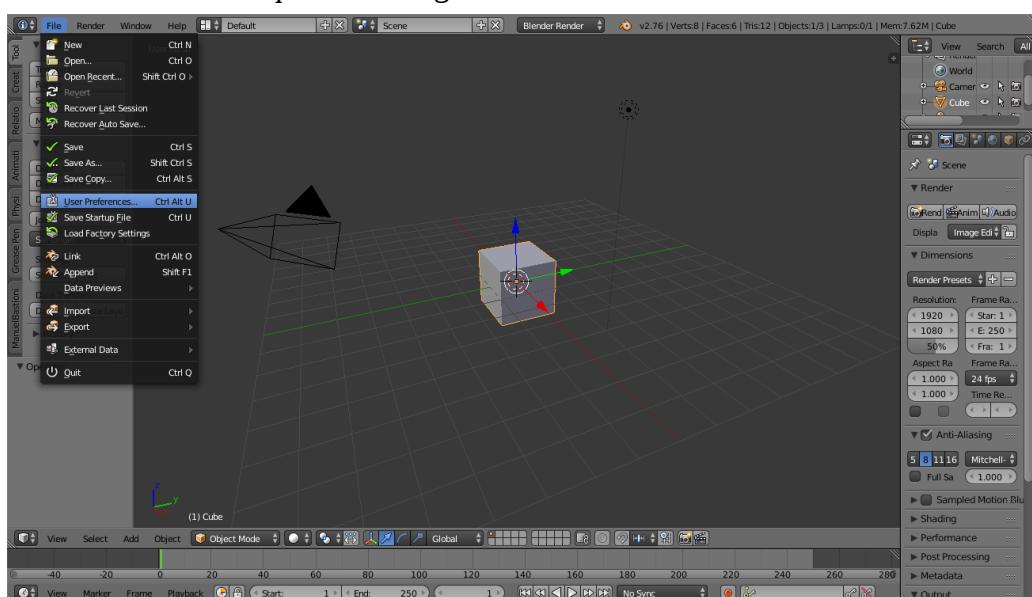
Le but de l'add-on manuelbastioni est de vous apporter un ensemble d'outils avancées pour faire de Blender un puissant laboratoire pour la création de personnages humanoïdes réalistes.

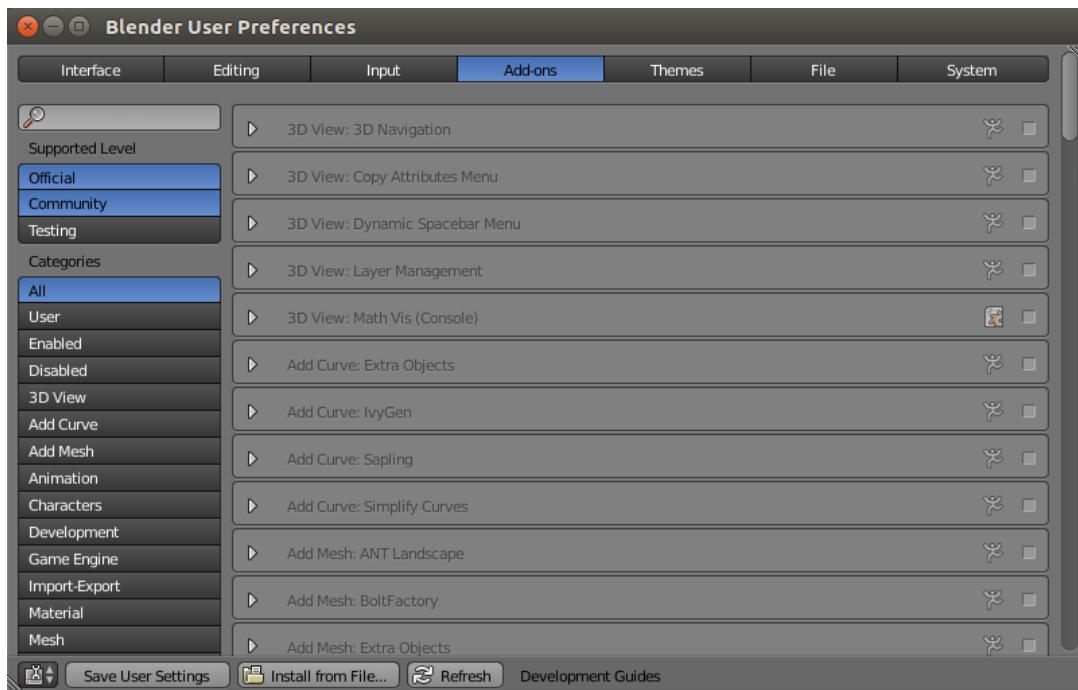
Vous pouvez le télécharger sur le site officiel de [manuelbastioni](http://www.manuelbastioni.com/manuellab.php) dans la section **Download** :

<http://www.manuelbastioni.com/manuellab.php>



Une fois que vous le téléchargez et l'enregistrez sur votre machine, il faut maintenant l'ajouter à notre logiciel Blender. Pour ce faire ouvrez Blender puis cliquez sur le menu File > User Preferences et ensuite vous cliquez sur l'onglet **Add-ons**

