



**UNIVERSITE SIDI MOHAMED BEN ABDELLAH**

**FACULTE POLYDISCIPLINAIRE DE TAZA**

**Formation : Master Systèmes Intelligents et Mobiles**

# MEMOIRE DE STAGE DE FIN D'ETUDES

**Effectué au sein de l'entreprise FCPO**

**Sous le thème :**

# DEVELOPPEMENT D'UNE APPLICATION MOBILE POUR LE TRANSPORT SCOLAIRE

**Réalisé par :**

**Mouad TAHIRI**

**Encadré par :**

Pr El Bachir TAZI (FPT USMBA)

Mr Soufiane BOUKHRISS (FCPO)

**Soutenu le 29 /07/2021 devant le jury :**

El Bachir TAZI

Professeur à la Faculté Polydisciplinaire de TAZA

Majid BEN YAKHLEF

Professeur à la Faculté Polydisciplinaire de TAZA

Ismail AKHARRAZ

Professeur à la Faculté Polydisciplinaire de TAZA

Année universitaire : 2020/2021

## Dédicace

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien-être.

Je vous remercie pour tout le soutien et l'amour que vous m'avez apporté depuis mon enfance et j'espère que votre bénédiction m'accompagnera toujours.

Que ce modeste travail soit l'exaucement de vos vœux tant formulés, le fruit de vos innombrables sacrifices,

Puisse Dieu, le Très Haut, vous accorde la santé, le bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.

À mes très chères frères et sœurs

En témoignage de notre affection fraternelle, de nos profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protège et vous garde. Avec tous mes vœux de bonheur, de santé et de réussite.

À ma famille

Veillez trouver dans ce travail l'expression de mon respect le plus profond et ma sincère affection. A mes très chers amis En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble. Veillez trouver dans ce travail l'expression de mon respect le plus profond.

À toutes les personnes qui ont participé de près ou de loin à l'élaboration de ce travail.

## Remerciements

Je remercie d'abord Allah qui m'a doté d'une grande volonté et qui m'a aidé et m'a donné le courage, la force et la patience d'accomplir ce travail.

Je tiens à remercier en premier mes deux encadrants le professeur **El Bachir TAZI** et Monsieur **Soufiane BOUKHRISS** pour l'attention qu'ils m'ont accordée et pour m'avoir orienté dans le bon sens en vue de mener à bien la réalisation de ce projet de stage de fin d'études.

Mes grands remerciements vont essentiellement aux membres du jury le professeur **Majid BEN YAKHLEK** et le professeur **Ismail AKHARRAZ** pour avoir accepté d'évaluer et enrichir ce modeste travail par leurs remarques, suggestions et critiques.

Je tiens à remercier tous les professeurs intervenants dans notre formation en master systèmes intelligents et mobiles pour leurs efforts et leurs enseignements les plus fructueux. Je remercie tout particulièrement le défunt le professeur **Abderrahim SAAIDI** le coordinateur pédagogique de la formation pour tous ses efforts et ses conseils que Dieu l'ait en sa miséricorde.

Enfin merci à tous les membres de ma famille et mes amis que je n'ai pas cité.

## Résumé

Le présent document est le fruit de mon travail dans le cadre du stage de fin d'étude, pour l'obtention du diplôme de mastère spécialisé en systèmes intelligents et mobiles, qui a été réalisé au sein de l'entreprise FCPO. Notre travail consiste à utiliser le Framework de développement mobile hybride «multi plateformes» React Native et réaliser une application mobile permettant de faciliter la mission de transport scolaire, minimiser le temps du voyage et garder une communication permanente entre les parents et les chauffeurs à travers des messages de notification.

Nous avons commencé par l'élaboration d'une étude fonctionnelle pour aboutir à l'élaboration d'une conception du projet. Ensuite nous avons dressé une étude comparative des deux approches de développement mobile à savoir l'approche Native et l'approche Hybride. L'approche hybride répondant mieux à notre besoin a été adoptée pour la réalisation de l'application. Reste à signaler que pour suivre l'évolution des technologies de l'information et l'émergence du monde mobile, notre application a été conçue pour tourner sous les deux plateformes mobiles à savoir : Android, IOS.

# Table des matières

Dédicace	2
Remerciements	3
Résumé	4
Table des matières	5
Liste des figures	8
Liste des tableaux	8
List d'acronymes	9
Introduction générale	10
Chapitre 1 : Présentation de l'organisme d'accueil	11
I. <i>Présentation de L'entreprise FCPO</i>	11
1.    FCPO en bref :	11
2.    Métiers et Secteurs d'activités :	11
Chapitre 2 : CONTEXTE GENERAL DU PROJET	13
I.    Présentation générale du projet	13
II.   Problématique :	13
III.  Solution à développer	14
Chapitre 3 : Analyse et spécification des besoins	15
I.    Introduction	15
II.   Etude fonctionnelle	15
1) <i>Identification de l'acteur</i>	15
2) <i>Besoins fonctionnels</i>	15
3) <i>Les besoins non fonctionnels</i>	15
4) <i>Diagrammes de cas d'utilisation</i>	16
i.    Explications textuelles des cas d'utilisation	17
III.  Conclusion	18
Chapitre 4 : ETUDE CONCEPTUELLE ET TECHNIQUE	19
I.    Introduction	19
II.   Etude Technique	19
1) <i>Spécifications techniques :</i>	19
2) <i>Architecture</i>	19

i.	Architecture mobile :	19
ii.	Architecture serveur :	20
3)	<i>Les services web</i>	20
i.	Intérêt des services web	21
ii.	Qu'est-ce qu'une API ?	21
iii.	Qu'est-ce qu'une Api REST ?	21
iv.	REST ou SOAP, lequel choisir ?	21
v.	La sécurité des appels des web services	23
4)	<i>Développement Hybride et Natif :</i>	23
i.	Quelle est la meilleure approche :	23
III.	Etude conceptuelle	24
1)	<i>Introduction</i>	24
2)	<i>Les modèles dynamiques</i>	24
i.	Diagrammes d'activité :	24
ii.	Diagrammes de séquences	27
3)	<i>Le modèle statique (diagramme de classe)</i>	29
i.	Explications textuelles de diagramme de classe	30
IV.	Conclusion	30
	Chapitre 5 : REALISATION	31
I.	Introduction	31
II.	Outils du développement	31
1)	<i>IDEs</i>	31
i.	VS Code	31
2)	<i>Outils Google</i>	31
i.	Google maps	31
ii.	Contrôle de version	32
3)	<i>Environnement de développement</i>	32
i.	NodeJS	32
ii.	React Native	34
iii.	TypeScript	34
iv.	Autres outils	35
III.	Mise en œuvre	37
1)	<i>Écran de démarrage</i>	37
2)	<i>Écran d'authentification</i>	38
3)	<i>L'espace de Chauffeur</i>	38
i.	L'écran d'accueil	38
4)	<i>L'espace de parent</i>	41
i.	L'écran d'accueil	41
ii.	Contacter l'établissement	42
iii.	Changer la localisation	45
iv.	Consultation du profil	45
v.	Recevoir les notifications	48

<b>IV. Conclusion</b>	<b>48</b>
<b>Conclusion générale</b>	<b>49</b>
<b>Références</b>	<b>50</b>

## Liste des figures

Figure 1 les services proposés par l'entreprise FCPO.....	12
<b>Figure 2: Diagramme de cas d'utilisation.....</b>	<b>16</b>
<b>Figure 3: Schéma de architecture MVC.....</b>	<b>20</b>
<b>Figure 4 : Diagramme d'activité : modification du login.....</b>	<b>25</b>
<b>Figure 5: Diagramme d'activité : Lancement de voyage .....</b>	<b>26</b>
Figure 6 Diagramme d'activité : changement de la destination d'un enfant.....	27
<b>Figure 7 : Diagramme de séquence déroulement de lancement de voyage .....</b>	<b>28</b>
<b>Figure 8 : Diagramme de classe .....</b>	<b>29</b>
Figure 9 splash screen .....	37
Figure 10 écran d'authentification [android/IOS] .....	38
Figure 11 écran d'accueil pour le chauffeur [android/IOS] .....	38
Figure 12 les étapes de lancement de voyage [android/IOS] .....	39
Figure 13 écran de navigation GPS de voyage [android/IOS] .....	40
<b>Figure 14 : Ecran d'accueil pour le parent [android/IOS] .....</b>	<b>41</b>
<b>Figure 15: Ecran contacter l'établissement.....</b>	<b>42</b>
<b>Figure 16 : Ecran de paramétrage [android/IOS] .....</b>	<b>43</b>
<b>Figure 17 : Démarche de modification, paramétrage de compte et notification [android/IOS] .....</b>	<b>44</b>
<b>Figure 18 : Ecran de changement de la localisation et la distance de notification .....</b>	<b>45</b>
<b>Figure 19 : Scénario de manipulation du profil de parent sur android .....</b>	<b>46</b>
<b>Figure 20 : Scénario à suivre pour manipuler le profil de parent sur IOS .....</b>	<b>47</b>
<b>Figure 21 : Ecran de notification android/IOS .....</b>	<b>48</b>

## Liste des tableaux

<b>Tableau 1: Spécification techniques du projet.....</b>	<b>19</b>
<b>Tableau 2 : Comparaison des protocoles de communication .....</b>	<b>22</b>



## List d'acronymes

- TPE : Très Petites Entreprises
- PME : Petites et Moyennes Entreprises
- API : Application programming interface
- REST : Representational State Transfer
- SOAP : Simple Object Access Protocol
- UML : Unified Modeling Language
- MVC : Model-View-Control
- GPS : Global Positioning System
- IDE : Integrated Development Environment
- IOS: Iphone operating system
- IHM : Interface Homme Machine
- W3C: The World Wide Web Consortium
- XML : eXtensible Markup Langage
- SDK : Software Development Kit

## Introduction générale

En 2020, le nombre d'internautes s'élève à plus de 4.5 milliards, en croissance de 5.5% par rapport à l'année 2019, le monde du digital s'agrandit à chaque instant. Et pour une entreprise, avoir un site internet, ou une application mobile est primordial. Ainsi elle pourra contrôler son image, apporter de la visibilité supplémentaire, créer une véritable histoire et accroître sa propre communauté.

Toutefois, le développement d'une telle application doit cibler plusieurs plateformes, quitte à faire face au problème de la fragmentation concernant les applications mobiles et par la suite faire limiter le nombre d'utilisateurs de l'application. C'est pour cela que le développement mobile multiplateforme apparaît.

C'est dans ce contexte l'entreprise FCPO m'a confié, dans la cadre de mon projet de fin d'étude, la mise en place d'une solution de mobile qui permettra d'exécuter des fonctionnalités de géolocalisation à partir du mobile pour envoyer les notifications de proximité qui sera bien servi et organiser le transport scolaire.

Le présent rapport est une synthèse du travail réalisé tout au long de la période de mon stage. Il est structuré selon quatre chapitres couvrants l'ensemble des axes de notre travail. Le premier chapitre décrit l'organisme d'accueil FCPO. Le deuxième chapitre montre la problématique et l'analyse des besoins, cite la solution à proposer et les fonctionnalités à mettre en place. Dans le troisième chapitre nous avons effectué une étude fonctionnelle du projet, après quoi nous avons établi le diagramme de cas d'utilisation. Quant au quatrième chapitre, il définit l'architecture, l'étude fonctionnelle, l'étude comparative entre les approches de programmation mobile, la conception générale du projet et la démarche que nous avons suivie. Le dernier chapitre, quant à lui, est consacré à la réalisation et la présentation des interfaces de notre application fruit travail réalisé. Enfin, une conclusion générale pour dresser le bilan de ce travail de projet de stage de fin d'études.

# Chapitre 1 : Présentation de l'organisme d'accueil

---

## Résumé :

Dans ce premier chapitre, on expose le cadre général de mon stage. Tout d'abord on présente l'organisme d'accueil, ses secteurs d'activités, et sa méthodologie. Par la suite, on va se focaliser à la présentation du projet, dans lequel nous précisons le contexte du stage, la problématique ainsi que les objectifs visés, et finalement nous allons éclaircir le planning suivi lors de la réalisation du projet.

## I. Présentation de L'entreprise FCPO

### 1. FCPO en bref :

L'entreprise FCPO est une agence de développement digitale. Est une société de services en ingénierie informatique experte dans l'ingénierie des technologies de pointe et dans les solutions informatiques d'entreprise. Alliant performance et innovation, cette société s'occupe d'accompagner les entreprises dans leurs transformations numériques et dans la construction de la stratégie digitale.

L'entreprise FCPO a été créée en 2013 à Oujda. Huit ans après, ils se considèrent encore comme une startup.

Le fondateur de FCPO est Soufyane BOUKHRISS, aujourd'hui c'est le directeur générale de l'agence.

### 2. Métiers et Secteurs d'activités :

FCPO s'articule autour du B2B et B2C : en effet, l'accompagnement proposé par FCPO concerne autant les entreprises (en particulier les PME et TPE) que les professionnels, sans oublier partiellement les blogueurs (qui sont sans structure entrepreneuriale).

La société FCPO Sarl a plusieurs métiers complémentaires qui sont le Marketing, Digital Design, le développement web et la conception créative. La société FCPO dispose de principaux atouts lui permettant d'innover avec le meilleur des technologies et de placer l'expertise des métiers de ses clients au cœur de son savoir-faire. Ces atouts sont les suivants :

- L'excellence technique est un principe fondateur de l'entreprise : Amélioration des processus, remise en cause permanente et analyse des succès et des échecs. Cette recherche constante de la qualité est aussi la garantie de la réussite durable.
- La passion est une valeur essentielle qui anime l'ensemble des équipes de FCPO. Cette passion nous pousse toujours à vouloir concevoir des solutions inédites.

