

**Institut Supérieur de l'Électronique et du Numérique**

Tél. : +33 (0)2.98.03.84.00

Fax : +33 (0)2.98.03.84.10

20, rue Cuirassé Bretagne

CS 42807 - 29228 BREST Cedex 2 - **FRANCE**

**« Outil de géolocalisation des souvenirs »**



Proposé par : « Cabon Mikael »

Thématique : Informatique

« Sechet Hugo »

Domaine professionnel : Ingénierie de Projet et d’Affaire

« L’Hostis Pierre »

Domaine professionnel : Ingénierie de Projet et d’Affaire

# Remerciements

Tout d'abord, nous souhaitons remercier notre responsable de projet, Mikael Cabon, qui nous a accompagné tout au long de ce premier projet en conditions d'entreprise en nous indiquant la bonne voie à suivre et en se plaçant à la fois dans le rôle de tuteur et de client, ajoutant un réalisme et un professionnalisme d'autant plus grand à ce projet.

Nous souhaitons également remercier Cyril de Sousa Cardoso qui a placé sa confiance en nous pour ce projet dont l'idée d'origine lui est venue il y a plusieurs années, renforçant l'importance de cette confiance. Nous le remercions également pour son expérience en tant qu'entrepreneur qu'il nous a transmise tout au long de ces quelques mois de projet et qui nous aidera sans aucun doute dans le futur.

Nous tenions également à remercier toutes les personnes qui nous entourent et nous ont aidé, à plus ou moins grande échelle sur ce projet, ainsi que pour le soutien moral et technique que certains nous ont fournis.

# Résumé

Dans le cadre du projet M1 de l'année 2022-2023, le sujet traité au sein de ce rapport a comme intitulé « Outil de géolocalisation des souvenirs » et est proposé par Mikael Cabon et Cyril De Sousa Cardoso.

Ce projet est basé sur le développement informatique d'un outil pouvant apparaître sous la forme d'une application mobile ou d'une site web, sur le modèle d'un réseau social. Ce réseau social a pour objectif de différer avec ceux déjà existants par le fait de recréer un lien plus intime entre les utilisateurs en se basant sur les émotions partagées issues des souvenirs. L'idée principale est le partage de médias (une photo, une vidéo, ...) par un utilisateur à un ou plusieurs autres, mais uniquement accessibles sous conditions géographiques, temporelles ou autres. Ces médias seraient porteurs d'un souvenir que l'utilisateur aurait vécu, avec les personnes à qui il l'a partagé ou non. Le cœur de l'outil est donc basé sur les émotions ressenties lorsqu'elles sont associées à un souvenir, aidant ainsi à un rapprochement entre les personnes concernées et à des rapports plus directs, plus humains que ce que les réseaux sociaux actuels proposent.

# Abstract

The main point of a project as the one we present in this document is to combine user-friendliness, sustain over time and the faculty to share emotions through a digital tool. We identified that a mobile application was the best choice for a project like this one. After investigations, we have chosen to use a coding language which appeared to be the main one for mobile application for few years and derived from Java: Kotlin. We also have chosen to use the Google Maps service which appeared to fit perfectly with the other systems we were using. With the combination of Kotlin and Android Studio which offers many options of creativity, we were able to create an application with an elegant style and a great ease of use for the users who want to share their memories.

# Mots-Clés

*Mots-clés*

*Définition*

<i>Kotlin</i>	Langage de programmation dérivé du Java
<i>Android Studio</i>	Environnement spécialisé dans le développement d'applications mobiles
<i>Geofencing ou géo repérage</i>	Technique de géolocalisation de zone basée sur l'entrée ou la sortie d'une personne dans un région définie
<i>Géolocalisation active</i>	Technique de géolocalisation repérant une personne lorsqu'une application particulière est ouverte dans son appareil mobile
<i>Géolocalisation passive</i>	Technique de géolocalisation opposée à l'active dont le principe est un repérage permanent de l'appareil peu importe les applications actives

## Table des matières

<i>Remerciements</i> .....	1
<i>Résumé</i> .....	2
<i>Abstract</i> .....	2
<i>Mots-Clés</i> .....	3
<i>Introduction</i> .....	6
<i>La Gestion du Projet</i> .....	7
<i>Développement de l'outil</i> .....	11
<i>Recherches</i> .....	12
<i>Réalisation de l'outil de géolocalisation des souvenirs</i> .....	16
<i>Fonctionnalités et améliorations futures</i> .....	30
<i>Le Web 3.0</i> .....	33
<i>L'aspect social de l'application</i> .....	36
<i>La dimension éthico-légale</i> .....	38
<i>Conclusion</i> .....	40
<i>Bibliographie</i> .....	41
<i>Annexes</i> .....	43

## Table des Figures

Figure 1 : Interface de l'application "GéoMémoire" .....	12
Figure 2 : Paul Ekman.....	13
Figure 3 : Comparatif d'utilisation de Kotlin et Java pour Android .....	14
Figure 4 : Interface Android Studio .....	14
Figure 5: Illustration du fonctionnement du Geofencing.....	15
Figure 6: Exemple de syntaxe d'un code Kotlin .....	16
Figure 7: Tables de la base.....	17
Figure 8: FirebaseAuthification.....	18
Figure 9 : Barre d'action.....	19
Figure 10: Page d'accueil avec le bouton de déconnexion.....	19
Figure 11: Page de connexion .....	20
Figure 12: Comparatif mot de passe caché/visible .....	21
Figure 13: Page d'inscription.....	21
Figure 14: Démonstration des messages d'erreur.....	22
Figure 15: Page de description d'un souvenir .....	23
Figure 16 : Page de profil utilisateur.....	24
Figure 17: Barre de recherche sur la carte .....	25
Figure 18: Souvenir sur la carte dans le rayon défini .....	26
Figure 19 : Notification reçue lorsqu'un souvenir est à portée .....	26
Figure 20 : Page d'ajout d'un souvenir .....	27
Figure 21: Diagramme Bête à Cornes d'analyse du besoin utilisateur .....	43
Figure 22 : Diagramme Pieuvre .....	43
Figure 23: Exemple de diagramme de Gantt .....	44
Figure 24 : Compte-rendu de réunion .....	45
Figure 26: Repère de la position actuelle.....	46
Figure 25: Souvenirs avec leurs coordonnées longitude/ latitude .....	46
Figure 27 : Souvenir depuis sa localisation actuelle .....	47

# Introduction

Depuis plusieurs années, l'idée d'un réseau social qui contribuerait à recréer un réel lien social entre les gens a émergé dans l'esprit d'entrepreneurs. La plupart des réseaux sociaux actuels incitent les utilisateurs à toujours se montrer et se mettre en avant de la manière la plus idéalisée possible, quitte à déformer la réalité. L'objectif est ainsi de rassembler les gens en se basant sur une caractéristique humaine : les émotions. « *L'outil de géolocalisation des souvenirs* » se base sur les émotions ressenties au moment de l'évocation ou de la vision de souvenirs. En ce qui concerne la géolocalisation, le concept est d'associer un média symbolisant un souvenir, que ce soit une photo, une vidéo, un message audio ou autre, avec une localisation précise et/ou une période donnée pendant laquelle le souvenir sera visionnable par la ou les personnes concernées aux localisations indiquées.

Le rapport de ce projet abordera donc dans un premier temps le cahier des charges incombant à cet outil de géolocalisation, suivi des outils et techniques mises en place en termes de gestion. Dans un second temps sera abordé l'état des arts et les recherches effectuées en amont du développement concret de l'outil qui constituera le cœur de ce rapport. Différents points sur le développement des aspects juridiques et économiques en perspective d'une continuation du projet à l'avenir précéderont enfin la conclusion de ce rapport.

# La Gestion du Projet

## Cahier des charges

### *Intitulé du projet : Outil de géolocalisation des souvenirs*

#### Contexte du projet :

Nous vivons aujourd'hui dans une société où la communication entre personnes se simplifie et les outils utilisables pour cela se font légion.

C'est dans le cadre du projet M1 de fin d'année que l'idée d'un nouvel outil prônant le rapprochement entre les gens via une géolocalisation de souvenirs partagés entre individus a émergé.

En effet, l'intention à l'origine de ce projet est d'aller à l'encontre de ce phénomène de « distanciation » de la population à cause des réseaux sociaux et de créer un outil se basant sur les émotions que procurent un souvenir partagé afin de recréer des liens sociaux réels entre individus.

#### Objectifs du projet :

En vue d'un développement plus poussé, l'objectif du projet est ici d'établir une première base ainsi que les différentes possibilités qui s'offrent au développement d'un outil basé sur la géolocalisation comme celui-ci.

L'objectif principal du projet est ainsi la création d'un outil de communication, d'interaction et de partage sur lequel les utilisateurs pourront rétablir un lien plus puissant, plus intimes entre eux en se basant sur les émotions que des souvenirs communs peuvent engendrer. Souvenirs qui dans le concept initial ne seront accessibles qu'en se rendant à l'endroit auquel il sera associé. Ce modèle permettra donc de sortir du modèle actuel étant la publication plus large de contenu « idéalisé » ou en tout cas orienté pour le plus grand nombre.

Le but est également de répondre aux exigences actuelles et futures en matière de sécurité des données et d'ergonomie. L'objectif est le développement d'un outil permettant à chaque utilisateur d'avoir la main mise sur ses propres données et ainsi le pouvoir de les faire disparaître ou de les modifier à tout moment sans qu'aucun organisme extérieur puisse les utiliser à son compte. Dans l'aspect ergonomique, un style épuré et simple sera de rigueur afin d'être en accord avec la facilité de prise en main nécessaire à ce type d'outil.

#### Périmètre du projet :

Le projet présente beaucoup d'ambition dans sa conception et doit respecter plusieurs contraintes dans son fonctionnement. La première limite rencontrée sera donc une limite de temps. La période étant de quatre mois, certains aspects de l'outil notamment la base et les premières fonctions primordiales à son fonctionnement sont la priorité de développement et devront être fonctionnelles dans la mesure qu'une première version doit montrer le principe global de l'outil. Certaines mesures moins établies dans le pur fonctionnement de l'outil comme la propriété des données totale pour les utilisateurs seront ainsi traitées dans un second temps uniquement si le délai restant le permet.

Étant un outil de géolocalisation, la question d'une limite géographique du moins au début pourrait se poser. Le développement se centrerait donc sur une zone plus restreinte comme la Bretagne ou la France mais sans doute pas directement sur le Monde dans son entièreté.

L'aspect du respect des données des utilisateurs est également une contrainte car la plupart des outils, notamment de géolocalisation, dépendent souvent du partage de données de localisation vers de plus grandes structures fournissant le service géographique.

### Analyse fonctionnelle :

*FP : Fonctions principales*

*FC : Fonctions de contraintes*

*FP / Publier du contenu média sur l'application*

- Publier une image
- Publier un audio
- Publier une vidéo
- Publier du texte

*FP / Associer un contenu média à une localisation*

- Pouvoir communiquer l'emplacement du média avec d'autres utilisateurs
- Placer un repère sur une carte interactive
- Définir une adresse précise à l'écrit

*FC / Communiquer avec les autres utilisateurs :*

- Utiliser une messagerie texte
- Réagir aux publications à l'aide d'une photo prise sur le moment
- Réagir à l'aide d'émotes

*FC / Pouvoir créer un compte, s'identifier et se déconnecter*

- Créer un compte à l'aide d'un identifiant mail et d'un mot de passe, ainsi que son prénom et nom
- Se connecter avec ses identifiants dans un onglet dédié
- Se déconnecter de son compte

*FP / Accéder aux publications des autres utilisateurs*

- Déterminer dans quel périmètre se trouve le souvenir
- Déterminer la proximité du souvenir
- Recevoir une notification d'accès au souvenir une fois à proximité
- Visualiser le souvenir

*FC / Revoir les souvenirs débloqués*

- Afficher une liste des souvenirs que l'on a débloqués précédemment
- Définir ses souvenirs favoris
- Classer ses souvenirs en fonction de la zone géographique, des personnes à leur origine ou autre

### Développement global :

- Détermination du format de l'outil : site web, application mobile
- Choix du langage de programmation dans le but d'un outil évolutif
- Choix du système de géolocalisation (passive, active ou géo repérage)
- Choix du distributeur du service de géolocalisation
- Choix de la plateforme

## Sécuriser les données de l'utilisateur :

- Assurer la sécurité des données dans la base
- Utiliser des technologies compatibles au Web 3.0 dont les caractéristiques principales sont : indépendance (disponible sur tous les types de support), universalité (sous l'emprise d'aucune marque ou fabricant) et accessibilité à tous.

## Contraintes :

### Compétences internes :

- Format du support de l'outil libre entre site web et application. Application est le format préférentiel cependant il n'y a pas de compétences dans la création d'applications au sein de l'équipe.
- Possibilité de réaliser le projet dans un langage orienté objet comme Java, langage déjà maîtrisé par l'équipe.
- La sécurisation des données, particulièrement de la base de données, ne fait pas partie des compétences premières internes à l'équipe.

## Méthode QOOQCP :

- Quoi ?  
Applicatif permettant une interaction entre les utilisateurs permettant la transmission de souvenir et d'émotions via un outil de géolocalisation, sous certaines conditions géographiques, temporelles, recherches...
- Qui ?  
Le particulier civil, de tout âge, mais aussi des institutions, des villes, ou tout organisme voulant créer des liens entre lieu et souvenirs / informations
- Où ?  
  
Accessible à partir de n'importe quel point sur le globe à condition d'avoir accès à un outil de communication mobile.
- Quand ?  
  
En continu afin d'être accessible par tout utilisateur concerné, dans la durée définie par l'utilisateur ayant publié le contenu

- Comment ?

En réalisant un outil accessible et maîtrisable facilement pour le plus grand nombre, pouvant être en fonction n'importe où et à n'importe quel moment.

- Pourquoi ?

Afin de rapprocher les gens entre eux là où les réseaux sociaux actuels les éloignent, en se servant des émotions générées par des souvenirs.

### Diagramme Bête à Cornes :

L'utilité du diagramme de bête à cornes est d'identifier les besoins des utilisateurs en identifiant sur quoi agit le produit à réaliser, sur qui agit-il et quel est son objectif final. Cette étape précède généralement le cahier des charges fonctionnel.

[\*Cf Annexes : Figure 21\*](#)

### Diagramme Pieuvre :

Le diagramme pieuvre est un outil contribuant à l'analyse fonctionnelle. Il sert à représenter de manière graphique certaines parties du cahier des charges comme l'identification des fonctions principales ou secondaires.

[\*Cf Annexes : Figure 22\*](#)

### Diagramme de Gantt :

Le diagramme de Gantt est un graphique servant à la prévisualisation de la réalisation des différentes tâches d'un projet en montrant leur répartition dans le temps et dans l'équipe. Cet outil de gestion permet ainsi d'optimiser le temps alloué au projet, de savoir en continu à quel niveau d'avancement se situe ce dernier et quelles fonctionnalités ont été réalisées ou sont en cours de réalisation.

[\*Cf Annexes : Figure 23\*](#)

### Comptes-rendus de réunion :

Régulièrement, tout au long des quinze semaines qui ont constitué ce projet, des réunions de mise au point étaient organisées avec le professeur tuteur ainsi qu'avec l'entrepreneur de la tech nous accompagnant. Ces réunions avaient principalement pour but de vérifier que le projet avançait correctement dans la direction voulue par les « clients » représentés par les personnes précédemment citées. Elles avaient également comme objectif de discuter de certaines fonctionnalités, prévues à l'origine ou non, des perspectives d'avenir du projet ou encore de ce que l'outil devait apporter humainement dans son fonctionnement ce qui est un aspect intrinsèque à ce projet en particulier. Durant ces réunions, la rédaction de comptes-rendus était ainsi de rigueur afin de garder une trace écrite de ces discussions capitales pour la suite du développement.

[\*Cf Annexes : Figure 24\*](#)

# Développement de l'outil

## État des arts :

En amont du développement, un état des arts fut à réaliser. Ce dernier consiste à chercher et observer le fonctionnement d'éléments en accord avec le projet déjà existant dans le but d'étudier les différentes possibilités disponibles, aborder le projet avec de nouveaux points de vue, de nouvelles idées. Dans le cadre de l'outil de géolocalisation des souvenirs à réaliser au sein de ce projet, les principales dimensions à étudier étaient les réseaux sociaux déjà existant ainsi que les différentes plateformes mettant en place de la géolocalisation.