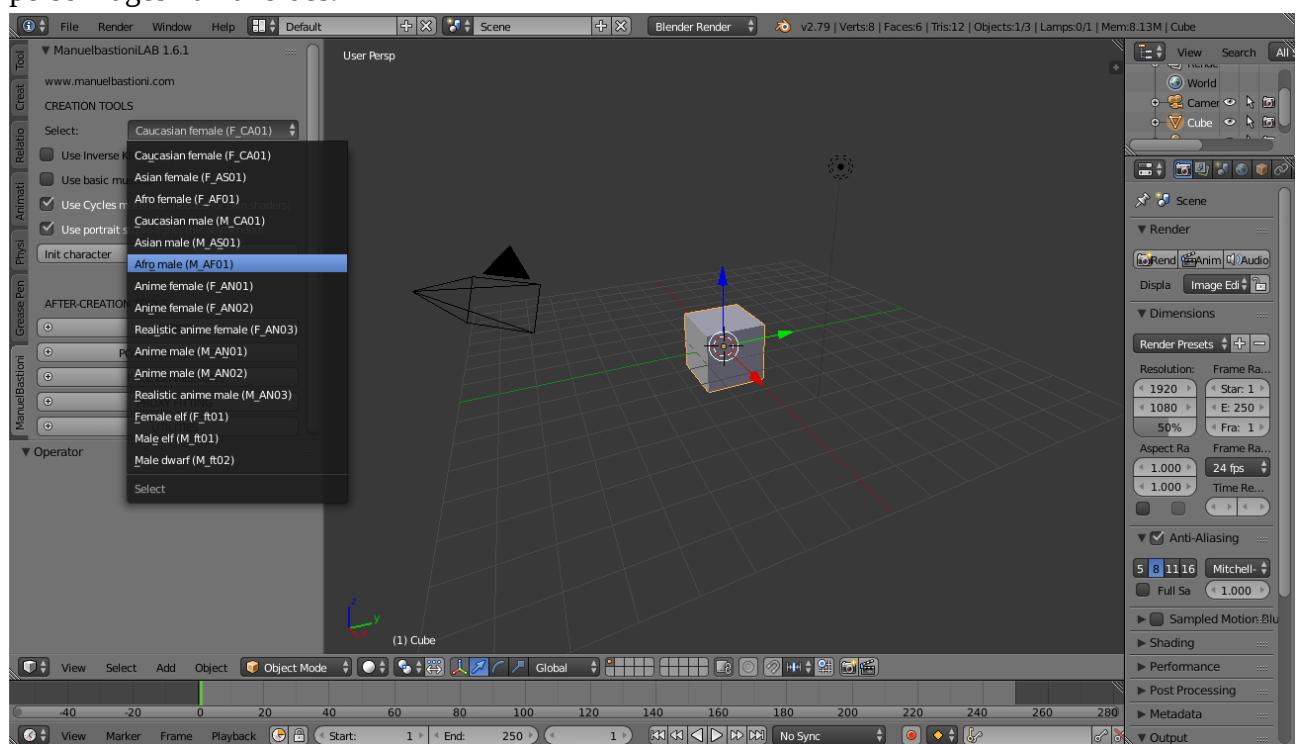




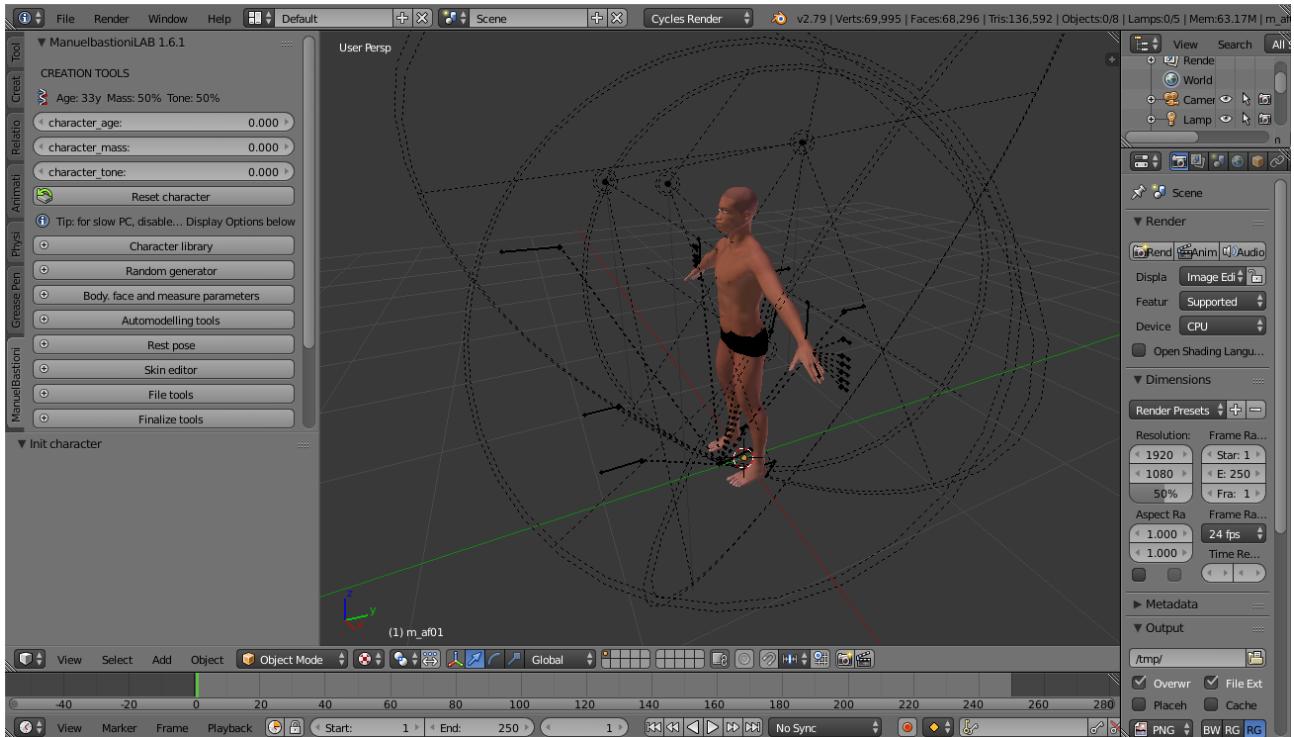
Vous devez maintenant cliquer sur *Install from File* et récupérer sur votre ordinateur l'archive de manuelbastioni dans le répertoire où vous l'avez enregistré.

Une fois manuelbastioni ajouté au Blender, vous avez là un puissant alliage qui va vous permettre de créer des personnages réalistes, de modifier leurs propriétés ou de les faire animé.

Si votre installation s'est bien passé, vous apercevez l'onlet manuelbastioni à gauche de l'interface de Blender et vous pourrez cliquer dessus pour créer et ajouter à la scène vos personnages humanoïdes.



Une fois que vous sélectionnez les différents paramètres de votre personnage, vous devez cliquer sur ***Init character*** pour le voir apparaître sur la scène vue 3D de Blender.



Vous pourrez toujours apporter des modifications à votre personnage.

l'add-on manuelbastioni nous offre tout un tas de paramètres modifiables allant des caractéristiques du personnage aux expressions faciales de ce dernier. Ainsi on peut faire parler notre personnage en lui bougeant les lèvres, fermer ses yeux, etc.

b. Ajout des clés de référence (keys)

Vous l'aurez deviné, notre but ici est de pouvoir faire parler notre personnage afin qu'il se présente. Pour réaliser cela, il nous faut ajouter des clés de référence à notre personnage c'est à dire lui montrer comment on prononce la lettre A, la lettre O, la lettre U, etc.

On n'est pas obligé de faire des clés pour toutes les lettres de l'alphabet mais il faut juste en avoir suffisamment (une dizaine par exemple) pour permettre au personnage de parler de manière assez réaliste.

Qu'est ce qu'une clé dans manuelbastioni ?

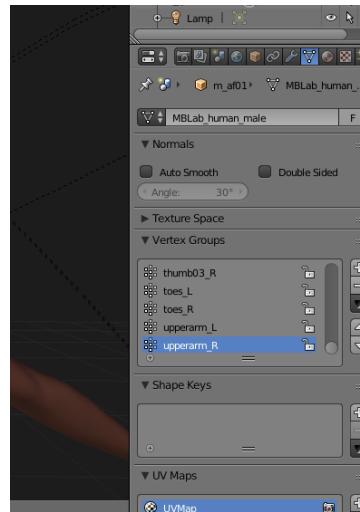
Une clé est un élément de base principalement utilisé pour les articulations ou l'animation du visage (les fameuses expressions faciales).

Dans notre projet, les clés ajoutées concernent uniquement les articulations des lèvres du personnage afin de spécifier la manière de prononcer certaines lettres.