- FCPO s'inscrit dans une démarche d'amélioration continue où l'innovation est une composante primordiale. L'ambition de devenir une référence sur les nouvelles technologies. Ainsi, les équipes voient et font les choses différemment pour trouver la meilleure voie.

FCPO regroupe tous les métiers interactifs au service des entreprises et des marques.

Sa mission principale est d'apporter des solutions concrètes et modernes qui correspondent aux besoins de ses clients, à des tarifs compétitifs.

L'agence FCPO accompagne ses clients dans la création de sites internet modernes et haut de gamme. Et il est capable de répondre à toutes ses exigences et d'élaborer une véritable stratégie digitale.



*Figure 1 les services proposés par l'entreprise FCPO*

## Chapitre 2 : CONTEXTE GENERAL DU PROJET

---

### Résumé :

Dans ce chapitre nous allons décrire la problématique et faire l'analyse des besoins, la solution à proposer et les fonctionnalités à mettre en place.

### I. Présentation générale du projet

La transformation digitale, que l'on appelle parfois aussi transformation numérique, désigne le processus qui permet aux entreprises d'intégrer toutes les technologies digitales disponibles au sein de leurs activités.

La mise en place d'une application mobile pour gérer et actualiser et assurer une bonne démarche de transport scolaire que ça soit à la part des parents des élèves et à la part de l'école est parmi les piliers de cette transformation.

### II. Problématique :

Pour les écoles qui opèrent toujours en mode présentiel, le recours aux bus de transport scolaire est toujours d'actualité. Avec le tumulte quotidien du trafic routier, un trajet vers et depuis l'école peut se transformer en un véritable calvaire.

D'autre part on connaît tous que Le retard posé par le transport scolaire au Maroc est un problème vécu tant que l'autobus doit quotidiennement transporter les élèves sous forme des groupes qui sont situé dans une zone.

Par exemple si on estime que autobus doit transporter par jour 20 élèves et que chaque élève peut retardait l'autobus 3 min donc on aura 60 min de retard . Ainsi que les parents ne connaissent pas localisation de autobus où il est pour qu'ils puissent préparer ses enfants et d'être prêt à l'heure.

Ce qui nous oblige de trouver une solution pertinente pour surmonter ce problème et assurer une démarche normale des séances et gagner le temps.

### **III. Solution à développer**

Notre projet consiste donc à proposer une solution répondant aux besoins fonctionnels écoles. Cette solution devra assurer les éléments suivants :

- ✓ Notifier les parents pour lui informer à la proximité de l'autobus.
- ✓ Permettre aux parents d'informer l'absence de ses enfants.
- ✓ Permettre aux parents de modifier la localisation où ils sont.
- ✓ Permettre aux parents de changer la distance minimale pour recevoir une notification selon la distance choisie.

## Chapitre 3 : Analyse et spécification des besoins

---

### Résumé :

Dans ce chapitre nous allons effectuer une étude fonctionnelle du projet, après nous présenterons le diagramme de cas d'utilisation.

### I. Introduction

La phase d'analyse et spécification des besoins présente une étape primordiale dans le cycle de développement d'un projet. En effet, l'expression du besoin est une phase très sensible nécessitant un effort de la part des deux contractants, le client doit bien expliquer et exprimer exactement le besoin. De son côté, l'équipe réalisatrice doit relever le maximum d'informations pour bien répondre à ces besoins.

### II. Etude fonctionnelle

L'objectif de notre travail est développer une application mobile qui permettra aux parents et les écoles d'intégrer tous les deux pour bien gérer la processus de transport scolaire qui ça sera bien organiser afin de diminuer le retard.

#### 1) Identification de l'acteur

Un acteur est une entité externe qui interagit avec un système. En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin.

Notre solution contient deux acteurs, le parent et le chauffeur, ils effectuent toutes les opérations qu'offre l'application.

#### 2) Besoins fonctionnels

Les besoins fonctionnels sont les besoins qui déterminent les fonctionnalités indispensables auxquels doit répondre notre solution. Nous avons cité précédemment que l'application doit agir comme une agence en ligne, afin de diminuer le recours aux moyen traditionnels de gestion, et d'intégrer de plus en plus les moyens technologiques.

#### 3) Les besoins non fonctionnels

Les besoins non-fonctionnels sont des besoins/contraintes liés à l'implémentation et à l'interopérabilité générale. Et pour compléter les besoins fonctionnels, notre projet devra respecter un ensemble de propriétés contribuant à une meilleure qualité de la solution obtenue. Parmi ces critères on trouve :

**Fiabilité** : L'application ne doit pas afficher aux utilisateurs des erreurs, par contre toutes les exceptions doivent être gérées et toute incidence doit être déclarée dans les logs du côté back-end pour l'identifier facilement, et pour la résoudre rapidement.

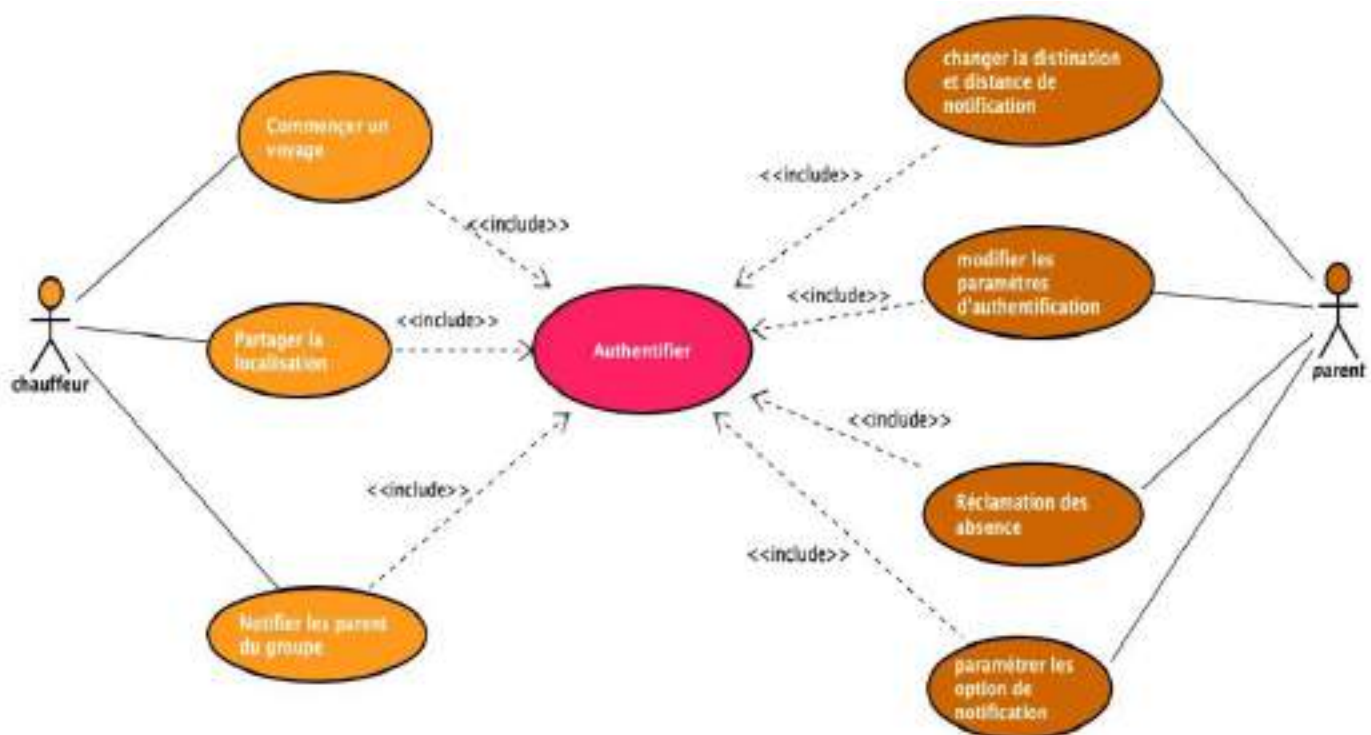
**La facilité d'utilisation (utilisabilité)** : L'application devra être capable de :

- ✓ Facilité de compréhension,
- ✓ Facilité d'apprentissage,
- ✓ Facilité d'exploitation/d'utilisation.

**La maintenabilité** : L'application devra être capable de :

- Facilité d'analyse,
- Facilité de modification,
- Stabilité,
- Testabilité.

#### 4) Diagrammes de cas d'utilisation



*Figure 2: Diagramme de cas d'utilisatioon*

Avant de terminer l'étude fonctionnelle et la spécification des besoins, nous allons présenter les diagrammes de cas d'utilisation permettant de donner une vision globale sur la solution à développer.



Le diagramme de cas d'utilisation ci-avant illustre toutes les fonctionnalités à effectuer par les deux acteurs de notre solution à savoir : chauffeur et parent.

L'application offre plusieurs services qui doivent s'effectuer juste avec l'authentification par un nom d'utilisateur et un mot de passe.

L'identification est faite une bonne fois pour toutes. En effet, l'utilisateur n'a pas besoin de s'authentifier une autre fois, sauf dans le cas de la désinstallation ou la suppression des données de l'application.

**i. Explications textuelles des cas d'utilisation**

- **Chauffeur**

- a. **Commencer un voyage**

**Scénario principal :** après l'authentification le chauffeur sélectionne le groupe des élèves qui seront s'afficher sous forme des points sur la carte géographique, puis il lance le mode navigation pour commencer le voyage.

- b. **Partager la localisation**

**Scénario principal :** après départ de voyage l'application sauvegarde fréquemment la position géographique de l'appareil où il est installé jusqu'à ce que le chauffeur appuie sur le bouton d'arrêt.

- c. **Notifier les parents du groupe**

**Scénario principal :** pendant le voyage et dans le moment où l'application sauvegarde le nouveau point géographique la distance entre cette dernier et le point de destination de chaque élève du groupe sera calculée et comparée avec la distance de notification défini par le parent de l'élève. si la distance calculé est inférieure ou égale la distance de notification le parent reçoit une notification.

- **Parent**

- a. **Changer la destination et la distance de notification**

**Scénario principal :** le parent peut changer la position géographique de destination et la distance où il veut reçoit une notification soit de façon collective dont ses enfants prennent la même destination et distance de notification, ou individuelle qui permet aux parents de modifier ces paramètres pour un enfant choisi.

- b. **Modifier les paramètres d'authentification**

**Scénario principal :** le parent sera authentifié par le saisi de son nom utilisateur et mot de passe et à chaque instant il peut modifier ses paramètres de l'authentification.

- c. **Réclamer les absences**

**Scénario principal :** pour éviter le retard le parent à le droit d'informer l'école à propos de l'absence de son enfant pour que l'autobus ne prend pas en considération la position de l'élève.

**d. Paramétrer les options d'alerte**

**Scénario principal :** le parent à le droit de désactiver le reçoit des notifications ainsi qu'il peut choisir la tonalité de l'alerte et il peut ajouter aussi l'option de vibration.

### **III. Conclusion**

Dans ce chapitre nous avons effectué une étude de la solution de notre projet selon l'approche fonctionnelle. En effet nous avons défini les besoins fonctionnels et non fonctionnels, puis nous avons présenté les différents cas d'utilisation de notre application ainsi que leur description et déroulement.

Avant de commencer la réalisation, il a fallu d'abord faire une étude technique de la conception détaillée de l'application en vue de contourner la solution de tous ses aspects, ceci sera l'objet du prochain chapitre.

## Chapitre 4 : ETUDE CONCEPTUELLE ET TECHNIQUE

### Résumé :

Dans ce chapitre nous allons présenter l'architecture, l'étude fonctionnelle, étude comparative entre les approches de programmation mobile, la conception générale du projet et la démarche que nous avons suivie.

### I. Introduction

Après l'étape de l'étude préalable et la spécification des besoins, nous réservons cette partie à la conception détaillée, et l'étude technique de notre projet. En effet, nous allons commencer par présenter l'univers Android, les services web, et l'architecture de notre projet. Ensuite on va décrire les différents modèles statiques et dynamiques, en utilisant le formalisme UML.

### II. Etude Technique

#### 1) Spécifications techniques :

L'outil doit être en mesure de répondre à un certain nombre d'exigences techniques. Celles-ci se résument dans les objectifs présentés dans le tableau suivant :

Objectif	Description
<b>Multiplateformes</b>	L'application doit pouvoir être accessible depuis les deux plateformes iOS et Android.
<b>Fluidité du déploiement</b>	L'application doit marcher sans problèmes dans toutes les versions antérieures du SE.
<b>Confidentialité</b>	Les utilisateurs du système sont identifiés par leurs noms et leurs mots de passe.
<b>Optimisation du temps de réponse</b>	Aucune donnée ne doit être stockée localement.
<b>Sécurité des transactions</b>	Toutes les transactions du client vers le serveur doivent être sécurisées.