## Réseaux sociaux :

Le premier aspect à explorer était celui du contact social entre individus sur les réseaux. De quelle manière opèrent les géants de ce secteur (Instagram, Snapchat, Facebook, ...) et comment essaient-elles de mettre les utilisateurs en relation fut la première question posée afin d'établir cet état des arts. Ces différents réseaux passent principalement par la publication de contenu photo et vidéo, sans aucune contrainte dès lors que le contenu ne présente pas d'aspect choquant visuellement. Ce manque de contraintes va à l'encontre de l'idée de notre projet dont les publications ont des contraintes géographiques et temporelles dans l'objectif de relier les utilisateurs à travers le contenu mais également le lieu associé.

Le réseau social majeur approchant le concept de ce projet est Snapchat. Snapchat est un réseau social où la communication privée est ainsi plus mise en avant qu'Instagram par exemple qui prône la publication de contenu adressé au plus grand nombre et mettant en avant l'utilisateur. Cela se rapproche donc de la dimension plus « intime » que ce projet veut afficher. De plus, la « carte Snapchat » est un outil de géolocalisation en temps réel de ses amis sur le réseau social donc l'application regroupe les caractéristiques principales du projet.

En recherchant plus particulièrement des réseaux sociaux centrés sur la géolocalisation, l'un d'entre eux, Zenly, se rapproche tout particulièrement du fonctionnement du projet. Ce réseau social français racheté par Snapchat permettait à ses utilisateurs, avant sa fermeture en 2023, de partager sur une carte des lieux marquants pour eux comme des restaurants, des magasins, et autres... avec ses amis sur le réseau. Le principe est donc similaire au détail prêt qu'ici c'est le partage de lieux publics qui sont mis en avant et non le partage de souvenirs personnels.

## Systèmes utilisant la géolocalisation :

Dans le but d'observer un comportement de la géolocalisation se rapprochant de l'utilisation faites par l'outil développé durant ce projet, une application s'est démarquée. « *GéoMémoire* » est une application servant à maintenir la mémoire de soldats français disparus durant la seconde guerre mondiale en associant le lieu où se trouve leur sépulture avec une fiche décrivant leur vie et leurs actions durant la guerre, accessible en se rendant sur les emplacements correspondants.





Figure 2 : Paul Ekman

## Support de l'outil : L'Application mobile

Concernant le support de cet outil de géolocalisation, aucune contrainte n'était imposée. Cela ouvrait ainsi à l'équipe de développement plusieurs possibilités. Premièrement, l'option du site web fut rapidement mise à l'écart pour différentes raisons : l'accès unique via un site web n'était pas optimisé pour un outil de type « réseau social » qui se doit d'être accessible facilement, préférablement de façon mobile. Ce format est également plus restrictif pour la mise en place de certaines fonctionnalités comme la présence de notifications. Globalement, un site web souffre d'un manque de fluidité d'utilisation qui porte préjudice à un projet comme celui-ci.

Suite à cette réflexion, il a donc été décidé d'opter pour une application mobile. Ce choix est motivé par son accessibilité à tout instant, un contexte bien plus adapté à un outil mêlant géolocalisation et contact social et une diversité de fonctionnalités supérieure. Un aspect non négligeable ayant influencé ce choix est également le point de vue technique, l'équipe de développement étant plus à l'aise de développer sous un format orienté « objet » que dans un format web. Une application mobile peut également permettre la présence d'une géolocalisation en temps réelle via une autorisation de la part de l'utilisateur, ce qui représente un enjeu majeur pour le bon fonctionnement d'un outil dont l'interaction des personnes avec des zones géographiques est le cœur.

## Le choix du langage de développement : Kotlin

Une fois le support choisi, la question du langage de développement s'imposa. Parmi les différentes possibilités (Java, Python, C++, ...), le choix s'est porté sur Kotlin. Kotlin est un langage de programmation orienté objet et fonctionnel dérivé du Java créé par la société JetBrains il y a une dizaine d'années. Ce langage s'est affirmé année après année comme l'un des langages privilégiés pour le développement d'applications Android jusqu'en 2018 où il devient le langage préconisé par Google pour le développement d'applications Android. L'utilisation de Kotlin a été choisie en raison de sa simplicité d'utilisation et de compréhension combinée à la présence de nombreuses bibliothèques de fonctions comparables à celles de Java.

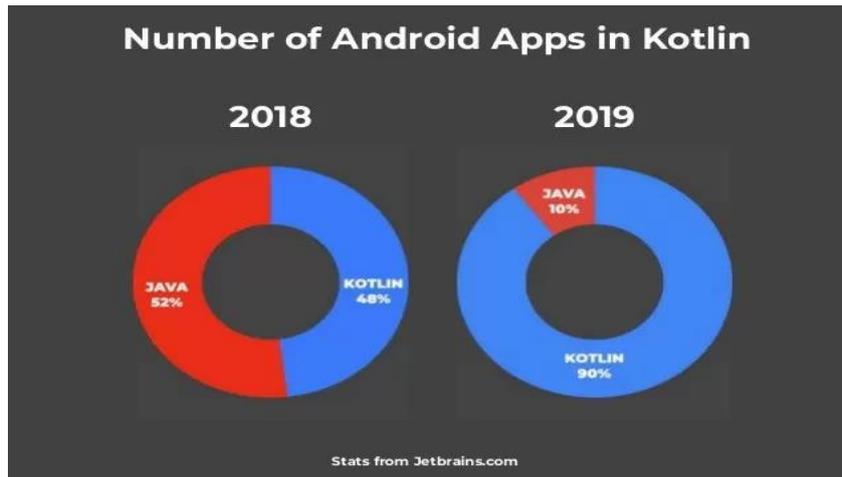


Figure 3 : Comparatif d'utilisation de Kotlin et Java pour Android

De plus, l'avènement de Kotlin comme langage de prédilection pour les applications Android fournit de nombreuses connexions avec d'autres services offerts par Google, rendant le développement plus fluide dans les interactions entre les services en question et le code source de l'application. Enfin, le développement de l'outil se doit d'être conçu pour que des développeurs futurs puissent agir sur l'application sans que cela ne soit trop compliqué en termes de langage qui serait trop obsolète ou non adapté dans quelques années. Ainsi, le Kotlin étant un langage en pleine expansion et soutenu par un géant de la tech, Google, celui-ci est en pleine expansion et demeurera donc sans risque l'un des langages de programmation majeurs pour les applications mobiles dans les prochaines années.

### Le choix de l'environnement de développement : Android Studio

Afin de mener la création technique de la manière la plus efficace possible, il fallait pour ce projet un environnement de développement offrant une simplicité d'interface pour du code en Kotlin, ainsi qu'une interface graphique afin de visualiser après chaque modification le résultat graphique de l'application. En réunissant ces informations, deux environnements se sont dégagés : IntelliJ IDEA, environnement créé par JetBrains pour leur propre langage qu'est Kotlin ou Android Studio, environnement issu de l'association de JetBrains avec Google qui recherchait une plateforme pour ce langage sélectionné comme étant la figure de proue du développement d'applications Android. Le choix s'est finalement porté sur Android Studio car l'avantage principale de cet environnement est sa connexion simplifiée avec différents services de Google, aidant ainsi au développement.

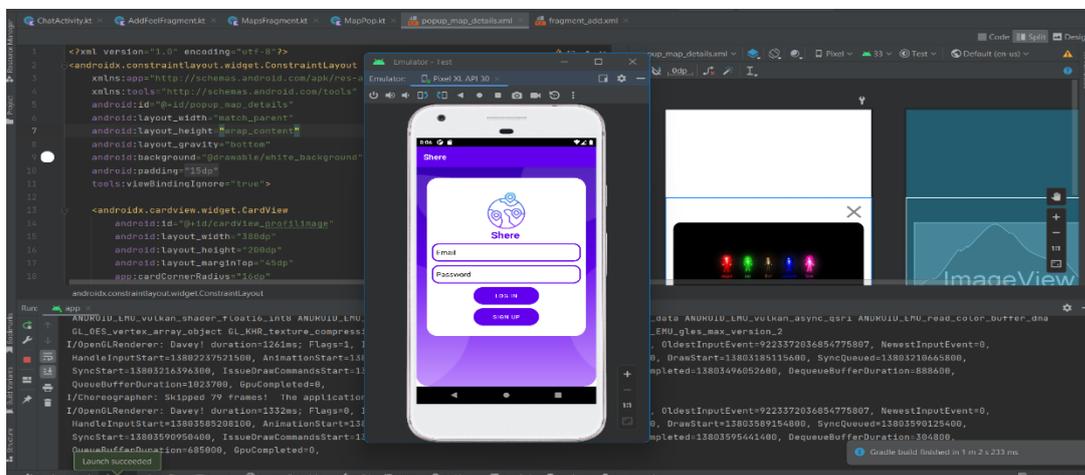


Figure 4 : Interface Android Studio

## Le choix de la plateforme : Android

Lors de la création d'une application mobile, la question de la plateforme est l'une des premières à se poser. En effet, les deux systèmes d'exploitation mobiles principaux, Android pour Google et iOS pour Apple, ne fonctionnent pas de la même manière et accueillent donc les nouvelles applications de façon différente au niveau des normes d'autorisation d'implémentation dans les différents « stores » ou du langage de programmation utilisé (les applications iOS utilisent Objective-C/Swift et celles Android utilisent Java ou ses dérivés comme Kotlin).

De plus, concernant iOS et Apple, chaque application se doit de passer par une phase de vérification avant d'être implémentée au sein du « store », phase prenant trois à quatre semaines. Pour ces raisons, le développement sur Android dans un premier temps a été choisi. Ce choix est également influencé par le fait que les machines Android sont celles les plus répandues parmi les utilisateurs de téléphone mobile.

## La technique de géolocalisation à employer : le Geofencing

Dans le but de mettre en place la dynamique de géolocalisation essentielle au projet, il a fallu aborder les différentes méthodes de géolocalisation existantes et sélectionner celle s'adaptant au mieux à cet outil et à ses contraintes. Ainsi, il y aurait trois types principaux de géolocalisation :

- La géolocalisation active, qui consiste à une localisation de la position et/ou des déplacements d'un utilisateur de manière constante une fois que la personne a donné son autorisation et ce jusqu'à ce que ce dernier change son paramètre de géolocalisation. La géolocalisation active est la méthode utilisée notamment dans les GPS, certaines applications de rencontre ou certains jeux.
- La géolocalisation passive, dont le principe se base sur l'enregistrement des déplacements de l'utilisateur sans que celui-ci n'interagisse avec une application. Ce mode de géolocalisation est celui qui crée le plus de débat car l'utilisateur ne choisit pas le moment où il est localisé, ce que le public considère comme étant trop intrusif vis-à-vis de la vie privée des utilisateurs.
- Enfin, le Geofencing ou géo repérage est un système mêlant dans son fonctionnement les géolocalisation actives et passives car elle se base sur de la localisation de zone. L'utilisateur va accepter d'être géolocalisé uniquement dans une zone définie et va recevoir des notifications dès l'instant où il entre dans la zone ou la quitte. Ce principe permet aux entreprises ou aux développeurs d'établir une connexion plus pertinente avec les utilisateurs.

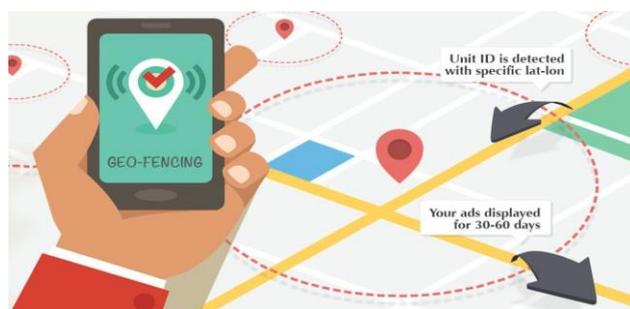


Figure 5: Illustration du fonctionnement du Geofencing

L'application créée durant ce projet est fondée sur l'association d'un souvenir avec une localisation. Cela est donc logique que le principe de géolocalisation sur lequel cet outil se basera soit le Geofencing. Le but est donc de créer des périmètres de géolocalisation autour des souvenirs, périmètres qui seront les seuls lieux où l'utilisateur sera localisé.

# Réalisation de l'outil de géolocalisation des souvenirs

## Apprentissage du langage et maîtrise de l'environnement :

Kotlin fut un choix de langage décidé majoritairement pour sa compatibilité avec la création d'applications mobiles et ses vastes bibliothèques de fonction. Cependant, même si l'équipe de développement maîtrise le Java, le Kotlin est différent sur plusieurs points notamment la syntaxe, l'appel de fonctions, l'interconnexion entre les fichiers source et autre. Il a donc fallu apprendre ce nouveau langage de programmation. Il est assez simple de s'auto former sur Kotlin en raison d'un grand nombre de formations notamment sur YouTube où l'on apprend rapidement à créer une application fonctionnelle. L'avantage des formations suivies est que certaines fonctionnalités mises en place dans l'outil final y furent présentées ce qui a permis un apprentissage réellement utile qui fit gagner du temps en plus des connaissances. Les tutoriels suivis sont également réalisés sur Android Studio, environnement de développement que l'équipe ne maîtrisait pas non plus à l'origine. Cela a permis de s'habituer à ce nouvel environnement, à son interface graphique et à la gestion des émulateurs permettant la visualisation de l'application.

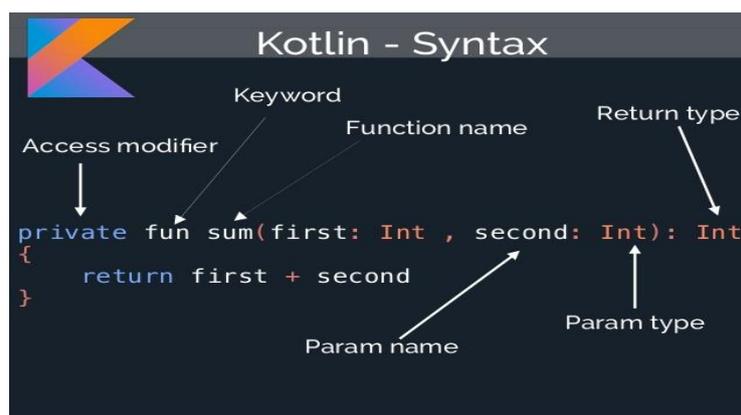


Figure 6: Exemple de syntaxe d'un code Kotlin

## Fonctionnement de la base de données :

Parmi les différents outils mis à disposition lorsque l'on crée une application via Android Studio, l'un des systèmes fournis par Google le plus utile est Firebase. Firebase, en particulier Firebase Realtime, est une base de données hébergée sur le cloud. Les données sont stockées et synchronisées en temps réel avec chaque client connecté à la base. L'utilisation de Firebase présente plusieurs avantages non négligeables. Premièrement, le fait que la base soit en temps réel implique une plus grande fluidité de l'édition des données car Firebase utilise la synchronisation des données. Ainsi, tous les appareils connectés à la base et donc exploitant une application comme celle développée lors de ce projet vont recevoir les données mises à jour en quelques millisecondes. Techniquement, cela entraîne le fait que la gestion du code réseau n'est pas nécessaire car Firebase l'exécute automatiquement.

Deuxièmement, les données sont conservées localement ce qui permet une réactivité en hors-ligne. Les données sont conservées jusqu'à la prochaine connexion lors de laquelle la base resynchronisera toutes les données qui ont été stockées en local durant la période hors-ligne. Enfin, l'avantage principal au sein d'un projet comme celui-ci est le fait que toutes les données soient stockées localement et non dans un serveur d'application concret comme il est plus commun d'observer. Au niveau de l'accès aux données, c'est le développeur qui fixe qui y a accès et peut les modifier, ce qui peut donner la possibilité d'accorder aux utilisateurs de changer, ajouter ou supprimer leurs données, ce qui est en accord avec le Web 3.0, l'une des contraintes liées à ce projet.