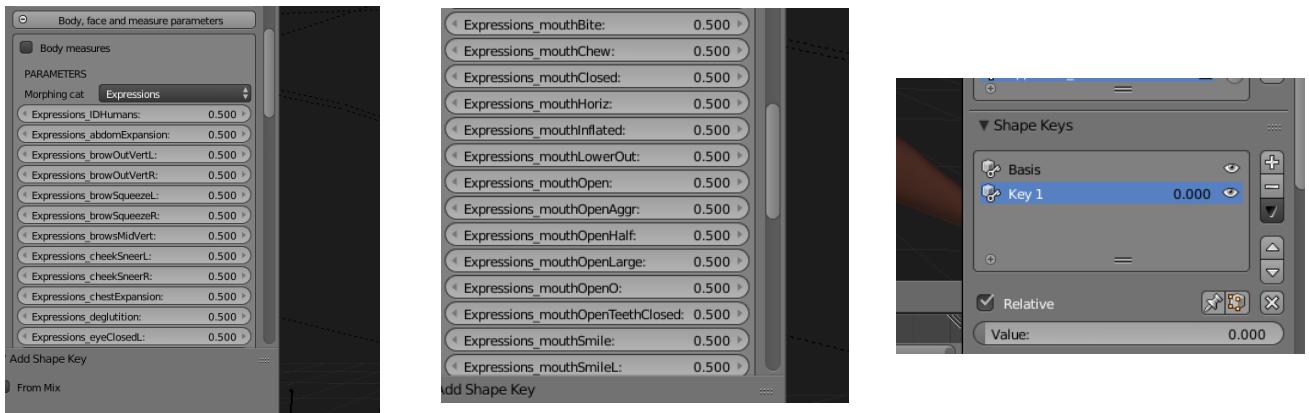
Par exemple voilà la clé correspondant à la prononciation de la lettre A.



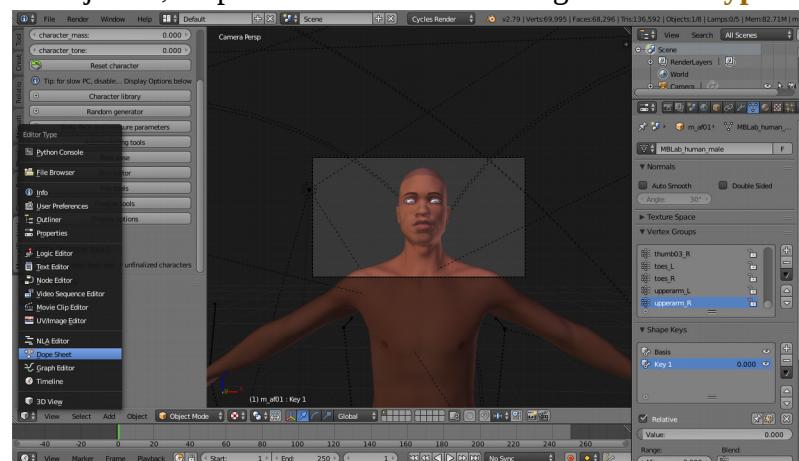
Pour ajouter vos clés, il faut aller dans l'onglet **Data**, dans la zone des propriétés et aller dans le sous-onglet **Shape Keys**



Pour en ajouter, vous cliquer sur le **+** et vous modifier les paramètres (lèvres plus précisément) du personnage et ainsi de suite.



Une fois toutes les clés ajoutées, on peut les visualiser via l'onglet **Editor Type > Dope Sheet**

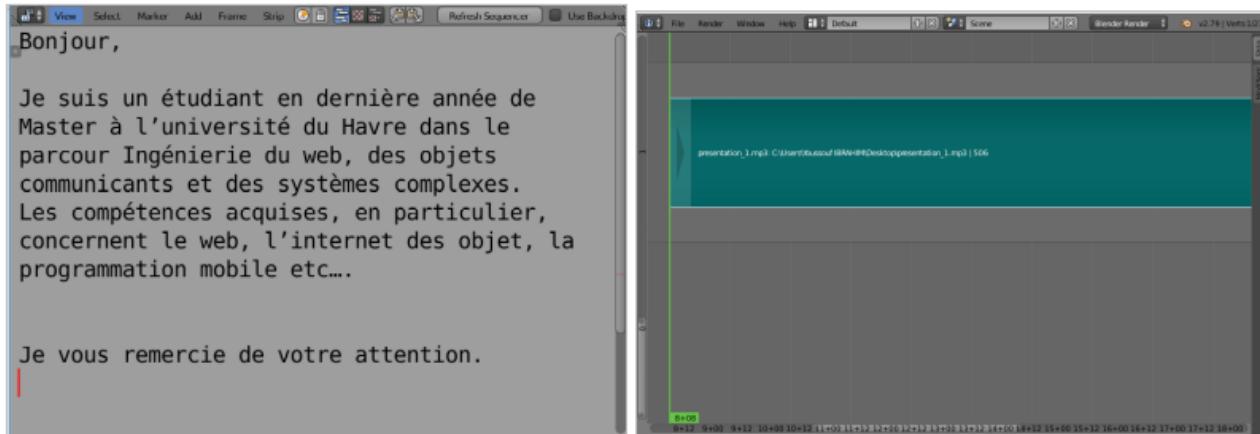


c. Synchronisation audio et keys

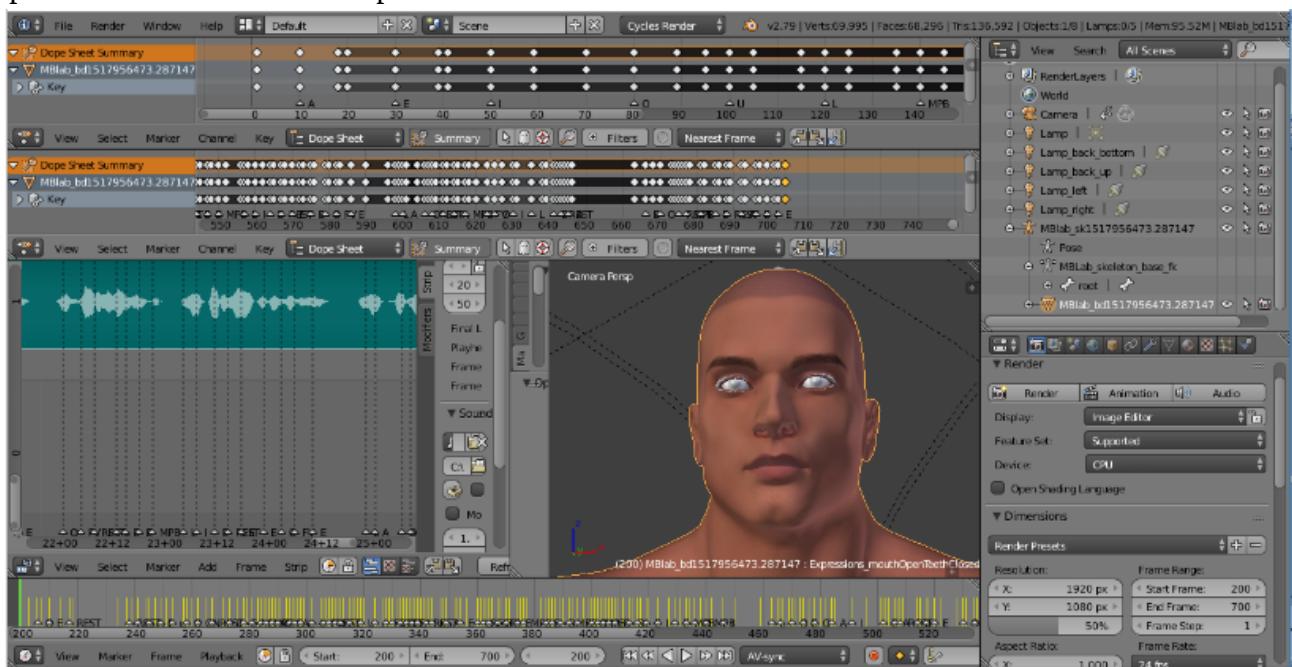
Le personnage 3D réalisé doit parler et se présenter quand on lance l'application.

