

## Travail de Bachelor 2018

# Sentiers ludiques



HES-SO Valais  
Filière Informatique de gestion  
[www.hevs.ch](http://www.hevs.ch)

Etudiant : Luca Centofanti  
Professeur : Mme Maria Sokhn  
Collaboratrice scientifique : Mme Camille Pellaton  
Déposé le : 30 juillet 2018

## RESUME

L'objectif principal de ce travail est de moderniser les sentiers mis à disposition par le Centre Régional d'Etudes des Populations Alpines (CREPA). La modernisation rend les sentiers plus ludiques pour les enfants et augmente ainsi l'attrait pour ces parcours. Pour atteindre cet objectif, un prototype d'une application disponible sur toutes les distributions existantes telles que, Android et IOS, sera créé. Cette application contiendra de la réalité augmentée ainsi que de l'orientation GPS. Elle accompagnera un projet déjà existant et devra apporter des connaissances de façon ludique et en aucun cas favoriser une utilisation excessive. Elle sera adoptée à des moments appropriés, afin de libérer l'utilisateur de l'emprise du téléphone portable.

L'application devra accompagner les utilisateurs à travers différents sentiers pédestres afin de leur faire découvrir des aspects de la région où ils se situent. De plus, des jeux seront disponibles à chaque étape afin d'apporter des connaissances supplémentaires.

Pour atteindre ce but, un état de l'art sera effectué sur différentes applications afin d'avoir une vue d'ensemble et de s'inspirer du marché. De plus, une étude sur les technologies existantes permettant la fusion de la réalité augmentée, du GPS et du concept de multiplateforme sera créée. L'application se doit d'être disponible sur le plus grand nombre d'appareils toutes distributions confondues.

Une fois l'analyse des différents outils permettant de répondre à chaque problématiques rédigée et acquise, les parties importantes du développement seront décrites afin d'exposer leur complexité ou au contraire leur simplicité.

L'application proposée contiendra des jeux ludiques ainsi qu'un GPS en réalité augmentée, en plus de trouver toutes les informations nécessaires à la pratique des différents sentiers. Elle se montrera complète et démontrera ce qui est possible de réaliser dans ce registre. Un aperçu global de l'application sera également présent au sein de ce rapport.

Mots clés : Réalité augmentée, jeux ludiques, GPS, multiplateforme, Unity, Vuforia

## AVANT-PROPOS

Ce projet a été proposé par le Centre Régional d'Etudes des Populations Alpines, une association de 1990 localisée dans la commune de Sembrancher. Leurs activités principales sont l'animation socioculturelle, l'histoire des vallées avoisinantes et un centre de documentation et archives. Grâce à ce travail, ils souhaiteraient moderniser un de leur projet.

Le but étant de leur remettre un prototype d'une application mobile qui compléterait leur projet de sentiers ludiques pédestres se trouvant sur les communes de Grand Entremont, la Vallée du Trient et de la Fouly. L'application intégrera des jeux ludiques, un album photos, de la navigation par GPS et de la réalité augmentée. Elle apportera amusement et outils pratiques afin de faciliter l'accès à ces parcours.

Mixant l'aspect plein air, jeux et réalité augmentée, la thématique de ce travail de Bachelor m'a tout de suite interpellé. De plus, les problématiques évoquées par le client sont contraignantes et ouvrent la voie à plusieurs défis ! La réalité augmentée est une nouvelle technologie ce qui rend chaque étape de la conception innovante et motivante.

## REMERCIEMENTS

Je souhaite remercier ainsi que témoigner toute ma reconnaissance aux personnes m'ayant encadré et soutenu durant ce travail. C'est grâce à elles que l'élaboration de ce projet a pu être possible :

Mme Maria Sokhn, professeure à la HES-SO Valais/Wallis, pour m'avoir guidé grâce à ses conseils avisés, ainsi que pour ses idées.

Mme Seraphine Mettan et M. Yann Decorzant, membres du CREPA, pour leur confiance et leur proximité.

M. Antoine Widmer, professeur à la HES-SO Valais/Wallis, pour m'avoir aidé lors de décisions techniques.

Mme Camille Pellaton, collaboratrice scientifique au sein du Data Semantics Lab, pour ses retours lors de nos meetings, qui ont toujours su faire avancer le projet.

M. Maxime Bétrisey, étudiant à la HES-SO, pour son aide lors de la compréhension des Quaternions et des rotations.

M. Patrick Clivaz, étudiant à la HES-SO, pour l'installation et les tests de l'application IOS grâce à son Mac et sa licence Xcode.

M. Tony Centofanti et Mme Sandy Millius pour la relecture et la correction orthographique du rapport.

## TABLE DES MATIÈRES

<b>LISTE DES ILLUSTRATIONS .....</b>	<b>VIII</b>
<b>LISTE DES TABLEAUX .....</b>	<b>X</b>
<b>LISTE DES ABBREVIATIONS .....</b>	<b>XI</b>
<b>GLOSSAIRE .....</b>	<b>XI</b>
<b>INTRODUCTION .....</b>	<b>1</b>
<b>CONTEXT .....</b>	<b>2</b>
1.1. LE CREPA .....	2
1.2. CHARLOTTE LA MARMOTTE .....	2
1.2.1. <i>Les sentiers</i> .....	2
<b>1. ETAT DE L'ART .....</b>	<b>7</b>
1.1. PROBLÉMATIQUE.....	7
1.1.1. <i>Les enfants</i> .....	7
1.1.2. <i>Disponibilités</i> .....	7
1.1.3. <i>Innovation</i> .....	7
1.2. LES SOLUTIONS EXISTANTES SUR LE MARCHÉ .....	8
1.2.1. <i>Mhikes</i> .....	8
1.2.2. <i>ViewRanger Skyline</i> .....	10
1.2.3. <i>Pokémon Go</i> .....	12
1.3. DISCUSSION DE L'ÉTAT DE L'ART .....	16
1.3.1. <i>Mhikes</i> .....	16
1.3.2. <i>ViewRanger Skyline</i> .....	16
1.3.3. <i>PokémonGo</i> .....	16
1.3.4. <i>Tableau récapitulatif</i> .....	17
<b>2. TECHNOLOGIES .....</b>	<b>18</b>
2.1. ARCORE & ARKIT.....	18
2.1.1. <i>ARCore</i> .....	18
2.1.2. <i>ARKit</i> .....	19
2.2. XAMARIN.....	19
2.2.1. <i>Réalité augmentée</i> .....	20
2.2.2. <i>Prototype de l'application proposé par Data Semantics Lab</i> .....	21
2.3. UNITY .....	22

2.3.1.	Réalité augmentée.....	22
2.4.	CHOIX TECHNOLOGIQUES .....	25
<b>3.</b>	<b>DÉVELOPPEMENT DE L'OUTIL .....</b>	<b>26</b>
3.1.	ARCHITECTURE INTERNE .....	28
3.2.	PERSISTENT .....	30
3.3.	MAIN MENU.....	32
3.3.1.	MainMenuCanvas.....	33
3.4.	FINDSENTIERSTART .....	39
3.5.	FOULY GAMES .....	43
3.5.1.	Le loup .....	43
3.5.2.	La prise de photo .....	46
3.5.3.	Les images prennent vies.....	49
3.5.4.	Le Quizz.....	51
3.6.	MULTIPLATEFORME .....	53
<b>4.</b>	<b>PROBLÈMES RENCONTRÉS .....</b>	<b>54</b>
4.1.	NOTIFICATION GPS .....	54
<b>5.</b>	<b>AMÉLIORATION DU PROTOTYPE .....</b>	<b>56</b>
<b>6.</b>	<b>EVALUATION DE L'APPLICATION .....</b>	<b>57</b>
<b>7.</b>	<b>PLANIFICATION .....</b>	<b>58</b>
<b>8.</b>	<b>CONCLUSION .....</b>	<b>59</b>
8.1.	FEEDBACK PERSONNEL.....	59
<b>9.</b>	<b>RÉFÉRENCES ET SOURCES .....</b>	<b>61</b>
<b>10.</b>	<b>ANNEXE I : CONFIGURATION DE VUFORIA .....</b>	<b>65</b>
<b>11.</b>	<b>ANNEXE II : PLANIFICATION .....</b>	<b>71</b>
<b>12.</b>	<b>ANNEXE III : CAHIER DES CHARGES .....</b>	<b>72</b>
12.1.	INTRODUCTION .....	72
12.2.	PROBLEMATIQUE .....	73
12.3.	LES ENFANTS .....	73
12.4.	DISPONIBILITÉS.....	73
12.5.	INNOVATION .....	73
12.6.	ETAPES DE REALISATION ENVISAGEES .....	74
12.6.1.	Analyse de l'état de l'art.....	74

12.6.2.	<i>Choix technologique</i> .....	74
12.6.3.	<i>Implémentation / validation</i> .....	74
12.7.	PLANIFICATION .....	75
<b>13.</b>	<b>DÉCLARATION DE L'AUTEUR</b> .....	<b>77</b>

## LISTE DES ILLUSTRATIONS

Figure 1 : Sentier de la Fouly, Document de jeux .....	3
Figure 2 : Sentier de la Fouly, Passeport .....	3
Figure 3 : Sentier de la Fouly, Départ.....	4
Figure 4 : Sentier de la Fouly, Exemple de poste .....	4
Figure 5 : Sentier de la Fouly, Autres postes .....	5
Figure 6 : Sentier de la Fouly, Plantes médicinales .....	5
Figure 7 : Sentier de Champex-Lac / Orsières, Postes .....	6
Figure 8 : Sentier de la Creusaz, Postes .....	6
Figure 9 : Mhikes, Parcours .....	9
Figure 10 : Mhikes, Question.....	9
Figure 11 : ViewRanger, Skyline.....	11
Figure 12 : PokémonGo, Map .....	13
Figure 13 : PokémonGo, Réalité augmentée.....	14
Figure 14 : PokémonGo, Informations.....	14
Figure 15 : PokémonGo, Météo.....	15
Figure 16 : Xamarin, Architecture .....	20
Figure 17 : Prototype Xamarin, Vues .....	21
Figure 18 : Vuforia, Ground Plane .....	23
Figure 19 : Vuforia, Model Target.....	24
Figure 20 : Vuforia, Image Target .....	24
Figure 21 : Vuforia, Cylinder Target.....	24
Figure 22 : Vuforia, Object Target .....	25
Figure 23 : Architecture du projet Unity .....	28
Figure 24 : Architecture, Vuforia Targets .....	29
Figure 25 : « Persistent » Dev. Script DataController .....	30
Figure 26 : Persistent, Dev. Script DataController, Méthodes GetCurrent. et PopulatesGames ..	30
Figure 27 : Persistent, Dev. Script RoundData.....	31
Figure 28 : Persistent, Dev. Script QuestionData .....	31
Figure 29 : Persistent, Dev. Script AnswerData .....	31
Figure 30 : Persistent, Dev. Script SaveLoad .....	32
Figure 31 : MainMenu, Architecture .....	32
Figure 32 : MainMenu, OtherObject, Dev. Script ShowCanvasDontDestroyed .....	33
Figure 33 : MainMenuPanel, Vue .....	34
Figure 34 : MainMenu, Clickable Button .....	34
Figure 35 : MainMenu. Bouton passeport .....	35
Figure 36 : SentiersPanels, Architecture .....	35
Figure 37 : SentiersPanels, Vue .....	35



Figure 38 : PlayPanel, Vue .....	36
Figure 39 : PlayerPanel & LevelPanel, Vues .....	36
Figure 40 : BlocPanel, Vue .....	37
Figure 41 : SentierInfoPanel, Vue.....	37
Figure 42 : PassportsPanel, Vue Fouly Games Passed .....	38
Figure 43 : AlbumsPanel, Vue avec photo .....	38
Figure 44 : InformationPanel, Vue .....	39
Figure 45 : FindSentierStart, Dev. Script Méthode StartlocationService .....	41
Figure 46 : FindSentierStart, Dev. Script Start TrackGpsCoordinates .....	41
Figure 47 : FindSentierStart, Dev. Script Méthode AngleFromCoordinate .....	42
Figure 48 : FindSentierStart, Dev. Script Update TrackGpsCoordinates .....	42
Figure 49 : FindSentierStart, Vue .....	42
Figure 50 : Jeu le loup, Architecture.....	43
Figure 51 : Jeu le loup, Vuforia prefabs .....	43
Figure 52 : Jeu le loup, ImageTarget.....	44
Figure 53 : Jeu le loup, l'asset 3D du loup .....	44
Figure 54 : Jeu Le loup, Vue .....	45
Figure 55 : Jeu Prise de photo, Architecture .....	46
Figure 56 : Jeu Prise de photo, AR Camera .....	46
Figure 57 : Jeu Prise de photo, Dev. Script FrontCameraAccess .....	47
Figure 58 : Jeu Prise de photo, Dev. Script TakePicture .....	47
Figure 59 : Jeu Prise de photo, Vue .....	48
Figure 60 : Jeu Les images prennent vies, photo du poste sans RA.....	49
Figure 61 : Jeu Les images prennent vies, photo du poste avec RA, Cerf .....	50
Figure 62 : Jeu Les images prennent vies, photo du poste avec RA, Bouquetin.....	50
Figure 63 : Jeu Les images prennent vies, Architecture.....	50
Figure 64 : Jeu Les images prennent vies, Image Target .....	51
Figure 65 : Jeu Quizz, Vue mauvaise réponse .....	51
Figure 66 : Jeu Quizz, Architecture .....	52
Figure 67 : Jeu Quizz, Dev. Méthode ShowQuestion.....	53
Figure 68 : Multiplateforme, Android et IOS .....	53
Figure 69 : Vuforia configuration, téléchargement.....	65
Figure 70 : Vuforia configuration, License Manager .....	66
Figure 71 : Vuforia configuration, License Details .....	66
Figure 72 : Vuforia configuration, Target Manager .....	67
Figure 73 : Vuforia configuration, Création de la base de données .....	67
Figure 74 : Vuforia configuration, Détails de la base de données .....	68
Figure 75 : Vuforia configuration, Téléchargement de la base de données.....	68

---

Figure 76 : Vuforia configuration, localisation de la base de données .....	69
Figure 77 : Vuforia configuration, Localisation de VuforiaConfiguration .....	69
Figure 78 : Vuforia configuration, Caméra RA configuration 2 .....	70
<b>LISTE DES TABLEAUX</b>	
Tableau 1 : Etat de l'art, Tableau récapitulatif .....	17

## LISTE DES ABREVIATIONS

AR : Augmented Reality

RA : Réalité Augmentée

VR : Virtual Reality

RV : Réalité virtuelle

CREPA : Centre Régional d'Etudes des Populations Alpines

GPS : Global Positioning System

SDK : Software development kit

UWP : Universal Windows Platform

QR : Quick Response

## GLOSSAIRE

Framework : « Un framework désigne en programmation informatique un ensemble d'outils et de composants logiciels à la base d'un logiciel ou d'une application. C'est le framework, encore appelé structure logicielle en français, qui établit les fondations d'un logiciel ou son squelette applicatif. » (La Rédaction, 2018)

## INTRODUCTION

Le Centre Régional d'Etudes des Populations Alpines (CREPA) est une association fondée en 1990. Son but est de poursuivre les recherches de la Commune de Bagnes et de les étendre dans les vallées proches. Le CREPA permet de développer des projets visant la formation des jeunes au patrimoine socioculturel des communes avoisinantes (CREPA, s.d. 2018), qui sont :

Grand Entremont / Vallée du Trient / Fouly

Ces projets peuvent être divers et variés mais l'objectif reste le même : créer des liens entre la population indigène et touristique, tout en accentuant l'identité et les richesses locales. (CREPA, s.d. 2018)

« Charlotte la Marmotte » est un projet du CREPA qui a été créé en 2012. Le rôle de ce projet est de faire découvrir au jeune public les particularités du patrimoine géographique alpin. Il est présent sur cinq sentiers :

La Fouly (2012), Champex-Lac (2014), La Creusaz (2014), Trient (2015) et Emosson

Actuellement, les enfants ont « seulement » un questionnaire papier à remplir durant leur périple sur l'un des cinq sentiers à disposition.

Le travail de Bachelor se focalise sur ce projet du CREPA. Il consiste à rendre les sentiers plus ludiques avec des technologies qui permettent aux enfants de s'amuser tout en ayant un complément d'information, mais également un aspect utilitaire pour les parents (description du sentier, carte, etc.) afin de faciliter leur périple.

## CONTEXT

### 1.1. Le CREPA

Le Centre Régional d'Etudes des Populations Alpines se veut d'avoir un œil sur sa région d'antan et l'autre sur la période actuelle ; elle se veut être un pont entre les générations ! Le CREPA est composé d'un centre de recherche et d'un institut lié au patrimoine qui permettent le développement de projets originaux (CREPA, s.d. 2018).

Ce travail de Bachelor consiste à moderniser le projet du CREPA, « Charlotte la Marmotte », décrit ci-dessous.

### 1.2. Charlotte la Marmotte

Le projet a été créé en 2012 et a pour but de faire découvrir aux enfants (8 à 12 ans) les particularités du patrimoine géographique alpin. L'idée est de mettre à disposition des sentiers didactiques pour les enfants, afin qu'ils puissent découvrir les paysages proposés de différentes façons et d'apprendre des notions scientifiques à travers leurs propres expériences ou représentations. (CREPA, s.d. 2018)

Pour ce faire un document avec des questions et des jeux 3D sur le développement touristique de la région et de la nature en général est fourni à chaque sentier. Grâce aux questions, l'enfant développe ses connaissances et les jeux 3D permettent d'entraîner sa dextérité. (CREPA, s.d. 2018)

Des postes sont proposés tout au long du sentier afin de s'instruire et de s'amuser. Chaque poste est différent et permet de sensibiliser la personne à une valeur de la région. Un panneau avec les réponses est disponible à la fin du sentier. (CREPA, s.d. 2018)

#### 1.2.1. Les sentiers

Cinq sentiers sont à disposition pour répondre aux besoins de ce projet. D'autres sont en cours de création.

##### I. La Fouly

Le sentier de la Fouly est celui qui a été parcouru à titre personnel afin d'avoir une vue globale sur l'idée du projet et d'obtenir des informations lors de son parcours. Le sentier de la Fouly est un sentier « facile » que l'on peut pratiquer avec des enfants. Cependant, les poussettes sont à éviter car des endroits sont péniblement accessibles. Le tracé est une boucle de 2 kilomètres qui commence au village de la Fouly et y finit. Pour se rendre au sentier, toutes les informations nécessaires sont disponibles sur le site internet de Charlotte la Marmotte.

