

RAPPORT DE PROJET EI4-SAGI

Château du Plessis Macé - Techniques
de réalité augmentée au service de
l'histoire et du patrimoine

Projet réalisé par :

Adrien Gagnon

Vincent Gruel

Doriane Jaunault

Encadrant :

Paul Richard

Remerciements

Nous souhaitons remercier tout particulièrement Mme Marie Boutteaud pour sa disponibilité et les concepts qu'elle a pu nous proposer tout au long de ce projet.

Nous remercions également l'intégralité du personnel du château pour leur sympathie lors de notre visite du château, mais aussi pour les précieuses informations historiques qu'ils nous ont données et qui nous ont été très utiles notamment lors du développement de la visite en réalité augmentée.

Nous remercions aussi Mr Paul Richard, notre enseignant référent, pour sa disponibilité et les conseils avisés qu'il nous a apportés lorsque nous étions dans le besoin.

Pour terminer, nous souhaitons remercier l'ISTIA qui nous a offert l'opportunité de travailler sur ce projet.

Table des matières

Table des matières	2
1. Introduction	3
2. Répartition des tâches.....	4
3. Présentation de Vuforia	5
4. Gestion de la base de données.....	5
4.1. Ajout d'une image	6
4.2. Reconnaissance de modèle 3D	7
4.3. Tutoriel.....	8
5. Présentation du projet	10
5.1. La cuisine	10
5.1.1. Optimisation.....	11
5.1.2. Rajout de nouvelles fonctionnalités	122
5.1.3. Le rendu graphique	133
5.2. Visite libre en réalité augmentée	155
5.2.1. Mise en place de la scène	166
5.2.2. Composition et fonctionnement des objets	177
5.2.3. Test tracking 3D	188
5.2.4. Écriture des scripts	200
5.3. Les perspectives d'avenir	233
6. Conclusion.....	255
7. Table des figures	266
8. Bibliographie	27

1. Introduction

Dans le cadre d'un partenariat entre l'Istia et le château du Plessis Macé, un projet a été débuté l'année dernière par 2 étudiants de notre formation dans le cadre de leur stage de fin d'année.

Dans le cadre de notre projet de 4^e année et dans l'optique d'acquérir de nouvelles connaissances en réalité augmentée, nous avons choisi de reprendre ce projet.

Le but de ce dernier est la mise en place de diverses applications de réalité augmentée qui pourront être utilisées par les visiteurs. Après concertation avec le personnel du château, les applications suivantes ont été définies :

- Cuisine en réalité augmentée (projet débuté par Alex l'année passée)
- Simulation de joute (projet réalisé par Arthur l'année passée)
- Chasse au trésor (chemin de ronde)
- Visite en réalité augmentée (uniquement pour les parties accessibles en visite libre)
- Plan du donjon en réalité augmentée (pas de plans disponibles)

Dans un premier temps et pour avoir une meilleure compréhension de ce qui avait été fait, nous avons récupéré les projets déjà existants et nous les avons soigneusement analysés.

Pour pouvoir comprendre le fonctionnement des options de réalité augmentée, nous avons décidé de suivre une phase importante de formation grâce à différents tutoriels et en naviguant sur différents forums qui nous ont bien souvent permis de surmonter les difficultés que nous avons rencontrées.

Au travers de ce rapport, nous allons développer les différentes activités que nous avons réalisées tout au long de ce projet. Nous parlerons également des différents problèmes que nous avons rencontrés tout en détaillant les solutions que nous avons apportées pour les résoudre.

Dans un premier temps nous nous concentrerons sur la répartition des tâches au sein de notre groupe. Dans un second temps nous ferons un point sur le logiciel Vuforia, ses fonctionnalités et la gestion des données avant de nous focaliser sur les différents projets. Enfin nous terminerons en évoquant les perspectives d'évolutions du projet et en proposant ce qu'il serait possible de développer dans le cadre d'un éventuel stage.

2. Répartition des tâches

Dans un premier temps nous allons aborder la répartition des différentes tâches au sein de notre groupe à travers un diagramme de Gantt, et la présentation d'un diagramme circulaire représentant la part de chacune des activités réalisées tout au long du projet.

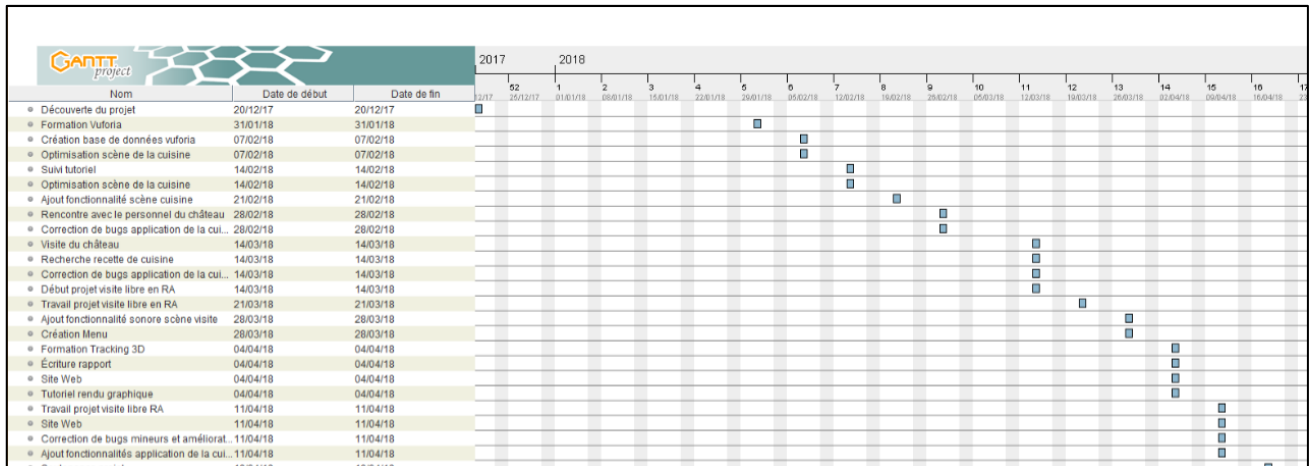


Figure 1 : Diagramme de Gantt

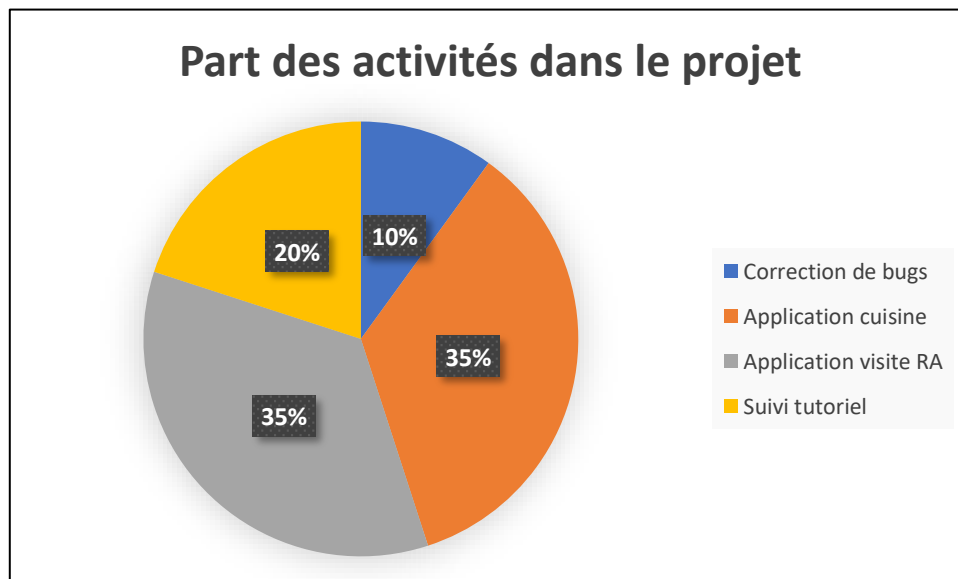


Figure 2 : Part de chaque activité dans le projet

3. Présentation de Vuforia

Avant de poursuivre la programmation du projet, nous avons choisi de nous former sur l'utilisation de Vuforia et de ses nombreuses fonctionnalités. Dans un premier temps nous avons récupéré les projets d'Alex et Arthur et avons soigneusement analysé ces derniers, avant de suivre un tutoriel qui nous a permis d'acquérir les bases nécessaires à la poursuite du projet.