Mais pour dire quoi en fait?

Pour pallier à cela nous avons créé un petit texte et un fichier audio de présentation et nous l'avons enregistré sur notre ordinateur avant de l'importer dans le logiciel Blender pour pouvoir l'utiliser sur notre personnage.



Désormais on a le personnage, les clés pour la prononciation de quelques lettres, le fichier audio. Nous avons par la suite synchroniser tous ces fichiers afin que le personnage ait l'air d'une personne réaliste en train de parler.



Une fois la création du personnage, l'ajout des clés, l'ajout du texte et de l'audio correspondant, et que l'on a synchronisé le tout, il faut maintenant exporter notre travail sous l'environnement Unity afin de réaliser notre application destinée aux mobiles sous Android.

2. Compilation sous Unity

La création de l'application Android en tant que telle s'est faite sous l'environnement Unity connu pour sa rapidité aux prototypages. C'est dans Unity que l'on va importer nos fichiers préalablement construit avec Blender pour le personnage 3D et ainsi intégrer de la réalité augmentée.

Comment faire de la réalité augmentée dans Unity ?

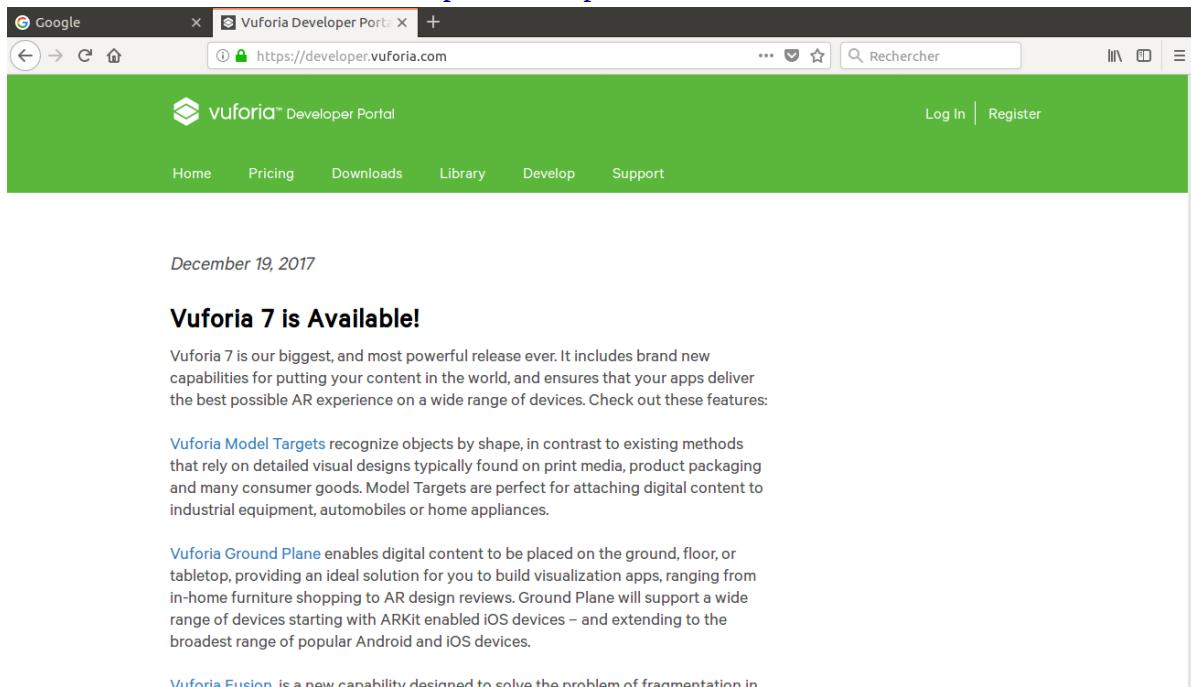
Pour faire de la réalité augmentée avec Unity, nous avons utilisé un kit spécial nommé **Vuforia**.

a. le kit Vuforia

Vuforia est un kit de développement pour la création d'applications de réalité augmentée pour les appareils mobiles. Il se couple parfaitement avec Unity et peut s'intégrer dans les fichiers du projet comme module.

Pour pouvoir l'utiliser nous l'avons téléchargé sur le portail développeur du site de vuforia.