Pour commencer le sentier didactique, il faut se rendre à l'office de tourisme qui se trouve au village, non loin du commencement du sentier. Les documents de jeux, passeports et informations sont fournis ici.

Discover the different kinds of burrows and underground tunnels dug by marmots (drawn on the game). Each one has to prepare its own **winter burrow** and bring grass to it to cover the walls of its "bedroom". Help Charlotte choose the right entrance.

After going over the catwalk, stop at the start of the path that leads to the A Neuve Cabane! Look at the glacier with the same name on your left and compare it to the photo taken in 1930! What do you notice?

Compare the photograph taken of La Fouly in the 1940s and the village today. How many chalets do you think have been built since then?

Circle the right number: 30 - 70 - 100 - 300

Taking into consideration the three photos (A, B, C) on the bottom of the board, which of them is the least suitable for the Iscles grasshopper?

Look at the rock face opposite you and the drawing of it on the board. The colours differentiate the two types of rock of which the mountain is composed. Using the board, join up the elements opposite!

In the 1900s, which tools were used to build this barn? Circle the right tools!

Watch out, an eagle has seen where Charlotte is! Help her to hide so that she doesn't get gobbled up! Balancing on the platform, go with Charlotte (the ball) into her burrow as quickly as possible! Time how long it takes for you to save her!

The signs above are all mixed up! Using the printed tarpaulins, link up the animals with the right characteristics! To end off, use the box to listen to the noises these animals make and find out what animal makes what noise!

If you join up the link between the rings on the exposed trunk and La Fouly's timeline, which ring (A, B, C) corresponds to:

- When the Grand Hotel was built? B
- The avalanche that destroyed a large part of the forest on the Neuve plateau? A
- The construction of the first ski-lift? C

Well done, you've just completed Charlotte's path and here you are at the finish line!

To check your answers, just shift the movable pieces where each answer is written. On the back, a beautiful picture of Charlotte will appear.

After the bridge, on the left bank, look out for the house on the right. Find it on the photograph taken in the 1930s and name three differences!

Figure 1 : Sentier de la Fouly, Document de jeux<sup>1</sup>

<p>Sentier Emersion</p> <p>date: 9/6/18</p>	<p>Sentier de la mine</p> <p>date:</p>
<p>Sentier</p> <p>date:</p>	<p>Sentier</p> <p>date:</p>

Bravo et merci de ta visite!

Figure 2 : Sentier de la Fouly, Passeport<sup>2</sup>

<sup>1</sup> Source : Scan de mon document personnel

<sup>2</sup> Source : Scan de mon passeport personnel de la Fouly

Le document de jeux est indispensable au bon déroulement du sentier car on y trouve toutes les questions. De plus, on peut également y trouver des informations, telle qu'une carte pour se repérer. Le passeport est la preuve que le sentier a bien été réalisé. Si tous les sentiers sont effectués, un cadeau est remis aux randonneurs.

Le tracé est guidé par des flèches blanches avec le logo de Charlotte. Il faut être attentif à ne pas les manquer car sinon, il est difficile de s'orienter. Il en est de même pour les postes, qui sont des lieux spécifiques à disposition où l'on peut répondre aux questions ou faire les jeux du questionnaire. En majorité, les postes sont de grand panneaux blancs recouverts d'informations ou de jeux afin de répondre au questionnaire. Cependant, les postes peuvent également être des petits panneaux blancs et carrés, sans information, qui ne sont pas toujours faciles à repérer.



Figure 3 : Sentier de la Fouly, Départ<sup>3</sup>



Figure 4 : Sentier de la Fouly, Exemple de poste<sup>4</sup>

<sup>3</sup> Source : Photo personnelle du panneau de départ du sentier de la Fouly

<sup>4</sup> Source : Photos personnelles du poste 6 du sentier de la Fouly et du panorama





Figure 5 : Sentier de la Fouly, Autres postes<sup>5</sup>

Le sentier de la Fouly comprend également vingt panneaux sur les plantes médicinales du Val Ferret mis à disposition par la commune d'Orsières. Il est donc à la fois possible de s'instruire sur ce sujet.



Figure 6 : Sentier de la Fouly, Plantes médicinales<sup>6</sup>

## I. Champex-Lac / Orsières

Le sentier de Champex-Lac / Orsières fait 8 kilomètres en descente. Onze postes sont répartis sur le parcours. Une grande partie de la marche se trouve en forêt, il est donc possible d'en apprendre beaucoup sur les champignons. (CREPA, s.d. 2018)

Le sentier comporte également des panneaux didactiques proposés par le Cercle mycologique de l'Entremont. (CREPA, s.d. 2018)

<sup>5</sup> Source : <https://www.charlottelamarmotte.ch/sentiers/a-la-fouly>

<sup>6</sup> Source : Photos personnelles des postes du sentier des plantes médicinales





Figure 7 : Sentier de Champex-Lac / Orsières, Postes<sup>7</sup>

## II. La Creusaz

Le parcours de La Creusaz présente huit postes répartis sur une distance de 4 kilomètres aller-retour. Les postes sont orientés sur la faune et la flore avoisinante. Le sentier se trouvant en altitude, il propose des panoramas grandioses tout en marchant en montagne. Une remontée mécanique est à disposition pour accéder au chemin. (CREPA, s.d. 2018)



Figure 8 : Sentier de la Creusaz, Postes<sup>8</sup>

## III. Bisse du Trient

Le sentier fait 6.5 kilomètres aller-retour et comprend dix postes. Le parcours est facile d'accès car relativement plat et ombragé. Il est possible d'apercevoir le Glacier du Trient dans un paysage alpin. (CREPA, s.d. 2018)

## IV. Emosson

Emosson est le dernier sentier mis à disposition par le CREPA. Le trajet est une boucle de 9 kilomètres. Douze postes sont disponibles sur ce parcours et se concentrent sur la géologie et l'hydroélectricité. (CREPA, s.d. 2018)

<sup>7</sup> Source : <https://www.charlottelamarmotte.ch/sentiers/champex-lac-orsieres>

<sup>8</sup> Source : <https://www.charlottelamarmotte.ch/sentiers/a-la-creusaz>

# **1. Etat de l'art**

## **1.1. Problématique**

Les technologies utilisées pour les sentiers didactiques du CREPA auront certaines contraintes à respecter.

### **1.1.1. Les enfants**

Les technologies proposées ne devront en aucun cas remplacer le format papier mais simplement le compléter, afin que les enfants ne soient pas en permanence sur leur téléphone. Le but étant de faire découvrir la nature et tout ce qui nous entoure en s'amusant. C'est pourquoi la solution proposée devra être légère à l'utilisation et ne pas interagir trop souvent avec les utilisateurs ; un juste milieu devra être trouvé.

### **1.1.2. Disponibilités**

Afin d'être accessible à tous, la solution proposée se doit d'être multiplateforme (Android, IOS), pour que le plus grand nombre puisse l'utiliser. Il est important de ne pas se limiter à un seul système d'exploitation sinon le manque à gagner d'utilisateurs serait trop conséquent. De plus, il faut prendre en compte que les sentiers se situent en dehors des villes ou des villages, c'est pourquoi les réseaux téléphoniques ou internet ne sont pas forcément présents. La solution se doit d'être légère également car les utilisateurs ne veulent pas « attendre » la fin d'un téléchargement avant de partir en ballade. Il est important d'avoir une solution locale et la plus légère possible.

### **1.1.3. Innovation**

De la réalité augmentée est intégrée afin de fusionner le côté ludique avec l'apprentissage de nouvelles connaissances. La réalité augmentée amène des solutions inenvisageables avant son arrivée. Elle va permettre une immersion totale des utilisateurs tout en restant dans le monde « réel ». Elle lie utilité, amusement et connaissance à travers plusieurs formes.

## **1.2. Les solutions existantes sur le marché**

### **1.2.1. Mhikes**

Mhikes est une application permettant aux utilisateurs de découvrir de nouvelles régions et de se reconnecter à la nature. Cette solution pour les randonneurs leur permet d'être guidés et d'en apprendre plus durant leur périple. Mélangeant GPS et contenu interactif, tout y est pour faciliter la vie de l'utilisateur lors de ses balades. (Easy Mountain, s.d. 2018)

#### **I. Création**

« L'aventure débute en 2014 autour d'une rencontre entre un ingénieur et un accompagnateur en moyenne montagne. Conscients des enjeux de la montagne, de la richesse de cet environnement et fins connaisseurs des comportements des randonneurs, ils ont imaginé un outil favorisant la découverte du territoire et ce en toute sécurité. En surfant sur l'explosion du numérique, ils ont développé une application mobile de navigation GPS et de découverte outdoor. Mhikes est né. » (Easy Mountain, 2017)

#### **II. Mission**

L'application a pour but de rendre la découverte urbaine et extérieure plus facile. Elle propose un catalogue de parcours, qui ont été créés par des professionnels du tourisme et de la montagne, ainsi qu'un système de navigation GPS précis et fiable. De plus, tout est échangeable, partageable, à travers la communauté. (Easy Mountain, 2017)

#### **III. Fonctionnalités**

Mhikes propose plus de 2500 parcours téléchargeables à travers l'Europe. Une fois que l'utilisateur a fait son choix, l'application le guide jusqu'au début du périple. Durant celui-ci, des points d'intérêt tels que châteaux, fontaines, parcs, ponts, etc. seront affichés et décrits par l'application. Ces points d'intérêt permettent aux utilisateurs d'en apprendre plus sur ce qui les entoure. Des questions peuvent également être posées durant le parcours afin de rendre les balades plus ludiques pour petits et grands. (Easy Mountain, 2017)

Les parcours sont classés par niveaux de difficulté, par type d'activités ou par critères spécifiques (famille, poussettes, eau, etc.).

Mhikes guide l'utilisateur à l'aide de flèches superposées à des photos du trajet, ce qui permet de ne pas se perdre et de faciliter l'orientation.



Figure 9 : Mhikes, Parcours<sup>9</sup>



Figure 10 : Mhikes, Question<sup>10</sup>

<sup>9</sup> Source : <https://mhikes.com/images/accueil/smartphone-2.png>

<sup>10</sup> Source : <https://www.youtube.com/watch?v=PvM6a5nc0jg>

### 1.2.2. ViewRanger Skyline

Skyline de ViewRanger redéfinit le guidage GPS en intégrant de la réalité augmentée à travers des parcours sur tout le globe ! Pour donner une nouvelle dimension aux cartes qui sont normalement plates, Skyline intègre la réalité augmentée pour leur donner une certaine profondeur. De plus, avec l'intégration d'informations sur les panoramas réels, elle permet d'assouvir la curiosité de certains et de parfaire leurs connaissances. (Augmentera Ltd, s.d. 2018)

#### I. Création

« L'entreprise a été fondée en 2006 suite aux réactions enthousiastes de certains de nos compagnons de randonnée qui ont aimé le logiciel cartographique pour Smartphone que nous avons développé. Il répondait à leur frustration dû aux limites des GPS et aux outils de cartographie disponible au grand public à l'époque. Depuis, la société continue à être reconnue pour son leadership innovant associé à sa passion pour des produits hautement fonctionnels et son service client performant. » (Augmentra Ltd, s.d. 2018)

#### II. Mission

Cette application a pour but de faire explorer le monde aux utilisateurs sous un autre angle. Grâce à la réalité augmentée il suffit de maintenir son téléphone sur le panorama qui se trouve devant soi, afin d'avoir les incrustations des noms des sommets aux alentours ou des lacs à proximité (jusqu'à 32km). Des lieux d'intérêts ont également été créés afin de faire découvrir des nouveaux points aux utilisateurs. (Augmentra Ltd, s.d. 2018)

Skyline a de plus pour mission de faciliter la navigation en plein air. Grâce à la réalité augmentée, l'application apporte une nouvelle dimension aux parcours. Elle permet de superposer des flèches de navigation sur les panoramas à travers l'appareil photo, afin de s'assurer de la bonne direction. (Augmentra Ltd, s.d. 2018)

#### III. Fonctionnalités

L'intégration de la réalité augmentée permet à cette application d'être innovante et de redéfinir les randonnées. Elle se montre pratique, ludique et complète les connaissances des utilisateurs.

Lors d'un périple il suffit de sortir son téléphone, lancer Skyline et choisir sa route. Celle-ci va ouvrir l'appareil photo de du téléphone et y incruster des éléments. C'est ainsi que le panorama devant l'utilisateur va transmettre des informations utiles et agréables. (Augmentra Ltd, s.d. 2018)

Les noms des sommets, des lacs et des lieux aux alentours apparaîtront afin de faire découvrir un autre aspect de la nature qui est à disposition. Des points d'intérêts sont également créés lors des périples afin que chaque détail soit relevé et que tous puissent en bénéficier. On peut également faire une photo directement depuis l'application afin de garder un souvenir du paysage mais aussi des informations incrustées. (Augmentera Ltd, s.d. 2018)

Skyline permet notamment de se déplacer aisément à travers les parcours, grâce à son système de guidage unique ; le sens de l'orientation est moins sollicité. Il suffit de suivre les flèches qui sont superposées au panorama. L'application propose un catalogue de 120'000 routes. (Augmentera Ltd, s.d. 2018)



Figure 11 : ViewRanger, Skyline<sup>11</sup>

L'application permet également de créer ses propres routes afin de les partager à la communauté. Skyline est disponible en mode hors ligne car les parcours étant parfois reculés, il est fort probable de ne pas avoir de couverture réseaux. (Augmentera Ltd, s.d. 2018)

<sup>11</sup> Source : <http://www.viewranger.com/fr/skyline/la-navigation-en-plein-air-reinventee>

### **1.2.3. Pokémon Go**

Pokémon Go est une application développée par Niantic. Elle intègre le gros succès Pokémon dans des smartphones. Cette solution ludique permet aux utilisateurs d'attraper des Pokémon tout en marchant. Grâce à la réalité augmentée et au GPS, ces jolies créatures apparaissent dans notre monde, ce qui rend le jeu encore plus ludique et plus amusant. Il permet de fusionner balade, connaissances et jeux vidéo.

#### **I. Création**

Le 1er avril 2014 The Pokemon Company et Google s'unissent pour faire un poisson d'avril dans lequel il faut chasser des Pokémon sur Google Map. (Agitateur, 2016)

La société Niantic, filiale de Google jusqu'en 2015 et créatrice du jeu « Ingress », va s'intéresser au concept de capturer des Pokémon en réalité augmentée. Niantic et The Pokémon Company vont s'unir afin de créer PokémonGo. La base du jeu « Ingress », qui ressemble énormément à la vision du chef de The Pokémon Company, a été réutilisée afin de créer cette nouvelle application qui a vu le jour en 2016. (Agitateur, 2016)

#### **II. Mission**

Pokémon Go veut faire passer les Pokémon du monde virtuel au monde réel avec, pour ce faire, la réalité augmentée. L'application permet aux utilisateurs de chasser des Pokémon dans le monde entier. Tous les Pokémon sont présents dans leur habitat respectif : eau, terre, vent, feu. L'utilisateur doit se balader et trouver tous les Pokémon du jeu. (Pokémon. TM, Nintendo, s.d. 2018)

Pour chasser des Pokémon il suffit de marcher dans son environnement quotidien. Le téléphone vibrera lors de l'apparition d'un Pokémon. Il suffit d'ouvrir l'application et de viser le Pokémon et de le capturer à l'aide d'une Pokéball ! (Pokémon. TM, Nintendo, s.d. 2018)

Afin d'obtenir les dites Pokéballs il faut se rendre proche de lieux stratégiques, tels que des structures artistiques exposées dans des lieux publics, des sites et monuments historiques. (Pokémon. TM, Nintendo, s.d. 2018)

#### **III. Fonctionnalités**

Pokémon Go permet de capturer des Pokémon virtuels dans notre monde réel. Pour ce faire, énormément d'outils sont mis à disposition. Grâce au GPS, des Pokémon apparaissent à nos côtés lors de nos déplacements. Les utilisateurs ont la possibilité de faire des photos d'eux et de les capturer. (Pokémon. TM, Nintendo, s.d. 2018)

Afin de le capturer, il est nécessaire d'être en possession de Pokéballs. Ces-dernières sont disponibles dans des Pokéstops qui se situent dans des lieux publics tels que des églises, des places, des parcs, proches des statues, etc. Là encore, la réalité virtuelle et le monde réel ne font qu'un. (Pokémon. TM, Nintendo, s.d. 2018)

Une fois que l'utilisateur a des Pokéballs, il peut les lancer sur les Pokémon lors de leur apparition avec une chance de les capturer. Le but du jeu étant de capturer tous les Pokémon différents. (Pokémon. TM, Nintendo, s.d. 2018)

Il existe plusieurs façons d'avoir des Pokémon différents (Pokémon. TM, Nintendo, s.d. 2018) :

- Les œufs : pour les faire éclore, il faut marcher une certaine distance (5, 10, 20 kilomètres).
- L'évolution : si vous possédez un Pokémon d'une même espèce assez de fois, vous pouvez le faire évoluer.
- L'environnement : les Pokémon de type eau auront tendances à apparaître proches des lacs et des océans, tandis que les types terre seront plus présents sur les montagnes. L'utilisateur peut alors trouver des Pokémon différents lors de ses voyages.
- La météo : les Pokémon de type vol apparaitront par temps de vent, alors que ceux de type feu lorsqu'il fait beau. Les utilisateurs ne doivent pas hésiter à sortir de chez eux par tout temps pour trouver des créatures rares.



Figure 12 : PokémonGo, Map<sup>12</sup>

<sup>12</sup> Source : Screenshot personnel de l'application sur mon téléphone





Figure 13 : PokémonGo, Réalité augmentée<sup>13</sup>



Figure 14 : PokémonGo, Informations<sup>14</sup>

<sup>13</sup> Source : Screenshot personnel de l'application sur mon téléphone

<sup>14</sup> Source : Screenshot personnel de l'application sur mon téléphone



Figure 15 : PokémonGo, Météo<sup>15</sup>

<sup>15</sup> Source : Screenshot personnel de l'application sur mon téléphone

### **1.3. Discussion de l'état de l'art**

Ces trois applications apportent des connaissances sur des technologies différentes les unes des autres, cependant aucune n'a toutes les spécificités que nécessite celle des sentiers ludiques. Pour créer notre solution, il nous faut respecter toutes les contraintes que le CREPA a définies. Voici un résumé des applications analysées et ce qu'elles apportent aux utilisateurs.

