

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMARI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT D'INFORMATIQUE

## **Mémoire de Fin d'Etudes de MASTER ACADEMIQUE**

Domaine : **Mathématiques et Informatique**

Filière : **Informatique**

Spécialité : **Conduite de Projet Informatique**

*Présenté par*  
**Meghenez lynda**

### **Thème**

**Développement d'une application JEE pour  
le suivi des véhicules équipés d'un module  
GPS en temps réel**

*Mémoire soutenu publiquement le ...../...../ 20..... devant le jury composé de :*

**Président : M Daoui mohammed**

**Encadreur : M Dib ahmed**

**Co-Encadreur :**

**Examineur : Mme Hadaoui**

**Examineur : M Habet mohammed said**



# Remerciements

Remerciements ... partie très importante puisqu'elle est dédiée à toutes les personnes qui m'ont permis d'en arriver où je suis et parce qu'elle symbolise également la fin d'une expérience et d'une aventure assez spéciale.

D'abord et avant toute chose je remercie le bon DIEU de m'avoir aidé et donné la force, le courage et la patience pour en arriver là.

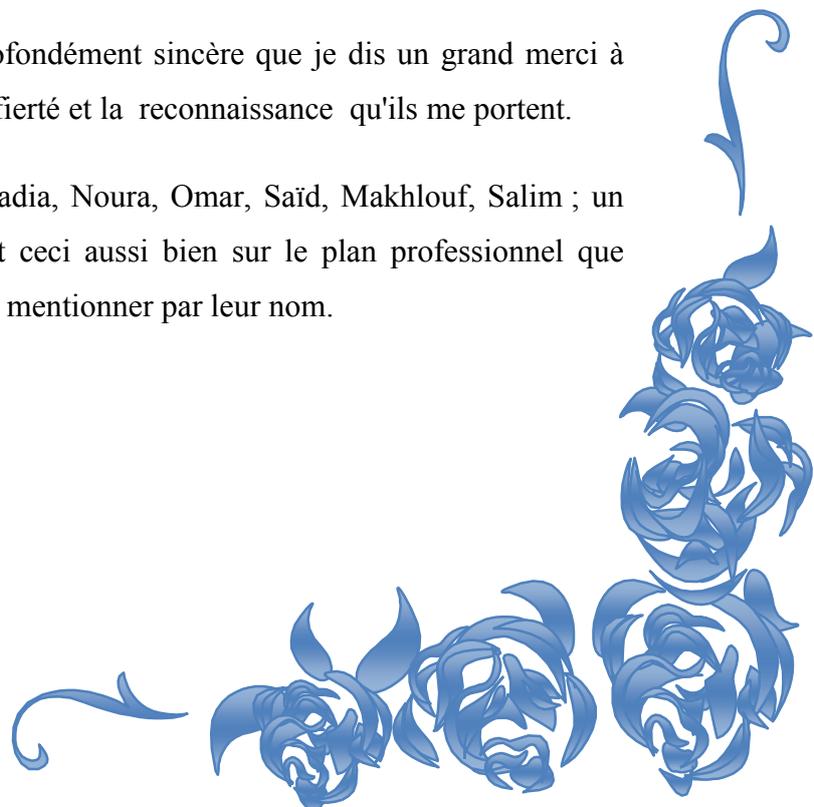
Ensuite, Je ne vais quand même pas repartir de ma petite enfance, même si j'ai une pensée à tous ceux qui m'ont donné le goût d'apprendre. J'ai de très bons souvenirs de l'école, alors je pense que cela aide, mais pour ne remonter qu'à des temps raisonnables, un très grand merci à mon promoteur Mr Dib je tiens à lui exprimer toute ma gratitude.

Un merci à l'ensemble des professeurs qui m'ont suivi tout au long de mon cursus universitaire.

Un remerciement particulier à Mr Karim AGGOUN et Mr HAMZA Hakim pour leurs aides, conseils, soutiens.... Je leur en suis particulièrement reconnaissante. Je tiens bien évidemment à exprimer ma reconnaissance aux membres du jury qui ont accepté de juger mon travail.

C'est d'une manière indubitable et profondément sincère que je dis un grand merci à mon frère et ma famille pour leur soutien, la fierté et la reconnaissance qu'ils me portent.

Petite dédicace à mes amis Akila, Saadia, Noura, Omar, Saïd, Makhoulf, Salim ; un grand merci pour leur aide, présence..... et ceci aussi bien sur le plan professionnel que personnel. Un merci à tous ce que j'ai pas pus mentionner par leur nom.





# Dédicaces

*A la mémoire de mes chers parents,*

*Que nulle dédicace ne puisse exprimer ce que je leurs dois, Trésors de bonté, de générosité et de tendresse, en témoignage de mon profond amour et ma grande reconnaissance « Que Dieu vous accueille dans son vaste paradis ».*

*A mes chers Nana et Dada,*

*En témoignage de mes sincères reconnaissances pour les efforts qu'ils ont consenti pour l'accomplissement de mes études. Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie.*

*A ma chère Soussou,*

*En témoignage de mon profond et sincère remerciement pour sa présence à mes côtés durant les moments les plus durs de mon existence, je lui dédie ce modeste travail en témoignage de mon grand amour.*

*A mes chers Amis,*

*Pour leur aide et leur soutien moral durant l'élaboration de mon travail.*

*Et en particulier à Karim, Abd Anour, Hakim, Yazid, Nassima, Noura, Salim et sa famille et surtout Akila merci pour tout.*

*A tous ceux dont l'oubli du nom n'est guère celui du cœur...*

*Lynda*



# Table des matières

I.	Introduction générale.....	1
1.	Introduction .....	2
II.	Chapitre I : Etat de l’art sur la géolocalisation	
1.	Introduction .....	3
2.	Méthodes et techniques de géolocalisation .....	3
2.1.	Méthodes géométriques .....	3
2.1.1.	Triangulation.....	3
2.1.2.	Trilatération .....	4
2.2.	Méthodes statiques.....	5
2.2.1.	Empreinte radio.....	5
2.3.	Méthodes Hybrides .....	5
2.4.	Techniques de mesure .....	6
2.4.1.	Localisation par ‘ puissance du signal reçu’ .....	6
2.4.2.	Localisation par ‘angles d’arrivée’ .....	6
2.4.3.	Localisation par ‘temps d’arrivée’ .....	7
2.4.4.	Localisation par ‘différentiel d’arrivée’ .....	7
2.4.5.	Localisation par ‘ la fréquence d’arrivée’ .....	8
2.4.6.	Localisation par ‘cellule d’origine’ .....	8
2.4.7.	Localisation par ‘ modélisation’ .....	8
3.	Systèmes de géolocalisation .....	8
3.1.	Le NAVSTAR/GPS américain .....	8
3.1.1.	Origine du GPS et description du système.....	8
3.1.2.	Constitution du système .....	9
a-	Segment spatial .....	10
b-	Segment de contrôle .....	11
c-	Segment utilisateur .....	12
3.1.3.	Structure du signal .....	13
3.1.4.	Fonctionnement du GPS.....	13
3.1.5.	Sources d’erreurs .....	15
3.1.6.	Points forts du GPS.....	16
3.1.7.	Points faibles du GPS .....	16
3.1.8.	Améliorations du GPS .....	17

3.1.8.1. Le Différentiel GPS.....	17
3.1.8.2. Le A-GPS.....	18
3.1.9. Systèmes concurrents au GPS .....	19
3.1.9.1. GLONASS .....	19
3.1.9.2. EGNOS .....	19
3.1.9.3. Galileo .....	19
3.1.9.4. Beidou .....	20
3.2. Wifi Positioning System (WPS) .....	20
4. Techniques de géolocalisation .....	21
a. Localisation par satellite .....	21
b. Localisation par GSM.....	21
c. Localisation par WI-FI.....	22
d. Localisation par adresse IP (sur Internet).....	22
e. Localisation par RFID.....	22
f. Localisation hybride.....	23
5. Applications de géolocalisation.....	24
6. Conclusion .....	25
III. Chapitre II : les services mobiles	
1. Introduction .....	26
2. Les terminaux mobiles .....	26
2.1. Les téléphones mobiles .....	26
2.1.1. La téléphonie mobile 1 G.....	26
2.1.2. La téléphonie mobile 2G .....	27
2.1.3. La téléphonie mobile 2.5G.....	27
2.1.4. La téléphonie mobile 2.75G .....	28
2.1.5. La téléphonie mobile 3G .....	28
2.1.6. La téléphonie mobile 4G .....	29
2.2. Les ordinateurs portables .....	29
2.3. Les terminaux PDA .....	29
2.4. Les Tablettes tactiles .....	30
2.5. Smartphone.....	30
2.5.1. Historique et évolution.....	30
2.5.2. Système d'exploitation mobile.....	31
3. Les services mobiles .....	32
3.1. Le service de messagerie .....	33

3.2.La visiophonie.....	33
3.3.Le service de géolocalisation .....	34
3.3.1. Types de services de géolocalisation.....	35
3.3.2. Positionnement de personnes et d'appareils .....	35
3.3.3. Services différents selon la position .....	36
4. Architecture d'une plateforme logicielle de géolocalisation .....	36
4.1. Plateformes logicielles de géolocalisation .....	36
4.2.Composants d'une plateforme logicielle de géolocalisation .....	36
4.3. Transmission des informations .....	38
4.4. Types de terminaux existants .....	40
4.5. Fonctionnalités offertes par les plateformes de géolocalisation .....	41
5. Les services mobiles pour entreprise.....	41
5.1.Caractéristiques des applications pour entreprises .....	42
6. Conclusion .....	42

### Chapitre III : Analyse et conception

1. Introduction .....	43
2. Présentation d'UML.....	43
2.1. Définition .....	43
2.2.Modélisation avec UML.....	43
2.3.La démarche de modélisation avec l'UML.....	44
3. Analyse .....	44
3.1.Spécification et analyse des besoins .....	44
3.1.1. Objectif du système .....	45
3.1.2. Identification des acteurs .....	45
3.1.3. Diagramme de contexte .....	47
3.2.Spécification des tâches .....	48
3.2.1. Définition .....	48
3.3.Spécification des scénarios .....	49
3.3.1. Définition .....	49
3.4.Diagrammes des cas d'utilisation .....	51
3.4.1. Définition .....	51
3.4.2. Au niveau de l'application Web.....	51
3.4.3. Au niveau de l'application mobile .....	54
3.5.Description des cas d'utilisation .....	54

3.5.1.	Au niveau de l'application web .....	54
3.5.2.	Au niveau de l'application mobile.....	63
4.	Conception .....	64
4.1.	Les diagrammes de séquences .....	64
4.2.	Diagrammes de classes .....	69
5.	Le modèle conceptuel des données .....	73
6.	Le modèle logique des données .....	74
7.	Conclusion .....	81
IV.	Chapitre IV : Implémentation et mise en œuvre	
1.	Introduction .....	82
2.	Architecture générale de l'application .....	82
3.	Choix des outils et technologies utilisés .....	83
3.1.	Langage de programmation .....	83
3.2.	Plateforme mobile .....	83
4.	Environnement et outils de développements.....	84
4.1.	Environnement matériel .....	84
4.2.	Partie mobile .....	84
4.2.1.	Environnement cible .....	84
4.2.2.	Environnement logiciel .....	85
4.2.3.	Implémentation de l'application.....	89
4.2.4.	Fonctionnement de l'application .....	92
4.3.	Partie Web.....	94
4.3.1.	Présentation de l'application .....	95
4.3.2.	Présentation du modèle MVC .....	97
4.3.3.	Langages de programmation .....	98
4.3.4.	Les Framework .....	99
4.3.5.	Outils de développement .....	101
4.4.	Implémentation des différentes parties de l'application.....	103
4.4.1.	Partie stockage .....	103
4.4.2.	Partie traitement.....	103
4.4.3.	Partie présentation .....	105
4.5.	Interfaces de l'application Web.....	105
4.5.1.	Interfaces communes.....	105
4.5.2.	Interfaces Administrateur.....	109

4.5.3. Interfaces client.....	112
5. Conclusion .....	120
V. Conclusion générale	
VI. Annexe A	
VII. Annexe B	

## Listes des figures :

Figure 1.1. Triangulation .....	3
Figure 1.2. Trilatération .....	4
Figure 1.3. Localisation par ‘ puissance du signal’ .....	6
Figure 1.4. Localisation par ‘ Angle d’arrivée ‘ .....	7
Figure 1.5. Localisation par ‘ temps d’arrivée’ .....	7
Figure 1.6. Constellation de satellites .....	10
Figure 1.7. Segment de contrôle .....	12
Figure 1.8. Segment utilisateur .....	12
Figure 1.9. Principe de localisation par satellite GPS .....	14
Figure 1.10. Géolocalisation en utilisant le Différentiel GPS.....	15
Figure 1.11. Principe de localisation avec A-GPS.....	16
Figure 2.1. Architecture d’une plateforme logicielle de géolocalisation.....	34
Figure 2.2. Architecture d’un système de géolocalisation GPS avec remontée des données via le réseau GSM/GPRS.....	35
Figure 2.3. architecture d’un système de géolocalisation GPS avec remontée des données via le réseau satellitaire .....	36
Figure 3.1. La démarche adoptée pour la modélisation .....	41
Figure 3.2. Diagramme de contexte.....	44
Figure 3.3. Diagramme de cas d’utilisation : acteur « visiteur ».....	49
Figure3.4. diagramme de cas d’utilisation : acteur « administrateur ».....	49
Figure3.5. diagramme de cas d’utilisation : acteur « user».....	50
Figure 3.6. Diagramme de cas d’utilisation : acteur « capteur de position».....	51
Figure3.7. diagramme de séquence de cas d’utilisation « authentification ».....	61
Figure3.8. diagramme de séquence de cas d’utilisation « inscription ».....	62
Figure3.9. diagramme de séquence de cas d’utilisation « ajouter véhicule ».....	62
Figure3.10. diagramme de séquence de cas d’utilisation « localiser un véhicule sur la MAP ».....	63