*Tableau 1: Spécification techniques du projet*

#### 2) Architecture

##### i. Architecture mobile :

Deux approches possibles lorsque l'on débute un projet d'application mobile ciblant plusieurs plateformes :

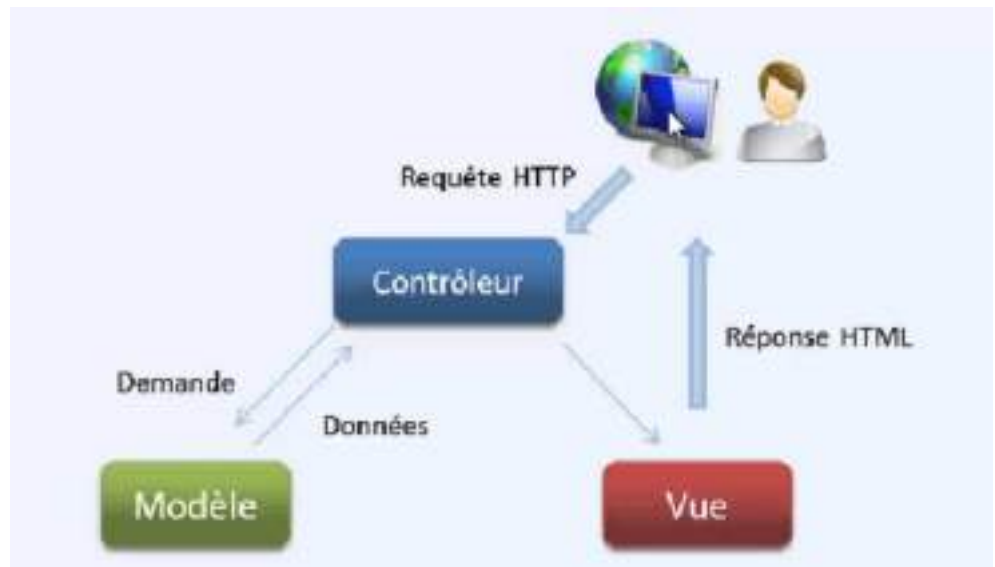
- L'approche native : consiste à développer l'application pour chacune des plateformes.

- L'approche hybride : qui consiste à développer une Application réalisée via un site Web optimisé pour mobile (**Web Apps**).

## ii. Architecture serveur :

Pour la séparation des données, des traitements et de la présentation, On adopte l'architecture MVC. Programmer en utilisant MVC sépare l'application en 3 parties principales : Le Modèle - La Vue - Le Contrôleur :

- **Le modèle** : Il représente les données et les règles métiers. C'est dans ce composant que s'effectuent les traitements liés au coeur du métier. Les données peuvent être liées à une base de données, des services web, etc.
- **La vue** : correspond à l'IHM. Elle présente les données et interagit avec l'utilisateur.
- **Le contrôleur** : se charge d'intercepter les requêtes de l'utilisateur, d'appeler le modèle puis de les rediriger vers la vue adéquate. Il ne doit faire aucun traitement. Il ne fait que de l'interception et de la redirection.



*Figure 3: Schéma de architecture MVC*

## 3) Les services web

Selon la définition du W3C (World Wide Web Consortium), un Web service (ou service Web) est une application callable via Internet - par une autre application d'un autre site Internet - permettant l'échange de données (de manière textuelle) afin que l'application appelante puisse intégrer le résultat de l'échange à ses propres analyses. Les requêtes et les réponses sont soumises à des standards et normalisées à chacun de leurs échanges.

### **i. Intérêt des services web**

- Décomposition d'une application en briques fonctionnelles ou unités logiques applicatives.
- Composant réutilisable fournissant des données et des services à d'autres Applications.
- Facilite l'interopérabilité des différents modèles de composants en interne comme en externe.
- Met en relation des systèmes hétérogènes.
- Utilisation et intégration très facilitées de composants métiers de partenaires.
- Agrégation de plusieurs services de métiers différents sur un même site (horaires/réservation train, assurance, hôtel...).

### **ii. Qu'est-ce qu'une API ?**

L'API, pour Application Programming Interface, est la partie du programme qu'on expose officiellement au monde extérieur pour manipuler celui-ci. L'API est au développeur ce que l'User Interface est à l'utilisateur. Cette dernière permet d'entrer des données et de les récupérer la sortie d'un traitement. Initialement, une API regroupe un ensemble de fonctions ou méthodes, leurs signatures et ordre d'usage pour obtenir un résultat.

### **iii. Qu'est-ce qu'une Api REST ?**

Une API REST se doit d'être sans état ou stateless en anglais. La communication entre le client et le serveur ne doit pas dépendre d'un quelconque contexte provenant du serveur. Ainsi, chaque requête doit contenir l'ensemble des informations nécessaires à son traitement. Cela permet au de traiter indifféremment les requêtes de plusieurs clients via de multiples instances de serveurs.

### **iv. REST ou SOAP, lequel choisir ?**

Après le choix des web services comme moyen de communication entre le client et le serveur, vient le choix du protocole de communication. Lors de nos recherches, nous avons trouvé deux protocoles qui ont suscité notre curiosité : REST et SOAP, ce qui nous a mené à faire une étude comparative afin de faire le bon choix.

REST et SOAP sont des styles architecturaux qui permettent la transmission d'objets distants. Ce sont eux qui autorisent les objets à invoquer des méthodes d'objets situées sur un autre serveur.

	<b>SOAP</b>	<b>REST</b>
<b>Vision et but de l'application</b>	Utiliser le web pour échanger des objets	Ouvrir le web aux applications
<b>Type d'administration</b>	SOAP utilise des requêtes cachées dans des fichiers XML ainsi son administration comme celle d'un EAI	REST utilise des requêtes HTTP et donc son administration est comme celle du Web.
<b>Type de développement</b>	Un service SOAP se met en place comme le déploiement d'un objet. Cela ouvre la porte à l'utilisation de langages de programmation à typage statique	Un service REST se met en place comme une page Web. Cela ouvre la porte à des développements rapides, aux tests possibles avec un simple navigateur et à l'utilisation de langage de scripts
<b>Interopérabilité et plateformes</b>	SOAP utilise beaucoup de normes et versions de chaque normes ce qui diminue son interopérabilité	REST n'a besoin que d'une pile HTTP pour n'importe quelle plateforme


**Tableau 2 : Comparaison des protocoles de communication**

- REST a été la meilleure solution comme protocole de communication du fait que :

**En raison de son évolutivité :** Ce protocole se distingue par son évolutivité. Grâce à la séparation client / serveur, le produit peut être mis à l'échelle par une équipe de développement sans trop de difficultés.

**En raison de sa flexibilité et de sa portabilité :** Avec l'exigence indispensable pour que les données d'une des requêtes soient correctement envoyées, il est possible d'effectuer une migration d'un serveur à un autre ou d'apporter des modifications à la base de données à tout moment. Avant et arrière peuvent donc être hébergés sur différents serveurs, ce qui constitue un avantage de gestion important.

**En raison de son indépendance :** En raison de la séparation entre le client et le serveur, le protocole facilite le développement indépendant des différents domaines d'un projet. De plus, l'API REST s'adapte à tout moment à la syntaxe et à la plateforme de travail. Cela offre la possibilité d'essayer plusieurs environnements en cours de développement.

 Le choix de la solution mobile que nous avons appréhendé était plus compatible avec ce protocole.

## v. **La sécurité des appels des web services**

### • **Le concept du jeton**

L'un des principes clés de REST est qu'il est sans état. Cela signifie que le serveur ne conserve jamais l'état utilisateur. Dans le contexte de la sécurité, cet aspect a des conséquences lors de la mise en oeuvre de la sécurité. Cela signifie que les indications d'authentification doivent être envoyées et vérifiées à chaque fois.

Le mécanisme du jeton consiste à intégrer dans le Header de chaque requête http une clé chiffrée et unique, qui permet d'identifier l'utilisateur qui a tenté d'effectuer cette requête.

Quel que soit la requête http, s'elle ne contient pas un jeton valide, elle sera refusée, et un message d'erreur sera relevé.

Le jeton est obtenu ou renouvelé à chaque connexion, ce dernier est stocké d'une façon sécurisée dans le stockage privé interne de l'application dans le terminal mobile.

## 4) **Développement Hybride et Natif :**

Les applications natives sont des applications construites et installées sur des plateformes spécifiques, comme iOS ou Android, en utilisant un SDK spécifique à chacune. Par exemple, pour l'iPhone et l'iPad d'Apple, les applications sont conçus et développées sur iOS : écrits dans Xcode en utilisant l'Objective-C. Quand-à Android a sa propre version de Java et Windows utilise C #, et ainsi de suite.

Parmi les inconvénients des applications natives, c'est qu'elles sont écrites pour une plate-forme et ne peuvent pas être déployées sur une autre. Aussi, les développeurs ont toujours besoin de ressources supplémentaires pour développer et maintenir chaque plate-forme, qui est toujours coûteux, d'où le développement mobile hybride apparaît.

### i. **Quelle est la meilleure approche :**

Le meilleur choix dépend du type d'application qu'on souhaite développer. Par exemple, les applications telles que des jeux vidéo et les applications qui contiennent des animations favoriseraient le développement natif, alors que les applications hybrides peuvent être mieux adaptées pour les applications d'entreprise mobiles, car ils sont multiplateformes, d'où la cible est un marché vaste. Ci-après quelques critères qu'on a pris en considération lors du choix de notre solution de développement.

**Complexité :** Quelle est le niveau de complexité de l'application ? Est-ce-que c'est une application rapide qui accède à un service de base de données ou sur le Web pour certaines données à afficher ? – dans notre cas une application Web mobile peut suffire.

- **Performance** : Quel type de rendement est exigé par l'application, par exemple, pour regarder en temps réel des données à partir du réseau, la performance de l'application mobile dépend de la latence du réseau et des capacités du serveur. -- Les deux : hybrides et natif offrent des bonnes performances.
- **La connectivité et la disponibilité** : Quel est le type de connectivité exigé ? Est-ce que l'application exige d'être connectée en permanence afin de récupérer les données récentes ? - Les applications natives et hybrides peuvent être construites pour fonctionner en ligne et même hors ligne.
- **multi-plate-forme Exigences** : Ce critère fait la grande différence, et seules les applications hybrides qui supportent le développement multi-plate-formes.
- **Appareil-Services d'accès** : Les applications natives ou hybrides permettent d'accéder aux services périphériques locaux, tels que la caméra, app contacts, accéléromètre, etc.
- **Ressources** : Pour le développement natif, Le nombre de développeurs consacrer à la construction et au développement de l'application mobile, leur niveau de compétence aussi l'environnement de travail « ici un système d'exploitation de chaque plate-forme, de langage de programmation bien spécifique, et des devices pour les tests » rend le travail coûteux, et si on souhaite ajouter une autre plate-forme, donc un autre langage, une autre SDK, de nouvelles compétences.

D'après cette étude on constate que le développement hybride est globalement plus productif. Ainsi il offre à nous un investissement sérieux dans le mobile. Mais sans oublier qu'ils ne vont jamais remplacer d'une manière entière le natif.

### III. Etude conceptuelle

#### 1) Introduction

Dans cette partie on va présenter la conception de notre application, une étude conceptuelle détaillée des différents diagrammes sera adoptée pour bien comprendre notre application et son interaction avec les différents composants de l'architecture précédemment présentée.

#### 2) Les modèles dynamiques

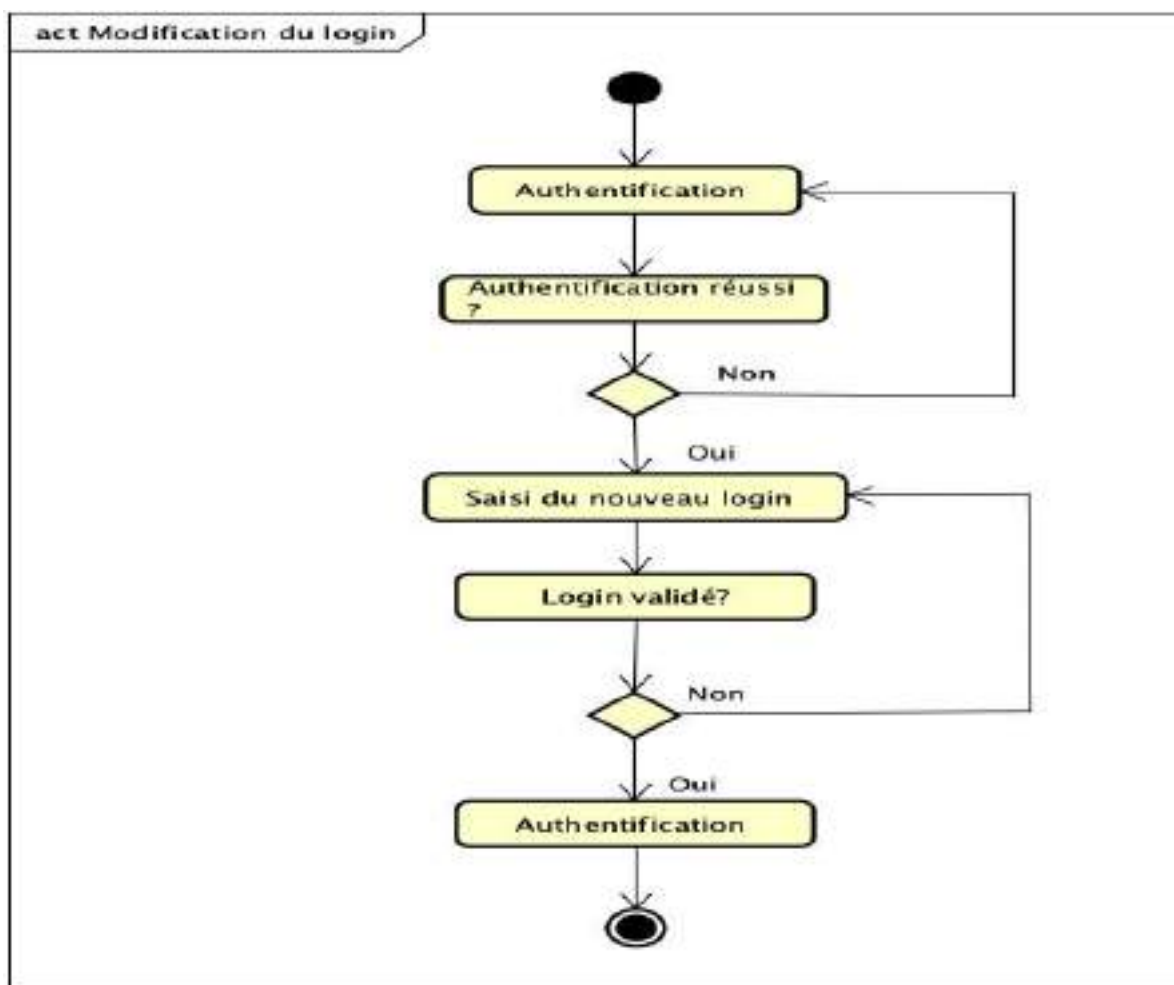
La modélisation dynamique d'un système consiste à décrire son comportement lors de sa réaction à son environnement.

