

# Rapport de stage

## MAVIS - Musée Anthropologique Virtuel du Sud

SAID Adam  
L2 CMI Informatique

Tutrice : RODRIGUEZ Nancy  
Equipe : LANSALOT Clara, COSSU Arnaud  
Lien du projet : [MavisWeb](#)



## Table des matières

<b>1</b>	<b>Remerciements</b>	<b>3</b>
<b>2</b>	<b>Mise en contexte</b>	<b>4</b>
2.1	Collaborations . . . . .	4
2.2	Le cahier des charges . . . . .	4
2.3	Contexte socio culturel . . . . .	4
2.4	Contexte sanitaire . . . . .	5
<b>3</b>	<b>Avancée du projet</b>	<b>6</b>
3.1	Mise en place . . . . .	6
3.2	Environnement . . . . .	6
3.3	Bâtiments . . . . .	8
3.4	Interactions . . . . .	8
3.5	Hébergement . . . . .	8
<b>4</b>	<b>Problèmes rencontrés</b>	<b>10</b>
4.1	Collaboration . . . . .	10
4.2	Performances . . . . .	10
4.3	Communication . . . . .	10
4.4	Export . . . . .	12
4.5	VR . . . . .	12
4.6	Encapsulation de l'export . . . . .	12
<b>5</b>	<b>Nettoyage d'objets 3D</b>	<b>14</b>
<b>6</b>	<b>Conclusion</b>	<b>16</b>
6.1	Apports personnels et perspectives d'avenir . . . . .	16
6.2	Résultats obtenus . . . . .	16
6.3	Sources . . . . .	17

## **1 Remerciements**

Avant toutes choses, j'aimerais remercier Mme. Nancy Rodriguez pour nous avoir guidé et assisté durant toute la durée du stage et pour nous avoir confié la réalisation de ce projet. Deuxièmement je voudrais remercier les partenaires de l'équipe MAVIS qui nous ont fait confiance pour leur projet et également Mme. Baert pour nous avoir aidé à trouver ce stage et nous avoir mis en relation avec Mme. Rodriguez.

## 2 Mise en contexte

### 2.1 Collaborations

Ce projet a été réalisé par la collaboration de deux organismes. Nous avons tout d'abord le LIRMM : Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier ainsi que l'équipe MAVIS : Musée Anthropologique Virtuel du Sud dirigé par Andrés Burbano, notre partenaire Colombien et Vladimir Montaña, un anthropologue qui travaille sur le terrain du musée. Cette collaboration est à l'initiative de Mme. Nancy Rodriguez, Maître de Conférences au LIRMM, dont les recherches portent sur la visualisation et l'interaction de la réalité virtuelle et augmentée, elle fut notre tutrice et encadrante durant tout le stage.

### 2.2 Le cahier des charges

Le projet devait être sous forme d'un jeu, jouable en *clavier/souris* mais également en réalité virtuelle avec un casque adapté afin de permettre une immersion plus parlante et une interaction au plus proche de la réalité en permettant aux utilisateurs d'interagir avec les objets du musée. Il devait contenir un environnement reprenant la flore locale ainsi que le musée en lui-même, composé d'un bâtiment principal et de deux secondaires. Dans le musée on devait pouvoir retrouver les objets du musée préalablement modélisés en 3D par les partenaires grâce à la *photogrammétrie* et on devait être capable d'interagir avec les objets en ayant par exemple la possibilité de les faire tourner ou d'afficher des informations. Le projet devait également contenir un panneau d'affichage avec une image que l'on pouvait changer à notre guise.

Le jeu devait être hébergé et jouable depuis un navigateur et également sous forme d'un *exécutable*. Il fallait donc créer un environnement dense ressemblant à celui de la région tout en gardant à l'esprit que les performances devaient être adaptées pour pouvoir jouer depuis le *web*.

### 2.3 Contexte socio culturel

Le but de ce projet était donc de recréer un musée et son environnement en 3D dans un style traditionnel colombien reprenant le musée existant se trouvant dans la région de Corinto (Colombie), afin de pouvoir y exposer des objets datant de l'époque précolombienne. La Colombie est un pays déchiré par la guerre des partis politiques qui a dégénéré en conflit armé ainsi que par la présence des différentes ethnies qui veulent faire entendre leur voix et luttent pour leurs droits. En effet la population du pays est composée d'une grande variété d'ethnies, on retrouve en majeure partie une population métisse, descendante d'une mixité européenne et amérindienne, une grande partie hispanique provenant de la colonisation ainsi que des autochtones amérindiens qui étaient présents originellement sur le territoire et qui tentent de récupérer leurs terres et préserver l'environnement menacé par des projets miniers. C'est cette multiplicité ethnique et la guerre des parties qui ont donné lieu à ces conflits armés, assassinats politiques et l'un des taux d'homicides les plus élevés du monde, ce qui n'aide en rien au développement économique du pays.

De plus, le pays est l'un des premiers producteurs et exportateurs de cannabis mondial malgré que la vente soit encore illégale, ce qui constitue l'un des facteurs majeurs de la situation conflictuelle actuelle. La région de Corinto n'échappe pas aux conflits et c'est donc dans une Colombie divisée entre progressisme et conservatisme que le musée de La Cristalina tente de voir le jour alors que les autorités locales gênent son développement et confisquent les objets du musée.

## 2.4 Contexte sanitaire

Notre objectif a donc été de constituer un environnement reprenant la végétation de la région ainsi que les méthodes de construction des bâtiments afin de retrouver un esprit traditionnel dans nos constructions. En raison de la situation sanitaire actuelle nous n'avons pas pu travailler en laboratoire et avons dû travailler avec nos ressources locales. Faire de la 3D est une tâche qui nécessite des ordinateurs performants et nous n'avions pas de casques de VR afin de faire des tests, ce qui était assez contraignant car nous n'avons pas pu tester le jeu en vision stéréoscopique.