Le développement d'une application Kotlin utilisant la fonctionnalité de Google Maps nécessite l'utilisation d'une « API key », pour cela il est nécessaire de se créer un compte et une organisation depuis la page de Google Maps Platform pour créer une clé API unique pour le projet.

Voici la page Firebase Realtime Database où il est possible d'avoir une vue générale de la base de données liée au projet avec les messages enregistrés dans « chat », les souvenirs enregistrés dans « Emotions » et les utilisateurs enregistrés dans « user » :

Le Firebase Realtime Database est une base de données NoSQL qui stocke les données sous forme de paires clé-valeur. Les données sont stockées sous forme d'arbres JSON, ce qui facilite leur lecture et leur écriture. L'avantage principal de Firebase Realtime Database est qu'elle permet aux utilisateurs de synchroniser les données en temps réel entre les clients et les serveurs, sans avoir besoin de mettre en place une infrastructure de serveurs supplémentaire.

Le principe de fonctionnement de Firebase Realtime Database est assez simple. Les développeurs créent une référence à la base de données, puis enregistrent des données sous forme d'arbres JSON. Ces données sont ensuite synchronisées en temps réel sur tous les clients qui y accèdent. Les développeurs peuvent également utiliser des événements pour détecter les modifications de la base de données et effectuer des actions en conséquence.

Chat, Emotions et user sont en réalité des classes Kotlin qui prennent en paramètres différents arguments à passer à la base de données. Leurs données sont traitées à partir d'autres classes appelées adapter qui permettent de faire le lien entre les données, les classes et les bases.

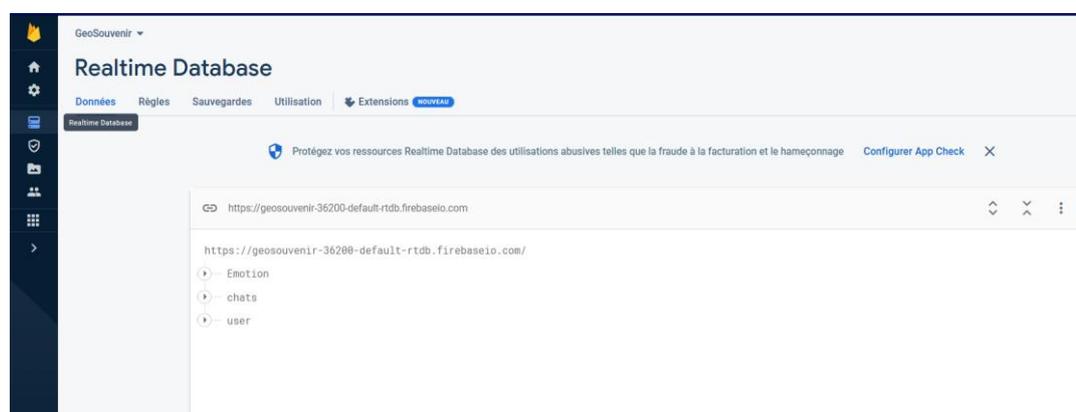


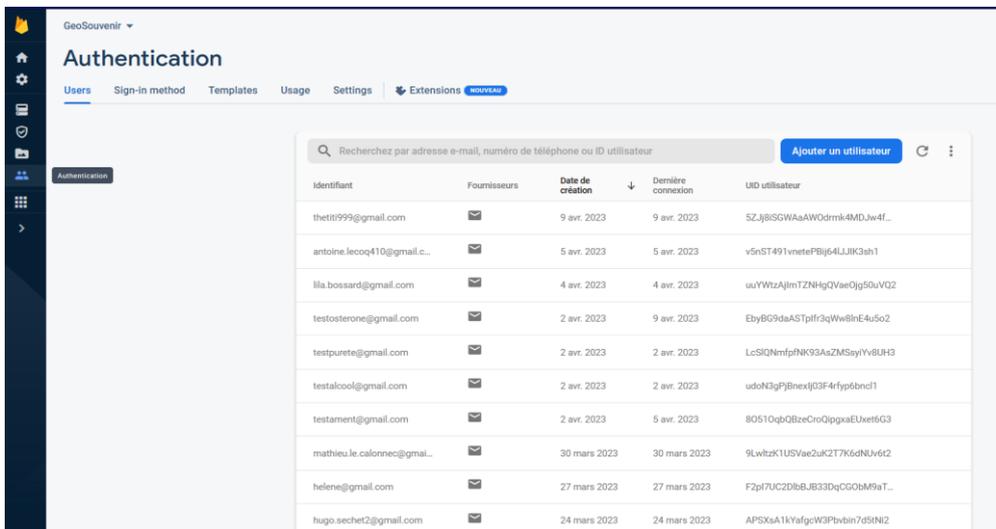
Figure 7: Tables de la base

Voici la page FirebaseBaseAuthentification où il est possible d'administrer et stocker les utilisateurs inscrits à notre application :

Firebase Authentication est un service qui permet aux développeurs de gérer l'authentification des utilisateurs de leur application. Ce service prend en charge l'authentification avec des comptes Google, Facebook, Twitter, GitHub, etc. Les développeurs peuvent également mettre en place une authentification par e-mail et mot de passe, ainsi qu'une authentification personnalisée.

Le principe de fonctionnement de Firebase Authentication est également assez simple. Les développeurs doivent configurer les options d'authentification dans la console Firebase, puis intégrer l'authentification dans leur application Android en utilisant les bibliothèque Firebase.

Les utilisateurs peuvent ensuite s'authentifier en utilisant les méthodes d'authentification prévues dans l'application, qui renvoient des jetons d'authentification pour permettre aux utilisateurs d'accéder aux ressources protégées de l'application



The screenshot shows the 'Users' tab in the Firebase Authentication console. It features a search bar at the top with the text 'Recherchez par adresse e-mail, numéro de téléphone ou ID utilisateur' and a blue button labeled 'Ajouter un utilisateur'. Below the search bar is a table with the following columns: 'Identifiant', 'Fournisseurs', 'Date de création', 'Dernière connexion', and 'UID utilisateur'. The table contains ten rows of user data.

Identifiant	Fournisseurs	Date de création	Dernière connexion	UID utilisateur
thetit999@gmail.com	📧	9 avr. 2023	9 avr. 2023	SZ_j8iSGWAAAWOdmk4MDJw4f...
antoine.lecoq410@gmail.c...	📧	5 avr. 2023	5 avr. 2023	v5nST491vnetePBijs4LJJK3sh1
lla.bossard@gmail.com	📧	4 avr. 2023	4 avr. 2023	uuYWtzAjImTZNHgQVaeOjg50uVQ2
testosterone@gmail.com	📧	2 avr. 2023	9 avr. 2023	EbyBG9daASTpfr3qWwblnE4u5o2
testpurete@gmail.com	📧	2 avr. 2023	2 avr. 2023	LcSIQNmfpRk93AsZMSyYv6UH3
testalcoool@gmail.com	📧	2 avr. 2023	2 avr. 2023	udoN3gpjBnxiJ03F4rfyp6bncl1
testament@gmail.com	📧	2 avr. 2023	5 avr. 2023	80510qb0BzeCro0pgxalEUvet6G3
mathieu.le.calonnec@gmal...	📧	30 mars 2023	30 mars 2023	9Lwltzk1USVae2ukZT7K6dNUv6t2
helene@gmail.com	📧	27 mars 2023	27 mars 2023	Fzpl7UCZDlbbJB33DqCCobM9aT...
hugo.sechet2@gmail.com	📧	24 mars 2023	24 mars 2023	AP5XsA1kYafgcW3Pwbin7d5nI2

Figure 8: FireBaseAuthentification

## Réalisation des fonctionnalités :

### ***La structure initiale :***

La MainActivity est une activité essentielle qui gère l'affichage de différents fragments de l'application via un menu de navigation en bas.



Figure 9 : Barre d'action

Le menu de navigation permet à l'utilisateur de naviguer facilement entre les différentes fonctionnalités de l'application.

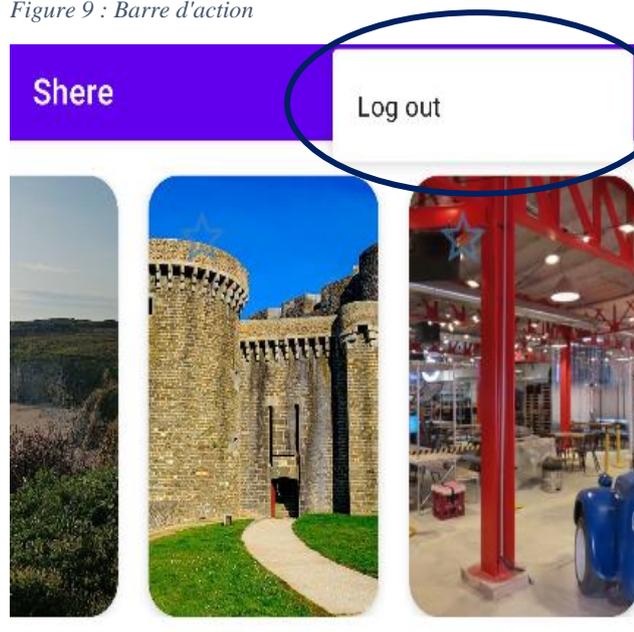


Figure 10: Page d'accueil avec le bouton de déconnexion

Cette activité est équipée d'un menu dans la barre d'action qui permet aux utilisateurs de se déconnecter de l'application et de se rediriger facilement vers l'activité Login.

Cette partie de l'application, malgré sa place graphique minime, est l'une des plus importantes. En effet, le menu de navigation est le cœur de toutes les fonctionnalités et constitue un avantage non négligeable d'un point de vue ergonomique.

Cette activité fût particulièrement complexe à développer pour l'équipe car elle doit en même temps gérer l'authentification à la base de données, charger les fragments, définir une nouvelle vue pour la page et administrer le « Back Stack » c'est-à-dire la pile de fragment déjà chargée.

Une autre grande source de bugs et de problèmes fut cette succession de récupération des données entre les différents fragments et les différentes pages. Plus le projet prenait de l'ampleur et le code se densifiait, plus il y avait de complications liées à la gestion des données pendant l'utilisation de l'app. En effet, il faut pouvoir récupérer les données de l'utilisateur, ou des souvenirs depuis la base de données, puis l'afficher dans le fragment en prenant en compte la propriété de « durée de vie » de chaque fragment. Il a fallu implémenter la capacité du menu à sauvegarder les données de chaque fragment lors de sa fermeture et recharger les données si on l'ouvre à nouveau, sinon un des problèmes était la réinitialisation des fragments et surtout la carte lors de la création d'un nouveau souvenir et lorsque la personne quittait la page, ce qui empêchait tout affichage.

Initialement, ce comportement de sauvegarde n'était pas géré par la Main Activity mais, selon les données à administrer dans les fragments, ce qui fut une erreur fatale pour l'équipe et à l'origine d'une dizaine d'heures de debug.

### ***La fonctionnalité de connexion :***

Le développement de l'activité Kotlin pour la page de login sur Android Studio implique plusieurs étapes clés. Tout d'abord, il est important de créer une interface utilisateur attrayante pour la page de login. Ensuite, il faut créer et configurer la base de données Firebase pour permettre l'authentification avec email et mot de passe. Il faut également configurer les règles de sécurité appropriées et de mettre en place les dépendances nécessaires dans le fichier build.gradle. Cette étape est essentielle pour lier la Firebase avec le projet Android Studio.

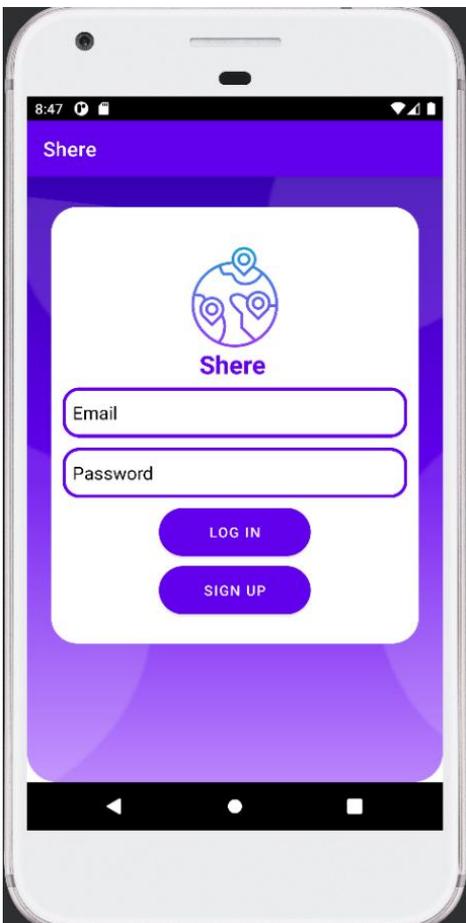


Figure 11: Page de connexion

Une fois que la configuration Firebase est en place, le codage de l'authentification peut commencer. Dans Kotlin, cela peut être réalisé en créant une classe d'authentification qui gère les interactions avec la base. Cette classe peut avoir des fonctions pour l'inscription d'un nouvel utilisateur, la connexion avec un utilisateur existant, la récupération du mot de passe et la déconnexion. Ces différentes fonctionnalités sont incontournables dans le développement d'une application et ont permis d'apprendre à l'équipe la méthode pour enregistrer et gérer les données de la firebase depuis Kotlin, ce qui sera particulièrement utile par la suite.

Les informations d'identification de l'utilisateur (email et mot de passe) doivent être stockées en toute sécurité dans Firebase, ce qui peut être réalisé en utilisant la fonctionnalité de stockage sécurisé de Firebase.

Une fois que l'authentification est en place, il est important de masquer le mot de passe lors de la saisie pour des raisons de sécurité. Cette sécurité peut être réalisée en utilisant des composants d'interface utilisateur avec le paramètre "inputType". Cela garantit que les caractères saisis sont masqués et ne sont pas affichés à l'utilisateur.

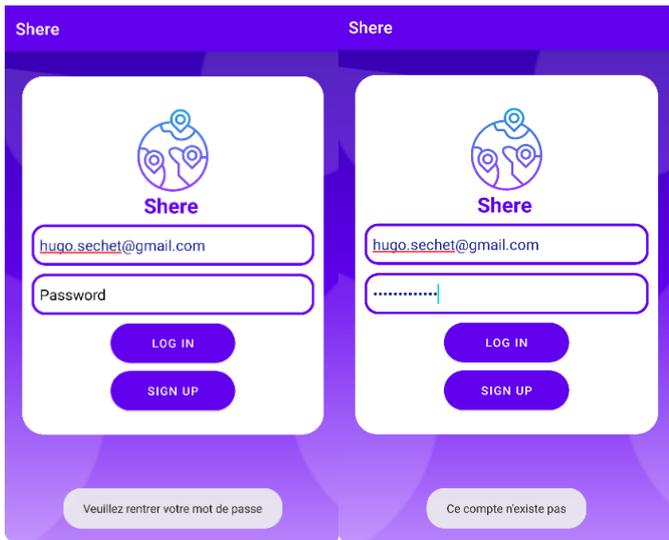


Figure 12: Comparatif mot de passe caché/visible

Enfin, dans le but d'améliorer l'expérience utilisateur, des messages d'erreur appropriés sont affichés en cas d'erreur d'authentification.

Exemple :

-Vérifier que l'utilisateur a bien renseigné son adresse email et son mot de passe.

-Vérifier que le mot de passe et l'email est le même que celui présent dans la firebase pour ce même utilisateur.

Pour la création de notre page d'inscription fonctionnelle et sécurisée de notre application nous avons utilisé Firebase pour stocker les informations des utilisateurs. L'inclusion des champs nom, prénom, email et mot de passe permettent aux utilisateurs de créer un compte. Une vérification dans le but de s'assurer que les utilisateurs saisissent des valeurs valides pour chaque champ est implémentée.

