

République Algérienne Démocratique et Populaire  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université mohamed el bachir el ibrahimi  
Faculté math et informatique



UNIVERSITE MOHAMED EL BACHIR EL IBRAHIMI  
BORDJ BOU ARRERIDJ  
جامعة محمد البشير الإبراهيمي - بورد بوجريد

# Mémoire

PRESENTÉ EN VUE DE L'OBTENTION DU DIPLOME DE  
Master Informatique

**Spécialité : Technologie de l'Information et de la  
Communication**

---

## THÈME :

Système gestion de distribution(DMS)  
Cas : Agroalimentaire

**Préparé par : BOUCHIDA Mohamed**

Soutenu le :

**Devant le jury :**

Président Mr.Moussaoi Ali

Examineur Mm Laifa Meriem

Examineur Mr Maach Salh

Encadreur Mr Maza Sofiane

MCB a L'U.El Bachir El Ibrahimi-bba.

MCB a L'U.El Bachir El Ibrahimi-bba.

MAA a L'U.El Bachir El Ibrahimi-bba.

MCB a L'U.El Bachir El Ibrahimi-bba.

**Année Universitaire 2019-2020**

# Remerciements

*Nous tenons tout d'abord à remercier ALLAH le tout puissant, qui nous a aidées et nous a donné la patience et le courage durant ces longues années d'étude, et qui nous a donné la force et la patience d'accomplir ce modeste travail.*

*En second lieu, nous tenons à exprimer toute nos reconnaissances à notre encadreur **Dr. Maza Sofiane**, nous la remercions de nous avoir encadrées, orientées, aidées et conseillées durant toute la période du travail.*

*Nos vifs remerciements vont également aux membres du jury monsieur **Moussaoui** de nous avoir présider et **Mr Maach et Mm Aïfa** pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.*

*On adresse nos sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé nos réflexions et ont accepté à nous rencontrer et répondre à nos questions durant nos recherches*

*Enfin, Nous souhaitant adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont participé de près ou de loin à la réalisation de ce travail.*

*Merci à tous et à toutes.*

# *Dédicaces*

*Je dédie ce modeste travail*

*A mon chère père et ma chère mère, qui m'ont encouragé en témoignage de ma gratitude, de ma reconnaissance, pour tous ses sacrifices ainsi que pour toute la confiance qu'ils m'ont donnée.*

*A mes chère sœurs **ikram, dallal** leur maris **samir et walid** et leur enfants et mon cher frère **NABIL, Hadjer***

*Et mes meilleurs amis **redha, abderzake, Mostapha, rafik, zakaria, momen, adem, amar, adel** et tous mes camarades de la promotion  
En leur souhaitant pleins de succès dans leurs vies.*

*A toute ma famille et à toutes les personnes que j'aime*

*Que Dieu vous garde !*

# SOMMAIRE

---

## SOMMAIRE

### LISTE D'ABREVIATION

### LISTE DES FIGURES

### LISTE DES TABLEAUX

<b>INTRODUCTION GENERALE</b>	<b>1</b>
<b>CHAPITRE I</b>	
I.1.Introduction	3
I.2.Système GPS	3
I.2.1. Le principe de géolocalisation	4
I.2.2. Principes de fonctionnement du GPS	4
I.3. Smartphone	5
I.4.Présentation d'Android	5
I.5.La distribution agroalimentaire	5
I.6. Description de la system	6
I.6.1.Les étapes de fonctionnement	7
I.6.2.Les avantages d'un système DMS	7
I.7.Besoins fonctionnels	7
Suivi de véhicule en temps réel	7
Historique du parcours	7
Gestion des données	7
QR code	8
Gestion des comptes utilisateurs et des privilèges	8
Génération de rapports statistiques	8
I.8. Besoins non fonctionnels	8
• Contrainte ergonomique	8
• Contrainte sur la fiabilité de l'application	8
• Contrainte d'évolution	9
I.9. Description générale du fonctionnement du système	9
I.10. Identification des acteurs	9
I.10.1. L'administrateur du système	9
I.10.2. L'utilisateur du système	9
I.11. Conclusion	10

# SOMMAIRE

---

## CHAPITRE II

II.1. Introduction	11
II.2. Le langage UML	11
II.3. Diagrammes UML	11
II.3.1. Diagrammes structurels ou statiques	12
II.3.2. Diagrammes comportementaux	12
II.3.3 Diagrammes d'interaction ou dynamiques	13
II.4. Logiciels de modélisation UML	14
II.5. Diagramme de cas d'utilisation	14
II.6. Diagramme de cas d'utilisation cote d'administrateur	15
II.6.1. Tableau résumé le diagramme de cas d'utilisation pour l'administrateur	16
II.7. Diagramme de séquence	17
II.8. Réalisation des diagrammes de séquences	18
II.9. Diagramme de séquence pour l'authentification	18
II.10. Diagramme de séquence pour la géolocalisation recherche et consulte les clients sur map	20
II.11. Diagramme de séquence pour consulter l'historique de localisation	21
II.12. La gestion des clients	22
II.13. Gestion des vans	22
II.14. Diagramme de séquence cote utilisateur application Android	23
II.15. Diagramme de séquence pour l'authentification	24
II.16. Diagramme de séquence de synchronisation d'une liste de client	25
II.17. Diagramme de séquence de géolocalisation	25
II.18. Diagramme de séquence de l'historique	26
II.19. Conclusion	26

## CHAPITRE III

III.1. Introduction	27
III.2. Langages de programmation	27
III.2.1. Langage HTML5	27
III.2.2. Langage CSS	27
III.2.3. Langage java	28
III.2.4. Langage java script	28

## SOMMAIRE

---

III.2.5. JQUERY	28
III.2.6. Langage PHP	28
III.2.7. Langage SQL	28
III.2.8. Langage XML	29
III.3. Environnements de développement	29
III.3.1. App serveur	29
III.3.2. Apache	29
III.3.3. Php My Admin	29
III.3.4. MySQL	30
III.3.5. Android Studio	30
III.3.6. SDK	31
III.3.7. Le SDK Android	31
III. 3.8. Les fonctions du SDK	32
III.3.9. Android virtual device (AVD)	32
III.3.10 Service web	33
III.3.11. Communication avec la base de données	33
III.3.12. Google Maps	34
III.4. Schéma de fonctionnement de l'application	35
III.5. Présentation de quelques interfaces	36
III.5.1. L'authentification	36
III.5.2. La page d'accueil	37
III.5.3. La géolocalisation des clients	38
III.5.4. Liste des clients	38
III.5.5. Liste des vans (véhicules)	39
III.5.6. L'historique des visites	39
III.5.7. Rapport	40
III.5.8. Paramètre de compte	40
III.6. Partie application	41
III.6.1 Présentation de quelques interfaces Android	41
III.6.2. L'authentification	41
III.6.3 Menu	42
III.6.4. La liste des clients	43
III.6.5. Itinéraire	44
III.6.6. Visite	45

## SOMMAIRE

---

III.6.7 Validation ou confirmation via QR code	45
III.7 Conclusion	46
<b>Conclusion général</b>	47

## Liste D'abréviation

---

**DMS** Distribution Management System  
**GPS** Global Positioning System  
**PHP** Personal Home Page  
**UML** Unified Modeling Language  
**JSON** JavaScript Object Notation  
**XML** Extensible Markup Language  
**GSM** Global System for Mobile  
**GPRS** General Packet Radio Service  
**UMTS** Universal Mobile Telecommunications System  
**Wi-Fi** Wireless Fidelity  
**3g/4g** 3<sup>rd</sup> Generation/4<sup>th</sup> Generation  
**SMS** Short Message Service  
**SGBD** Système Gestion de Base de Données  
**HTTP** HyperText Transfer Protocol  
**UP** Unified Process  
**2TUP** 2Track Unified Process  
**JPG** Joint Photographic Group  
**PNG** Portable Network Graphics  
**CSS** Cascading Style Sheets  
**HTML** Hypertext Markup Language  
**API** Application Programming Interfaces  
**SQL** Structured Query Language  
**SDK** Software Development Kit

## Liste Des Figures

---

### Chapitre I

<b>Figure I.1</b> : Principe de triangulation	5
<b>Figure I.1</b> : Principe de fonctionnement de notre système	6

### Chapitre II

<b>Figure II.1</b> : Les 14 diagrammes d'UML	13
<b>Figure II.2</b> : Diagramme de cas d'utilisation d'administrateur	15
<b>Figure II.3</b> : Diagramme de séquence pour l'authentification	19
<b>Figure II.4</b> : Diagramme de séquence pour la géolocalisation	20
<b>Figure II.5</b> : Diagramme de séquence pour l'historique de localisation	21
<b>Figure II.6</b> : Diagramme de cas d'utilisation de La gestion des clients	22
<b>Figure II.7</b> : Diagramme de cas d'utilisation de La gestion des vans	22
<b>Figure II.8</b> : Diagramme de séquence pour l'utilisation d'application Android	23
<b>Figure II.9</b> : Diagramme de séquence pour l'authentification	24
<b>Figure II.10</b> : Diagramme de séquence de synchronisation d'une liste de client	25
<b>Figure II.11</b> : Diagramme de séquence de géolocalisation	25
<b>Figure II.12</b> : Diagramme de séquence de l'historique	26

### CHAPITRE III

<b>Figure III.1</b> : Php My Admin	29
<b>Figure III.2</b> : SDK	31
<b>Figure III.3</b> : Android virtual device (AVD)	32
<b>Figure III.4</b> : Architecture	33
<b>Figure III.5</b> : Schéma de fonctionnement de l'application	35
<b>Figure III.6</b> : Présentation des interfaces « L'authentification »	36
<b>Figure III.7</b> : La page d'accueil	37
<b>Figure III.8</b> : La géolocalisation des clients	38
<b>Figure III.9</b> : Liste des clients	38
<b>Figure III.10</b> : Liste des vans (véhicules)	39
<b>Figure III.11</b> : L'historique des visites	39
<b>Figure III.12</b> : Rapport	40

## Liste Des Figures

---

<b>Figure III.13 :</b> Paramètre de compte	40
<b>Figure III.14 :</b> Présentation de quelques interfaces Android	41
<b>Figure III.15 :</b> L'authentification	42
<b>Figure III.16 :</b> Menu	42
<b>Figure III.17 :</b> La liste des clients	43
<b>Figure III.18 :</b> Itinéraire	44
<b>Figure III.19 :</b> Visite	45

## Liste Des Tableau

---

**Tableau II.1** : Résume le diagramme de CU pour l'administrateur

16

# **Introduction Générale**

### INTRODUCTION GENERALE :

Les entreprises commerciales ont recours aux technologies de l'information et de la communication les plus récentes pour gagner de la valeur ajoutée et offrir le meilleur service Client possible. Elles adoptent de plus en plus des solutions d'automatisation de la vente et de livraison sur terrain. En effet, le contexte actuel favorise le développement de ce type de solutions (multiplication des terminaux mobiles sophistiqués et autonomes, et les technologies d'accès avec plus de bande passante, etc.)

Notre projet consiste à définir une solution informatique qui automatise la livraison en terrain et l'optimise. Pour cela on essaye de proposer des solutions traitant la problématique qui consiste à optimiser et sécuriser les processus de transport, de livraison et de ventes en intégrant des systèmes innovants.

Nous pouvons trouver des solutions de la gestion de distribution dans les ERPs, mais encore faut-il en avoir un. D'autant plus que leur coût est trop élevé. En outre l'activité terrain n'est pas directement visible à l'ERP1 sauf s'il y a une extension de celui-ci .

Les problèmes recensés dans le domaine de la vente terrain sont nombreux, nous citons :

- Retard dans les livraisons.
- Difficultés de communication entre l'entreprise et les agents terrain.
- Double saisie de l'information.
- Difficultés à trouver de l'information en temps réel (catalogues produits, fiches clients).
- Mauvaise distribution des agents sur le terrain.
- Etc.

Pour cela nous proposons un système de planification de tournées (sortie des agents sur le terrain) qui permettra d'automatiser les livraisons.

Nous avons développé un environnement de gestion des livraisons itinérantes qui permettra la gestion automatique, entre autres, des livraisons, des clients et de stock.

Ce système qui mêle traçabilité et mobilité utilisant les dernières technologies

## INTRODUCTION GENERALE

---

logicielles et matérielles est composé de deux modules :

Une solution web (back office): pour la gestion des livraisons, des agents, des commandes des clients, des produits, des prix et la supervision des tournées et ventes.

Une solution mobile (Front office) : C'est le développement d'une application mobile pour la réalisation des ventes et livraison sur le terrain. Aussi, il faut assurer la synchronisation entre la partie FrontOffice et la partie BackOffice pour l'échange de données.

Ce rapport s'articule autour de trois chapitres.

Chapitre 01 : « GÉNÉRALITÉS » : une étude de l'environnement dans lequel s'inscrit notre projet. Présentation et spécification des besoins tout en essayant de recenser tous les acteurs et la nature des interactions avec le système et de préciser les besoins fonctionnels et non fonctionnels.

Chapitre 02 : « SPECIFICATION DES BESOINS » tout au long du deuxième chapitre, nous détaillons la conception de l'application.

Chapitre 03 :« REALISATION » Ce troisième chapitre décrit l'environnement de développement et les interfaces de notre système. Ce rapport sera clôturé par une conclusion qui récapitule les apports de notre projet et les perspectives à entreprendre

Nous avons continué et amélioré notre ancien travail de licence en ajoutant des nouvelles fonctionnalités utile pour faciliter, accéléré et fiabilisé le travail de distribution.

# **CHAPITRE I**

## **Généralités**

## **I.1.Introduction**

La géolocalisation désigne de façon très large un ensemble de techniques employées pour localiser, sur un plan ou une carte, un objet ou un individu en fonction de ses coordonnées géographiques, Il existe de nombreuses techniques de géolocalisation, parmi lesquelles celles par satellite, par Wi-Fi, par GSM, par adresse IP, etc.

Aujourd'hui, la géolocalisation est utilisée dans de nombreux secteurs d'activités, autant que dans le domaine de la vie privée.

On la retrouve en particulier dans le transport et la protection de marchandises (on sait à un moment précis à quel endroit se trouve une marchandise), la protection, la sécurité et le suivi de personnes (skieurs hors-piste, personnes dépendantes, personnalités, etc.), la sécurité des biens (voiture, camion, avion par exemple)...

Une application mobile pour chauffeur livreur est un logiciel que l'on intègre au terminal mobile de communication qu'ils utilisent lors de leurs tournées. Il s'agit généralement à la fois d'un outil de gestion de tournée et de suivi que l'on installe sur le Smartphone ou la tablette tactile qui remplacent leurs fiches de tournée et de contrôle.

## **I.2.Système GPS**

Le Système de positionnement mondial (GPS, Global Positioning System) est un utilitaire qui appartient aux Etats-Unis et qui assure des services de positionnement, de navigation et de référence temporelle, dits « services PNT » (positioning, navigation, and timing). Il se compose de trois segments : le segment spatial, le segment de contrôle et le segment utilisateur. L'Armée de l'Air des Etats-Unis assure le développement, l'entretien et le fonctionnement du segment spatial et du segment de contrôle.

### **I.2.1. Le principe de géolocalisation**

La géolocalisation est un procédé permettant de positionner un objet ou une personne sur un plan ou une carte à l'aide de ses coordonnées géographiques. Cette opération est réalisée à l'aide d'un terminal capable d'être localisé (grâce à un récepteur GPS ou à d'autres techniques) et de publier (en temps réel ou de façon différée) ses coordonnées géographiques (latitude/Longitude). Les positions enregistrées peuvent être stockées au sein du terminal et être extraites postérieurement, ou être transmises en temps réel vers une plateforme logicielle de géolocalisation. Ceci permet de visualiser la position du terminal au sein d'une carte à travers une plateforme de géolocalisation le plus souvent accessible depuis internet.

Ce système de satellites est conçu de façon à ce qu'il y en ait toujours au moins quatre visibles par le navigateur GPS, sans quoi la position ne peut pas être déterminée. Pour qu'un terminal soit capable de se géolocaliser grâce au réseau GPS, il doit absolument être équipé d'une puce électronique GPS. (1)

### **I.2.2. Principes de fonctionnement du GPS**

Le principe de fonctionnement du GPS repose sur la mesure de la distance d'un récepteur par rapport à plusieurs satellites (les satellites sont répartis de telle manière que 4 à 8 d'entre eux soient toujours visibles). Chaque satellite émet un signal, capté sur Terre par le récepteur, permettant ainsi de mesurer très précisément la distance séparant l'émetteur du récepteur grâce au temps de parcours.