#### **1.3.1. Mhikes**

L'application Mhikes a été sélectionnée car elle utilise grandement le GPS durant les parcours. En effet, elle comprend énormément de contenu sur les objets avoisinants et vise à instruire l'utilisateur. De plus, des questions peuvent être posées sur certains parcours ce qui ressemble grandement aux désirs du CREPA. Cependant il faut télécharger les parcours à l'avance et ils peuvent être plus ou moins lourds. En effet, les trajets peuvent contenir énormément de points d'intérêts qui sont des images, des vidéos et des contenus audios. Le fait de devoir télécharger un parcours peut donc prendre du temps et de la place. Une fois le trajet téléchargé la connexion internet n'est plus nécessaire ce qui reste un plus. Cette application ne contient malheureusement pas de réalité augmentée.

#### **1.3.2. ViewRanger Skyline**

ViewRanger Skyline a été choisie pour sa grande utilisation de la réalité augmentée et du GPS. En effet, tout ce qui se fait sur Skyline est en réalité augmentée et permet une immersion totale des utilisateurs dans l'environnement avoisinant. Le système de guidage correspond parfaitement au souhait du CREPA de pouvoir trouver le début des sentiers en réalité augmentée. De plus, l'ajout d'informations sur le panorama permet aux utilisateurs curieux d'en apprendre plus et de découvrir de nouveaux endroits. Cependant, l'application est faite pour des utilisateurs adultes ou jeunes adultes car assez complexe et avec du contenu payant. De plus, aucun « jeu » n'est proposé ; l'application apporte simplement des connaissances supplémentaires. La possibilité de pouvoir créer son parcours est intéressante, mais inutile pour le projet « Charlotte la Marmotte ». De plus, comme pour Mhikes, il faut télécharger son parcours à l'avance (qui peut être payant).

#### **1.3.3. PokémonGo**

PokémonGo est l'inverse des applications sélectionnées plus haut. C'est un jeu qui permet à ses utilisateurs d'agir à la fois dans le monde réel et dans le monde virtuel. Il n'a pas pour but d'apprendre ou de développer des connaissances, mais simplement de divertir ses utilisateurs. Il a été choisi car il est l'un des jeux qui à faire connaître la réalité augmentée. Il utilise des notifications GPS, ce qui est en accord avec les envies du CREPA de ne pas toujours avoir son téléphone dans la main. En effet, la notification permet d'utiliser son téléphone ou de le regarder seulement à un endroit du parcours.

De plus il sert d'inspiration pour des jeux en réalité augmentée que nous retrouvons dans « Charlotte la Marmotte », tel que faire apparaître un animal.

#### 1.3.4. Tableau récapitulatif

Voici un tableau récapitulatif de tous les points que les applications respectent ou apportent.

	Mhikes	ViewRanger Skyline	PokémonGo	Charlotte la Marmotte
GPS	X	X	X	X
GPS, Notification			X	X
GPS, AR		X		X
Jeux	X		X	X
Jeux, Questions	X			X
Jeux, AR			X	X
Apport de connaissances	X	X		X
Multiplateforme	X	X	X	X
Découverte de la nature	X	X	X	X
Utilisation simple			X	X
Accessible hors réseaux	Oui, mais téléchargement des parcours en avance	Oui, mais avec téléchargement des parcours en avance	Oui, mais pas de jeu en ligne	Oui
Légèreté d'utilisation			X	X

Tableau 1 : Etat de l'art, Tableau récapitulatif<sup>16</sup>

<sup>16</sup> Source : Tableau personnel regroupant les informations de l'état de l'art de ce document

## 2. Technologies

Le CREPA souhaite que la solution soit multiplateforme, ce qui signifie être disponible sur plusieurs systèmes d'exploitation tel que Android et IOS. De la réalité augmentée doit également être intégrée. C'est pourquoi nous nous sommes tournés vers deux technologies connues : Unity et Xamarin.

Durant ce travail, Xamarin a été étudié et testé car une ébauche du projet (sans réalité augmentée) avait été développée sur cette technologie par Mme Camille Pellaton. Plus d'informations sont disponibles dans la section « Prototype de l'application proposé par Data Sementics Lab ». Unity a été approché car c'est l'outil le plus utilisé pour implémenter de la réalité augmentée en multiplateforme.

Pour comprendre le fonctionnement de la réalité augmentée, il faut étudier les propositions faites par Android (ARCore) et IOS (ARKit), qui sont ensuite réutilisées dans Unity, Xamarin ou d'autres programmes.

### 2.1. ARCore & ARKit

ARCore et ARKit sont des plateformes permettant le développement d'applications en réalité augmentée sur les systèmes d'exploitation de Android et IOS. Grâce à ces plateformes, les développeurs peuvent ajouter des fonctionnalités à leurs applications, telles que le suivi de mouvements et l'évaluation de la taille ou de la position des objets qui les entourent. ARCore et ARKit permettent aux utilisateurs une sensation plus réaliste dans tous types d'environnements.

#### 2.1.1. ARCore

ARCore est la plateforme de Google pour intégrer la réalité virtuelle dans des applications Android principalement. ARCore permet à votre téléphone de comprendre le monde qui l'entoure et d'interagir avec les informations reçues de celui-ci. Trois capacités sont utilisées pour intégrer du contenu virtuel au monde réel : le suivi de mouvement, la compréhension de l'environnement et l'estimation de la lumière. (Google, 2018)

Le suivi de mouvement permet au téléphone de comprendre et de suivre sa propre position. ARCore utilise un processus appelé odométrie et cartographie simultanée, pour comprendre où se situe le téléphone par rapport au monde réel. (Google, 2018)

La compréhension de l'environnement donne la capacité au téléphone de détecter la taille et l'emplacement des surfaces qui l'entourent (surfaces horizontales, verticales et inclinées, telles que le sol, les tables ou les murs). (Google, 2018)

L'estimation de la lumière va permettre au téléphone d'estimer les conditions d'éclairage actuelles de son environnement. Grâce à cette capacité, ARCore va pouvoir fournir la correction moyenne d'intensité et de couleur d'une image. Les objets virtuels pourront être éclairés dans les mêmes conditions que les objets réels. (Google, 2018)

### 2.1.2. ARKit

ARKit est la plateforme d'Apple pour intégrer de la réalité augmentée au sein des applications IOS. Elle permet de mélanger les objets numériques et l'information avec l'environnement qui nous entoure (Apple Inc., s.d. 2018). ARKit est le ARCore d'Apple. Trois caractéristiques sont importantes pour cette solution :

#### I. La caméra

Elle va permettre un suivi de visage robuste dans les applications de réalité augmentée. L'application arrive à détecter la position, la topologie et l'expression du visage de l'utilisateur. (Apple Inc., s.d. 2018)

#### II. L'odométrie visuelle inertielle

ARKit utilise l'odométrie visuelle inertielle pour suivre avec précision le monde qui l'entoure. Cet outil fusionne les données des capteurs de caméra et les données de mouvements de base. Ces deux fonctions permettent de suivre le déplacement de l'appareil à travers l'espace. (Apple Inc., s.d. 2018)

#### III. Compréhension de la scène et estimation de l'éclairage

Cet outil utilise les mêmes technologies que ARCore.

Des optimisations pour ARCore et ARKit sont disponibles sur des logiciels tels que Unity ou Unreal Engine.

### 2.2. Xamarin

Xamarin est une technologie fournissant des outils de développement d'applications mobiles multiplateformes. Elle est née d'une compagnie créée en 2011 aux Etats-Unis qui se fait racheter en 2016 par Microsoft. Cette technologie permet la création d'applications Android, IOS ou Windows Phone natives à leur système d'exploitation, avec toutes les particularités de celui-ci. Le développement de telles applications se fait donc en une fois, ce qui constitue un gain de temps non-négligeable. (LavishT@TWC, 2017)

Les outils Xamarin sont une extension de Visual Studio, qui est un IDE pour le .Net et C#. Les développeurs utilisent du C# pour coder les applications, ce qui est un avantage car ils n'ont pas besoin d'apprendre du JAVA, Objective-C ou Swift pour construire leur application ; seul un langage

est nécessaire. En effet, pour développer une application Android, il nous faut savoir le JAVA, pour une application Apple (IOS) c'est le Swift qui est requis et pour Windows Phone il nous faut connaître de l'Objective-C. Xamarin nous permet de développer sur ces trois plateformes en connaissant seulement un langage. L'outil offre un gain de temps et de ressources considérable. (LavishT@TWC, 2017)

Ci-dessous, un graphique qui nous permet de comprendre le fonctionnement de Xamarin. La partie User Interface Code est géré par ce qu'on appelle des Xamarin.Forms et la partie Logic est également partagée pour toutes les plateformes.

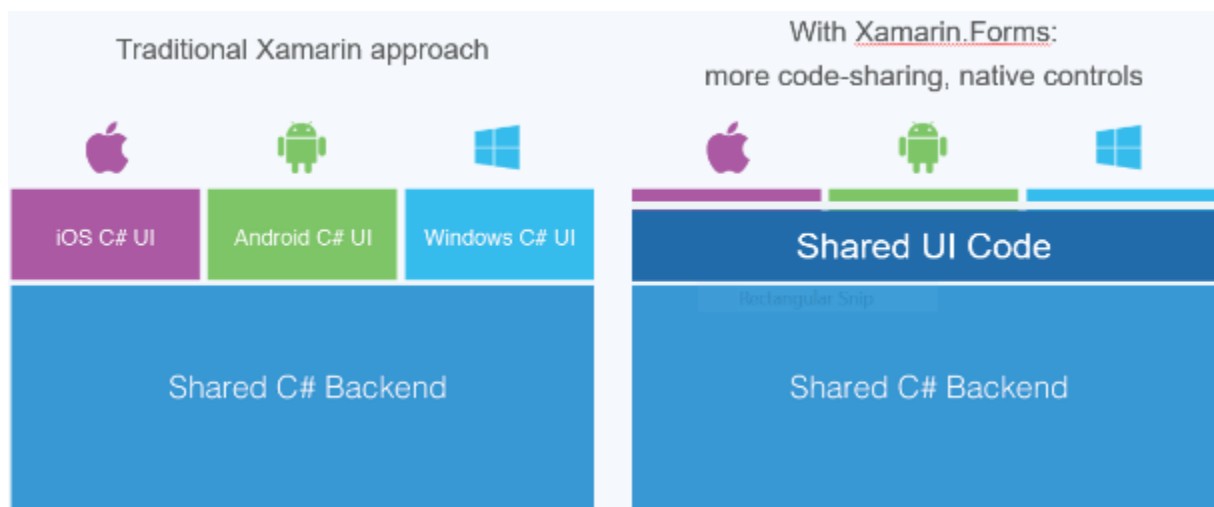


Figure 16 : Xamarin, Architecture<sup>17</sup>

### 2.2.1. Réalité augmentée

Xamarin permet l'intégration de la réalité augmentée en utilisant ARCore et ARKit. Cependant ces deux technologies sont en JAVA et en native IOS, ce qui les rend difficilement utilisable en C#. Durant les recherches, seulement des solutions natives ont été trouvées afin d'intégrer ARCore ou ARKit dans Xamarin. Ce qui rend le cross plateforme de Xamarin dans ce genre de situations moins optimal.

En Xamarin, la réalité augmentée doit actuellement être codée nativement, ce qui nécessite de développer deux sections : une pour la partie Android et une autre pour IOS. Pour le moment, il n'existe pas de lien entre les sections, les deux doivent être implémentées séparément.

<sup>17</sup> Source : <https://www.sitepoint.com/build-cross-platform-android-ios-uis-xamarin-forms/>

## 2.2.2. Prototype de l'application proposé par Data Semantics Lab

Avant ce travail de Bachelor, le CREPA avait déjà demandé à Data Semantics Lab qui est au sein de l'institut d'informatique de gestion, de fournir une esquisse de l'application de Charlotte la Marmotte. Cependant, bien qu'elle dût être multiplateforme, elle ne comprenait pas de réalité augmentée. Ce prototype créé par Mme Camille Pellaton a été implémenté avec Xamarin. C'est pourquoi, les possibilités qu'offre cet outil ont été analysées.

Le code a été revu et de brèves tentatives d'intégration de réalité augmentée ont été réalisées, malheureusement sans succès. Xamarin ne propose actuellement que des solutions natives pour le codage de la réalité augmentée, ce qui rend le multiplateforme plus compliqué. C'est pourquoi, un autre outil connu a été analysé.

Voici des images du prototype proposé par Mme Camille Pellaton avec Xamarin :

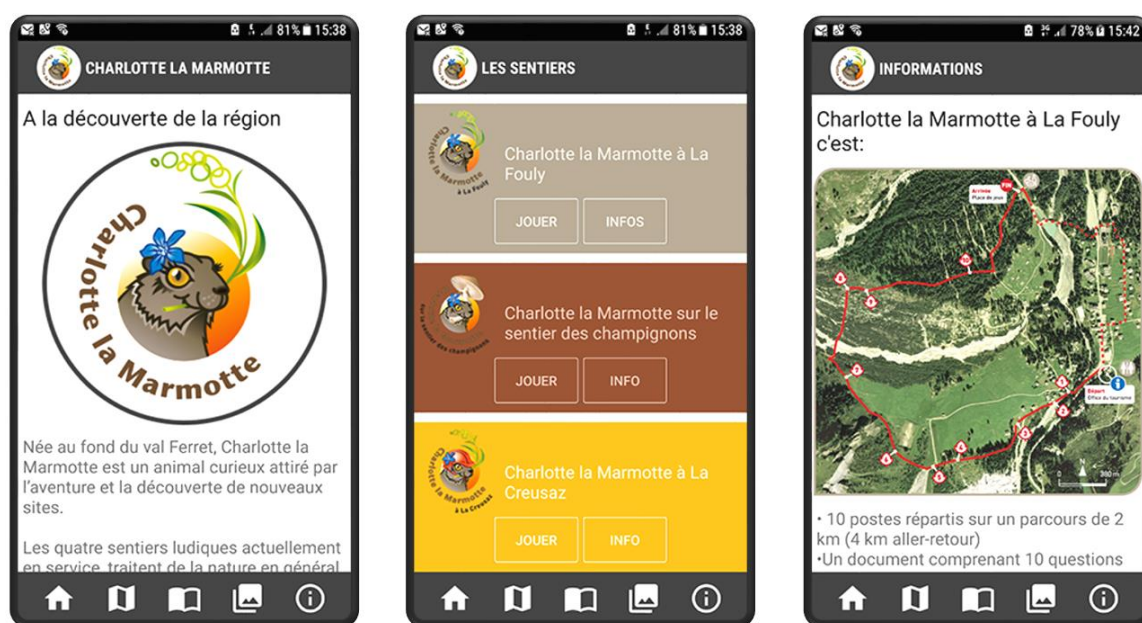


Figure 17 : Prototype Xamarin, Vues<sup>18</sup>

<sup>18</sup> Source : Image fournie par Camille Pellaton



## 2.3. Unity

Unity est la plateforme numéro une dans le monde du jeu vidéo, étant la plateforme de développement la plus populaire. Unity est en pleine croissance et énormément de développeurs se tournent vers celle-ci pour créer leurs applications. Parmi ses clients, on compte : Cartoon Network, Coca-Cola, Disney, Electronic Arts, LEGO, etc. (Unity Technologies, s.d. 2018)

Cette plateforme permet de créer des jeux vidéo et des applications en 2D, 3D, RV (réalité virtuelle) et RA (réalité augmentée). De plus, tout le contenu de Unity est multiplateforme, la réalité augmentée également. (Unity Technologies, s.d. 2018)

Unity offre aussi des tutoriel gratuits, efficaces pour les débutants ou les plus avancés. Des formations en ligne ou en direct sont disponibles sur la plateforme. De plus, un catalogue « d'assets », des objets 3D indispensables pour les jeux, est disponible dans le programme ; certains sont gratuits et d'autres payants. Les assets sont téléchargeables et intégrables directement dans Unity, ce qui ne nécessite aucune transformation de la part du développeur. (Unity Technologies, s.d. 2018)

Les tutoriels et les Assets intégrés sont un énorme plus pour la plateforme, car cela permet un gain de temps inimaginable pour les développeurs. En effet, pas besoin de créer ses propres assets, de « perdre » du temps à transformer certains formats ou de passer par un logiciel tiers pour avoir la capacité d'utiliser des objets 3D.

Le plateforme Unity fonctionne avec un éditeur de scènes graphiques. Les scènes sont des objets qui contiennent l'environnement et les menus de notre application. Chaque scène est un niveau à travers notre solution. Chacune est différente et peut contenir : des menus, des objets 3D, de la réalité augmentée, du texte, etc.

La partie logique de Unity se fait sur Visual Studio en C#. Les scripts sont créés sur la partie graphique de Unity et sont éditables sur Visual Studio.

### 2.3.1. Réalité augmentée

Pour faire de la réalité augmentée au sein de Unity, deux choix sont possibles : utiliser ARCore et ARKit ou utiliser Vuforia.

#### I. ARCore & ARKit

Comme expliqué ci-dessus, ARCore et ARKit ouvrent la voie à de nouvelles catégories d'applications et de jeux, par la compréhension de l'environnement, le suivi de mouvement et l'estimation de la lumière. Elles vont permettre à un iPhone ou un appareil Android de reconnaître des surfaces et d'y implémenter des objets.