##### i. Diagrammes d'activité :

Le diagramme d'activité décrit les comportements d'une opération (en termes d'actions). À travers les diagrammes d'activités, on va modéliser les différents processus, et des opérations liées à notre application



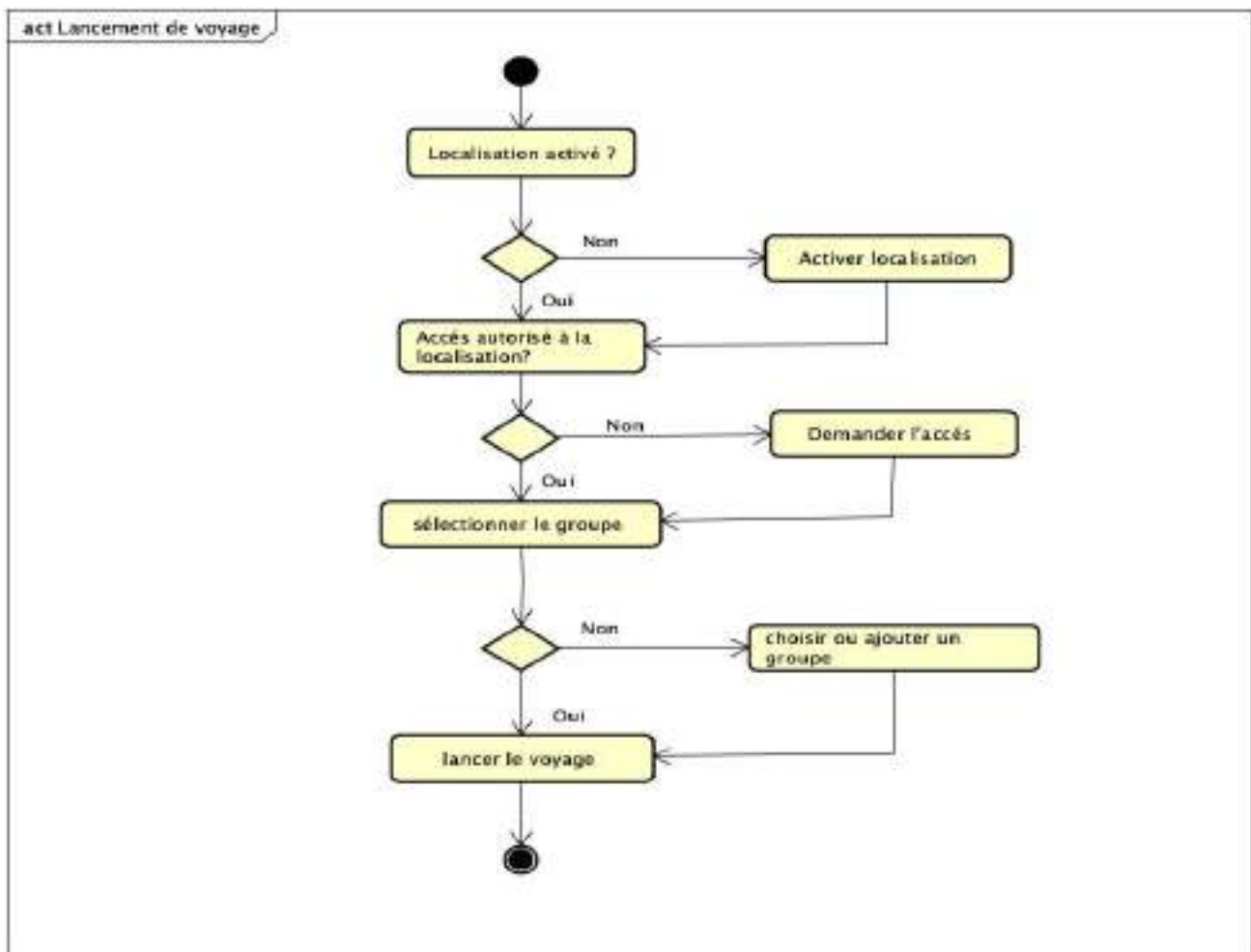
- Diagramme d'activité « Modification du login »



*Figure 4 : Diagramme d'activité : modification du login*

La modification du login de l'utilisateur est une opération très sensible, avant d'effectuer cette opération, l'utilisateur doit d'abord s'authentifier, pour être sûr que celui qui vient de modifier son login est lui-même l'utilisateur concerné. La même démarche est appliquée pour la modification du mot de passe.

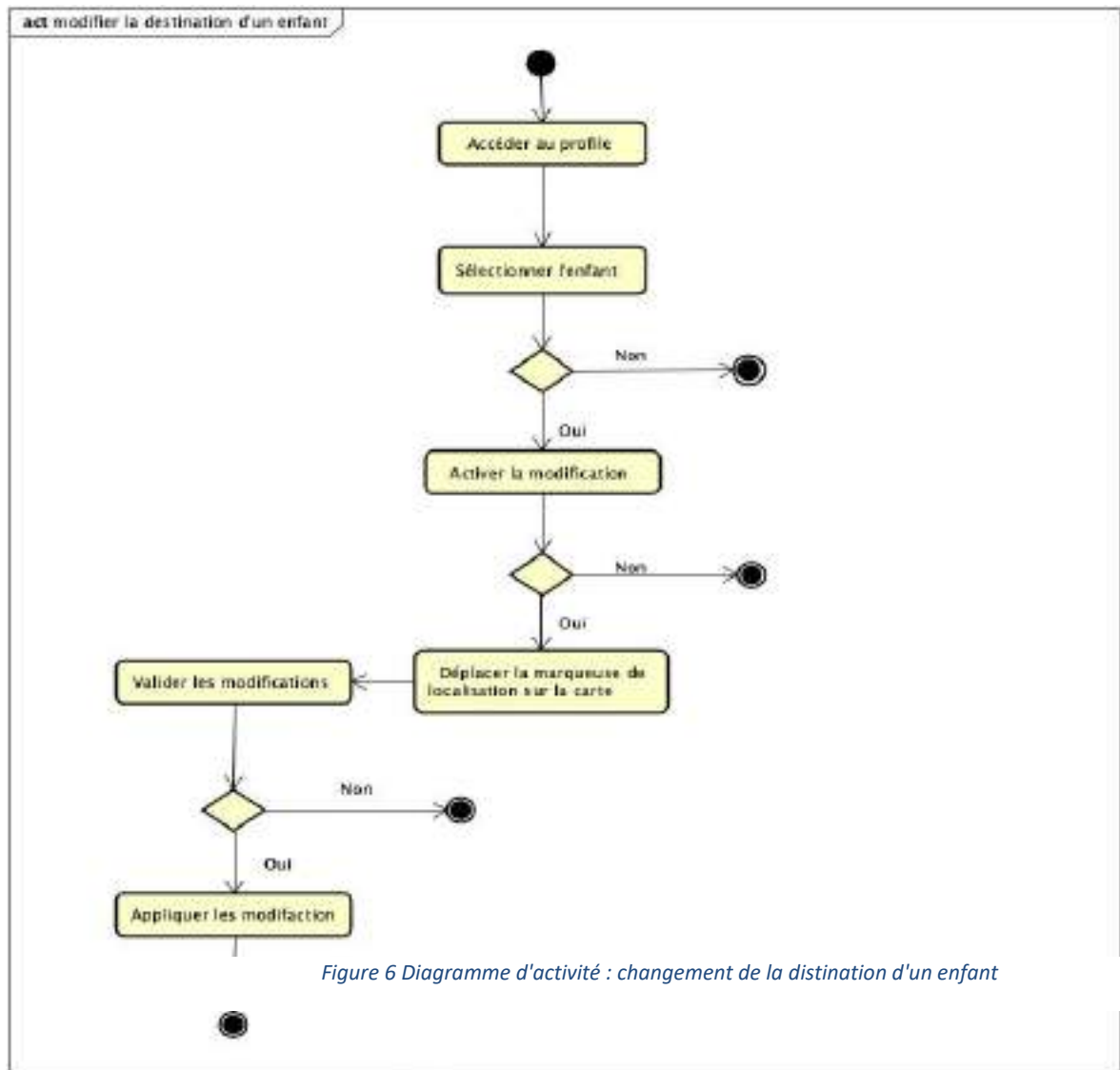
- Diagramme d'activité « lancement de voyage »



*Figure 5: Diagramme d'activité : Lancement de voyage*

Pour lancer un voyage l'application va vérifier le statut de localisation (activé ou non activé) s'elle est non activé l'application va demander au chauffeur de l'activer. D'autre part s'elle est déjà activée l'application va vérifier la permission d'accès de localisation, si l'accès est déjà autorisé le chauffeur va sélectionner le groupe afin que le voyage sera lancé, sinon l'application va demander la permission d'accéder à la localisation.

- **Diagramme d'activité « Modifier la destination d'un enfant »**



**Figure 6 : Diagramme d'activité : Changer la destination d'un enfant**

Pour modifier la destination d'un seul enfant le parent doit accéder à son profil, puis il doit sélectionner l'enfant concerné situé dans la liste des enfants, en suite il fallait activer le mode modification. Ensuite il doit déplacer la marqueuse de la localisation sur la carte et valider la modification voulue.

## ii. Diagrammes de séquences

Un diagramme de séquence décrit l'aspect dynamique du système. Il modélise les interactions entre les objets ou entre utilisateur et objet, en mettant l'accent sur la chronologie des messages échangés.

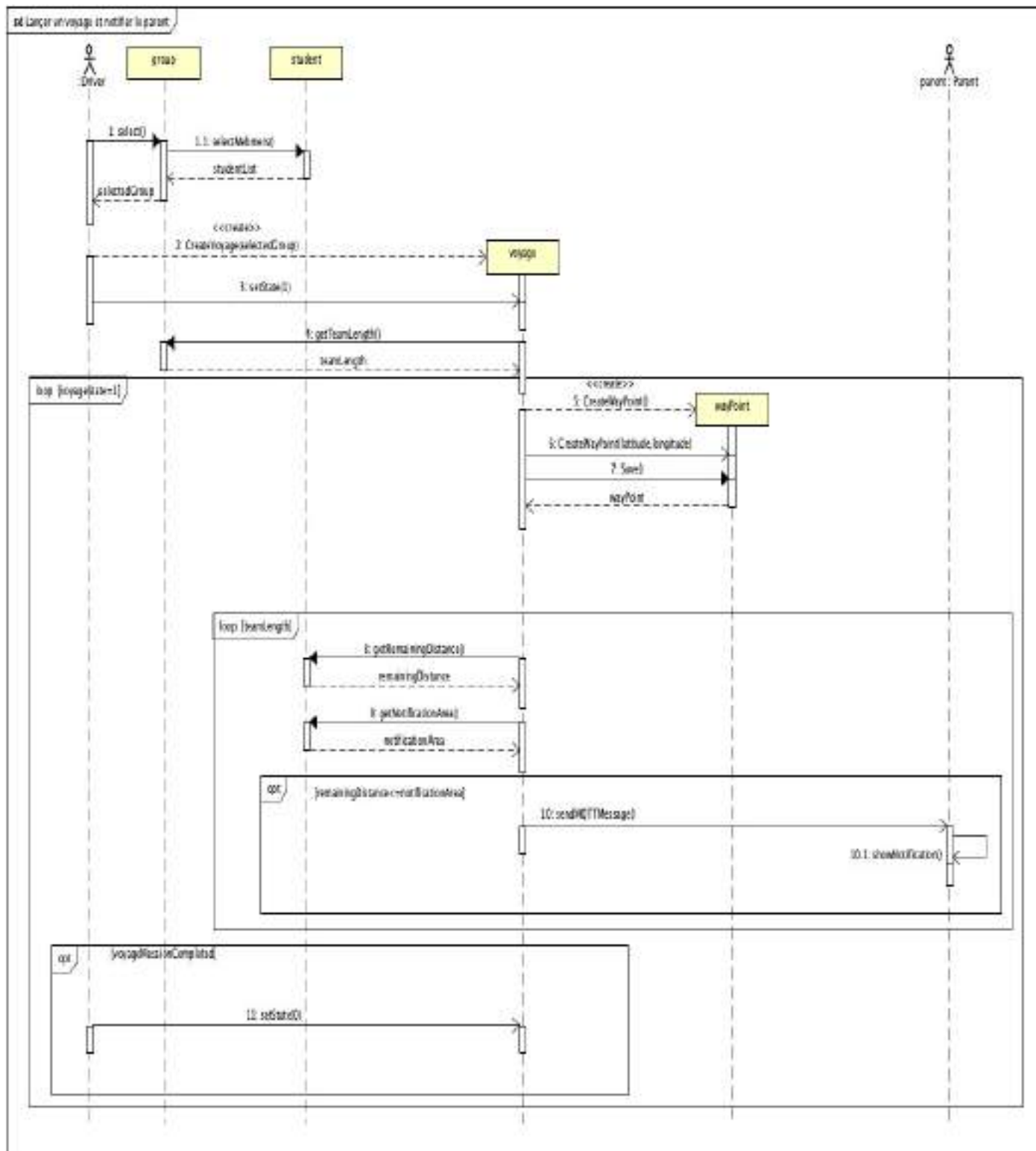


Figure 7 : Diagramme de séquence déroulement de lancement de voyage

- ### Déroulement

Pour notifier un parent à propos du voyage d'un autobus, son enfant doit être un membre du groupe concerné. Après que le chauffeur a choisi le groupe, le système va récupérer la liste des élèves qui sont membres du groupe sélectionné, puis le chauffeur peut lancer un voyage pour ce groupe, tant que le voyage n'a pas encore

terminé l'application sauvegarde automatiquement le trajet du voyage sous forme des points géographique (latitude, longitude), en parallèle l'application va calculer la distance restante pour chaque élève de groupe et la comparer avec sa distance de notification qui a déjà été définie par le parent.