Vuforia a été récemment ajouté comme option lors de l'installation de base de Unity, nous avons donc été confrontés à des soucis de cohérence entre les différentes versions.

En effet, bien que les projets réalisés par Alex et Arthur disposent d'un package incluant les fonctionnalités de Vuforia, ce dernier provoque un conflit lorsqu'on l'installe directement via Unity.

3.1. Gestion de la base de données

La gestion de la base de données représente l'une des parties les plus importantes de ce projet. Ce sont les images que l'on ajoute à la base qui nous permettent par la suite d'interagir avec les objets du monde réel sur Unity.

Afin de pouvoir utiliser les fonctionnalités de réalité augmentée, il est indispensable de disposer d'une licence pour laquelle on fait la demande directement sur le site de Vuforia.

Pour plus de simplicité nous avons choisi de refaire une demande de licence puis nous avons modifié les projets d'Alex et Arthur pour y assigner la nouvelle base de données.

Pour pouvoir utiliser la base de données sous Unity il suffit de rentrer la clé d'accès à la licence dans les options de configuration de Vuforia.

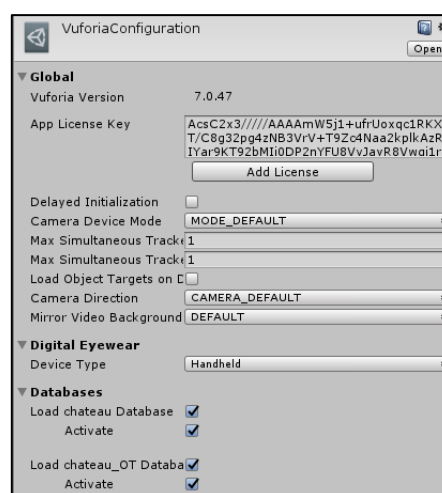


Figure 3 : Configuration Vuforia

Il suffit ensuite de cocher les bases de données que l'on souhaite utiliser et de les activer pour avoir accès à l'intégralité des images contenues dans ces dernières.

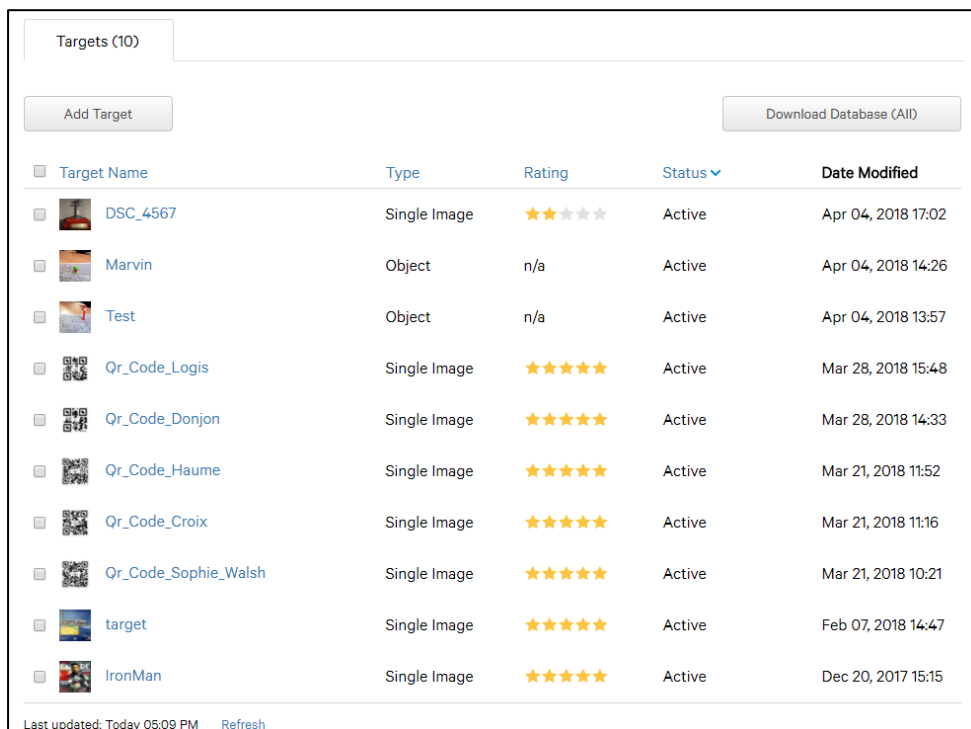
Notre projet comporte deux versions de la base de données :

- Une première contenant les images utilisées pour notre projet de visite libre en réalité augmenté.
- Une seconde qui contient les différents modèles 3D que nous avons essayé d'utiliser en fin de projet.











3.2. Ajout d'une image

Pour ajouter une image sur la base de données, il suffit de se rendre sur le site de Vuforia et de se connecter avec les identifiants préalablement choisis lors de l'inscription. Une fois la connexion effectuée, il est possible d'accéder à l'intégralité de la base de données mais seuls certains formats sont reconnus sur Vuforia (jpeg, png) et il est donc parfois nécessaire de redimensionner ou de retravailler ces images pour qu'elles puissent correspondre aux contraintes du site.

Les contraintes de Vuforia dans la reconnaissance d'images nous ont souvent posé des problèmes au cours du projet. En effet certaines des images qui nous ont été fournies n'ont pas pu être utilisées (ou tout du moins pas de la façon dont nous le souhaitions), car afin d'être reconnues par le logiciel les images doivent présenter un nombre suffisant de points de référence qui sont calculés lors de l'ajout de l'image.



The screenshot shows the Vuforia database interface with a table of targets. The table has columns for Target Name, Type, Rating, Status, and Date Modified. There are 10 targets listed, including DSC_4567, Marvin, Test, and several QR codes. The interface also includes an 'Add Target' button and a 'Download Database (All)' button.

<input type="checkbox"/>	Target Name	Type	Rating	Status	Date Modified
<input type="checkbox"/>	 DSC_4567	Single Image	★☆☆☆☆	Active	Apr 04, 2018 17:02
<input type="checkbox"/>	 Marvin	Object	n/a	Active	Apr 04, 2018 14:26
<input type="checkbox"/>	 Test	Object	n/a	Active	Apr 04, 2018 13:57
<input type="checkbox"/>	 Qr_Code_Logis	Single Image	★★★★★	Active	Mar 28, 2018 15:48
<input type="checkbox"/>	 Qr_Code_Donjon	Single Image	★★★★★	Active	Mar 28, 2018 14:33
<input type="checkbox"/>	 Qr_Code_Haume	Single Image	★★★★★	Active	Mar 21, 2018 11:52
<input type="checkbox"/>	 Qr_Code_Croix	Single Image	★★★★★	Active	Mar 21, 2018 11:16
<input type="checkbox"/>	 Qr_Code_Sophie_Walsh	Single Image	★★★★★	Active	Mar 21, 2018 10:21
<input type="checkbox"/>	 target	Single Image	★★★★★	Active	Feb 07, 2018 14:47
<input type="checkbox"/>	 IronMan	Single Image	★★★★★	Active	Dec 20, 2017 15:15

Last updated: Today 05:09 PM Refresh

Figure 4 : Base de données Vuforia

Une image ne présentant pas suffisamment de points ne pourra pas être détectée lors de l'utilisation de l'application. Pour pallier à ce problème, nous avons proposé au personnel du château d'utiliser des QR Code personnalisés.



Figure 5 : QR Code + points de repère

Ces QR Code ont été générés à l'aide d'un site internet, grâce aux différents textes que nous leur avons assignés. Pour que ces derniers soient propres au château, nous avons pris l'initiative d'y ajouter le logo du château.

Les QR Code, présentant de nombreux points de référence, obtiennent la note maximale lorsqu'on les ajoute sur la base de données.

3.3. Reconnaissance de modèle 3D

Dans le cadre du projet de visite libre en réalité augmentée, nous avons envisagé d'utiliser l'une des fonctionnalités de Vuforia qui permet de reconnaître des objets 3D, préalablement scannés à l'aide d'une application.