Avec la réception des signaux de quatre satellites (trois pour obtenir le point d'intersection des trois sphères, un quatrième pour la synchronisation du temps), le récepteur mobile est capable de calculer sa position géographique par triangulation.

Il faut au moins 3 satellites pour déterminer la position sur la terre

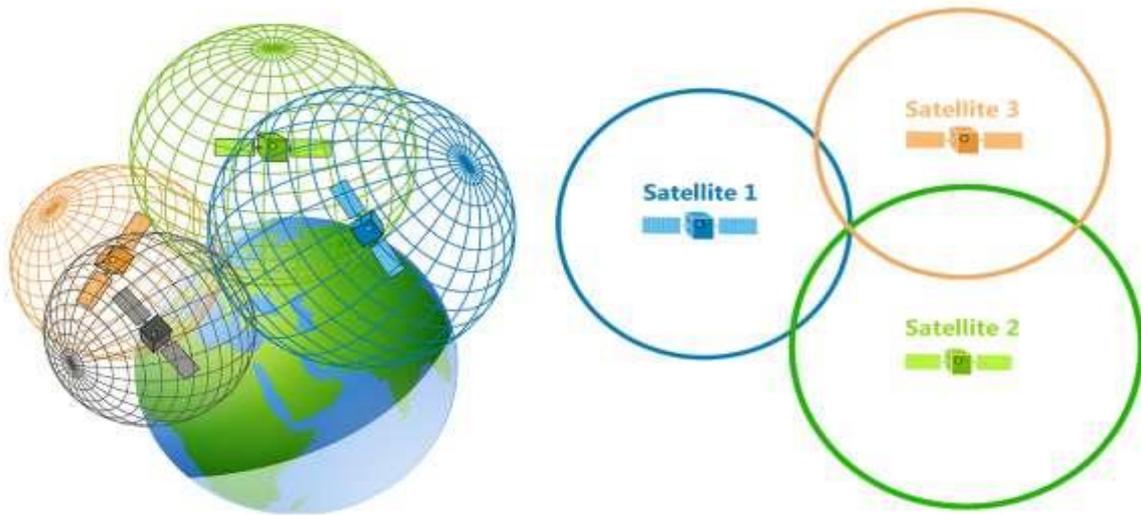


Figure I.1 : Principe de triangulation [3]

### I.3.Smartphone

Un smartphone (téléphone intelligent) est un téléphone portable doté de fonctionnalités très avancées. Un smartphone typique dispose d'un écran tactile haute résolution, de la connectivité WIFI, des capacités de navigation Web et de la capacité d'accepter des applications sophistiquées. La plupart de ces appareils fonctionnent sur l'un de ces systèmes d'exploitation mobiles populaire : Android, IOS, BlackBerry OS et Windows.

### I.4.Présentation d'Android

Il existe plusieurs types de système d'exploitation pour le téléphone mobile, chaque système à ses avantages et ses limites. Il est donc très difficile de choisir la plateforme répondant bien à l'objectif d'une société ou d'une personne.

Dans cette partie nous allons présenter la raison du choix d'Android, nous aurons aussi à présenter ce système et les outils pour créer une application dans la pratique.

### I.5.La distribution agroalimentaire :

La distribution agroalimentaire est d'assurer la ventes la livraison de certaines gamme de produits agroalimentaires sous un territoire bien précis a une base de clients délimité géographiquement avec un programme de visite hebdomadaire.

## I.6. Description de system

Pour atteindre les objectifs et pallier aux insuffisances citées précédemment, l'idée majeure est de concevoir et développer une solution nommée «Distribution Management system». Le système doit offrir le maximum de fonctionnalités nécessaires pour la gestion de flotte de véhicules avec un temps de réponse optimal et avec certitude des informations fournies.

De ce fait, nous proposons la conception et le développement d'un site web et d'une application de géolocalisation qui permet le contrôle de toutes les ressources de la société (véhicules, chauffeurs, équipements, etc.), le suivie de véhicule en temps réel, la consultation de l'historique des trajets parcouru.

DMS est un système de Planification de gestion logistique, qui fait partie des solutions que nous destinons aux entreprises de distribution. Elle vous permettra d'augmenter la productivité de ces derniers en réduisant le temps de livraison des commandes.

DMS est une solution facile à utiliser et performante pour simplifier la gestion des activités de vos livreurs. L'application mobile pour chauffeur livreur offre la possibilité d'élaborer des plannings optimisés et d'effectuer un suivi en temps réels des tournées des chauffeurs et de l'état de livraison des commandes, Grâce à son interface site web soignée et à la performance de ses fonctionnalités.



Figure I.2 : Principe de fonctionnement de notre système (26)

### **I.6.1. Les étapes de fonctionnement**

- le smart phone installé dans le véhicule détermine sa position grâce à GPS.
- La position est envoyée par connexion 3G/4G.
- La position est transférée à un serveur connecté au réseau internet.
- Vous visualisez la position du véhicule à partir de n'importe quel ordinateur disposant d'une connexion internet.

### **I.6.2. Les avantages du système DMS**

En tant qu'outil de gestion, le système DMS constitue le moyen le plus simple et le plus pratique pour faciliter d'une part la planification des tournées de distribution et l'optimisation des circuits de cette catégorie de travailleurs itinérants, et d'autre part le suivi et le contrôle de la part des responsables au sein de l'entreprise.

Notre système permet d'introduire de nouvelles instructions dans son planning, même quand il est en cours de route. Non seulement elle optimise les tournées et les livraisons des commandes, mais contribue à réduire les coûts de gestion. (5)

### **I.7. Besoins fonctionnels**

Nous allons décrire, dans cette partie, les besoins fonctionnels aux quels devrait répondre notre outil logiciel:

#### **➤ Suivi de véhicule en temps réel**

Visualiser les positions instantanées des véhicules et des conducteurs sur une carte géographique.

#### **➤ Historique du parcours**

Afficher la liste des trajets parcourus pour chaque véhicule.

#### **➤ Gestion des données**

Gérer la liste des véhicules : ajouter une nouvelle voiture à la troupe de véhicules ou supprimer une voiture si elle n'en fait plus partie, les véhicules sont organisés par groupes.

Gérer la liste des conducteurs: ajouter, supprimer ou modifier les paramètres d'un conducteur de la liste.

Définitions des points d'intérêt : notre système permet une saisie simple des points d'intérêt de l'entreprise. On s'intéresse pour chaque point à savoir sa position géographique ainsi qu'un ensemble de données attributaires : Le nom, l'adresse, description...

➤ **QR code**

C'est un nouveau service qui nous permet d'assurer et de valider efficacement que notre transporteur a bien fait la visite au bon endroit.

➤ **Gestion des comptes utilisateurs et des privilèges**

- ✓ Consiste à créer, modifier, supprimer ou à afficher les comptes utilisateurs ayant accès à l'application.
- ✓ Un compte utilisateur est défini par un login et un mot de passe, ainsi l'administrateur peut définir les comptes utilisateurs tout en précisant les privilèges associés.
- ✓ L'application contient un système de gestion des rôles et des permissions.

➤ **Génération de rapports statistiques**

L'application permettra de générer des rapports d'activité imprimables et qui sont également disponibles sous format PDF,EXCEL.

### **I.8.Besoins non fonctionnels**

➤ **Contrainte ergonomique**

Notre système doit présenter des informations extraites de la base de données dans une interface conviviale et ergonomique pour faciliter l'utilisation de l'application par un utilisateur, qu'il soit spécialiste ou non, Cet interface doit également assurer la maintenabilité et la réutilisabilité de notre application.

➤ **Contrainte sur la fiabilité de l'application**

Le serveur d'applications doit être capable de gérer un grand nombre d'accès et de requêtes simultanées. D'autre part, en matière de vitesse ou de temps de réponse, l'accès des utilisateurs à leurs tableaux de bords doit être fourni au bout d'un temps réduit, ce qui met au point la nécessité d'un SGBD relationnel qui peut prendre en charge un taux élevé de requêtes.

➤ **Contrainte d'évolution**

Notre application doit permettre une maintenance facile et doit être évolutive.

### **I.9. Description générale du fonctionnement du système**

Le Smartphone permet au récepteur GPS de définir précisément sa localisation, puis il envoie ces coordonnées via 3G/4G vers le serveur en utilisant le protocole HTTP. Après réception et enregistrement des données GPS chez le serveur, le système récupère ces informations à travers un socket (PHP & json) qui permet d'insérer les données du serveur dans la base de données.

### **I.10. Identification des acteurs**

Les acteurs se recrutent parmi les utilisateurs du système et aussi parmi les responsables de sa configuration et sa maintenance. Ils se répartissent dans les catégories suivantes :

#### **I.10.1. L'administrateur du système**

Le système doit permettre à l'administrateur de :

- Gérer les comptes utilisateurs.
- Attribuer des privilèges.
- Gérer les groupes.
- Paramétrer le système (Profil, connectivités, seuil de vitesse).
- Générer des rapports des équipements (rapports détaillés, rapports de performance).
- Localiser un véhicule en temps réel

#### **I.10.2. L'utilisateur du système**

Le système doit permettre à l'exploitant de :

- Consulter l'historique des trajets parcourus.
- Générer des rapports imprimables détaillant l'historique des véhicules.
- Gérer les chauffeurs.
- Gérer les véhicules.
- Gérer les points d'intérêt

### **I.11.Conclusion**

Dans ce chapitre nous avons présenté les méthodes de géolocalisation qui permettent d'obtenir et éventuellement transmettre la position géographique d'une personne ou une machine. La combinaison de différentes techniques est capable de géo localiser le terminal dans n'importe quelle position. Dans le prochain chapitre, nous allons présenter un autre élément que nous utiliserons pour la modélisation de notre système, il s'agit d'ULM, et nous expliquerons le choix des méthodes agiles et les spécifications des besoins.

# **CHAPITRE II**

## **Spécification des besoins**

### II.1.Introduction

Le développement de n'importe quel système d'information nécessite une démarche très importante dans le cycle de vie d'un logiciel, et la modélisation en pratique est importante dans les développements des logiciels, dans notre projet nous choisissons le langage de modélisation UML qui nous allons exposer dans ce chapitre

Le présent chapitre sera consacré à la spécification des besoins du projet et à sa modélisation. Pour cela, nous allons commencer par présenter le langage de modélisation UML que nous avons choisi pour ses divers avantages (modularité, abstraction, structuration cohérente des fonctionnalités et des données, diversité des diagrammes,). Nous définirons par la suite les méthodes de développement auxquelles nous justifierons le choix de cette méthode. Ensuite nous allons exposer le contexte du projet, les exigences de la cliente réaliser les diagrammes de cas d'utilisations.

### II.2.Le langage UML

UML est l'abréviation de « Unified Modeling language », c'est un langage unifié pour la modélisation. UML est un ensemble d'outils pour aider la modélisation de la future des Applications informatiques.

UML modélise l'ensemble des données et des traitements en élaborant des différents diagrammes.

En clair, il ne faut pas désigner UML en tant que méthode mais plutôt comme une boîte d'outils qui sert à améliorer les méthodes de travail. (6)

### II.3.DiagrammesUML

La version UML 2.4 s'articule autour de quatorze types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont repartis en deux grands groupes. (8)

### II.3.1. Diagrammes structurels ou statiques

**Diagramme de classes** : il montre les briques de base statiques : classes, associations, interfaces, attributs, opérations, généralisations, etc.

**Diagramme d'objets** : il montre les instances des éléments structurels et leurs liens à l'exécution.

**Diagramme de packages** : il montre l'organisation logique du modèle et les relations entre packages.

**Diagramme de structure composite** : il montre l'organisation interne d'un

Élément statique complexe.

**Diagramme de composants** : il montre des structures complexes, avec leurs interfaces fournies et requises.

**Diagramme de déploiement** : il montre le déploiement physique des "artefacts" sur les ressources matérielles.

**Diagramme de profile** : il permet de spécialiser, de personnaliser pour un domaine particulier un méta-modèle de référence d'UML.

### II.3.2. Diagrammes comportementaux

**Diagramme de cas d'utilisation** : il montre les interactions fonctionnelles entre les acteurs et le système à l'étude.

**Diagramme d'activité** : il montre l'enchaînement des actions et décisions au sein d'une activité.

**Diagramme d'états** : il montre les différents états et transitions possibles des objets d'une classe.

### II.3.3 Diagrammes d'interaction ou dynamiques :

**Diagramme de vue d'ensemble des interactions :** il fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.

**Diagramme de séquence :** il montre la séquence verticale des messages passés entre objets au sein d'une interaction.

**Diagramme de communication :** il montre la communication entre objets dans le plan au sein d'une interaction.

**Diagramme de temps :** il fusionne les diagrammes d'états et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.

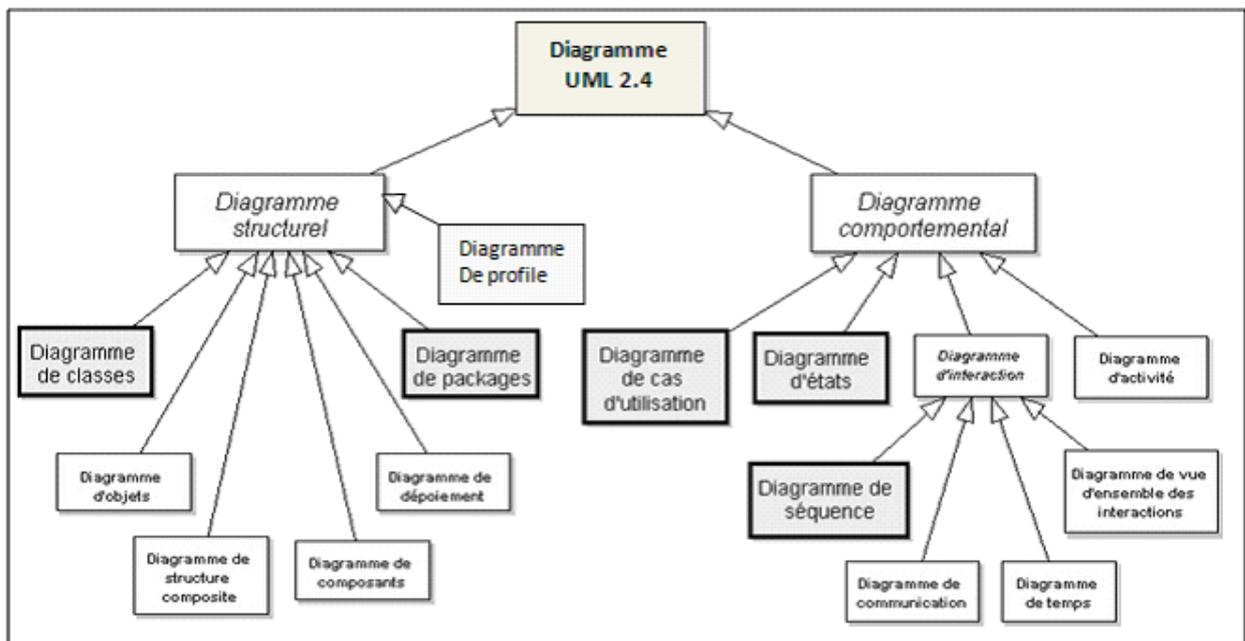


Figure II.1 : Les 14 diagrammes d'UML (9)

#### II.4. Logiciels de modélisation UML

Il existe de nombreux outils logiciels de modélisation UML. Comme le logiciel de dessin Enterprise Architect qui permet la création de diagrammes UML. Le dessin s'effectue en choisissant des formes (flèches, acteurs, classes, notes, états) et en utilisant des bibliothèques de Symboles pour aboutir à un schéma correspondant. Les créations peuvent être exportées dans différents formats (PDF, JPG, PNG ...). Dans ce qui suit, on utilisera la méthode agile pour permettre de répondre mieux au besoin réel du client.

#### II.5. Diagramme de cas d'utilisation

**Définitions :** Il montre les interactions fonctionnelles entre les acteurs et le système à l'étude.

**Acteur :** rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.

**Cas d'utilisation (use case) :** ensemble de séquences d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Collection de scénarios reliant par un objectif utilisateur commun.

**Association :** utilisée dans ce type de diagramme pour relier les acteurs et les cas d'utilisation par une relation qui signifie simplement " participe à ".