Si la distance calculée est inférieure ou égale à la distance de notification le système va envoyer un message MQTT au parent afin que le parent soit notifié.

Lorsque le voyage est terminé le chauffeur arrête manuellement la sauvegarde de trajet à travers le changement de l'état de voyage.

### 3) Le modèle statique (diagramme de classe)

Le diagramme de classes représente les classes intervenant dans le système. Le diagramme de classe est une représentation statique des éléments qui composent un système et de leurs relations.

Le diagramme de classe ci- dessous illustre tous les classes et relation entre eux ainsi que la contribution de chaque classe dans ces associations.

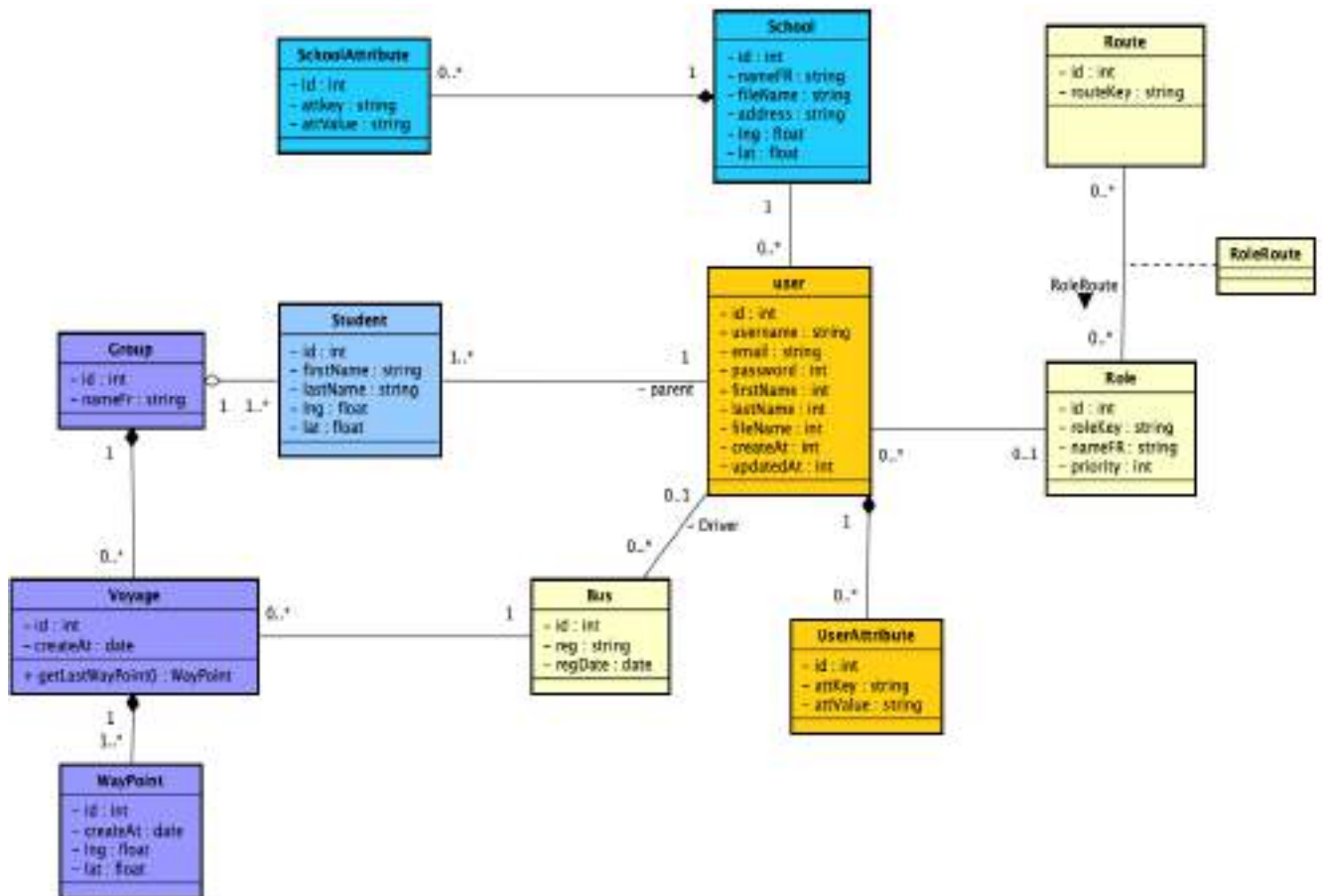


Figure 8 : Diagramme de classe

### **i. Explications textuelles de diagramme de classe**

Les classes sur lesquelles se porte notre application sont les suivantes :

User : c'est la classe la plus essentielle de notre application qui représente les différents utilisateurs.

à partir de ces règles de gestion suivantes nous avons établi notre modèle :

- Un utilisateur (User) correspond à un seul école(School).
- Un utilisateur doit avoir un rôle (Role).
- Un utilisateur peut être un parent d'un ou plusieurs élèves (Student).
- Un utilisateur peut être un chauffeur d'un ou plusieurs autobus (Bus)
- Un groupe (Group) contient plusieurs élèves.
- Un groupe peut avoir un ou plusieurs voyages (Voyage) se transporter par un autobus.
- Chaque voyage doit avoir plusieurs points géographique (Waypoint).
- Chaque route (Route) est un lien hypertexte.
- Chaque rôle doit avoir un ou plusieurs routes.

La relation entre la classe Route et la classe Rôle est dédié pour sécuriser et limiter l'accès de fonctionnalité de back office<sup>1</sup>, autrement dit chaque rôle peut accéder à des routes bien définies.

## **IV. Conclusion**

Ce chapitre a été au cœur de notre projet, nous avons présenté dans un premier lieu, tous les aspects techniques du projet, et dans un second lieu, nous avons présenté les différents diagrammes nécessaires pour décrire le processus de fonctionnement de notre application. Nous avons terminé le chapitre avec le modèle statique à travers le diagramme de classes.

### Résumé :

Dans ce chapitre nous allons présenter les outils et les technologies utilisées, puis l'ensemble des fonctionnalités offertes par notre application.

## I. Introduction

Dans ce chapitre nous allons présenter les outils que nous avons utilisés pour la mise en place de cette solution, ainsi que la démonstration de notre application à travers des captures d'écran.

## II. Outils du développement

Dans cette partie nous allons présenter les différents outils utilisés pour la mise en place de notre solution.

### 1) IDEs

#### i. VS Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS2.

Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires.

Le code source de Visual Studio Code provient du projet logiciel libre et open source VSCode de Microsoft publié sous la licence MIT permissive, mais les binaires compilés sont des logiciels gratuits pour toute utilisation.

Dans le Stack Overflow 2019 Developer Survey, Visual Studio Code a été classé comme l'outil d'environnement de développement le plus populaire, avec 50,7 % des 87 317 répondants déclarant l'utiliser.

### 2) Outils Google

#### i. Google maps



Google Maps est un service de cartographie développé par Google. Il offre des images satellites, des photographies aériennes, des cartes routières, des vues panoramiques à

360° des rues, des conditions de circulation en temps réel et la planification d'itinéraires pour se déplacer à pied, en voiture, à vélo, en avion ou en transport public.

## ii. Contrôle de version

- **Git**



Git est un système de contrôle de version distribué pour suivre les changements de code source pendant le développement logiciel. Il est conçu pour coordonner le travail des programmeurs, mais il peut être utilisé pour suivre les changements dans n'importe quel ensemble de fichiers. Ses objectifs incluent la vitesse, l'intégrité des données et la prise en charge des flux de travail distribués et non linéaires.

- **GitHub**



GitHub est une plateforme open source de gestion de versions et de collaboration destinée aux développeurs de logiciels. Elle repose sur Git, un système de gestion de code open source créé par Linus Torvalds dans le but d'accélérer le développement logiciel.

GitHub permet aux développeurs de modifier, d'adapter et d'améliorer le logiciel gratuitement à partir de référentiels publics, mais facture les interventions pour les référentiels privés, selon trois formules de paiement. Chaque référentiel public ou privé contient tous les fichiers d'un projet, ainsi que l'historique des révisions de chaque fichier. Plusieurs collaborateurs sont susceptibles de travailler dans les référentiels, qu'ils soient publics ou privés.

## 3) Environnement de développement

### i. NodeJS



Développé par Ryan Dahl en 2009 Node.js est un environnement permettant d'exécuter du Javascript côté serveur. Construit sur l'interpréteur Javascript V8 de Google (qui équipe Chrome), Node.js donne la possibilité de développer des applications web en



réseau. Il est particulièrement utilisé dans la programmation événementielle asynchrone (Websockets, Mashups,...).

Scalable, il favorise la conception souple et évolutive. Donc Node JS n'est pas un Framework, mais plutôt un outil permettant de communiquer avec le système grâce à différentes librairies.

Quelques avantages à son utilisation :

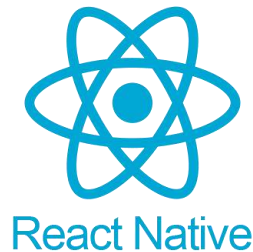
- exploitable sous licence libre
- sa communauté est active
- sa performance lié au moteur Javascript V8
- permet d'exploiter le javascript pour la création d'application web, desktop

## • **Caractéristiques de Node.js**

Node.js a connu une croissance rapide au cours des dernières années. Cela est dû à la vaste liste de fonctionnalités qu'il offre :

- a. Facile – Easy-Node.js est assez facile à prendre en main. C'est un choix incontournable pour les débutants en développement web. Grâce à de nombreux tutoriels et à une vaste communauté, il est très facile de se lancer.
- b. Évolutif – Il offre une grande évolutivité aux applications. Node.js, étant single-thread, est capable de gérer un grand nombre de connexions simultanées avec un débit élevé.
- c. Vitesse – L'exécution non bloquante des threads rend Node.js encore plus rapide et plus efficace.
- d. Paquets – Un vaste ensemble de paquets Node.js open source est disponible et peut simplifier votre travail. Aujourd'hui, il y a plus d'un million de paquets dans l'écosystème NPM.
- e. Backend solide – Node.js est écrit en C et C++, ce qui le rend rapide et ajoute des fonctionnalités comme le support réseau.
- f. Multi-plateforme – La prise en charge multi-plateforme vous permet de créer des sites web SaaS, des applications de bureau et même des applications mobiles, le tout en utilisant Node.js.
- g. Maintenable – Node.js est un choix facile pour les développeurs, car le frontend et le backend peuvent être gérés avec JavaScript comme un seul langage.

## ii. React Native



React Native est un framework d'applications mobiles open source créé par Facebook. Il est utilisé pour développer des applications pour Android, iOS et UWP en permettant aux développeurs d'utiliser React avec les fonctionnalités natives de ces plateformes.

### Historique

En 2012, Mark Zuckerberg a déclaré: "La plus grande erreur que nous ayons commise en tant que société a été de parier trop sur HTML par opposition au natif"<sup>5</sup>. Il a promis que Facebook offrirait bientôt une meilleure expérience mobile.

À l'intérieur de Facebook, Jordan Walke a trouvé un moyen de générer des éléments d'interface utilisateur pour iOS à partir d'un thread JavaScript. Ils ont décidé d'organiser un Hackathon interne pour perfectionner ce prototype afin de pouvoir créer des applications natives avec cette technologie.

Après des mois de développement, Facebook a publié la première version de la configuration JavaScript de React en 2015. Lors d'un entretien technique, Christopher Chedeau a expliqué que Facebook utilisait déjà React Native en production pour son application de groupe et son application de gestionnaire de publicités.

## iii. TypeScript



Le TypeScript est un langage de programmation développé par Microsoft en 2012. Son ambition principale est d'améliorer la productivité de développement d'applications complexes.

C'est un langage open source, développé comme un sur-ensemble de Javascript. Ce qu'il faut comprendre par là, c'est que **tout code valide en Javascript l'est également en TypeScript**.

Cependant, le langage introduit des fonctionnalités optionnelles comme le typage ou encore la programmation orientée objet. Pour pouvoir bénéficier de ces fonctionnalités aucune librairie n'est requise. Il suffit d'utiliser l'outil de compilation de TypeScript pour le transpiler (C'est le fait de compiler le code source d'un

langage en un autre langage) en Javascript. Ainsi, le code exécuté sera un équivalent Javascript du code TypeScript compilé.