L'utilisation de cette application est indispensable si l'on souhaite utiliser des modèles 3D comme cible. En effet, seuls les fichiers générés par cette application peuvent être ajoutés à notre base de données. Le scanner Vuforia est très simple d'utilisation mais il n'est possible de scanner que des objets de petite taille. En effet, pour pouvoir scanner correctement un objet il est nécessaire d'utiliser une feuille (format A4) mise à disposition sur le site de Vuforia. Sans cette dernière il est impossible de générer les objets 3D.

La création d'un modèle 3D passe par la reconnaissance de différents points sur le modèle, plus le nombre de points est important plus il sera simple de reconnaître l'objet lors d'une application réelle.

Tous les objets scannés apparaissent sur le menu de l'application, il suffit ensuite d'exporter ces données pour les récupérer sur un ordinateur.

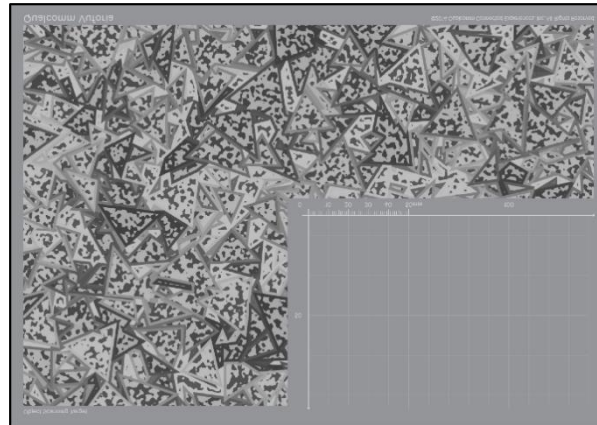


Figure 6 : Object Scanning Target

Une fois l'exportation sur ordinateur terminée, il faut ajouter ce modèle comme nous le ferions pour une simple image. C'est cette action qui entraîne la création d'une seconde base de données propre aux différents modèles 3D.

Tout comme pour les images il est nécessaire de procéder à l'activation de cette base pour pouvoir accéder aux modèles sous Unity.

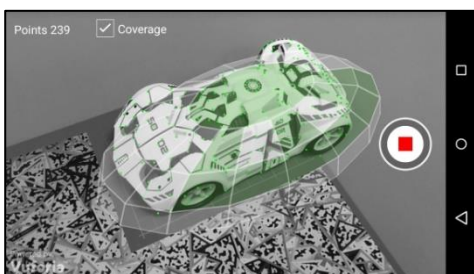


Figure 8 : Scanner Vuforia

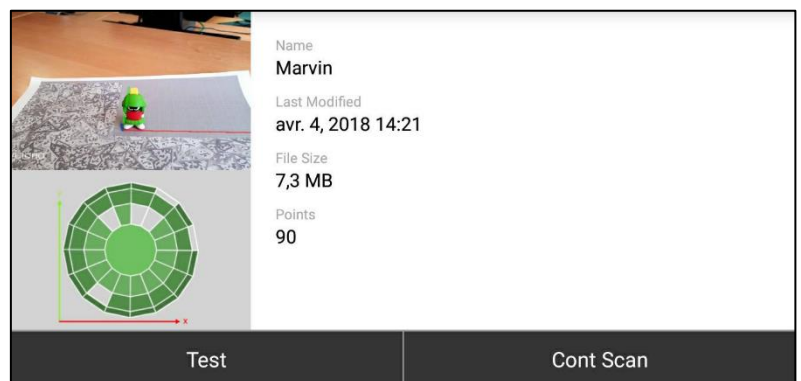


Figure 7 : Modèle 3D

3.4. Tutoriel

Pour comprendre pleinement le fonctionnement des programmes préalablement réalisés, nous avons jugé indispensable de suivre un premier tutoriel sous Vuforia. Celui-ci nous a permis d'acquérir les bases nécessaires au bon déroulement de notre projet.

Ce tutoriel très simple nous a appris dans un premier temps à déclarer une image cible sur Unity, puis à gérer la détection de cette dernière pour pouvoir interagir et y faire apparaître les éléments de notre choix.

Pour mener à bien ce tutoriel, nous avons donc téléchargé un asset proposant un personnage à mettre en scène. Nous avons choisi une plaquette de l'Istia comme élément cible, cette dernière possédant de nombreux points de repère.



Figure 9 : Image cible projet tutoriel

Le projet finalement créé fonctionne de la façon suivante : lorsque l'on détecte l'image via la webcam de l'ordinateur ou encore la caméra d'un smartphone, le personnage de la scène jusqu'ici caché apparaît à l'écran. Il aurait bien évidemment été possible d'écrire un script pour lui permettre de se déplacer, mais ne souhaitant pas gaspiller du temps inutilement pour une application qui ne faisait pas partie du projet nous avons choisi de nous arrêter à cette étape.

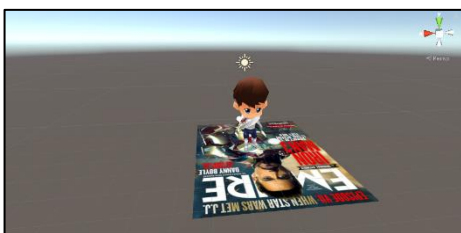


Figure 10 : Scène tutoriel

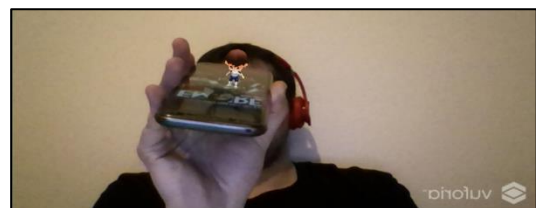


Figure 10 : Chevalier en réalité augmenté

Les mouvements actuels de l'asset présent dans la scène sont gérés à l'aide d'un Animator pré-régulé dans le Prefab.

4. Présentation du projet

Comme évoqué en introduction nous nous sommes répartis les différentes parties du projet et nous avons choisi de laisser de côté la partie joute qui avait été terminée par Arthur lors de son stage.

Il nous a hélas été impossible de travailler sur la partie du donjon car cela fait déjà un certain temps que le personnel du château est à la recherche de plans qui permettraient d'avancer significativement sur cette partie du projet.

Suite à un accord commun il a été décidé que Doriane travaillerait à l'amélioration et l'aboutissement du projet « Cuisine du logis », Adrien travaillerait à la correction des éventuels bugs sur le projet de joute et Vincent travaillerait à la réalisation du projet de visite libre en réalité augmenté évoqué lors de la visite.

Nous sommes cependant restés disponibles pour apporter nos conseils et nos connaissances sur les autres projets.

4.1. *La cuisine*

Lorsque nous avons repris le projet, il disposait d'un écran d'introduction, d'un bouton permettant la navigation vers un menu, et de la scène de la cuisine. Celle-ci, s'enclenchait après l'appui sur le bouton « C'est parti ! ». En premier lieu, l'utilisateur doit chercher l'image cible qui permet au jeu de se lancer. Une fois celle-ci trouvée, un chaudron apparaît ainsi que trois parchemins. En cliquant sur le nom d'un légume sur l'onglet « Ingrédients », celui-ci apparaît sur l'écran mais également dans l'onglet « Recette ». Tout au long du jeu, un personnage situé en bas à droite de l'écran encourage le joueur à trouver l'image cible ou à sélectionner des ingrédients.

4.1.1. Optimisation

Lorsque nous avons récupéré la scène de la cuisine, celle-ci permettait de pouvoir choisir entre deux ingrédients, d'afficher celui sélectionné et de stocker son image dans un tableau « Recette » à une place prédéfinie.



Figure 11 : Scène initiale de la cuisine

Le script utilisé, « TargetDisp » pour gérer les actions décrites précédemment, n'était pas optimisé pour gérer plus de deux ingrédients. En effet, chaque ingrédient était initialisé individuellement (le bouton associé et le GameObject) et possédait une fonction « Press » unique. Celle-ci se déclenchait après l'appui sur le nom de l'ingrédient sélectionné.