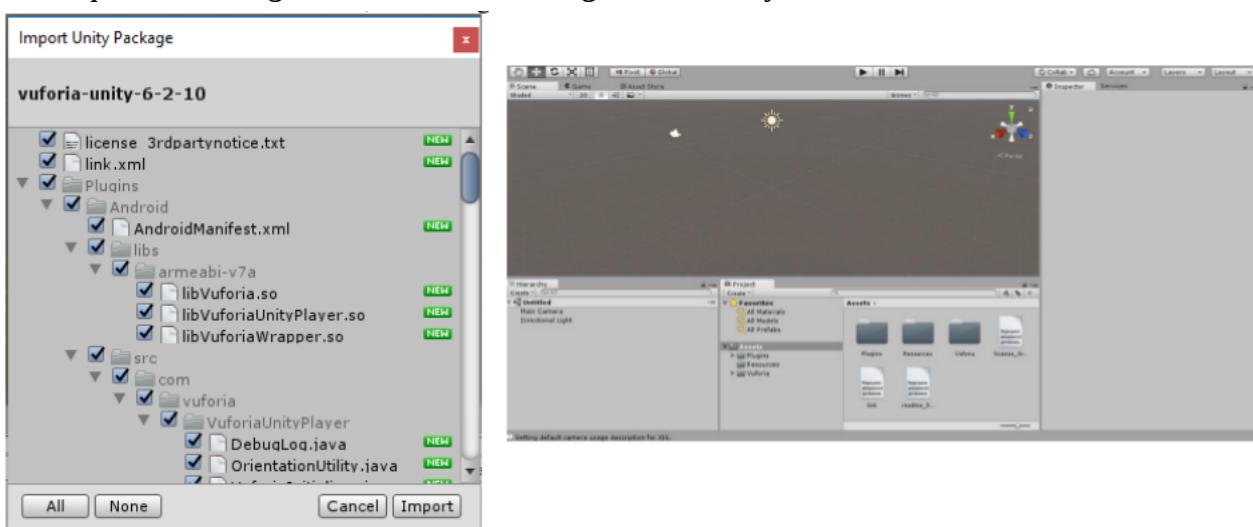
<https://developer.vuforia.com/>



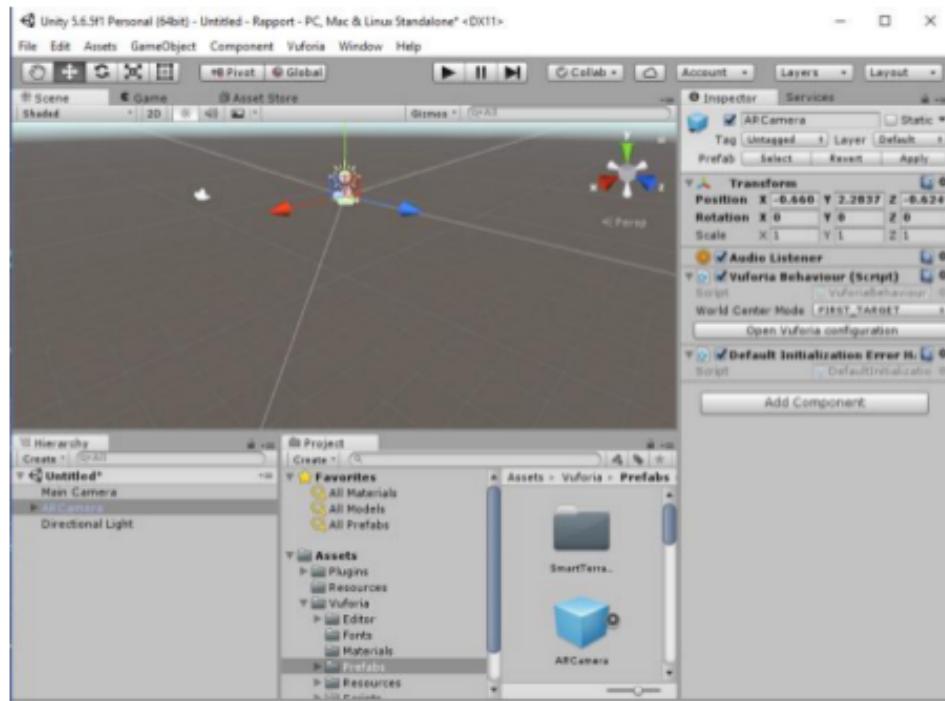
The screenshot shows the Vuforia Developer Portal homepage. At the top, there's a navigation bar with links for Home, Pricing, Downloads, Library, Develop, and Support. On the right side of the header, there are 'Log In' and 'Register' buttons. Below the header, a green banner displays the text 'December 19, 2017' and 'Vuforia 7 is Available!'. The main content area features several sections: 'Vuforia Model Targets' (describing object recognition), 'Vuforia Ground Plane' (describing digital content placement), and 'Vuforia Fusion' (described as a new capability). There are also links for 'Documentation' and 'API Reference'.

Il va falloir créer un compte dans ce site pour pouvoir l'utiliser.

Une fois que le téléchargement fini, il faut l'intégrer dans Unity via la section **Assets**.



Automatiquement après ça, vous aurez un dossier nommé **vuforia** dans la section **Assets** de Unity. Ouvrez le dossier **vuforia → prefabs**, vous y trouvez la caméra vuforia. Prenez cette caméra et glisser la dans la scène ou dans la section **Hierarchy** de Unity.



Mais à ce moment là, Vuforia n'est pas encore active et vous ne pouvez pas encore faire de la réalité augmentée. Pour l'activer, il faudra dans un premier temps vous connecter sur le site de vuforia et récupérer votre clé (qui permet d'utiliser vuforia) et puis ajouter cette clé dans Unity.

CarteVisite Edit Name Delete License Key

Type: Develop Status: Active

Please copy the license key below into your app:

```
AdVZahE:///AAAAmbLxRQEXDaaqQE2+R4QId8Xe+teG5C8rFq3
NMFjzJGDa+sEpHwI1d4REX5o+XATLd12jhm2dQ7ctno/up1sa+
m2e171Ne24wtpBpB/180B2/7EHnvypjwDMeIzL+B9o7UWjeVf7U
OR1Ce2s+HFKq1mL+Vb3C011LgvfTg/3Rkhc55P0xbq2m0gf6JV
BQgywfLjh/ZDp0ysD3c19pWvN7d/w9nqxxA1eog#9tewxDzYL
bkk886/sGyHn1m4cqfT112FaXiity2BRVn12AOfpSyyrsO+gRjW
EdUEK83+Qfogx09psfPV3+NaFkJy6pu89VH78jFLTQLebgeMh3hk
z2VhMu749+RcKo90
```

Dans la section **Develop > License Manager** de vuforia, vous pouvez récupérer la clé. Vous la copiez juste comme indiqué, pour ensuite pouvoir la coller dans la section correspondante dans Unity (**App License Key**). Une fois que vous faites ça, cela veut dire que Vuforia est opérationnel.

b. notion d'Image Target dans Vuforia

Une fois que nous connectons Vuforia à notre logiciel Unity, il reste toujours le problème de comment tout ceci fonctionne. En fait ce qu'il faut comprendre c'est que Vuforia scanne et tente de reconnaître une image cible préalablement enregistrée dans vuforia et lorsqu'il l'aperçoit est capable de lancer par dessus cette image, un modèle objet en réalité augmentée (ici notre personnage 3D qui se présente).

Les **Image Target** donc, ce sont des images que le SDK Vuforia peut détecter et suivre. C'est comme tenter de détecter des QR Code, mais contrairement à ces derniers, les Image Target (cible image) n'ont pas besoin de codes spéciaux en noir et blanc pour être reconnus. Le SDK Vuforia détecte et suit les fonctionnalités qui se trouvent naturellement dans l'image.

Pour ajouter un Image Target, il faut utiliser le Target Manager sur le site de vuforia et cliquer sur **Add Database**.

The screenshot shows the Vuforia Developer Portal's Target Manager interface. At the top, there's a navigation bar with links for Home, Pricing, Downloads, Library, Develop, and Support. Below that is a sub-navigation bar with License Manager and Target Manager. The main area is titled "Target Manager" and contains the message "Use the Target Manager to create and manage databases and targets." A "Add Database" button is visible. A table lists a single database entry:

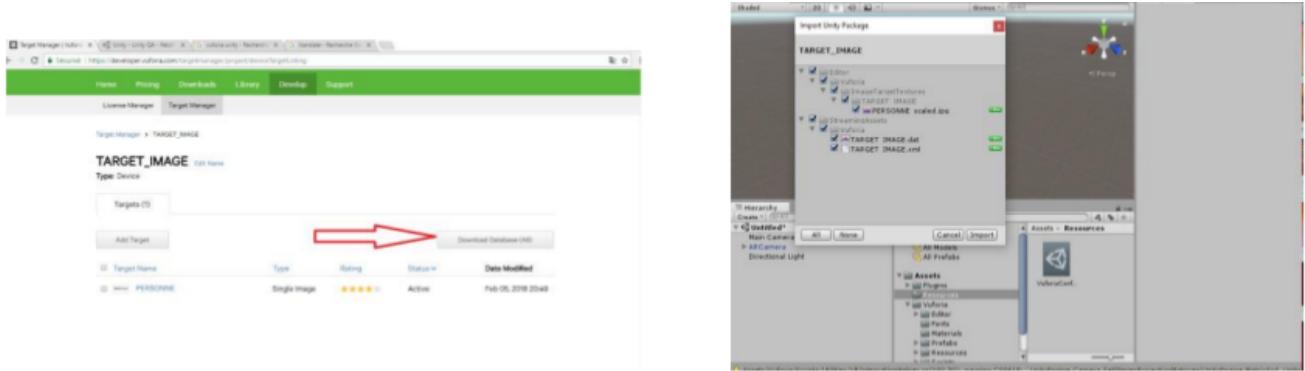
Database	Type	Targets	Date Modified
TARGET_IMAGE	Device	1	Feb 05, 2018 20:49

Ensuite, on ajoute la cible (qui peut être autre chose qu'une image) en important une image présente sur notre ordinateur par exemple.