#### iv. Autres outils

- **Postman**



Postman sert à exécuter des appels HTTP directement depuis une interface graphique. Vous pourrez simplement choisir l'URL, la méthode HTTP (le plus souvent GET, POST, PUT, PATCH et DELETE), les headers, les query params et dans certains cas le body de la requête.

Comment fonctionne Postman ?

Les requêtes HTTP est la fonctionnalité la plus triviale à laquelle répond cet outil, pouvoir faire des requêtes HTTP simplement. L'interface graphique rend l'exercice aussi facile que surfer internet sur un navigateur moderne. Il suffit de renseigner l'URL et la méthode HTTP.

Les méthodes HTTP diffèrent suivant l'opération souhaitée et l'API sur laquelle vous faites votre opération. Voici les méthodes HTTP les plus utilisées et leurs fonctions:

- GET: les requêtes GET sont celles effectuées par un navigateur lorsque vous entrez une URL dans la barre de navigation. Elles ont pour but d'aller chercher une page ou de la donnée.
- POST: les requêtes POST ont pour but d'envoyer de l'information, contenue dans le body de la requête, vers le serveur.
- PUT: les requêtes PUT vont écraser une ressource avec de la nouvelle donnée, là aussi présente dans le body de la requête. Elle est utilisée pour mettre à jour de la donnée à condition qu'on soit capable de fournir la ressource mise à jour dans son intégralité.
- PATCH: les requêtes PATCH servent également à mettre à jour une ressource mais en ne modifiant que l'élément envoyé en body de la requête.
- DELETE: comme son nom l'indique, la requête DELETE sert à effacer une ressource

Ces usages des requêtes HTTP varient suivant le serveur et la façon dont il a été développé, mais ces méthodes sont les plus communes pour les API REST.

- **NPM**



npm est le plus grand registre de logiciels au monde. Les développeurs open source de tous les continents utilisent npm pour partager et emprunter des packages, et de nombreuses organisations utilisent également npm pour gérer le développement privé.

npm se compose de trois composants distincts :

- le site Web
  - l'interface de ligne de commande (CLI)
  - le registre
1. Utilisez le site Web pour découvrir des packages, configurer des profils et gérer d'autres aspects de votre expérience npm. Par exemple, vous pouvez configurer des organisations pour gérer l'accès aux packages publics ou privés.
  2. La CLI s'exécute à partir d'un terminal et c'est ainsi que la plupart des développeurs interagissent avec npm.
  3. Le registre est une grande base de données publique de logiciels JavaScript et des méta-informations qui l'entourent.

- **MQTT**



MQTT est un protocole de messagerie publish-subscribe basé sur le protocole TCP/IP, il a été créé en 1999 par Andy Stanford-Clark, ingénieur chez IBM, et Arlen Nipper, chez EuroTech, principalement dans la communication M2M pour permettre à deux appareils utilisant des technologies différentes de communiquer.

**Comment fonctionne le protocole MQTT ?**

Le protocole se compose par trois éléments principaux, le serveur nommé Broker et ces client Subscriber et Publisher

Pour d'être communiquer entre eux, le client Subscriber doit être souscrit sur la même chanel dont le Publisher envoie ces messages, cette chanel nommée Topic dans le contexte du protocole MQTT.

### **III. Mise en œuvre**

#### **1) Écran de démarrage**



*Figure 9 splash screen*

Le démarrage de l'application commence par une splash screen, qui s'exécute un certain temps. En ce moment l'application vérifie l'existence des informations d'utilisateur concernant leur authentification dans le stockage interne de l'appareille. Si le cas est vérifié l'application sera redirigée vers l'écran d'accueil selon le rôle de l'utilisateur connecté, sinon l'écran de notification sera affiché.

## 2) Écran d'authentification

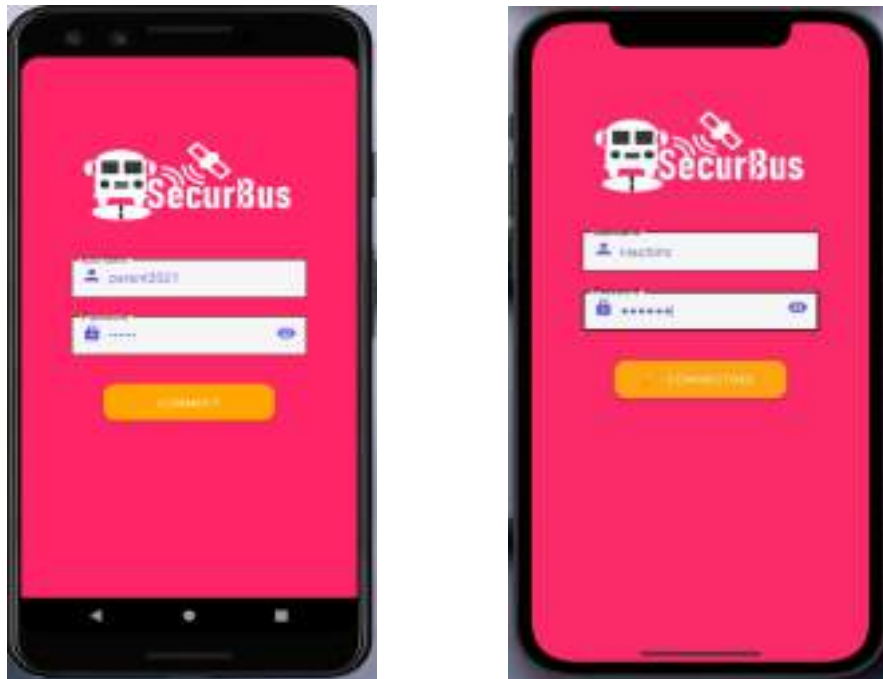


Figure 10 écran d'authentification [android/IOS]

L'utilisateur est appelé à s'authentifier en tapant son nom d'utilisation et son mot de passe, s'il est déjà enregistré et les deux sont validés alors l'authentification sera réalisée sans aucun problème sinon elle sera refusée.

Notre application contient deux espaces, un pour les chauffeurs et l'autre pour les parents.

## 3) L'espace de Chauffeur

### i. L'écran d'accueil

Une fois l'utilisateur est authentifié en tant que chauffeur, il peut accéder à la rubrique qu'il souhaite.



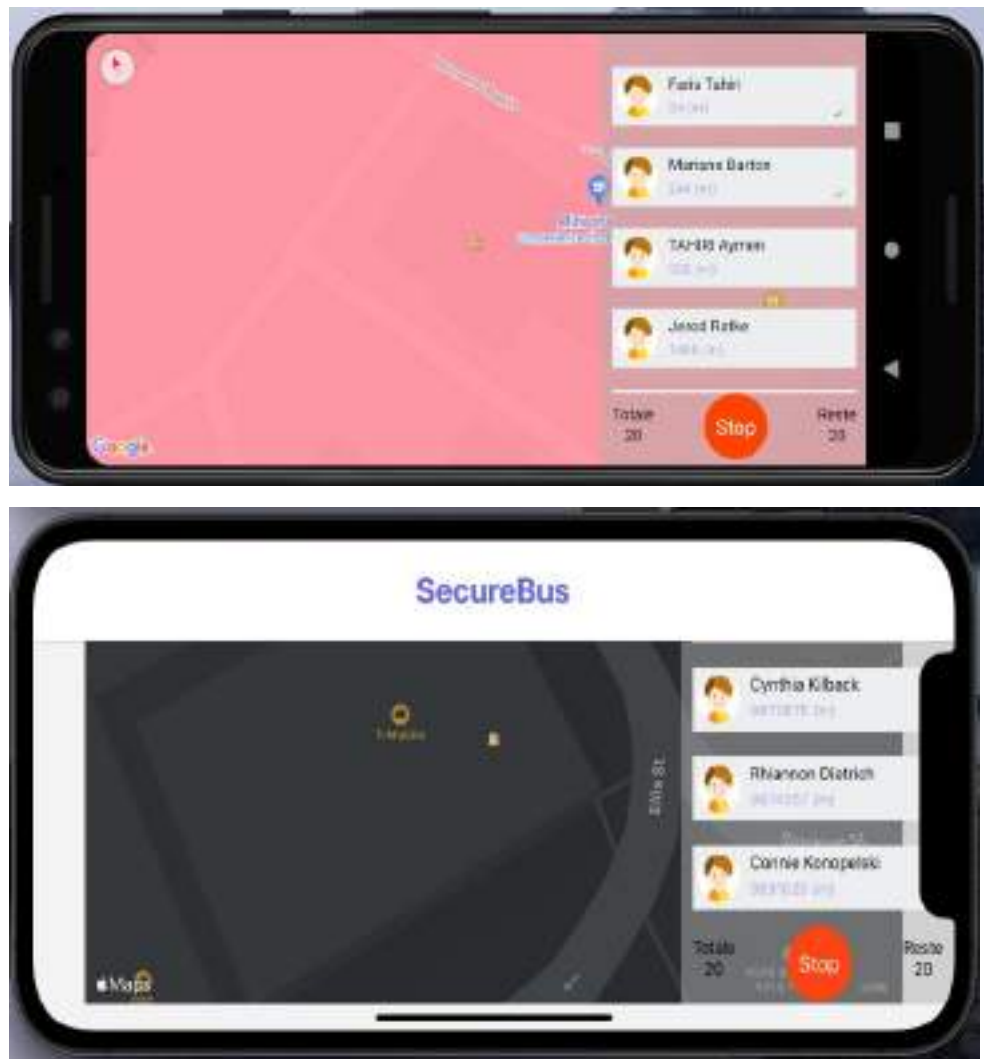
Figure 11 écran d'accueil pour le chauffeur [android/IOS]

Par la suite, le chauffeur peut lancer un voyage pour un groupe choisi par lui-même, les groupe qui ont déjà un voyage avec le chauffeur connecté seront affichés dans la zone nommé « Groupes », l'application permet aussi au chauffeur d'ajouter un nouveau groupe à son choix en cliquant sur le bouton « ajouter » situé dans la même zone des groupes.



Figure 12 les étapes de lancement de voyage [android/iOS]

Pour finaliser lancement du voyage le chauffeur doit cliquer sur le bouton afficher au-dessous de la carte géographique.



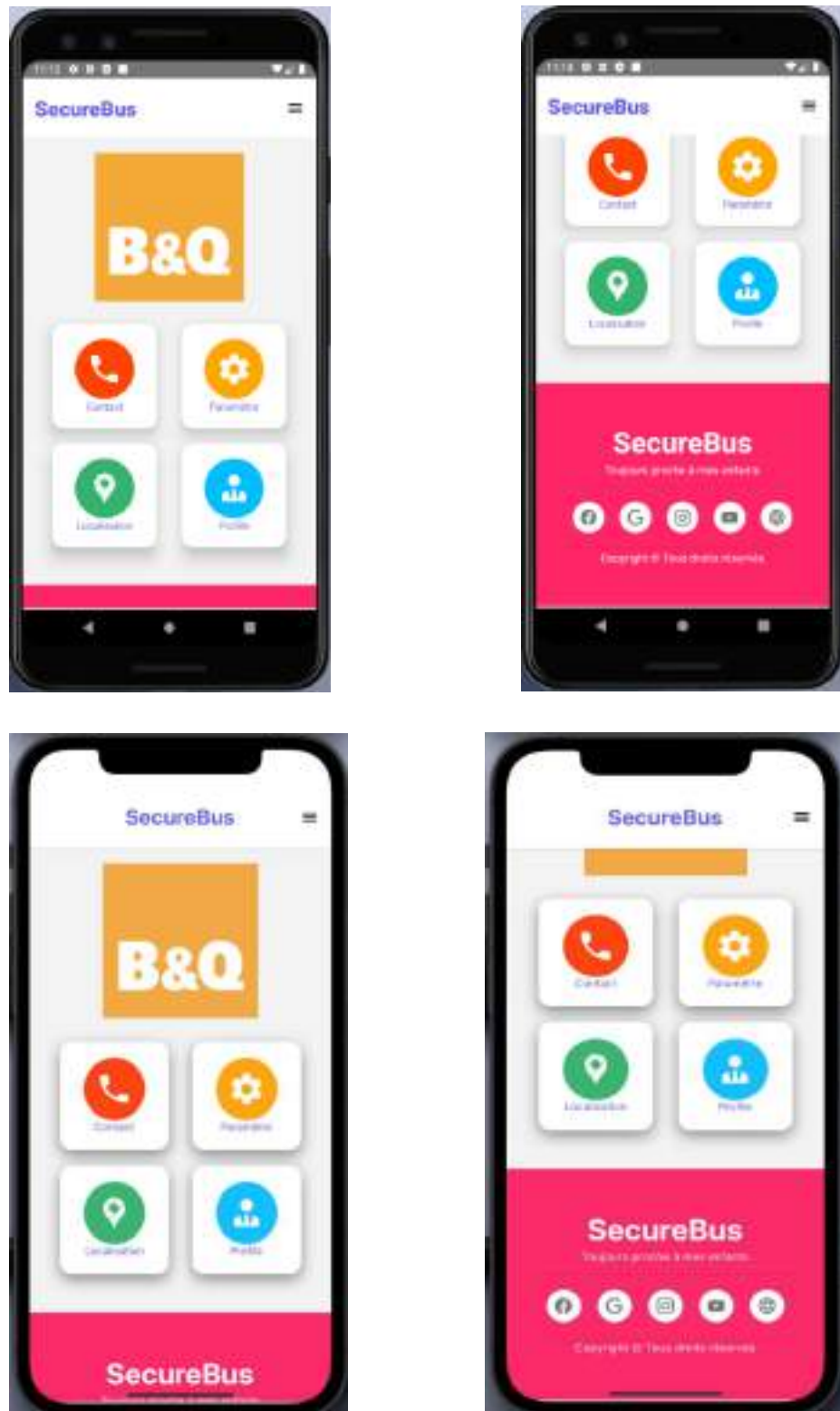
*Figure 13 écran de navigation GPS de voyage [android/IOS]*

Après avoir terminé leur voyage, il doit cliquer sur le bouton « stop » situé au-dessous des listes des élèves pour arrêter l'enregistrement de trajet.