**Inclusion :** le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifique dans ses enchaînements.

**Extension :** le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifique indirectement dans celui qui procède à l'extension.

Après la concertation avec les responsables de l'entreprise nous avons recensé un seul type d'acteur qui est :

L'administrateur : c'est le responsable informatique qui gèrera l'application.

### II.6. Diagramme de cas d'utilisation cote d'administrateur

En partant des besoins de l'administrateur, on peut détecter les cas d'utilisation correspondant l'acteur " administrateur " et qui sont représentés dans le diagramme de la Figure 2.3 :

- L'administrateur doit s'identifier afin d'accéder à l'espace administrateur
- Les cas d'utilisation correspondant à l'acteur " administrateur " dans la Figure 2.3 :

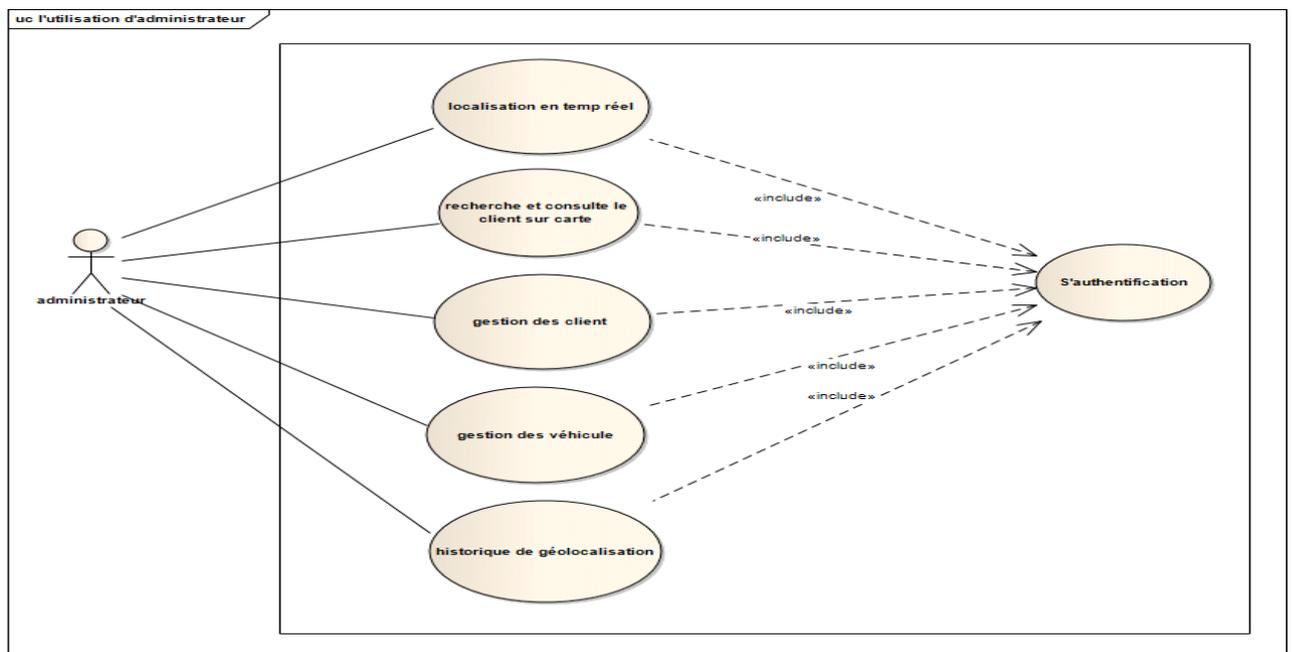


Figure II.3 : Diagramme de cas d'utilisation d'administrateur

## II.6.1. Tableau résume le diagramme de cas d'utilisation pour l'administrateur

Cas d'utilisation	Acteurs	Messages émis/reçus
<b>Gestion du client</b>	Administrateur	<b>Emis :</b> Ajouter/supprimer /modifier un client <b>Reçu :</b> List des clients
<b>Gestions des camions</b>	Administrateur	<b>Emis :</b> Ajouter/supprimer/modifier un camion <b>Reçu :</b> List des camions
<b>Gestions des comptes utilisateur</b>	Administrateur	<b>Emis :</b> Ajouter/supprimer/modifier un compte <b>Reçu :</b> List des comptes
<b>Recherche et consultation des clients sur la carte</b>	Administrateur/Utilisateur	<b>Emis :</b> Saisir le lieu <b>Reçu :</b> Une carte géographique Avec un marqueur contenant l'information de l'endroit
<b>Historique de localisation</b>	Administrateur/Utilisateur	<b>Emis :</b> Sélectionner un camion <b>Reçu :</b> l'information de l'endroit Affichage des données sous forme d'un tableau

TableauII.1 : Tableau de C.U pour l'administrateur

## II.7. Diagramme de séquence

### ❖ Définitions

Les notations suivantes sont utilisées lors de la réalisation des diagrammes de séquence.

### ❖ Diagramme de séquence

Il montre la séquence verticale des messages passés entre objets au sein d'une interaction.

- **Rôle** : Montre la séquence verticale des messages passés entre objets au sein d'une interaction.

- **Ligne** : De vie représentation de l'existence d'un élément participant dans un diagramme de séquence. Cela peut être un acteur ou le système en modélisation d'exigences, des objets logiciels en conception préliminaire ou conception détaillée.

- **Message** : Elément de communication unidirectionnel entre objets qui déclenche une activité dans l'objet destinataire. La réception d'un message provoque un événement dans l'objet récepteur. La flèche pointillée représente un retour au sens UML. Cela signifie que le message en question est le résultat direct du message précédent.

- **Spécification d'activation** : Bande bleu qui représente une période d'activité sur une ligne de vie.

- **Message synchrone** : Envoi de message pour lequel l'émetteur se bloque en attente du retour et qui est représenté par une flèche.

- **Occurrence d'interaction** : Une interaction peut faire référence explicitement à une autre interaction grâce à un cadre avec le mot-clé `ref` et indiquant le nom de l'autre interaction.

- **Les cadres d'interaction** : Chaque cadre possède un opérateur et peut être divisé en fragments. Les principaux opérateurs sont :

- **Loop** : boucle. Le fragment peut s'exécuter plusieurs fois, et la condition de garde explicite l'itération.

- **opt** : optionnel. Le fragment ne s'exécute que si la condition fournie est vraie.
- **alt** : fragments alternatifs. Seul le fragment possédant la condition vraie s'exécutera.

### II.8.Réalisation des diagrammes de séquences

Nous allons maintenant présenter quelques diagrammes de séquence qui sont les plus Utiles pour la suite de notre travail

### II.9.Diagramme de séquence pour l'authentification

L'authentification se fait de la manière suivante : L'utilisateur demande l'accès au Système (demande la page authentification).Le système affiche le formulaire, l'utilisateur le remplit par le pseudo et le mot de passe une fois valides, le système vérifie l'existence de l'utilisateur et trois cas sont possibles :

- Les données sont correctes donc, la page correspondante s'affiche pour l'utilisateur.
- Champs obligatoires non valides et/ou vides, le système affiche un message d'erreur Champs obligatoire.
- Les données sont incorrectes donc, un message d'erreur apparait et réaffichage du formulaire d'authentification

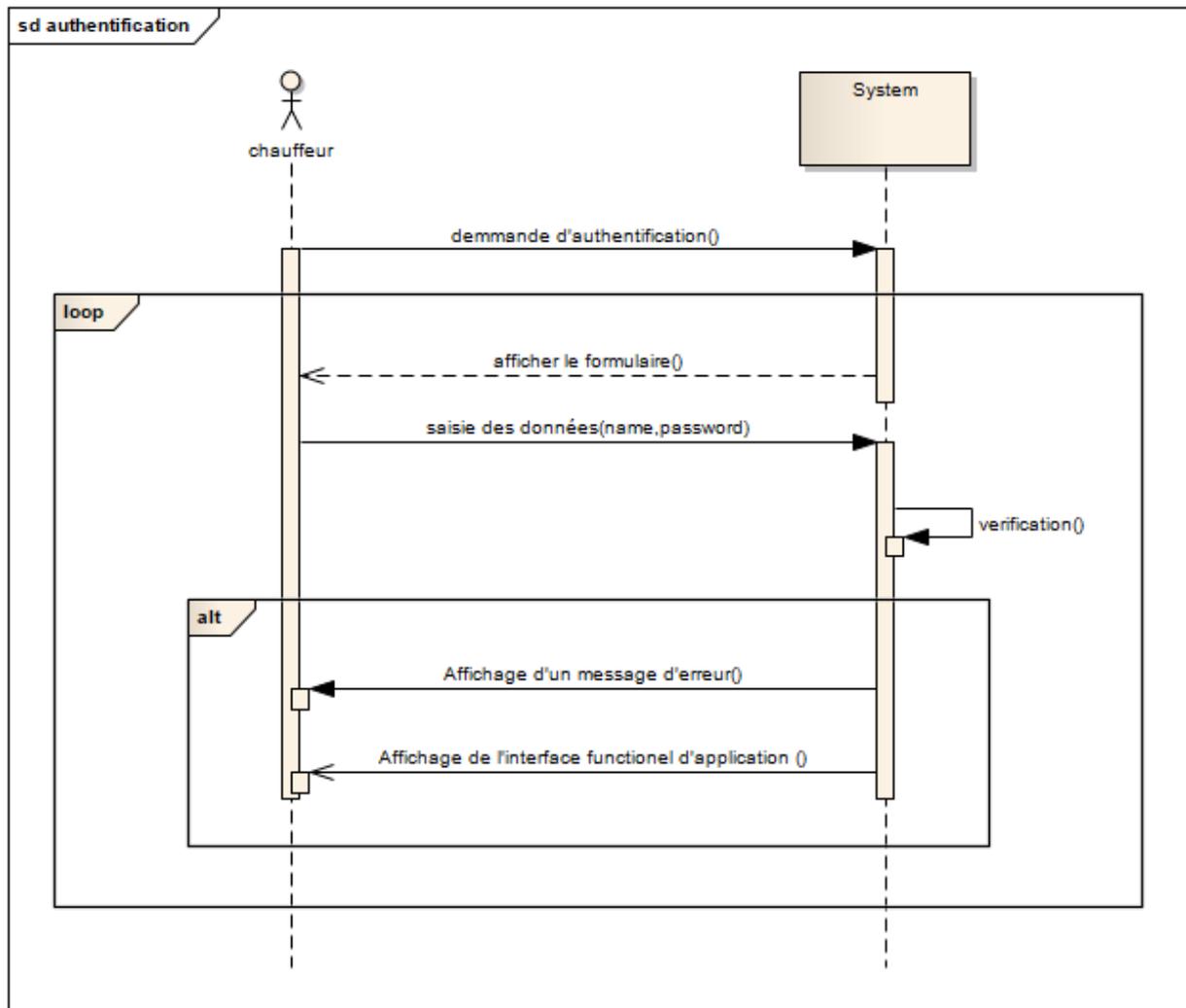
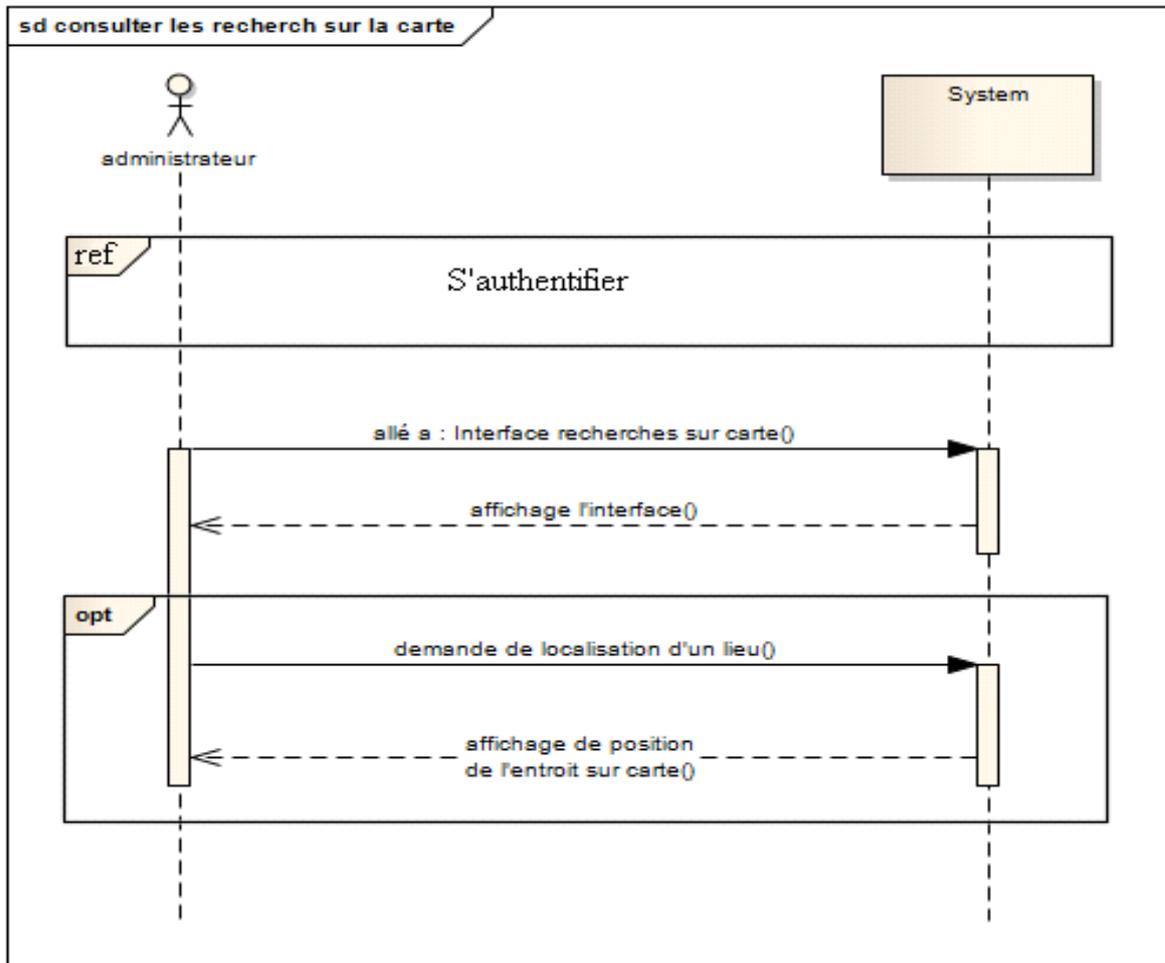


Figure II.4 : Diagramme de séquence pour l'authentification

**II.10. Diagramme de séquence pour la géolocalisation recherche et consulte les clients sur map****Figure II.5 :** Diagramme de séquence pour la géolocalisation

### II.11. Diagramme de séquence pour consulter l'historique de localisation

Une fois que l'utilisateur est authentifié. Il se présente devant l'interface de consultation de l'historique, puis il demande la liste des camions, le système répond avec une liste de camions. L'utilisateur sélectionne un camion et fixe un intervalle temporel pour la consultation. Deux options sont offertes pour affichage :

- Affichage de l'historique de la localisation du camion sous forme d'un tableau.