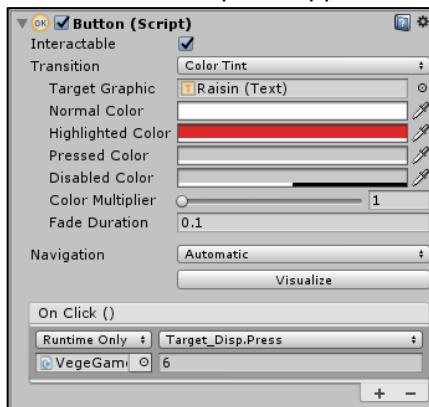


Figure 12 : Nouveau bouton

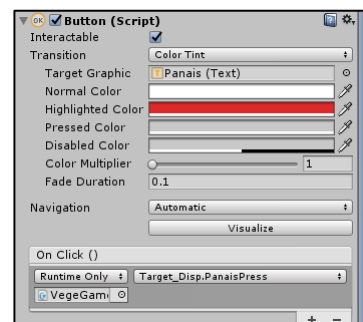


Figure 13 : Ancien bouton

Nous avons trouvé judicieux de créer plusieurs tableaux contenant les différents paramètres d'entrées. Cela permet de ne pas appeler individuellement chaque paramètre pour les modifier mais de parcourir un tableau et modifier la variable courante. Le code est ainsi beaucoup plus clair et peu importe le nombre d'ingrédients passés en paramètres. Cela permet aussi de ne faire qu'une seule fonction « Press » qui reçoit en entrée une variable « place » donnant l'index de l'ingrédient sélectionné.

Dans la scène de base, lorsqu'un ingrédient était sélectionné, la place qu'il prenait dans l'onglet « Recette » était prédéfinie dans les fonctions « Press » associées. Ainsi, même si on sélectionnait un ingrédient en premier, on pouvait le retrouver à la deuxième place dans l'onglet « Recette ». Afin de pallier à ce problème, nous avons créé une variable « k » qui est initialisée à la valeur 0 au lancement de la scène puis qui s'incrémente de 1 à chaque ingrédient choisi. Lorsque cette valeur a atteint la valeur maximale de places disponibles pour les ingrédients, un message s'affiche, avertissant le joueur.



Figure 14 : Gestion des ingrédients dans l'onglet "Recette"

4.1.2. Rajout de nouvelles fonctionnalités

Afin que le joueur puisse créer une recette en utilisant les ingrédients mis à disposition, nous avons créé une classe « Recette ». Celle-ci contient un « string » correspondant à son nom et un tableau de booléens de la taille du nombre d'ingrédients qui correspond à la présence ou non de ceux-ci. Les recettes initiales sont stockées dans un objet de type List « LivreCuisine » et initialisé au lancement de la scène. Une recette « EnCours » est également créée au lancement de la scène avec un nom vide et un tableau de booléens initialisé à « false ».

```

152
153
154
155
156
157
158
159
160
161
162
163
164
165
class Recette
{
    public string _nom;
    public bool[] _ingredients;

    public Recette(string nom, bool[] ing)
    {
        _nom = nom;
        _ingredients = ing;
    }
}

```

Figure 15 : Classe "Recette"

A chaque fois que le joueur sélectionne un ingrédient, celui-ci est affiché et ajouté dans l'onglet « Recette ». Dans l'objet « EnCours », la variable booléenne correspondant à l'index de l'ingrédient est mise à « true » afin de conserver en mémoire le choix du joueur. On parcourt ensuite « LivreCuisine » et on regarde si l'ingrédient sélectionné est présent. Si ce n'est pas le cas, la recette actuelle est supprimée de la liste. Cela nous permet de ne garder que les recettes potentiellement réalisables.

Afin que le joueur puisse vérifier si les ingrédients qu'il a sélectionnés correspondent à une recette, il était nécessaire de rajouter un bouton. Dans le but de ne pas surcharger l'écran avec un nouvel élément, nous avons donc décidé que le chaudron servirait de bouton. Nous avons voulu dans un premier temps rajouter un objet de type Button au chaudron, cependant ceci n'est pas possible sur Unity. Nos recherches nous ont orientés vers l'utilisation d'un objet de type Raycast qui s'apparente à un rayon laser qui renvoie le premier GameObject qu'il a touché. Dans la fonction « Update », nous avons donc testé si l'objet sur lequel cliquait l'utilisateur était le chaudron. Si c'est le cas, alors on lance la fonction « VérificationRecette ».

La fonction « VérificationRecette » permet de vérifier si les ingrédients sélectionnés par le joueur correspondent à une recette du Moyen-Age. Pour cela, on compare chaque recette présente dans « LivreCuisine » avec « EnCours ». Si une comparaison est trouvée, alors on affiche le nom de la recette créée tout en félicitant le joueur. Si « LivreCuisine » ne contient plus de recettes ou si « EnCours » ne correspond à aucune recette, on affiche un message comme quoi le plat est raté en encourageant le joueur à recommencer. Pour relancer le jeu, il suffit de cliquer sur le chaudron.



Figure 17 : Cas où la recette est réussie

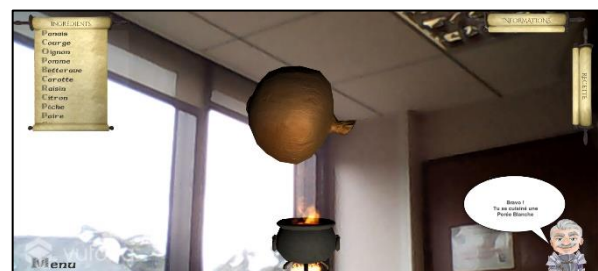


Figure 16 : Cas où la recette est ratée

4.1.3. Le rendu graphique

Une fois ces étapes terminées et vérifiées, nous avons dû rajouter plusieurs ingrédients afin de tester le programme dans sa généralité. La première difficulté a été de les insérer correctement. En effet, il y a plusieurs étapes à respecter. Tout d'abord, il faut ajouter le GameObject de l'ingrédient, en lui mettant les bonnes coordonnées et en lui ajoutant le script « Rotate ». Ensuite, il faut lui assigner un objet enfant, une image au bon format. Puis, il faut créer un bouton dans le volet « Ingrédients »

4.2. Visite libre en réalité augmentée

Initialement pensé comme un projet de chasse au trésor se déroulant sur le parcours du chemin de ronde, ce projet s'est transformé en une visite libre en réalité augmentée.

Le manque d'assets à notre disposition et les contraintes liées à l'environnement nous ont obligés, en accord avec le personnel du château, à mettre de côté ce projet pour le moment. Il s'avère que le parcours choisi n'était en réalité composé que de 2 tours et d'un couloir et outre le manque de place pour dissimuler les objets c'est avant tout le manque de luminosité qui nous empêche actuellement de donner vie à ce concept.