Figure 3.11. Diagramme de séquence de cas d'utilisation « consulter historique ».....	63
Figure 3.12. Diagramme de séquence de cas d'utilisation « paramétrer alerte ».....	64
Figure 3.13. Diagramme de séquence de cas d'utilisation « valider inscription ».....	64
Figure 3.14. Diagramme de séquence de cas d'utilisation « envoi des données ».....	65
Figure 3.15. diagramme de classe générale du cas d'utilisation « authentification ».....	67
Figure 3.16. Diagramme de classe générale du cas d'utilisation « inscription ».....	68
Figure 3.17. Diagramme de classe générale du cas d'utilisation « Ajouter véhicule » .....	68
Figure 3.18. Diagramme de classe générale du cas d'utilisation « localiser véhicule sur la MAP ».....	69
Figure 3.19. Diagramme de classe générale du cas d'utilisation « envoi des données » .....	69
Figure 3.21. Modèle conceptuel de données .....	70
Figure 4.1. Architecture générale de l'application.....	81
Figure 4.1. Interface principale d'Eclipse .....	84
Figure 4.2. Choix de la version SDK.....	85
Figure 4.3. Installation des Packages.....	86
Figure 4.4. Effectuer le choix lors du lancement .....	87
Figure 4.5. Ecran d'accueil de l'application mobile .....	92
Figure 4.6. Architecture JEE .....	93
Figure 4.7. modèle MVC .....	94
Figure 4.8. Interface « accueil ».....	105
Figure 4.9. Interface « Fonctionnalités du système ».....	106
Figure 4.10. Interface « Inscription ».....	107
Figure 4.11. Interface « authentification ».....	107
Figure 4.12. Interface « Espace administrateur ».....	108
Figure 4.13. Interface « Ajouter marque véhicule ».....	109
Figure 4.14. Interface « gestion utilisateurs ».....	110
Figure 4.15. Interface « Espace Client ».....	111

Figure4.16. Interface « Ajouter véhicule ».....	112
Figure 4.17. Interface « Ajouter chauffeur ».....	113
Figure4.18. Interface « consulter historique » .....	114
Figure 4.19. Interface « consulter alerte ».....	115
Figure 4.20. Interface « affecter véhicule/Chauffeur ».....	116
Figure 4.21. Interface « MAP tracer ».....	117
Figure4.22. Interface « MAP position ».....	118

## Liste des tableaux :

Tableau 2.1. Comparaison des différentes techniques de géolocalisation.....	23
Tableau 3.1. Spécification des tâches .....	48
Tableau 3.2. Spécification des scénarios .....	49
Tableau 3.3. Description des cas d'utilisation « inscription ».....	54
Tableau 3.4. Description des cas d'utilisation « authentification ».....	55
Tableau 3.5. Description des cas d'utilisation « localiser véhicule sur la Map en temps réel ».....	56
Tableau 3.6. Description des cas d'utilisation « Ajouter véhicule ».....	56
Tableau 3.7. Description des cas d'utilisation « Ajouter chauffeur ».....	57
Tableau 3.8. Description des cas d'utilisation « Ajouter Zone ».....	57
Tableau 3.9. Description des cas d'utilisation « paramétrer alerte ».....	58
Tableau 3.10. Description des cas d'utilisation « affecter paramètre/véhicule ».....	59
Tableau 3.11. Description des cas d'utilisation « affecter zone /véhicule ».....	59
Tableau 3.12. Description des cas d'utilisation « affecter chauffeur/véhicule ».....	60
Tableau 3.13. Description des cas d'utilisation « consulter historique ».....	60
Tableau 3.14. Description des cas d'utilisation « ajouter marque véhicule » .....	61
Tableau 3.15. Description des cas d'utilisation « type marque véhicule ».....	61
Tableau 3.16. Description des cas d'utilisation « valider inscription ».....	62
Tableau 3.17. Description des cas d'utilisation « envoi des données ».....	63

# ***Introduction Générale***

# INTRODUCTION GENERALE

---

## 1. Introduction :

L'évolution des technologies et moyens de communications à ouvert un champ immense dans les services et fonctionnalités qu'offre les appareils que l'on utilise au quotidien. Aujourd'hui le téléphone portable connaît une avancée sans précédent. On l'appel Smartphone, cet appareil qui est à la fois téléphone et télécopie, baladeur, appareil photo et caméscope, console de jeu, réveil et agenda, terminal internet, balise GPS. Chacun des ces appareils existe mais seul le Smartphone semble en faire la synthèse. Le Smartphone a révolutionné le monde et la façon avec laquelle nous nous déplaçons. Pratiquement tout les Smartphones et appareils embarqués disposent d'une puce GPS intégré afin de localiser l'appareil et son utilisateur. Le service de géolocalisation offre diverses possibilités dans la localisation et positionnement sur carte des utilisateurs et de leurs biens dans différents domaines.

La géolocalisation au sein des entreprises est en plein essor, des entreprises qui possèdent un parc automobiles ou leurs activités fait appel a plusieurs véhicules sont de plus en plus intéressé par la gestion automatique de leurs flotte en temps réel. Ces entreprises rencontrent des problèmes liées a l'optimisation et la rentabilité de leur activité, savoir ou sont ses véhicules, dans qu'elles états sont t'ils, sont des enjeux crucial pour l'entreprise.

Un système de géolocalisation d'une flotte de véhicules se compose de deux principaux éléments :- l'ordinateur de bord qui est l'appareil embarquée au sein du véhicule qui permet la récupération des coordonnées géographiques du véhicule. Ainsi que l'outil ou logiciel qui permet la gestion de la flotte en exploitant les données collectées depuis les véhicules.

Le système de positionnement par satellite GPS est le système le plus répandu à l'heure actuelle, avec sa couverture mondiale et sa précision qui est de quelques mètres c'est le système le plus utilisé en extérieur. Mais d'autres systèmes existe qui peuvent être des concurrents ou juste complémentaires, car les signaux émis par les satellites ne peuvent pas traversés les murs ce qui rend le GPS non fonctionnel à l'intérieur des bâtiments.

Les réseaux sans fils connaissent eux aussi une évolution fulgurante, avec la révolution de la norme GPRS, les débits des réseaux sans fils ne cessent d'augmenter, le transfert des données se fait de manière plus rapide et plus facile.

Le présent thème consiste à développer une plateforme logicielle, de gestion et suivi d'une flotte de véhicules en temps réel, comme invoquée précédemment une solution de géolocalisation d'une flotte de véhicule comporte un outil de récupération des coordonnées du véhicule, a cet effet on utilisera un Smartphone embarquée au sein du véhicule qui utilisera le système GPS pour la récupération des coordonnées géographiques et le réseau GSM/GPRS pour la communication de ces informations à la plateforme web qui permet l'exploitation de ces informations en les affichant sur la map.

Le présent rapport est structuré comme suit :

D'abord, le rapport commence par une introduction générale, puis ;

# INTRODUCTION GENERALE

---

- Le premier chapitre s'intitule « Etat de l'art sur la géolocalisation », qui consiste en une présentation des différentes méthodes, systèmes et technologies de géolocalisation.
- Le deuxième chapitre s'intitule « les services mobiles », dont nous avons présentés l'évolution des terminaux et réseaux mobiles. les différentes notions relatives aux services géolocalisés sont également introduites pour mettre en évidence la notion de plateforme logicielle de géolocalisation et plus précisément la localisation d'un véhicule sur route.
- Le troisième chapitre sous le nom « Analyse et Conception », est consacré à l'analyse et la conception de notre système à travers son architecture globale puis détaillée en définissant les différentes fonctionnalités attendues du système. Et pour permettre de détailler toutes les étapes du travail, une modélisation est faite avec le langage UML.
- Le quatrième et le dernier chapitre intitulé « Réalisation et mise on œuvre », comporte quant à lui la présentation de l'environnement dont lequel notre application a été réalisée, les outils utilisés et quelques interfaces de celle-ci.

Enfin, le rapport se termine par une conclusion générale qui résume le travail réalisé et présente les perspectives pour améliorer le projet.

## 1.1. Objectifs du projet :

La localisation en temps réel permet d'avoir la maîtrise sur les véhicules en mission, en ayant un moyen de suivi et de communication entre les différents intervenants.

L'objectif global d'un système de gestion et suivi de flotte est d'offrir à l'entreprise en question les fonctionnalités suivantes :

- Suivi des déplacements d'un véhicule en temps réel ;
- Consultation de l'historique des déplacements d'un véhicule ;
- Création de zones d'intérêt pour la configuration d'alertes également ;
- Calcul d'itinéraires à emprunter par un véhicule pour l'optimisation de ses parcours ;
- Configuration d'alertes (entrée/sortie de zone d'intérêt, dépassement de vitesse, atteinte d'un point cible ....) ;

CHAPITRE

I

*Etat de l'art sur la  
géolocalisation*

## I. Introduction :

La géolocalisation est une technologie avancée qui permet de collecter des informations afin de localiser un objet ou une personne sur une carte, à l'aide de ses coordonnées géographiques. Créer pour des besoins militaires de l'armée américaine, la géolocalisation a servie tout d'abord pour la localisation d'objets ou de personnes, ces dernières années son utilisation a pris de l'ampleur dans différents domaines. Différents systèmes, méthodes et techniques ont été développés.

Dans ce qui suit-on donnera un aperçu sur les différentes méthodes et techniques de géolocalisation.

## II. Méthodes et techniques de géolocalisation : [8]

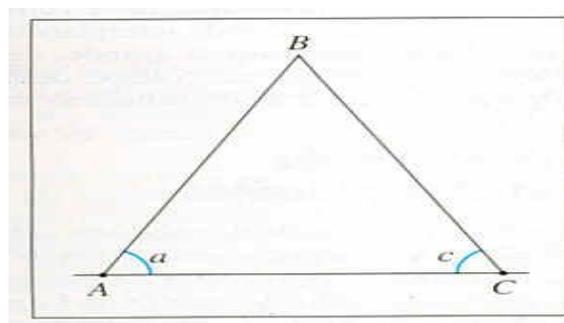
### II.1. Méthodes géométriques :

Trois techniques mathématiques sont généralement utilisées pour le calcul de la position d'un récepteur de signaux reçus de plusieurs émetteurs : triangulation, trilatération et multilatération. Les deux dernières sont très semblables, mais ne devrait pas être confondu.

#### II.1.1. Triangulation :

La triangulation est une technique très ancienne, elle permet à un observateur de calculer sa position en mesurant deux directions vers deux points de référence. Etant donné que les positions des points de référence sont connues, il est donc possible de construire un triangle dont l'un des côtés et deux des angles sont connus, à l'observateur du troisième point. Cela est suffisant pour construire complètement le triangle et ainsi définir la position de l'observateur.

Pour mieux expliquer, prenons l'exemple d'un triangle ABC, dont on connaît les angles  $a$  et  $c$  ainsi que la distance AB et on cherche le point B.



**Figure1. 1** : Triangulation

Il est simple de calculer l'angle  $b$ , sachant que la somme des sommets d'un triangle est égale à  $180^\circ$ , il suffit de faire  $180-(a+c)$ .

Ensuite, c'est la loi des sinus que nous utilisons :

$$\frac{AC}{\sin b} = \frac{AB}{\sin c} = \frac{BC}{\sin a}$$

Ainsi, grâce à deux produits en croix, nous connaissons les distances AB et BC. Alors on peut trouver B.

### II.1.2. Trilatération :

La trilatération consiste à utiliser uniquement les distances pour localiser l'objet cherché. Ce principe utilise le plus souvent le système de signal radio. Il se base sur la connaissance de la vitesse de propagation de l'onde.

Les points autour de celui que l'on cherche à connaître émettent une onde. A partir de ce moment, un chronomètre se déclenche. Lorsque l'objet inconnu reçoit l'onde, il la renvoi vers son émetteur. Quand celui la capte, il stoppe le chronomètre. Le calcul devient alors simple (en théorie) :

$$v = \frac{d}{t} \Leftrightarrow d = v * t$$

Or c'est un aller-retour que l'onde fait, il suffit donc de diviser la distance obtenue et nous obtenons la distance entre le point inconnu et l'émetteur.

Une fois les distances connues, il suffit de tracer des cercles de centres des points connus et de rayons égaux à la distance propre de ce chaque point à l'objet inconnu. Le point d'intersection de ces cercles indique l'endroit où se situe le point recherché.

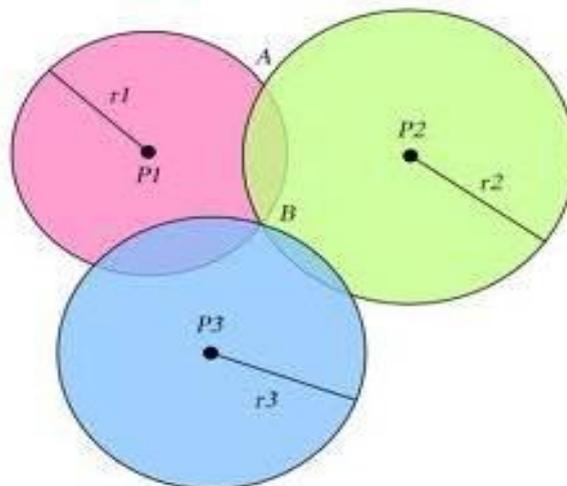


Figure 1.2 : Trilatération

## **II.2. Méthodes statistiques :**

Ce sont les méthodes qui consistent à effectuer des mesures « références » puis à comparer les mesures réalisées pour l'objet mobile(OM) avec ces mesures de références pour en déduire sa position. Ces approches comprennent donc deux étapes: une étape de calibrage ou d'apprentissage et l'étape d'estimation de la position. Une des méthodes les plus efficaces est la méthode d'empreinte radio (en anglais RF fingerprinting), qui se base sur les mesures des puissances reçues (RSSI pour Received Signal Strength Indicator) par l'OM à partir de balises radios (comme les points d'accès Wifi par exemple).

### **II.2.1. Empreintes radio :**

La méthode d'empreinte radio se fait en deux phases : «Formation» (calibrage ou apprentissage) et «Positionnement» (localisation) (appelées en anglais «Training» et «Positioning» ou encore «Off-line training phase » et «Real-time or Online location determination phase»).

Les besoins de calibrage et de mise à jour de la base de données sont les inconvénients de cette approche.

Durant la phase de formation, des mesures de RSS sont prises : des balises radios sont installées sur le site désiré, des points échantillons (des positions sur le site) sont choisis et les RSS mesurés et sauvegardés dans une base de données.

Pendant la phase de positionnement, la position de l'OM est déterminée en comparant les RSS mesurés par l'OM avec les RSS sauvegardés dans la base de données (appelée carte radio ou map). Plus le nombre de balises radios augmente, plus la précision s'améliore. Plusieurs mesures successives de puissance doivent être réalisées pour une meilleure évaluation de la position, les ondes radios étant affectées par plusieurs facteurs.