De plus, comme nous travaillions avec des partenaires en Colombie, la communication était assez compliquée, tout d'abord car tout le monde ne parlait pas français, nous avons donc eu besoin de notre tutrice afin de nous traduire ce que disaient les partenaires et avec le décalage horaire et les emplois du temps de chacun, il était difficile de faire des réunions. Par ailleurs, aucun de nous trois (stagiaire) n'avait utilisé Unity ou fait du C# pour les scripts.

## 3 Avancée du projet

### 3.1 Mise en place

Avant de commencer les plans de notre environnement il a fallu nous décider sur la plateforme que nous allions utiliser pour créer cet environnement, plusieurs solutions s'offraient à nous et nous n'en connaissions aucune. Il y avait : **Unity**, très malléable et pratique pour les différents exports (Web, exécutable, VR) mais pas simple d'utilisation, **AltSpaceVR**, conçu pour la VR et qui comporte par défaut un système d'interaction avec les objets, mais pas exportable en web ou application "**standalone**", il faut passer par l'application fournie et il était très compliqué d'importer ses propres objets, puis **Mozilla Hub**, qui est déjà sur le web et est optimisé pour la VR par défaut, mais qui nécessite une bonne connexion pour ne pas **laguer** et qui n'est pas pratique pour le design du terrain.

Après réflexion nous avons décidé d'utiliser **Unity**, plus professionnel et libre.

Ensuite il nous a fallu mettre en place un système pour pouvoir collaborer à plusieurs sur le projet 3D, pour cela nous avions commencé en utilisant le **plugin** de collaboration inclus dans **Unity** mais sans succès. Nous nous sommes donc tournés vers un **git** hébergé sur **GitHub**, ce qui nous permettait à tous d'être toujours à jour du projet et de pouvoir revenir en arrière si besoin.

C'est en mettant en place ce système que nous nous sommes répartis les tâches de manière assez automatique, Arnaud s'est occupé de la conceptualisation des bâtiments 3D dans **SolidWorks**, Clara s'est occupée des **scripts** en **C#** pour les interactions avec les objets, les mouvements du personnage et les menus de pause et de démarrage, quant à moi je me suis occupé de la création du terrain, l'ajout de la végétation, la mise en place des bâtiments et objets, de la texturation, des exports et de l'hébergement.

### 3.2 Environnement

Afin de créer l'environnement, nous avons au préalable dessiné un plan de comment nous imaginions l'environnement et le chemin à parcourir jusqu'au musée qui permettait à l'utilisateur de rentrer dans l'ambiance colombienne [Figure 1]. Nous avons décidé de faire un petit chemin sinueux entouré de végétation infranchissable avec un petit cours d'eau à traverser. Une fois le plan fait, tout a démarré d'un terrain plat auquel on a rajouté des reliefs et textures de montagnes et plaines afin de se rapprocher de la topographie de la Colombie en utilisant des brosses spéciales et plusieurs empilements de textures pour donner un sol sablonneux, terreux et parsemé de cailloux. Puis nous avons ajouté deux cours d'eau représentant ceux que nous avons vu en photo et une végétation luxuriante. Cette partie était assez délicate car nous ne connaissions pas la région et il nous a fallu trouver des modèles d'arbres et de plantes typiques en 3D mais gratuits. Nous avons donc utilisé des modèles d'arbres et de plantes en **polygones** mais aussi des "**billboards**" qui permettent de donner du volume sans perdre en performance et nous avons également ajouté un arrière-plan avec une image de jungle afin de donner une impression de profondeur dans la scène. Un peu plus tard nous avons dû rajouter des **billboards** de champs de cannabis comme le demandaient les partenaires car le pays et l'un des premiers producteurs mondiaux [Figure 2].

En parallèle nous avons ajouté le musée, le kiosque, le bâtiment annexe ainsi que le pont modélisé par Arnaud à partir des photos que nous avons reçues. Un travail de collecte de données et d'observations a été effectué afin de retranscrire au mieux les constructions typiques de la Colombie comme on peut le voir sur les ornements de portes du musée [Figure 3].

Afin d'ajouter encore plus de réalisme à la scène et une impression de grandeur, tout le fond de la scène, qu'on appelle **Skybox**, comporte une image à 360° représentant un paysage de plaines et montagnes ressemblant à ceux qu'on pourrait trouver en Colombie.

Une fois l'environnement terminé nous avons ajouté des bruits d'eau et de forêt tropicale pour une meilleure immersion, avec une spatialisation du son, ce qui fait que dans la forêt on entend beaucoup les oiseaux mais les sons baissent en s'approchant du musée.

L'un des objectifs était d'ajouter un panneau d'affichage avec une image qui pouvait être changée en ajoutant une nouvelle dans un dossier ou depuis une adresse web. Pour ce faire, j'ai utilisé un modèle de panneau d'affichage trouvé sur internet auquel j'ai rajouté un plan avec une image par défaut et j'y ai ajouté un **script** pour charger l'image depuis une URL [Figure 2].