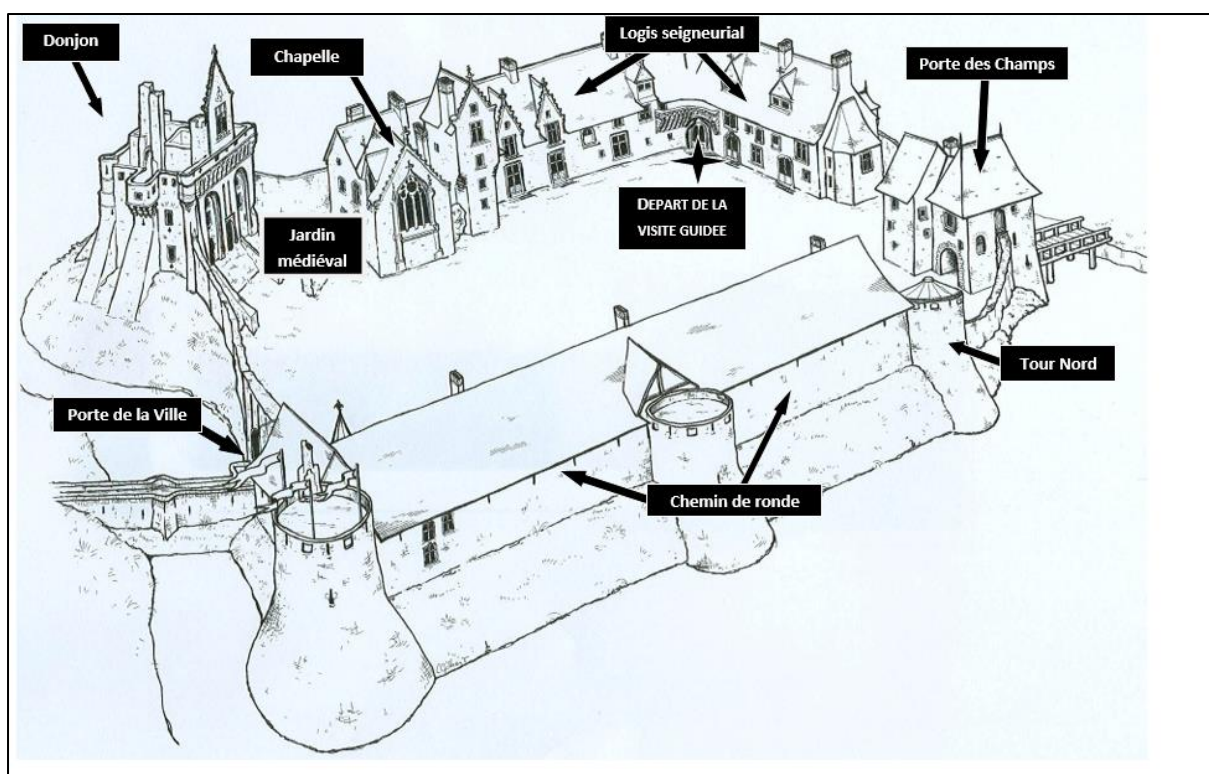


Figure 20 : Plan visite libre

Après discussion avec Mme Boutteaud, nous avons convenu qu'il serait compliqué d'étendre le projet de visite en réalité augmentée à l'intégralité du château puisque l'accès aux parties intérieures telles que le logis, la cuisine et les chambres est strictement limité au cadre des visites guidées. Une extension de l'application n'apporterait pas réellement plus d'informations aux visiteurs cependant il serait possible d'envisager l'utilisation de certains objets à des fins plus ludiques, ce que nous développerons lorsque nous aborderons les perspectives d'avenir de ce projet.

Les différentes activités qui ont été réalisées au cours de ce mini projet sont les suivantes : Création de la scène, écriture des scripts, ajout d'un menu.

4.2.1. Mise en place de la scène

Dans un premier temps nous avons travaillé à la mise en place de la scène et de ces différents composants.

On y retrouve plusieurs objets ainsi qu'une caméra propre aux fonctionnalités de Vuforia. L'utilisation de cette dernière nécessite la suppression de la caméra ajoutée par défaut lors de la création d'un nouveau projet et c'est cette nouvelle caméra qui est reliée à notre webcam qui va détecter les images cibles présentées par les visiteurs.

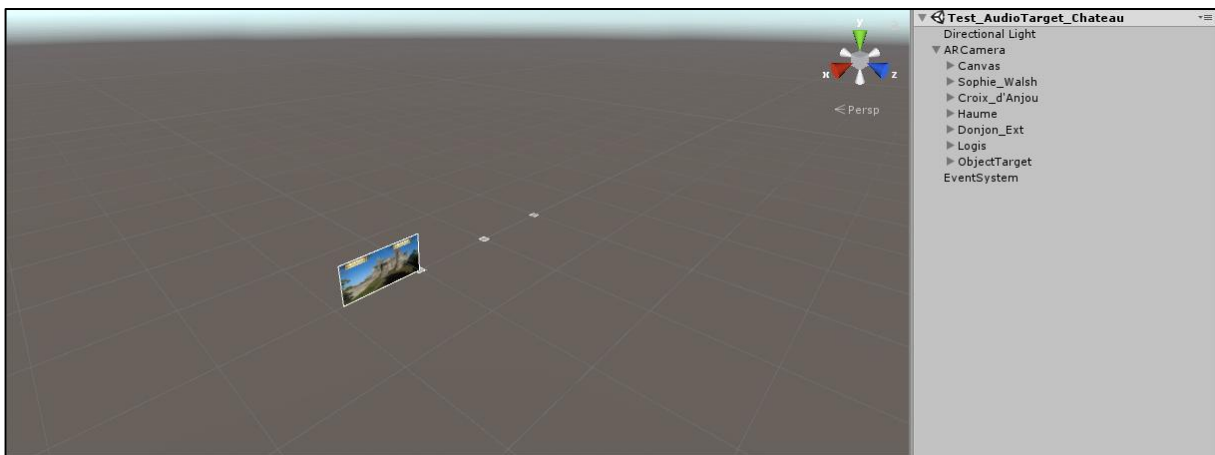


Figure 21 : Scène actuelle

Toutes les images sont positionnées au même endroit sur la scène, en fonctionnement cela ne pose aucun problème puisqu'il est impossible d'avoir plusieurs images actives simultanément.

Cette scène reste relativement simple tout en répondant aux attentes du château, dans l'optique de permettre la transmission de plus d'informations sans surcharger l'écran, nous avons également travaillé à l'ajout d'un fichier audio.

L'une des propositions de M. Richard, notre professeur encadrant lors de ce projet, était d'effectuer des enregistrements audios de la visite "papier" pour les incorporer dans notre projet. Toutefois notre manque de connaissance, le manque de matériel et le manque de temps causé par la demande tardive de cette application nous ont empêchés de nous focaliser sur cette tâche.

Nous avons trouvé également plus pertinent de solliciter le personnel du château pour cette partie, en effet il passe énormément de temps sur place et possède des connaissances bien supérieures à celle que nous avons pu acquérir au cours de ce projet, il leur sera également plus simple d'effectuer d'éventuelles mises à jour par la suite.

Nous allons maintenant aborder la partie plus technique du projet et nous attarder dans un premier temps sur la composition des différents objets que nous avons ajoutés.

Dans un second temps nous allons décrire la partie écriture des scripts tout en donnant quelques exemples détaillés de notre code.

4.2.2. Composition et fonctionnement des objets

Pour atteindre le fonctionnement attendu nous avons commencé par créer une image cible à laquelle nous avons ajouté un canevas qui nous sert actuellement à transmettre les différentes informations. Pour le moment, lorsque l'image est détectée, l'application affiche une des photos qui nous ont été fournies ainsi qu'une description de l'objet qui correspond au QR Code choisi.

Pour afficher la description, nous avons décidé de réutiliser l'un des assets achetés préalablement par le château et utilisé par Alex pour l'application de la cuisine. Ce dernier nous permet d'afficher un parchemin avec lequel le visiteur pourra interagir pour obtenir une courte description.



Figure 22 : Casque + description Heaume

À terme l'un de nos objectifs serait de remplacer l'image de fond par un asset téléchargé ou bien créé par un professionnel afin de profiter au maximum des fonctionnalités proposées par le logiciel. Pour le moment, lorsque le visiteur se déplace, il a accès à un menu sur lequel il retrouve deux onglets :

- Le premier lui indique l'objectif de la visite, nous avons envisagé la possibilité d'y ajouter le plan de la visite libre pour permettre aux visiteurs de s'orienter en autonomie.

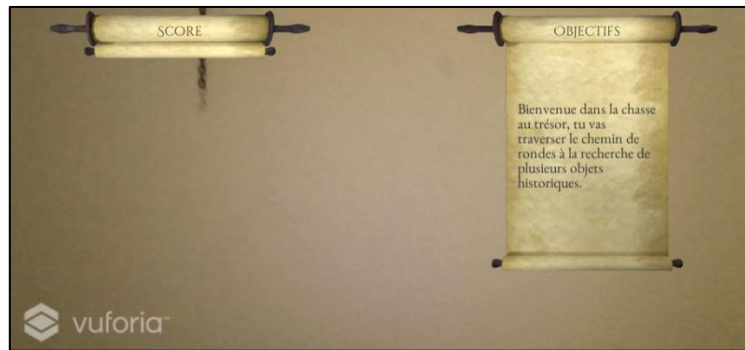


Figure 23 : Menu accueil

- Sur le second onglet on retrouve un compteur que nous avons implémenté dans le cadre de la chasse au trésor initialement prévue. Nous avons choisi de le garder en cas d'une éventuelle utilisation future. Le compteur est initialisé avec la valeur 0, il s'incrémente lorsque l'une des images disposant du script correspondant est détectée.