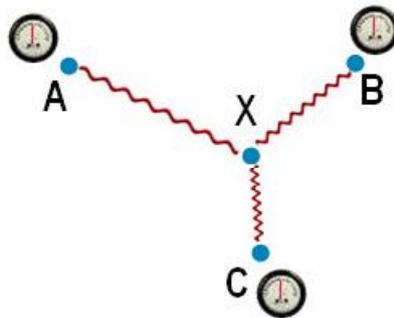
## **II.3. Méthodes hybrides :**

Des méthodes hybrides ont été proposées. Elles comportent deux étapes : dans la première étape, elles utilisent la méthode d'empreinte radio avec une phase rapide de formation (ceci indique qu'on a peu de PR et que la base de données est très petite) pour obtenir une première évaluation de positionnement (indiquant par exemple dans quelle pièce du bâtiment se trouve l'OM), et dans la deuxième étape, un modèle empirique précis de la propagation du signal sera utilisé pour calculer la distance exacte entre émetteur et récepteur ; la triangulation pourra alors être utilisée pour calculer la position de l'OM plus précisément.

## II.4. Techniques de mesure :

### II.4.1. Localisation par ' Puissance du Signal Reçu' (RSS pour Received Signal Strength) :

La localisation par estimation de la puissance du signal reçu est très répandue, notamment dans les réseaux sans fils. Elle se base sur une relation entre la puissance reçue et la position de la source. C'est la méthode la plus utilisée en Wifi, les techniques de localisation par RSS sont généralement robustes et peu chères mais elles sont souvent d'une précision limitée.

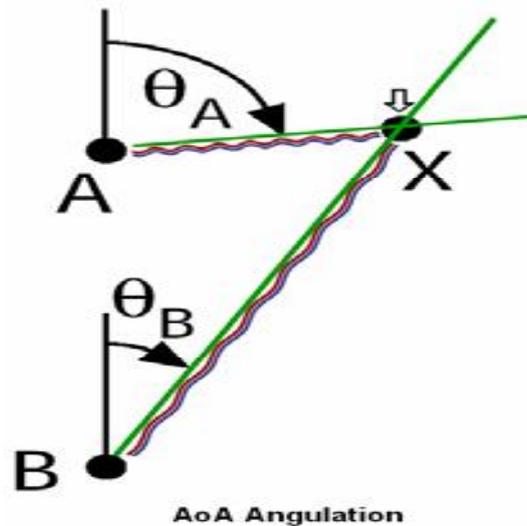


**Figure 1.3 :** Localisation par ' puissance du signal'

### II.4.2. Localisation par 'Angles d'Arrivée' (Angle of Arrival) :

Les systèmes de localisation basés sur l'estimation des AOAs sont relativement complexes et coûteux. Les systèmes dédiés sont en effet par nature multi capteurs et requièrent une calibration régulière de ceux-ci. Le principe de localisation par AOA est aussi couramment appelé triangulation. La direction d'arrivée dans le plan est liée géométriquement à la position.

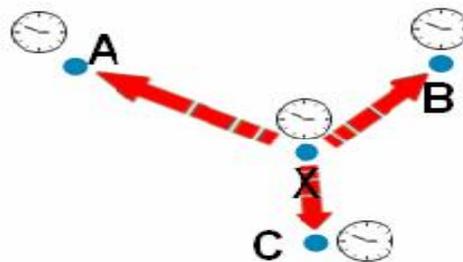
C'est la technique utilisée par les radars aériens, elle consiste à calculer l'angle de réception d'un signal par 2 ou 3 radars, et, en utilisant cette information, positionner l'objet dans l'espace. Elle est très précise, mais demande des antennes motorisées pour déterminer l'angle de réception.



**Figure 1. 4 :** Localisation par 'Angle d'arrivée'

#### II.4.3. Localisation par 'temps d'Arrivée' (Time Of Arrival) :

Dans cette technique on se base sur le temps de propagation du signal. La source envoie un signal daté qui sera reçu par les récepteurs. Ces derniers vont dater l'heure d'arrivée du signal. La mesure du temps d'arrivée nous permet d'obtenir une information sur la distance à laquelle est situé l'émetteur. Ceci requiert une synchronisation précise entre la source et les récepteurs.



**Figure 1.5 :** Localisation par ' temps d'arrivée'

#### II.4.4. Localisation par 'Différentiel d'arrivée' (TDOA - Time Difference Of Arrival) :

Le principe est similaire, mais en utilisant une source quelconque qui ne date pas son émission. Le signal est alors reçu par les récepteurs et le système de géo localisation va déterminer la position de la source en fonction de la différence du temps d'arrivée. Cette technologie est utilisée par différents systèmes de géo localisation par Wifi mais elle demande que les points d'accès radio possèdent une horloge très précise et synchronisée. Une très faible désynchronisation peut entraîner une erreur de positionnement de plusieurs dizaines de mètres.

Cette solution est bien adaptée dans les environnements ouverts où le signal est plus facilement. Elle peut connaître des limites en intérieur, le signal pouvant rebondir sur les murs avant d'atteindre un récepteur.

#### **II.4.5. Localisation par la fréquence d'arrivée (FOA pour Frequency-of-arrival) ou la fréquence différentielle d'arrivée (FDOA pour Frequency-difference-of-arrival) :**

Lorsqu'une source ou une station se déplace elle induit un décalage fréquentiel en vertu de l'effet Doppler. La mesure de ce décalage fréquentiel permet alors de remonter à la position et à la vitesse de la source. La mesure du FDOA ne nécessite plus la connaissance de la fréquence centrale des signaux émis. Le choix des paramètres de localisation dépend essentiellement du contexte et des ressources matérielles disponibles.

#### **II.4.6. Localisation par 'Cellule d'origine' :**

C'est la plus simple et la moins précise, elle consiste simplement à donner la localisation de l'objet et fonction de la borne radio sur laquelle il est associé. Sachant qu'une borne radio peut couvrir plusieurs centaines de mètres, la précision est très faible.

#### **II.4.7. Localisation par « Modélisation » (Patterning) :**

Cette autre méthode est aussi utilisée en géo localisation Wifi. Elle consiste à modéliser l'ensemble des lieux où l'on veut avoir de la géo localisation en faisant une série de calibration de chaque point de localisation. Le système va alors pour chaque mesure de la source rechercher dans son modèle le point le plus « proche » de la mesure et positionner la source sur le point.

Cette solution apporte de bons résultats mais demande un gros travail de modélisation.

### **III. Systèmes de géolocalisation :**

#### **III.1. Le NAVSTAR/GPS américain : [2][1]**

##### **III.1.1. Origine du GPS et description du système :**

Connaitre sa position a été et demeure une des préoccupations de l'être humain, l'idée du GPS vient de ce besoin que l'homme moderne après avoir eu les étoiles comme référentiel pour s'orienter et parcourir la terre , a exprimé afin de créer un système de référentiel sous son contrôle. Ainsi, sont apparus les systèmes dits GNSS (Global Navigation Satellite System) dont le principe est de se localiser moyennant un système de satellites dans un référentiel terrestre. Les systèmes GNSS remplissent également un rôle important de datation précise, qui profite à tout utilisateur muni d'un récepteur.

Il existe actuellement plusieurs systèmes candidats pour remplir les fonctions de positionnement et de datation du GNSS, cependant Le GPS (Global Positioning System) s'est imposé face à ses rivaux compte tenu des difficultés économiques de ces derniers.

Le GPS est actuellement le plus utilisé et le plus médiatisé des systèmes de localisation par satellite. Développé à partir des années 70, Le GPS est géré conjointement par le Ministère de la Défense (DoD) et le Ministre des Transports (DoT) américains. Il a été déclaré pleinement opérationnel par l'U.S Air Force Space Command (AFSC) officiellement le 27 avril 1995.

Le GPS fournit aux utilisateurs civils et militaires deux services pour la localisation en mode absolu et le transfert de temps :

- Le **SPS** (Standard Positioning Service) réalisé avec le C/A-code, destiné aux utilisateurs civils, qui avait été spécifié à l'origine pour fournir une précision sur la localisation absolue de 100 mètres (95%) en horizontal et de 156 mètres (95%) en vertical et un transfert du temps UTC à moins de 340 nanosecondes (95%);
- Le **PPS** (Precise Positioning Service), réalisé avec le P-code est une version plus précise du SPS destinée aux utilisateurs militaires et civils habilités. L'accès au service PPS est contrôlé par le Ministère de la Défense américain (US DoD).

-  
La dégradation volontaire SA (Selective Availability) qui diminuait la précision du service SPS en introduisant des erreurs sur les paramètres d'horloges et d'éphémérides diffusés par les satellites a été désactivée par décision des autorités américaines en mai 2000. La précision théorique actuelle du SPS est de 13 mètres pour le positionnement horizontal et 22 mètres en vertical.

Le système GPS permet donc de se positionner sur la terre en fournissant à l'utilisateur des coordonnées. Celles-ci peuvent être de deux types :

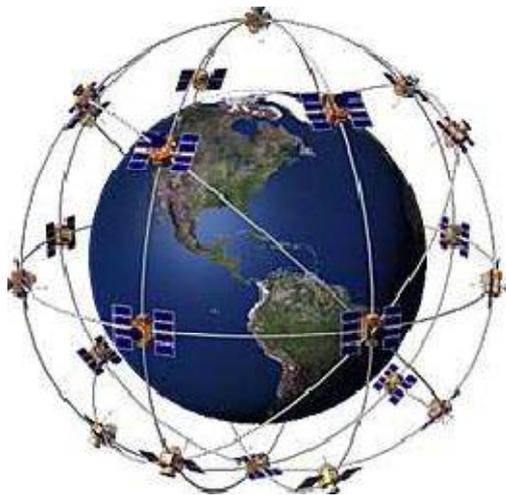
1. soit les coordonnées géographiques WGS84 (c'est-à-dire longitude, latitude et hauteur ellipsoïdale -  $\lambda$ ,  $\phi$ ,  $h$ ) qui sont liées à ce référentiel géodésique.
2. soit les coordonnées planes ou rectangulaires (X, Y) qui sont indissociables des systèmes de projection plane.

### **III.1.2. Constitution du système GPS :**

Dans le langage courant, on utilise le mot « GPS » pour désigner le récepteur. Or «GPS» désigne le système de positionnement en entier, et ce système se compose de trois éléments : le segment spatial, le segment contrôle et le segment utilisateur

**a. Le segment spatial :**

Le segment spatial est composé d'un réseau de 28 satellites (dont trois de secours) en orbite quasi-circulaire autour de la terre, à une altitude à peu près de 20 200 km pour un satellite situé au Zénith et 25 60 km pour un satellite situé au ras de l'horizon. Les satellites sont répartis sur six orbites ayant chacune quatre satellites répartis. Les orbites sont inclinées de  $55^\circ$  par rapport à l'équateur et séparées d'environ  $60^\circ$  sur l'équateur, presque circulaires de 26600 km de rayon. Un satellite décrit une orbite en une demi-journée sidérale soit 11h58m02s en temps UTC. Un observateur terrestre qui observe un satellite en particulier, le voit toutes les 23h56m04s UTC au même endroit du fait de la différence entre la période de rotation des satellites et de la terre. Les satellites sont positionnés de manière à ce qu'un minimum de six soit visible. En limitant l'angle de visibilité à  $15^\circ$  au-dessus de l'horizon (angle d'élévation), l'utilisateur dispose d'un minimum de quatre satellites à tout moment et quel que soit sa position dans l'environnement proche de la terre. Onze satellites sont occasionnellement visibles au-dessus de l'horizon (élévation nulle).



**Figure1. 6** : constellation des satellites.

Chaque satellite pèse 846 kg une fois placé sur son orbite.

Chaque satellite possède un oscillateur qui fournit une fréquence fondamentale de 10,23 MHz calibrée sur des horloges atomiques (deux horloges au caesium et deux horloges au rubidium) et permet de :

- Transmettre les informations aux utilisateurs par l'intermédiaire de deux porteuses (L1 et L2) de fréquences respectives 1575,42 MHz et 1227,60 MHz
- Asservir sa position et son altitude.
- Assurer une liaison UHF entre les différents satellites.
- Transmettre régulièrement des signaux horaires, la description de l'orbite suivie et diverses autres informations

Des panneaux solaires de  $7.25 \text{ m}^2$  fournissent une puissance électrique de 700 W.

Chaque satellite peut être identifié de différentes manières :

- Par son numéro séquentiel SVN (Space Vehicle Number ) qui correspond à son ordre de lancement ;
- Par son numéro PRN (PseudoRandom Noise), le plus utilisé, qui permet d'identifier le code pseudo-aléatoire émis par le satellite.
- Par un numéro dépendant du plan de l'orbite et de la position du satellite sur cette orbite ;
- Par son numéro COSPAR, lié à sa date de lancement;
- Par son identifiant dans le catalogue interne de la NASA;

### **B. Segment de contrôle :**

Ce segment est constitué d'une station principale (Master Station), de cinq stations de contrôle (Monitor Station) et de trois stations de téléchargement (Ground Antenna).

La station principale est située à Schriever, traite toutes les informations diffusées par les satellites y compris les informations de télémétrie. Les orbites des satellites sont paramétrées et la dérive d'horloge des satellites modélisée permettant ainsi de synchroniser le temps satellite et le temps GPS. Ces données sont régulièrement transmises vers les satellites sous la responsabilité de la station principale.

Les informations GPS sont collectées en permanence par cinq stations monitrices globalement dispersées et couvrant la quasi-totalité de l'espace dans lequel gravitent les satellites. Ces cinq stations sont situées à Hawaï, Colorado Springs, sur l'île de l'Ascension, Diego Garcia et à Kwajalein. Seules trois stations (L'île de l'Ascension, Diego Garcia et Kwajalein) assurent en plus les fonctions de téléchargement des données vers les satellites et contrôlent le bon transfert. Le chargement des données s'effectue une fois par jour par une liaison à 2.2 GHz. Le satellite dispose de 14 jours de capacité mémoire. Au-delà de cette période et sans rafraîchissement des données, il devient risqué de naviguer avec ces satellites car la précision des mesures se dégrade rapidement.

Le rôle des stations de contrôle est de poursuivre passivement les satellites en code et en porteuses qu'elle que soit leur position et de faire l'acquisition des messages de navigation.

Le niveau des signaux GPS est également surveillé. Ces informations sont envoyées à la station principale. Elles lui permettent de calculer les éphémérides (orbites, paramètres d'horloge, codes, etc.) Et d'estimer la dérive d'horloge des satellites. Par ailleurs, la station estime le retard ionosphérique et calcule les paramètres du modèle de correction.