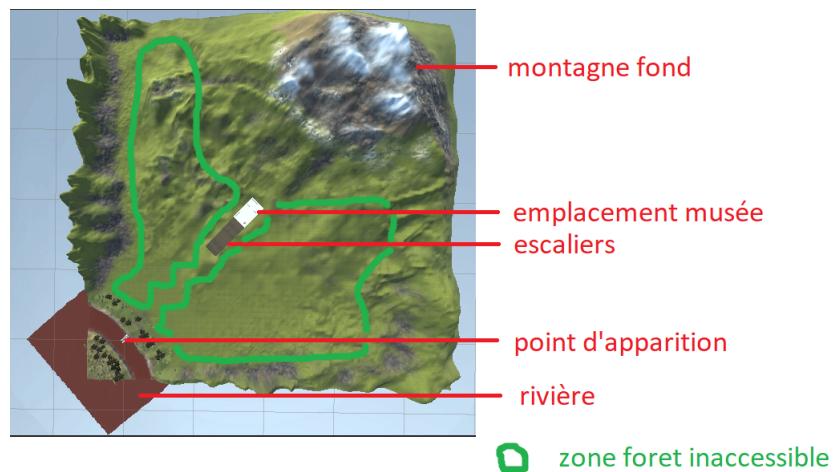


FIGURE 1 – Plan dessiné avant la conception 3D

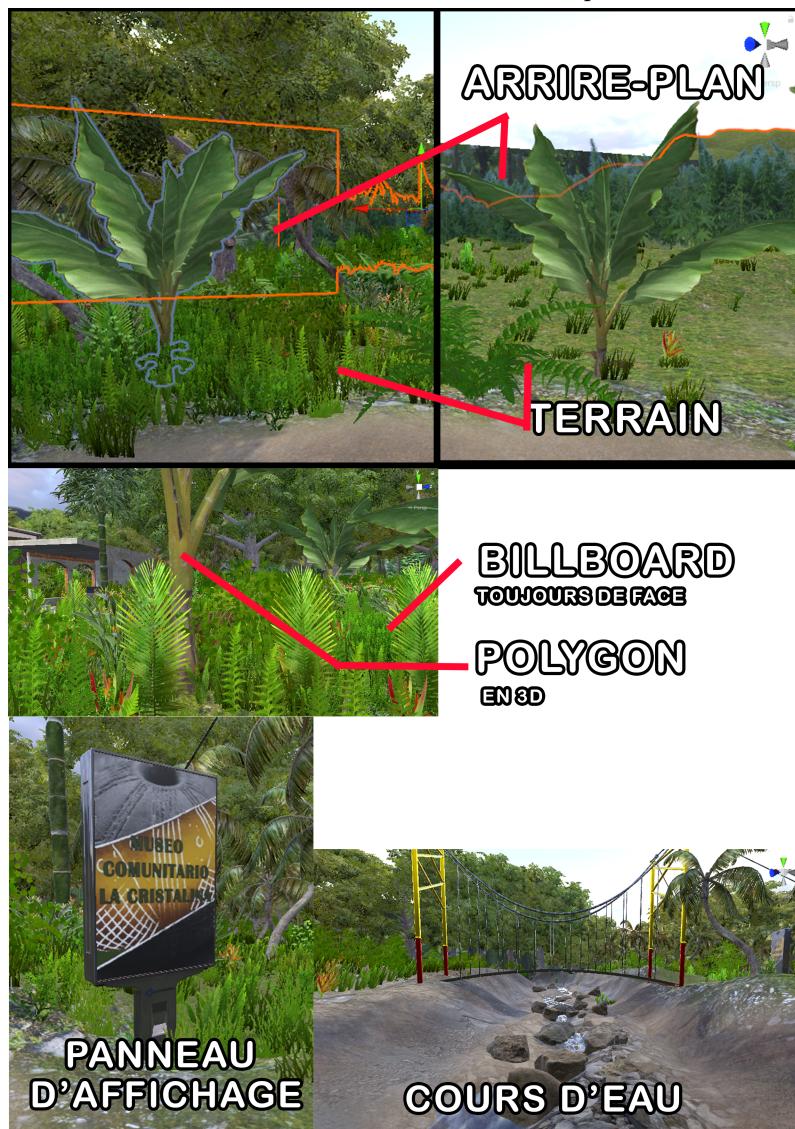


FIGURE 2 – Images des différents éléments de l'environnement

### 3.3 Bâtiments

Comme dit précédemment, tous les bâtiments ont été spécialement conçus sur **SolidWorks**. Nous avons utilisé les photos et vidéos que nous avaient envoyées nos partenaires afin de recréer le plus fidèlement possible ses bâtiments. Mais l'apparence du musée a été modifiée car le vrai est très petit et comme nous n'avons pas de restriction de taille en 3D nous en avons fait un plus grand. Pour les textures elles proviennent soit de **SolidWorks** soit de **TextureHaven**, un site proposant des textures de très haute qualité et gratuite. Cela nous a permis d'avoir un joli rendu sur les bâtiments en utilisant par exemple des briques un peu usées ou un mur de cailloux comme dans le vrai musée. Ces textures comportent en plus ce qu'on appelle une "**normal map**", une couleur permettant de créer un faux relief, ce qui nous donne cet aspect de profondeur et de réalisme sur les textures [Figure 4].

Ensuite nous avons créé des présentoirs, étagères et bibliothèques qui ont par la suite servi pour entreposer les objets dans le musée [Figure 5].

Comme nous n'avons pas reçu les objets à mettre dans le musée à temps, nous avons dû utiliser des modèles 3D d'objets précolombiens ou ressemblant pour meubler le musée et ajouter les interactions dessus.

### 3.4 Interactions

Dans **Unity** les scripts permettent de créer des animations, et dans notre jeu et de passer d'un projet statique à un jeu où l'on peut faire des actions telles que se déplacer ou utiliser des objets.

La majorité des **scripts** ont été réalisés par Clara, qu'il s'agisse des interactions avec les objets ou les menus. Pour ces derniers, il s'agit d'un "**canva**" caché par défaut qui contient les boutons nécessaires (pause, paramètres, quitter) et qui est rendu visible suivant la pression d'une certaine touche, et de même, chaque bouton va charger une scène, en cacher une autre ou baisser certaines valeurs suivant le **script**, par exemple le niveau sonore ou la sensibilité de la souris.

Pour les interactions objets, il a fallu ajouter à chaque objet une **boîte de collision** (**box collider**) qui nous permet de rentrer en contact avec l'objet et si c'est le cas on peut appuyer sur une touche pour qu'il fasse une rotation sur lui-même ou bien qu'on affiche un certain texte. Pour créer le personnage en vue à la première personne nous avons simplement utilisé le **préfabriqué** de la bibliothèque standard de **Unity**.