### ***La fonctionnalité d'inscription :***

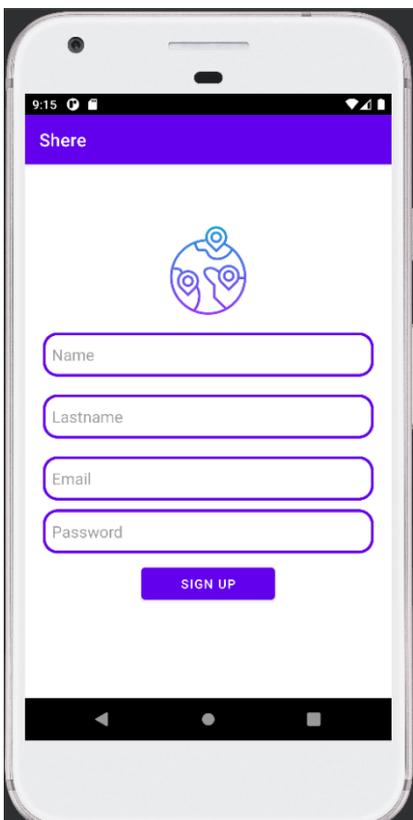


Figure 13: Page d'inscription

Le codage de l'activité Kotlin pour la page de Sign Up dans Android Studio peut être effectué en utilisant Firebase Authentication pour l'enregistrement de l'utilisateur. Pour cela, il faut tout d'abord créer les champs pour le nom, prénom, email et mot de passe dans l'interface utilisateur de l'activité. Ensuite, il est possible de créer une instance de Firebase Authentication et d'appeler la méthode `createUserWithEmailAndPassword` pour enregistrer l'utilisateur avec son email et son mot de passe. Les informations de nom et prénom peuvent être stockées dans la base de données Firebase Realtime Database en utilisant l'identifiant unique de l'utilisateur généré par Firebase Authentication.

Enfin, de la même façon que pour la page de login, il est important de gérer les erreurs lors de l'enregistrement pour éviter les exceptions non traitées. Pour cela, une vérification de la bonne entrée des champs nom, prénom, adresse email et son mot de passe est effectuée. Une vérification s'effectue également sur la longueur du mot de passe afin qu'il possède un minimum de 6 caractères mais aussi sur la validité de l'adresse mail renseignée.

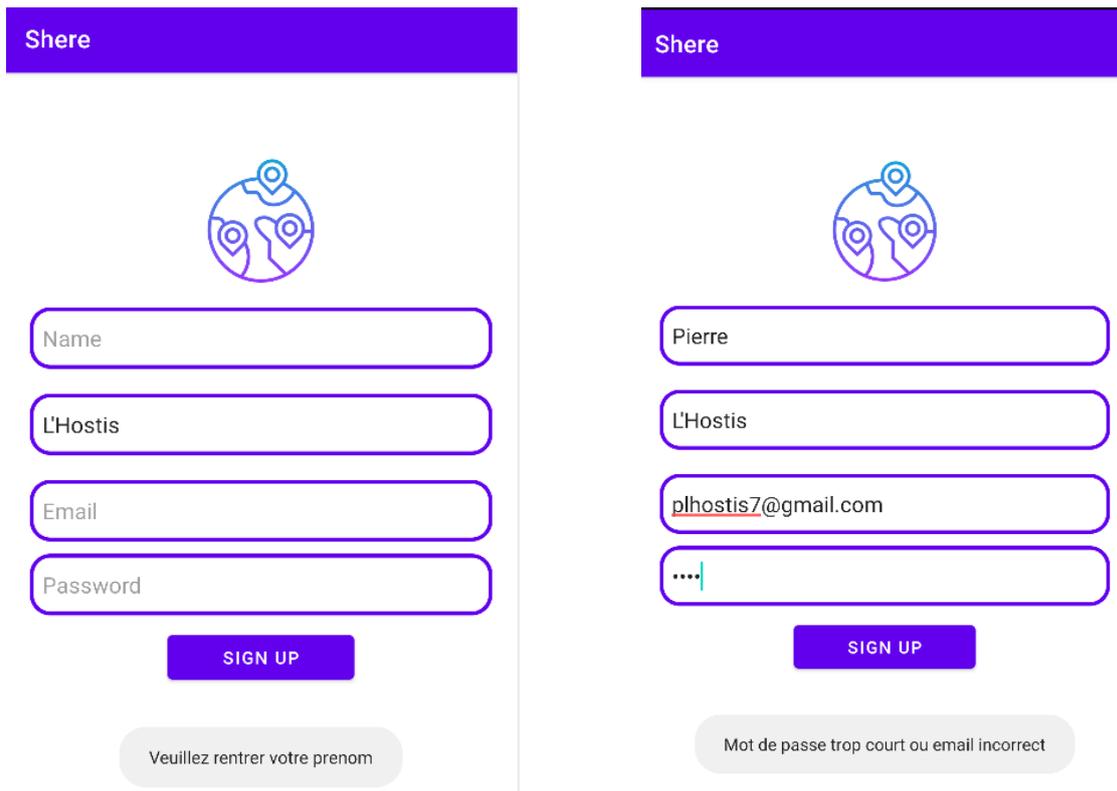


Figure 14: Démonstration des messages d'erreur

## La page d'accueil : HomeFragment

Voici le premier fragment de l'application Shere qui fait office de page d'accueil. Une page d'accueil est souvent la première chose que les utilisateurs voient lorsqu'ils accèdent à une application. Il est donc important de créer une page d'accueil qui soit conviviale, facile à utiliser et qui présente des informations pertinentes. Une des meilleures façons de créer une page d'accueil efficace est d'utiliser des RecyclerViews. Les RecyclerViews permettent de créer des listes de manière dynamique, en utilisant des éléments de disposition et de contenu réutilisables.



Première « RecyclerView » :  
Menu déroulant horizontal ayant pour objectif la visualisation des souvenirs enregistrés après contact avec ces derniers. Ce menu affiche les souvenirs n'étant pas classés dans les favoris.

Seconde « RecyclerView » :  
Menu déroulant vertical ayant pour objectif la visualisation des souvenirs enregistrés après contact avec ces derniers. Ce menu affiche les favoris et les classes à la fin de la liste.

Pour établir la page d'accueil, il est nécessaire d'utiliser la méthode `setContentview` afin de lier le layout xml associé à l'activité. Une fois le layout associé il faut initialiser chaque `RecyclerView` dans la méthode `onCreate` de l'activité en utilisant `findViewById` et liez les adaptateurs aux `recyclerViews` en utilisant `setAdapter`.

Chaque souvenir est associé à différentes fonctionnalités :

La première d'entre elles consiste en l'ouverture d'une page de description spécifique à chaque souvenir lorsque l'on en sélectionne un. Cette page est composée du nom du souvenir, l'image du souvenir, la latitude, longitude et rayon du souvenir. On peut retrouver également un bouton découvrir qui affiche directement le souvenir sur la carte présente dans le deuxième fragment, ainsi qu'un bouton qui permet de supprimer le souvenir de la Firebase.

Il y a également la possibilité d'ajouter aux favoris les souvenirs en cliquant sur l'étoile visible à trois endroits différents : en haut à gauche de l'image dans le menu horizontal, à droite de l'image dans le menu vertical ainsi que sur la page de description.

Les souvenirs en favoris peuvent être retrouver dans la page collection dans le fragment Profil.

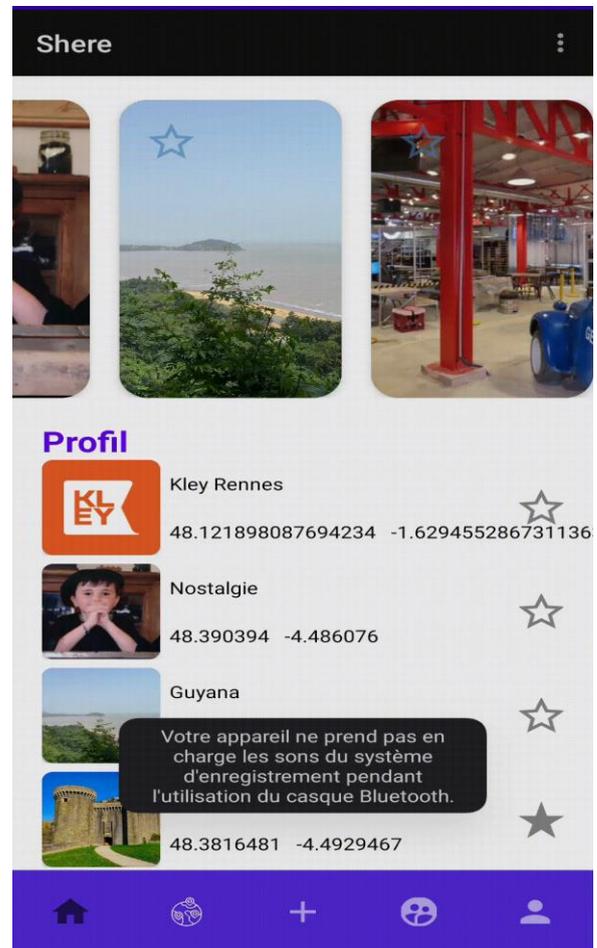


Figure 15: Page de description d'un souvenir

Le `RecyclerView` a pour but de recycler les vues pour améliorer les performances et éviter de charger trop de données dans la mémoire de l'application.

Les `RecyclerViews` nécessitent l'utilisation de ce qu'on en appelle en Kotlin un « adapter », c'est une classe qui permet la gestion des données du recycler à partir de la Firebase, cet adapter récupère les données de la base pour les implémenter dans chaque objet de la liste.

Les objets et notamment leurs propriétés sont gérées dans une autre classe pour avoir un affichage personnalisé selon nos besoins. Ce mode de fonctionnement propre aux recyclers est technique et a posé problème lors du lien entre l'affichage de la page d'accueil au sein du fragment et les données récupérées dans une autre partie de l'application et stocker dans un tiers lieu indépendant au fragment.

## La page de profil :



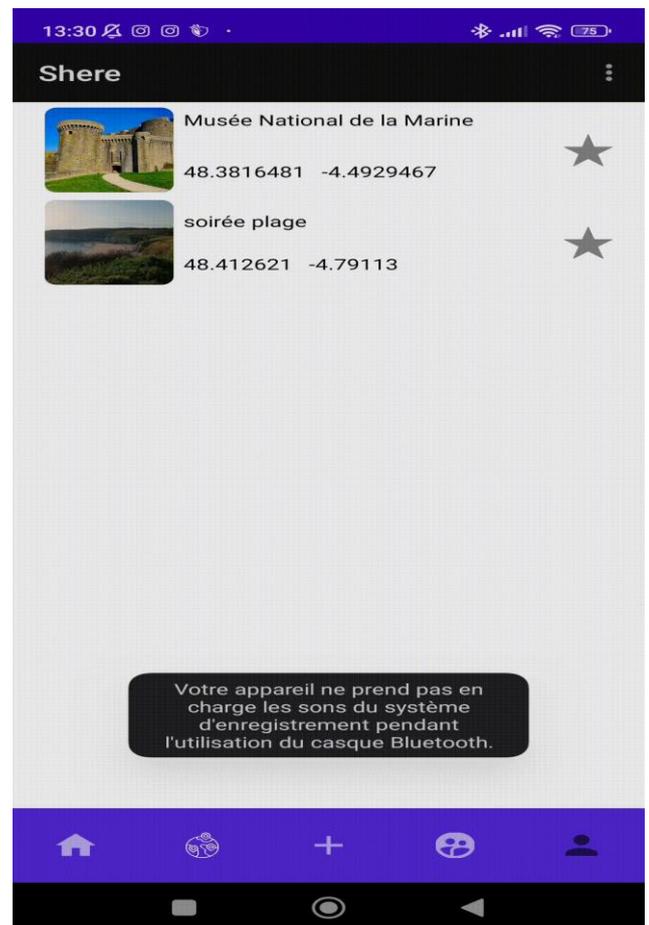
La page de profil permet d'afficher le profil associé à chaque session utilisateur. Une attention particulière a été portée à l'esthétique de la page pour correspondre au code couleur de l'application et pour participer à l'ergonomie de Shere.

Il affiche notamment le nom et prénom et l'adresse mail de l'utilisateur à partir de la Firebase.

Quand on clique sur l'image de profil en haut du fragment on a la possibilité de choisir une nouvelle photo de profil, qui s'actualise dans la base de données sous le nom de « imageUrl » et pour tous les utilisateurs de l'application notamment dans la page de messagerie.

```
r5htKldKY3RaZY1rxusToGw9vC23
├── email: "hugo.sechet1@gmail.com"
├── imageUrl: "r5htKldKY3RaZY1rxusToGw9vC23.jpg"
├── lastname: "Sechet"
├── name: "Hugo"
└── uid: "r5htKldKY3RaZY1rxusToGw9vC23"
```

Figure 16 : Page de profil utilisateur



On peut également voir les souvenirs enregistrés dans la page de collection.

Les souvenirs sont triés dans l'ordre d'ajout en favoris, les plus récents se situent en bas de la page.

Depuis cette page on peut cliquer sur le bouton associé comme dans la page d'accueil et ouvrir la page de détails associée.

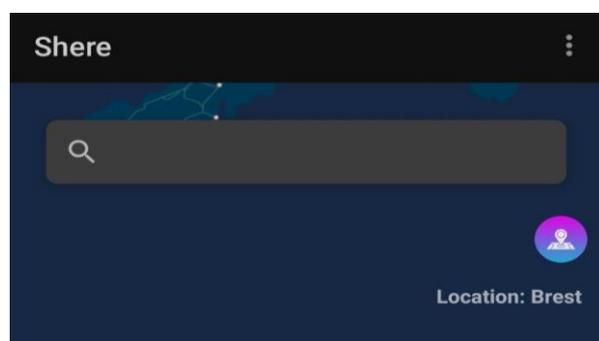
Les autres éléments visibles sur cette page, hormis la collection ont justes un but décoratif, le bouton settings et notifications ne sont pas fonctionnels tout comme le bouton « Edit profile » qui ne permet pas d'éditer les données ci-dessus.

Pour rendre utilisable ce bouton et autoriser la modification, il faudrait modifier la base de données et la classe qui prend en paramètres les données pour ajouter le numéro de téléphone, l'adresse et le code postal, cependant l'équipe a estimé que cette modification était trop mineure pour consacrer du temps dessus.

La plus grande difficulté rencontrée lors du développement fut l'imbrication des layouts les uns sur les autres afin d'obtenir cet affichage en relief, et l'implémentation complexe de cette page de collection.

### ***La carte interactive :***

Le cœur d'une application de géolocalisation est sans nul doute la carte interactive. Dans cette partie est donc introduit le fragment la carte qui affiche les souvenirs. Cette partie de l'application offre une expérience de navigation intuitive et complète pour l'utilisateur au sein de notre carte personnalisée en JSON au couleur de l'application.



*Figure 17: Barre de recherche sur la carte*

Ce fragment représente une carte GoogleMaps est très riche en fonctionnalités pour l'utilisateur. Tout d'abord, il dispose d'une barre de recherche pour faciliter la recherche d'adresses et de lieux spécifiques sur la carte. La ville dans laquelle on se trouve est également affichée sur la carte.

De plus, un bouton d'ajout de souvenirs  est présent pour permettre à l'utilisateur d'ajouter des souvenirs à des endroits particuliers sur la carte.

Ce bouton permet l'ouverture d'une page de dialogue pour ajouter un nouveau souvenir à la base de données et donc sur la carte. La particularité de cette page d'ajout réside dans le fait que l'utilisateur ne peut utiliser que sa position actuelle pour sauvegarder le souvenir. Ce mode de fonctionnement permet de simplifier l'ajout d'un souvenir et d'apporter une seconde option d'ajout à l'utilisateur.

### [Cf Annexes : Figure 27](#)

Une étape importante est de préciser le rayon de visibilité du souvenir pour définir si le souvenir est plutôt accessible avec un rayon assez grand, le rayon pouvant aller jusqu'à 100 000 mètres, ou opter pour un rayon plus petit et donc un souvenir plus difficile d'accès. Une fonction permet de récupérer la position actuelle pour la transformer, lors de l'envoi du formulaire, en une latitude, et longitude.