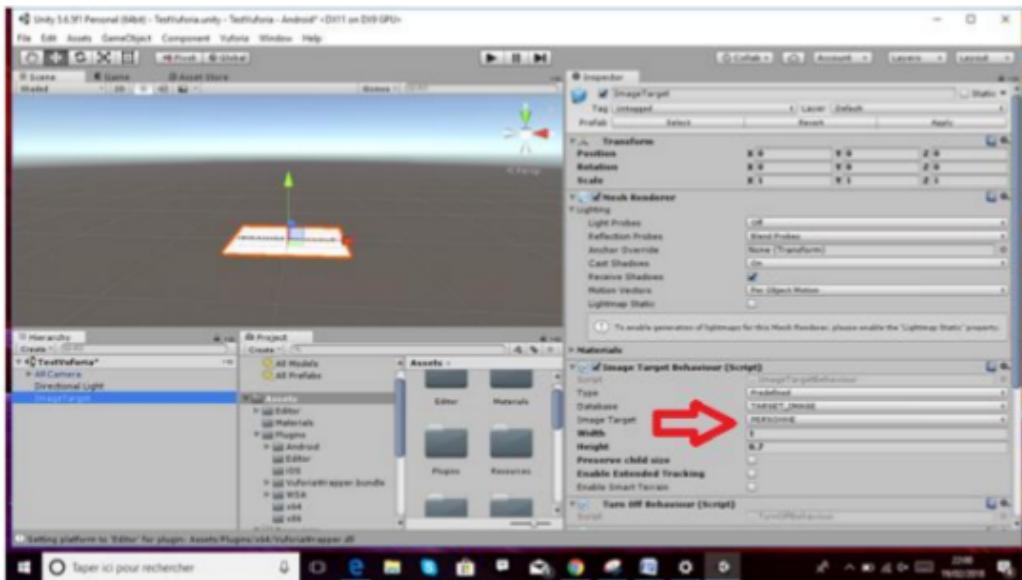
The screenshot shows the "Add Target" form. It has several sections: "Type:" with options for Single Image, Cuboid, Cylinder, and 3D Object; "File:" with a "Choose File" button and a "Browse..." button; "Width:" with a text input field and a note about target scale; and "Name:" with a text input field and a note about uniqueness. There is also a note at the bottom about naming rules.

Une fois l'image importée, Vuforia y ajoute des points de repères pour pouvoir le détecter. Maintenant nous devons télécharger l'image target créée et l'importer dans le dossier Assets de Unity pour pouvoir l'utiliser.

On le glisse sur la scène pour le déposer. L'image va constituer la première brique de notre application car c'est sa détection qui lance l'animation 3D.
Notre image target est donc l'élément déclencheur de notre application de réalité augmentée.
Il faut donc bien veiller à la créer proprement sur le site de Vuforia et l'importer dans Unity.



A noter que lorsque vous mettez l'image sur la scène, vous pouvez changer son nom.

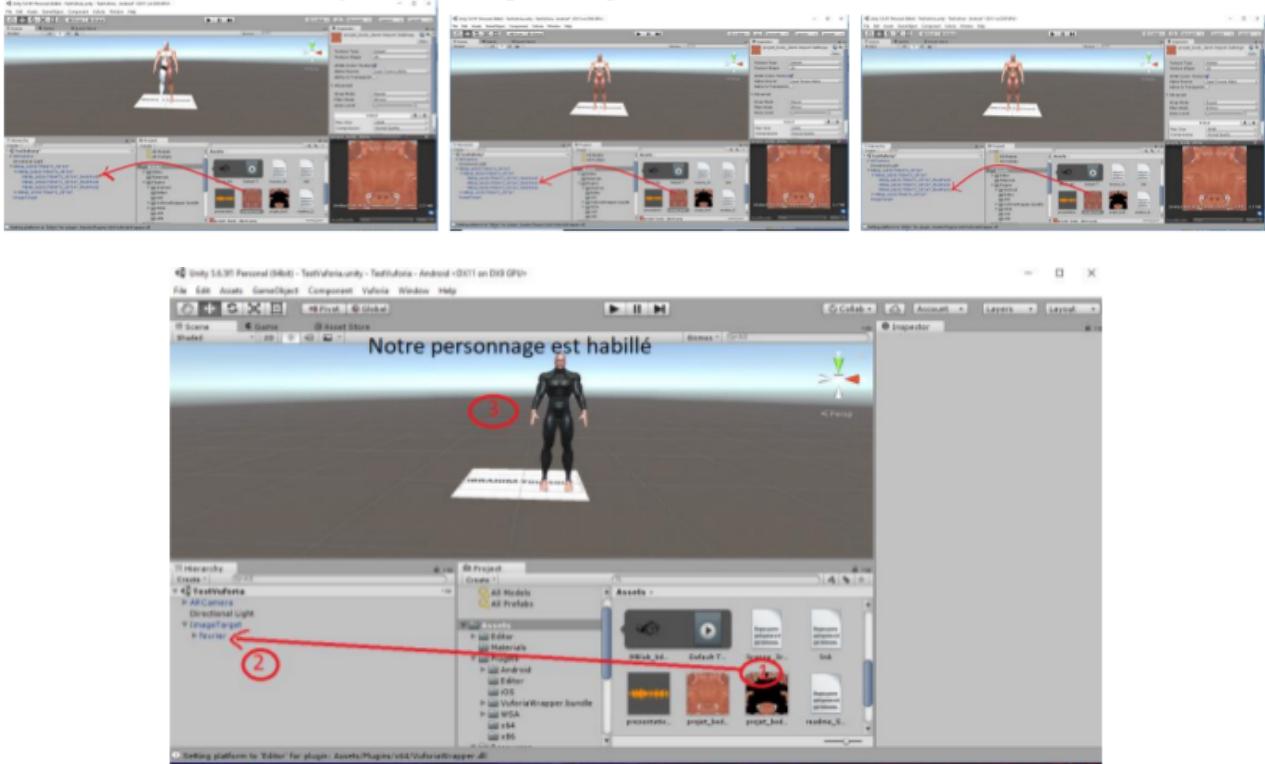


c. intégrer les fichiers Blender

Une fois cela fait, il est temps d'intégrer les fichiers issus de Blender, toujours dans le dossier **Assets** de Unity, vous faites un clic droit puis **Import new Asset** et vous sélectionnez le fichier **.blend** de votre projet Blender. Puis vous la glissez sur l'onglet **Hierarchy** de Unity.



La texture et l'audio ne sont pas automatiquement connectés au personnage lorsqu'on l'importe dans Unity. Il faut donc dans un premier temps relier la texture disponible dans les fichiers au modèle 3D en faisant simplement un cliqué glisser dans le personnage de l'onglet **Hierarchy**.

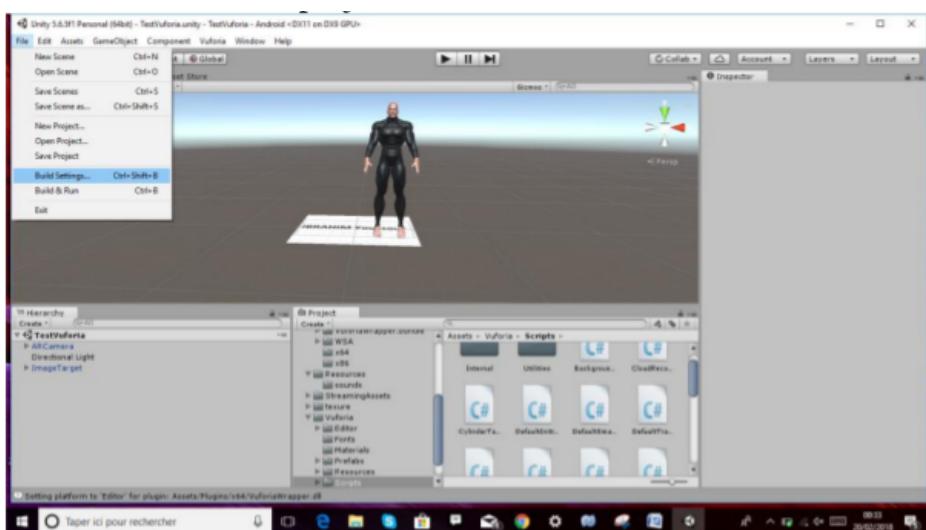


En ce qui concerne le fichier audio de présentation du personnage, nous l'avons importé dans le dossier Assets et ensuite ajouter un script C# de telle sorte que l'audio démarre lorsque la reconnaissance de l'image est faite et que le personnage 3D apparaît en réalité augmentée.