## II. Vuforia

Vuforia révolutionne le développement de la réalité augmentée. Cette plateforme permet de créer des expériences plus immersives et de toucher un plus vaste public. Elle permet de déployer son projet sur de nombreux appareils, tels que smartphones, casques et UWP (Universal Windows Platform). Pour se faire, plusieurs caractéristiques sont nécessaires. (Unity Technologies, s.d. 2018)

Un des grands avantages de Vuforia est sa capacité à faire une seule application RA pour tout type de support. Le développeur peut se consacrer à une seule application qui sera retranscrite pour IOS ou Android par exemple. La combinaison de Unity et Vuforia facilite grandement la construction d'applications contenant de la réalité augmentée. (PTC Inc., s.d. 2018)

Une autre fonctionnalité est disponible avec Vuforia : le Ground Plane. Elle permet de placer du contenu numérique sur des surfaces horizontales de l'environnement, tel que les sols ou les tables. Il permet également d'afficher ce contenu dans les airs grâce à des points d'ancrage. (PTC Inc, s.d. 2018)



Figure 18 : Vuforia, Ground Plane<sup>19</sup>

Vuforia permet également d'utiliser plusieurs types de « Target ». Le Target est l'objet à viser pour faire apparaître la réalité augmentée. En voici une liste :

**Model Targets :** Ce Target permet de reconnaître les objets en fonction de leur géométrie, à l'aide d'objets 3D existants. Il donne la possibilité de placer de la RA sur des articles comme les véhicules ou les jouets. (PTC Inc, s.d. 2018)

<sup>19</sup> Source : <https://library.vuforia.com/articles/Training/ground-plane-guide.html>



Figure 19 : Vuforia, Model Target<sup>20</sup>

**Image Targets :** C'est la façon la plus facile d'ajouter du contenu de réalité augmentée sur un objet plat, tel qu'un logo ou un magazine. (PTC Inc, s.d. 2018)



Figure 20 : Vuforia, Image Target<sup>21</sup>

**Multi Targets :** Ils sont utilisés pour des objets avec des surfaces planes mais qui ont plusieurs côtés ou qui ont des images multiples, telles que les posters ou l'emballage des produits. (PTC Inc, s.d. 2018)

**Cylinder Targets :** Ce Target permet d'ajouter de la réalité augmentée sur des surface en forme de cylindre, comme des cannettes ou des bouteilles. (PTC Inc, s.d. 2018)



Figure 21 : Vuforia, Cylinder Target<sup>22</sup>

<sup>20</sup> Source : <https://unity3d.com/profiles/unity3d/themes/unity/images/pages/partners/vuforia/model.jpg>

<sup>21</sup> Source : <https://unity3d.com/profiles/unity3d/themes/unity/images/pages/partners/vuforia/image.jpg>

<sup>22</sup> Source : <https://bit.ly/2xFGz9T>

**Object Targets :** Ceux-ci sont créés avec la numérisation d'objets plus complexes et plus riches en détails, tels que des jouets ou des peluches. (PTC Inc, s.d. 2018)



Figure 22 : Vuforia, Object Target<sup>23</sup>

Vuforia permet également de gérer les bases de données des Targets en local, sur l'appareil ou sur un cloud. Une section développeur est disponible sur leur site afin de faciliter l'accès et de regrouper les informations.

#### 2.4. Choix technologiques

Après étude des différentes technologies et discussion avec des personnes connaissant la réalité augmentée, dont mon professeur M. Antoine Widmer, il en résulte ceci :

Xamarin est une technologie performante pour les applications multiplateformes mais permet une intégration de la réalité augmentée plus difficile ou du moins qui prend plus de temps. De plus, la gestion des objets 3D n'est pas intégrée à Xamarin et il faudrait utiliser des logiciels tiers pour les rendre accessibles.

Unity est fait pour la réalité augmentée et le multiplateforme ce qui facilite grandement l'intégration de celle-ci au sein de l'application. L'application sera orientée vers le jeu et moins vers une application d'information. La gestion des menus est plus compliquée et le « material design » difficilement intégrable. Grâce à son « Asset Store », l'acquisition d'objets 3D et l'intégration de ceux-ci est grandement simplifiée !

Nous avons donc choisi la combinaison Unity - Vuforia pour la création de ce projet. Ce qui permet une grande maniabilité dans le multiplateforme.

<sup>23</sup> Source : <https://unity3d.com/profiles/unity3d/themes/unity/images/pages/partners/vuforia/object.jpg>

### 3. Développement de l'outil

Le prototype proposé en Xamarin a donc été mis de côté et tout a été recommencé sur Unity. Les idées ont bien entendu été gardées. L'application a été créée sur Unity, toute la partie visuelle se trouve sur ce programme et toute la partie logique se situe sur Visual Studio. Tous les éléments sont accessibles depuis Unity.

Un guide technique pour la configuration de Vuforia est disponible en annexe. Il permet une meilleure compréhension du développement ainsi qu'une approche plus tutoriel.

Termes techniques :

**Les Scènes** contiennent les environnements et les menus de l'application. Une scène contient la logique qui est stockée sous forme de script en C# et la vue qui est sous forme d'Asset dans l'éditeur Unity. Les scènes sont les objets principaux de Unity. (Unity Technologies, 2018)

**Les Canvas** sont des zones qui contiennent tous les éléments de l'interface. Ces éléments tels que les boutons, les textes, les images etc. doivent être des enfants d'un Canvas. Les Canvas facilitent le placement de ces différents éléments au sein de leurs zones. (Unity Technologies, 2018)

**Les Panels** sont des panneaux auxquels il est possible de modifier la taille, le fond, la transparence, leur contenant etc. Ils sont modulables à souhait pour permettre une souplesse à l'interface.

**Les Scripts** sont des fichiers en C# qui contiennent tout l'aspect logique.

**Les Loops (boucles)** permettent de répéter une suite d'instructions, tant qu'une condition n'est pas respectée. Il en existe plusieurs types tel que la boucle « for » qui est la plus utilisée dans Charlotte la Marmotte.

**Les Quaternions** permettent de représenter l'orientation ou la rotation au sein de Unity. Cette représentation interne se compose de quatre nombres référencés dans Unity comme x, y, z et w. (Unity Technologies, 2018)

**Les Préfabriqués** sont des modèles d'objets. Afin d'éviter de dupliquer des objets qui seront modifiables indépendamment, Unity donne la possibilité de créer son propre modèle d'objets. Le préfabriqué (« prefabs » dans Unity) agit comme un modèle à partir duquel il est possible de créer de nouvelles instances d'objets dans la scène. Toutes les modifications apportées à un bien préfabriqué sont immédiatement reflétées dans toutes les instances produites à partir de celui-ci, mais Il est également possible de remplacer les composants et les paramètres de chaque instance individuellement. (Unity Technologies, 2018)

L'**Object pooling** permet de réutiliser des objets déjà créés à la place de les détruire et de les recréer. Ce script permet de pré-instancier tous les objets voulus avant le démarrage du jeu. (Geig, s.d. 2018)

Les **Image Targets** représente des images que Vuforia peut détecter et suivre. Vuforia va analyser la cible est la comparer à la base de données liée à notre projet, afin d'y faire l'action qui lui est attribuée. Grâce à ce procédé, il est possible à travers l'appareil photo de notre téléphone de déclencher une action, telle que l'affichage d'un animal sur un panneau. Le fonctionnement est le même que les QR codes cependant, les images Targets n'ont pas besoin de régions ou de codes en noir et blanc spéciaux pour être reconnues. (PTC Inc, s.d. 2018)

Le **Bearing** est la direction de la boussole depuis la position actuelle jusqu'à la destination voulue.

Le **Slerp** est l'interpolation sphérique entre un point a et un point b. (Unity Technologies, 2018)

Le **Lerp** est aussi l'interpolation sphérique entre un point a et un point b. Cependant l'interpolation est plus rapide ce qui peut donner un moins bon effet sur des objets éloigné l'un de l'autre. (Unity Technologies, 2018)

Les **Frame** représentent la mise à jour d'un écran.

La **Coroutine** est une fonction qui a la capacité de mettre en pause l'exécution et de renvoyer le contrôle à Unity. (Unity Technologies, 2018)

### 3.1. Architecture interne

Voici les différents éléments liés à l'architecture interne :

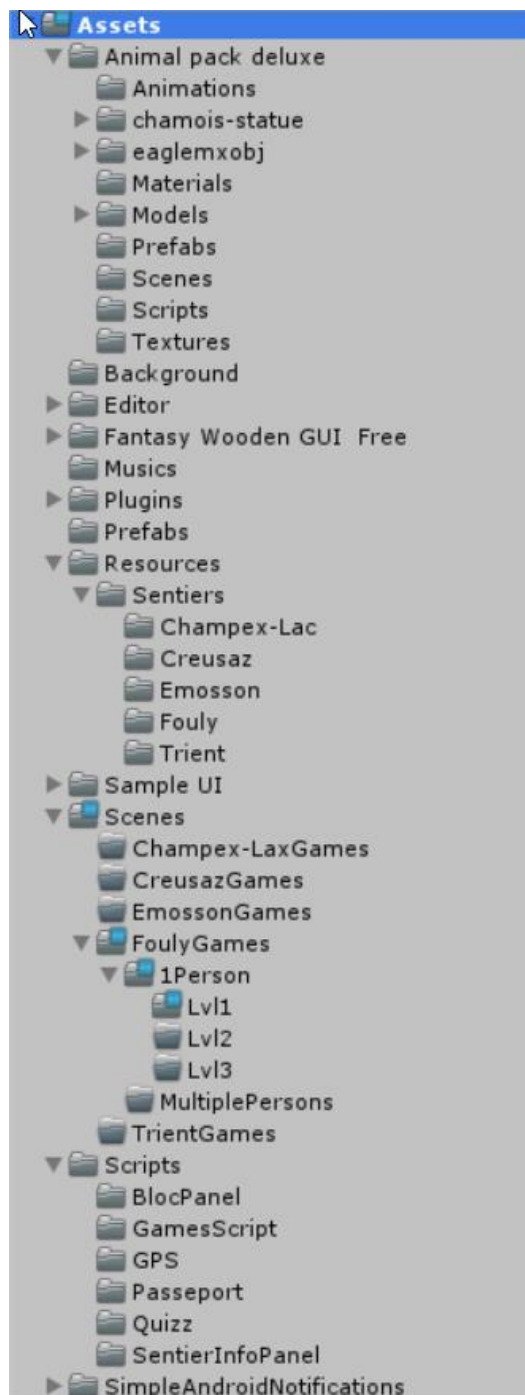


Figure 23 : Architecture du projet Unity<sup>24</sup>

<sup>24</sup> Source : Screenshot personnel de l'architecture du projet Unity CharlotteLaMarmotte

**Assets** : Les Assets sont tous les objets utilisés au sein de l'application

**Animal Pack Delux** : Ce pack contient tous les animaux 3D afin de les incruster en réalité augmentée. Ces assets ont été téléchargés depuis l'Asset store de Unity.

**Background** : Ce dossier contient les fonds d'écrans et les styles d'écritures (fontes) de l'application.

**Editor** : Il contient la base de données de Vuforia avec les images que l'application doit reconnaître afin de pouvoir y mettre une action.

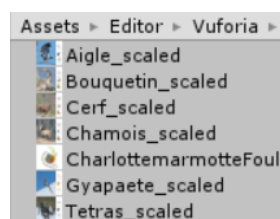


Figure 24 : Architecture, Vuforia Targets<sup>25</sup>

**Fantasy Wooden GUI Free** : Ceci est un Asset téléchargé depuis l'Asset Store également. Il contient une grande partie des objets servant à l'interface de l'application. D'autres éléments y ont été ajoutés afin que tout le GUI se trouve au même endroit.

**Musics** : Ce dossier contient les musiques de la solution.

**Plugin** : Les plugins s'y trouvant sont ceux de Vuforia afin de permettre le multiplateforme de la solution. On y trouve les plugins Android et IOS par exemple.

**Prefabs** : Les objets préfabriqués qui peuvent être réutilisés tel que les boutons. Les prefabs sont utilisés par « l'object pooling » qui sera décrit dans la suite du document.

**Ressources** : Comprend toutes les informations relatives aux sentiers telles que les images ou les descriptions. On y trouve également la configuration de Vuforia.

**Scenes** : Inclus toutes les scènes de notre application. Les scènes principales telles que le menu se trouvent à la racine. Les scènes des jeux de sentiers sont enregistrées dans une hiérarchie qui correspond aux Sentiers -> Nombre de joueurs -> Niveaux de difficultés.

**Scripts** : Contient tous les scripts en C# de l'application. Ils sont classés par jeux et utilités.

**Simple Android Notification** : Asset pour tester les notifications Android.

<sup>25</sup> Screenshot personnel de l'architecture du projet Unity CharlotteLaMarmotte



### 3.2. Persistent

La scène « Persistent » contient tous les éléments en persistance. Elle va être utilisée pour stocker des informations telles que le nombre de jeux dans les sentiers ou les données pour le quizz. Elle permet d'initialiser des données et de lancer la scène « MainMenu ». Cette scène est liée au script « DataController » :

```
public class DataController : MonoBehaviour
{
    //Permet de remplir les données pour le quizz
    public RoundData[] allRoundData;

    // Use this for initialization
    void Start()
    {
        //DontDestroyOnLoad permet de garder les objets apres la fermeture et l'ouverture d'une autre scene
        DontDestroyOnLoad(gameObject);
        //Permet de définir le nombre de jeux dans un sentier
        populateGames();
        //Appelle la scene MainMenu
        SceneManager.LoadScene("MainMenu");
    }
}
```

Figure 25 : « Persistent » Dev. Script DataController<sup>26</sup>

```
//Get RoundData
public RoundData GetCurrentRoundData()
{
    return allRoundData[0];
}

void populateGames()
{
    //Fouly Games 1

    for (int i = 1; i <= 4; i++)
    {
        Game game = new Game(i , false);
        SaveLoad.savedFoulyGames.Add(game);
        //SaveLoad.Save();
    }
}
```

Figure 26 : Persistent, Dev. Script DataController, Méthodes GetCurrent. et PopulatesGames<sup>27</sup>

<sup>26</sup> Source : Screenshot personnel de la classe DataController

<sup>27</sup> Source : Screenshot personnel des méthodes GetCurrentRoundData et PopulatesGames

La classe « RoundData » permet le stockage des informations pour le Quizz, elle se compose ainsi :

```
[System.Serializable]
public class RoundData {

    public string RoundName;
    public int timeLimitInSeconds;
    public int pointsAddedForCorrectAnswer;
    public QuestionData[] questions;
}
```

Figure 27 : Persistent, Dev. Script RoundData<sup>28</sup>

```
[System.Serializable]
public class QuestionData {

    public string questionText;
    public AnswerData[] answers;
}
```

Figure 28 : Persistent, Dev. Script QuestionData<sup>29</sup>

```
[System.Serializable]
public class AnswerData {

    public string answerText;
    public bool isCorrect;
}
```

Figure 29 : Persistent, Dev. Script AnswerData<sup>30</sup>

<sup>28</sup> Source : Screenshot personnel de la classe RoundData

<sup>29</sup> Source : Screenshot personnel de la classe QuestionData

<sup>30</sup> Source : Screenshot personnel de la classe AnswerData

La classe « Game » contient les informations sur un jeu.

La classe « SaveLoad » permet de sauver et charger les informations relatives aux jeux de l'application afin de savoir l'avancée de l'utilisateur sur ceux-ci :

```
public static class SaveLoad
{
    public static List<Game> savedFoulyGames = new List<Game>();

    //it's static so we can call it from anywhere
    public static void Save()
    {
        //SaveLoad.savedFoulyGames.Add(Game.current);
        BinaryFormatter bf = new BinaryFormatter();
        FileStream file = File.Create(Application.persistentDataPath + "/savedGames.gd"); //you can call it anything you want
        bf.Serialize(file, SaveLoad.savedFoulyGames);
        file.Close();
    }

    public static void Load()
    {
        if (File.Exists(Application.persistentDataPath + "/savedGames.gd"))
        {
            BinaryFormatter bf = new BinaryFormatter();
            FileStream file = File.Open(Application.persistentDataPath + "/savedGames.gd", FileMode.Open);
            SaveLoad.savedFoulyGames = (List<Game>)bf.Deserialize(file);
            file.Close();
        }
    }
}
```

Figure 30 : Persistent, Dev. Script SaveLoad<sup>31</sup>

La méthode « save » permet de sauver la liste « savedFoulyGames » dans un fichier, tandis que la « load » permet de charger le fichier dans cette même liste.

### 3.3. Main Menu

Cette scène est le menu de toute l'application. Elle gère toute la navigation à travers les différents Panels et donne accès à toutes les fonctionnalités. « MainMenu » se compose ainsi :

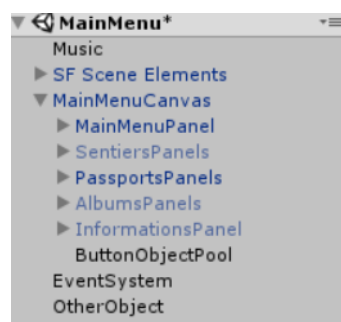


Figure 31 : MainMenu, Architecture<sup>32</sup>

<sup>31</sup> Source : Screenshot personnel de la classe SaveLoad

<sup>32</sup> Source : Screenshot personnel de l'architecture de la scène MainMenu

**Music** : Contient la musique choisie dans une Loop.

**SF Scene Elements** : Se compose du fond d'écran, d'une nuée de particules en mouvement et de la caméra.

**OtherObject** : Objet qui regroupe des fonctionnalités qui doivent être appelées lors du démarrage de la scène. On y trouve notamment la méthode « SaveLoad.Load » afin de charger les données du fichier « savedGames.gd ».

C'est également dans le OtherObject que l'affichage du Canvas est en partie gérée, car les Panels ne seront pas détruits lors du lancement d'un jeu. Si c'était le cas, à sa fermeture, le menu principal serait chargé et non pas le panel courant. Pour pallier ce problème, le Canvas sera simplement caché et réactivé lors du retour sur le menu principal.

```
public class ShowCanvasDontDestroyed : MonoBehaviour {  
  
    // Use this for initialization  
    void Start () {  
        Canvas canvas = GameObject.Find("MainMenuCanvas").GetComponent<Canvas>();  
        canvas.enabled = true;  
    }  
}
```

Figure 32 : MainMenu, OtherObject, Dev. Script ShowCanvasDontDestroyed<sup>33</sup>

### 3.3.1. MainMenuCanvas

Le « MainMenuCanvas » est un Canvas contenant cinq Panels :

#### 1. MainMenuPanel

On y trouve les quatre boutons pour la navigation à travers l'application. Chacun de ces boutons active ou désactive des Panels différents.

<sup>33</sup> Source : Screenshot personnel de la classe ShowCanvasDontDestroyed



Figure 33 : MainMenuPanel, Vue<sup>34</sup>

La navigation se fait avec des méthodes intégrées au bouton. Le bouton sentier permet d'accéder aux informations et aux jeux des sentiers. Le passeport permet de voir si l'on a validé tous les jeux d'un sentier. Photos contient l'albums photos de notre application et Informations comprend le « à propos » et les liens vers le Facebook du CREPA ainsi que le lien pour le site officiel de Charlotte la Marmotte.

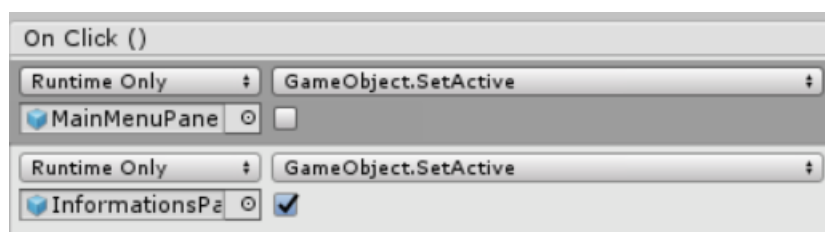


Figure 34 : MainMenu, Clickable Button<sup>35</sup>

Le bouton passeport possède également un script pour que l'on vérifie si l'utilisateur a fait les jeux de Fouly et s'il possède donc le passeport pour ce sentier. Les objets et les méthodes de « SaveLoad » vont être utilisés afin de vérifier chaque jeu de la Fouly. Si tous les jeux ont été réalisés avec succès, l'image passée en paramètre (FoulyPassImg) sera en couleur, sinon elle restera en noir et blanc.