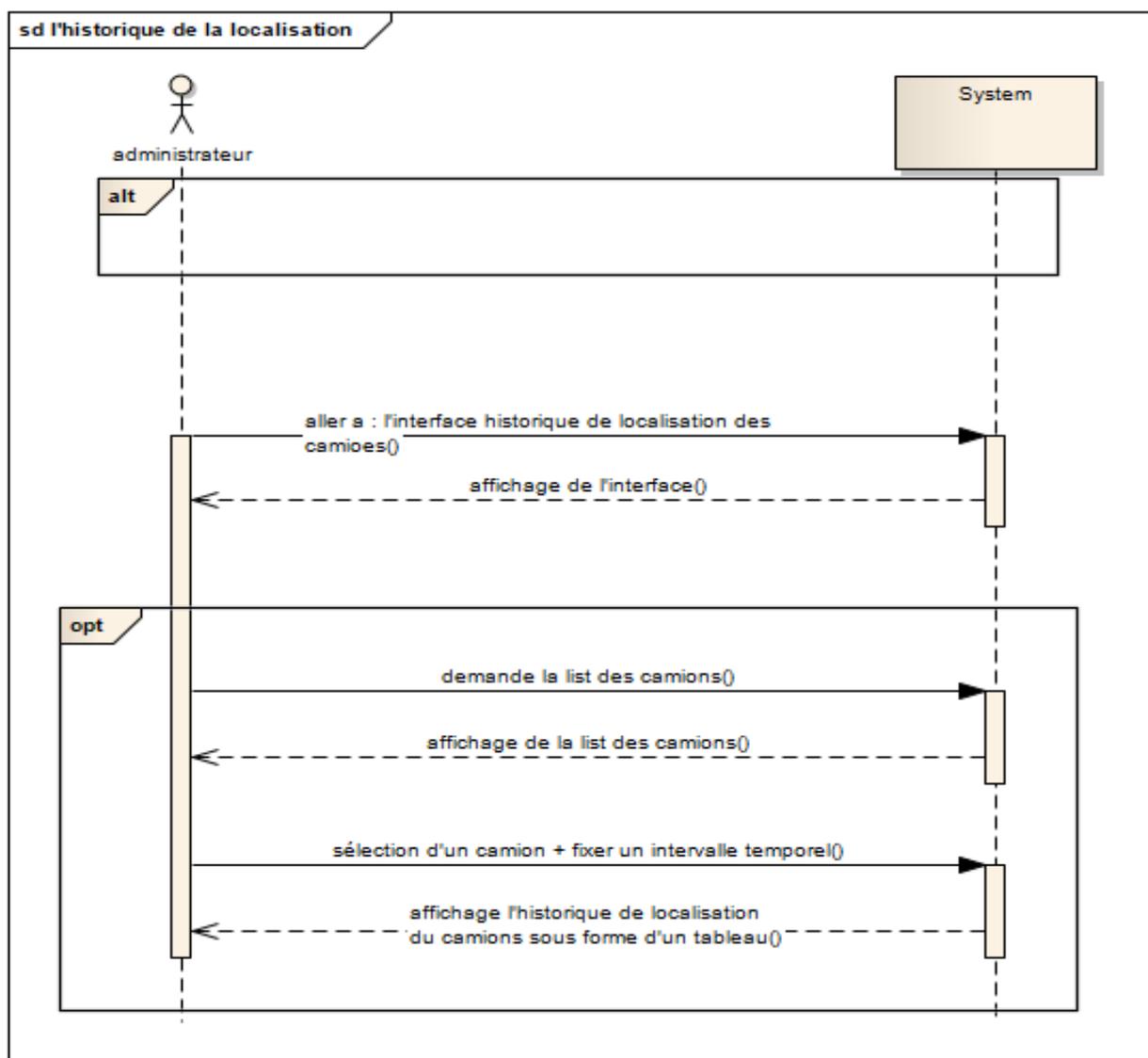


Figure II.6 : Diagramme de séquence pour l'historique de localisation

II.12. Gestion des clients

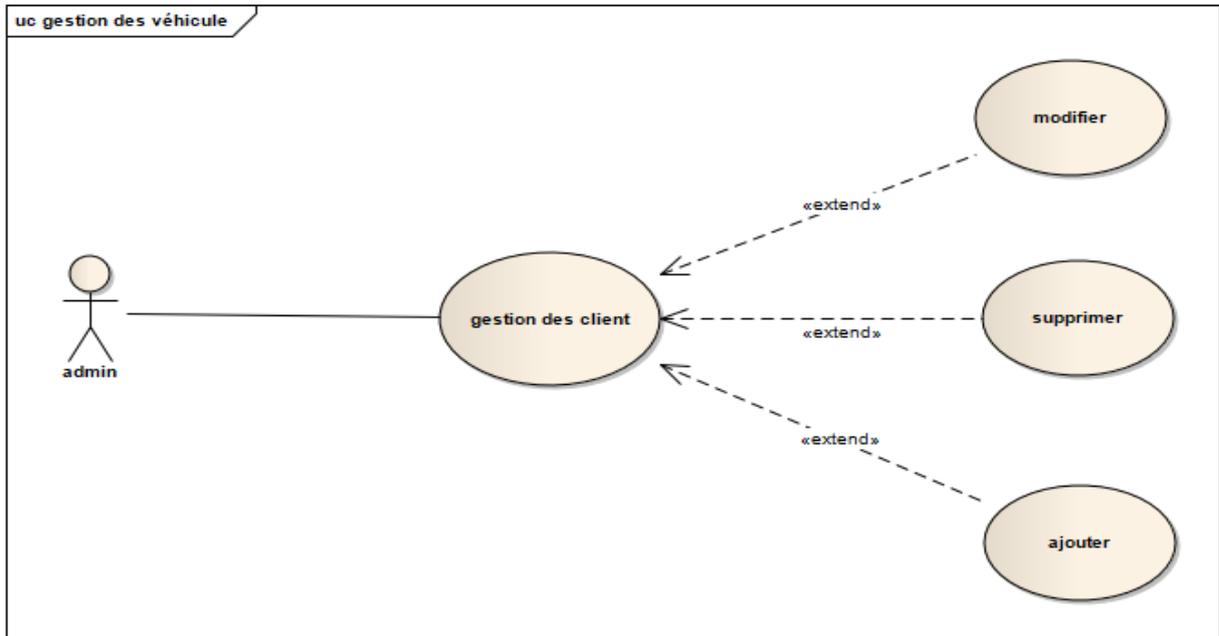


Figure II.7 : Diagramme de cas d'utilisation de La gestion des clients

II.13. Gestion des vans

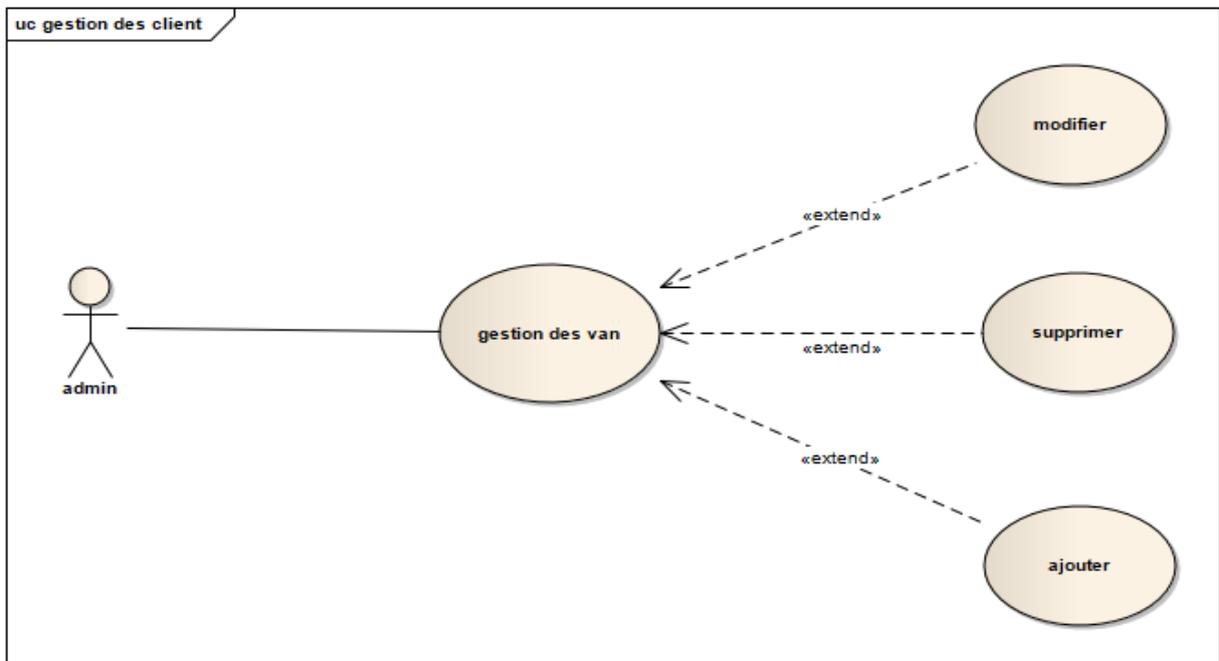


Figure II.8 : Diagramme de cas d'utilisation de La gestion des vans

II.14. Diagramme de séquence cote utilisateur application Android

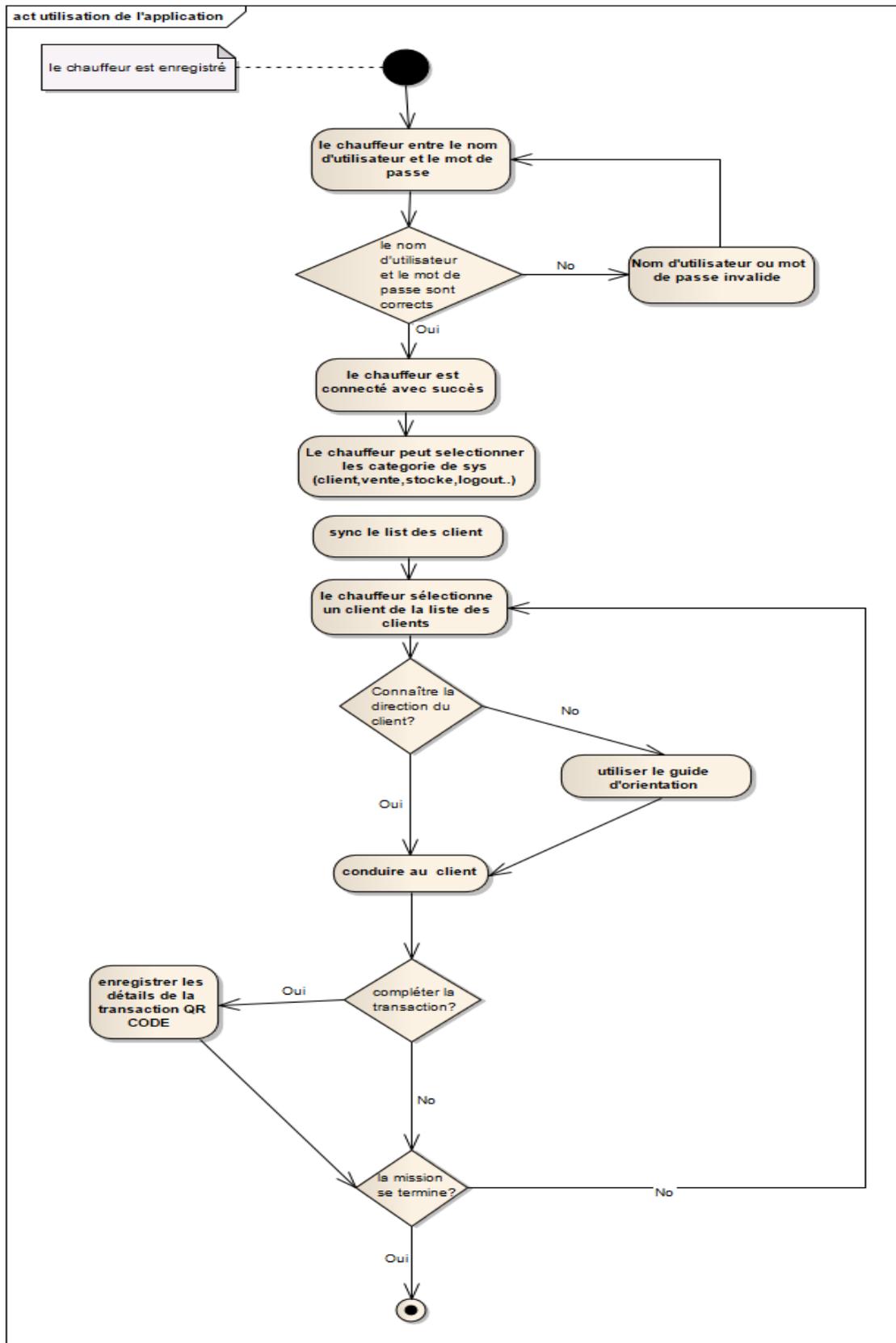


Figure II.9 : Diagramme de séquence pour l'utilisation d'application Android

### II.15. Diagramme de séquence pour l'authentification

L'authentification se fait de la manière suivante : L'utilisateur demande l'accès à l'application Android (demande l'activité d'authentification). L'application affiche le formulaire, l'utilisateur le remplit par le pseudo et le mot de passe une fois valides, le système vérifie l'existence de l'utilisateur et trois cas sont possibles :

- Les données sont correctes donc, la page correspondante s'affiche pour l'utilisateur.
- Champs obligatoires non valides et/ou vides, le système affiche un message d'erreur.
- Les données sont incorrectes donc, un message d'erreur apparaît et réaffichage du formulaire d'authentification.