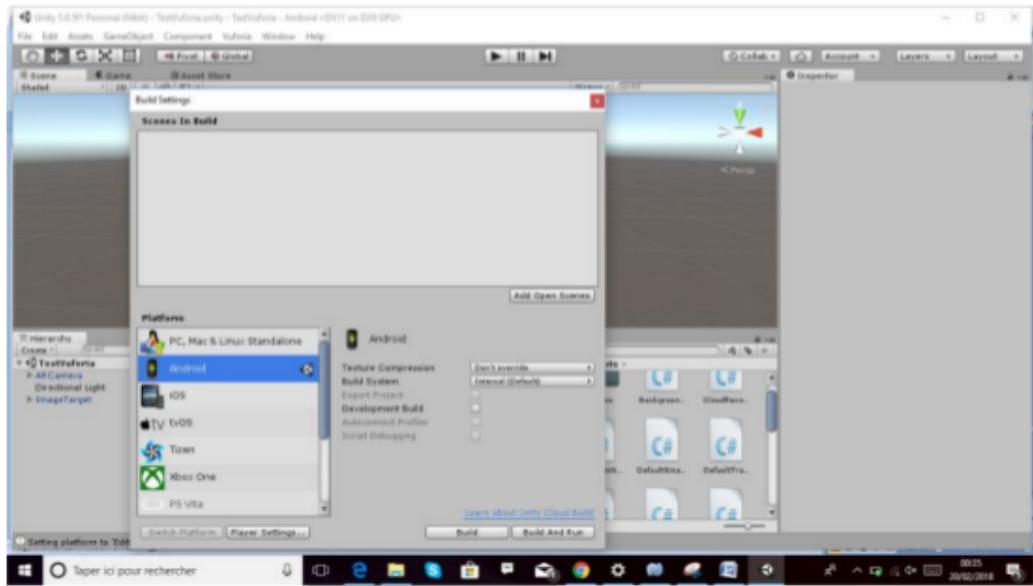
d. compiler pour Android

Tout ceci étant réalisé, nous sommes passés au build du projet, c'est à dire la partie où l'on compile notre projet en destination des plateformes cibles. Ici notre but est de créer une application pour Android donc nous avons fait une compilation destinée aux mobiles Android.

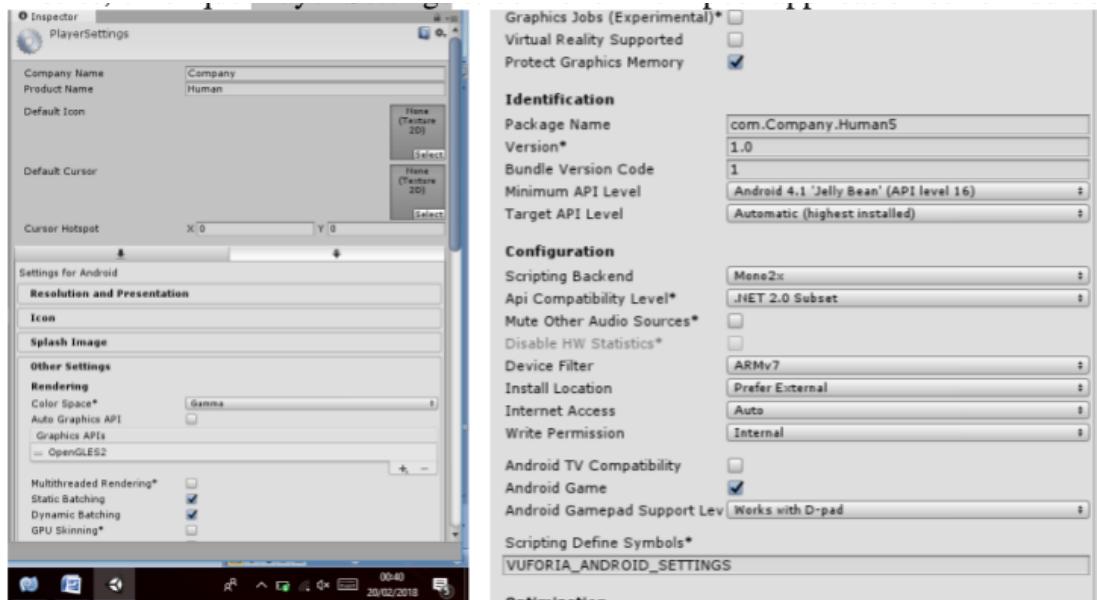
Pour réaliser une compilation, il faut cliquer sur le menu File de Unity puis sur Build Settings.



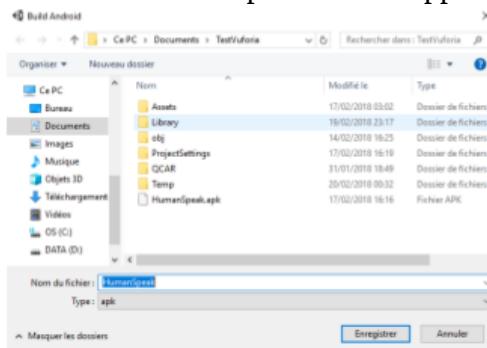
Ensuite une fenêtre de dialogue s'ouvre devant vous pour que vous choisissez la plateforme cible (Android, PC, iOS, etc.). Pour nous ce sera Android.



Après cela, nous allons cliquer sur Player Settings pour modifier les paramètres du projet (lui donner un nom suivant la syntaxe de Unity, choisir le niveau minimum de l'API, etc.).



Et enfin on clique sur Build pour construire un fichier .apk que l'on enregistre sur notre ordinateur pour plus tard l'envoyer sur des mobiles Android pour tester l'application.



3. Difficultés rencontrées

Durant la réalisation de ce projet, nous avons eu quelques difficultés que ce soit au niveau des 2 outils utilisés (Blender et Unity) ou pour la compréhension du monde de la réalité augmentée.