La station de contrôle principale asservit la position des satellites et gère la constellation spatiale. Elle actualise l'almanach à chaque modification de la constellation spatiale, du lancement ou à la fin de vie d'un satellite. L'almanach est une information qui calcule une position approximative d'un satellite. Elle sert pour déterminer si un satellite est visible. Ceci facilite la recherche des satellites et leurs acquisitions.



**Figure 1.7:** le segment de contrôle.

### C. Segment utilisateur :

Le segment utilisateur consiste en une variété de récepteurs GPS militaires ou civils. Un récepteur est conçu pour recevoir, décoder et traiter les signaux émis par les satellites GPS (la position, la vitesse et le temps de l'utilisateur). Les récepteurs intégrés (carte ou circuits spécifiques au GPS) dans d'autres systèmes sont également inclus.

Les utilisateurs disposent d'un moyen unique pour leurs applications de localisation, de navigation, de référence de temps, de détermination d'altitude.

Ces applications diverses ont conduit à développer différents types de récepteurs chacun pouvant inclure différentes fonctions adaptées aux besoins.



**Figure 1.8 :** le segment utilisateur.

### III.1.3. Structure du signal GPS :

Chaque satellite GPS transmet ses informations dans la bande L sur deux fréquences L1 (1575.43 MHz) et L2 (1227.60 MHz). La fréquence d'émission des signaux des satellites GPS est très stable grâce à des horloges atomiques embarquées. L'onde porteuse L1 est modulée par 2 codes C/A (Coarse/Acquisition code) et P (Precise code) et par un signal contenant les messages de navigation de fréquence 50Hz. Ces messages sont constitués des données d'éphémérides du satellite, des données de correction de propagation ainsi que du temps des satellites. Ces messages permettent au récepteur d'identifier le satellite et de connaître précisément sa position dans le référentiel de temps absolu GPS. La seconde onde porteuse L2, quant à elle, est modulée uniquement par le code P ainsi que par les mêmes messages de navigation dans L1. L2 est utile lorsque la fréquence L1 est perturbée. Elle permet aussi de corriger les retards de transmission liés à l'ionosphère.

Chaque satellite GPS possède ses propres codes pseudo-aléatoires, ce qui permet d'éviter les problèmes de brouillage non intentionnels.

- Le code C/A (coarse/acquisition ou "grossier") est présent sur la porteuse L1 uniquement. Il consiste en une séquence de 1023 bits se répétant toutes les millisecondes. A chaque satellite GPS correspond un unique code C/A (on parle de numérotation PRN). Il en existe au total 37 dont 5 réservés au secteur de contrôle. La fréquence élevée de répétition du code permet au récepteur de facilement identifier le satellite émetteur.
- Le code P (precise) est présent sur les deux fréquences GPS. Depuis le 1er février 1994, le code P est crypté (on parle alors du code Y) et accessible uniquement aux utilisateurs habilités par le Ministère de la Défense américain (US DoD). La génération complète du code P, à raison de  $10.23 \cdot 10^6$  bits/s, nécessite 266 jours. En fait, la séquence complète du code P est divisée en 37 segments et chaque satellite utilise un segment différent, qui se répète toutes les semaines. Contrairement au code C/A, le code P n'est pas ambigu. Néanmoins, de part sa longueur, le récepteur aura plus de difficultés pour retrouver la bonne portion du code correspondant au satellite émetteur. Là encore, la présence du message de navigation permet de pallier la difficulté.

### III.1.4. Fonctionnement du système GPS :

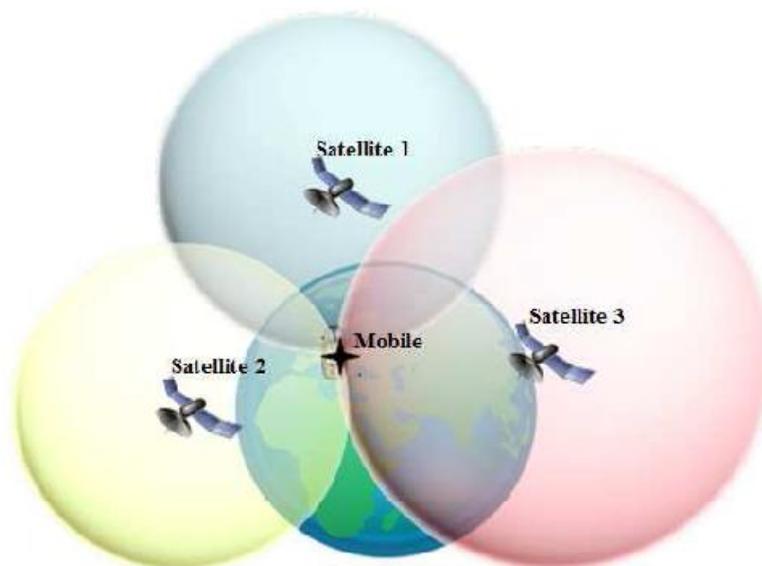
La position donnée par le GPS est celle de l'antenne du récepteur de l'utilisateur. Le référentiel GPS est le WGS84 (World Geodetic System 1984). Le principe de la détermination de cette position est la *trilatération* : on mesure la distance entre l'utilisateur et un certain nombre de satellites, dont on connaît les positions puis, de façon imagée, on obtient des sphères centrées sur des satellites dont l'intersection permet de reconstituer les trois informations de position de l'utilisateur (latitude, longitude et hauteur ellipsoïdale). La distance par rapport à chaque satellite est obtenue grâce à sa relation de proportionnalité avec le temps de parcours du signal émis par chaque satellite respectif. En effet, connaissant  $T_{si}$

l'heure exacte d'émission d'un message par un satellite  $S_i$  et l'heure exacte  $t_R$  de ce message sur le récepteur R, on peut en déduire le temps de propagation du signal. En multipliant ce temps par la célérité de la lumière, on obtient la distance  $\Delta_{S_i,R}$  du satellite  $S_i$  au récepteur R.

Dans le cas idéal, trois satellites suffisent pour déterminer la position du récepteur. En effet, l'intersection des trois sphères distinctes donne 2 points. L'un de ces deux points se situe loin de la surface du globe terrestre et n'est donc pas crédible. En réalité, quatre satellites au moins sont nécessaires, si on prend en compte les erreurs d'horloge. En effet, on peut remarquer que la distance  $\Delta_{S_i,R}$  mesurée qui est la pseudo distance est faussée par le décalage temporel  $\Delta T_i$  entre le temps GPS et celui du récepteur. Ce décalage qui peut prendre de grandes valeurs est considéré comme une quatrième variable à estimer.

En outre, pour une utilisation plus efficace du GPS, les satellites NAVSTAR reçoivent quotidiennement des paramètres de corrections diffusés par les stations. Il est donc possible à l'utilisateur de régler son horloge lors du calcul de positionnement qui fournit l'imprécision  $\Delta T$  de l'horloge utilisateur (en négligeant l'erreur de l'horloge GPS).

La prise en compte de toutes les erreurs qui perturbent la propagation du signal GPS permet d'atteindre des précisions élevées. Les sources d'erreurs sont vues en détail dans ce qui suit.



**Figure1.9 :** Principe de localisation par satellite GPS.

### III.1.5. Sources d'erreurs : [8]

L'erreur moyenne sur la position d'un récepteur du système GPS est de l'ordre d'une quinzaine de mètres. Le positionnement issu de mesures GPS dépend de la géométrie de la distribution des satellites et du nombre de ces derniers au moment de la mesure. En effet, puisqu'il existe quatre inconnues principales dans le calcul GPS (la position en trois dimensions et un décalage d'horloge), il faut au moins la visibilité de quatre satellites pour déterminer une position GPS. De surcroît, une bonne configuration géométrique des satellites est nécessaire pour une bonne précision du positionnement. Ainsi, les récepteurs utilisent le critère de GDOP (Géométric Dilution Of Precision) pour évaluer la qualité de cette configuration satellitaire. Au dessus d'un certain seuil (généralement  $GDOP < 5$ ), la mesure n'est pas jugée utilisable.

Les erreurs peuvent être subdivisées en erreurs de manipulation et en erreurs systématiques. Dans les erreurs de manipulations, on range les erreurs humaines qui peuvent être commises par les utilisateurs du GPS ou dans les stations de contrôle. Les erreurs systématiques sont, quant à elles, difficiles à modéliser et leur estimation permet d'obtenir de grandes précisions.

On peut détailler ces différentes erreurs de la façon suivante :

- **Erreurs dues aux matériels**

Les erreurs matérielles les plus significatives sont celles au niveau des antennes des récepteurs ou celles qui découlent de la résolution des mesures par les récepteurs. Les erreurs d'antenne sont dues à une variation d'un point appelé "centre de phase". C'est un point immatériel qui correspond au point où est capté le signal GPS. Les variations du centre de phase sont fonction de l'azimut et de l'élévation des satellites ainsi que de leur configuration. En pratique ce point est fixé pour un récepteur donné et les constructeurs calibrent les antennes et fournissent les modèles de calibration permettant de corriger les mesures.

- **Erreurs d'orbites**

Les satellites sont soumis à de nombreux phénomènes physiques tels que la gravitation (attraction lunaire ou solaire, perturbation des marées océaniques), l'accélération relativiste, la pression des radiations solaires, etc. Les trajectoires prévues des satellites sont donc influencées par ces phénomènes. Les messages de navigation émis par le satellite ne permettent donc de calculer l'orbite du satellite en temps réel qu'avec une certaine erreur (à 20 m près). En outre, un terme correctif lié à la rotation terrestre est pris en compte lors du calcul de positionnement GPS (en effet, la terre a tourné pendant le temps de vol du signal).

- **Erreurs de propagation du signal**

Plusieurs facteurs peuvent affecter la propagation signal. On peut citer parmi ceux-ci, les erreurs non critiques liées à des retards de l'électronique des récepteurs et celle des satellites. Les erreurs qui influent le plus sur le temps de propagation, sont liées à la traversée de la troposphère (partie basse de l'atmosphère) et de l'ionosphère (partie haute de l'atmosphère). Ces deux couches de l'atmosphère, l'ionosphère plus particulièrement, rallongent le temps de

parcours du signal. Ces erreurs sont d'autant plus difficiles à estimer qu'elles dépendent de l'élévation des satellites. Pour la troposphère, l'erreur dépend aussi de la température, de la pression atmosphérique et de l'humidité.

- **Erreurs liées à l'environnement du récepteur**

Ces erreurs sont essentiellement dues aux multi-trajets du signal ou à un masquage satellitaire. Elles dépendent de l'environnement du récepteur et ne sont donc pas modélisables ou quantifiables a priori. Pour limiter l'effet de ces multi-trajets, les antennes des récepteurs sont équipées de systèmes qui filtrent en partie les multi-trajets.

### **III.1.6. Points forts du GPS :**

Le GPS s'impose pour l'utilisateur comme le seul système GNSS disponible et donc le seul système qui, utilisé seul, peut fournir une position dans un repère global (WGS84), ce qui en fait un outil quasi indispensable pour des applications de localisation à la surface de la terre. L'utilisation du GPS est gratuite, ce qui implique que la réception des informations ne nécessite pas le paiement d'aucune redevance. Il faut ajouter à cela que sa précision s'est améliorée en permanence au cours du temps grâce à diverses techniques qui ont évolué très rapidement. Les récepteurs sont passifs et le nombre d'utilisateurs est donc illimité. Un autre avantage du GPS est le coût des récepteurs qui est constamment revu à la baisse grâce, d'une part, à sa constante évolution technologique, mais aussi, grâce à l'immensité des champs d'application et au nombre gigantesque de récepteurs.

### **III.1.7. Points faibles du GPS :**

Le principal point faible du GPS, qui est en même temps une limitation, se situe dans la disponibilité des données. En effet, Le système GPS est, généralement, le plus utilisé pour se déplacer à l'extérieur. Cependant, il reste peu fiable voire non fonctionnel à l'intérieur des bâtiments (centre commercial, aéroport, métro, musée, magasin...) voire même dans certains milieux mal adaptés (en milieu urbain, forêt, tunnel, bâtiment, etc.). Ce problème étant difficilement soluble par les récepteurs, il justifie amplement la nécessaire hybridation du GPS avec des capteurs proprioceptifs pour des applications critiques. Ainsi dans des applications critiques dans des milieux hostiles, le GPS est surtout utilisé pour recalculer les dérives des estimations issues des capteurs proprioceptifs. Un autre point faible est le temps de localisation qui est assez long puisqu'il peut atteindre 15 minutes le temps que le téléphone accroche les satellites qui permettront le calcul de sa position après une perte de réception par exemple.

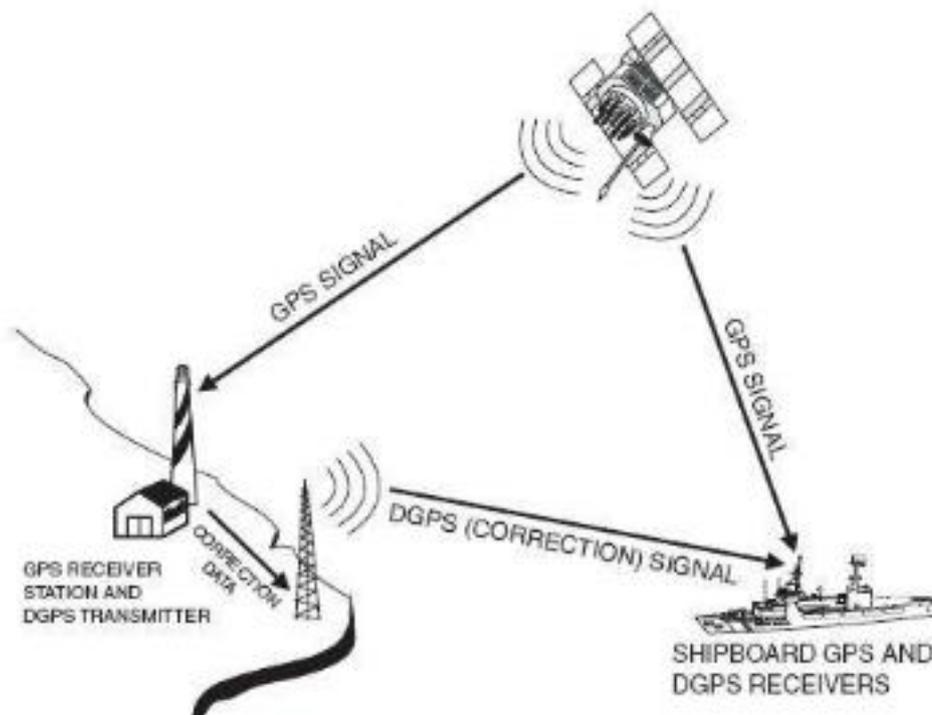
### III.1.8. Amélioration du GPS [2]

Afin d'améliorer la précision du système GPS plusieurs systèmes d'améliorations ont été développés, comme illustrer dans ce qui suit.