Pour le panneau d'affichage, le **script** se rend sur internet à une adresse donnée et utilise l'image reçue comme un matériau pour l'objet auquel on a appliqué le **script**, ici le plan contenant l'image, mais comme parfois le projet est hébergé par un service qui bloque cet accès à internet il a fallu modifier le code pour que, dans le cas où on ne parvient pas à se connecter, une image par défaut doit être utilisée.

### 3.5 Hébergement

Afin de partager notre projet au monde nous devions faire des exports pour plusieurs plateformes, **Windows**, **Android** et **WebGL** pour un accès plus universel. Nous avions pour objectif d'utiliser **BabylonJS**, un **plugin** en **JavaScript** permettant d'exporter notre projet en **Web** mais plus léger et donc plus fluide que le **WebGL** de base de **Unity**. Mais après de nombreuses tentatives nous avons compris qu'il nous serait impossible d'utiliser ce **plugin** car il était trop complexe et les documentations assez peu étouffées. Nous avons donc utilisé l'export de base de **Unity** en **Web** que nous avons hébergé sur un site conçu spécialement pour ça : **GamePipe** et pour avoir un rendu plus professionnel, nous avons utilisé une balise **iframe** afin d'afficher notre jeu sur une page web indépendante.

Finalement, nous avons une page web principale qui nous propose soit d'ajouter une nouvelle image pour le panneau soit de jouer. En cliquant sur jouer nous sommes redirigés vers une autre page qui nous permet soit de télécharger le jeu pour différentes plateformes soit de jouer depuis le navigateur en utilisant le **iframe** du jeu fourni par **GamePipe** qui s'adapte à la taille de l'écran et en cliquant sur l'autre bouton on accède à une page où il faut se connecter pour pouvoir téléverser une nouvelle image qui va remplacer l'ancienne.