<sup>34</sup> Source : Screenshot personnel de la vue MainMenuPanel

<sup>35</sup> Source : Screenshot personnel de la configuration du On Click () sur le bouton passeport

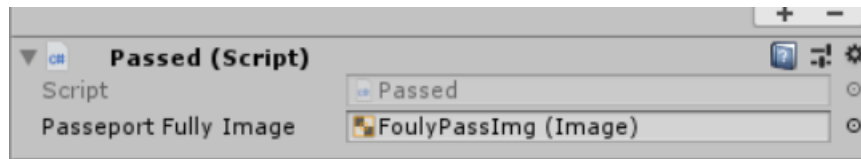


Figure 35 : MainMenu. Bouton passeport<sup>36</sup>

## 2. SentiersPanels

Le Panel des sentiers est le cœur de l'application avec les fonctionnalités principalement recherchées par le CREPA, à savoir les jeux et la recherche du début du sentier. La hiérarchie se présente comme suit :



Figure 36 : SentiersPanels, Architecture<sup>37</sup>

Le premier Panel, « SentierPanel » se compose de cinq différents boutons comprenant le nom des cinq sentiers. Ces boutons permettent de charger les autres Panels avec le sentier voulu. Ainsi, à partir d'ici, les informations relatives au sentier sélectionné seront disponibles.



Figure 37 : SentiersPanels, Vue<sup>38</sup>

<sup>36</sup> Source : Screenshot personnel du contenu du bouton passeport

<sup>37</sup> Source : Screenshot personnel de l'architecture du Panel de SentiersPanels

<sup>38</sup> Source : Screenshot personnel de la vue de SentiersPanels

Le deuxième Panel, « PlayPanel », comprend trois boutons :



Figure 38 : PlayPanel, Vue<sup>39</sup>

a. New start

**NEW START** : Permet de lancer une partie avec des nouveaux jeu en choisissant le nombre de personnes qui jouent et le niveau de difficulté.



Figure 39 : PlayerPanel & LevelPanel, Vues<sup>40</sup>

Il serait possible de jouer à une personne sur un téléphone ou à plusieurs personnes sur ce même téléphone. On y trouve également trois niveaux de difficulté, du plus faible au plus fort :

Fourmi < Ecureuil < Renard

Une fois la difficulté choisie, l'utilisateur est amené à la sélection des jeux. Plusieurs jeux sont disponibles par sentier et par difficulté. Pour la Fouly par exemple, quatre jeux sont répertoriés pour une personne avec un niveau de difficulté fourmi.

<sup>39</sup> Source : Screenshot personnel de la vue de PlayPanel

<sup>40</sup> Source : Screenshot personnel des vues de Playerpanel et de LevelPanel





Figure 40 : BlocPanel, Vue<sup>41</sup>

Les jeux sont organisés par postes sur le réel sentier de la Fouly, c'est pourquoi l'utilisateur ouvre l'application au poste où il se situe et vérifie si un jeu mobile est disponible pour le poste en question.

Les différents jeux sont expliqués sous « FoulyGames ».

#### b. Information



: Emmène l'utilisateur sur un panel d'information sur le sentier, « SentierInfoPanel », qui contient son nombre de kilomètres, si le parcours est faisable pour tous, la carte du sentier et d'autres informations importantes.



Figure 41 : SentierInfoPanel, Vue<sup>42</sup>

Une classe va permettre de retrouver quel sentier veut être affiché et y télécharger les informations et l'image relative à ce sentier.

<sup>41</sup> Source : Screenshot personnel de la vue de BlocPanel

<sup>42</sup> Source : Screenshot personnel de la vue de SentierInfoPanel



### c. Navigation



: Permet de trouver le début du sentier grâce à la géolocalisation et aux coordonnées GPS. Ce bouton lance la scène « FindSentierStart ». La fonctionnalité de cette scène est à découvrir dans la section « FindSentierStart » qui se situe ci-dessous dans le document.

### 3. PassportsPanel

Le Panel de passeports contient cinq images avec le logo de chaque sentier. Si les jeux du sentier ont été effectués, le logo apparaîtra en couleur. Les images sont cliquables afin d'ajouter une éventuelle méthode pour la suite de l'application.



Figure 42 : PassportsPanel, Vue Fouly Games Passed<sup>43</sup>

### 4. AlbumsPanel

L'album photos propose de revoir les photos prises à travers l'application. Il est possible de consulter toutes les images grâce aux flèches jaunes. Un script va récupérer un tableau de photos qui est enregistré lors de la prise desdites photos.



Figure 43 : AlbumsPanel, Vue avec photo<sup>44</sup>

<sup>43</sup> Source : Screenshot personnel de la vue de PassportPanel, avec la réussite des jeux de la Fouly

<sup>44</sup> Source : Screenshot personnel de la vue de AlbumsPanel, avec une photo déjà dans l'album

## 5. InformationPanel

Des informations sur Charlotte la Marmotte peuvent être trouvées dans cette section ainsi que des liens vers le Facebook du CREPA et le site officiel.

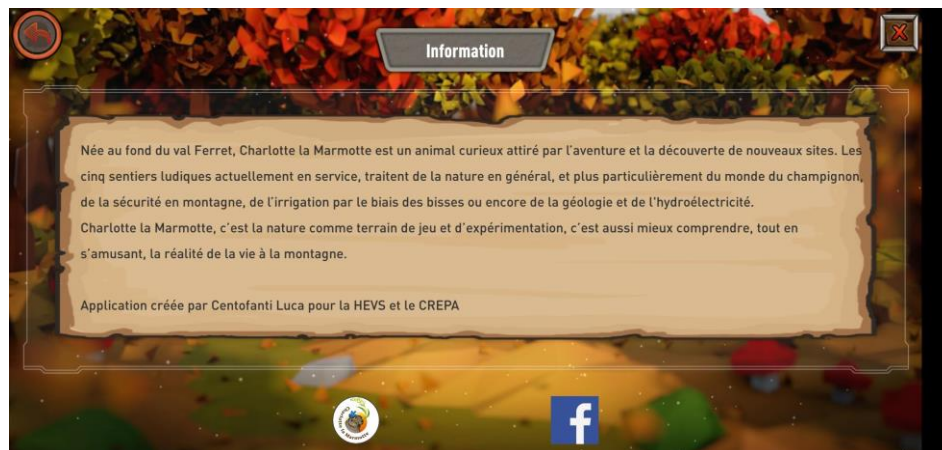


Figure 44 : InformationPanel, Vue<sup>45</sup>

### 3.4. FindSentierStart

Sur certains sentiers comme celui de la Fouly, il est difficile de trouver le début du parcours, les panneaux étant petits et pas visibles. C'est pourquoi l'application doit apporter une aide afin de trouver le début du sentier facilement et de ne pas se perdre.

L'idée étant de réinventer le GPS traditionnel en l'intégrant dans une solution en réalité augmentée. Plusieurs outils ont été analysés tels que :

#### 1. Mapbox

C'est une plateforme de données de localisation pour des applications mobiles ou web. La plateforme fournit des blocs de construction pour ajouter des fonctions de localisation. Mapbox contient des outils pour la réalité augmentée, tels que des données de localisation en direct qui permettent au gameplay de s'adapter, d'un atlas mondial de géolocalisation des lieux les plus intéressants et de modèles numériques d'élévation en 3D. Le tout directement importable au sein de Unity. (Mapbox, s.d. 2018)

#### 2. Motive.io

Cette solution utilise Mapbox et permet la construction simple de jeux géolocalisés. La plateforme s'intègre facilement à Unity et met à disposition un hébergement, des outils de création ainsi que des

<sup>45</sup> Source : Screenshot personnel de la vue de InformationPanel

Templates pour faciliter la prise en main. Elle intègre également des services géoréférencés et permet une utilisation combinée de « Vision Augmented Reality » et de « Location-based Augmented Reality ». (Motive.io, s.d. 2018)

« Vision Augmented Reality » couvre des technologies comme Vuforia, Wikitude et ARKit qui utilisent le suivi d'images du monde réel pour ancrer les médias et les éléments interactifs sur une caméra en direct. (Motive.io, s.d. 2018)

« Location-based Augmented Reality » est utilisé pour les jeux et applications comme PokémonGo, l'appareil photo du téléphone est utilisé pour interagir avec des objets virtuels connectés à des emplacements du monde réel. (Motive.io, s.d. 2018)

Mapbox et Motive.io sont des outils très complets qui proposent des milliers de fonctionnalités, ce qui prend une place non-négligeable dans l'application. De plus, certaines données doivent être stockées sur un serveur, ce qui peut être dérangement étant donné la localisation des sentiers. Ces outils sont des usines à gaz dans lesquels une infime partie des outils seront utilisés. En outre, ils sont difficiles à prendre en main. C'est pourquoi ils n'ont pas été utilisés et qu'une solution a été créée « from scratch ».

La solution créée permet de s'orienter en direction du début d'un sentier tout en y intégrant de la réalité augmentée. Le but étant d'orienter l'aiguille d'une boussole en 2D vers la direction voulue. Pour ce faire, il faut accéder aux données GPS du téléphone afin de définir la position de l'utilisateur, au gyroscope et au compas afin de savoir dans quelle direction l'utilisateur regarde. Les coordonnées du début du sentier ont été prises sur place (Fouly).

Une fois les deux coordonnées acquises, celle de l'utilisateur et celle du début du sentier, il faut orienter l'objet vers la destination voulue. Pour ce faire, il faut étudier comment calculer la distance et le « bearing » entre deux coordonnées (composée d'une longitude et d'une latitude) qui a été vu ici : <https://www.movable-type.co.uk/scripts/latlong.html>.

De plus, il a également fallu étudier les rotations des objets dans Unity notamment des Quaternion, des fonctions Slerp et Lerp, des EulerAngles, des rotations et des rotations.local. Nous nous sommes aidés d'un problème similaire rencontré par « ibrahimAlfors » qu'il est possible de retrouver ici : <https://answers.unity.com/questions/1500577/gps-2d-compass-to-point-to-longitude-and-latitude.html>

## 1. Avoir accès aux données GPS du téléphone :

```

90 private IEnumerator StartLocationService()
91 {
92     //Si la localisation n'est pas activée par l'utilisateur
93     if (!Input.location.isEnabledByUser)
94     {
95         Debug.Log("User has not activate the gps");
96         yield break;
97     }
98     //Start la localisation
99     Input.location.Start();
100     int maxWait = 20;
101
102     while (Input.location.status == LocationServiceStatus.Initializing && maxWait > 0)
103     {
104         yield return new WaitForSeconds(1);
105         maxWait--;
106     }
107     //Plus de temps
108     if (maxWait <= 0)
109     {
110         Debug.Log("Time out");
111         yield break;
112     }
113     //Le téléphone ne peut pas être localisé
114     if (Input.location.status == LocationServiceStatus.Failed)
115     {
116         Debug.Log("Can't located the phone");
117         yield break;
118     }
119
120     //Acces à la longitude et à la latitude du téléphone
121     currentLatitude = Input.location.lastData.latitude;
122     currentLongitude = Input.location.lastData.longitude;
123 }

```

Figure 45 : FindSentierStart, Dev. Script Méthode StartlocationService<sup>46</sup>

Cette IEnumerator est appelé dans la fonction « Start » afin qu'il se lance quand le script est activé :

```

29 private void Start()
30 {
31     Instance = this;
32     //Start Location service
33     StartCoroutine(StartLocationService());
34     //Active le compass
35     Input.compass.enabled = true;
36 }

```

Figure 46 : FindSentierStart, Dev. Script Start TrackGpsCoordinates<sup>47</sup>

## 2. Calculer le « Bearing », l'angle entre deux coordonnées :

<sup>46</sup> Source : Screenshot personnel de la méthode StartlocationService

<sup>47</sup> Source : Screenshot personnel de la fonction Start de la classe TrackGpsCoordinates

```

56 private float angleFromCoordinate(float lat1, float long1, float lat2, float long2)
57 {
58     lat1 *= Mathf.Deg2Rad;
59     lat2 *= Mathf.Deg2Rad;
60     long1 *= Mathf.Deg2Rad;
61     long2 *= Mathf.Deg2Rad;
62
63     float dLon = (long2 - long1);
64     float y = Mathf.Sin(dLon) * Mathf.Cos(lat2);
65     float x = (Mathf.Cos(lat1) * Mathf.Sin(lat2)) - (Mathf.Sin(lat1) * Mathf.Cos(lat2) * Mathf.Cos(dLon));
66     float brng = Mathf.Atan2(y, x);
67     brng = Mathf.Rad2Deg * brng;
68     brng = (brng + 360) % 360;
69     brng = 360 - brng;
70     return brng;
71 }
72

```

Figure 47 : FindSentierStart, Dev. Script Méthode AngleFromCoordinate<sup>48</sup>

La fonction « AngleFromCoordinate » est appelée dans l'Update pour que les coordonnées de l'utilisateur soient mises à jour après chaque frame (constamment). De plus, le résultat de cette fonction est stocké dans une variable « bearing » afin qu'elle puisse être réutilisée.

```

void Update()
{
    //Calcule du bearing
    float bearing = angleFromCoordinate(Input.location.lastData.latitude, Input.location.lastData.longitude, FoulyLatitude, FoulyLongitude);

    //Orienté l'aiguille de la boussole vers une localisation précise
    SentierNeedleLayer.localRotation = Quaternion.Slerp(SentierNeedleLayer.localRotation, Quaternion.Euler(0, 0, Input.compass.trueHeading + bearing), 100f);
    //Afficher la latitude et longitude actuel de l'utilisateur sur l'écran
    coordinates.text = "Latitude: " + Input.location.lastData.latitude + "\nLongitude: " + Input.location.lastData.longitude;
}

```

Figure 48 : FindSentierStart, Dev. Script Update TrackGpsCoordinates<sup>49</sup>

Dans cette même fonction « Update » l'aiguille de la boussole va être orientée dans la bonne direction, grâce au Quaternion.Slerp.



Figure 49 : FindSentierStart, Vue<sup>50</sup>

<sup>48</sup> Source : Screenshot personnel de la méthode angleFromCoordinates

<sup>49</sup> Source : Screenshot personnel de la fonction update de la classe TrackGpsCoordinates

<sup>50</sup> Source : Screenshot personnel de la vue de la scène FindSentierStart

### 3.5. Fouly Games

Pour cette application, quatre jeux ont été créés pour le sentier de la Fouly afin de montrer ce qui peut être fait pour ce genre de projets. Ces jeux sont facilement intégrables pour d'autres sentiers avec des changements de données.

#### 3.5.1. Le loup

Ce jeu consiste à faire apparaître un loup grâce au logo de Charlotte la Marmotte. Une fois le loup apparu, l'utilisateur peut le faire bouger comme bon lui semble.

L'architecture de la scène se présente ainsi :

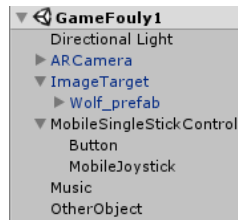


Figure 50 : Jeu le loup, Architecture<sup>51</sup>

L'AR Camera a été configurée comme expliqué dans la configuration de Vuforia en annexe I, Configuration de Vuforia. Un préfabriqué AR Camera existe afin de faciliter le développement. Il se situe dans :

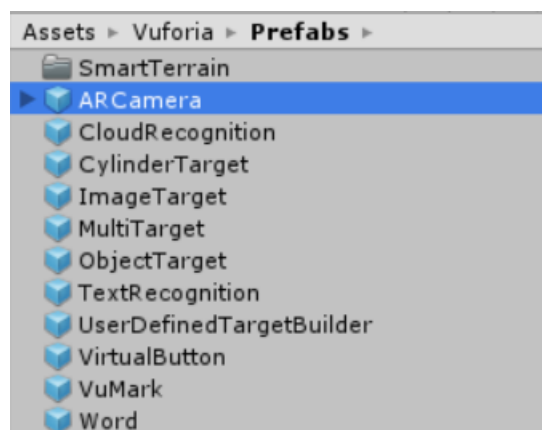


Figure 51 : Jeu le loup, Vuforia prefabs<sup>52</sup>

<sup>51</sup> Source : Screenshot personnel de l'architecture de la Scène GameFouly1

<sup>52</sup> Source : Screenshot personnel des préfabriqués de Vuforia



L'image Target est également intégrée depuis le préfabriqué de Vuforia. Le Target doit être configurée avec la base de données stockée sur Vuforia pour que la caméra reconnaisse l'image. La base de données et l'image souhaitées doivent être sélectionnées.

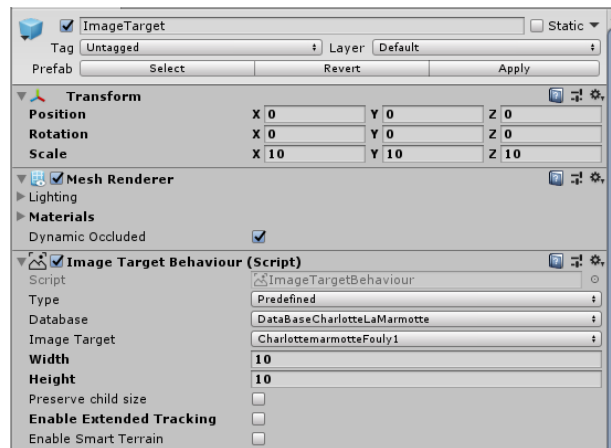


Figure 52 : Jeu le loup, ImageTarget<sup>53</sup>

Dans cette ImageTarget, on trouve l'objet que l'on aimerait faire apparaître, dans ce cas un loup (« Wolf\_prefab »). Ce loup est un asset 3D téléchargé sur l'Asset store de Unity. Il comprend également des animations. Un script de commandes a été ajouté afin de contrôler les mouvements et les animations du loup.