#### III.1.8.1. DGPS (GPS différentiel) :

Differential Global Positioning System ou GPS différentiel permet d'améliorer la précision du GPS en réduisant la marge d'erreur du système. Son principe est basé sur l'hypothèse raisonnable que deux positions mesurées à un instant donné, par deux récepteurs distants l'un de l'autre de moins d'une centaine de Km présentent la même erreur. Cette hypothèse se justifie en considérant la grande distance entre les satellites et les récepteurs, qui est de l'ordre de 20000 Km. En effet, les deux récepteurs vus par un satellite sont quasiment confondus et donc les perturbations dues à la traversée de l'atmosphère sont quasi identiques.

En partant de cette hypothèse, si on effectue régulièrement une mesure à un point donné (station de référence) dont on connaît parfaitement la position, il est possible de calculer toutes les erreurs et de transmettre des valeurs de correction à tous les utilisateurs dans un rayon de quelque dizaines de Km. Les récepteurs peuvent de fait sensiblement améliorer leur calcul de position. On appelle ligne de base, le vecteur constitué de deux récepteurs. . On peut ainsi passer d'une précision de l'ordre de 15 mètres à une précision de 5 à 3 mètres

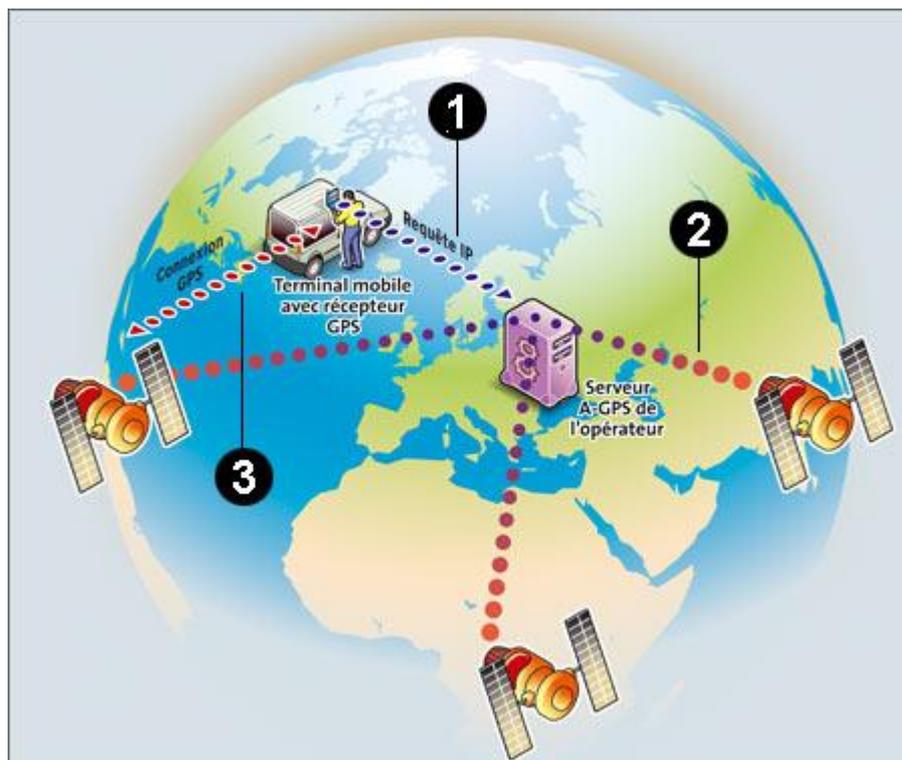


**Figure1.10** : géolocalisation en utilisant le GPS différentiel.

### III.1.8.2. A-GPS (Assisted GPS)

L'Assisted GPS (A-GPS) a été conçu par Giovanni Vannucci, chercheur au centre de recherche de Lucent Technologies, Bell Labs. Au milieu des années 90, il entreprend de localiser les téléphones portables à l'aide de liaisons satellites afin de pouvoir secourir les gens en cas d'urgence. De sa collaboration avec Bob Richton, du Wireless Technology Laboratory, naît la méthode A-GPS qui réduit le temps nécessaire pour établir la première connexion satellite (TTFF pour Time To First Fix). Le délai passe de plusieurs minutes avec un système GPS traditionnel à quelques secondes avec l'A-GPS.

L'A-GPS travaille en relation avec un serveur A-GPS hébergé chez l'opérateur. Le terminal mobile, muni d'un récepteur GPS miniaturisé, envoie une requête au serveur par le réseau IP. Celui-ci, qui connaît en temps réel le positionnement des satellites, sert d'aiguilleur et indique au terminal les signaux GPS à suivre. Avec cette méthode, le récepteur A-GPS du terminal mobile peut, à la différence des récepteurs GPS traditionnels, détecter des signaux de très faible amplitude. Le niveau de précision est élevé puisqu'il est annoncé entre 5 et 10 mètres contre 3 à 50 mètres avec le GPS traditionnel.



**Figure1.11** : principe de localisation avec A-GPS.

### **III.1.9. Systèmes concurrents au GPS :[1]**

Il existe bien d'autres systèmes de navigation par satellite tel que :

#### ***III.1.9.1. GLONASS :***

Système Global de Navigation par Satellite est le nom du système de positionnement utilisé par la Russie. Il repose aussi sur une constellation de 24 satellites mis dans 3 plans orbitaux (8 satellites pour chaque plan orbital), dont, depuis 2008, 16 satellites sont actifs et en orbite. La partie au sol est formée de cinq stations de contrôle, la principale se trouve dans la région de Moscou.

Les utilisateurs du système, comme pour le GPS, doivent disposer d'appareils de positionnement équipés de récepteurs spécifiques. Comme le GPS GLONASS ne peut pas être utilisé à l'intérieur des bâtiments, la puissance des ondes radios reçues des satellites étant trop atténuée par les murs.

#### ***III.1.9.2. EGNOS :***

EGNOS ou European Geostationary Navigation Overlay Service est le premier système européen utilisant les ondes des satellites et qui a pour but d'améliorer la précision des deux systèmes GPS et GLONASS. Il utilise les constellations de satellites GPS et GLONASS et s'appuie sur une quarantaine de stations terrestres. Ces stations terrestres. Elles constituent un maillage serré qui complète très utilement la triangulation obtenue à partir des satellites du GPS.

Elles reçoivent les signaux des constellations de satellites et différentes données climatologiques. Elles communiquent avec des centres de contrôle qui exploitent les informations reçues et transmettent le résultat des corrections différentielles à un satellite géostationnaire ((INMARSAT-3 AOR-E, INMARSAT-3 IOR ou ESA ARTEMIS). Ces satellites géostationnaires peuvent ensuite les communiquer aux récepteurs des utilisateurs. Une précision de 20 m obtenue avec le GPS peut passer à une précision de 2 mètres avec EGNOS, avec des signaux fiables. Il est prévu que ce système soit complètement déployé à la fin de 2009.

#### ***III.1.9.3. Galileo :***

Le système EGNOS préfigure Galileo. Galileo a pour but de supprimer la dépendance de l'Europe du système américain GPS. Il s'appuie sur une constellation de 30 satellites en orbite moyenne. Il est prévu que dans quelques années (implémentation prévue en 2014-2015) Galileo fournira une bonne précision de positionnement et différents types de qualité de service.

**III.1.9.4. Beidou :**

Est le système chinois. Actuellement, il n'est composé que de cinq satellites géostationnaires au dessus de la Chine, couvrant ainsi l'intégralité de son territoire. Une deuxième phase de Beidou va être d'envoyer 30 satellites non géostationnaires afin de couvrir en entier la Terre.

L'unique raison d'être de ce système est de s'affranchir de la dépendance des Etats-Unis. Le Japon et l'Inde sont en train de développer des systèmes locaux, composés de satellites géostationnaires.

Tous les principes de fonctionnements, calculs et problèmes physiques propres au GPS le sont également pour tous les GNSS.

**III.2. Wifi Positioning System (WPS) : [10]**

Le GPS ne peut pas être utilisé correctement dans les milieux intérieurs car les signaux satellites ne sont pas toujours assez puissants pour pénétrer dans ces zones ; d'autre part, Le déploiement d'infrastructures spécifiques dédiées à la localisation indoor peut avoir un coût important pour une utilisation limitée. Il est donc utile de concevoir un système de positionnement qui peut utiliser les infrastructures existantes (utilisée pour la communication ou pour d'autres buts) et qui soit utilisable en indoor et en outdoor pour permettre une continuité de service. Les systèmes de positionnement utilisant les ondes radio Wifi ont été proposées dans ce but et deviennent de plus en plus répandus.

Le géo positionnement à l'aide de la technologie Wifi est baptisé WPS pour Wifi Positioning System. En comparaison avec le GPS, le WPS remplace l'infrastructure des satellites par les infrastructures radios des réseaux Wifi et dispose de plusieurs atouts :

- Sa couverture intérieure et extérieure, lui permettant, au contraire du GPS, de continuer à fournir un géo-positionnement relativement précis en indoor et dans certaines zones urbaines denses avec des effets de canyon urbain. La technologie donne même les meilleurs résultats dans un environnement particulièrement dense, en raison de la multiplication des points d'accès.
- Il n'implique pas de matériel supplémentaire, l'équipement Wifi étant déjà présent au sein des différents appareils de communication.

Il présente cependant des inconvénients :

- WPS pose un problème de couverture en environnement rural ou dans des zones peu équipées en points d'accès Wifi.
- Les points d'accès Wifi sont des récepteurs plus mobiles que les infrastructures GPS, ce qui peut fausser les calculs si les bases de données ne sont pas mises à jour régulièrement. Plusieurs techniques de mesure (TOA, TDOA, AOA, RSS) existent pour le calcul des distances et des angles dans les systèmes de géo localisation utilisant les ondes radio. Cependant, la méthode TOA est peu envisageable pour du Wifi car les points d'accès ne sont pas synchronisés avec les récepteurs ; de même dans la TDOA, les points d'accès radio

doivent avoir une horloge très précise et bien synchronisée. Des systèmes comme « AirLocation » et « AeroScout » utilisent cette technique TDOA mais nécessitent du matériel supplémentaire (des points d'accès ou des récepteurs spécifiques) pour mesurer cette différence de temps. La technique AOA demande des antennes motorisées (ou à balayage) pour déterminer l'angle de réception et est peu utilisée actuellement avec les antennes des points d'accès Wifi mais l'arrivée des systèmes MIMO Wifi pourrait modifier cette situation. La technique basée sur le RSS est la plus souvent utilisée en Wifi, elle suppose cependant que le modèle d'atténuation des lieux (obstacles, murs...) soit bien connu, ou appris par calibration.

#### **IV. Techniques de géolocalisation : [6]**

##### **a. Géolocalisation par satellite :**

La géolocalisation par satellite est la technique de géolocalisation la plus répandue actuellement. Elle permet à un équipement équipé d'une puce compatible de calculer sa position en utilisant les signaux émis par différents satellite prévue à cet effet. Cette position est alors traduite en termes de latitude, longitude et parfois altitude (ex : 43° 5494 N - 1° 48472 E) et peut alors être représentée physiquement sur une carte. Le réseau satellite de positionnement le plus connu est le GPS (Global Positioning System).

##### **b. Géolocalisation par GSM :**

La géolocalisation en utilisant les réseaux de téléphonie mobile digitale (GSM) ou « Global System Mobiles » est une norme de transmission téléphonique et numérique utilisée en Europe par les téléphones sans fil. Elle a été utilisée avant la démocratisation des puces GPS dans les Smartphones, mais continue de l'être. Avec cette technique de géolocalisation, on trouve la position d'un GSM en se basant sur certaines informations relatives aux antennes GSM auxquels ce dispositif est connecté.

Ces antennes captent le signal émis par le téléphone et le calcul de la position peut être effectué par plusieurs méthodes :

- ✓ en fonction du temps que le signal met pour atteindre l'antenne.
- ✓ en fonction de l'angle d'arrivée et de la force du signal.
- ✓ en fonction des identifiants des antennes GSM auxquels un terminal se connecte, cette méthode, appelée Cell ID, est couramment utilisée. Une base de données fait le lien entre les identifiants des antennes et leur position.

La précision va dépendre, ici, de l'environnement, du milieu où se situe l'appareil. En milieu urbain, la densité d'antennes est supérieure par rapport au milieu rural. Moins précise, la

précision du positionnement par GSM peut aller de 200 mètres à plusieurs kilomètres. C'est pour cela que cette technique ne peut pas être utilisée seul pour géolocaliser un utilisateur et son Smartphone. De plus, la couverture géographique n'est pas complète en Algérie ni dans le monde.

Il existe trois techniques différentes de géolocalisation par GSM, le différentiel temps, l'identification de cellule et la triangulation.

### **c. Géolocalisation par Wi-Fi :**

Le WiFi (Wireless Fidelity) est une technologie de communication sans fil permettant de relier des ordinateurs portables, des ordinateurs de bureau, des assistants personnels (PDA) ou tout type de périphérique à une liaison haut débit sur un rayon de plusieurs dizaines de mètres en intérieur (généralement entre une vingtaine et une cinquantaine de mètres) à plusieurs centaines de mètres en environnement ouvert.

La géolocalisation utilisant les points d'accès Wi-Fi est une technologie beaucoup plus récente, particulièrement adaptée à la ville. Pour se géolocaliser, un Smartphone commence par dresser la liste de tous les BSSID présents autour de lui. Les BSSID sont les adresses MAC des cartes réseau Wi-Fi des points d'accès. Ceux-ci sont envoyés régulièrement par les points d'accès pour permettre aux terminaux Wi-Fi de s'y connecter. Une fois une liste des BSSID à proximité établie, il y a deux possibilités :

– soit le téléphone envoie cette liste à un serveur qui connaît la plupart des coordonnées géographiques des points d'accès. Ce serveur calcule alors la position du Smartphone et la lui envoie.