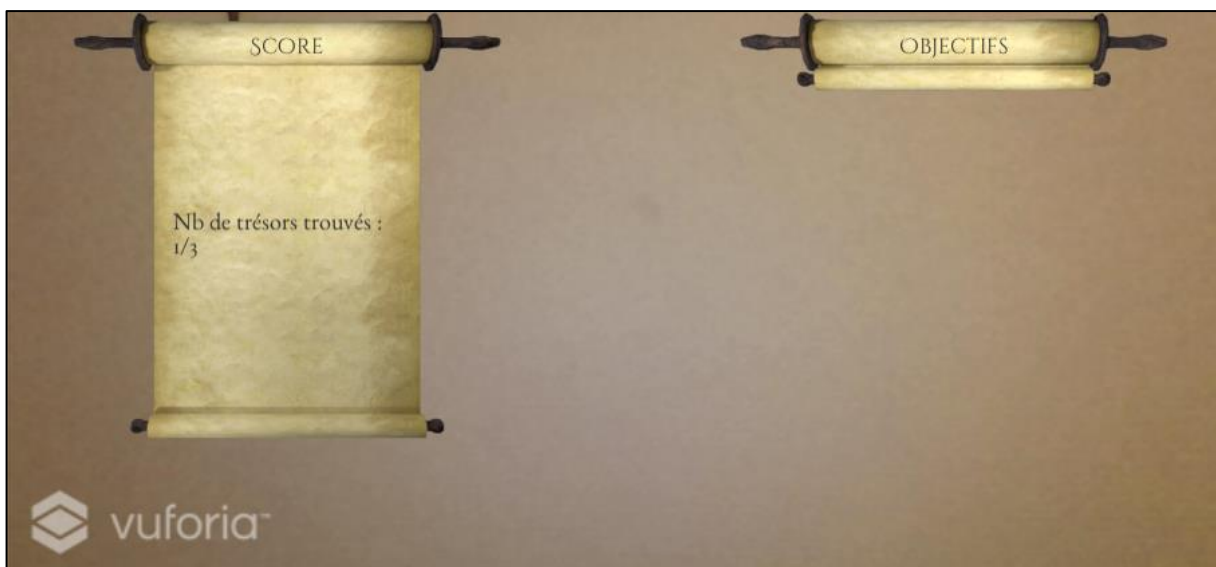


Figure 24 : Affichage score

Nous allons maintenant faire un point sur la partie tracking 3D et expliquer ce qui nous empêche à ce jour d'en utiliser les fonctionnalités.

4.2.3. Test tracking 3D

L'un des projets du château étant la mise en place d'une application de réalité augmentée du donjon nous avons exploré la possibilité d'utiliser la fonction de reconnaissance de modèle 3D évoquée plus tôt. Cependant les contraintes imposées par le scanner nous ont forcées à abandonner cette idée.

Nos nombreuses recherches et discussions avec des étudiants spécialisés en RA nous ont permis de faire émerger une alternative viable qui pourrait relancer le projet du donjon via l'utilisation des fonctionnalités de « Kudan ». Malheureusement faute de temps nous n'avons pas eu l'occasion d'obtenir plus d'informations sur ce dernier, mais il nous semblait essentiel de l'évoquer en cas d'une reprise du projet.

L'objectif de ce projet étant également de nous offrir la possibilité d'acquérir de nouvelles connaissances, nous avons choisi de nous former à l'utilisation du scanner Vuforia via l'ajout de cette fonctionnalité sur notre application de visite en réalité augmentée. Nous avons donc débuté par la création d'un nouveau projet test dédié exclusivement aux modèles 3D. Pour ce faire, nous avons commencé par scanner une clé USB qui nous a servi de modèle test.

Par la suite, nous avons ajouté le modèle scanné à notre base de données Vuforia puis nous avons ajouté un objet sur notre scène jusqu'ici vierge.

L'un des points importants de Vuforia réside dans la nécessité de recharger systématiquement la base de données sur Unity pour disposer de l'intégralité des images cibles. C'est l'une des contraintes qui nous a parfois fait perdre un temps précieux.

La scène test finalement créée comporte donc une modélisation 3D d'une voiture de sport disponible gratuitement sur l'asset store.

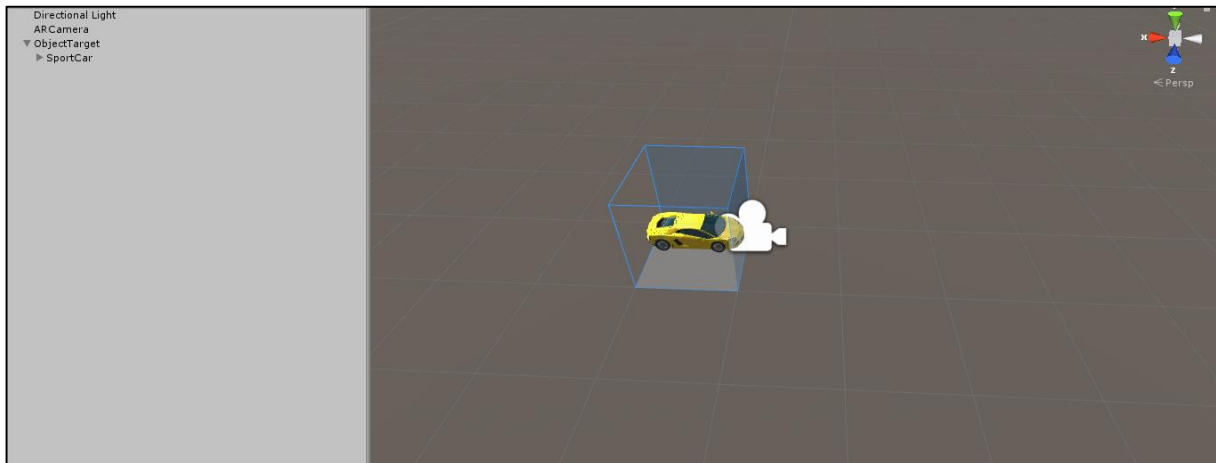


Figure 25 : Test tracking 3D

Ce projet est censé permettre à l'utilisateur d'afficher le modèle 3D en réalité augmentée lorsque la clé USB est détectée par la caméra. Malgré tout à ce jour l'application n'est pas fonctionnelle, nous pensons que le manque de points de référence de notre objet peut en être la cause. Une autre explication pourrait résider dans l'utilisation d'une webcam et non de la caméra d'un smartphone qui nous a semblé au travers des tutoriels observés et des forums consultés présenter une meilleure qualité d'image.

D'après nos recherches, une exportation de l'application pourrait nous permettre de résoudre notre problème, mais l'exportation d'une application sous smartphone Android nécessite l'utilisation d'un ajout de Unity avec lequel nous rencontrons de grandes difficultés que nous ne parvenons pas à résoudre, et ce malgré nos nombreuses recherches sur des forums.

Au vu de nos difficultés actuelles avec la reconnaissance de modèle 3D nous avons préféré ne pas implémenter cette fonctionnalité sur la visite libre en réalité augmentée cependant le projet test remis pourrait se révéler être un bon tutoriel en cas de reprise du projet par de nouveaux étudiants.

Nous allons maintenant développer la partie écriture des scripts de ce mini-projet.

Contrairement au projet de la cuisine évoqué plus tôt celui-ci ne comporte pas énormément de scripts et ces derniers ne se révèlent pas extrêmement complexes, cependant ils restent indispensables au bon fonctionnement du projet et peuvent nécessiter quelques explications pour être compris.

4.2.4. Écriture des scripts

Comme évoqué au cours des parties précédentes l'un de nos objectifs était d'ajouter une fonctionnalité permettant le lancement d'un fichier audio lorsque l'image est détectée. Pour ce faire nous avons donc procédé à l'écriture d'un script qui nous a permis d'implémenter cette fonctionnalité.