4) L'espace de parent  
i. L'écran d'accueil

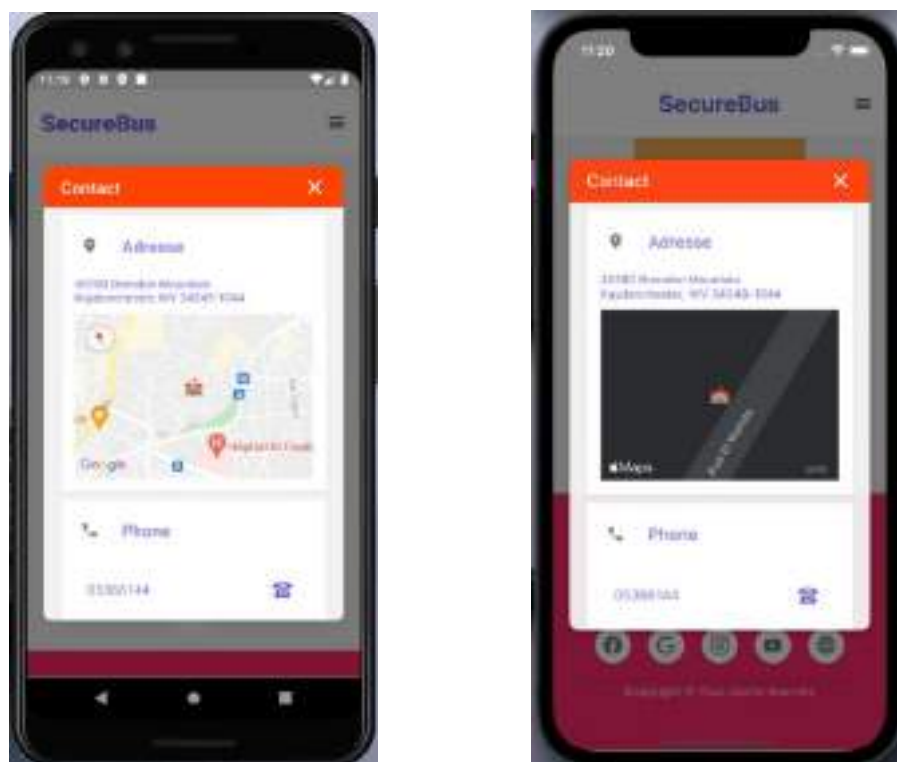
Une fois l'utilisateur est authentifié en tant que parent, il peut accéder à la rubrique qu'il souhaite.



*Figure 14 : Ecran d'accueil pour le parent [android/IOS]*

## ii. Contacter l'établissement

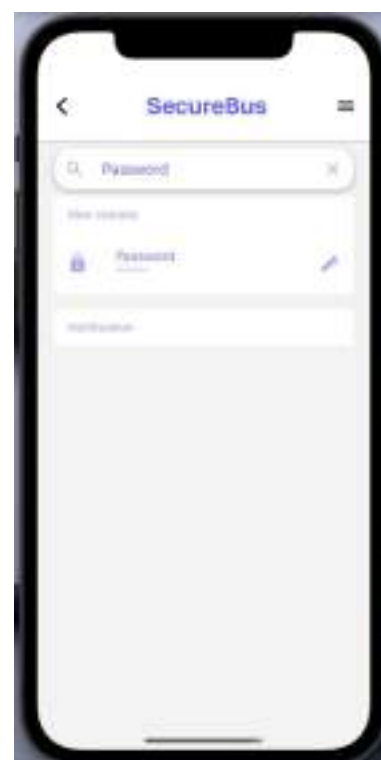
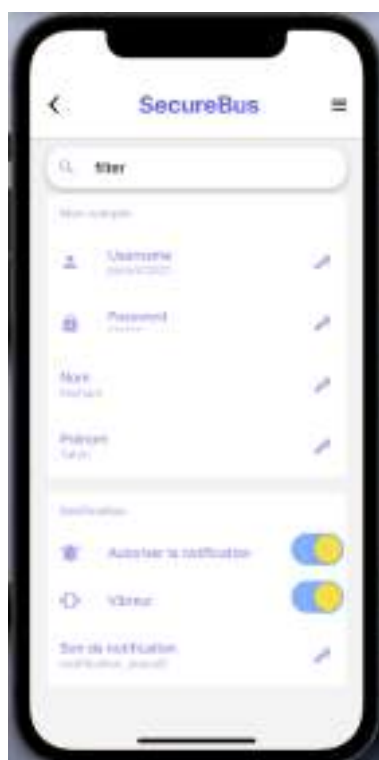
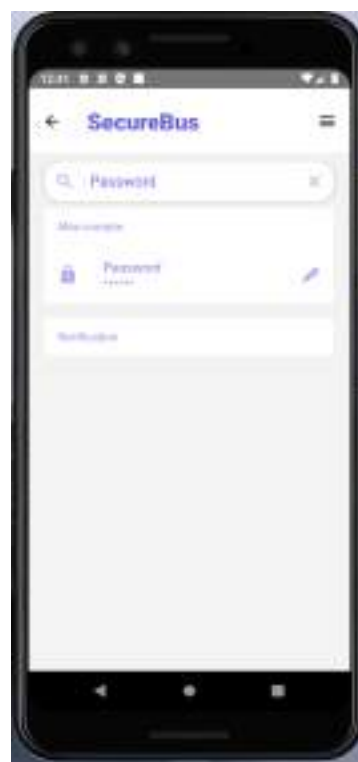
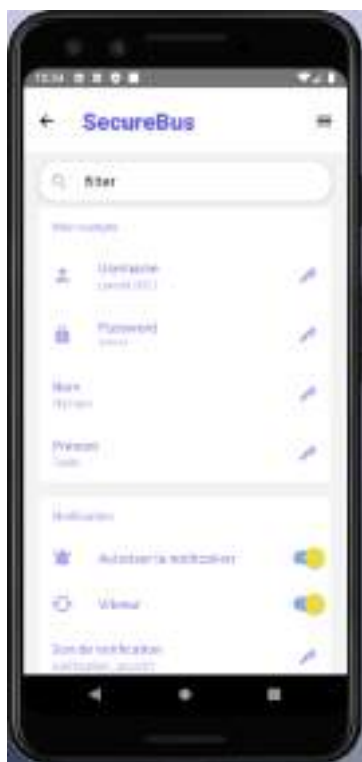
Cette fonctionnalité permet au parent de consulter plusieurs informations qui concernent l'établissement tel que sa position géographique et la liste des téléphones.



*Figure 15: Ecran contacter l'établissement*

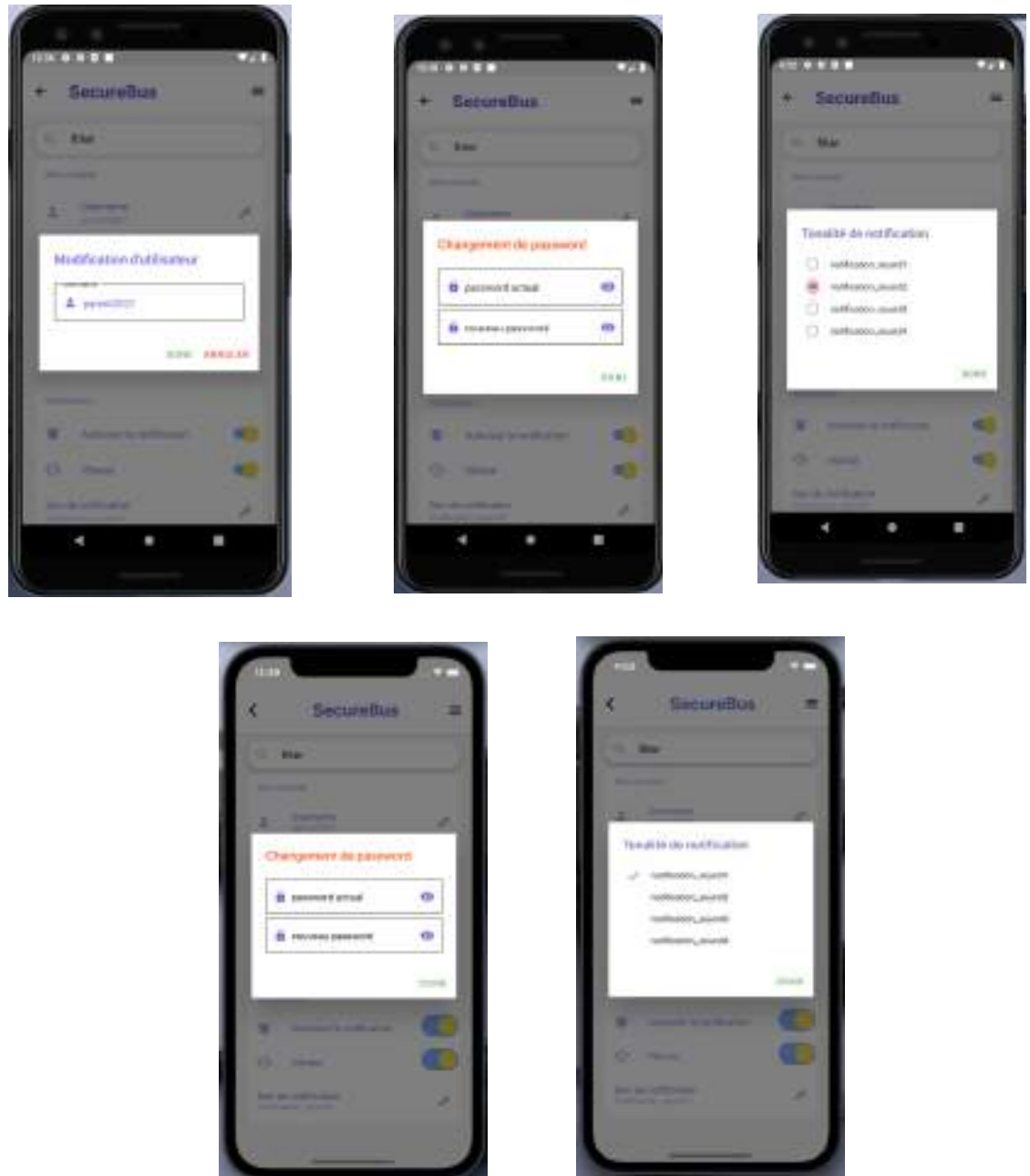
Le parent peut aussi paramétrer les notifications et modifier ses paramètres d'authentifications par un simple clic sur le bouton « paramètre ».

La modification des paramètres de compte de parent se fait par un clic sur l'icône placer à la droite de chaque attribut. Il est aussi possible de modifier les paramètres dédiés aux notifications tel que activation/désactivation de l'option « vibreur », « autorisation de notification » et le son de notification.



*Figure 16 : Ecran de paramétrage [android/IOS]*

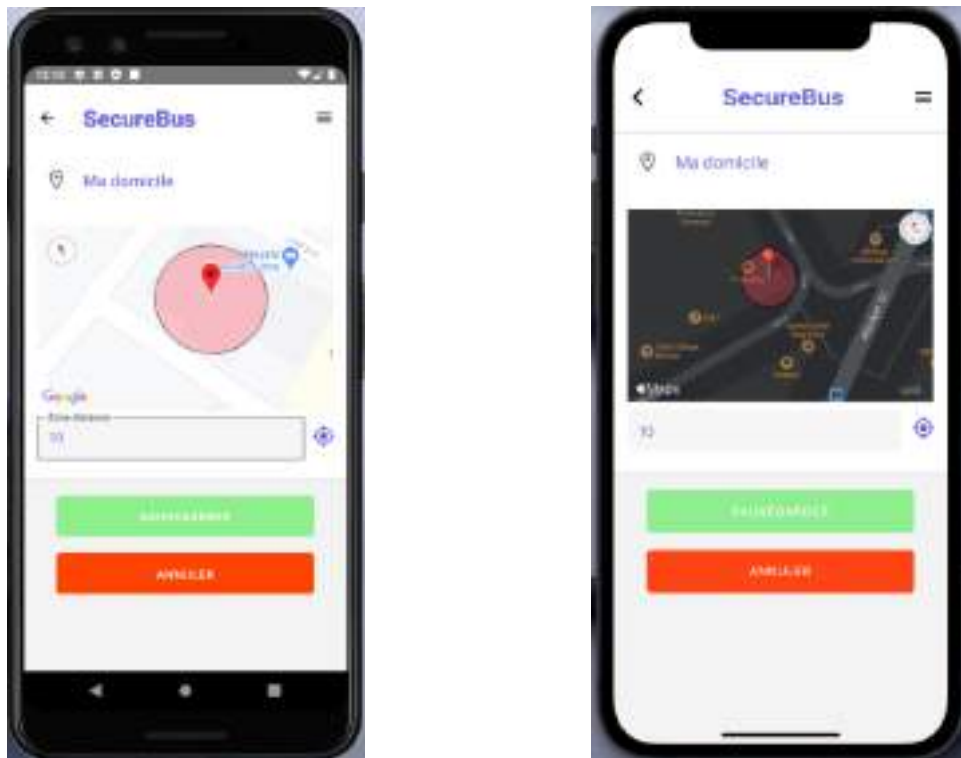
Les images ci-dessous montrent les étapes précitées



*Figure 17 : Démarche de modification, paramétrage de compte et notification [android/IOS]*

### iii. Changer la localisation

Les parents ont le droit de changer la localisation de leurs enfants en cliquant sur le bouton de « localisation » comme indiqué sur les images ci-dessous.