– soit le téléphone envoie une partie de la liste des BSSID au serveur, qui lui répond

Une autre liste de BSSID sur un plus grand périmètre avec leurs coordonnées et c'est le Smartphone qui calcule sa position.

### **d. Géolocalisation par adresse IP (sur Internet) :**

Grâce à cette technique, on peut déterminer la position géographique d'un ordinateur ou de n'importe quel autre terminal connecté à Internet. On se base ici sur l'adresse IP (numéro d'identification attribué à un appareil connecté à un réseau informatique). Ces adresses IP sont gérées par l'IANA, une organisation qui se charge de distribuer les adresses IP disponibles aux pays demandeurs. Les attributions sont donc très bien documentées et il est possible de déterminer facilement dans quel pays se trouve un terminal.

### **e. Géolocalisation par RFID :**

Le RFID est un sigle couramment utilisé pour désigner la radio-identification. Il s'agit d'une méthode destinée à mémoriser et récupérer des données à distance grâce à des marqueurs appelés "radio-étiquette" (ou RFID tag). Ces étiquettes sont de petits objets comme des étiquettes autoadhésives. Elles peuvent facilement être incorporées à des produits et des

organismes vivants (animaux,...). Les "radio-étiquettes" sont équipées d'une antenne et d'une puce électronique. Ces puces contiennent un identifiant. Grâce à un lecteur RFID, la puce peut être détectée. L'identifiant est ensuite assimilé à une zone géographique.

#### **f. La géolocalisation mixte :**

La géolocalisation hybride utilise plusieurs des méthodes de géolocalisation évoquées précédemment. Elle peut ainsi combiner les avantages de la géolocalisation GSM (très rapide) Wi-Fi (rapide et très précis en ville) et GPS (précis et possible partout en extérieur). Car aucune technique n'est parfaite. Chacune a ses propres faiblesses, généralement liées à leur dépendance à un certain réseau. Ainsi, la géolocalisation par GPS est impossible à utiliser à l'intérieur et elle nécessite un temps de réponse à l'allumage. La géolocalisation par GSM a une couverture géographique limitée. Enfin, la géolocalisation par WiFi souffre, quant à elle, d'une dépendance à la présence de bornes d'accès WiFi et plus complexe en milieu rural.

D'où l'existence de certains dispositifs qui combinent ces trois techniques pour limiter leurs faiblesses. Ce type de dispositif peut permettre, par exemple, de géolocaliser un individu à l'extérieur en utilisant le GPS et de garder sa trace à l'intérieur grâce aux techniques GSM et WiFi.

Il est bon de voir aussi que la géolocalisation par Wi-Fi est un très bon complément du GPS, car elle permet de se géolocaliser très précisément en ville où le signal GPS peut ne pas être assez puissant, mais est totalement inefficace en campagne, là où le signal GPS est le plus fort.

Nous pouvons voir un tableau de comparaison des différentes techniques de localisation en Tableau 1.

Technologie	GSM	Wifi	GPS	Hybride
Vitesse de géolocalisation	Très rapide	Rapide	Lent	Progressif
Utilisation de "data"	Oui	Oui	Non	Oui ou Non
Précision	100m–10km	10–50m	10–100m	10m–10km
Fonctionne	Mieux en ville	Ville	Mieux hors ville	Partout

**Tableau 1** : Comparaison des différentes techniques de géolocalisation.

## **V. Applications de géolocalisation :**

Plusieurs travaux dans le domaine de géolocalisation ont été mis en œuvre, étant donné que de plus en plus d'entreprises, sociétés de transport de biens ou de personnes, dépanneurs, loueurs, ... expriment leur besoin d'une solution de géolocalisation, de communication et de sécurité.

La solution c'est pouvoir garder maîtrise et pouvoir gérer la flotte de véhicule depuis son bureau afin d'améliorer la productivité des employés qui sont sur la route. Intervenir plus rapidement en cas d'accidents, etc.

La solution est un outil de géolocalisation par GPS qui va faire la liaison entre les véhicules et le bureau.

Plusieurs systèmes existent déjà sur le net, et plusieurs thèmes et recherches sur ce sujet en étaient mis en œuvre au niveau de plusieurs universités que ce soit sur le territoire national ou à l'étranger.

Ces thèmes sont basés sur l'architecture logicielle des systèmes de géolocalisation et la différence qui surgit dans chacun d'eux est apparente dans l'analyse et conception du système ainsi que la réalisation du système ou chacun utilise des outils et technologies différents.

**VI. Conclusion :**

Dans ce chapitre on a dressé l'état de l'art sur la géolocalisation, on invoquant les différentes techniques, systèmes et technologies de positionnement. Le but de cela est de chercher parmi ces technologies la mieux adaptée et la plus précise à notre système de positionnement qui consiste en la géolocalisation de véhicules sur route. Certainement le système le mieux adapté à la localisation en extérieur est le système GPS.

CHAPITRE

---

# II

*Les services mobiles*

## **I. Introduction :**

Les développements récents dans le domaine des communications sans fil et l'introduction de l'internet dans les réseaux cellulaires ont permis aux utilisateurs d'avoir une plus grande gamme de services. L'utilisateur n'est plus obligé de se déplacer et s'asseoir devant un PC pour faire une recherche, consulter son courrier électronique ou encore faire des téléchargements de fichiers, de sonneries, de musique, etc. Dans ce chapitre, nous présentons les différents services et applications offerts par les réseaux mobiles et les terminaux pour lesquels ils sont destinés.

Nous nous intéressant plus exactement au service de géolocalisation qui est un service en plein essor, qui offre divers possibilités dans la localisation et le positionnement sur carte des points dans différents domaines.

## **II. Les terminaux mobiles :**

On distingue plusieurs catégories de terminaux mobiles, selon le service pour lesquels ils sont dédiés :

### **II.1. Les téléphones mobiles : [10]**

#### **II.1.1. La téléphonie mobile 1 G :**

La première génération de téléphonie mobile (notée **1G**) possédait un fonctionnement analogique et était constituée d'appareils relativement volumineux. Il s'agissait principalement des standards suivants :

- **AMPS** (Advanced Mobile Phone System), apparu en 1976 aux Etats-Unis, constitue le premier standard de réseau cellulaire. Utilisé principalement Outre-Atlantique, en Russie et en Asie, ce réseau analogique de première génération possédait de faibles mécanismes de sécurité rendant possible le piratage de lignes téléphoniques.
- **TACS** (Total Access Communication System) est la version européenne du modèle AMPS. Utilisant la bande de fréquence de 900 MHz, ce système fut notamment largement utilisé en Angleterre, puis en Asie (Hong-Kong et Japon).
- **ETACS** (Extended Total Access Communication System) est une version améliorée du standard TACS développé au Royaume-Uni utilisant un nombre plus important de canaux de communication.

Les réseaux cellulaires de première génération ont été rendus obsolètes avec l'apparition d'une seconde génération entièrement numérique.

### II.1.2. La téléphonie mobile 2G :

La seconde génération de réseaux mobiles a marqué une rupture avec la première génération de téléphones cellulaires grâce au passage de l'analogique vers le numérique. Les terminaux de seconde génération ont des capacités de traitement et de stockage très limitées. Leurs écrans sont de petites dimensions et généralement bicolores (vert et noir ou bleu et noir,...). Les principaux standards de téléphonie mobile 2G sont les suivants :

- **GSM** (*Global System for Mobile communications*), le standard le plus utilisé en Europe à la fin du XX<sup>e</sup> siècle, supporté aux Etats-Unis. Ce standard utilise les bandes de fréquences 900 MHz et 1800 MHz en Europe. Aux Etats-Unis par contre, les bandes de fréquences utilisées sont les bandes 850 MHz et 1900 MHz. Ainsi, on appelle **tri-bande**, les téléphones portables pouvant fonctionner en Europe et aux Etats-Unis.
- **CDMA** (*Code Division Multiple Access*), utilisant une technique d'étalement de spectre permettant de diffuser un signal radio sur une grande gamme de fréquences.
- **TDMA** (*Time Division Multiple Access*), utilisant une technique de découpage temporel des canaux de communication, afin d'augmenter le volume de données transmis simultanément. La technologie TDMA est principalement utilisée sur le continent américain, en Nouvelle Zélande et en Asie Pacifique.

Grâce aux réseaux 2G, il est possible de transmettre la voix ainsi que des données numériques de faible volume, notamment des messages textes **SMS** (Short Message Service). La norme GSM permet un débit maximal de 9,6 kbps.

### II.1.3. La téléphonie mobile 2.5G :

Ces terminaux sont conçus pour supporter des applications à haut débit (pouvant atteindre 2 Mbit/s) tels que les services multimédias. Des extensions de la norme GSM ont été mises au point afin d'en améliorer le débit. C'est le cas notamment du standard **GPRS** (*General Packet Radio System*), qui permet d'obtenir des débits théoriques de l'ordre de 114 kbit/s, plus proche de 40 kbit/s dans la réalité. Cette technologie ne rentrant pas dans le cadre de l'appellation « 3G » a été baptisée **2.5G**.

Les terminaux de la 2.5 génération sont conçus pour fonctionner à la fois sur les réseaux GSM et sur les réseaux GPRS. Les applications classiques de voix et de messageries courtes passent par le réseau GSM, tandis que les applications de mode de transfert par paquets tels que les services EMS (Enhanced Message Service), le MMS (MultiMedia Message service), la navigation WAP ou l'i-mode (l'internet mobile) passent par le réseau GPRS.

Les terminaux GSM/GPRS offrent une meilleure capacité de calcul, de traitement et de stockage (Acquisition de photos et de vidéos éventuelles via des circuits dédiés ou via l'interface infrarouge lecteur de vidéo, visionnement de photos, jeux 3D, gestionnaire de fichiers, etc.).

### II.1.4. La téléphonie mobile 2.75G :

La norme **EDGE** (Enhanced Data Rates for Global Evolution, présentée comme **2.75G** quadruple les améliorations du débit de la norme GPRS en annonçant un débit théorique de 384 Kbps, ouvrant ainsi la porte aux applications multimédias. En réalité la norme EDGE permet d'atteindre des débits maximum théoriques de 473 kbit/s, mais elle a été limitée afin de se conformer aux spécifications IMT-2000 (International Mobile Telecommunications-2000) de l'ITU (International Telecommunications Union).

### II.1.5. La téléphonie mobile 3G :

Les spécifications IMT-2000 (*International Mobile Telecommunications for the year 2000*) de l'Union Internationale des Communications (UIT), définissent les caractéristiques de la **3G** (troisième génération de téléphonie mobile). Ces caractéristiques sont notamment les suivantes :

- un haut débit de transmission :
  - 144 Kbps avec une couverture totale pour une utilisation mobile,
  - 384 Kbps avec une couverture moyenne pour une utilisation piétonne,
  - 2 Mbps avec une zone de couverture réduite pour une utilisation fixe.
- compatibilité mondiale,
- compatibilité des services mobiles de 3ème génération avec les réseaux de seconde génération,

La 3G propose d'atteindre des débits supérieurs à 144 kbit/s, ouvrant ainsi la porte à des usages multimédias tels que la transmission de vidéo, la visio-conférence ou l'accès à internet haut débit. Les réseaux 3G utilisent des bandes de fréquences différentes des réseaux précédents : 1885-2025 MHz et 2110-2200 MHz.

La principale norme 3G utilisée s'appelle **UMTS** (*Universal Mobile Telecommunications System*), utilisant un codage **W-CDMA** (*Wideband Code Division Multiple Access*). La technologie UMTS utilise la bande de fréquence de 5 MHz pour le transfert de la voix et de données avec des débits pouvant aller de 384 kbps à 2 Mbps. La technologie **HSDPA** (*High-Speed Downlink Packet Access*) est un protocole de téléphonie mobile de troisième génération baptisé « 3.5G » permettant d'atteindre des débits de l'ordre de 8 à 10 Mbits/s. La technologie HSDPA utilise la bande de fréquence 5 GHz et utilise le codage W-CDMA.

### **II.1.6. La téléphonie mobile 4G :**

La technologie 4G est la nouvelle génération des standards téléphoniques, en voie d'expansion à travers le monde. Alors qu'au sein du réseau 3G, les données internet et conversations téléphoniques étaient séparées, le débit devant alors se partager entre les utilisateurs connectés, la 4G réunit l'ensemble de ces données. Cela garantit un transfert de données de meilleure qualité.

Les débits vont de 100Mb/s à 1Go/s. Le réseau mobile 4G peut donc offrir un débit nettement supérieur à celui de la fibre optique actuelle, du moins en théorie car en pratique il n'est pour l'instant que de quelques dizaines de Mb/s, du fait que la bande passante soit partagée entre les différents utilisateurs du réseau dans une même zone. La norme 4G utilisée est le **LTE** (Long Term Evolution) et elle utilise les bandes de fréquences des 2 600 MHz et des 800 MHz.

Comme pour le passage de la 2G à la 3G, les terminaux mobiles doivent être adaptés à la nouvelle génération 4G, ce qui est déjà le cas pour bon nombre de produits qui ont été mis sur le marché récemment. En effet, qu'il s'agisse d'une clé mobile ou d'un Smartphone, leur adaptation aux nouveaux protocoles IPv6, fournis par la connexion 4G, est nécessaire. De leur côté, les opérateurs commencent peu à peu à proposer des forfaits adaptés.

### **II.2. Les ordinateurs portables : [w2]**

Un ordinateur portable ou ordinateur personnel dont le poids et l'encombrement limités permettent un transport plus facile. Les ordinateurs portables possèdent plusieurs objectifs, qu'ils soient lucratifs et éducatifs, ou servent généralement aux stockages de médias personnels. Un ordinateur portable est muni de mêmes composants qu'un ordinateur de bureau classique, incluant notamment écran, haut-parleurs, et clavier. Depuis les années 2010, la plupart possèdent en plus une webcam intégrée et un microphone préinstallé. Un ordinateur portable est muni d'un accumulateur électrique, ou batterie, rechargeable à l'aide d'une prise adaptable.

### **II.3. Les terminaux PDA : [w2]**

Un assistant numérique personnel, un pocket PC, ou un agenda électronique est un appareil numérique portable, souvent appelé par son sigle anglais « PDA » pour « Personal Digital Assistant ». Un PDA est un ordinateur de poche composé d'un processeur, de mémoire vive, d'un écran tactile avec un stylet et de fonctionnalités réseau (Bluetooth, WiFi, etc.), le tout mis dans un boîtier compact d'une petite taille. Les PDA possèdent généralement les possibilités de communication classique (téléphone, navigation internet) mais ils sont plus adaptés à l'organisation des tâches quotidiennes (gestion des rendez vous, des contacts personnels, des dépenses, etc.).