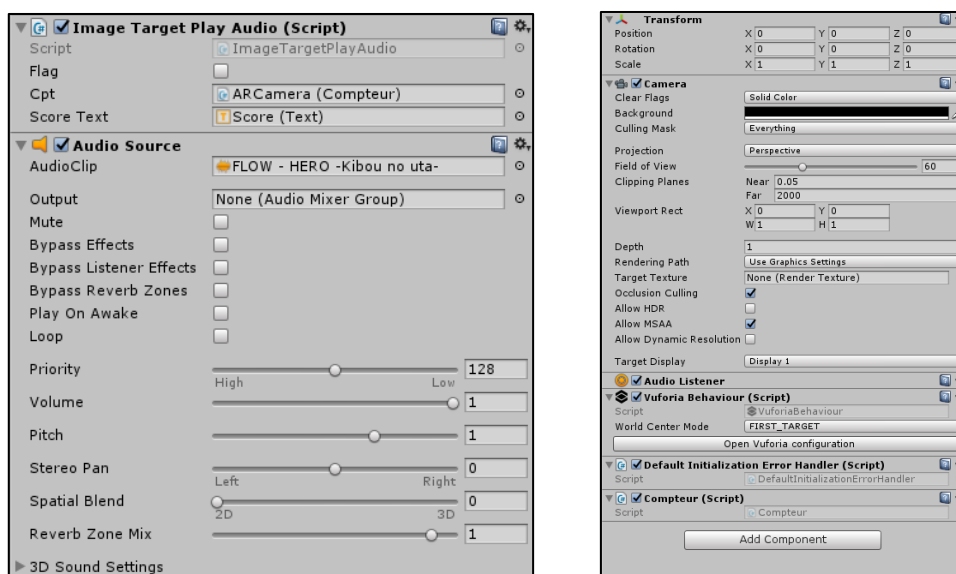


Figure 26 : Audio Source + Script AR Camera

Le script que nous allons maintenant vous présenter est indispensable au bon fonctionnement de notre application. Sans celui-ci il est impossible de démarrer le fichier audio lorsque l'on détecte

l'image. Il est de plus nécessaire de décocher l'option « Play on Awake » si l'on ne souhaite pas que la musique se lance lors du démarrage de l'application.

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using Vuforia;
5 using UnityEngine.UI;
6
7 public class ImageTargetPlayAudio : MonoBehaviour,
8 ITrackableEventHandler
9 {
10     private TrackableBehaviour mTrackableBehaviour;
11     public bool Flag;
12     public Compteur cpt;
13     public Text scoreText;
14
15     void Start()
16     {
17         Flag = false;
18         mTrackableBehaviour = GetComponent<TrackableBehaviour>();
19         if (mTrackableBehaviour)
20         {
21             mTrackableBehaviour.RegisterTrackableEventHandler(this);
22         }
23     }
24
25     public void OnTrackableStateChanged(
26         TrackableBehaviour.Status previousStatus,
27         TrackableBehaviour.Status newStatus)
28     {
29         if (newStatus == TrackableBehaviour.Status.DETECTED ||
30             newStatus == TrackableBehaviour.Status.TRACKED ||
31             newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
32         {
33             // Play audio when target is found
34             GetComponent<AudioSource>().Play();
35             if (Flag == false)
36             {
37                 cpt.Add ();
38                 scoreText.text = "Nb de trésors trouvés : " + cpt.GetCompteur () + "/3";
39                 Flag = true;
40             }
41         }
42         else
43         {
44             // Stop audio when target is lost
45             GetComponent<AudioSource>().Stop();
46         }
47     }
48 }
49
```

Figure 27 : Script audio

N'ayant pas à disposition d'enregistrements audio provenant du château nous nous sommes orientés sur un effet sonore pour tester notre scène.

Concernant notre script il existe deux versions de ce dernier :

- Une première qui dispose d'un compteur
- Une seconde, sans compteur

Cela permettra aux futurs étudiants d'EI4 en charge de ce projet de reprendre l'application de chasse au trésor en cas d'une éventuelle demande du château.

Afin de pouvoir suivre l'image, et ce même lorsqu'on la déplace, il est nécessaire d'ajouter un gestionnaire de tracking (ITrackableEventHandler). Pour ce qui est de la gestion du son, cela dépend de la librairie par défaut de Unity qui nous permet d'utiliser les options « Play » et « Stop » sur des fichiers audio. La combinaison de ces deux fonctionnalités nous permet actuellement de ne lancer le

son que lorsque l'image est détectée (le son se coupe si l'image sort du champ de la caméra), en application il serait probablement plus judicieux de supprimer la fonction Stop de notre script pour éviter une interruption lors d'explications.

Le compteur s'initialise dans un script placé sur l'AR Caméra.

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Compteur : MonoBehaviour {
6
7     static int Score; // Use this for initialization
8     void Start () {
9         Score = 0;
10    }
11
12    // Update is called once per frame
13    void Update () {
14
15    }
16
17    public void Add()
18    {
19        Score += 1;
20    }
21
22    public int GetCompteur()
23    {
24        return Score;
25    }
26
27 }
28
```

Figure 28 : Script Init compteur

L'incréméntation du compteur s'effectue dans un script attaché directement à l'objet. L'une des principales difficultés que nous avons rencontrées avec ce script réside dans le fait que nous ne devons pas incrémenter le compteur lorsqu'une image est scannée plusieurs fois.

Pour pallier à cette difficulté, nous avons ajouté une variable booléenne « flag » initialisée avec une valeur « false » que l'on passe à « true » lorsque l'image est détectée pour la première fois.

À chaque nouveau scan de l'image la valeur de « flag » est testée, si la valeur est « true » on ne procède pas à l'incréméntation du compteur.

4.3. Les perspectives d'avenir

Nous avons songé à plusieurs éléments qui pourraient être, soit améliorées, soit créés, afin de rendre le projet plus complet. Deux d'entre eux concernent la scène de la cuisine médiévale, une autre, le donjon et la dernière, la joute.

L'ajout de nouveaux ingrédients dans la scène de la cuisine serait favorable à la finalisation de notre projet. En effet, en suivant le tutoriel que nous avons réalisé il serait ultérieurement possible de reprendre notre scène pour la compléter. Il serait ainsi possible de diversifier les choix et d'augmenter considérablement le nombre de recettes réalisables.

Il pourrait également être envisagé de rajouter plusieurs scènes à notre projet. Nous avons notamment pensé à 4 possibilités :

- Une scène où il serait possible d'observer la liste des légumes, d'obtenir plus d'informations sur chacun d'entre eux et savoir dans quelles recettes ils étaient utilisés à l'époque.
- Puis deux autres scènes où il en serait de même pour les viandes et les épices.
- Une quatrième scène permettant de voir quels ustensiles étaient utilisés à l'époque et quelle fût leur évolution au cours du temps.

Au château du Plessis Macé, il est possible de voir les vestiges d'un donjon. Avec notre tuteur et les responsables du château, nous avons envisagé de faire une scène où il aurait été possible de faire apparaître le donjon, identique à celui de l'époque. Pour cela, nous devons recevoir des plans du donjon, pour pouvoir les scanner avec Vuforia par le biais de la reconnaissance 3D des objets, comme décrit précédemment. Malheureusement nous avons rencontré deux problèmes. Le premier est qu'avec Vuforia nous ne pouvons scanner que de petits objets, le deuxième est que les plans de ce donjon n'ont pas été retrouvés. Nous nous sommes donc vus dans l'incapacité de mener ce projet à bien. Il nous semble indispensable pour le château que le projet du donjon voie le jour.

Lorsque nous avons repris le projet à la suite des deux précédents étudiants, il comportait une scène sur un combat de joute. Nous avons voulu la continuer comme il était décrit dans le rapport d'Arthur Emrot afin de pouvoir changer les éléments du décor à la guise de l'utilisateur sans que ceci n'influe sur la scène :

- Améliorer la trajectoire et les mouvements du cavalier
- Permettre à l'utilisateur de choisir le matériau des armes et des armures.

Malheureusement, sur demande de notre tuteur, nous nous sommes focalisés sur la cuisine et la visite en réalité augmentée.

En ce qui concerne le projet de visite libre en RA nous avons envisagé certaines possibilités d'évolution. Ce projet nous ayant été proposé par le château assez tardivement nous n'avons pas pu

le mener à terme, mais il nous semble toutefois représenter une base solide pour toute personne qui le reprendra.

Nous profitons donc de cette section pour évoquer toutes les perspectives d'évolution que nous avons envisagées durant ces dernières semaines.

Dans un premier temps nous pensons qu'il pourrait être intéressant comme évoqué un peu plus tôt de relancer le projet de chasse au trésor, cela apporterait un côté ludique à la visite qui permettrait au personnel du château de capter l'attention des enfants. L'utilisation de QR Code, d'assets 3D et des fonctionnalités de reconnaissance d'objets pourrait permettre de surmonter les contraintes de luminosité imposées par le chemin de ronde.