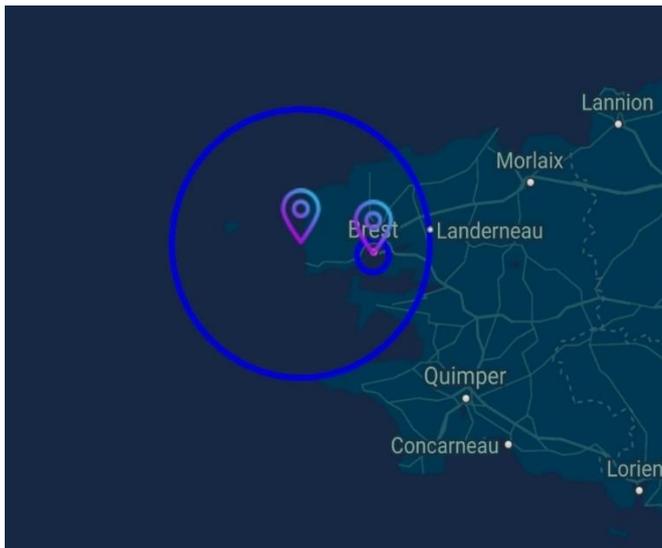


Figure 18: Souvenir sur la carte dans le rayon défini

Cette étape est cruciale pour le bon fonctionnement de l'application car il fait le lien entre la base de données et l'ajout d'un marker qui représente le souvenir sur la carte. Cependant elle fut la source de beaucoup de complications et de bugs avec d'une part la transformation de la position en une donnée qui peut être mise sur une carte, c'est-à-dire la latitude et longitude. D'une autre part la récupération de l'image dans l'explorateur de fichier de l'utilisateur et la mise à jour de l'aperçu dans la page de dialogue. Faire le lien entre la page de dialogue et la carte, donc de transmettre les données entre ces deux pages a été une étape compliquée lors de la confection de cette fonctionnalité.

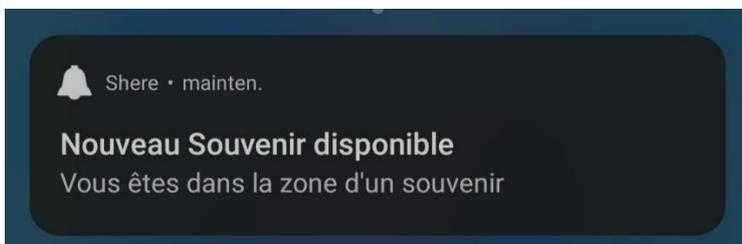


Figure 19 : Notification reçue lorsqu'un souvenir est à portée

Ce ne sont pas les seuls problèmes que l'on a eu lors de la confection de cette carte. La simple implémentation d'une carte GoogleMaps, qui est normalement une activité entière, dans un fragment a été une étape complexe au début de la réalisation du projet. En effet, lors de la création d'une activité basée sur la géolocalisation, on peut choisir un code pré-généré qui facilite son implémentation. Mais le code entre une activité et un fragment change drastiquement, que ce soit dans les imports, les initialisations etc.

Une fois le souvenir créé et ajouté à la Firebase, si la personne connectée se trouve dans la zone du rayon, on peut le retrouver sur la carte car tous les markers présents dans la base de données sont affichés sur la carte si l'utilisateur se trouve dans leur rayon de visibilité.

La carte est donc composée de multiples markers ajoutés par tous les utilisateurs de l'application, mais également par notre avatar qui se trouve à la position exacte de l'utilisateur.

Pour ajouter plus de dynamique et d'interactions au sein de la carte, chaque marker est doté d'une page qui s'ouvre si l'utilisateur clique dessus avec tous les détails du souvenir en question, nom, photo, latitude, longitude, rayon du souvenir et un bouton pour ajouter, comme dans la page d'accueil, le souvenir aux favoris.

L'objectif de l'équipe pendant la réalisation de cette carte était de mettre l'accent sur l'aspect actif du fragment, c'est pour cela que à l'ouverture de la carte un zoom est effectué sur la position de la

personne pour directement pouvoir observer les souvenirs à proximité, une notification est également envoyée quand l'utilisateur se trouve dans la zone d'un souvenir.

### ***L'ajout d'un souvenir :***

Ajouter est un fragment central pour le bon fonctionnement de Shere, il permet l'ajout d'un souvenir à partir d'une adresse et non plus de la position actuelle de l'usager.



Figure 20 : Page d'ajout d'un souvenir

On peut voir la même architecture que le bouton ajouter présent dans le fragment contenant la carte :

Un bouton Charger Image qui permet de parcourir l'explorateur de fichier de la personne pour charger l'image du souvenir.

L'utilisateur peut renseigner le nom du souvenir dans le premier champ « EditText », le lieu du souvenir sous forme d'adresse comme par exemple « ISEN, Brest » à noter que s'il y a plusieurs adresses qui porte le même nom il faut préciser la ville ou le Pays.

Ce champ permet de rentrer également des noms de lieu comme un nom de bar, lieu historique...

Le rayon de visibilité est toujours un champ obligatoire pour quantifier la portée du souvenir.

Le bouton confirmer récupère le lieu du souvenir pour récupérer sa latitude et longitude, puis envoie le formulaire à la base de données dans le champ « Emotions »

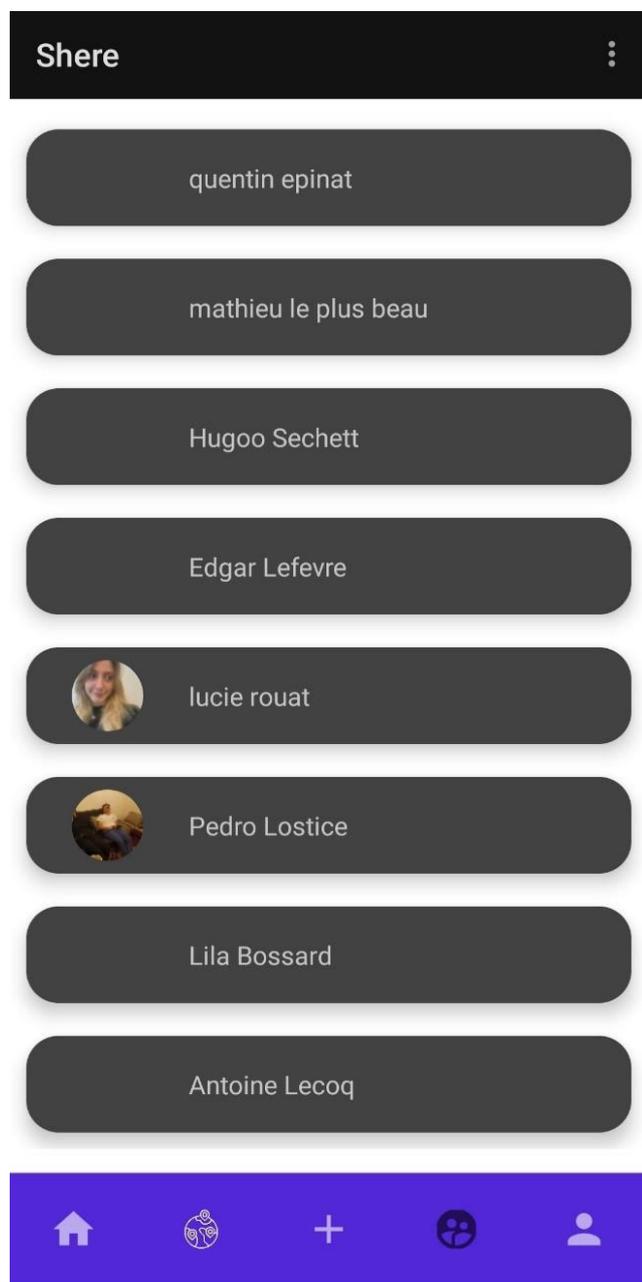
La fonctionnalité qui a causé le plus d'erreurs et de difficultés est bien sûr le lieu du souvenir, en langage Kotlin la gestion des adresses se fait grâce à une fonction GeoCoder, mais pour permettre de récupérer une adresse valide, il a fallu prendre en compte les virgules pour les villes ou les pays, prendre en compte le code postal...

Une multitude de test pour être sûr que l'utilisateur ne rentre pas n'importe quelle valeur pour notamment éviter les injections SQL donc protéger l'application.

De la même manière, des tests sont effectués dans chaque formulaire pour vérifier la véracité des informations.

## La messagerie :

Enfin, la messagerie est le dernier point à aborder au sein de cette partie accordée au développement.



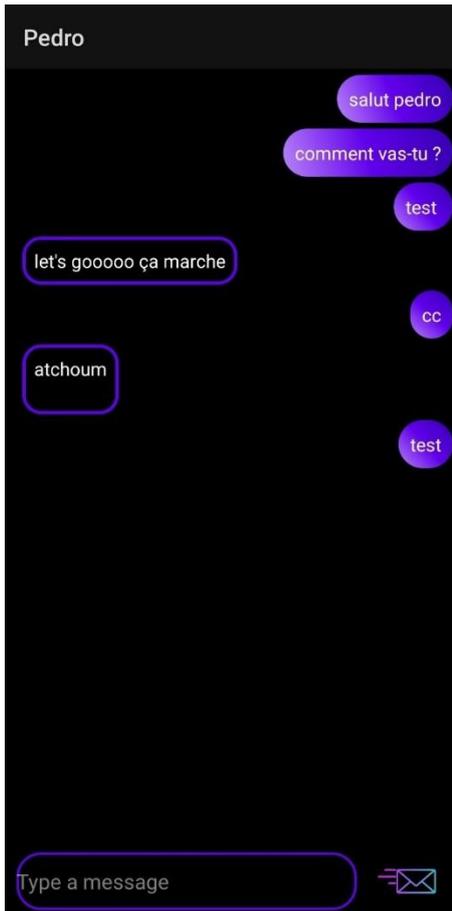
Un fragment qui n'était pas aux centres des discussions lors des premières réunions avec les chargés de projet mais qui semblait pour l'équipe un indispensable dans une optique d'ergonomie et d'interactions entre les usagers est celui de la messagerie. En comparaison, ce qui fait le succès d'Instagram est sa capacité à s'échanger du contenu sans limite avec n'importe qui n'importe où. C'est un système exponentiel où plus il y a de contenu dans l'application plus il y a d'interactions et donc plus l'application suscite de l'engouement et incite les utilisateurs à rester et utiliser notre outil. C'est dans cette optique que ce fragment est né.

Au sein de cette page il est possible de retrouver tous les utilisateurs de l'application avec leur photo de profil personnalisée dans la page de profil, les personnes sans photos ont tout simplement gardés dans notre exemple la photo par défaut.

Cette page a posé beaucoup de problèmes de confections, notamment pour actualiser dans les items du recycler les images associées dans la Firebase. Il faut récupérer ce qu'on appelle le « `currentPath` » pour l'image pour chaque utilisateur ce qui est techniquement pénible car il est nécessaire de d'abord récupérer la personne actuellement connecté à la session puis créer un tableau qui parcourt toute la Firebase et se remplit avec les informations présentent. Puis on retourne ce tableau de classe `User` et on remplit le `RecyclerView` avec ce même tableau.

Les informations sont stockées dans un `RecyclerView` qui administre les utilisateurs et les messages à l'aide d'un adapter, il a le même fonctionnement que les deux adapter de la page d'accueil.

Une fois dans cette page, on a la possibilité de cliquer sur chaque item du recycler pour avoir accès à la conversation avec la personne concernée, on a donc à ce moment accès à une nouvelle page qui est une nouvelle activité de l'application.



Les messages sont envoyés à l'autre utilisateur quasiment instantanément et l'affichage du message change selon s'il est reçu ou envoyé.

On peut également retrouver le nom du contact en haut de la page.

Cette application est plutôt basique, l'équipe n'a pas voulu ajouter d'autres fonctionnalités car jugées inutiles pour la démonstration du projet.

Les fonctionnalités liées à la base de données permettent de sauvegarder et de charger les messages à partir de la base de données, ainsi que de les trier et de les filtrer en fonction des différents critères tels que l'expéditeur, le destinataire, la date et le contenu du message. La base de données permet également de garantir la cohérence des données en évitant les doublons ou les conflits.

Les messages, ainsi que le destinataire, sont chiffrés pour garantir l'intimité et l'anonymat des personnes.

L'ajout d'image depuis le tchat, la possibilité de partager des souvenirs directement depuis cette page ou l'ajout de gifs, la capacité d'appeler la personne etc. seraient des fonctionnalités à ajouter dans le futur.

Android Studio est un excellent IDE de développement, il effectue de nombreuses tâches en arrière-plan qui permet de faciliter la vie du développeur, il détecte les erreurs, leurs sources et des moyens de les résoudre. Il offre un émulateur intégré pour directement compiler le code sur un appareil virtuel. Mais les tâches qu'il exécute en arrière-plan peuvent parfois faire des mises à jour automatique ou apporter des modifications inopinées qui provoquent de nombreuses erreurs très difficiles à détecter et résoudre. Ce fut notamment le cas lors d'une mise à jour vers la dernière version en date d'Android Studio, où mon code ne marchait plus sous la nouvelle version de l'IDE. Les différentes tentatives de debug pour régler des problèmes indépendants du code peuvent entraîner d'autres problèmes étranges. L'équipe s'est notamment retrouvé bloquer pendant deux jours et a dû désinstaller et réinstaller

L'application, en supprimant tous les fichiers de caches de l'appareil notamment dans des fichiers de l'explorateur comme le %App Data% ou des fichiers comme JetBrains ou Android Studio.

L'équipe de développement à rencontré d'autres erreurs et problèmes liés au code cette fois. Comme dans un fichier .xml pour établir un espacement sur différents boutons, un simple décalage angulaire de 5° provoquait lors du démarrage sur l'émulateur d'Android Studio une erreur.

# Fonctionnalités et améliorations futures

## *Implémentations de nouvelles fonctionnalités :*

L'application pourrait contenir un système d'amis/abonnées pour trier le contenu que l'on souhaite voir ou non, avec la possibilité de partager aux amis d'autres réseaux de nous rejoindre et partager du contenu avec nous sur Shere. Former des groupes privés etc. Comme il peut déjà exister sur d'autres réseaux, mais ici on pourrait voir les regroupements de nos amis sur la carte et voir des souvenirs en commun être ajoutés.

Une fonctionnalité bonus qui paraît pour l'équipe centrale, serait le fait d'avoir une page découverte où l'on peut voir les souvenirs basés sur les souvenirs que nous avons aimés, souvenirs qui n'appartiennent pas à nos amis. Les souvenirs prendraient la forme de gifs, vidéos courtes ou moyennes dans la limite de 1 minute pour cette partie découverte.

L'application pourrait contenir une partie vlog avec des vidéos plus longues cette fois qui peut dépasser les dix minutes. Certains endroits sur la carte pourraient devenir une sorte de repère, qui regrouperaient les souvenirs d'une même personne ou plusieurs. On aurait la possibilité de cliquer dessus et avoir une sorte de storytelling des histoires que la personne a vécu dans ce même endroit, comme la vie étudiante dans un appartement, ou des soirées endiablées sur une plage... Et donc voir l'évolution au fil des années de la personne à travers ce lieu.

Une idée de développement de l'application serait la possibilité de vendre des services publics à des villes. Par exemple pour expliquer l'histoire de la ville à travers un parcours, et pouvoir découvrir les moments forts de la ville grâce aux souvenirs et aux fameux « repères », on peut prendre pour exemple l'église Notre-Dame de Paris qui a vécu de nombreux changements, bouleversements depuis sa construction. Proposer des visites guidées interactives et personnalisées aux touristes dans les villes peut être très attractif. Cela permettrait de découvrir l'histoire, la culture, les monuments et les anecdotes des villes d'une manière nouvelle et plus immersive.

Enfin, cette application pourrait également offrir des options pour les entreprises et les services publics de la ville. Par exemple, les autorités locales pourraient utiliser l'application pour informer les touristes des événements locaux, des transports en commun et des itinéraires. Les entreprises locales pourraient également utiliser l'application pour promouvoir leurs produits et services auprès des touristes.

On peut imaginer de créer un statut « créateur » ou « entreprise » pour inciter les influenceurs et entreprises à poster des souvenirs et augmenter la visibilité et l'utilisation de l'application. Ces statuts auraient des avantages comme un marker spéciale sur la carte, l'envoi d'une notification spéciale à ses abonnés, avoir accès justement à ces « vlogs ».

Le design de la carte pourrait être revu pour être en 3D ou encore plus dynamique, elle pourrait permettre de voir ses amis bouger sur la carte, voir leurs statuts comme « au boulot », « en cours » « au restaurant » etc. On pourrait voir les regroupements de souvenirs très importants sur la carte lors de concert, événements etc. pour suivre certains événements en différé.