FIGURE 3 – Image des différentes textures et ornements colombien utilisés

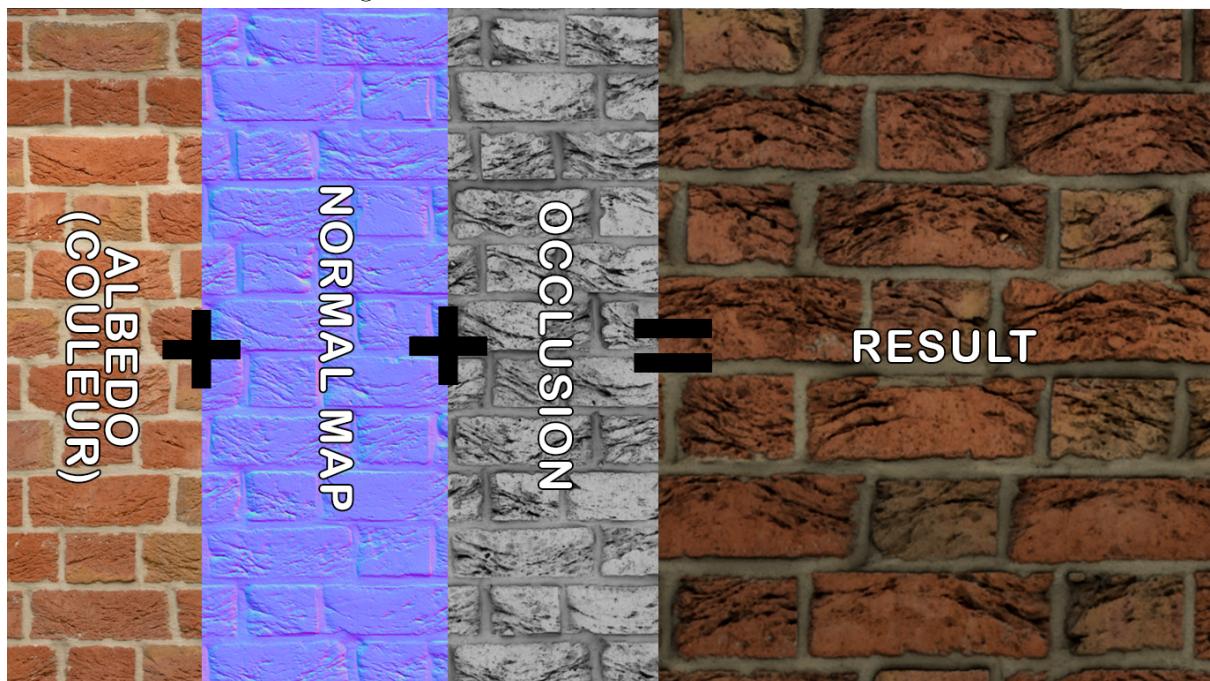


FIGURE 4 – Ajout des différentes textures pour créer un matériau

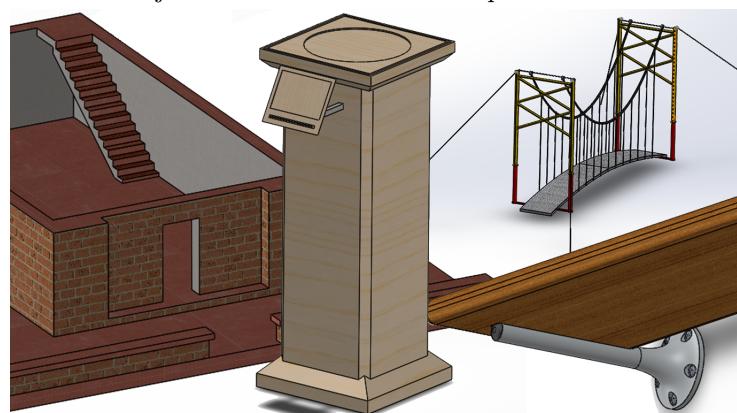


FIGURE 5 – Plusieurs éléments modélisés dans SolidWorks

## 4 Problèmes rencontrés

### 4.1 Collaboration

Nous avons dû faire face à de nombreux problèmes et contre-temps durant le développement de ce projet. Le premier a été de trouver un moyen pour collaborer sur le projet **Unity**. Nous nous sommes penchés premièrement sur la solution proposée par le logiciel lui-même mais dès le début cela n'a pas marché et personnes ne voyaient les modifications que pouvait faire quelqu'un, donc après plusieurs tentatives nous avons laissé tomber cette solution et configuré un **Git** comme alternative. La encore ce n'était pas optimal car si nous étions deux à modifier le projet en même temps lors du **commit** les deux versions allaient se fusionner et corrompre le fichier. Nous avons donc travaillé à des heures différentes mais surtout sur des versions locales du projet que l'on mettait en commun plusieurs fois par jour en mettant tout sur le **Git** et nous communiquions par **Discord** afin d'avertir les autres d'une future modification.

### 4.2 Performances

Les performances étaient un point très important dans le projet car l'utilisation du jeu sur des plateformes à basses performances (téléphone, laptop, navigateur) devait être simple et fluide comme sur des gros ordinateurs. Tout au long du développement nous avons donc utilisé des outils tels que le **profiler** de **Unity** [Figure 6] afin d'analyser les performances et chute de fluidité et les corriger. L'une des causes majeures des pertes de performances est la présence des nombreux **polygones** qui constituent les arbres et plantes qui font la végétation de la forêt tropicale. Pour pallier cela nous avons remplacé bon nombre d'arbres et plantes par des **billboards**, qui ne consomment que très peu et ajouté des arrière-plans qui sont juste des images mais donnent une impression de profondeur et de volume. Une autre cause était les **polygones** utilisés dans les bâtiments et notamment les 600 tuiles du kiosque, pour corriger ça nous avons simplement remplacé les tuiles par une texture ressemblante [Figure 8]. Une fois les performances corrigées, il a fallu les optimiser pour en gagner un maximum. Pour ce faire nous avons utilisé plusieurs technologies présentes dans **Unity**. Premièrement, nous avons fait ce qui s'appelle : de l'**occlusion culling**, cela consiste à analyser la scène pour cacher les éléments qui sont obstrués par d'autres objets, par exemple un objet du musée n'a pas besoin d'être affiché s'il n'est pas visible à la caméra, il a donc fallu créer des zones d'occlusion et marquer les objets qui cachent et les objets cachés [Figure 7]. Une autre technique est le **Light Baking**, c'est en quelque sorte un pré-rendu des réflexions et rebondissements que va avoir la lumière. En préparant ceci, cela évite au processeur d'avoir à le faire en temps réel et on lui épargne donc certains calculs.

### 4.3 Communication

La communication fut un problème de taille durant ce projet. En premier lieu car nous n'avons pu rencontrer qu'une seule fois nos partenaires Colombiens qui n'ont pu donc faire qu'un seul retour sur notre travail. De plus, nous avons attendu plus d'une semaine avant de recevoir des photos du lieu pour mieux nous rendre compte de l'environnement et des constructions ainsi que des mesures et plans du musée pour faire une reconstruction à l'identique. De la même manière, nous étions censés recevoir les objets modélisés en 3D pour les placer dans le musée et écrire certaines informations dessus, mais nous ne les avons pas reçus et avons dû meubler le musée avec des objets typiques traditionnels ou ressemblant trouvés sur internet.