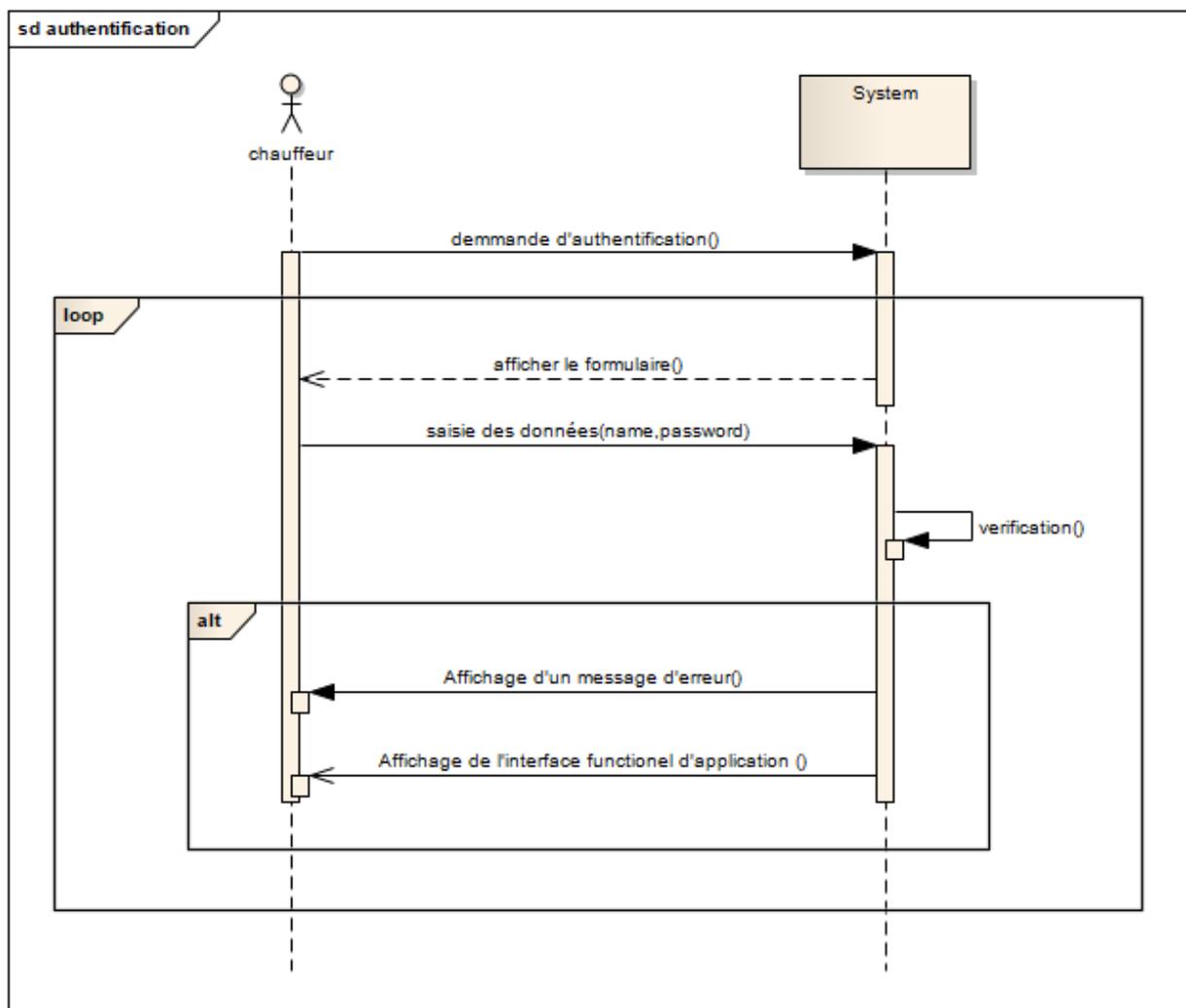


Figure II.10 : Diagramme de séquence pour l'authentification

II.16. Diagramme de séquence de synchronisation d'une liste de client

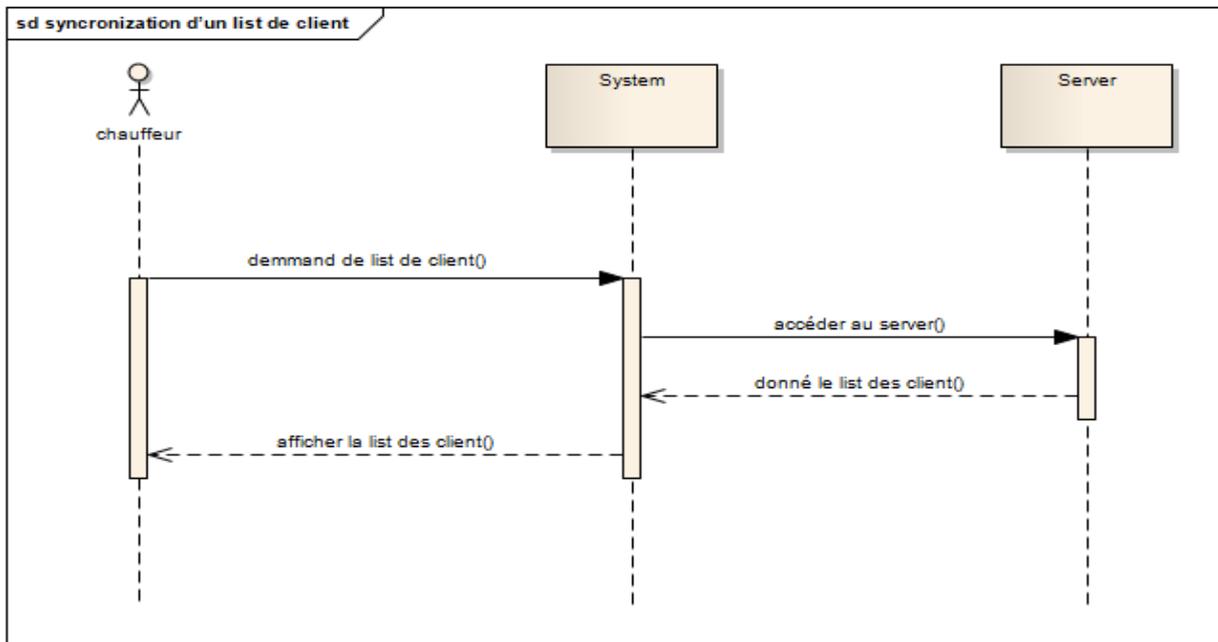


Figure II.11 : Diagramme de séquence de synchronisation d'une liste de client

II.17. Diagramme de séquence de géolocalisation

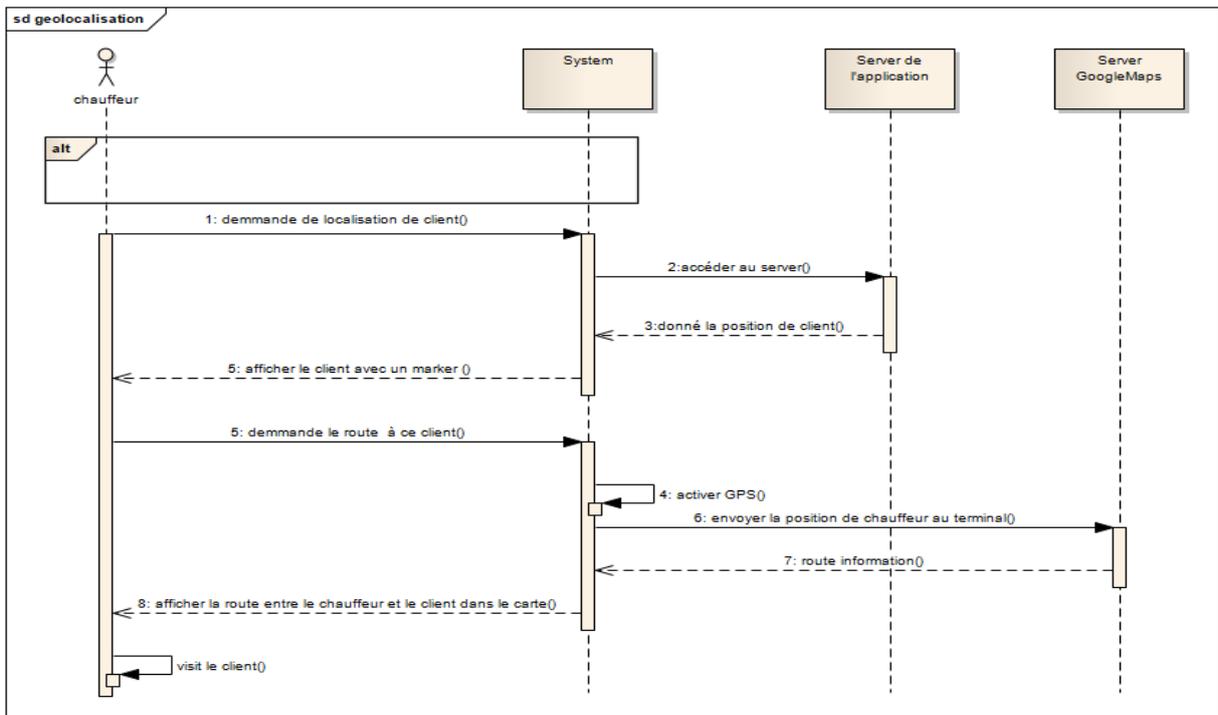


Figure II.12 : Diagramme de séquence de géolocalisation

## II.18. Diagramme de séquence de l'historique

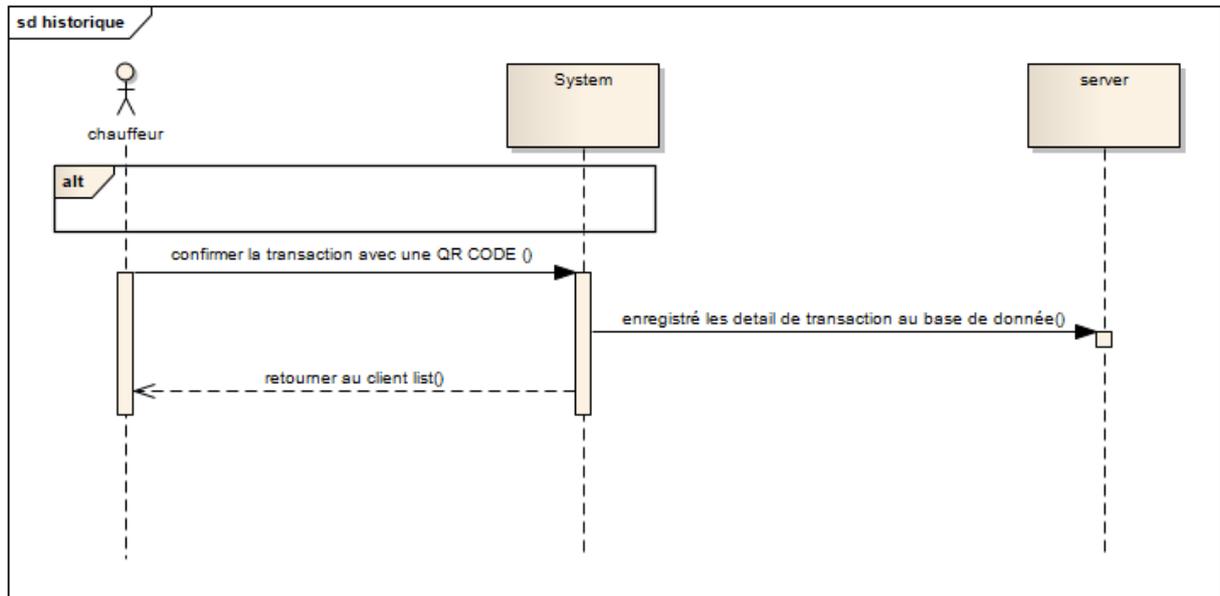


Figure II.13 : Diagramme de séquence de l'historique

## II.19. Conclusion

Ce chapitre nous a permis de présenter le langage de modélisation UML et le processus de développement unifié (Unified Processus). Ensuite on a déterminé le cadre du projet de réalisation de l'application et de définir les besoins, en identifiant toutes les entités internes qui vont interagir avec le système (acteurs). Nous avons représenté l'ensemble de séquence d'actions, en identifiant les cas d'utilisation de tous les acteurs du système. Ensuite, nous avons recensé la description graphique des cas d'utilisation, en réalisant des diagrammes de séquence décrivant les scénarios nominaux des cas d'utilisations essentiels.

# **CHAPITRE III**

## **Réalisation**

### III.1. Introduction

Dans ce troisième et dernier chapitre, on s'intéressera à la réalisation de notre système (DMS). Nous allons tout d'abord présenter les différents outils et langages utilisés pour la réalisation. Ensuite, nous présenterons les différentes fonctionnalités offertes par le système et cela sera illustré par la présentation des différentes interfaces de l'application.

### III.2. Langages de programmation

#### III.2.1. Langage HTML5

C'est un langage de balise permettant le codage

Des pages WEB. HTML permet également de structurer sémantiquement et de mettre en forme l'interface des sites, d'inclure des ressources multimédias telles que les images,

les formulaires de saisie, et les programmes informatiques. Il permet de créer des documents Interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation et des formats de présentation (feuilles de style en cascade).

HTML est initialement dérivé du Standard Generalized Markup Language (SGML).

#### III.2.2. Langage CSS

CSS est l'acronyme de Cascading Style Sheet, est un langage de conception simple destiné à simplifier le processus de présentation des pages Web, donc utilisé sur l'internet pour mettre en forme les fichiers HTML ou XML, donc ce code pour gérer le design d'une page web.

#### III.2.3. Langage java

« Java est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables. »



### III.2.4. Langage java script

Le javascript est un langage informatique utilisé sur les pages web. Ce langage a la particularité de s'activer sur le poste client, en d'autres mots c'est votre ordinateur qui va recevoir le code et qui devra l'exécuter. C'est en opposition a d'autre langage qui sont activé côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Firefox ou Internet Explorer.



### III.2.5. jQuery

Est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript (comprenant Ajax) et HTML, et a pour but de simplifier des commandes communes de JavaScript.

### III.2.6. Langage PHP

Plus connu sous le nom de PHP, c'est un langage de programmation WEB principalement utilisé pour produire des pages Web dynamiques (client/serveur) via un serveur HTTP (ex : Apache), on désigne parfois PHP comme une plateforme plus qu'un simple langage. Les codes du PHP sont appelés « scripts », et ils sont inclus dans le code HTML.



### III.2.7. Langage SQL

SQL ("Langage de requêtes structuré" ou "Structured Query Language") est un langage informatique destiné à interroger ou piloter une base de données.



### III.2.8. Langage XML

XML (eXtensible Markup Language) (16) est un langage HTML amélioré permettant de définir de nouvelles balises (markup) et la mise en forme des documents. Le XML se veut également compatible avec le web afin que les échanges de données puissent se faire facilement à travers le réseau Internet.

## III.3. Environnements de développement

### III.3.1. App serveur

App serveur est un serveur d'évaluation qui regroupe tous les éléments indispensables à l'évaluation d'un site web dynamique (serveur web APACHE, serveur de bases de données MySQL et PhpMyAdmin) :

### III.3.2. Apache

Est un serveur HTTP créé et maintenu au sein de la fondation Apache. Utilisé principalement sur les hébergements Internet en Linux, bien qu'il soit également utilisable en Windows.



### III.3.3. PhpMyAdmin

PhpMyAdmin est un utilitaire qui consiste en un ensemble de scripts PHP rendant plus conviviale l'administration de bases de données MySQL en passant par un navigateur web.

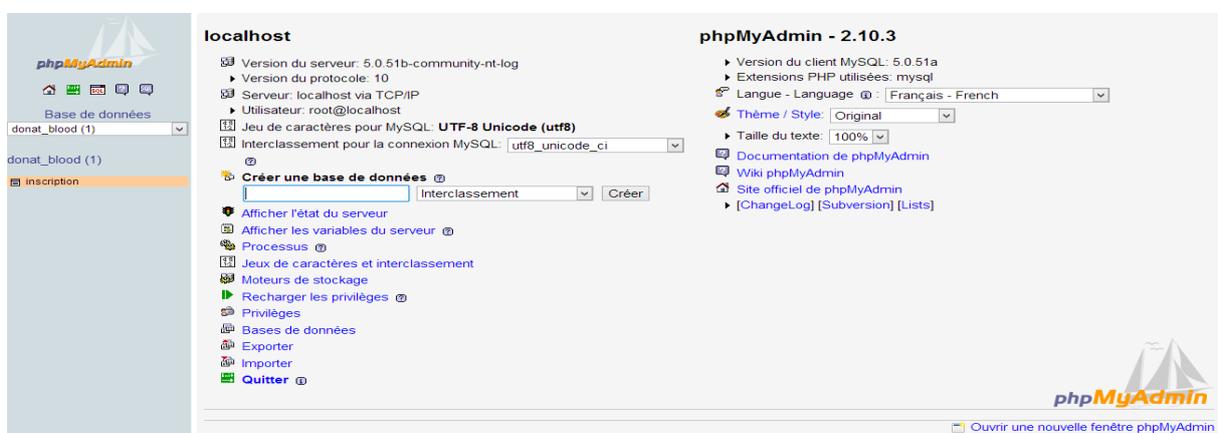


Figure III.1: Php My Admin

### III.3.4. MySQL

Est un système de gestion de base de données (SGBD). Comme serveur de bases de données relationnelles Open Source, Basé sur Structured Query Language (SQL). Aussi MySQL est le plus souvent associé à des applications basées sur le Web



Nous utilisons MySQL pour la gestion des bases de données de notre application.

### III.3.5. Android Studio

Android Studio est un environnement de développement intégré (IDE) pour le développement sur la plateforme Android. Il a été annoncé en mai 2013. Android est disponible librement sous la licence Apache 2.0. Basé sur le logiciel IDEA de JetBrains 'IntelliJ', Android Studio est conçu spécifiquement pour le développement Android. Il est disponible en téléchargement sur les systèmes d'exploitation ; Windows, Mac OS et Linux.

Android Studio permet principalement d'éditer les fichiers Java et les fichiers de configuration d'une application Android. Il propose aussi des outils pour gérer le développement d'applications multilingues et permet de visualiser la mise en page des différents types et tailles d'écrans avec des résolutions variées simultanément



### III.3.6. SDK

Le SDK est un ensemble d'outils que met à disposition Google afin de nous permettre de développer des applications pour Android. Il est disponible pour Windows, MacOS X et linux et inclut des outils ainsi qu'un émulateur Android pour exécuter des applications

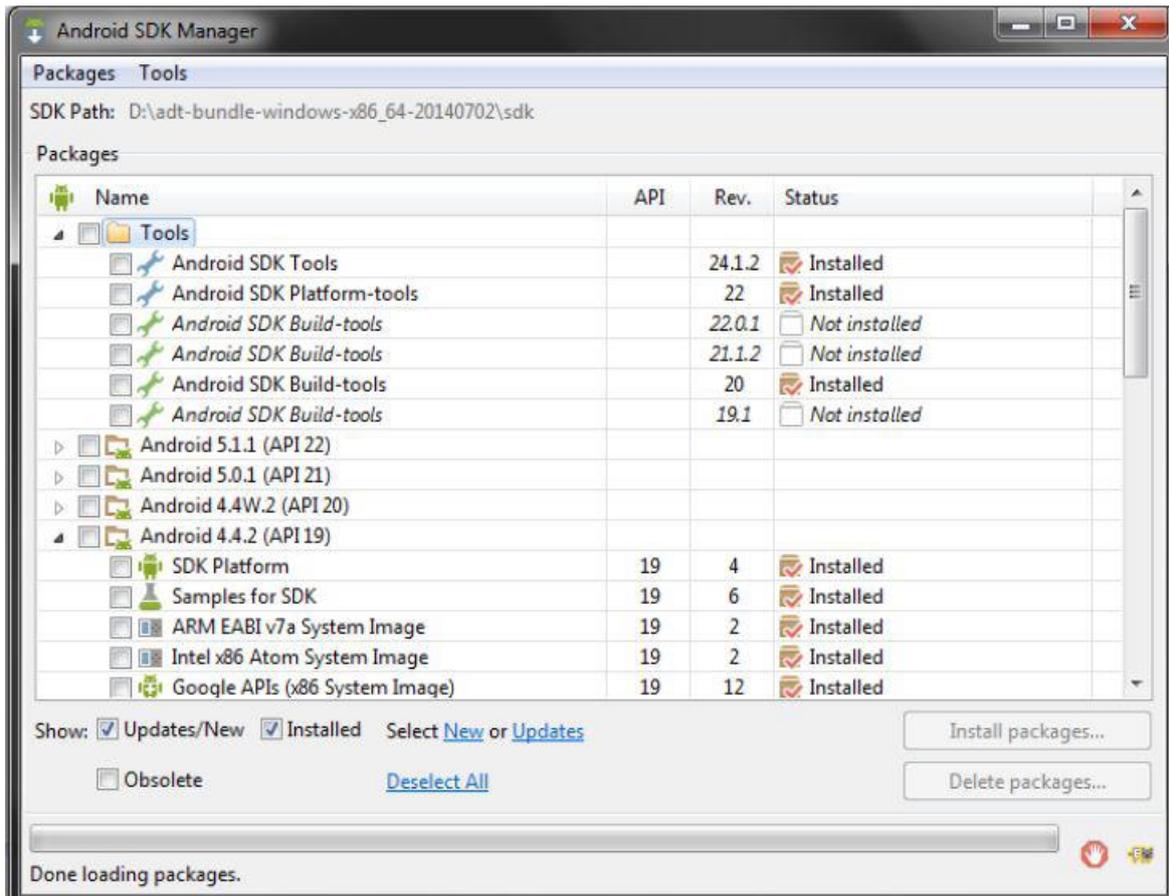


Figure III.2 : SDK

### III.3.7. Le SDK Android

Le Software Development Kit Android est un ensemble de fichiers d'aide et d'exemples. On y trouve aussi des utilitaires pour la mise au point et les tests.