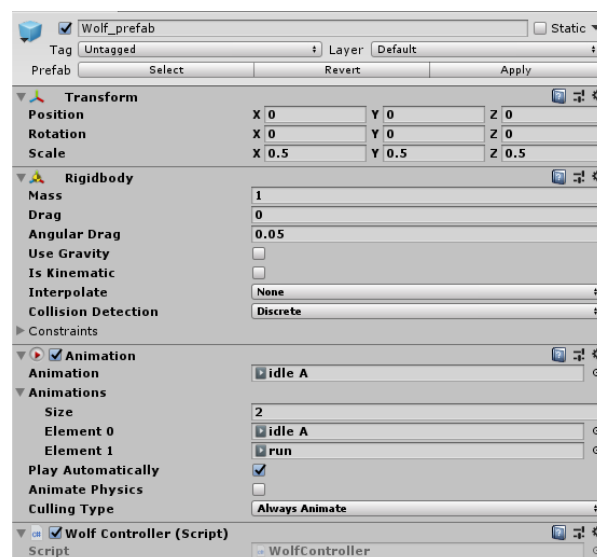


Figure 53 : Jeu le loup, l'asset 3D du loup<sup>54</sup>

<sup>53</sup> Source : Screenshot personnel de la configuration de l'image Target

<sup>54</sup> Source : Screenshot personnel de l'objet Wolf\_prefab

Le « WolfController » permet d'accéder aux propriétés du loup afin de déclencher des animations lorsque celle-ci sont demandées. Il permet aussi de faire avancer le loup dans des directions choisies par le bouton « mobileJoystique ».



Figure 54 : Jeu Le loup, Vue<sup>55</sup>

<sup>55</sup> Source : Screenshot personnel de la vue du jeu du loup de l'application



### 3.5.2. La prise de photo

Avec cette fonction, l'utilisateur peut prendre une photo ou un selfie avec un animal en réalité augmentée. Le but est de regarder autour de soi pour trouver l'animal et, une fois trouvé, l'animal fait du sur place mais est muni d'animations. L'utilisateur peut donc l'observer sans le perdre de vue. La photo apparaît dans son album au sein de l'application.

Voici la structure de la scène de la prise de photo :

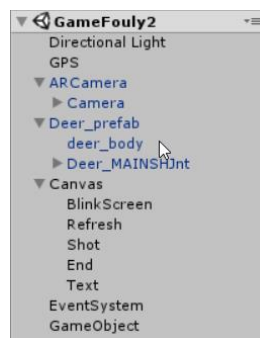


Figure 55 : Jeu Prise de photo, Architecture<sup>56</sup>

L'objet GPS contient un script qui permet l'accès aux informations GPS du téléphone afin d'afficher dans le Canvas, Text, la latitude et la longitude actuelles de l'utilisateur.

La AR Camera est configurée comme toutes les autres caméras de réalité augmentée du projet. Il est important d'avoir l'option « DEVICE\_TRACKING » et d'activer les options « Enable device pose tracking » et « Enable prediction » car sans ces options, l'Asset 3D reste figé dans l'écran et n'apporte rien de plus qu'une image. Grâce à ces outils, le gyroscope du téléphone est utilisé et permet à l'Asset de sortir et revenir dans l'écran en bougeant le téléphone. Ceci donne l'impression que l'Asset 3D est fixé dans un endroit du décor.

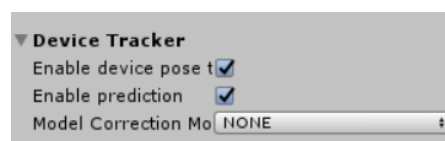


Figure 56 : Jeu Prise de photo, AR Camera<sup>57</sup>

Dans cette scène, un Asset 3D de cerf est utilisé avec des animations pour accentuer son réalisme. Une boucle d'animations y est ajoutée afin que l'objet soit en mouvement constant.

<sup>56</sup> Source : Screenshot personnel de l'architecture de la scène GameFouly2

<sup>57</sup> Source : Screenshot personnel de la configuration de Vuforia

Le Canvas contient tous les éléments de l'interface :

1. « BlinkScreen » est l'écran affichant un flash blanc lors de la prise de photo.
2. « ChangeMode » est un bouton changeant le mode de photo entre selfie ou normal. Le script lié à cet objet permet d'accéder aux différents objectifs photo de l'appareil.

```
public class FrontCameraAccess : MonoBehaviour{

    public void ChangeDirection()
    {
        CameraDevice.CameraDirection curent = CameraDevice.Instance.GetCameraDirection();

        if(curent == CameraDevice.CameraDirection.CAMERA_BACK)
        {
            CameraDevice.Instance.Stop();
            CameraDevice.Instance.Deinit();
            CameraDevice.Instance.Init(CameraDevice.CameraDirection.CAMERA_FRONT);
            CameraDevice.Instance.Start();
        }
        else
        {
            CameraDevice.Instance.Stop();
            CameraDevice.Instance.Deinit();
            CameraDevice.Instance.Init(CameraDevice.CameraDirection.CAMERA_BACK);
            CameraDevice.Instance.Start();
        }
    }
}
```

Figure 57 : Jeu Prise de photo, Dev. Script FrontCameraAccess<sup>58</sup>

3. « Shot » prend la photo, le script que contient ce bouton est constitué ainsi

```
public class TakePicture : MonoBehaviour {

    Animator blink;

    public void Start()
    {
        blink = GameObject.Find("BlinkScreen").GetComponent<Animator>();
    }

    public void TakeAShot()
    {
        blink.enabled = true;
        StartCoroutine("CaptureIt");
    }

    IEnumerator CaptureIt()
    {
        string timeStamp = System.DateTime.Now.ToString("dd-MM-yyyy-HH-mm-ss");
        string fileName = "CharlotteLaMarmotte" + timeStamp + ".png";
        string pathToSave = fileName;
        ScreenCapture.CaptureScreenshot(pathToSave);
        yield return new WaitForEndOfFrame();
        Instantiate(blink, new Vector2(0f, 0f), Quaternion.identity);
    }
}
```

Figure 58 : Jeu Prise de photo, Dev. Script TakePicture<sup>59</sup>

La coroutine IEnumerator crée un nom au fichier photo qui sera sauvegardé. A la suite de la création du nom, une capture d'écran peut être réalisée. Dans ce script, l'animation de

<sup>58</sup> Source : Screenshot personnel du script FrontCameraAccess

<sup>59</sup> Source : Screenshot personnel du script TakePicture

« BlinkScreen » est également gérée afin qu'elle soit activée lors de la pression sur le bouton « Shot ».

4. « End » est le bouton de retour au menu
5. « Text » contient la latitude et longitude actuelles de l'utilisateur



Figure 59 : Jeu Prise de photo, Vue<sup>60</sup>

<sup>60</sup> Source : Screenshot personnel de la vue photo de l'application

### 3.5.3. Les images prennent vies

Ce jeu est spécifique au sentier de la Fouly. Sur ce sentier au post n°9 on y trouve des photos d'animaux qui sont coupées en quatre ; le but de l'exercice étant de reconstituer les photos des animaux une à une. De plus, une fois la photo reconstruite, un dispositif auditif est placé à côté afin de lier l'animal à son cri. Voici une illustration du poste sans le dispositif de réalité augmentée :



Figure 60 : Jeu Les images prennent vies, photo du poste sans RA<sup>61</sup>

Ce jeu consiste à scanner l'image reconstruite afin d'avoir une animation 3D de l'animal en question. Ceci a pour but d'immerger complètement l'utilisateur autour des animaux présents dans la Fouly. Voici deux images du poste avec l'incrustation de l'animal en 3D. L'animal contient des animations et paraît vivant, malheureusement ceci ne peut pas être constaté sur une simple image :

<sup>61</sup> Source : Photo fournie par le CREPA





Figure 61 : Jeu Les images prennent vies, photo du poste avec RA, Cerf<sup>62</sup>



Figure 62 : Jeu Les images prennent vies, photo du poste avec RA, Bouquetin<sup>63</sup>

Afin de parvenir à ce dispositif une scène a été créée. Elle contient une AR Camera et six images Target afin de différencier les animaux.



Figure 63 : Jeu Les images prennent vies, Architecture<sup>64</sup>

Les images Target représentent les animaux du panneau. Elles doivent être configurée de la même manière que celles contenues dans le jeu du loup.

<sup>62</sup> Source : Screenshot personnel de l'application CharlotteLaMarmotte sur mon téléphone

<sup>63</sup> Source : Screenshot personnel de l'application CharlotteLaMarmotte sur mon téléphone

<sup>64</sup> Source : Screenshot personnel de l'architecture de la scène GameFouly3

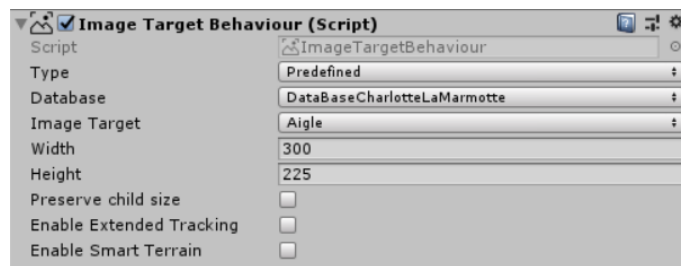


Figure 64 : Jeu Les images prennent vies, Image Target<sup>65</sup>

La base de données créée pour le projet doit être sélectionnée ainsi que l'image mise à disposition pour l'animal en question.

#### 3.5.4. Le Quizz

Le Quizz permet aux utilisateurs de répondre à des questions. Il peut être réutilisé pour tous les sentiers car les questions sont facilement modifiables. Ainsi, il est possible de faire des Quizz sur des postes mais également sur d'autres sujets. Dans l'exemple créé, les questions se posent sur le poste numéro 6. Si l'utilisateur lit bien le texte du poste, il lui sera aisé de répondre aux questions.

Les utilisateurs sont obligés de trouver la bonne réponse afin de continuer, ceci permettant l'acquisition certaine des connaissances voulues. Cependant, une contrainte de temps a été ajoutée. Initialement, un Quizz sous forme de points avait été créé, mais il est vrai que l'on ne pouvait jamais savoir la réponse exacte à part en refaisant l'exercice maintes fois.

Le Quizz se présente sous cette forme :



Figure 65 : Jeu Quizz, Vue mauvaise réponse<sup>66</sup>

<sup>65</sup> Source : Screenshot personnel de la configuration de l'image Target

<sup>66</sup> Source : Screenshot personnel de la vue du Quizz

L'architecture de ce jeu comprend un Canvas contenant l'interface, un « GameController » qui gère la logique du jeu et un « ObjectPool » :

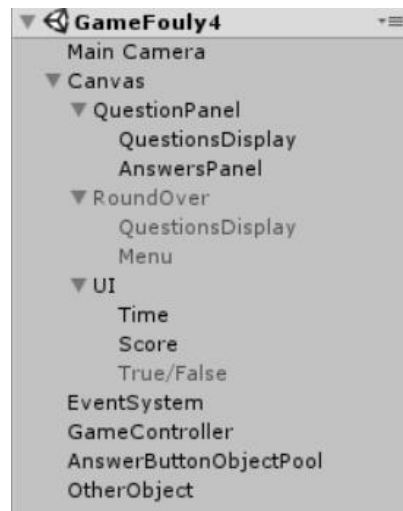


Figure 66 : Jeu Quizz, Architecture<sup>67</sup>

Le « QuestionPanel » affiche les questions ainsi que les différentes réponses à choix. Quand l'utilisateur choisi la réponse, s'il celle-ci est fausse, il en est notifié avec un message. Dans le cas contraire, il passe à la question suivante. Lorsqu'il n'y a plus de question ou que le temps imparti est terminé, le panel « RoundOver » s'affiche. Celui-ci contient un bouton pour revenir au menu.

« UI » est l'incrustation du score, du temps restant et de l'affiche du message d'erreur en cas de mauvaise réponse. Le panel « UI » apparaît sur les deux autres panels, « QuestionPanel » et « RoundOver ».

L'élément principal de ce Quizz est l'objet « GameController » qui contient le script « QuizzGameController ». Ce script permet de charger les questions avec leurs différentes réponses et de contrôler si celles choisies par l'utilisateur sont correctes ou non. C'est également avec celui-ci que l'affichage va être géré autant pour le passage à la question suivante que pour l'affichage du panel « RoundOver ». Il calcule également les points obtenus et le temps qui s'écoule.

La fonction ci-dessous fait partie du script « QuizzGameController ». Elle permet l'affichage des différentes questions avec leurs réponses respectives :

<sup>67</sup> Source : Screenshot personnel de l'architecture de la scène GameFouly4

```
private void ShowQuestion()
{
    RemoveAnswerButtons();
    QuestionData questionData = questionPool[questionIndex];
    questionText.text = questionData.questionText;

    for (int i = 0; i < questionData.answers.Length; i++)
    {
        GameObject newAnswerButton = answerButtonsObjectPool.GetObject();
        answerButtonsGameObjects.Add(newAnswerButton);
        newAnswerButton.transform.SetParent(answerButtonParent);

        AnswerButton answerButton = newAnswerButton.GetComponent<AnswerButton>();
        answerButton.Setup(questionData.answers[i]);
    }
}
```

Figure 67 : Jeu Quizz, Dev. Méthode ShowQuestion<sup>68</sup>

« AnswerButtonObjectPool » permet de réutiliser un objet préfabriqué donné en paramètre. Plus de détails sur l'Object pooling sont disponibles au début du chapitre.

### 3.6. Multiplateforme

L'application de Charlotte la Marmotte peut facilement être installée sur Android ou IOS, sans aucune modification du code. La solution fonctionne exactement de la même manière sur l'une ou l'autre des distributions. Il faut cependant un ordinateur doté de Xcode pour la construire sur IOS (Xcode étant exclusivement sur Mac).



Figure 68 : Multiplateforme, Android et IOS<sup>69</sup>

<sup>68</sup> Source : Screenshot personnel du script QuizzGameController et de sa méthode ShowQuestion

<sup>69</sup> Source : Photo personnelle d'un OnePlus et de l'iPhone de l'institut de recherche de la HEVS



## 4. Problèmes rencontrés

Durant le développement de l'application, plusieurs problèmes ont été rencontrés, dont beaucoup avec le GPS et le multiplateforme. Une solution a été trouvée pour permettre à l'utilisateur de situer le début d'un sentier, mais l'accès au GPS, lorsque l'application est fermée ou inactive, est plus compliqué avec Unity.

### 4.1. Notification GPS

Des notifications activées lors de l'arrivée à une certaine cordonnée voulaient être intégrées dans l'application Charlotte la Marmotte. Cela aurait évité l'utilisation excessive du smartphone qui aurait été sorti de la poche ou du sac seulement lors de la notification par vibration ou par son. De plus, cette fonctionnalité aurait permis d'ajouter des jeux entre certains postes ou à des endroits spécifiques, tels que sur un beau panorama ou la vue sur un glacier. C'est pourquoi l'idée des notifications par GPS avait fait son apparition.

Malheureusement, il n'a pas été possible de l'intégrer dans l'application aujourd'hui par manque de temps pour la recherche sur des problèmes rencontrés. Les notifications locales sont facilement réalisables, c'est l'accès à la position du téléphone lorsque notre application est en pause ou ne tourne pas du tout qui est plus compliqué.

Deux types de notifications existent : la notification locale et la notification push. La notification locale va permettre de notifier l'utilisateur sur une donnée propre au téléphone. Par exemple, si après une heure l'utilisateur n'a pas ouvert une application, celle-ci va créer une notification et l'informer qu'il serait bien d'y retourner. Une notification push sert à notifier l'utilisateur sur des éléments externes, contenus dans une base de données. Les notifications « push » sont souvent utilisées par les applications de messages, telles que WhatsApp ; lors de la réception d'un message sur notre serveur, une notification apparaît. Ces notifications sont également utilisées par les applications de « journaux » ; lorsqu'il se passe quelque chose de nouveau dans le monde, une notification apparaît sur notre smartphone.

La notification push utilisant des serveurs externes et donc nécessitant une connexion internet a été écartée pour se concentrer uniquement sur les notifications locales.

Des plugins sur Unity permettent de réaliser simplement des notifications locales sur un téléphone muni d'Android ou IOS (séparément). Il existe également des Assets qui permettent de gérer des notifications multiplateformes tels que :

UTNotifications - Professional Local & Push Notification Plugin (65 \$)

La notification locale sur Android est donc disponible et il est facile de l'actionner à l'aide d'un bouton ou lors de la fermeture de l'application et d'y ajouter un timer d'une heure. Il est en revanche beaucoup plus compliqué d'accéder au GPS du téléphone une fois notre application Unity éteinte ou en pause.

Le GPS Trigger est possible en natif sur Android en créant un service. Un service est un composant qui s'exécute en arrière-plan sans interaction avec l'utilisateur, ce qui lui permet d'accéder aux coordonnées GPS du téléphone et de notifier l'utilisateur lorsqu'il arrive sur une coordonnée précise. Il a aussi la capacité de démarrer une application. Cependant, il est possible d'utiliser les services uniquement sur Android, car c'est une fonctionnalité qui lui est propre.

Une méthode native est également disponible sur IOS, mais aucun moyen n'a été actuellement trouvé pour faire du « GPS Trigger » multiplateforme.

Malheureusement, par manque de temps, il n'a pas été possible de fournir cette fonctionnalité sur Android ou sur IOS mais il est important de savoir que cela est faisable nativement.

## 5. Amélioration du prototype

L'application de Charlotte la Marmotte n'est pour le moment qu'un prototype. On y trouve la présentation des outils principaux qui peuvent être réutilisés afin de la mettre en production. Le prototype fonctionne et de grandes évolutions ne sont pas nécessaires. Une meilleure organisation du code, telle que l'amélioration de la programmation orientée objets ainsi que des tests seraient néanmoins judicieux.

L'interface, la page d'informations, l'album photo, la description de tous les sentiers, la recherche du début d'un sentier ainsi que les jeux principaux sont implémentés. Il y a une grande quantité de jeux par sentier. Ils n'ont donc pas pu être tous réalisés. Cependant, les jeux tels que la prise de photos ou le Quizz peuvent être facilement intégrables à tous les sentiers du CREPA et également sur plusieurs niveaux de difficultés.

La création de jeux « multijoueur » sur un seul téléphone avait été évoquée, cependant le prototype ne couvre pas ce sujet. Seul le bouton est visible sur l'application mais il ne contient aucune fonction. Il est vrai que tous les enfants ne possèdent pas forcément leurs propres téléphones portables et qu'ils pourraient éventuellement jouer sur celui de leurs parents. Un mode « multijoueur » local peut être intéressant dans ce cas-là.

La notification par GPS doit être intégrée. Ainsi, les jeux ne seront pas disponibles seulement sur les postes, mais également à d'autres points importants du sentier. Cette idée plait particulièrement au CREPA.

Il serait également intéressant d'ajouter des informations sur les photos prises sur le sentier, telles que la température, le dénivelé ou le temps mis pour finir le sentier. Un montage photos peut être intégré grâce aux photos prises sur chaque sentier. Les utilisateurs pourraient alors l'imprimer afin d'avoir un souvenir de leur épopée.