Le design des markers pourraient être revu pour avoir un affichage plus moderne, dynamique avec par exemple l'image de profil du détenteur du souvenir, pouvoir avoir accès à son profil, les souvenirs qu'il a postés...

L'application pourrait contenir un système de commentaire sous les souvenirs, vlogs. Les créateurs pourraient créer des « emotes » spéciales pour leurs abonnés en lien avec un souvenir en particulier.

L'application pourrait contenir un système de « tag » où les utilisateurs pourraient identifier leurs amis sur des souvenirs, ces souvenirs seraient visibles sur les deux comptes, le compte de la personne qui poste le souvenir et de la personne qui est identifiée.

En plus de cela, l'application pourrait offrir la possibilité aux utilisateurs de trier leurs souvenirs en fonction de différents critères tels que la date, le lieu, le type de souvenirs, etc. Ils pourraient également ajouter des notes ou des commentaires pour se rappeler les détails importants.

L'application pourrait contenir un système d'envoi de souvenir à une personne spécifique depuis les messages directement, ou un système de partage de souvenirs sur d'autres plateformes de médias sociaux.

Pour rendre l'expérience encore plus personnalisée, l'application pourrait également proposer des filtres et des effets pour modifier l'apparence des souvenirs. Les utilisateurs pourraient également ajouter de la musique ou des clips audio pour accompagner leurs souvenirs.

En outre, pour encourager la participation des utilisateurs, l'application pourrait organiser des concours de souvenirs, où les utilisateurs pourraient partager leurs souvenirs les plus mémorables et remporter des récompenses intéressantes.

Il faut donc implémenter un système de paiement intégré, pour cela il faut choisir un ou plusieurs fournisseurs de services de paiement tels que Stripe, PayPal, Braintree, etc. qui offrent des bibliothèques SDK. Intégrez la bibliothèque SDK du fournisseur de services de paiement : Après avoir choisi le fournisseur de services de paiement, vous devez intégrer sa bibliothèque SDK dans votre application Android. La bibliothèque SDK vous permet de créer des transactions de paiement et de recevoir des réponses de paiement.

Créer une page de paiement dans l'application Android pour permettre aux utilisateurs de saisir les informations de paiement telles que le nom, l'adresse de facturation, le numéro de carte de crédit, etc. Vous pouvez utiliser une disposition personnalisée pour concevoir votre page de paiement.

Il faut implémenter une certaine logique de paiement, lorsque l'utilisateur soumet les informations de paiement, il faut utiliser la bibliothèque SDK du fournisseur de services de paiement pour créer une transaction de paiement. Vous devez également gérer les erreurs de paiement et afficher des messages appropriés à l'utilisateur.

Il est également important de prendre en compte les exigences de conformité en matière de sécurité des données de paiement, telles que PCI DSS, et de suivre les meilleures pratiques de sécurité pour garantir que les données de paiement des utilisateurs sont traitées en toute sécurité.

## *Améliorations de fonctionnalités existantes :*

Notre application actuellement ne répond pas exactement aux attentes du cahier des charges sur le WEB 3.0.

AndroidStudio est un outil développé par Google, il est normal que cet IDE facilite l'utilisation d'outils Google pour son bon fonctionnement, Firebase, GoogleMaps sont des outils directement implémentés.

De la même façon, pour utiliser une carte GoogleMaps il est nécessaire de posséder une API key et donc d'utiliser GoogleMapsPlatform pour stocker la carte.

Il est cependant possible de passer outre et utiliser d'autres outils.

OpenStreetMap entre autres est un exemple de carte Open Source qui pourrait être utilisé afin d'héberger et modifier la carte.

OpenStreetMap à l'avantage d'avoir une carte très détaillée et à jour (zones de travaux, passages piétons, sens de circulation, feux tricolores, etc.). Elle donne la possibilité de choisir le calque de son choix parmi 5 disponibles.

Une nouvelle mesure prise par Google a été de limiter le nombre de requêtes gratuites par jour. Si cela n'a pas vraiment eu d'impact sur les petits utilisateurs qui n'atteignent pas la limite d'utilisation, certains gros utilisateurs ont alors préféré d'autres solutions comme OpenStreetMap ((par exemple, Foursquare et Pinterest).

Les derniers changements (de juin 2018) ont beaucoup plus d'impact : Google a non seulement décidé d'encre réduire la limite du nombre de requêtes gratuites (de 25000 cartes par jour à 28000 cartes par mois, soit plus ou moins 1000 cartes par jour, donc 25 fois moins) mais aussi d'obliger les utilisateurs à fournir leur numéro de carte de crédit même si la limite gratuite n'est pas dépassée !

Google peut changer les conditions d'utilisation de leurs services quand ils veulent et ce ne sera plus que probablement pas à l'avantage de l'application, L'équipe n'a aucun contrôle sur la partie carte. Google Maps ne donne pas accès aux données utilisées pour créer la carte (et c'est le cas de la plupart des solutions propriétaires) mais un accès complet à la base de données OpenStreetMap est possible. Cependant dans le cadre de notre projet et une contrainte de temps limité, pour atteindre les autres objectifs du cahier des charges dans le temps imparti, l'équipe à décider d'utiliser les outils proposés par l'entreprise américaine, ce qui implique de sauvegarder les données de l'application dans un serveur aux Etats-Unis et accessible pour la multinationale Google

La typographie de l'application pourrait être améliorée, elle est très commune et simple. On pourrait la personnaliser afin de développer l'image de marque de l'application. Par exemple la typographie d'Instagram ou encore Star Wars est typiquement reconnaissable au premier regard.

Cette page à poser beaucoup de problèmes de confections, notamment pour actualiser dans les items du recycler les images associées dans la firebase. Il faut récupérer ce qu'on appelle le « `currentPath` » pour l'image pour chaque utilisateur ce qui est techniquement pénible car il est nécessaire de d'abord récupérer la personne actuellement connecté à la session puis créer un tableau qui parcourt toute la firebase et se remplit avec les informations présentent. Puis on retourne ce tableau de classe User et on remplit le recyclerView avec ce même tableau.

# Le Web 3.0

## Les principes du Web 3.0

La contrainte principale que ce projet implique à terme est le respect des règles établies par le Web 3.0. Le Web 3.0 est depuis quelques années au cœur des discussions du monde de la tech. Il serait le successeur des Web 1.0 et 2.0, le premier étant celui des débuts d'internet, simple portail d'information plutôt désorganisé permettant la navigation entre des pages statiques et n'impliquant donc aucune dynamique. Depuis les années 2000, c'est le Web 2.0 que nous connaissons aujourd'hui. Un web dit « participatif » où la dynamique sociale est mise en avant notamment via l'avènement des géants des réseaux sociaux : Facebook, Instagram, Snapchat, ... mais aussi des principaux sites web que nous utilisons au quotidien tels que Amazon ou Uber. L'idée d'un Web 3.0 émergea au sein du monde de la cryptomonnaie dans l'esprit de Gavin Wood, créateur d'Ethereum une blockchain ayant vue sa propre cryptomonnaie, l'ether, s'imposer comme la deuxième monnaie du marché après le bitcoin. Les motivations de Wood sont issues du constat que les données fournies sur la majorité des plateformes en ligne se retrouvent beaucoup trop centralisées, simplifiant leur contrôle par les plus grandes entreprises de la tech (Amazon, Alphabet, Apple, Meta, ...) qui ont ainsi récoltées d'immenses quantités de données personnelles sans aucun contrôle de la part des utilisateurs concernés. L'intention serait donc de créer une sorte de Web « décentralisé » où chaque utilisateur serait entier propriétaire de ses données personnelles et les feraient circuler eux-mêmes de plateforme en plateforme en supprimant purement et simplement les intermédiaires que sont les grandes entreprises citées plus tôt. On pourrait être amené à se demander quel est le rapport entre Web 3.0 et cryptomonnaie. Celui-ci est simplement dû au fait que dans le monde de la blockchain chaque échange, financier ou non, est consigné dans un registre décentralisé et stocké sur les machines des utilisateurs. Si l'on rapporte ce principe au Web 3.0, cela permettrait de créer un tout nouveau web basé sur des plateformes gérées, détenues et améliorées par des communautés d'individus plutôt que par des grandes entreprises de la tech.

## Les enjeux du Web 3.0

A présent que le principe initial du Web 3.0 a été introduit, quels sont ses enjeux concrets ? Tout d'abord, rappelons que le premier d'entre eux est la décentralisation, de faire migrer un Web 2.0 où les données sont stockées sur les serveurs de grandes entreprises vers un Web 3.0 où chacun serait propriétaire et unique propriétaire de ses données personnelles. Ensuite, la sécurité des données sera au cœur des problématiques du Web 3.0 et a comme ambition d'utiliser un système identique à la blockchain afin de garantir l'intégrité des données via une association directe des données à une machine.

L'interopérabilité entre les plateformes fait également parti des principes fondamentaux du Web 3.0. Cela signifie que l'utilisateur pourra faire circuler librement ses données d'une plateforme à l'autre sans limites. De plus, l'arrivée d'applications décentralisées pourra offrir de nouveaux modèles économiques basés sur les mêmes principes que ceux de la blockchain. Un principe de gouvernance décentralisée sera également mis en place. Ainsi le développement et l'évolution du Web seront issus de la communauté et non plus par des entreprises. Les utilisateurs pourront donc participer dans les prises de décision, la transparence et l'équité dans la gouvernance des plateformes du Web 3.0.

Le dernier enjeu est l'inclusion numérique visant à fournir un accès ouvert et équitable peu importe les origines géographiques, sociales ou le niveau d'éducation.

## La sécurisation des données

Le Web 3.0 dépend de par son principe d'une très forte sécurisation des données qui sera essentielle à mettre en place pour un projet visant à développer une application respectant ses principes. L'application en question devra donc se construire en pensant

- **Contrôle des données** : Le Web 3.0 cherche à donner aux utilisateurs un meilleur contrôle sur leurs données personnelles. Avec la décentralisation, les données sont distribuées sur un réseau de nœuds plutôt que d'être stockées dans des serveurs centralisés. Cela signifie que les utilisateurs peuvent décider comment leurs données sont utilisées, partagées et stockées, et peuvent révoquer l'accès à leurs données à tout moment.
- **Confidentialité** : La confidentialité est un aspect clé de la confiance en ligne. Le Web 3.0 vise à offrir des solutions pour protéger la vie privée des utilisateurs, notamment par le biais de techniques de chiffrement, qui permettent de protéger les données sensibles des utilisateurs tout au long de leur transmission et de leur stockage sur le réseau décentralisé.
- **Sécurité des transactions** : Le Web 3.0 met en avant l'utilisation de la technologie de la blockchain, qui offre des mécanismes de sécurité avancés. Si à l'avenir le business plan associé à l'application implique des transactions, ces dernières devront absolument suivre le schéma de la blockchain afin d'exister dans le Web 3.0. Les transactions sur la blockchain sont vérifiables et immuables, ce qui renforce la confiance dans l'intégrité des données. De plus, les contrats intelligents, qui sont des programmes auto-exécutables sur la blockchain, permettent de créer des relations de confiance sans avoir besoin d'intermédiaires.
- **Authenticité des données** : Le Web 3.0 vise à lutter contre la désinformation et la falsification des données en ligne. Les technologies de vérification d'authenticité, telles que les preuves de provenance basées sur la blockchain, peuvent être utilisées pour garantir l'authenticité des données, des médias et des informations en ligne, ce qui permet aux utilisateurs de mieux discerner les informations fiables des informations trompeuses.
- **Sécurité des applications décentralisées** : Les applications décentralisées sont un élément clé du Web 3.0, mais elles peuvent également présenter des défis en matière de sécurité. Le Web 3.0 nécessite des solutions de sécurité robustes pour protéger les utilisateurs contre les attaques, les fraudes et les vulnérabilités potentielles des applications décentralisées, notamment par le biais d'audits de sécurité, de normes de développement sécurisées et de mécanismes de gouvernance pour résoudre les problèmes de sécurité.
- **Éducation et sensibilisation** : La confiance en ligne dépend également de l'éducation et de la sensibilisation des utilisateurs. Le Web 3.0 nécessite des efforts pour sensibiliser les utilisateurs aux bonnes pratiques en matière de sécurité en ligne, ainsi qu'à leur donner les outils et les compétences nécessaires pour protéger leurs données et leur vie privée dans un environnement décentralisé.

L'essentiel de la mise en place de ces différents points sur la sécurisation des données est de pouvoir instaurer le Web 3.0 comme espace de confiance afin d'inciter les utilisateurs à interagir avec lui et les autres personnes, à partager des données et à utiliser des services en ligne de manière sécurisée tout en préservant la vie privée et en minimisant des risques de fraudes.

### Manières d'implémenter le Web 3.0 au sein de l'application

Utiliser des technologies de blockchain : Les blockchains sont des registres décentralisés et sécurisés qui offrent la possibilité de stocker et de partager des données de manière transparente et immuable. Utiliser une blockchain permettra de stocker les données de géolocalisation de manière décentralisée, ce qui permettra aux utilisateurs de garder le contrôle sur leurs données de localisation.

Mettre en place l'interopérabilité des données : Le Web 3.0 encourage l'interopérabilité des données, ce qui signifie que les données doivent être facilement partageables et utilisables entre différentes applications et plates-formes. Mettre en place des API ou des protocoles standardisés peut permettre aux utilisateurs de partager leurs données de géolocalisation avec d'autres applications ou services conformément aux principes du Web 3.0.

Implémenter des mécanismes de gouvernance décentralisée : Le Web 3.0 prône la gouvernance décentralisée, où les décisions sont prises collectivement par la communauté des utilisateurs plutôt que par une entité centrale. La possibilité de mettre en place des mécanismes de gouvernance décentralisée dans l'application, tels que des systèmes de vote ou de gouvernance basés sur la blockchain, va permettre aux utilisateurs de participer activement à la prise de décision concernant l'utilisation de leurs données de géolocalisation.

Offrir la propriété des données aux utilisateurs : Le Web 3.0 met l'accent sur la propriété des données par les utilisateurs. Vous pouvez permettre aux utilisateurs de posséder et de contrôler leurs propres données de géolocalisation, par exemple en utilisant des clés de chiffrement ou en offrant des options pour la suppression ou l'exportation des données de géolocalisation à tout moment.

Éduquer les utilisateurs sur le Web 3.0 : L'éducation des utilisateurs sur les principes du Web 3.0 et sur la manière dont votre application implémente ces principes est importante. Assurez-vous de fournir des informations claires et compréhensibles sur la façon dont vous protégez la confidentialité, la sécurité et la propriété des données de géolocalisation des utilisateurs, et de promouvoir la compréhension et la sensibilisation au Web 3.0.

En somme, mettre en place les principes du Web 3.0 dans une application Android basée sur la géolocalisation nécessite une approche holistique, incluant la décentralisation, la sécurité, la confidentialité, la gouvernance décentralisée et la propriété.

# L'aspect social de l'application

## La capacité de diffusion

L'outil de géolocalisation des souvenirs, comme tout nouveau concept, doit trouver son public le plus efficacement et rapidement possible afin de pouvoir commencer à réellement exister. Il y aurait ainsi plusieurs moyens de le faire, en commençant ironiquement par passer par le modèle actuel de réseaux sociaux en publiant du contenu mettant en avant les fonctionnalités et le caractère innovant de l'application. Une communication passant par des partenaires stratégiques pouvant être en rapport avec le principe initial de l'application. Cela pourrait donc être des organisations, entreprises liées aux voyages ou à tout ce qui peut s'avérer être en lien avec la production de souvenirs : photographie, tourisme ou autre. Ces organisations ou entreprises pourront ainsi créer du contenu collaboratif et attirer les utilisateurs vers l'application. La communication pourra également passer par des organisations ou entités dont le produit ou service pourrait profiter grandement des services fournis par l'application. Certaines mairies de ville ou de quartier voulant faire la promotion de certains lieux clés ou historiques environnants, des musées voulant créer des chemins précis d'œuvres en œuvres ou encore des créateurs de jeux en plein air comme des Escape Games ou des courses d'orientation trouveront un intérêt bien particulier à un outil comme celui-ci.