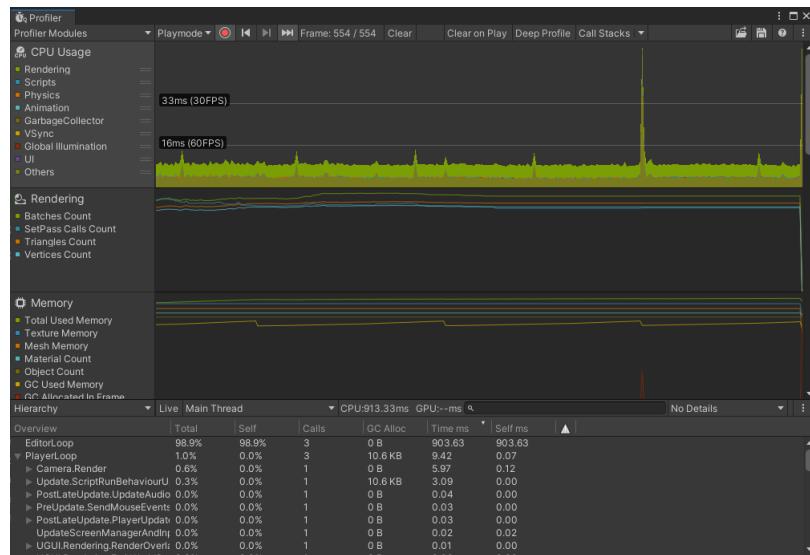


FIGURE 6 – Aperçu du profiler de Unity

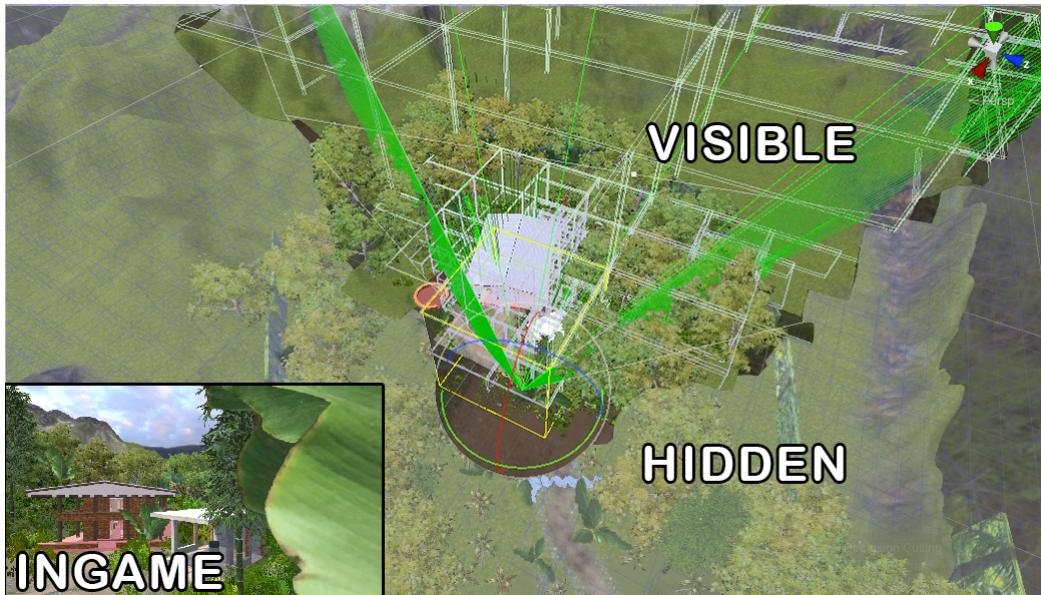


FIGURE 7 – Schéma d’explication du fonctionnement de l’occlusion culling

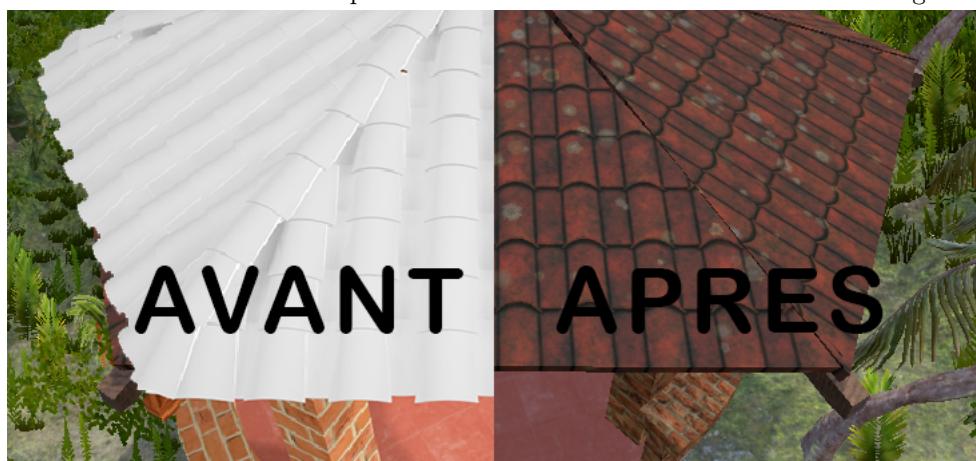


FIGURE 8 – Image montrant les changements fait sur le kiosque

## 4.4 Export

Afin de pouvoir partager notre travail avec le monde il a fallu décider de comment nous allions le faire, sous quel format et sur quelle plateforme. Le but premier était de le partager sur un maximum de systèmes, **Windows**, **Android**, **Web**, **VR**, **Mac...** Mais nous nous sommes vite rendu compte que certains supports allaient devoir être oubliés, en effet, le projet est trop lourd pour pouvoir être exécuté sur un téléphone **Android**, nous nous sommes donc ravisé et avons travaillé pour faire des exports en **WebGL**, ce qui permet à un maximum de personnes d'accéder au jeu, qu'importe le système d'exploitation, **Windows** ou **Linux**. L'une des ressources qui devait être utilisée pour l'export est **BabylonJS**, un plugin pour **Unity** qui permet de compiler notre projet en format web et avec des **scripts** en **JavaScript** à la place du **C#** que nous avait recommandé notre tutrice. Pendant plusieurs semaines j'ai tenté de faire fonctionner ce plugin mais ne l'ayant jamais utilisé et même en lisant la documentation et des tutoriels je n'ai jamais réussi à faire une compilation correcte ou viable. **BabylonJS** est un plugin assez récent mais d'une version à l'autre tous les boutons et réglages changent de place et les tutos sur internet ne valent plus rien. De plus, je me suis rendu compte que **Babylon** n'utilisait pas les **scripts** en **C#** pour les recompiler en **JS** mais utilisait du **TypeScript**, un dérivé du **JS**, hors il était un peu tard pour tout refaire et il aurait même fallu refaire les **scripts** par défaut de **Unity** pour l'eau ou les mouvements du personnage. Nous avons donc dû abandonner l'idée d'utiliser **BabylonJS** qui aurait pu nous permettre de faire des exports pour le web très légers et fluides comparé à ceux natifs à **Unity**. Pour héberger le jeu et permettre à tout le monde d'y jouer sans avoir à télécharger quoi que ce soit nous avons utilisé **GamePipe**, un **WaaS** (**website as a service**) qui permet justement d'exécuter des projets **WebGL** de **Unity**, le tout totalement gratuitement [Figure 9].

Il a fallu également créer et héberger la page web permettant de changer l'image présente sur le panneau d'affichage. Pour ce faire j'ai créé une petite page en **PHP** qui permet l'import d'un fichier au format **.jpg** et qui juste avant l'upload va renommer le fichier en un nom prédéfini pour remplacer l'ancien. Dans **Unity** on fournit directement l'adresse de l'image qui ne change jamais puisque c'est le fichier qui change.

Pour stocker et centraliser tout ça nous avons fait un petit dossier sur un domaine **Replit** qui contient toutes les pages web.

## 4.5 VR

Un autre souci majeur est que le projet devait initialement pouvoir être joué en vue "normale" mais également en vue **stéréoscopique** avec un casque de réalité virtuelle. Mais aucun de nous trois ne possède de casque de ce type et nous n'avons pas pu en emprunter au labo pour nos tests. Mais le jeu a été conçu pour détecter si un casque est branché ou non et la vue va s'adapter automatiquement et les manettes fournies avec le casque vont simuler le clavier et la souris afin d'interagir avec l'environnement. Avec un peu de temps en plus et le matériel adéquat je pense que nous aurions été capable de pallier ce problème et optimiser le changement d'une vue à l'autre.