Voilà énumérées ci-dessous les difficultés rencontrées :

- ✓ **Prise en main** : La prise en main de ces 2 outils a été un peu longue et fastidieux parsemée par des erreurs de compilation, des modèles non conformes aux attentes, etc.
Savoir utiliser Blender pour réaliser des modèles, des animations, mémoriser les raccourcis clavier qui permettent d'aller vite est un travail long et demande de la patience. La création de personnage 3D ou modèles de quoi que ce soit demande un certain degré de perfectionnement et de précision.
Le logiciel Unity aussi a exigé du temps afin de l'utiliser proprement pour pouvoir construire notre projet, y intégrer des fichiers provenant de Blender, intégrer de la réalité augmentée, compiler, etc.
- ✓ **Recherche documentation** : Trouver des tutoriels sur Blender et Unity en français est très difficile et la plupart quand ils sont en Anglais c'est pas vraiment fait pour des débutants.
Du coup on s'est rabattu sur ce qu'on avait et couplé à notre curiosité intellectuelle, nous avons pu atteindre nos objectifs petit à petit.
- ✓ **Format export de fichiers** : Lorsqu'on a fini la création du personnage 3D sous Blender, il fallait l'exporter pour pouvoir l'utiliser dans le logiciel Unity afin de construire l'application. Mais il a fallu choisir le bon format parmi ceux proposés par Blender et qui ne perd pas certaines données utiles. Après avoir testé le format **.obj**, le format **.fbx**, etc, nous avons choisi d'utiliser le format **.blend** du projet car il nous permettait d'emmener toute la structure du personnage ainsi que ces différentes propriétés dans Unity.
- ✓ **Faire de la réalité augmentée avec Unity** : Pour faire de la réalité augmentée avec Unity, nous avons recouru à un module nommé **Vuforia** et nous l'avons intégré aux fichiers du projet ce qui veut dire qu'il a fallu se documenter et comprendre le fonctionnement de ce module.
- ✓ **La gestion des textures sous Unity** : les textures désignent les images qui habillent nos personnages. Lorsqu'on crée un modèle 3D, on peut lui assigner une texture c'est à dire qui la colorie afin de faire ressembler le modèle à une image réaliste. Cette manipulation a été faite avec Blender. Mais une fois le fichier exporté vers Unity, le modèle perd sa texture et redevient grisâtre comme à l'origine. Il nous a fallu voir comment gérer les textures sous Unity et les appliquer sur les modèles afin de régler ce problème.
- ✓ **Compilation pour Android** : Nous avons également rencontré des erreurs au cours de la compilation destinée aux mobiles pour Android. Il faut bien vérifier que le SDK Android est bien reconnu par Unity. Mais également quand on fait une compilation sous Unity, il y'a de petits détails qui peuvent faire planter le Build, comme par exemple la façon de nommer son projet (majuscule au début).

IV. Résultats et Tests

Dans cette partie, nous allons montrer les résultats obtenus lors de la conception du personnage 3D avec Blender ainsi que lors de la compilation pour Android via Unity.

Une fois l'application compilée sous Unity et que nous l'avons installé sur nos Smartphones Android, nous avons commencé à le tester sur 2 cartes de visites différentes.



A noter que même si vous n'avez pas l'application installée, vous pourrez le télécharger en scannant le QR Code qui est dessus et donc automatiquement vous aurez le lien de téléchargement du fichier apk de notre application.

L'une des deux cartes a été préenregistré dans Vuforia afin que lorsqu'on le scanne, l'application le reconnaît et lance par dessus la réalité augmentée montrant le personnage 3D qui se présente.

Nous avons imprimé les 2 cartes afin de percevoir la différence voulue quand on teste.

Quand on scanne la carte préenregistrée, voilà ce qu'on obtient : L'animation du personnage 3D se lance et il se présente :



Et Quand on scanne la carte non préenregistrée, rien ne se passe (ce qui est normal) :



CONCLUSION

Dans ce projet, il s'agissait de réaliser une application Android qui introduit le concept de la réalité augmentée sur des cartes de visite afin qu'une animation 3D de la personne se lance dès que la carte est reconnue par l'application.

Pour mener à bien ce projet, on a orienté la réflexion en deux parties symbolisée par l'utilisation de 2 outils : **Blender** et **Unity**.

Dans un premier temps nous nous sommes contrés sur la conception d'une animation 3D d'un personnage qui se présente en utilisant le puissant logiciel Blender.

Ensuite, une fois cette animation réalisée, nous l'avons transféré sous Unity pour implémenter la réalité augmentée et pouvoir faire une compilation pour mobiles Android.

Travailler sur ce projet nous a permis de sortir des chantiers battus et de découvrir un monde fascinant et instructif à savoir le monde de la 3D mais également d'acquérir de solides connaissances en terme de réalité augmentée en découvrant les nombreuses possibilités qu'offre cette dernière.

Comme perspective pour ce projet, il y a la possibilité d'ajouter des boutons virtuels sur la réalité augmentée afin d'interagir directement avec le visuel qui nous est proposé.

On peut également aller plus loin en enregistrant plusieurs animations 3D sous un certain format dans une base de données et ensuite connecter notre application à cette base données afin qu'elle puisse scanner plusieurs cartes de visite différentes et en fonction de l'identité de chaque personne pouvoir afficher l'animation correspondante stockée au préalable.

Bibliographie et Webographie

Lien du dépôt github du projet avec les fichiers Blender, les fichiers Unity, le fichier apk de l'application ainsi que les images pour tester :

- ✓ <https://github.com/IBRAHIM-ELMI/CarteVisiteAugment-e.git>

Liens pour bien comprendre le principe de la réalité augmentée :

- ✓ https://fr.wikipedia.org/wiki/R%C3%A9alit%C3%A9_augment%C3%A9e
- ✓ <http://www.augmented-reality.fr/cest-quoi-la-realite-augmentee/>
- ✓ <https://www.realite-virtuelle.com/definition-realite-augmentee>
- ✓ <https://fredcavazza.net/2017/04/24/usages-et-enjeux-de-la-realite-augmentee/>

Liens Blender (prise en main, conception humain, objets 3D, etc.) :

- ✓ <https://www.blender.org/>
- ✓ <https://www.blender.org/support/tutorials/>
- ✓ <http://www.manuelbastioni.com/>
- ✓ <https://openclassrooms.com/courses/debutez-dans-la-3d-avec-blender>
- ✓ <https://www.youtube.com/watch?v=9Y6hgVait7s>
- ✓ <https://fr.tuto.com/blender/creer-des-tonneaux-realistes-blender,40677.html>
- ✓ <https://fr.tuto.com/blender/modeliser-un-logo-en-3d-blender,12165.html>
- ✓ <https://www.pinterest.fr/pin/366199013442831629/>
- ✓ <https://cgmasters.net/free-tutorials/totally-new-to-3d-animation-or-blender-start-here/>
- ✓ <https://www.youtube.com/watch?v=mP4n8kzTiaY&t=131s>

Liens Unity (install, prise en main, compiler pour Android, faire de la Réalité Augmentée, etc.) :

- ✓ <https://unity3d.com/fr>
- ✓ <http://www.unity3d-france.com/unity/>
- ✓ <https://www.youtube.com/watch?v=qjYMHdhJSiE>
- ✓ <https://openclassrooms.com/courses/realisez-votre-premier-jeu-video-avec-unity>
- ✓ <https://docs.unity3d.com/Manual/index.html>
- ✓ <https://docs.unity3d.com/Manual/android-GettingStarted.html>
- ✓ <https://creerunjeuvideo.fr/logiciels-3d/tuto-unity-3d/>
- ✓ <https://www.youtube.com/watch?v=WdfStRynCLw>
- ✓ <https://www.youtube.com/watch?v=HnjbTytHH6U>
- ✓ <https://thinkmobiles.com/blog/how-to-make-augmented-reality-app-with-vuforia/>
- ✓ <https://library.vuforia.com/>
- ✓ <https://library.vuforia.com/articles/Training/getting-started-with-vuforia-in-unity-2017-2-beta.html>