### III.3.8. Les fonctions du SDK

- ✓ Accès au Hardware, y compris Camera, GPS, et Accéléromètre.
- ✓ Base de données MySQL.
- ✓ Données et dépôt de données partagées et communication inter application par échange de messages.
- ✓ Ecran d'accueil riche par l'utilisation des Widgets, Live Folders, and Live Wallpaper Support Média très riche et graphiques 2D/3D : Rendus graphiques par HW optimisé pour la mobilité, incluant une librairie " path - based " pour les rendus 2D et le support pour les graphiques 3D utilisant OpenGL ES 2.0
- ✓ Accès au HW Wifi et librairies pour l'utilisation du Bluetooth pour le transfert Peer to-Peer.
- ✓ Technologies réseau GSM, EDGE, et 3G pour la téléphonie ou le transfert de données, permettant de placer des appels téléphoniques, des SMS, et d'envoyer et de recevoir des données en utilisant les réseaux de données mobiles.

### III.3.9. Android virtual device (AVD)

L'Android Virtual Device, aussi appelé AVD, est un émulateur de terminal sous Android, c'est-à-dire que c'est un logiciel qui se fait passer pour un appareil sous Android sur ordinateur. C'est la raison pour laquelle on a pas besoin d'un périphérique sous Android pour développer et tester la plupart des applications ! En et, une application qui un calendrier par exemple peut très bien se tester dans un émulateur, mais une application qui exploite le GPS doit être éprouvée sur le terrain pour que l'on soit certain de son comportement. (20)



Pour créer ou bien lancer un émulateur.

**Figure III.3 :** Android virtual device (AVD)

### III.3.10 Service web

Les services web (21) (en anglais web services) représentent un mécanisme de communication entre applications distantes à travers le réseau internet indépendant de tout langage de programmation et de toute plate-forme d'exécution.

### III.3.11. Communication avec la base de données

#### ❖ Architecture

Pour que le client Android se connecter à une base de données MySQL, la solution la plus répandue est d'écrire et d'exécuter des scripts PHP à l'aide d'un Protocole http. Ensuite coder les données dans le format JSON, afin de récupérer les données par le client Android. Le client Android, le serveur web et le serveur de base de données forment une architecture

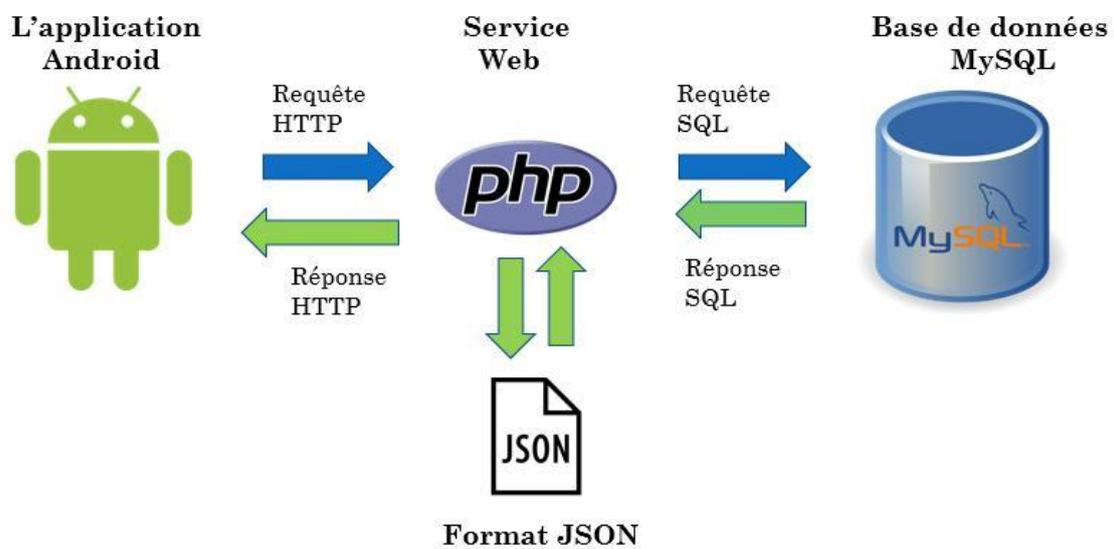


Figure III.4: Architecture (17)

### ❖ Json

JSON (22) (JavaScript Object Notation – Notation Objet issue de JavaScript) est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou générale par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd Edition - Décembre 1999).

JSON est un format texte complètement indépendant de tout langage, mais les conventions qu'il utilise seront familières à tout programmeur habitué aux langages descendant du C, comme par exemple : C lui-même, C++, C#, Java, JavaScript, Perl, Python et bien d'autres. Ces propriétés font de JSON un langage d'échange de données idéal. (cf.annexe1)

### ❖ Choix de l'architecture de l'application

Dans l'architecture à trois niveaux, les applications au niveau serveur sont délocalisées, c'est-à-dire que chaque serveur est spécialisé dans une tâche (serveur web/ serveur de base de données par exemple). Il permet :

- une plus grande flexibilité/souplesse ;
- une sécurité accrue car la sécurité peut être définie indépendamment pour chaque service, et à chaque niveau ;
- de meilleures performances, étant donné le partage des tâches entre les différents serveurs.

Cette architecture (appelée 3 tiers) fait intervenir trois parties indépendantes les unes des autres.

### III.3.12. Google Maps

#### ❖ Définitions

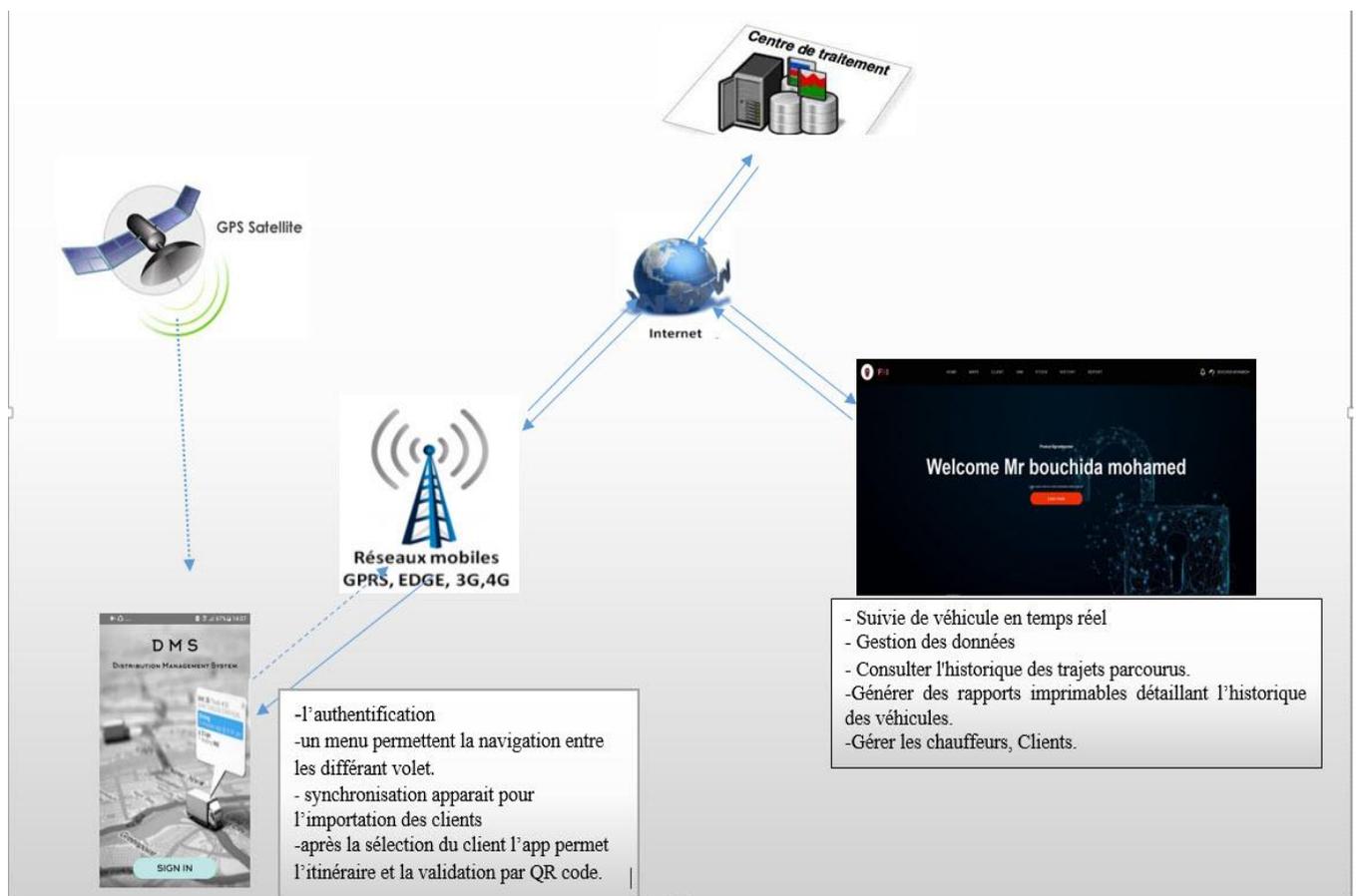
Google Maps (24) est un service gratuit de cartographie en ligne. Google Maps est un service basé sur le web qui fournit des informations détaillées sur les régions géographiques et des sites dans le monde entier. En plus des cartes routières classiques, Google Maps propose des vues aériennes et par satellite de nombreux endroits.

❖ **Api GoogleMaps**

Api de Google (25) permettant de geolocaliser des adresses sur une carte à l'aide de ses coordonnées (latitude et longitude). Api permet de localiser tout type de données sur une carte (routière, satellite, mixte) à partir de sa localisation. Cet api s'avère très utile pour proposer aux internautes une vision globale et géographique de données. L'élément important que l'on retrouve dans le code est l'importation de la librairie Google Maps grâce aux balises. `<script type="text/JavaScript" src="http://maps.google.com/maps/api/js?sensor=false"></script>`

GoogleMaps est basé sur JavaScripts. Lorsque l'utilisateur cherche un camion, une petite bulle est affichée sur la carte pour indiquer la position du terminal.

**III.4. Schéma de fonctionnement de l'application**



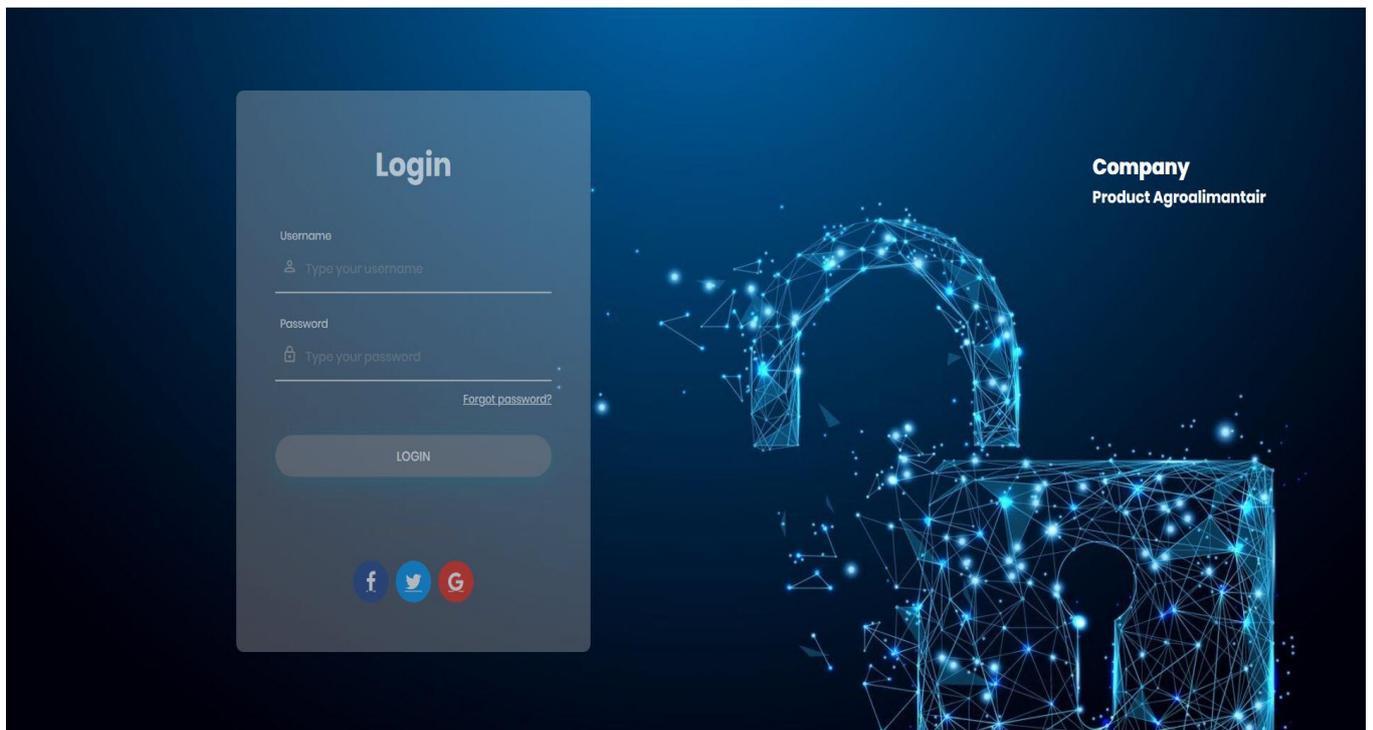
**Figure III.5 :** Schéma de fonctionnement de l'application

### III.5. Présentation du site web

#### III.5.1. L'authentification

Dans cette partie, nous allons présenter quelques cas d'utilisations, sous forme d'un guide utilisateur.