## 4.6 Encapsulation de l'export

En ajoutant le panneau d'affichage au jeu nous sommes tombés face à un autre problème auquel nous n'avions pas pensé. En effet, le but du panneau était de pouvoir afficher une image qui peut être remplacée en ajoutant une nouvelle depuis une page web. Il fallait donc à un moment ou un autre que le projet puisse accéder à internet pour récupérer l'image. Or les sites comme **GamePipe** ne permettent pas un accès à internet au projet, les fichiers sont encapsulés et inaccessibles de l'extérieur. Nous avons donc dû modifier le **script** pour qu'en cas d'échec de la connexion on charge une image par défaut sur le panneau, mais cela ne règle pas réellement le problème puisque dans ce cas là, l'image restera la même [Figure 10].

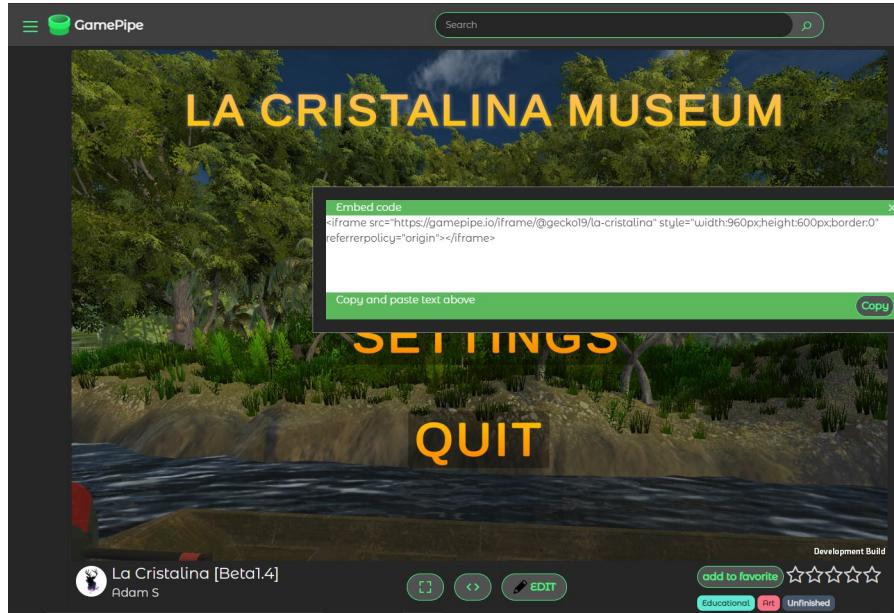


FIGURE 9 – Aperçu de la plateforme GamePipe



FIGURE 10 – Image montrant que l'image n'est pas chargée sur GamePipe

## 5 Nettoyage d'objets 3D

Une problématique nous a été rajoutée lors de l'une de nos réunions, il s'agissait de trouver un moyen pour nettoyer des objets 3D préalablement conçus par photogrammétrie. En effet, lorsqu'un objet est pris en photo pour une modélisation 3D, il n'est pas rare que certaines impuretés se rajoutent à l'objet. Il peut s'agir d'artefacts visuels en plus qui n'ont pas lieu d'être ou bien des trous dans l'objet par manque de photos.

Afin de régler ces petits détails et de retoucher l'objet, j'ai trouvé un logiciel répondant tout à fait à cette problématique, il s'agit d'Autodesk MeshMixer, un logiciel gratuit permettant la retouche 3D. Et en utilisant les outils incorporés, on arrive tout de suite à un bon résultat [Figure 11].