Dans un futur sans doute plus lointain, un système parrainage accordant des avantages dans l'application pourra être mis en place afin d'inciter les gens à parler et diffuser l'application et accélérer par cette occasion le phénomène de bouche à oreille qui demeure l'un des médiateurs principaux pour faire connaître ce style de technologie.

## Les différents publics visés :

Une application de souvenirs, de découverte de lieux liées aux émotions de plusieurs personnes, inconnues ou non, peut évidemment intéresser les proches avec qui les souvenirs sont partagés mais aussi différents secteurs qui trouveraient un intérêt particulier à cela :

Premièrement, l'application peut offrir aux touristes et aux voyageurs une expérience enrichissante en leur fournissant des informations sur les lieux d'intérêt à proximité, des recommandations d'activités, des offres spéciales dans les restaurants ou les boutiques locaux, et des fonctionnalités de suivi des itinéraires. Elle peut également faciliter la découverte de nouveaux endroits et permettre aux utilisateurs de créer et de partager leurs souvenirs de voyage.

L'application peut également être destinée aux amateurs d'activités de plein air tels que les randonneurs, les coureurs, les cyclistes, les campeurs, etc. Elle peut fournir des informations sur les sentiers, les parcs, les aires de camping, les pistes cyclables, et permettre aux utilisateurs de planifier et de suivre leurs activités en temps réel.

L'application peut aussi être conçue pour les amateurs de culture et d'histoire, en fournissant des informations sur les monuments, les musées, les sites historiques, et en proposant des itinéraires thématiques pour découvrir ces lieux. Elle peut également permettre aux utilisateurs de partager des informations sur les sites qu'ils visitent, de participer à des jeux de piste ou des quiz, et de contribuer à la préservation du patrimoine culturel.

Ensuite, l'application peut cibler les consommateurs locaux en leur proposant des offres spéciales, des promotions ou des événements dans les commerces, les restaurants, les cinémas, les salles de sport, etc. Elle peut également permettre aux utilisateurs de découvrir de nouveaux endroits dans leur région, de partager leurs recommandations avec d'autres utilisateurs locaux, et de participer à des programmes

de fidélité. L'application peut logiquement aussi être destinée aux entreprises locales, telles que les restaurants, les boutiques, les hôtels, les lieux de divertissement, etc., pour promouvoir leurs offres spéciales, leurs événements ou leurs services. Elle peut également offrir des outils de gestion des offres et des promotions, ainsi que des fonctionnalités de collecte de données pour mieux comprendre leur clientèle et améliorer leur marketing.

## L'utilité de l'application pour des entreprises

Tout au long de ce projet, une dynamique d'entreprise fut installée. Cette dynamique fut intéressante pour plusieurs points mais celui abordé ici sera celui de la création d'une application par une entreprise pour potentiellement d'autres entreprises. Durant l'élaboration de l'application, l'accent fut mis sur le respect du Web 3.0. Or, cette catégorie d'application récoltant les données géographiques des utilisateurs possède un immense intérêt aux yeux des entreprises et ce pour plusieurs raisons :

**Le marketing ciblé :** L'utilisation du Geofencing permet de cibler spécifiquement les utilisateurs dans des zones géographiques précises. Les entreprises privées peuvent ainsi envoyer des notifications push, des offres spéciales ou des promotions personnalisées aux utilisateurs se trouvant à proximité de leurs points de vente ou de leurs emplacements physiques. Cela permet de maximiser l'impact du marketing en atteignant directement les utilisateurs pertinents au bon moment et au bon endroit.

**L'engagement de la clientèle :** L'application basée sur la géolocalisation peut permettre aux entreprises de mieux engager leur clientèle. Par exemple, elles peuvent envoyer des notifications pour informer les utilisateurs sur des événements ou des activités se déroulant à proximité, les inciter à participer à des programmes de fidélité, ou les encourager à partager leurs expériences sur les réseaux sociaux. Cela peut favoriser l'interaction avec les clients et renforcer leur engagement envers la marque ou l'entreprise.

**La collecte de données et analyse :** L'application peut permettre de collecter des données précieuses sur les comportements des utilisateurs, leurs préférences et leurs habitudes de consommation. Ces données peuvent être analysées pour mieux comprendre les besoins et les attentes des clients, identifier les tendances du marché, et prendre des décisions éclairées en matière de marketing, de développement de produits ou d'amélioration des services. Cela peut également être un atout pour attirer des actionnaires potentiels en démontrant la capacité de l'entreprise à collecter et à utiliser efficacement des données pour stimuler la croissance.

**La différenciation concurrentielle :** L'utilisation d'une application basée sur la géolocalisation peut permettre aux entreprises de se différencier de leurs concurrents en offrant une expérience utilisateur unique et innovante. Cela peut attirer l'attention des consommateurs et les inciter à choisir une entreprise plutôt qu'une autre. Pour les actionnaires potentiels, cela peut également représenter un avantage concurrentiel et montrer que l'entreprise est à la pointe de la technologie et des nouvelles tendances du marché.

**La monétisation potentielle :** Selon le modèle d'affaires de l'application, il peut y avoir des opportunités de monétisation, notamment par le biais de la publicité, des partenariats ou des abonnements. Cela peut représenter un intérêt pour les actionnaires potentiels, car cela peut générer des revenus supplémentaires et contribuer à la valorisation de l'entreprise.

# La dimension éthico-légale

Autour de ce projet, certaines questions peuvent se poser concernant l'éthique ou la législation liée à celui-ci. Tout d'abord, la question de la géolocalisation est souvent au cœur des débats liés à la vie privée et internet. Dans l'application développée durant ce projet, le système de géolocalisation est celui du Geofencing, technique de repérage en zone. Ce système est basé sur un mélange entre les géolocalisations actives et passives. La géolocalisation active nécessitant la permission de l'utilisateur pour s'activer n'est pas un problème. C'est donc du côté de la géolocalisation passive que des problèmes liés à la vie privée peuvent intervenir. Contrairement à l'active, la passive implique un enregistrement en continue des déplacements de l'utilisateur, sans pour autant que l'application soit en fonctionnement.

Le Geofencing, dans la manière dont il est utilisé au sein de ce projet, délimite une zone établie sur un périmètre défini par l'utilisateur autour d'un objet : un souvenir. Cependant, le fonctionnement est basé sur l'arrivée d'un autre utilisateur dans la zone en question. Cela implique donc un enregistrement constant des déplacements des personnes en quête du souvenir afin de savoir lorsque ces derniers pénètrent ou non dans la zone. Cela correspond donc à la partie passive du Geofencing. La partie active implique la géolocalisation précise de la personne à compter du moment où elle se trouve dans la zone.

Le problème éthique et législatif est donc le même que celui lié à la géolocalisation passive. L'atteinte à la vie privée qu'implique ces systèmes est due au fait que le Web d'aujourd'hui dépend d'entreprises privées, particulièrement les grandes entreprises de la tech américaines. Cela implique que les données privées peuvent être utilisées par ces entreprises à des fins commerciales ou marketing sans le consentement des utilisateurs. Les principes du Web 3.0 mentionnés plus tôt prouvent ainsi que cette application s'inscrirait parfaitement dans cette nouvelle version du Web, chacun devenant unique propriétaire de ses données.

Ensuite, les souvenirs placés à des endroits précis peuvent également servir de sorte de « balises ». En s'inscrivant dans un espace où chacun serait entier propriétaire de ses données, avec un pouvoir de contrôle total sur celles-ci, cela pourrait donner une impression de liberté laissant place à des idées s'inscrivant dans l'illégalité.

La communication de lieu de cache d'objets ou de consommables illicites, de réunion, de transaction devient extrêmement simplifiée en raison du partage d'utilisateur à utilisateur d'informations pouvant à terme être supprimé sans aucune trace restante.

En somme, les questions d'éthiques et de légalisation sont au cœur d'une application comme celle-ci. Malgré le fait que les lois sont floues lorsque l'on parle de géolocalisation et des données privées qu'il en résulte, certaines utilisations peuvent en effet s'avérer illégales sans doute possible.

# Business Plan potentiel

Dans l'idée de la création d'un business plan lié à « Shere », plusieurs idées ont émergées :

On peut imaginer un système de récompenses de recherche au sein de l'application, comme « vous avez visité 15 villes » nous gratifie d'une monnaie virtuelle qui permet d'augmenter la visibilité de nos souvenirs dans l'onglet « Découverte » expliqué plus tôt ci-dessus.

La possibilité de gagner des récompenses en visitant des villes inciterait les utilisateurs à explorer davantage de lieux et à interagir avec l'application. Ces récompenses pourraient être échangées contre des avantages tels que des réductions dans les magasins locaux, des visites gratuites de musées ou des bons pour des restaurants. Cela encouragerait les utilisateurs à dépenser de l'argent dans les villes visitées, créant ainsi un cercle vertueux pour les commerçants locaux.

En plus de proposer des partenariats avec des marques pour intégrer un espace shopping dans l'application, Shere pourrait également utiliser les données des utilisateurs pour personnaliser les publicités. L'algorithme de l'application pourrait analyser les habitudes d'achat des utilisateurs et leurs préférences en matière de produits pour proposer des publicités pertinentes.

De plus, les marques pourraient sponsoriser les souvenirs partagés par les utilisateurs, en ajoutant leur logo ou leur nom sur les photos ou les vidéos. Cela offrirait une opportunité pour les marques d'augmenter leur visibilité tout en permettant aux utilisateurs de gagner de l'argent en partageant des souvenirs sponsorisés.

Enfin, l'application pourrait également offrir des réductions ou des offres spéciales pour les produits proposés dans l'espace shopping, afin d'encourager les utilisateurs à acheter directement depuis l'application. Tout cela contribuerait à augmenter l'engagement des utilisateurs et à renforcer la présence de Shere sur le marché des applications de partage de souvenirs.

# Conclusion

En conclusion, ce rapport de projet représente une étape significative dans le développement d'un outil innovant basé sur la géolocalisation et les réseaux sociaux pour la création et la diffusion de souvenirs dans des zones géographiques spécifiques. Ce projet a été guidé par un cahier des charges clair établi au départ, nous permettant de mener à bien les recherches nécessaires, de réaliser l'outil et d'élaborer son design de manière approfondie.

Au cours de ce projet, nous avons également exploré les opportunités offertes par le Web 3.0, en identifiant les fonctionnalités clés à implémenter dans le futur pour améliorer encore davantage l'application. La prise en compte des aspects sociaux, éthiques et légaux a été au cœur de nos préoccupations, soulignant notre engagement envers une approche responsable et éthique du développement technologique.

D'un point de vue technique, ce projet a été enrichissant, nous permettant d'acquérir de nouvelles compétences en programmation et de maîtriser un nouvel environnement de développement. Nous avons également développé une compréhension approfondie des défis et des opportunités liés à la création d'un outil basé sur la géolocalisation et les réseaux sociaux, en mettant l'accent sur la qualité, la sécurité et la convivialité de l'application.

Sur le plan conceptuel, ce projet a été stimulant, amenant la réflexion au niveau des implications sociales, éthiques et légales de l'outil développé. La prise en compte des préoccupations relatives à la vie privée, à la sécurité des données, à l'éthique de l'utilisation des informations de géolocalisation et à la conformité aux lois et réglementations en vigueur fut également une excellente source de réflexion avant de futurs projets, sans aucun doute dans un contexte d'entreprise.

En résumé, ce projet a été une expérience enrichissante et prometteuse, ouvrant la voie à de nouvelles possibilités dans le domaine des réseaux sociaux basés sur la géolocalisation. Nous exprimons notre reconnaissance envers toutes les personnes qui ont contribué à la réalisation de ce projet, notamment nos encadrants, partenaires et collègues. Nous sommes fiers du résultat obtenu et nous sommes convaincus que cet outil peut avoir un impact positif en créant des souvenirs précieux qui reconnecteront les utilisateurs entre eux, tout en respectant les principes éthiques et légaux.

Dans un contexte où la technologie continue d'évoluer rapidement, nous restons attentifs aux développements futurs du Web 3.0 et aux évolutions légales et éthiques du domaine des réseaux sociaux. Nous sommes déterminés à poursuivre notre engagement envers la responsabilité et l'éthique dans le développement de technologies innovantes.

En conclusion, ce projet représente une ambition importante et nous sommes confiants dans son potentiel à contribuer à la création d'expériences sociales positives et à renforcer les liens entre les utilisateurs dans des zones géographiques spécifiques. Nous sommes reconnaissants envers tous ceux qui ont soutenu ce projet et nous sommes impatients de voir les futures applications de cet outil.

# Bibliographie

## **Sources Web :**

[stack overflow error late init property](#)

[stack overflow way to style text view](#)

[stack overflow failed for task](#)

[stack overflow duplicate class](#)

[stack overflow get size firebase](#)

[stack overflow permission error](#)

[stack overflow async error](#)

[stack overflow error inflating](#)

[Mobile app - Kotlin](#)

[Convert Latitude and Longitude to an Address - Kotlin](#)

[Konfetti - Kotlin](#)

[Chat App - Kotlin](#)

[Your Android SDK is missing out of date or corrupted - Error Kotlin](#)

[build gradle internal tasks - Kotlin](#)

[Error execution failed AndroidStudio](#)

[Bug gradle AndroidStudio](#)

[current location Kotlin](#)

[current location Kotlin](#)

[Notification Kotlin device to device](#)

[Notification Kotlin](#)

[error Kotlin](#)

[mapping Permissions and OnMapReady Callback](#)

[Send data fragment to fragment - Kotlin](#)

[how to design an app for android?](#)

[Android material design](#)

[How to pass data between fragments using Bundles](#)

[Geocoding to get Location Address in Android Using Kotlin](#)

[Introducing Firebase App Check - Kotlin](#)

[How to use App Check to protect your custom backend](#)

[Introducing Play Integrity API: Protect your apps and games](#)

[Keeping your app secure with App Check](#)

[Android Material UI Login Design Android Studio](#)

[Android Dashboard UI Design Tutorial](#)

[Xd Android studio profile page](#)

[Session Manager in Android Studio](#)

[how to go one fragment to another fragment by button](#)

[How to Implement Search Bar in Android Studio](#)

[How to Create SearchView on Google Map in Android Studio](#)

[how to open fragment from activity on button click](#)

[How to animate Buttons, TextView, ImageView, etc. in Android Studio](#)

[4 Nice Animations for RecyclerView](#)

[Create RecyclerView Item Animation - Kotlin](#)

[Android Chat App Development](#)

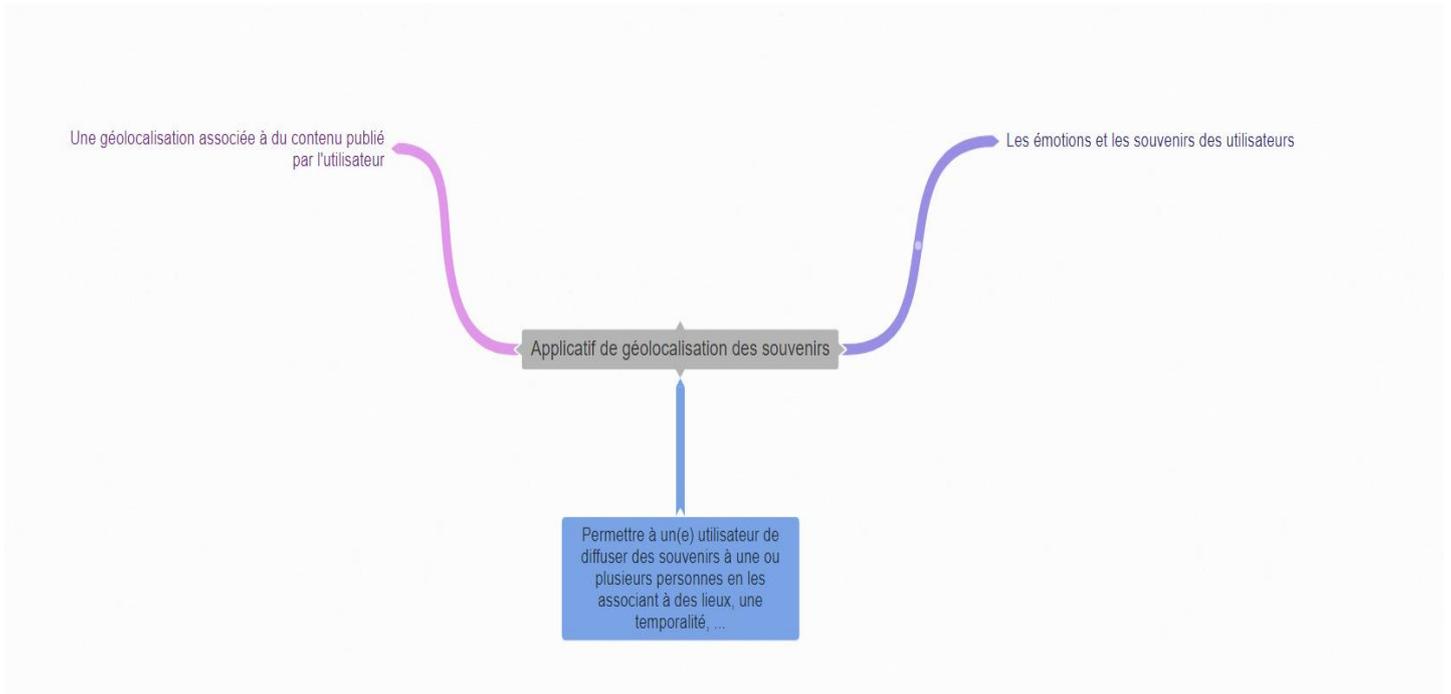
[Open Camera and Display in Image View](#)