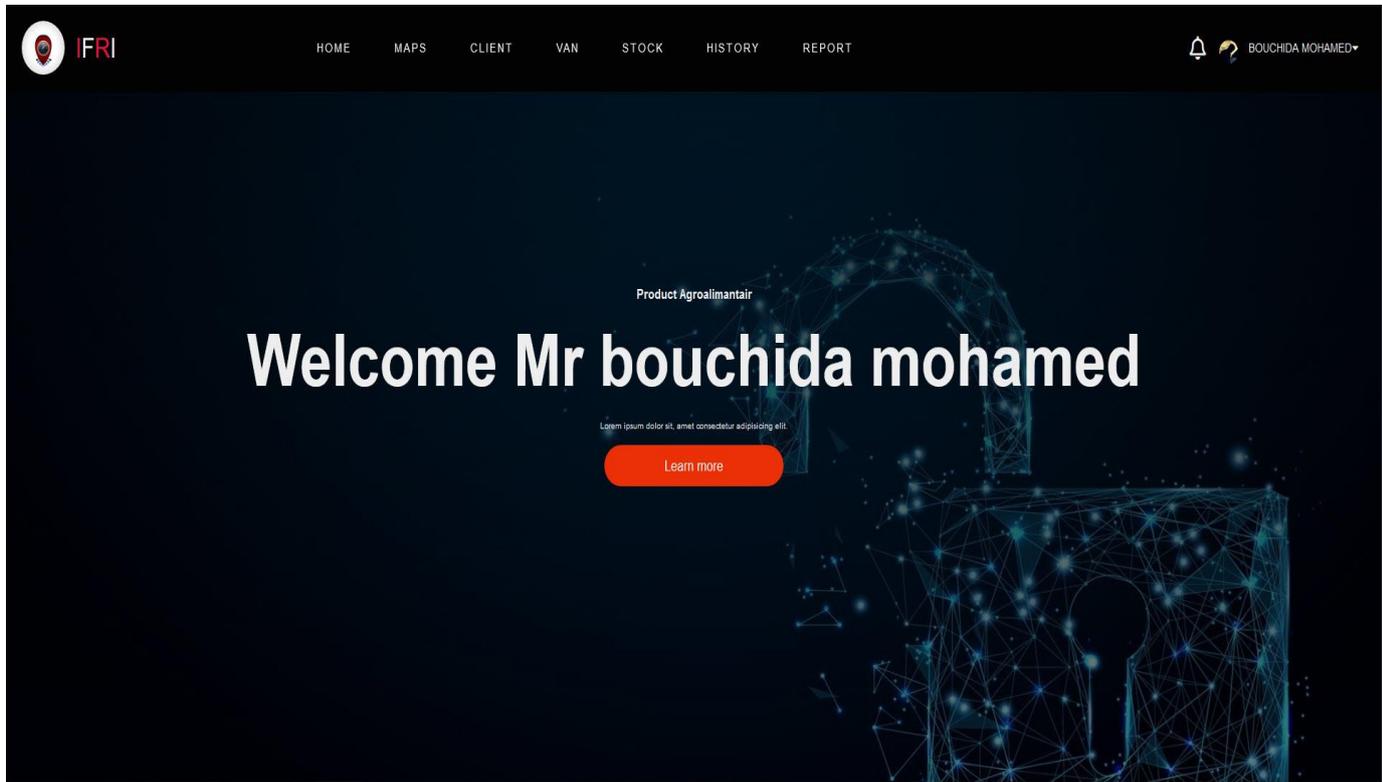
Pour accéder à notre site web l'administrateur de notre système DMS doit s'authentifier. Comme tout système, la sécurité d'accès est nécessaire. La figure ci-après donne l'interface à travers laquelle l'administrateur s'identifie. Il saisit son username et son password puis le serveur vérifie ces informations.



**Figure III.6 :** Présentation de quelques interfaces « L'authentification »

### III.5.2. La page d'accueil

Une fois les données sont valides, l'administrateur accède au menu. Dans cette interface l'administrateur peut gérer toutes les fonctionnalités de notre système. ainsi qu'un espace pour communiquer avec le développeur si besoin.



**Figure III.7 :** La page d'accueil

### III.5.3. La géolocalisation des clients

Nous permet visualiser et rechercher nos clients sur la carte et afficher les informations nécessaires de chaque client

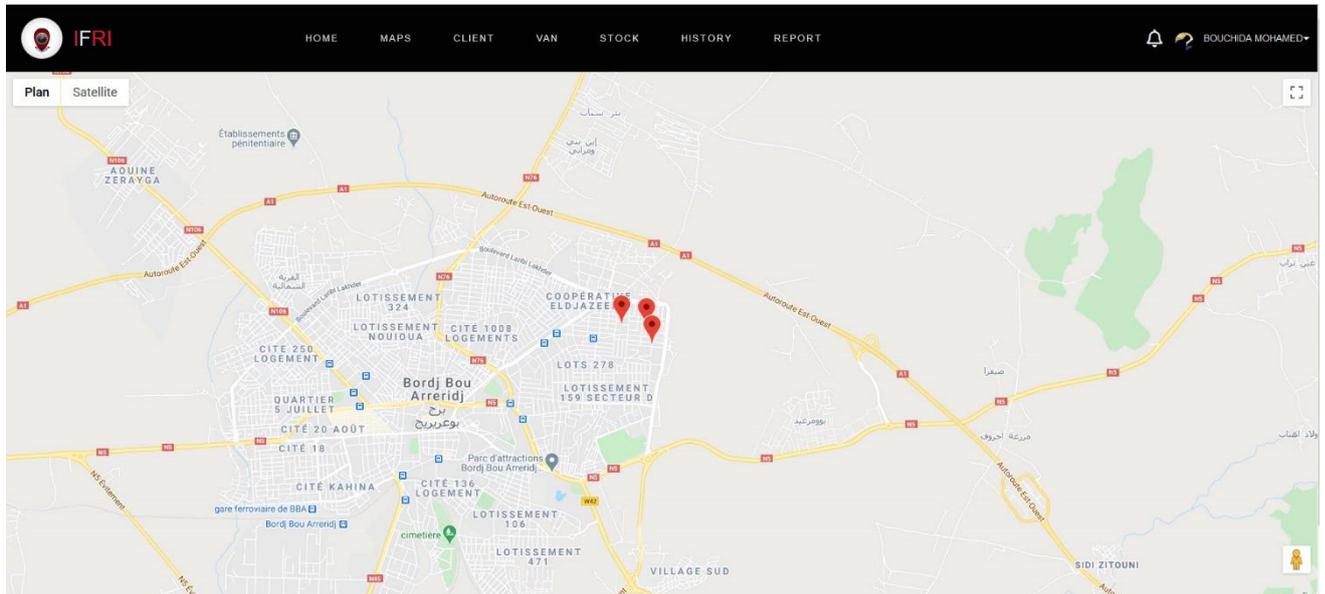


Figure III.8 : La géolocalisation des clients

### III.5.4. Liste des clients

Grace à cette interface on peut visualiser les informations des clients (nom, adresse, position GPS) et permet d’ajouter des nouveaux clients qui sont ajouter automatiquement une marque sur la carte ainsi que de supprimer et éditer les paramètres et informations .

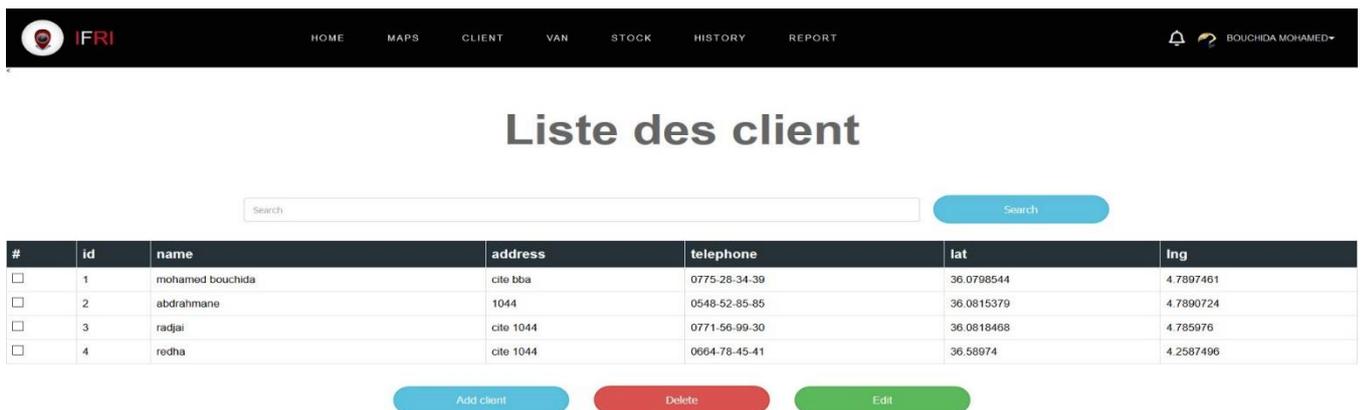
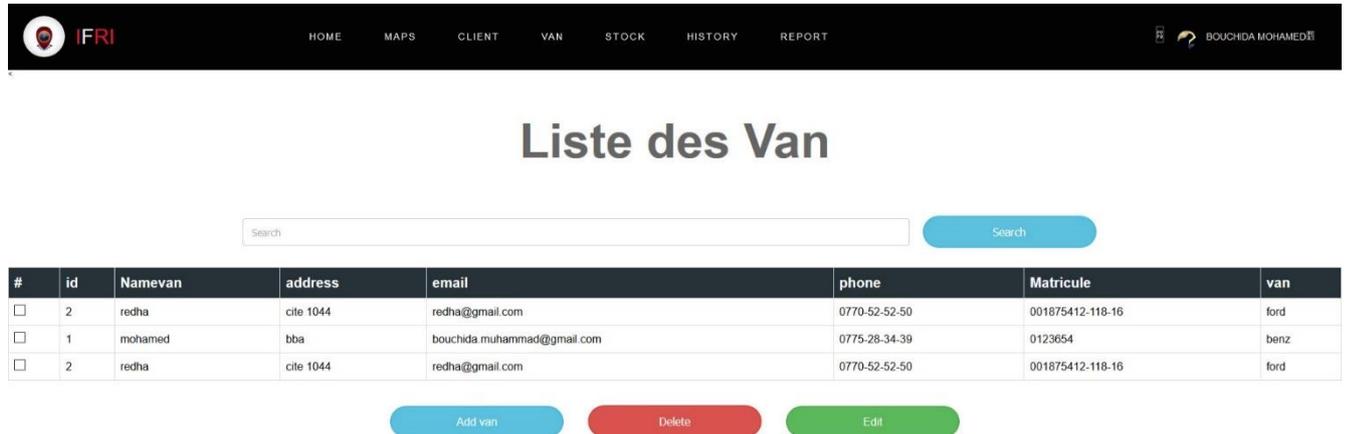


Figure III.9 : Liste des clients

### III.5.5. Liste des vans (véhicules)

Sur cette interface on peut visualiser les informations des chauffeurs (nom d'utilisateur, mot de passe, adresse, num tel) et permet d'ajouter des nouveaux van ainsi que de supprimer et éditer ses paramètres et informations).



#	id	Namevan	address	email	phone	Matricule	van
<input type="checkbox"/>	2	redha	cite 1044	redha@gmail.com	0770-52-52-50	001875412-118-16	ford
<input type="checkbox"/>	1	mohamed	bba	bouchida.muhammad@gmail.com	0775-28-34-39	0123654	benz
<input type="checkbox"/>	2	redha	cite 1044	redha@gmail.com	0770-52-52-50	001875412-118-16	ford

Figure III.10 : Liste des vans (véhicules)

### III.5.6. L'historique des visites

Grace à cette interface on peut consulter les visites de chaque van et suivre en temps réel la progression de notre flotte, avec le nom de client et sa position et la date et l'heure de visite.

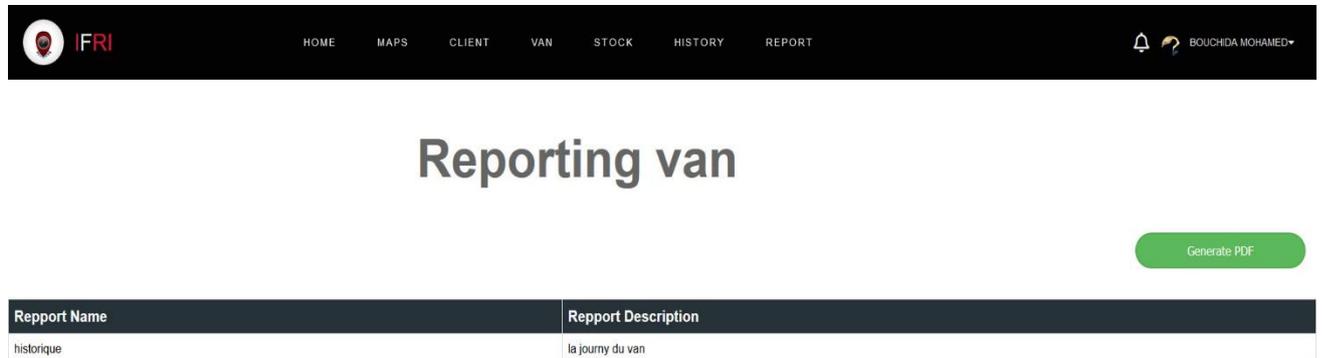


#	id	Name	Address	Lat	Lng	Date	Time
<input type="checkbox"/>	1	mohamed bouchida	cite 1044	34.625487	4.7125487	12/10/2020	16 21

Figure III.11 : L'historique des visites

### III.5.7. Rapport

Grace à cette interface on peut consulter et sauvegarder sous forme PDF les visites de chaque van, avec le nom de client, position et la date et l'heure de visite.

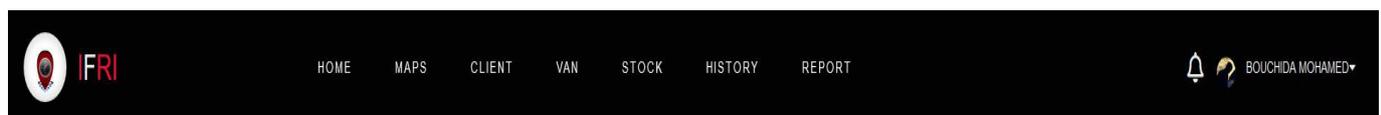


Repport Name	Repport Description
historique	la journy du van

**Figure III.12 : Rapport**

### III.5.8. Paramètre de compte

Cette section est dédié a la gestion du compte admin, elle permet la modification du mot de passe ainsi que la déconnection.

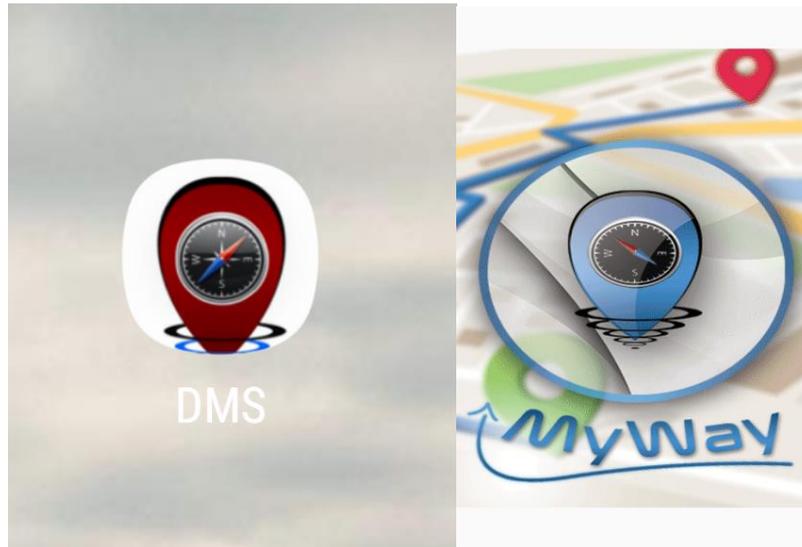


**Figure III.13 : Paramètre de compte**

### III.6. Partie application mobile :

#### III.6.1 Présentation des interfaces Android

Tout d'abord notre application est affichée parmi les applications sur l'écran avec un logo **DMS** est pensé pour rappeler la vocation de notre système, après l'entrée d'une image gif est affichée lors du chargement.



**Figure III.14 :** Présentation de quelques interfaces Android

#### III.6.2. L'authentification

Dans cette interface, nous allons présenter quelques cas d'utilisations, sous forme d'un guide utilisateur.

Pour accéder à l'application DMS l'utilisateur de notre système DMS doit s'authentifier. Comme tout système, la sécurité d'accès est nécessaire. La figure ci-après donne l'interface à travers laquelle l'utilisateur s'identifie. Il saisit son username et son password dans la boîte de dialogue puis le serveur vérifie ces informations.