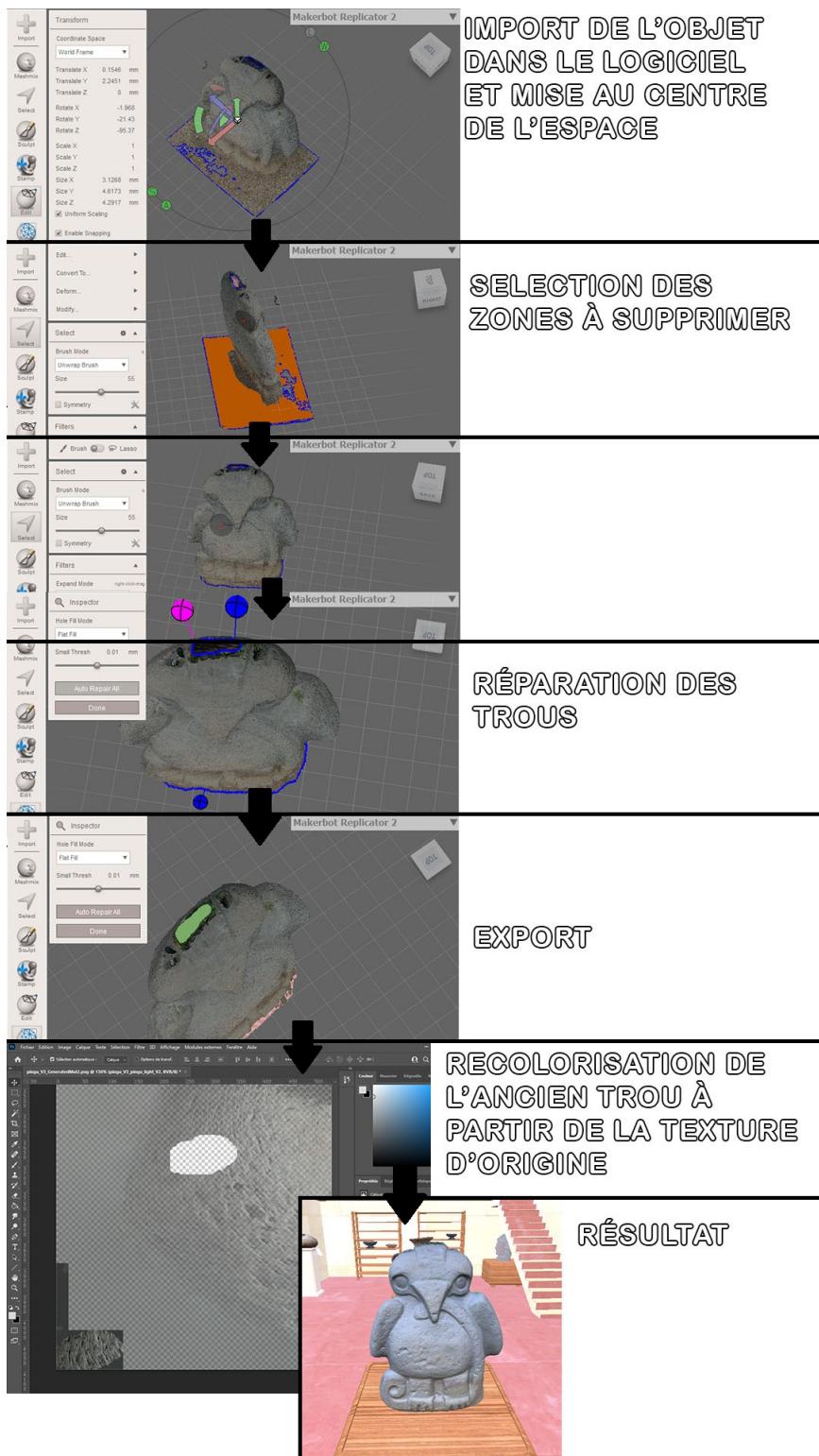


FIGURE 11 – Etapes de nettoyage d'un objet 3D

## 6 Conclusion

### 6.1 Apports personnels et perspectives d'avenir

Pour conclure, ce mois de stage a été très intéressant sur plusieurs points. Tout d'abord nous avons appris à manier plusieurs logiciels et langages auxquels nous n'avions jamais touché auparavant et nous avons eu accès à de nouveaux domaines de l'informatique, la conception de jeux et environnements 3D ainsi que la VR.

Nous avons dû tout apprendre en commençant de zéro, et en divisant le projet en plusieurs tâches et étapes, nous avons réussi à atteindre un résultat plus que satisfaisant.

Cela nous a également permis d'apprendre à mieux travailler en groupe et collaborer sur un même projet assez conséquent, avec un Git par exemple et en communiquant bien l'avancée de chacun sur une certaine tâche et en faisant des rapports hebdomadaires pour mieux se rendre compte de la situation, les problèmes à corriger et les prochaines tâches à mener.

Le fait de travailler sur un projet avec des partenaires (voire des clients) qui plus est à un niveau international, est une opportunité très intéressante car on sait qu'en fin de compte ce projet servira vraiment et possède une réelle finalité pour le grand public et cela nous permet aussi d'avoir des retours sur notre travail et de nous préparer à nos futurs projets professionnels.

Nous avons également appris à gérer des situations de "crises" où il faut recommencer certaines choses ou en ajouter parce que le partenaire n'est pas satisfait ou bien qu'il veut rajouter des choses au projet qui n'étaient pas prévues initialement et que nous n'aurions pas le temps de faire.

D'un point de vue plus personnel, ce stage m'a permis d'en apprendre plus sur la création 3D en informatique, domaine dans lequel je ne connaissais rien, et j'ai beaucoup aimé faire l'environnement extérieur, apprendre les différents types d'objets, de texture, les techniques d'optimisation, mais malgré tout, je suis conforté dans mon idée que le développement de jeu vidéo n'est pas ce que j'ai envie de faire plus tard comme métier et dans un futur plus proche, que je ne serai pas à ma place dans un master tel que IMAGINA à la Fac des Sciences mais plutôt en AIGLE en tant que développeur logiciel.

### 6.2 Résultats obtenus

Je retiens une très bonne expérience de ce stage et j'aurais apprécié qu'il dure un peu plus longtemps, nous aurions pu fignoler certains détails, car un mois était un peu court pour prendre en main Unity et développer un environnement dont les partenaires n'auraient rien eu à redire. Mais malgré tout je pense que nous avons réussi à remplir le cahier des charges et à proposer un produit tout à fait satisfaisant, qui répond à la problématique et qui est fonctionnel, stable et avec toute une plateforme fonctionnelle autour pour le grand public (site web). A la fin de ce mois de travail nous sommes en capacité de présenter un environnement complet reprenant les caractéristiques et la végétation de la Colombie avec les bâtiments demandés et des objets avec lesquels nous pouvons interagir, et ce depuis plusieurs plateformes.

### 6.3 Sources

Logo Faculté des Sciences : [sciences.edu.umontpellier.fr](http://sciences.edu.umontpellier.fr)

Logo Université Montpellier : [umontpellier.fr](http://umontpellier.fr)

Logo LIRMM : [lirmm.fr](http://lirmm.fr)

Logo Réseau Figure : [reseau-figure.fr](http://reseau-figure.fr)

Liste des sites où nous avons trouvé les modèles 3D utilisés :

- Sketchfab : [sketchfab.com](http://sketchfab.com)
- Turbosquid : [turbosquid.com](http://turbosquid.com)
- CGTrader : [cgtrader.com](http://cgtrader.com)
- Free3D : [free3d.com](http://free3d.com)
- Unity Asset Store : [assetstore.unity.com](http://assetstore.unity.com)

Site des textures utilisées :

TextureHaven : [texturehaven.com](http://texturehaven.com)

Site pour la Skybox :

HDRIsHaven : [hdrihaven.com](http://hdrihaven.com)