Dans un deuxième temps nous avons soulevé au sein du groupe la possibilité d'étendre cette chasse à l'intégralité des parties accessibles en visite libre via la disposition de cibles en extérieur. Cependant c'est une initiative qui n'a pas encore été évoquée avec le personnel du château et qui nécessitera donc une concertation avec ces derniers.

5. Conclusion

Ce projet s'est révélé être très intéressant. Nous avons d'une part, acquis de nouvelles connaissances en réalité virtuelle et augmentée et d'autre part, acquis bon nombre de connaissances historiques sur le château et ses différents propriétaires. Au cours de ce projet nous avons tous développé de nouvelles compétences dans des domaines qui n'ont pas forcément été étudiés à l'ISTIA dans le cadre de notre formation (tracking 3D, réalité augmentée, ...). Avec du recul, nous nous apercevons que notre manque d'organisation nous a parfois fait perdre un temps précieux.

Nous avons souvent été confrontés à notre manque de connaissances, ce qui a nécessité de longues phases de formation et de suivis de tutoriels pour pouvoir les résoudre.

Nous avons rencontré quelques difficultés dans la gestion des différents emplois du temps. En effet, il nous a parfois été difficile de prendre contact avec le personnel du château, lui-même très occupé par la préparation des différents spectacles. De ce fait, nous sommes parfois restés sans réponse à nos questions ce qui a par moment ralenti notre progression.

Nous regrettons de ne pas avoir osé prendre contact avec le château plus tôt. Cela nous aurait permis d'avoir une meilleure vision de leurs attentes et nous aurait également permis de démarrer plus tôt sur les nouveaux projets.

L'ambiance de travail, la sympathie et les compétences du personnel ainsi que la disponibilité de Mme Boutteaud et des autres membres du personnel du château restent des éléments essentiels qui nous ont permis de travailler dans les meilleures conditions.

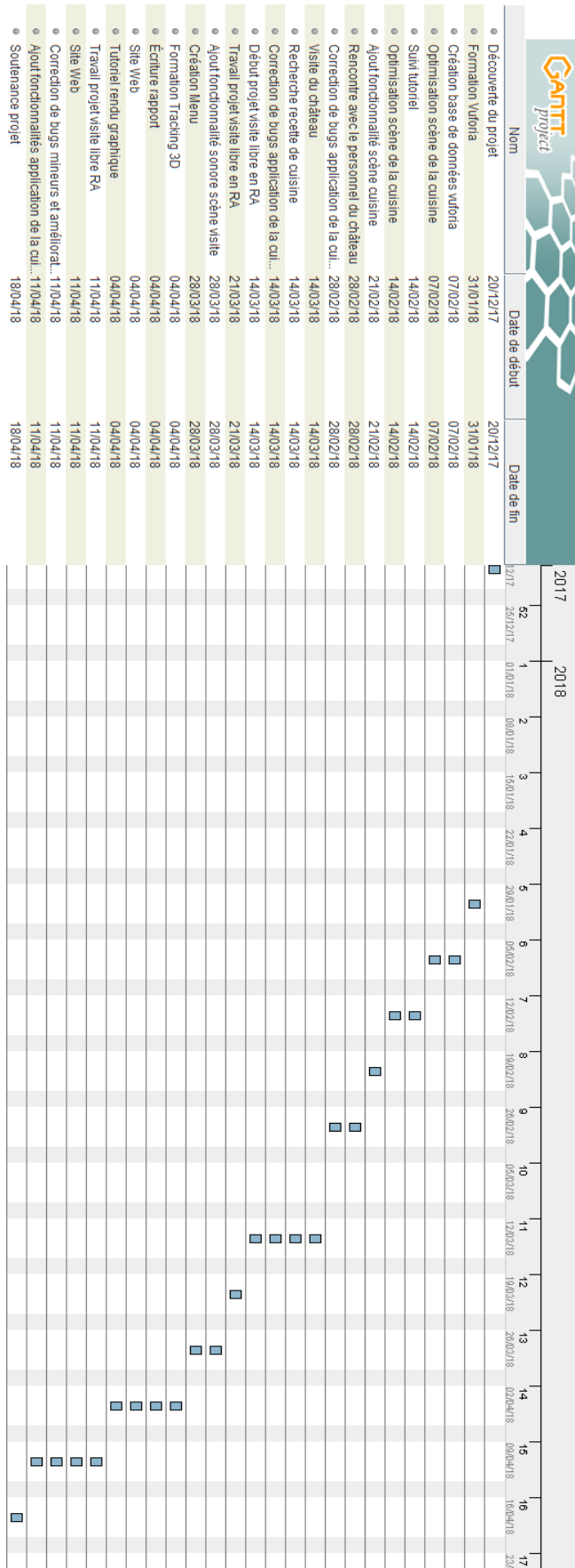
6. Table des figures

Figure 1 : Diagramme de Gantt	4
Figure 2 : Part de chaque activité dans le projet	4
Figure 3 : Configuration Vuforia	5
Figure 4 : Base de données Vuforia	6
Figure 5 : QR Code + points de repère.....	7
Figure 6 : Object Scanning Target.....	8
Figure 7 : Modèle 3D	8
Figure 8 : Scanner Vuforia	8
Figure 9 : Image cible projet tutoriel	9
Figure 11 : Chevalier en réalité augmenté.....	9
Figure 11 : Scène tutoriel	9
Figure 12 : Scène initiale de la cuisine.....	11
Figure 13 : Nouveau bouton	11
Figure 14 : Ancien bouton	11
Figure 15 : Gestion des ingrédients dans l'onglet "Recette"	12
Figure 16 : Classe "Recette"	12
Figure 17 : Cas où la recette est ratée	13
Figure 18 : Cas où la recette est réussie	13
Figure 19 : Cas avec Scroll Rect	14
Figure 20 : Cas sans Scroll Rect.....	14
Figure 21 : Plan visite libre	15
Figure 22 : Scène actuelle.....	16
Figure 23 : Casque + description Heaume	17
Figure 24 : Menu accueil	18
Figure 25 : Affichage score	18
Figure 26 : Test tracking 3D.....	19
Figure 27 : Audio Source + Script AR Camera	20
Figure 28 : Script audio.....	21
Figure 29 : Script Init compteur	22
Figure 30: Annexe 1.....	28

7. Bibliographie

- <https://www.vuforia.com/tools-and-resources.html>
- http://www.oldcook.com/cuisine-recettes_medievaales
- <https://library.vuforia.com/articles/Training/Vuforia-Object-Scanner-Users-Guide>
- <https://forum.unity.com/>
- <https://unity3d.com/fr/learn/tutorials>
- <https://unity3d.com/fr/learn/tutorials/topics/user-interface-ui/scroll-view>

Figure 29: Annexe 1



Château du Plessis Macé - Techniques de réalité augmentée au service de l'histoire et du patrimoine

Résumé : L'objectif de ce projet était de réaliser plusieurs applications de réalité augmentée en partenariat avec le château du Plessis-Macé. Ce projet ayant été débuté l'année passée le but de ce projet de 4e année était d'améliorer celui déjà existant.

Les parties essentielles développées au cours de ces semaines de projet sont les suivantes :

- Amélioration et finition de l'application cuisine du logis
- Mise en place d'une visite libre en réalité augmentée

Au terme de ce projet la première application citée semble être terminée, un ajout de recettes peut être envisagé mais celle-ci est désormais fonctionnelle.

Concernant la visite en RA, le projet ayant été commencé tardivement il n'est pas encore totalement abouti mais semble être une base solide de travail en cas de reprise du projet, de plus les fonctionnalités ajoutées permettent au personnel du château d'envisager la reprise de certaines applications qui n'ont pas pu être traitées durant ce projet.

Mots-clés : Réalité augmentée, Tracking 3D, Aspect graphique, Optimisation, Vuforia

Summary: The aim of this project was to create several applications in augmented reality for the castle of « Plessis-Macé », in order to do this, two projects were imagined:

- Optimization of the cook application
- Beginning of an AR visit of the castle

Finally, the cook application is almost finished, some recipes can be added to make it more complete but it seems functional and presently works.

Due to the late start of the AR visit project, this second application is not finished yet but it seems to be a good base for the persons who will take the project later. It allows the user to scan some QR Code to print different pictures on the screen.

All the functionalities which were added during this project should be used to start new applications.

Keywords: Augmented reality, Vuforia, 3D tracking, QR Code, Optimization