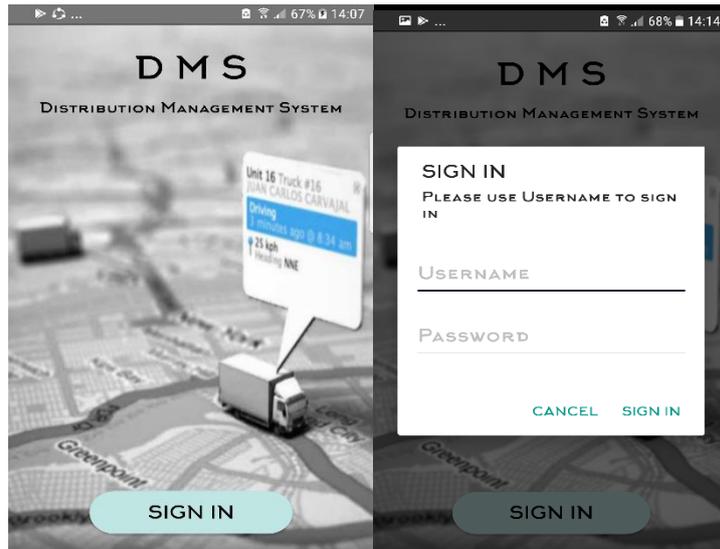


Figure III.15 : L'authentification

### III.6.3 Menu

C'est la première interface qui s'ouvre et contient tous les services qu'offre notre application, et permet la navigation facile entre elles.

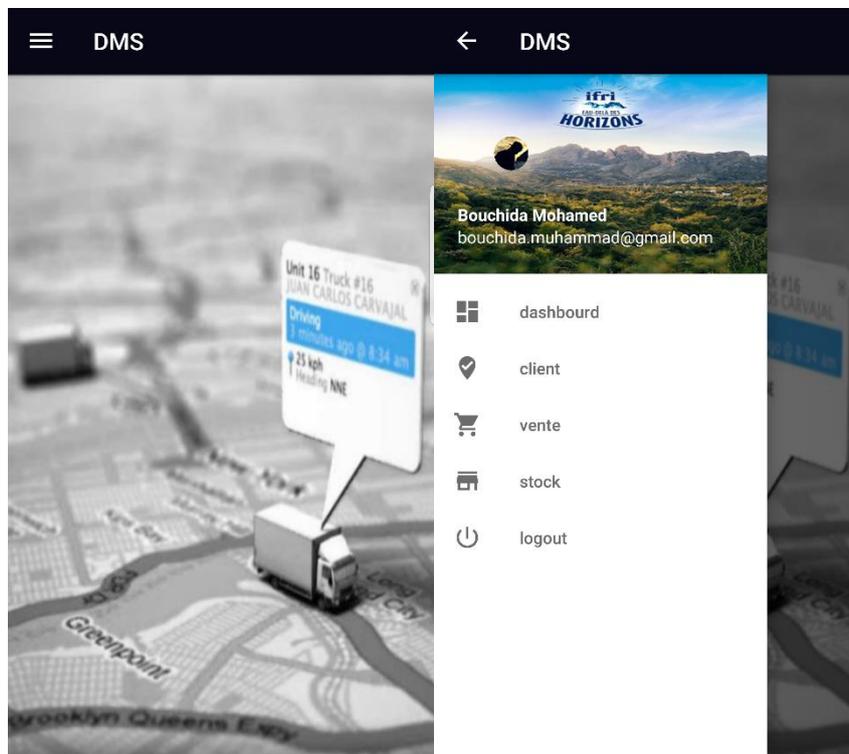


Figure III.16 : Menu

### III.6.4. La liste des clients

Après l'authentification le bouton de synchronisation apparait pour l'importation et la synchronisation en temps réel de la liste définie par l'administrateur et partager à travers une base de donnée qui affiche les noms des clients et adresse

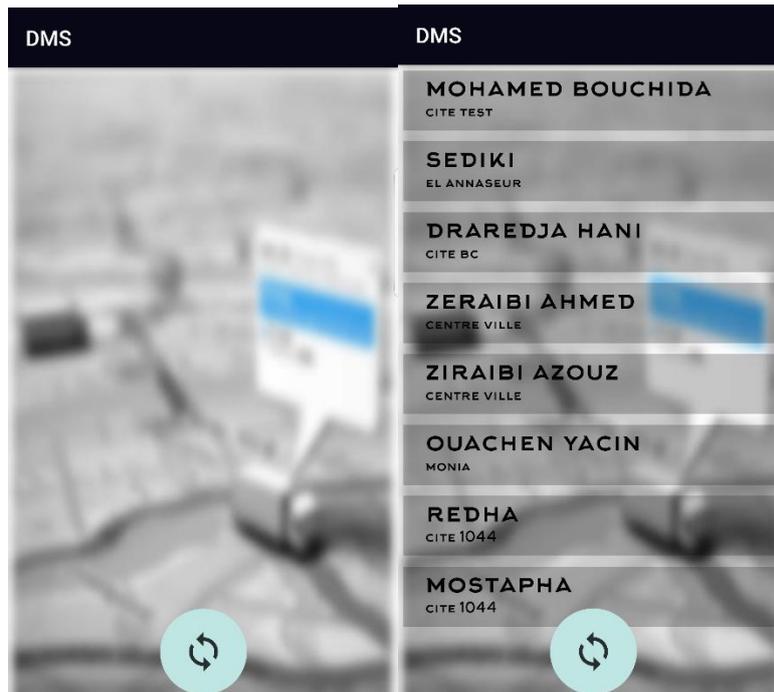


Figure III.17 : La liste des clients

### III.6.5. Itineraire

Après la sélection du client sur la liste une marque affiche la géolocalisation de client sur la carte, notre application offre une fonction de guidage vers le client qui facilite et améliore le trajet en choisissant le plus court chemin avec le bouton itinéraire.

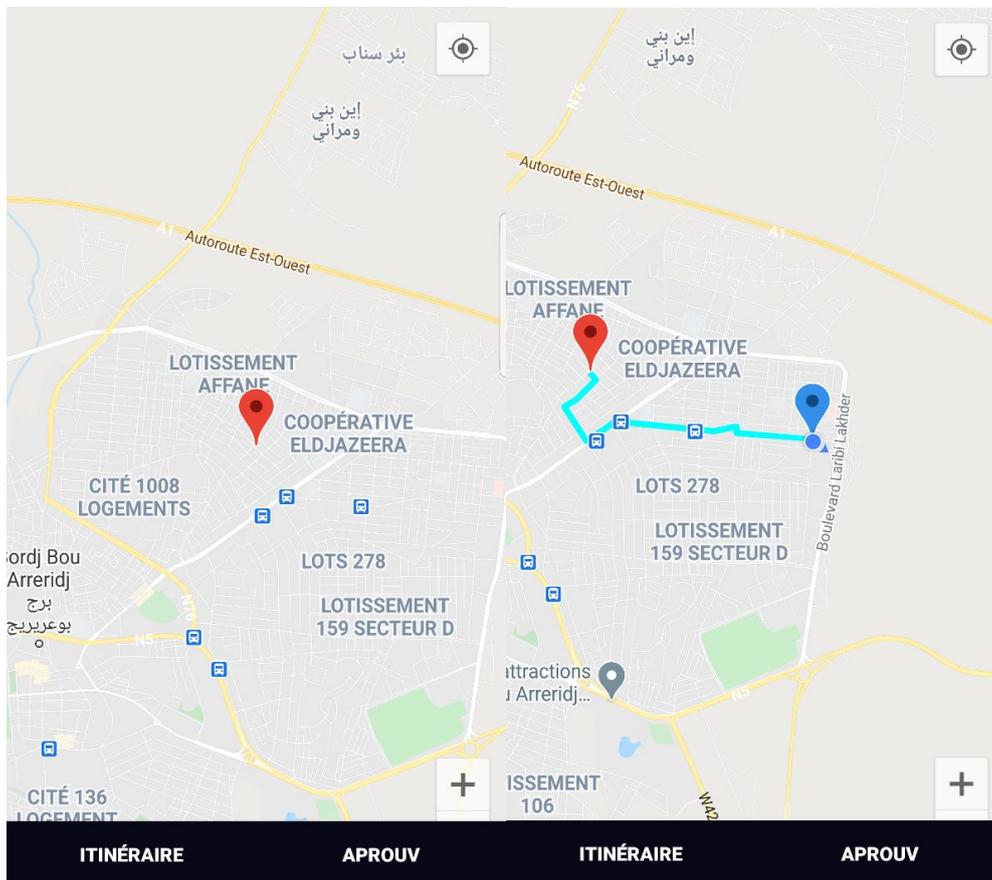


Figure III.18 : fonctionnalité d'itinéraire

### III.6.6. Visite :

Après l'arriver au client en appuie sur visiter pour uploader les informations (date et heure est coordonnées GPS) de la visite vers notre base de donnée pour le suivie par l'admin en temps réel.

### III.6.7 Validation ou confirmation via QR code

Nous avons besoin de codifier notre base de donnée clients et l'afficher sous forme de code QR dans chaque magasin, le livreur doit scanner le QR code pour valider la visite.

Notre application compare le code codifier et scanner du client avec celui afficher dans l'interface, si c'est le même donc notre visite est confirmer et valider par notre application et envoie le rapport vers notre base de donnée en temp reel.



Figure III.19 : Visite

**III.7 Conclusion :**

Dans ce chapitre, nous avons parlé dans un premier lieu de l'objectif de notre application, de la création de la base de données, le fonctionnement de système DMS l'application android ainsi l'intégration du system de validation par QR code et site web.

Et dans un second lieu nous avons parlé de la réalisation de l'application en voyant les outils de développement utilisés, les limitations et les interfaces graphiques de l'application.

# Conclusion générale

Au terme de notre travail, nous espérons avoir atteint l'objectif que nous nous sommes fixé au début, à savoir la réalisation d'un system DMS permettant principalement la distribution et la gestion logistique des camions sur une carte géographique. Nous espérons également que ce system sera bénéfique et rentable au entreprises et ce dans la mesure où elle facilitera les taches de ses utilisateurs en ce qui concerne la bonne gestion et la localisation de ses camions.

Pour concrétiser ce travail, nous avons étudié le fonctionnement théorique des systèmes de localisation et nous avons montré l'utilité et l'efficacité des systèmes GPS. Ce qui permettra dans cette perspective la généralisation de cette technologie en Algérie.

Nous avons présente aussi les fonctionnalités de l'application modélisées avec le langage UML 2.4 que nous avons appris à appliquer durant ce travail. Pour la réalisation de notre system Nous avons choisi (html, css, jQuery, bootstrap, PHP) pour notre site web. Serveur atwebpages, Android studio pour le développement de notre application Android.

Ce travail nous a amené à confirmer qu'avec de la détermination il est possible d'explorer des domaines totalement nouveaux. Un travail qui nous a permis également d'apprendre et de nous initier aux services de géolocalisation qui feront bientôt partie de la vie quotidienne de chacun de nous.

Notre system permet une progression et une amélioration au besoin de l'activité, on peut y intégré un system de gestion de stock dans notre site pour un dépôt, ou et un system de facturation des ventes dans l'application Android.

## **Bibliographie**

-Etude et Simulation d'un Système de Suivi d'un Véhicule basé sur ARDUINO et GPS-GSM module UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU 2016.

-Etude de la géolocalisation dans le réseau GSM Université A. MIRA Bejaïa 2011.

## Webographie

1. Géolocalisation. *Wikipedia*. [En ligne] [Citation : 15 05 2018.]  
<https://fr.wikipedia.org/wiki/G%C3%A9olocalisation>.
2. le principe de fonctionnement du GPS. *Couleur Science*. [En ligne] [Citation : 15 05 2018.]  
<https://couleur-science.eu/?d=2016/06/02/17/54/23-quel-est-le-principe-de-fonctionnement-du-gps>.
3. Implémentation, cartographie et compas. *slideplayer*. [En ligne] [Citation : 15 05 2018.]  
<http://slideplayer.fr/slide/7299289/>.
4. Smartphone. *Wikipedia*. [En ligne] [Citation : 15 05 2018.]  
<https://fr.wikipedia.org/wiki/Smartphone>.
5. Planificateur(trice) logistique/planification et gestion logistique/logistique. [En ligne] [Citation : 15 05 2018.] <http://www.metiers-caoutchouc.fr/Les-metiers/Logistique/Planification-et-gestion-logistique/Planificateur-trice-logistique>.
6. UML (informatique). *Wikipedia*. [En ligne] [Citation : 15 05 2018.]  
[https://fr.wikipedia.org/wiki/UML\\_\(informatique\)#Diagrammes\\_de\\_structure\\_ou\\_diagrammes\\_structurales](https://fr.wikipedia.org/wiki/UML_(informatique)#Diagrammes_de_structure_ou_diagrammes_structurales).
7. Processus de développement en Y(Processus 2TUP). *scribd*. [En ligne] [Citation : 15 05 2018.]  
<https://www.scribd.com/doc/49697489/Processus-de-Developpement-Y-Processus-2TUP>.
8. UML. *developpez.com*. [En ligne] [Citation : 15 05 2018.] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-cas-utilisation>.
9. Conception et réalisation d'un portail web. *memoireonline*. [En ligne] [Citation : 15 05 2018.]  
[https://www.memoireonline.com/11/13/7812/m\\_Conception-et-realisation-d-un-portail-web--l-intention-des-differents-acteurs-burundais13.html](https://www.memoireonline.com/11/13/7812/m_Conception-et-realisation-d-un-portail-web--l-intention-des-differents-acteurs-burundais13.html).
10. HTML. [En ligne] [Citation : 01 06 2018.] [projet.eu.org/pedago/sin/1ere/4-html.odt](http://projet.eu.org/pedago/sin/1ere/4-html.odt).
11. HTML 5 et CSS 3. *Informatique répartie - Développement d'applications*. [En ligne] [Citation : 02 06 2018.] <http://remy-manu.no-ip.biz/HTML5-CSS3/HTML-CSS.html>.
12. Java (langage). *wikipedia*. [En ligne] [Citation : 01 06 2018.]  
[https://fr.wikipedia.org/wiki/Java\\_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage)).
13. JavaScript. *wikipedia*. [En ligne] [Citation : 01 06 2018.]  
<https://fr.wikipedia.org/wiki/JavaScript>.
14. PHP. *wikipedia*. [En ligne] [Citation : 01 06 2018.] <https://fr.wikipedia.org/wiki/PHP>.
15. Structured Query Language. *wikipedia*. [En ligne] [Citation : 01 06 2018.]  
[https://fr.wikipedia.org/wiki/Structured\\_Query\\_Language](https://fr.wikipedia.org/wiki/Structured_Query_Language).

16. XML - Introduction à XML. *CommentCaMarche*. [En ligne]  
<https://www.commentcamarche.com/contents/1332-xml-introduction-a-xml>.
17. Easy Android MySQL Basic CRUD Operation Tutorial for Android Studio Project. *freeprojectcode*. [En ligne] [Citation : 07 02 2018.] <http://freeprojectcode.com/android/android-mysql-basic-crud-operation-tutorial-for-android-studio-project/>.
18. Définition Android. *Définition*. [En ligne] [Citation : 01 06 2018.] <https://cours-informatique-gratuit.fr/dictionnaire/android/>.
19. Définition : SDK. *definitions marketing*. [En ligne] [Citation : 01 06 2018.]  
<https://www.com/definition/sdk/>.
20. Create and manage virtual devices. *developer*. [En ligne] [Citation : 07 02 2018.]  
<https://developer.android.com/studio/run/managing-avds>.
21. Web Services. *CCM*. [En ligne] [Citation : 07 02 2018.]  
<https://www.commentcamarche.com/contents/1244-web-services>.
22. Présentation de JSON. *JSON*. [En ligne] [Citation : 02 06 2018.]  
<http://www.json.org/jsonfr.html>.
23. Android inicio de sesión y registro con PHP, MySQL y SQLite. *EOMblog*. [En ligne] [Citation : 07 02 2018.] <https://eomblog.blogspot.com/2012/10/android-inicio-de-sesion-y-registro-con.html>.
24. Google Maps. *wikipedia*. [En ligne] [Citation : 07 02 2018.]  
[https://fr.wikipedia.org/wiki/Google\\_Maps](https://fr.wikipedia.org/wiki/Google_Maps).
25. Developer Guides. *Developer*. [En ligne] [Citation : 02 02 2018.]  
<https://developer.android.com/guide/>.
26. Geolocation. *wikimedia*. [En ligne] [Citation : 15 05 2018.]  
<https://commons.wikimedia.org/wiki/File:Geolocation.png>.
27. gps tarking. *LAUX d.o.o*. [En ligne] [Citation : 15 05 2018.] <http://laux.si/gps-tracking/>.
28. Définition d'un processus de développement LOGICIEL. *Architecte*. [En ligne] [Citation : 15 05 2018.] <http://ar.21-bal.com/law/1616/index.html?page=4>.
29. Apprendre MySql. *creersonsiteweb*. [En ligne] [Citation : 16 05 2018.]  
<http://creersonsiteweb.net/page-apprendre-mysql>.