#### **II.4. les Tablettes tactiles : [w2]**

Une tablette tactile est un ordinateur portable très léger et sans clavier. Le texte est entré manuscrit à l'aide d'un stylo, en touchant l'écran, ou par oral. La puissance de calcul d'une tablette est inférieure à celle d'un ordinateur de bureau. Sur les tablettes convertibles, l'écran peut être retourné, et l'appareil peut ainsi être utilisé comme un notebook. Les tablettes électroniques existent depuis les années 1990 et ont été remises à la mode par le lancement de l'iPad en 2010. Ces appareils sont à mi-chemin entre le téléphone et l'ordinateur personnel. En 2013 la majorité de ces appareils sont équipés d'un système d'exploitation pour Smartphone et ils sont utilisés pour les mêmes activités qu'un ordinateur personnel.

#### **II.5. Smartphone : [w2]**

Un Smartphone, Ordiphone ou téléphone intelligent, est un téléphone mobile évolué disposant des fonctions d'un assistant numérique personnel, d'un appareil photo numérique et d'un ordinateur portable. La saisie des données se fait le plus souvent par le biais d'un écran tactile ou, plus rarement d'un clavier ou d'un stylet. Selon le principe d'un ordinateur, il peut exécuter divers logiciels/applications grâce à un système d'exploitation spécialement conçu pour mobiles, et donc en particulier fournir des fonctionnalités en plus de celles des téléphones mobiles classiques comme : l'agenda, la télévision, le calendrier, la navigation sur le Web, la consultation et l'envoi de courrier électronique, la géolocalisation, le dictaphone/magnétophone, la calculatrice, la boussole, l'accéléromètre, le gyroscope, la messagerie vocale visuelle, la cartographie numérique etc. Les appareils les plus sophistiqués bénéficièrent rapidement de la reconnaissance vocale et de la synthèse vocale.

##### **II.5.1. Historique et évolution :**

Le premier Smartphone, l'IBM Simon, fut conçu en 1992 puis commercialisé en août 1994. Progressivement les avancées technologiques se multiplient avec des débits de plus en plus importants, plusieurs fabricants de téléphones mobiles se lancent dans l'aventure mais l'événement qui marque la révolution du Smartphone est l'arrivée en 2007 de l'iPhone d'Apple avec interface tactile multipoint, sensible aux doigts de l'utilisateur. L'appareil fonctionne comme un ordinateur avec un système d'exploitation et des applications. Son succès commercial sera à l'origine de la généralisation des Smartphones à écrans tactiles chez tous les fabricants. Et depuis une course à la performance se met en place. Les processeurs sont désormais double cœur, voire quatre cœurs, avec des fréquences qui atteignent 1,5 GHz. La diversité des Smartphones augmente toujours, avec des écrans plus grands tout en produisant toujours les petits écrans. La résolution (pixels par pouce) continue d'augmenter. De façon général, plus le Smartphone a un grand écran plus la batterie est lourde.

### **II.5.2. Système d'exploitation mobile :**

Un système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface. On trouve ces systèmes d'exploitation installés sur les Smartphones et tablettes tactiles, il en existe plusieurs SE mobile mais les principaux systèmes sont Android, iOS, Windows phone 8.1 et Blackberry OS, représentant la quasi-totalité des parts de marché en 2014.

➤ **Android de Google :**

Un système d'exploitation développé par une petite startup qui fut acheté par Google (annoncé officiellement le 5 novembre 2007) qui poursuit activement son développement. C'est un système d'exploitation open source, fondé sur un noyau Linux, disponible via une licence Apache version 2, il inclut tous les utilitaires requis par un constructeur ou par un opérateur pour mettre en œuvre un téléphone portable. Android est conçu pour intégrer au mieux des applications existantes de Google comme le service de courrier G-mail, ou celui de cartographie, Google Maps, ou encore Google Agenda, Google Talk, YouTube. Il a eu douze versions, chacune portant un « nom de code » spécifique. Chaque version a subi des mises à jour et des modifications. Le nombre de Smartphone qui adopte l'Android ne cesse d'augmenter d'où son importante part du marché mondial.

➤ **IOS d'Apple :**

Le téléphone iPhone, l'iPod Touch et la tablette iPad utilisent tous le système d'exploitation iOS, qui est dérivé de Mac OS X. Les applications tierces sont supportées depuis juillet 2008. La boutique Apple propose désormais plus d'un million d'applications.

➤ **Windows Phone de Microsoft :**

Le système Windows CE était largement distribué en Asie. Les deux variantes de ce système d'exploitation et Windows Mobile 6 standard lancé en février 2007 ont été peu adoptées.

Le 15 février 2010 Microsoft a lancé un nouveau système d'exploitation pour mobile, Windows Phone 7. Il inclut des services de Microsoft comme Windows Live, Zune, Xbox Live et Bing. Il intègre aussi des fonctionnalités média sociaux tels Facebook, Twitter, Snapchat etc... Comme Windows Phone 7 est une nouvelle plateforme, il n'existe aucune compatibilité avec les anciennes applications Windows Mobile.

Les versions actuelles sont Windows Phone 7. Windows Phone 8 et Windows Phone 8.1.

Microsoft rachète en 2013 la division mobile de Nokia afin de pouvoir améliorer et faire progresser son OS.

➤ **BlackBerry OS de Research In Motion (RIM):**

Ce système fut créé à l'origine en 1999 pour le marché des hommes d'affaires. Il occupait 18,2 % de parts de marché au 2<sup>e</sup> trimestre 2010. Récemment, RIM a favorisé la création d'application web et supporte les applications multimédia. À ce jour, Blackberry's App World dispose de 6 000 applications.

### III. Les services mobiles :[10]

#### III.1. Le service de messagerie :

- **SMS** : (*Short Messaging Service*) a été lancé en 1992 et est rapidement devenu jusqu'à ce jour, le service d'envoi de données sans fil qui a connu le plus grand succès. Il peut aussi être appelé en fonction des terminologies des opérateurs : Mini-message, texto, télémessagerie.

Le SMS permet aux utilisateurs de téléphones mobiles d'envoyer et recevoir des messages textes pouvant contenir jusqu'à 160 caractères, ceci de façon rapide et peu coûteuse. Le SMS est un « store and forward » service, ce qui signifie qu'il n'est pas envoyé directement entre les utilisateurs mais via un Centre de Service de messages courts (SMSC).

Cette méthode confère au SMS ses principaux avantages, comme la possibilité de délivrance immédiate sans perturbations liées au réseau ou la tarification à l'unité (par SMS envoyé).

Le SMS peut aussi être utilisé en même temps que le transfert de la voie où utiliser le roaming sans frais internationaux supplémentaires.

Le SMS est aussi utilisé pour la notification de la réception d'emails, de messages vocaux ou de fax ou pour transmettre tout type d'information courte (météo, cours de bourse, grands titres de l'actualité,...).

- **MMS** :

Le MMS est actuellement la forme la plus aboutie de la messagerie mobile. On lui prédit un avenir aussi florissant que celui qu'a pu connaître le SMS. Le MMS n'a virtuellement plus de limite quand aux possibilités de son contenu : texte mis en page, graphiques, données, animation, séquence vidéo ou audio, voix. L'envoi de cartes postales digitales ou de documents de type PowerPoint devrait être parmi les applications les plus populaires du MMS.

Le MMS est un « store and forward » service, combinant ainsi les avantages que peut avoir un email (qu'on peut stocker) tout en pouvant être délivré immédiatement à son destinataire qui n'a pas à aller « relever » sa boîte.

Le MMS est envoyé au travers du protocole WAP et tire pleinement avantage des hautes vitesses de transmission permises par le GPRS, EDGE ou UMTS. Les MMS passent par un centre de service MMS (MMS-C) qui peut notamment exécuter certaines tâches limitées de conversion, comme la conversion du MMS en SMS afin de ne pas envoyer inutilement – en utilisant aussi inutilement la capacité du réseau - un MMS à un utilisateur équipé d'un terminal ne pouvant lire que des SMS. Le MMS est simple d'utilisation, l'utilisateur final n'a pas à configurer manuellement son mobile, la configuration pouvant être faite directement au niveau de l'opérateur.

### **III.2. La visiophonie :**

Un terminal compatible 3G intégrant une caméra suffit pour communiquer par la voix et l'image avec des correspondants également équipés. Ces téléphones sont dotés d'un bouton spécifique pour les appels en visiophonie, le numéro reste le même que pour une communication classique. L'écran du mobile diffuse l'image de votre correspondant, tandis que vous pouvez aussi visualiser en incrustation votre propre image. Il sera également possible de laisser un message vidéo à la personne appelée, en l'enregistrant auprès d'un service dédié, similaire en tout point à la messagerie vocale, ou en envoyant un MMS vidéo.

### **III.3. Le service de géolocalisation :**

De façon générale, la géolocalisation permet aux opérateurs de téléphonie mobile de fournir à un utilisateur des informations liées à son positionnement et de lui proposer en temps réel des services de plus en plus ciblés. L'enjeu est stratégique car il concerne tant les professionnels que le grand public.

En pratique, l'opérateur localise l'abonné via son portable pour sélectionner les informations envoyées sur son écran. Le service géolocalisé suppose la possibilité de tracer l'utilisateur d'un téléphone mobile afin de lui proposer des services personnalisés et de proximité en liaison avec sa position géographique.

A l'heure actuelle, il existe quatre techniques de géolocalisation des mobiles :

\_ L'identification de cellules : les téléphones mobiles communiquent à chaque instant avec une antenne reliée qui couvre une cellule. L'opérateur relève de quelle antenne relais le mobile dépend et relève ses coordonnées.

\_ La triangulation : le portable est en liaison permanente avec l'antenne de sa cellule et communique également avec les cellules voisines. Sa distance par rapport à trois antennes relais est déterminée en fonction de l'intensité des signaux radio-électriques qu'il reçoit. Puis ces données sont transmises à un terminal externe qui calcule sa position géographique. C'est la technique utilisée par les services d'urgence.

\_ Le différentiel temps : le téléphone émet un signal vers les antennes environnantes puis calcule le temps que met cette onde pour revenir tel un écho ou encore les antennes envoient un signal au téléphone et calcule ce temps.

\_ La localisation par GPS : une constellation de satellite envoie des signaux vers des téléphones mobiles qui les captent et peuvent en déduire leur propre position. Cette technique implique d'intégrer une puce GPS dans les portables. Observons que la longueur d'onde utilisée par les satellites ne passe pas à travers les bâtiments.

Cependant, les opérateurs de téléphonie sont indécis sur la technologie à employer pour ce qui concerne la géolocalisation car, parmi les différentes solutions disponibles, toutes ne présentent ni le même degré d'investissement, ni la même fiabilité dans le résultat.

En revanche, comme il n'existe aucune normalisation des technologies de localisation, l'utilisation de ces services risque d'être limitée sur le plan international en fonction de la technologie retenue dans chaque pays. C'est la raison pour laquelle le LIOF (Location Inter Operability Forum) a été créé dans le but de parvenir, si possible, à une standardisation des protocoles.

Toutefois ce type de services, qui concerne des millions de personnes compte tenu du nombre de détenteurs de téléphone portable, présente de réels dangers en matière de protection des données personnelles, de la vie privée et de liberté d'aller et venir, c'est pourquoi avant de proposer ou de souscrire ce type de services il convient de prendre connaissance des obligations légales que doivent respecter les services de géolocalisation.

### **III.3.1. Types de services de géolocalisation : [11]**

Les services géolocalisés peuvent être classés selon la manière d'accéder à l'information. On y trouve deux façons « *pull* » et « *push* »

➤ Les applications « *pull* » :

Les applications de type **pull** qui, comme leur nom anglais l'indique, consistent à récupérer des informations géolocalisés depuis le serveur à la demande du client. C'est donc l'utilisateur qui initie le transfert d'informations depuis le serveur. Les services de type « *pull* » sont le tracking de marchandises, de personnes ...etc.

➤ Les applications « *push* » :

Les applications du type **push** envoient des informations au client sans qu'il en fasse la demande explicite, par exemple lorsqu'il arrive dans une certaine zone géographique. Cependant, dans les applications de type « *push* », afin que le serveur connaisse de manière constante la position de l'utilisateur, il est nécessaire que celui-ci mette régulièrement à jour sa position. Les coûts en bande passante (communications entre les clients ou entre la plateforme et les services) et en puissance de calcul (calcul régulier de la distance des utilisateurs par rapport à une certaine zone) sont plus importants que les services de type « *pull* ». C'est pourquoi, il n'est pas encore très clair si ce genre de services est économiquement rentable. Les services du type « *push* » sont par exemple l'avertissement des enfants quittant leur école, la publicité quand on rentre dans un magasin ou encore les alertes d'incendies.

### **III.3.2. Positionnement de personnes et d'appareils : [11]**

Dans les services géolocalisés, on peut vouloir localiser soit des objets, soit des personnes. La localisation d'un objet se fait en localisant un appareil géolocalisable couplé à l'objet. Par exemple dans le cas de tracking de marchandises, on ne s'intéresse pas à savoir où se trouve le chauffeur mais bien où se trouve le camion puisque la personne qui conduit le camion n'est pas pertinente pour un tel système. De manière similaire, dans un système de localisation de voiture, généralement utilisé pour retrouver une voiture volée, ce n'est pas le conducteur dont on souhaite connaître la position mais bien le véhicule.

La localisation d'un appareil peut correspondre à une personne mais ce n'est pas une obligation.

Dans le cas de la localisation de personnes, on souhaite connaître la position d'une personne en particulier et celle-ci garde généralement un certain contrôle sur cette localisation en ayant la possibilité de ne pas faire connaître sa position. Cette notion est également utile si on imagine qu'une personne peut posséder plusieurs appareils de positionnement et que par exemple, il en oublie un. Il faudra dans ce cas être capable de localiser la personne et non pas les appareils qui donneront des localisations différentes.

### **III.3.3. Services différents selon la position : [11]**

Une utilité des services géolocalisés est la capacité d'offrir des services différents en fonction de la position de l'utilisateur ou de lui fournir des données différentes en fonction de cette position. Donner la liste des restaurants à proximité, obtenir automatiquement des offres de réduction en entrant dans un magasin. Dans un musée, un visiteur pourrait automatiquement recevoir des informations sur l'œuvre d'art qu'il a devant soi. Dans une grande surface, un client peut être conduit jusqu'à un article ou un produit de consommation.