## 6. Evaluation de l'application

Il a été compliqué de trouver des personnes intéressées à tester l'application sur le terrain, la Fouly n'étant à proximité pour tout le monde. Cependant, deux personnes et moi-même avons pu l'essayer durant un beau jour de juillet.

Les personnes l'ayant essayé ont eu ces retours :

« Mes impressions pour l'application de Charlotte La Marmotte à la Fouly (VS).

J'aime beaucoup le mélange du côté ludique dans la nature et la modernisation qu'apporte l'application avec mon portable. Ces deux aspects se marient très bien. Je n'ai pas vu passé le temps car j'essayais de repérer les points où l'on pouvait découvrir des informations de la Fouly et répondre aux questions autant sur le formulaire que sur le Quizz de l'application. L'aspect Quizz m'a réellement plus !

L'application est un plus, surtout pour la nouvelle génération. Cela leur permet de découvrir les environs en étant à l'extérieur entourés de l'air frais du Valais, tout en restant en contact avec leur portable.

Le seul point négatif que j'ai trouvé, c'est que malheureusement on ne peut pas zoomer sur la carte dans les informations des sentiers. Ce qui rend la lecture de l'itinéraire compliqué. Sinon l'application m'a conquise et j'espère qu'elle sera au plus vite disponible pour tous les sentiers pour que je puisse en profiter. » (Centofanti S. , 2018)

« Mon parcours de la Fouly avec Charlotte La Marmotte.

J'ai beaucoup aimé l'apport de la technologie au sein de la nature. Ceci m'a permis d'apprendre de nouvelles choses tout en m'amusant. J'avais entendu parler de réalité augmentée sans forcément en voir et il est vrai que cela m'a bluffée. L'incrustation des animaux sur les divers postes est vraiment bien faite et apporte un réel plus au sentier, les vrais animaux se faisant rares...

Je trouve l'application facile à prendre en main, mais il est vrai que l'on doit beaucoup jongler entre sortir le portable ou pas. Une indication sur les postes, où un jeu virtuel est possible, serait la bienvenue. Ceci permettrait de ne pas regarder à chaque poste si un jeu virtuel est disponible et donc de sortir parfois le téléphone pour rien.

J'ai passé une magnifique journée et j'ai appris beaucoup de choses, je le referai tout volontiers ! » (Centofanti T. , 2018)

## 7. Planification

La gestion du projet ainsi que les outils utilisés sont des points importants durant le développement d'un travail tel que celui-ci. Durant ce projet des réunions avec la responsable de Bachelor, Mme Maria Sokhn accompagnée de Mme Camille Pellaton, sont effectuées une fois chaque deux semaines. Cela nous permet de suivre l'avancée du projet et de valider une à une les fonctionnalités de l'application. De plus, une réunion avec le client, les membres du CREPA, est effectuée une fois par mois pour qu'ils puissent avoir une vue d'ensemble sur le travail fourni et ainsi partager leurs idées et leurs envies.

Ces réunions sont vitales pour le projet car elles apportent un échange indispensable lors de la création d'une application. De plus, tous les partis sont enthousiastes à l'idée de la réalisation de ce projet, ce qui rend les meetings plus instructifs !

La communication se fait par email avec le CREPA et à travers SLACK entre Mme Maria Sokhn, Mme Camille Pellaton et moi-même. SLACK étant un outil de messagerie instantanée professionnel.

La planification est réalisée à travers « TeamGantt », un site permettant la gestion facilitée d'un diagramme de Gant. Mme Maria Sokhn et moi-même faisons partie de cet outil. Une capture d'écran de « TeamGantt » durant le développement du projet est disponible en annexe II.

## 8. Conclusion

Durant l'analyse des applications existantes, nous avons pu constater que la réalité augmentée prend son envol sur le marché. En effet, beaucoup d'entreprises commencent à utiliser cette technologie. Le phénomène PokémonGo n'y est pas pour rien. Ceci est possible grâce à des logiciels tels que Unity et des plugins comme Vuforia, qui permettent une implémentation facile de la réalité augmentée. La vaste communauté Unity met à disposition des guides pour tous niveaux, ce qui rend l'apprentissage de cette technologie relativement aisé.

Les bases sont accessibles pour tous, mais plus nous avançons à travers l'application plus les sujets devenaient complexes. Ces programmes limitent fortement certaines fonctionnalités qui sont possible grâce au code natif, notamment l'accès à certains outils du téléphone portable. La navigation sur Unity est orientée pour les jeux où l'interface reste petite et simple. La conception d'une interface plus complexe se créer de manière plus laborieuse.

Les technologies utilisées au sein de Charlotte la Marmotte permettent un multiplateforme exemplaire qui fait gagner énormément de temps. Durant le développement, tout a été testé sur Android. A la fin, il a juste fallu construire l'application d'un simple clic pour une distribution IOS et l'installer sur un iPhone à travers un Mac, pour que le tout fonctionne sans aucun changement.

Finalement, Charlotte la Marmotte mixe toutes les envies que le CREPA a exprimé à savoir : la fusion de jeux simples, de jeux en RA, d'orientation GPS et de prise de photos. Cette application crée une surprise à chaque niveau !

### 8.1. Feedback personnel

Ce projet m'a permis de renforcer mon envie de travailler dans le domaine informatique, car j'ai eu beaucoup de plaisir à réaliser cette application ! Le fait d'avoir une certaine liberté avec les fonctionnalités, ce qui est plutôt rare dans le milieu, m'a énormément plu.

Cependant, cela a été un réel défi du début à la fin. Je ne connaissais la réalité augmentée qu'en théorie et je n'avais utilisé Unity qu'une seule fois. C'est pourquoi, l'organisation du code a été laborieuse. De plus, les problématiques qui devaient être respectées étaient particulièrement contraignantes. Chaque étape a été un challenge, j'ai dû me dépasser pour concevoir un prototype fonctionnel.

La plus grande difficulté que j'ai rencontrée est l'orientation par GPS. J'ai tenté d'utiliser Mapbox et Motive.io sans succès réel pour l'application. C'est pourquoi, je me suis retourné vers une solution plus « simple » en matière d'Assets. Cependant, les calculs d'orientation entre deux coordonnées et la rotation avec les Quaternions d'un objet 2D m'ont éprouvé.

Je suis fier du travail rendu, bien que je pense que la programmation orientée objets puisse être mieux intégrée à ce projet ainsi que l'organisation générale du code.

Le fait de travailler avec des personnes à l'écoute, professionnelles et agréables m'a extrêmement motivé. De plus, les retours constructifs après chaque séance m'ont permis d'y voir plus clair et d'avancer sur des bases saines. Ce projet m'a permis de renforcer mes connaissances et d'en apprendre plus sur l'organisation des projets informatiques.

Mon idée d'intégrer la réalité augmentée au souhait du CREPA n'était pas la plus aisée, mais je pense qu'elle apporte un réel atout à cette application déjà bien pensée.

## 9. Références et sources

- Agitateur. (2016, septembre 1). *Le saviez-vous ? Au départ, Pokémon Go n'était qu'un poisson d'avril* ! Récupéré sur Agitateur: <http://agitateur.com/pokemon-go-poisson-avril/>
- Apple Inc. (s.d. 2018). *ARKit*. Récupéré sur Developer Apple: <https://developer.apple.com/arkit/>
- Apple Inc. (s.d. 2018). *Documentation Arkit*. Récupéré sur Developer Apple: <https://developer.apple.com/documentation/arkit>
- Augmentera Ltd. (s.d. 2018). *Skyline*. Récupéré sur ViewRanger: <http://www.viewranger.com/fr/skyline>
- Augmentra Ltd. (s.d. 2018). *Donnez vie à vos cartes*. Récupéré sur ViewRanger: <http://www.viewranger.com/fr/skyline/donnez-vie-a-vos-cartes>
- Augmentra Ltd. (s.d. 2018). *La navigation en plein air réinventée*. Récupéré sur ViewRanger: <http://www.viewranger.com/fr/skyline/la-navigation-en-plein-air-reinventee>
- Augmentra Ltd. (s.d. 2018). *L'entreprise & son histoire*. Récupéré sur ViewRanger: <http://www.viewranger.com/fr/a-propos-de-nous/lentreprise-son-histoire>
- Centofanti, S. (2018, juillet 22). Retours sur le sentier de la Fouly. (L. Centofanti, Intervieweur)
- Centofanti, T. (2018, juillet 22). Retours sur le sentier de la Fouly. (L. Centofanti, Intervieweur)
- CREPA. (s.d. 2018). *A la creusaz*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/sentiers/a-la-creusaz>
- CREPA. (s.d. 2018). *A propos*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/a-propos>
- CREPA. (s.d. 2018). *Champex-lac-orsieres*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/sentiers/champex-lac-orsieres>
- CREPA. (s.d. 2018). *Emosson*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/sentiers/emosson>
- CREPA. (s.d. 2018). *La Fouly*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/sentiers/a-la-fouly>
- CREPA. (s.d. 2018). *le crepa...* Récupéré sur CREPA: <https://www.crepa.ch/le-crepa>



- CREPA. (s.d. 2018). *Trient*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/sentiers/bisse-du-trient>
- CREPA. (s.d.). *Accueil*. Récupéré sur CharlotteLaMarmotte: <https://www.charlottelamarmotte.ch/>
- Dick, J. (2017, Octobre 30). *Augmented Reality in Xamarin.Android with ARCore*. Récupéré sur Blog Xamarin: <https://blog.xamarin.com/augmented-reality-xamarin-android-arcore/>
- Easy Mountain. (2017). *A propos*. Récupéré sur Mhikes: [https://mhikes.com/presse/Dossier de press B2C Mhikes 2017.pdf](https://mhikes.com/presse/Dossier%20de%20press%20B2C%20Mhikes%202017.pdf)
- Easy Mountain. (s.d. 2018). *Index*. Récupéré sur Mhikes: <https://mhikes.com/>
- EN, N. (2017, février 12). *Unity Mobile GPS - Real World Location - Unity 3D [Tutorial]*. Récupéré sur Youtube: <https://www.youtube.com/watch?v=g04jaC-Tpn0>
- Geig, M. (s.d. 2018). *Object Pooling*. Récupéré sur Unity3D: <https://unity3d.com/fr/learn/tutorials/topics/scripting/object-pooling>
- Google. (2018, mai 8). *ARCore*. Récupéré sur Developers Google: <https://developers.google.com/ar/discover/>
- Google. (2018, mai 8). *Concept*. Récupéré sur Developers Google: <https://developers.google.com/ar/discover/concepts>
- La Redaction. (2018, avril 23). *Framework ou infrastructure logicielle : définition et traduction*. Récupéré sur JDN: <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203355-framework/>
- LavishT@TWC. (2017, mai 5). *What is Xamarin*. Récupéré sur TheWindowsClub: <https://www.thewindowsclub.com/what-is-xamarin-and-cross-platform-mobile-development>
- Mapbox. (s.d. 2018). *Augmented reality*. Récupéré sur Mapbox: <https://www.mapbox.com/augmented-reality/>
- Mapbox. (s.d. 2018). *SDK*. Récupéré sur Mapbox: <https://www.mapbox.com/unity-sdk/overview/>
- Marin, J. (2016, juillet 12). *Niantic, l'ex-start-up de Google derrière le succès de Pokemon Go*. Récupéré sur Siliconvalley blog: <http://siliconvalley.blog.lemonde.fr/2016/07/12/niantic-la-petite-start-up-derriere-le-succes-de-pokemon-go/>

Marin, J. (2016, juillet 13). *Pokémon Go, l'heure de gloire pour la start-up Niantic*. Récupéré sur L'Echo:

[https://www.lecho.be/entreprises/divertissement/Pokemon\\_Go\\_l\\_heure\\_de\\_gloire\\_pour\\_la\\_start\\_up\\_Niantic/9788057.html](https://www.lecho.be/entreprises/divertissement/Pokemon_Go_l_heure_de_gloire_pour_la_start_up_Niantic/9788057.html)

michaelzaletel. (s.d. 2018). *ARFormsSample*. Récupéré sur GitHub:  
<https://github.com/michaelzaletel/ARFormsSample>

Mobileprocessing. (s.d. 2018). *Using Geolocation and Compass*. Récupéré sur Mobileprocessing:  
<https://www.mobileprocessing.org/geolocation.html>

Motive.io. (s.d. 2018). *Augmented reality*. Récupéré sur Motive.io:  
<https://www.motive.io/features/augmented-reality/>

Niantic Inc. (s.d. 2018). *Niantic*. Récupéré sur Nianticlabs: <https://www.nianticlabs.com/>

O'Brien, L. (2017, septembre 19). *Augment Reality and ARKit with Xamarin and iOS 11*. Récupéré sur Blog Xamarin: <https://blog.xamarin.com/augment-reality-xamarin-ios-11/>

Pokémon. TM, Nintendo. (s.d. 2018). *Pokémon*. Récupéré sur Pokémon:  
<https://www.pokemon.com/fr/>

Pokémon. TM, Nintendo. (s.d. 2018). *Pokémon Go*. Récupéré sur Pokémon:  
<https://www.pokemon.com/fr/app/pokemon-go/>

PTC Inc. (s.d. 2018). *All of it*. Récupéré sur Developer Vuforia: <https://developer.vuforia.com/>

PTC Inc. (s.d. 2018). *Features*. Récupéré sur Vuforia: <https://www.vuforia.com/features.html>

PTC Inc. (s.d. 2018). *Image Target*. Récupéré sur Library Vuforia:  
<https://library.vuforia.com/articles/Training/Image-Target-Guide>

PTC Inc. (s.d. 2018). *Engine*. Récupéré sur Vuforia: <https://www.vuforia.com/engine.html>

ThinkMobiles. (s.d. 2018). *Quel prix pour une application en réalité virtuelle ?* Récupéré sur ThinkMobiles: <https://thinkmobiles.com/fr/blog/cout-application-realite-virtuelle/>

Unity Technologies. (2018, juillet 9). *Coroutine*. Récupéré sur Docs Unity:  
<https://docs.unity3d.com/Manual/Coroutines.html>

Unity Technologies. (2018, juillet 9). *Lerp*. Récupéré sur Docs Unity3D:  
<https://docs.unity3d.com/ScriptReference/Quaternion.Lerp.html>

- Unity Technologies. (2018, juillet 9). *Prefabs*. Récupéré sur Docs Unity3D: <https://docs.unity3d.com/Manual/Prefabs.html>
- Unity Technologies. (2018, juillet 9). *QuaternionsAndEuler*. Récupéré sur Docs Unity3D: <https://docs.unity3d.com/Manual/QuaternionAndEulerRotationsInUnity.html>
- Unity Technologies. (2018, juillet 9). *Scenes*. Récupéré sur Docs Unity3D: <https://docs.unity3d.com/Manual/CreatingScenes.html>
- Unity Technologies. (2018, juillet 9). *Slerp*. Récupéré sur Docs Unity3D: <https://docs.unity3d.com/ScriptReference/Quaternion.Slerp.html>
- Unity Technologies. (2018, juillet 9). *UICanvas*. Récupéré sur Docs Unity3D: <https://docs.unity3d.com/Manual/UICanvas.html>
- Unity Technologies. (s.d. 2018). *All of it*. Récupéré sur Docs Unity: <https://docs.unity3d.com>
- Unity Technologies. (s.d. 2018). *Unity*. Récupéré sur Unity3D: <https://unity3d.com/fr/unity>
- Unity Technologies. (s.d. 2018). *Unity*. Récupéré sur Unity3D: <https://unity3d.com/fr>
- Unity Technologies. (s.d. 2018). *Unity Personal*. Récupéré sur Unity store: [https://store.unity.com/fr/products/unity-personal?\\_ga=2.102841288.919698751.1532009361-1060898543.1532009361](https://store.unity.com/fr/products/unity-personal?_ga=2.102841288.919698751.1532009361-1060898543.1532009361)
- Unity Technologies. (s.d. 2018). *Vuforia*. Récupéré sur unity3D: <https://unity3d.com/fr/partners/vuforia>

## 10. Annexe I : Configuration de Vuforia

Vuforia est le centre de la réalité augmentée au sein de l'application Charlotte la Marmotte. Pour utiliser cette extension, il faut la télécharger sur le site officiel de Vuforia pour les développeurs (un compte gratuit est requis) :

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

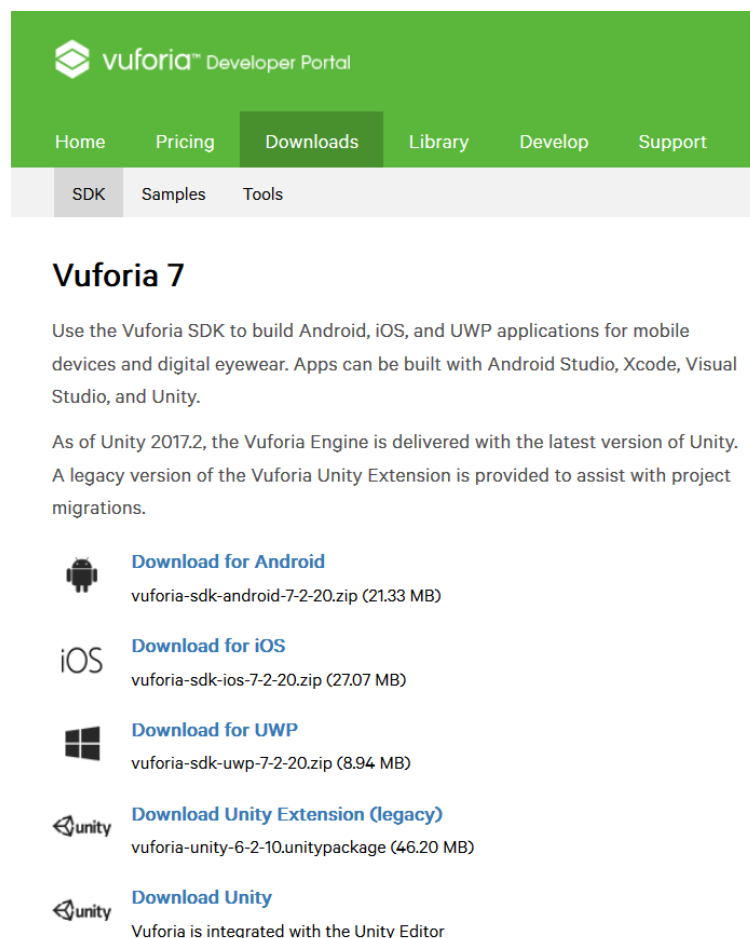


Figure 69 : Vuforia configuration, téléchargement<sup>70</sup>

Il est possible de télécharger uniquement le plugin si Unity est déjà installé sur la machine, sinon il est également faisable de télécharger tout Unity avec l'extension déjà installée.