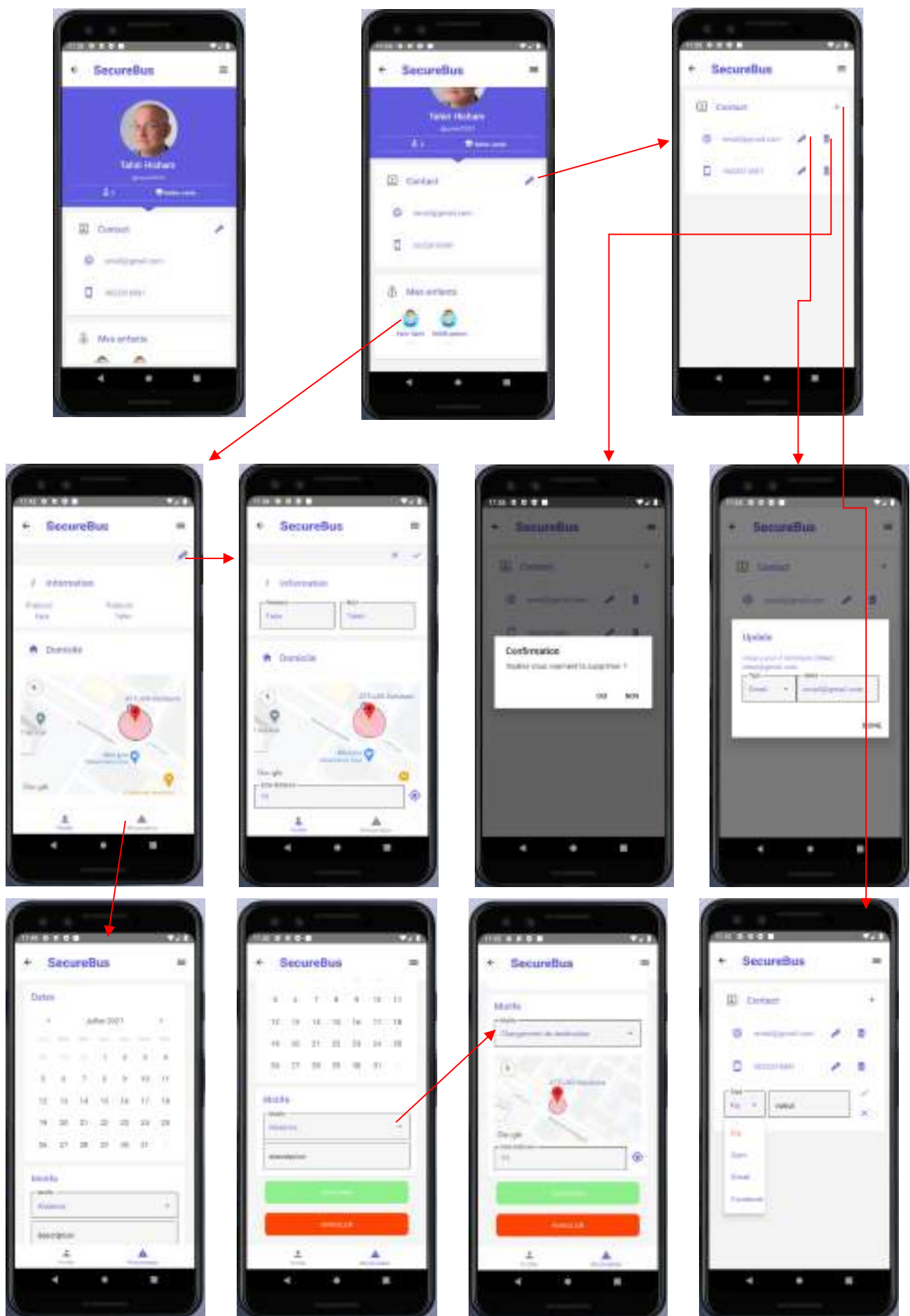


*Figure 18 : Ecran de changement de la localisation et la distance de notification*

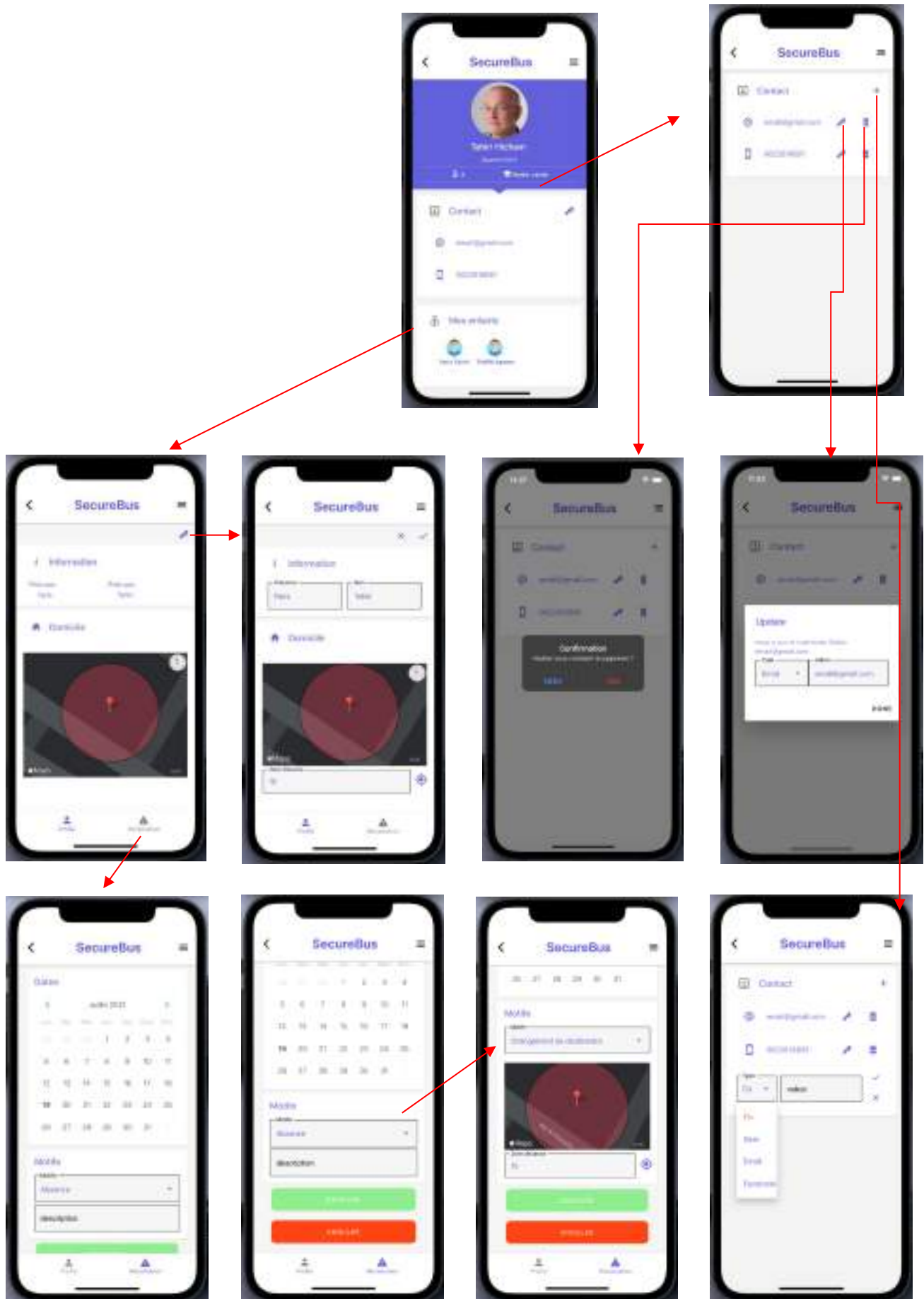
Pour effectuer une modification le parent peut déplacer la marqueuse de position et ainsi il peut modifier la distance de notification puis il fallait cliquer sur le bouton « sauvegarder ».

### iv. Consultation du profil

Dans cette partie le parent a le droit de modifier, ajouter et supprimer un attribut de contact et aussi il peut modifier la localisation et la distance de ses enfants d'une façon individuelle, il peut ainsi effectuer des réclamations concernant l'absence de ses enfants.



*Figure 19 : Scénario de manipulation du profil de parent sur android*

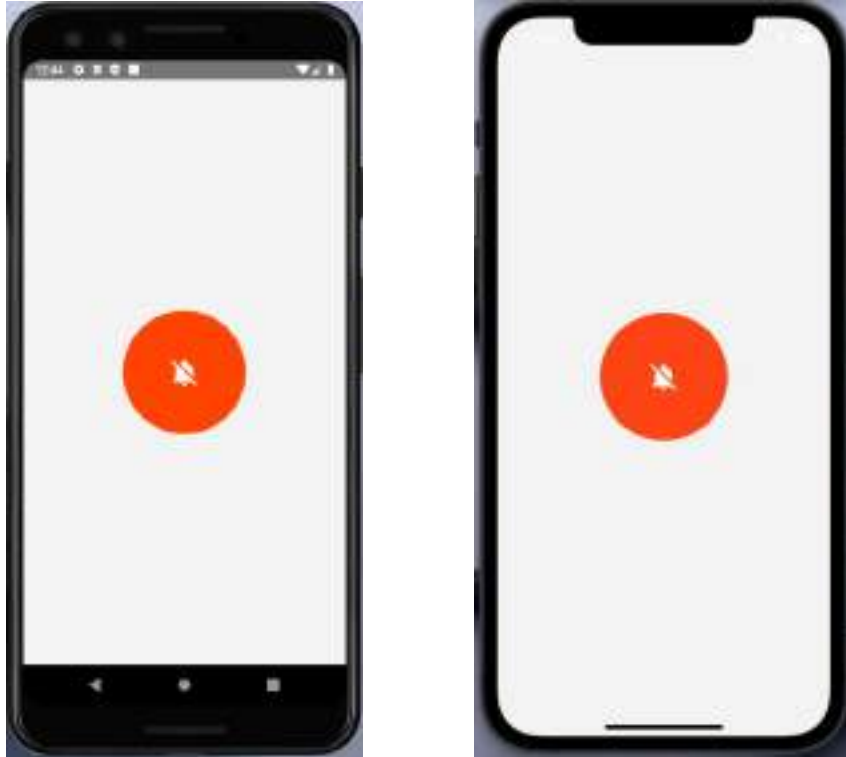


*Figure 20 : Scénario à suivre pour manipuler le profil de parent sur IOS*



#### **v. Recevoir les notifications**

Le parent peut recevoir des notifications à travers le compte de chauffeur en utilisant le protocole MQTT. Au moment où l'autobus est proche au point d'arrivée, l'application va publier un message dans le topic (chanel) où le parent est s'abonner. Lorsque le message est bien reçu, une notification sous forme d'une alarme sera lever.



*Figure 21 : Ecran de notification android/IOS*

#### **IV. Conclusion**

Ce chapitre a été consacré pour la présentation de la solution que nous avons mis en place. Ainsi nous avons commencé par la présentation des différents outils utilisés pour le développement de notre application pour terminer par l'illustration des différentes interfaces de cette application via des captures d'écran sur un smartphone.



## Conclusion générale

---

En guise de conclusion, nous pouvons affirmer que le développement multiplateforme mobile est un domaine enrichissant au niveau expérimental et qu'il est en évolution constante. Ce stage a été pour moi l'occasion de faire le lien entre mes connaissances académiques et le monde professionnel. Il m'a permis de développer mes compétences techniques, d'approfondir mes connaissances théoriques et les mettre en pratique. Cette expérience a aiguisé mes capacités d'analyse et de synthèse, et m'a permis de renforcer mes connaissances concernant le développement mobile ainsi que le développement hybride.

Après la réalisation technique du projet avec le framework React Native, je constate que toutes les fonctionnalités réalisées sont fonctionnelles sur le mobile. Ce projet m'était aussi une occasion pour mieux comprendre et apprendre des notions en relation avec le domaine de transport scolaire. Ainsi cette application va faciliter la communication et rendre la vie plus jolie aussi bien pour les parents que pour les chauffeurs.

Comme perspective, il est toujours envisageable de développer et ajouter plus des fonctionnalités sur l'application tel que le partage des informations et des événements entre l'école et les parents. Il est aussi possible de rajouter la fonctionnalité de partage de la localisation des parents d'élèves qui ne sont pas souscrits au transport scolaire en vue de notifier leur arrivée à l'école pour récupérer leurs enfants aux moments de fin cours.

Enfin, ce stage a été pour moi une expérience très enrichissante aussi bien sur le plan personnel que professionnel. En effet, il a été une occasion de découvrir l'esprit d'équipe qui règne au sein du monde d'entreprise à travers les moments inoubliables de stage passés avec l'équipe de la société FCPO.

## Webographie

- <https://www.chakray.com/advantages-of-rest-api/>
- <https://www.guru99.com/comparison-between-web-services.html>
- <https://www.ladn.eu/adn-business/experts-metiers/digital/solutions-mobile/applications/applications-mobiles-couts-minimises-solutions-hybrides/>
- <https://www.codeur.com/blog/application-mobile-hybride-ou-native/>
- <https://blog.jscrambler.com/10-frameworks-for-mobile-hybrid-apps>
- <https://reactnative.dev/docs/getting-started>

## Bibliographie

- Cours « Développement mobiles » – Prof **Abderrahim SAAID**