[Custom Listview with image and text](#)

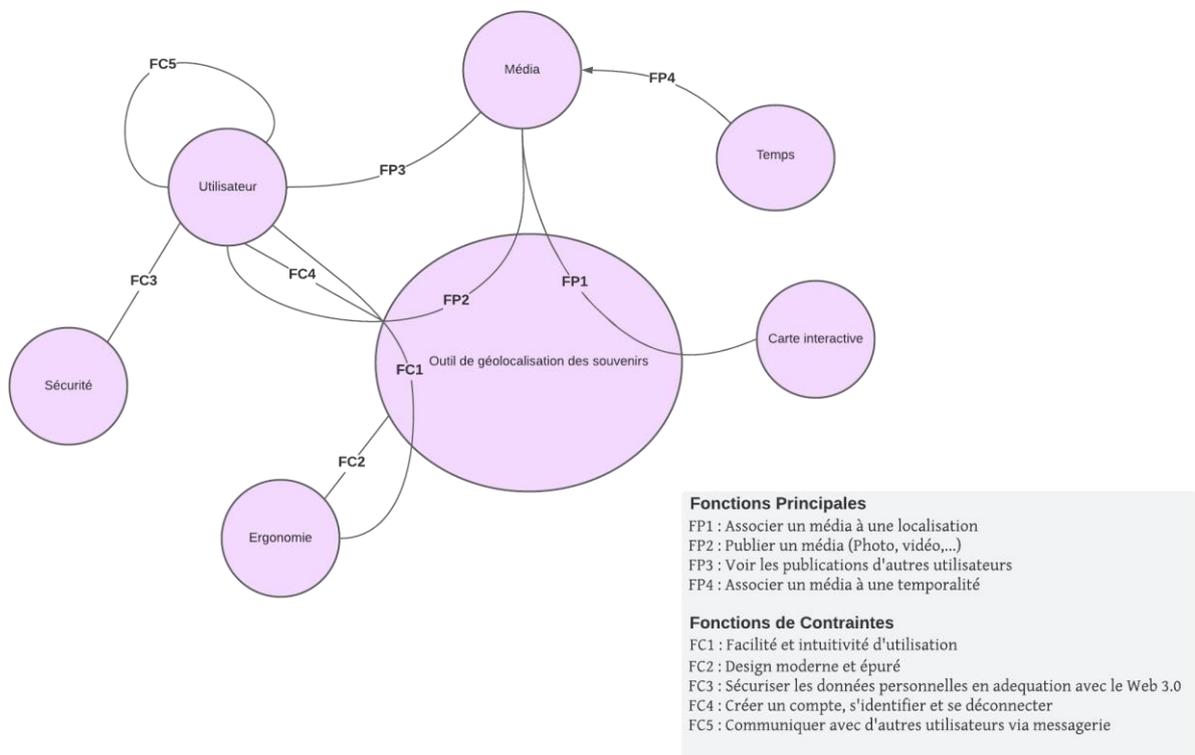
***Sources littéraires :***

***Constants across cultures in the face and emotion – Paul Ekman & Wallace V. Friesen***

# Annexes



*Figure 21: Diagramme Bête à Cornes d'analyse du besoin utilisateur*



*Figure 22 : Diagramme Pieuvre*

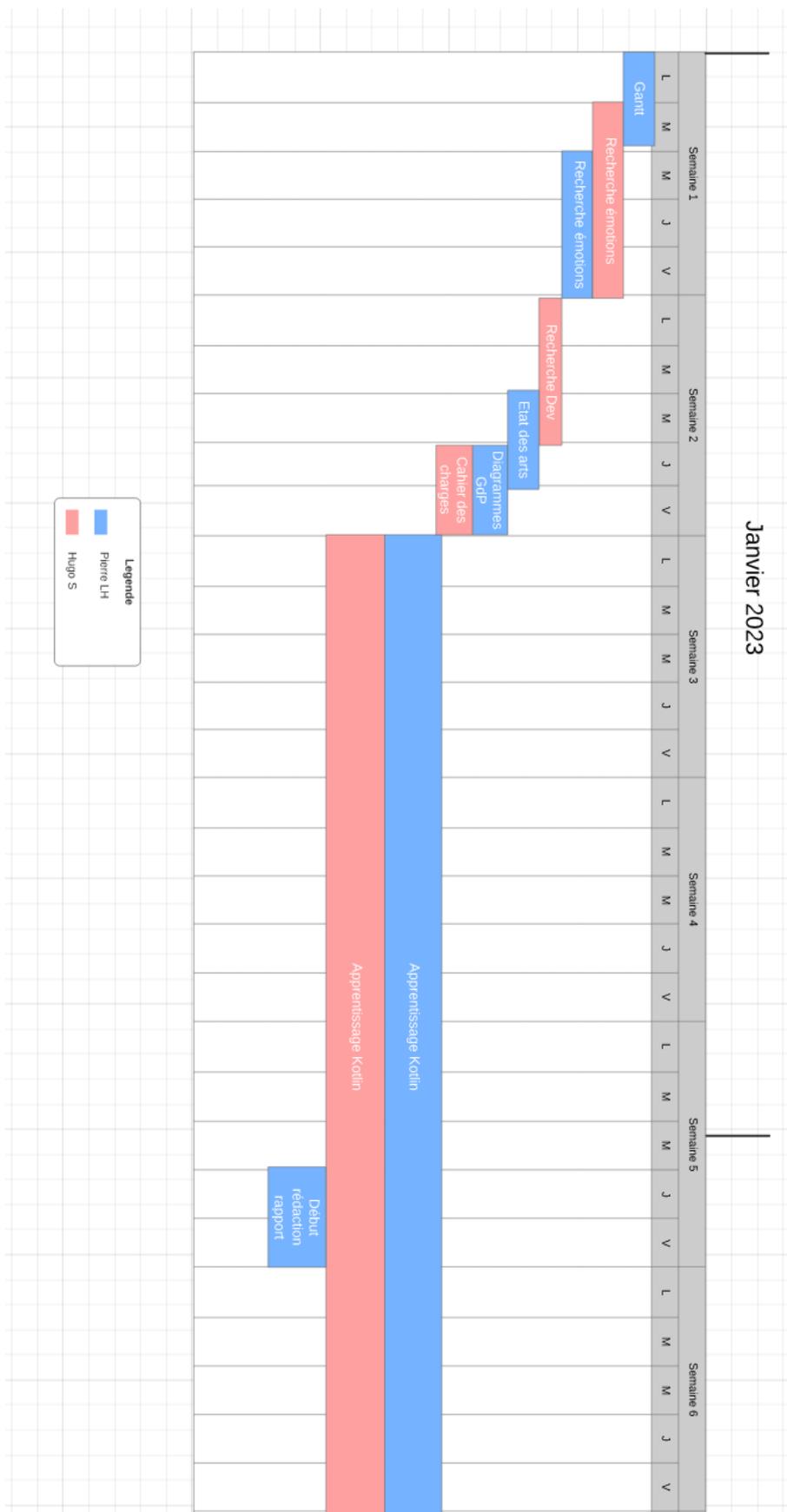


Figure 23: Exemple de diagramme de Gantt

# COMPTE RENDU DE RÉUNION

Réunion n° 3

Mercredi 1er mars 2023

## Participants

- Mikaël Cabon
- Cyril De Sousa Cardoso
- Hugo Sechet
- Pierre L'Hostis

## Ordre du jour

- Mise au point avant le dernier mois et demi

## Mises à jour

- Place de l'émotion en fond, sous-jacente dans l'outil

## Mesures à prendre

- Partir sur le geofencing
- Système de "thermostat"
- Déployer la géolocalisation sans délaisser le côté design

## Annonces finales

- Aller au plus simple techniquement pour la géolocalisation et pour l'application



## Remarques

- Outil de géolocalisation : définir si l'objet est transmis à Google ou non
- Mon média est-il géolocalisé ou je le suis ?
- Comment faire pour paramétrer les personnes avec qui je partage mon contenu ( très privée, public - ?)
- Attention au "trop de contenu" -> aller au plus simple pour l'utilisateur
- Dimension d'émotion sous-entendue
- Détails graphiques ( balise "cœur".)
- Rapport : idée de page Web de transition entre lien de partage et DL de l'app
- Dimension de précision de la géolocalisation à aborder ( en vue de la présentation )
- Contact "BaguetteAdvisor" Antoine Blondeau & Gilles Biannic
- Rapport : penser à la dimension éthico-légale de l'application et ses limites
- Rapport : développer comment mettre progressivement en place le Web 3.0 dans l'app



Figure 24 : Compte-rendu de réunion

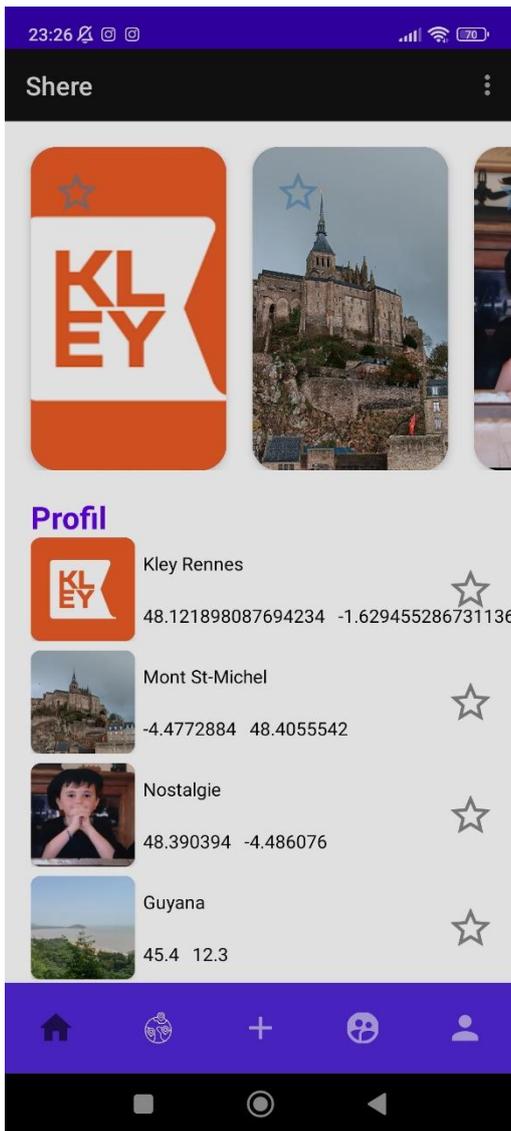


Figure 26: Souvenirs avec leurs coordonnées longitude/ latitude

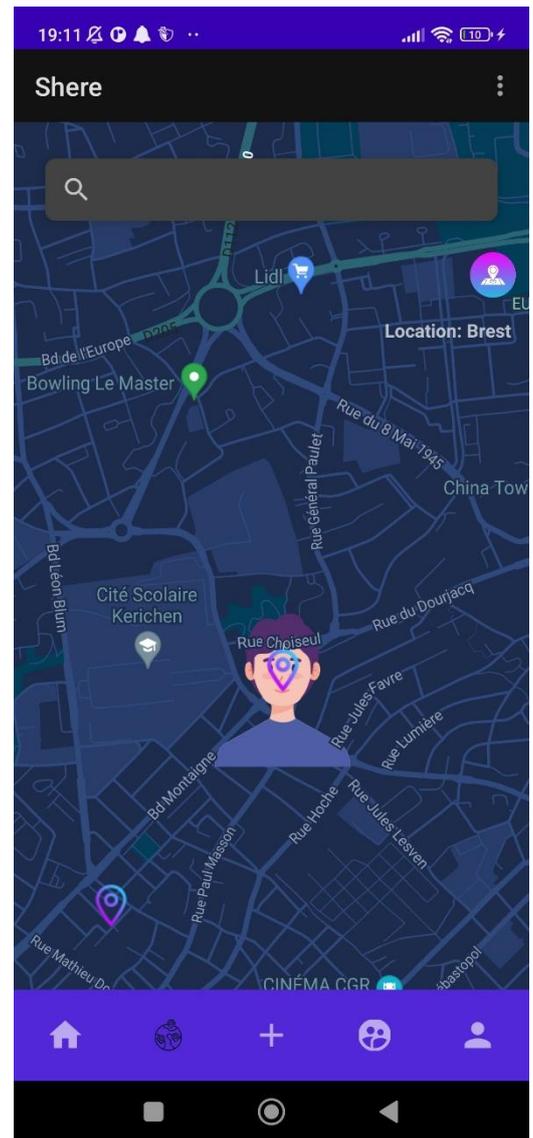
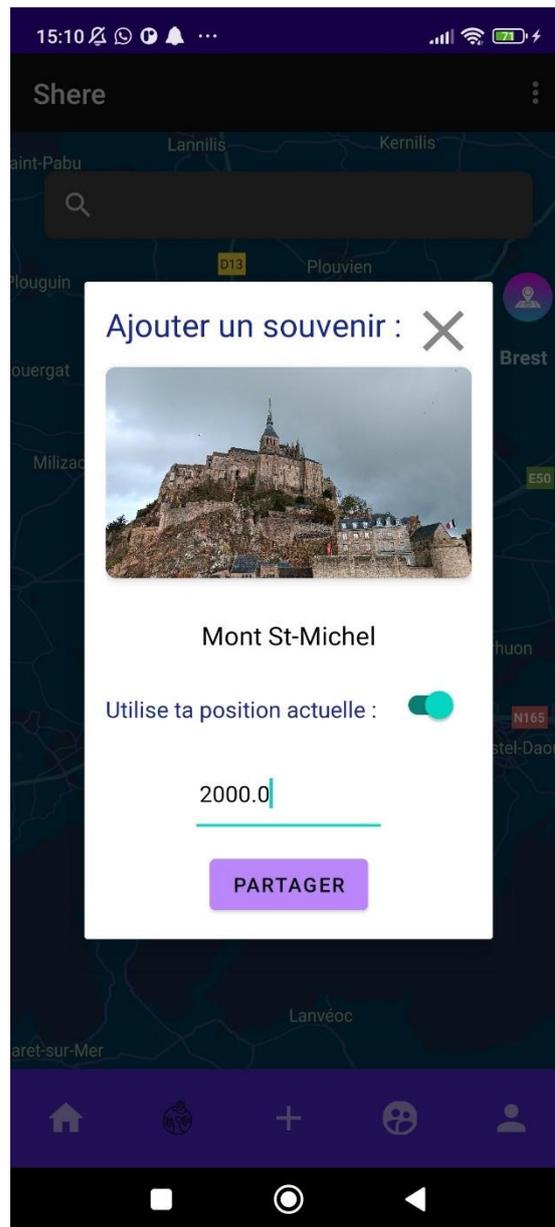


Figure 25: Repère de la position actuelle



*Figure 27 : Souvenir depuis sa localisation actuelle*