Pour installer uniquement le plugin, après son téléchargement, il suffit de lancer le :

`vuforia-unity-*.*.unitypackage`

<sup>70</sup> Source : <https://developer.vuforia.com/downloads/sdk>

Cela va ouvrir Unity s'il n'est pas déjà ouvert et une fenêtre d'import apparaîtra, après avoir tout sélectionné un bouton « import » permet d'importer le package Vuforia dans votre projet Unity.

Une fois Vuforia importé dans Unity il faut le configurer à partir de :

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

Dans la section « Develop » se trouvent les projets déjà créés ainsi que leurs licences.

Name	Type	Status	Date Modified
CharlotteLaMarmotte	Develop	Active	May 04, 2018 11:02
TrumpAR	Develop	Active	May 02, 2018 14:43

Figure 70 : Vuforia configuration, License Manager<sup>71</sup>

Le bouton « Get Development Key » permet d'obtenir des clés de licences gratuites. Des versions payantes sont également disponibles.

Le projet créé, la clé de licence est accessible. Il faut la garder précieusement :

**CharlotteLaMarmotte** Edit Name Delete License Key

License Key Usage

Please copy the license key below into your app

**Type:** Develop  
**Status:** Active  
**Created:** May 04, 2018 11:02

**History:**  
 License Created - May 04, 2018 11:02

Figure 71 : Vuforia configuration, License Details<sup>72</sup>

<sup>71</sup> Source : <https://developer.vuforia.com/license-manager>

<sup>72</sup> Source : <https://developer.vuforia.com/targetmanager/licenseManager/licenseListingDetails>

La clé de licence obtenue, il faut maintenant créer la base de données avec les « Targets ». Pour ce faire, il suffit de se rendre sous « Target Manager » :

Database	Type	Targets	Date Modified
<a href="#">DataBaseCharlotteLaMarmotte</a>	Device	7	Jun 06, 2018 11:22
<a href="#">DataBaseTrumpAR</a>	Device	1	May 02, 2018 14:47

Figure 72 : Vuforia configuration, Target Manager<sup>73</sup>

Dans cette partie, il est possible d'accéder à ses bases de données de Targets ce qui est indispensable pour une application en RA. Pour créer une base de données, il suffit de presser le bouton « Add Database », d'y indiquer un nom et de sélectionner « Device » :

### Create Database

Name:

Type:

- ☒ Device  
☐ Cloud  
☐ VuMark

Cancel

Create

Figure 73 : Vuforia configuration, Création de la base de données<sup>74</sup>

Voici la base de données des Targets de Charlotte la Marmotte. On peut y apercevoir les différents éléments utilisés dans l'application :

<sup>73</sup> Source : <https://developer.vuforia.com/targetmanager/>

<sup>74</sup> Source : <https://developer.vuforia.com/targetmanager/>








## DataBaseCharlotteLaMarmotte [Edit Name](#)

Type: Device

Targets (7)

Add Target

Download Database (All)

<input type="checkbox"/> Target Name	Type	Rating	Status ▾	Date Modified
<input type="checkbox"/>  Tetras	Single Image	★★★★★	Active	Jun 06, 2018 11:22
<input type="checkbox"/>  Chamois	Single Image	★★★★★	Active	Jun 06, 2018 11:21
<input type="checkbox"/>  Gypaete	Single Image	★★★★★	Active	Jun 06, 2018 11:20
<input type="checkbox"/>  Bouquetin	Single Image	★★★★★	Active	Jun 06, 2018 11:20
<input type="checkbox"/>  Cerf	Single Image	★★★★★	Active	Jun 06, 2018 11:18
<input type="checkbox"/>  Aigle	Single Image	★★★★★	Active	Jun 06, 2018 11:14
<input type="checkbox"/>  CharlotteMarmotteFouly1	Single Image	★★★★★	Active	May 04, 2018 11:03

Last updated: Today 12:18 PM [Refresh](#)

Figure 74 : Vuforia configuration, Détails de la base de données<sup>75</sup>

Une fois la base de données complète, il faut la télécharger pour l'intégrer au projet Unity. Pour ce faire, un bouton « Download Database » est à disposition :

## Download Database

7 of 7 active targets will be downloaded

Name:

DataBaseCharlotteLaMarmotte

Select a development platform:

☐ Android Studio, Xcode or Visual Studio

☒ Unity Editor

Cancel

Download

Figure 75 : Vuforia configuration, Téléchargement de la base de données<sup>76</sup>

Une fois la base de données téléchargée, l'importer également dans Unity. La base de données doit apparaître dans le projet Unity, sous « Editor ». Le dossier doit contenir les images Targets :

<sup>75</sup> Source : <https://developer.vuforia.com/targetmanager/project/deviceTargetListing>

<sup>76</sup> Source : <https://developer.vuforia.com/targetmanager/project/deviceTargetListing>

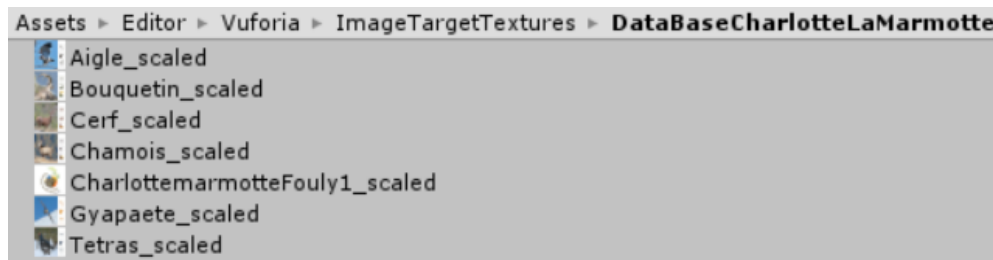


Figure 76 : Vuforia configuration, localisation de la base de données<sup>77</sup>

Les Targets bien importés dans le projet, il faut configurer notre outil. La configuration se situe dans :

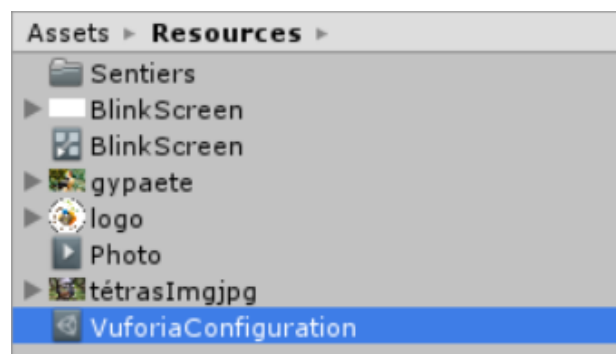


Figure 77 : Vuforia configuration, Localisation de VuforiaConfiguration<sup>78</sup>

Dans cet outil de configuration, il faut ajouter la clé de licence obtenue grâce au site de Vuforia sous la rubrique « App license Key ». De plus, la caméra devra charger la base de données des Targets pour les reconnaître. Cette action s'exécute dans la partie « Datasets » ; il faut la « load » et « activate ». Cet outil de configuration est valable pour toutes les cam  ras AR utilis  es au sein du projet.

<sup>77</sup> Source : Screenshot personnel de l'architecture du projet Unity CharlotteLaMarmotte

<sup>78</sup> Source : Screenshot personnel de l'architecture du projet Unity CharlotteLaMarmotte



**Vuforia**

App License Key

Delayed Initialization ☐

Camera Device Mode

Max Simultaneous Tracked Images

Max Simultaneous Tracked Objects

Load Object Targets on Detection ☐

Camera Direction

Mirror Video Background

**Digital Eyewear**

Eyewear Type

**Datasets**

Load DataBaseCharlotteLaMarmotte ☒

Activate ☒

**Video Background**

Enable video background ☒

Overflow geometry

Matte Shader

**Smart Terrain Tracker**

Start Automatically ☐

**Device Tracker**

Enable device pose tracking ☒

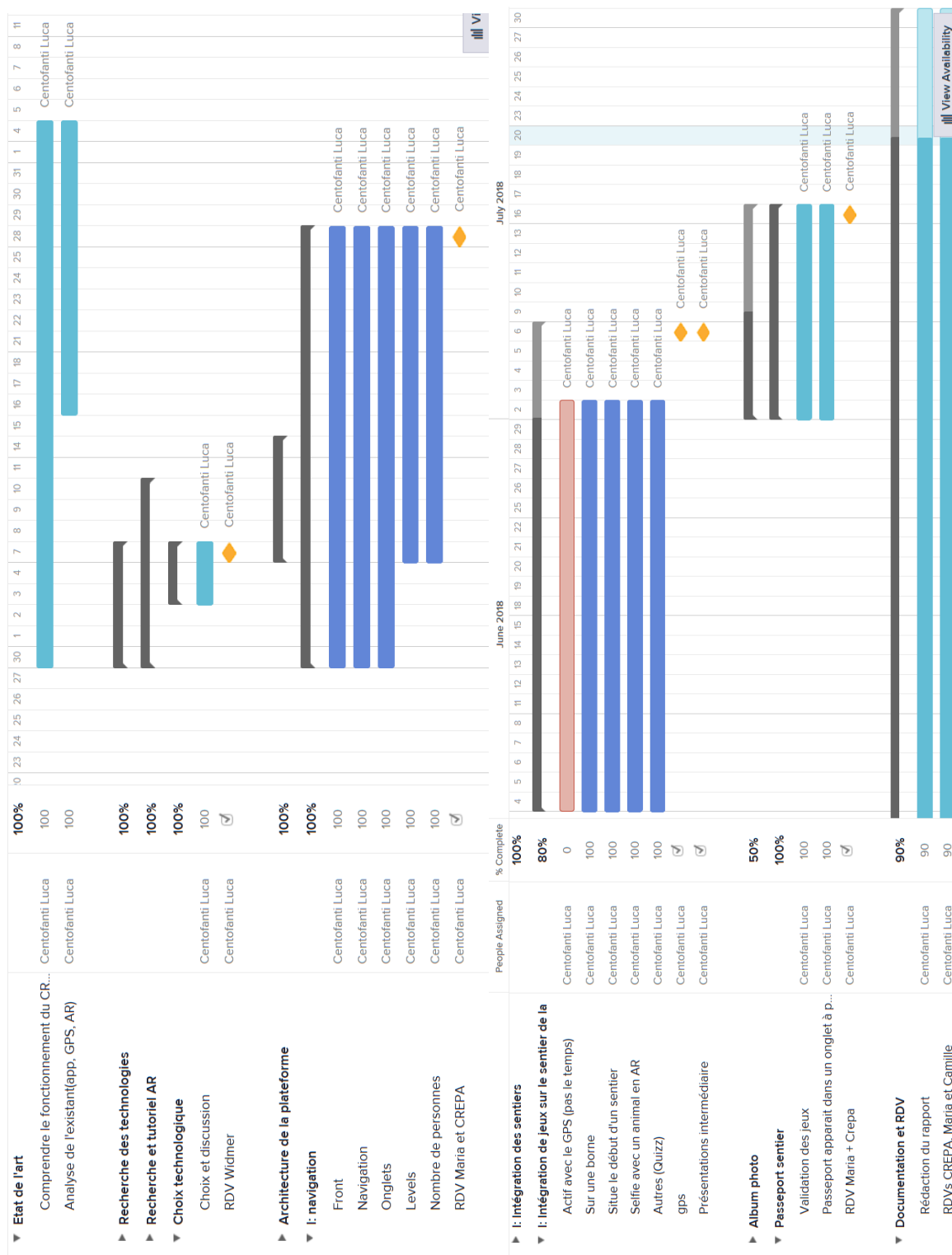
Enable prediction ☒

Model Correction Mode

Figure 78 : Vuforia configuration, Caméra RA configuration 2<sup>79</sup>

<sup>79</sup> Source : Screenshot personnel d'une caméra RA du projet Unity CharlotteLaMarmotte

## 11. Annexe II : Planification



## 12. Annexe III : Cahier des charges

### 12.1. INTRODUCTION

Le Centre Régional d'Etudes des Populations Alpines (CREPA) est une association fondée en 1990, son but est de poursuivre des recherches de la Commune de Bagnes et de les étendre dans les vallées proches. Le CREPA permet de développer des projets visant la formation des jeunes au patrimoine socioculturelle des communes avoisinantes (CREPA, s.d. 2018), qui sont :

Grand Entremont / Vallée du Trient / Fully

Ces projets peuvent être divers et variés mais l'objectif reste le même : créer des liens entre la population indigène et touristique, tout en accentuant l'identité et les richesses locales. (CREPA, s.d. 2018)

Mon travail de Bachelor se focalisera sur un projet du CREPA, « Charlotte la Marmotte », qui a été créé en 2012. Le rôle de ce projet est de faire découvrir au jeune public les particularités du patrimoine géographique alpin. Il est présent sur cinq sentiers :

La Fouly (2012), Champex-Lac (2014), La Creusaz (2014), Trient (2015) et Emosson

Actuellement, les enfants ont « seulement » un questionnaire à remplir durant leur périple sur l'un des cinq sentiers à disposition. Mon travail de Bachelor consistera à rendre les sentiers plus ludiques avec des technologies qui permettront aux enfants de s'amuser tout en ayant un complément d'information, mais également un aspect utilitaire pour les parents (description du sentier, carte, etc.) afin de faciliter leur périple.

## **12.2. PROBLEMATIQUE**

Les technologies utilisées pour les sentiers didactiques du CREPA auront certaines contraintes à respecter.

### **12.3. Les enfants**

Les technologies proposées ne devront en aucun cas remplacer le format papier mais simplement le compléter, afin que les enfants ne soient pas en permanence sur leur téléphone. Le but étant de faire découvrir la nature et tout ce qui nous entoure, tout en s'amusant.

### **12.4. Disponibilités**

Afin d'être accessible à tous, la solution proposée se doit d'être multiplateforme (Android, IOS). De plus, il faut prendre en compte que les sentiers se situent en dehors des villes ou des villages, c'est pourquoi il n'y aura pas forcément de réseaux téléphonique ou internet partout.

### **12.5. Innovation**

De la réalité augmentée devra être intégrée afin de fusionner le côté ludique avec l'apprentissage de nouvelles connaissances.

## **12.6. ETAPES DE REALISATION ENVISAGEES**

### **12.6.1. Analyse de l'état de l'art**

Durant cette étape, je dois comprendre le fonctionnement du CREPA en consultant les différents documents fournis par mes responsables du travail de Bachelor et à travers les rendez-vous que j'aurai avec le CREPA. De plus, je découvrirai le fonctionnement des sentiers en en pratiquant un.

Il me faudra également analyser des projets similaires, afin de me faire une idée du marché et de ce qui est réalisable ou non. Des solutions comportant des similitudes avec la réalité augmentée devront également être trouvées.

### **12.6.2. Choix technologique**

L'analyse des technologies adéquates à ce projet, telles que XAMARIN, UNITY, ARCore & ARKit et la réalité augmentée en générale, est indispensable. En effet, il est important de trouver la technologie qui répondra au mieux à toutes les contraintes imposées par le CREPA. Il serait également judicieux de trouver des solutions mélangeant la réalité augmentée et le GPS, afin de se faire une idée sur les technologies utilisées.

La solution se devra d'être légère et pouvant être utilisée sans connexion internet ou réseau téléphonique. C'est pourquoi une architecture compatible avec ces différentes contraintes devra être proposée.

### **12.6.3. Implémentation / validation**

Le module prioritaire identifié avec le partenaire (CREPA) devra être implémenté. Plusieurs éléments devront être pris en considération, tels que la navigation qui permettra de s'orienter dans l'application, les sentiers qui contiendront une description et une carte d'orientation, ainsi que des « jeux » pour chacun des sentiers.

Tous les jeux ne seront pas implémentés, le but étant de montrer ce qui est possible de faire. Il serait aussi judicieux de marquer un sentier comme « réussi » une fois que tous les « jeux » du sentier seront effectués, afin de voir une avancée dans la solution.

La création d'un album photos est optionnelle. Elle dépendra du temps restant à disposition.

Il me faudra également valider et tester la solution. Je le ferai bien entendu moi-même mais j'aimerais également faire participer des personnes externes, telles que les personnes qui encadrent mon projet, mes collègues ou ma famille.

## 12.7. PLANIFICATION

N°	Description des tâches	Commentaires	Date d'échéance
1	Cahier des charges		14.05.2018
2.1	Comprendre le fonctionnement du CREPA (les sentiers, les besoins)	Tâches en parallèle	04.06.2018
2.2	Analyse de l'existant (type d'application, GPS, AR)		
3.1	Recherche des technologies	(XAMARIN & UNITY)	07.05.2018
3.2	Recherche et Tutoriel sur l'AR	ARCore & ARKit	07.05.2018
3.3	Choix technologique	UNITY, C#	07.05.2018
4	Architecture de la plateforme	<ul style="list-style-type: none"> <li>- Simple</li> <li>- Légère</li> <li>-Fonctionnant hors réseau</li> </ul>	14.05.2018
5	Conception de la navigation d'un projet Milestone 04.06.2018	<ul style="list-style-type: none"> <li>- Les onglets</li> <li>- Les niveaux</li> <li>-1 personne ou en famille</li> </ul>	28.05.2018
6.1	Intégration des sentiers	Description et image	14.05.2018
6.2	Intégration de jeux sur le sentier de la Fouly	Validation des concepts : <ul style="list-style-type: none"> <li>- S'active avec le GPS</li> <li>- Sur une borne</li> <li>-Situe le début d'un sentier</li> <li>- Selfie avec un animal en AR</li> <li>- Autre en fonction du CREPA</li> <li>-Affichage du passeport</li> </ul>	02.07.2018
7	Conception d'un album photo	Optionnel	

8	Remise du travail final		30.07.2018
9.1	Rédaction du rapport final	A faire tout au long du TB	En continu
9.2	Documentation technique		
9.3	Rendez-vous (CREPA, Maria et Camille)		

## **13. Déclaration de l'auteur**

Je déclare, par ce document, que j'ai effectué le travail de Bachelor ci-annexé seul, sans autre aide que celles dûment signalées dans les références, et que je n'ai utilisé que les sources expressément mentionnées. Je ne donnerai aucune copie de ce rapport à un tiers sans l'autorisation conjointe du RF et du professeur chargé du suivi du travail de Bachelor, y compris au partenaire de recherche appliquée avec lequel j'ai collaboré, à l'exception des personnes qui m'ont fourni les principales informations nécessaires à la rédaction de ce travail et que je cite ci-après : Mme Maria Sokhn.