Les applications de géolocalisation sont omniprésentes dans la vie quotidienne ainsi que le monde professionnel, dans des domaines comme : le transport de marchandise et logistique, propreté urbaine et assainissement, transport de passagers, suivi et protection de personnes, protection de marchandises et véhicules, publicité et marketing, etc.

La géolocalisation dans le monde professionnel est synonyme de gain de productivité, économie de carburant, communication et sécurité accrue.

Cela permet à l'entreprise utilisant un système de géolocalisation d'améliorer son service client et de réduire ses coûts afin d'accroître sa compétitivité.

Un système de géolocalisation permet d'afficher les informations relatives aux déplacements de l'objet géolocalisé mais en plus, il donne la possibilité de les renouveler en temps réel et d'en assurer le suivi.

Pour cela un système de géolocalisation a une architecture logicielle bien spécifique, ce que nous allons détailler dans ce qui suit.

#### **IV. Plateformes logicielles de géolocalisation :[12]**

##### **IV.1. Composants d'une plateforme logicielle de géolocalisation :**

Les composants essentiels d'une plateforme de géolocalisation sont les suivants :

- Terminal communicant : C'est le terminal qui reçoit ses coordonnées géographiques (via GPS ou tout autre moyen) et qui les envoie via un réseau de télécommunications à la plateforme;
- Système informatique capable de recevoir, stocker et traiter les informations : il s'agit des serveurs informatiques qui hébergent l'infrastructure et qui reçoivent et traitent les données envoyées par les terminaux. Ce sont les mêmes serveurs qui mettront à disposition l'information aux utilisateurs (via une interface web par exemple);
- Module cartographique : c'est le module intégré au système informatique qui va permettre d'afficher la position des terminaux sur un fond cartographique adapté. Ce module prend en charge les calculs de distances, d'itinéraires, détecte l'interaction avec les zones et permet d'avoir accès à des informations terrain (sens interdits, restrictions pour les poids lourds, vitesses autorisées...).

##### **IV.2. Architecture d'une plateforme logicielle de géolocalisation :**

La position géographique d'un terminal géolocalisé reste néanmoins une information brute qui peut être exploitée et couplée à d'autres données afin de créer une vaste quantité de services à forte valeur ajoutée.

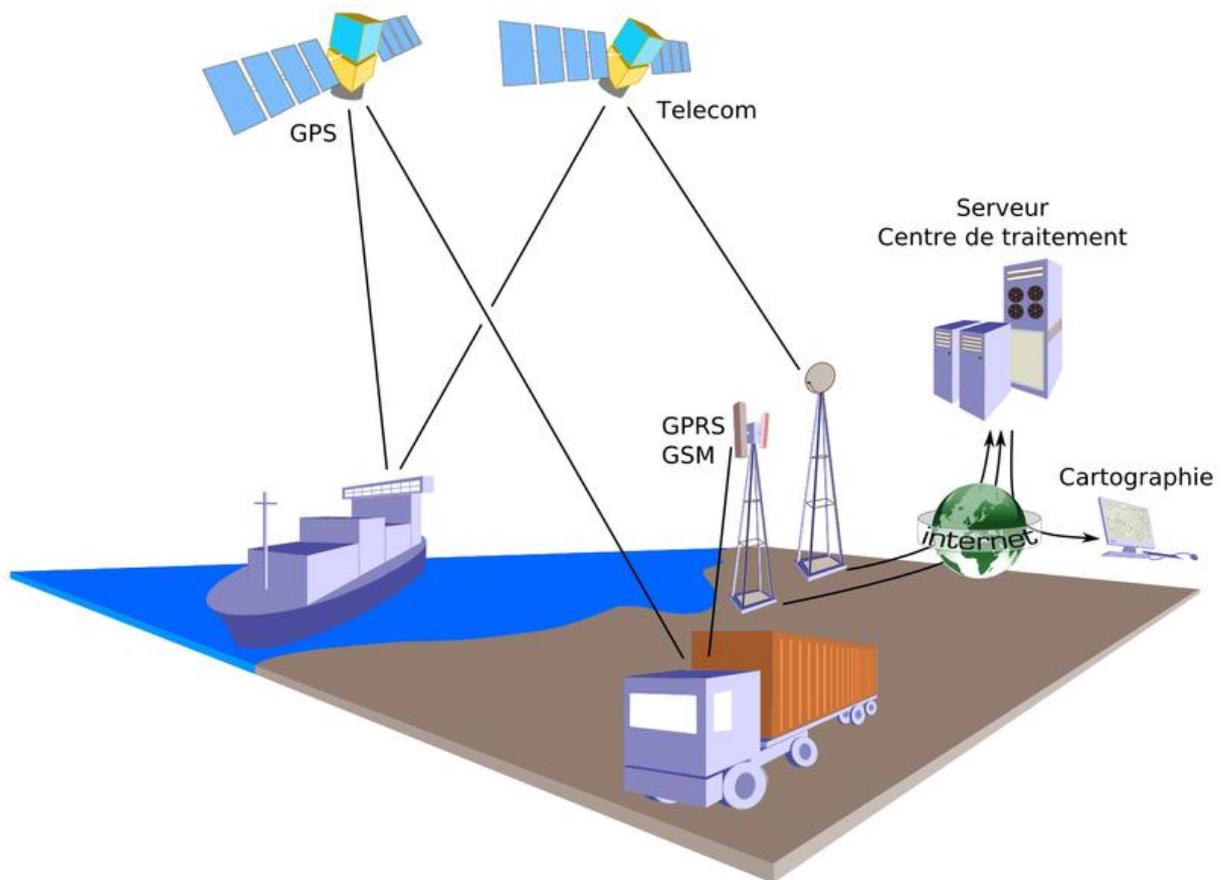
Afin d'exploiter cette information, la donnée (position) générée par un terminal qui se trouve sur le terrain doit être transmise à une plateforme logicielle qui va la traiter, la présenter graphiquement à l'utilisateur et l'associer à d'autres données afin d'enrichir les informations relatives à l'état du terminal ou de la flotte de terminaux.

Voici les étapes de la chaîne de traitement :

- Le terminal détermine sa position géographique grâce à une des techniques de géolocalisation citées précédemment (de préférence GPS, GSM et/ou Wifi);
- Il envoie ces données vers une plateforme logicielle soit par le réseau GSM/GPRS soit par un réseau satellitaire.
- La plateforme logicielle de géolocalisation traite la donnée et positionne le terminal géographiquement sur une carte moyennant la précision offerte par la technique de géolocalisation utilisée. De plus, en combinant plusieurs informations, notamment récupérées via un système de télé relèvement (trafic routier, autonomie du véhicule, points à visiter etc...), des calculs d'itinéraires ou de tournées peuvent par exemple être générés.

Cette carte ainsi que tous les traitements effectués sont mis à disposition de l'utilisateur à travers un portail web hébergé sur un serveur accessible depuis internet, ou à travers une application métier installée sur le poste de travail.

Pour transmettre les différentes informations récupérées par le terminal (position géographique ou données provenant de capteurs) il existe deux principaux moyens de transmission : le réseau GSM/GPRS et le réseau satellitaire. Les architectures types illustrant ces deux modes de transmission seront visualisé dans la section suivante.



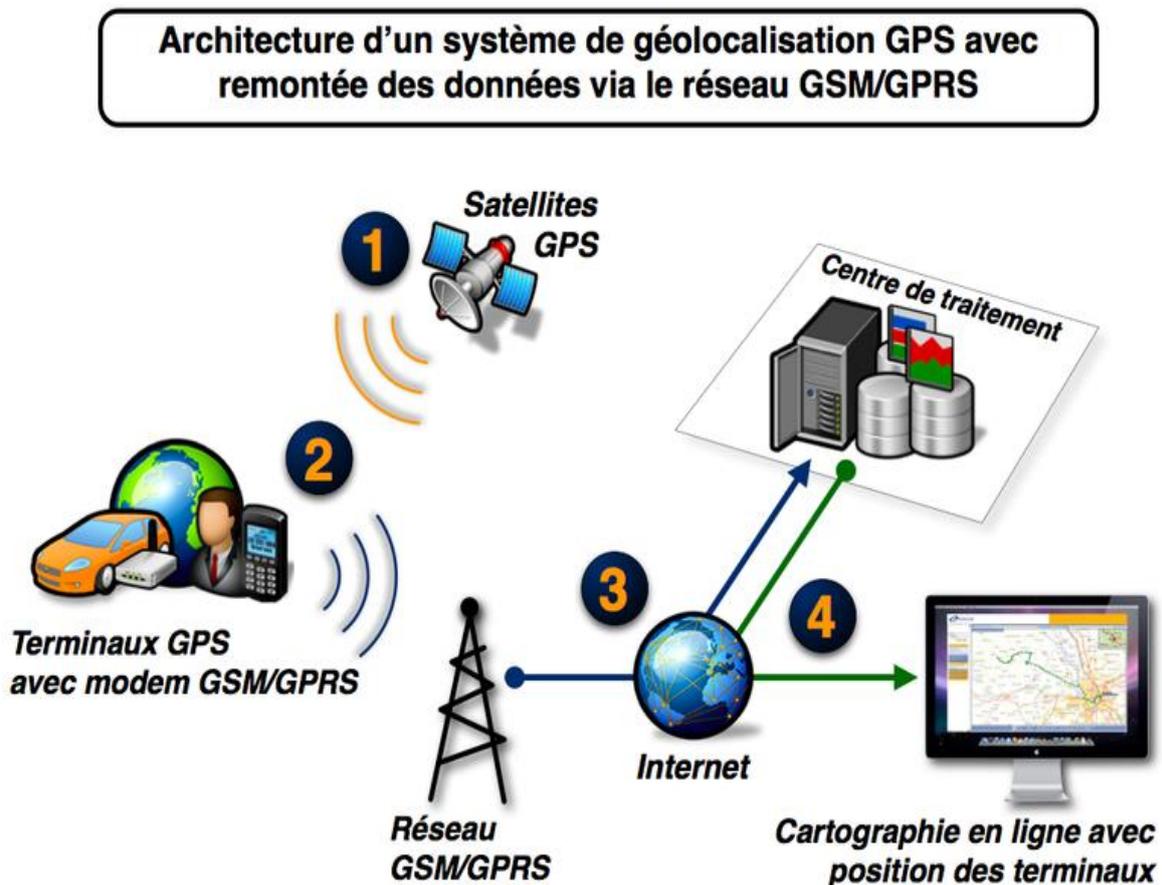
**Figure2.1** : Architecture d'une plateforme logicielle de géolocalisation.

### IV.3. transmission des informations : [12]

Les terminaux peuvent se connecter pour transmettre leurs coordonnées géographiques selon différentes techniques. Selon le terminal choisi, il existe deux manières de procéder : le GPRS et le réseau satellitaire.

#### a- La transmission par GSM/GPRS :

Le GPRS, ou connexion 3G est utilisée par les téléphones portables. Elle est économique mais peut présenter des problèmes de couverture selon la zone où l'on se trouve (donc pas d'informations transmises). Les informations récupéré par le terminal placé au sein du véhicule sont transférées via le réseau GPRS de l'opérateur et sont récupérées par le serveur qui les renvoie ensuite vers les ordinateurs des clients via une connexion Internet sécurisée. Les données sont ensuite directement consultables par les gestionnaires de flotte depuis un ordinateur ou autres terminaux.



**Figure2.2:** Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau GSM/GPRS.

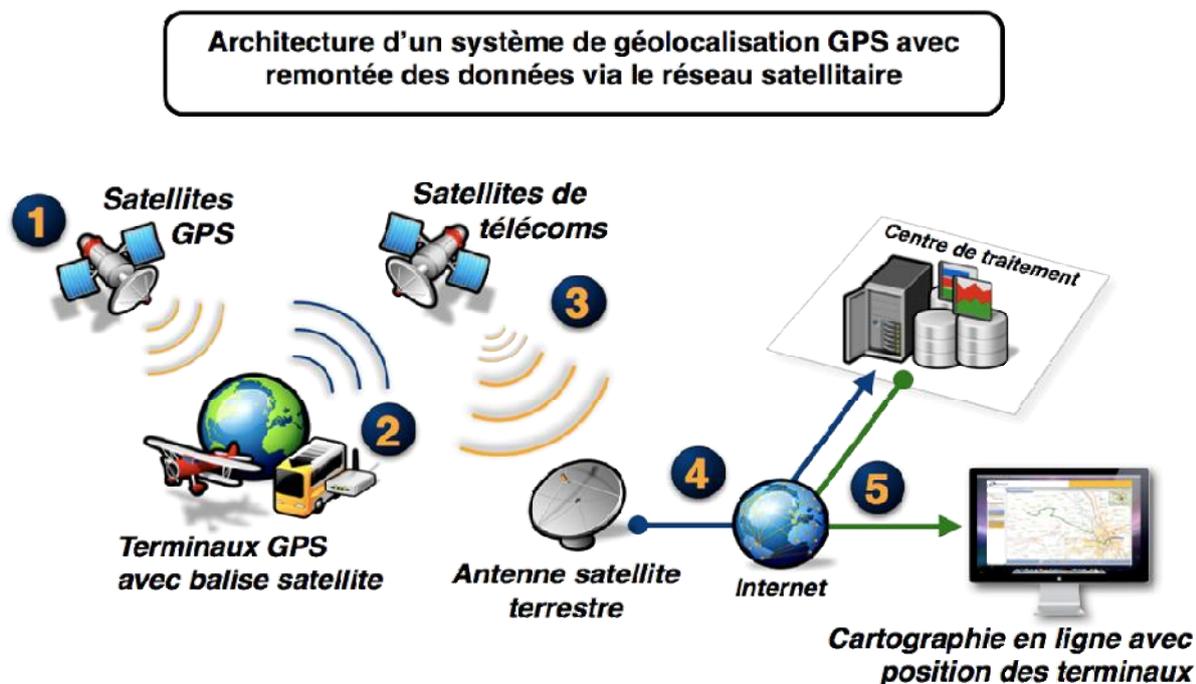
Ce moyen de transmission nécessite un terminal disposant d'un modem GSM/GPRS ou 3G/UMTS, ainsi que d'une carte SIM de n'importe quel opérateur avec un forfait "données" (Data) adapté. Le terminal nécessite d'être sous couverture GSM/GPRS pour pouvoir envoyer les données vers la plateforme de traitement. Ce type de terminal est utilisé lorsque l'objet ou la personne à géolocaliser reste dans une zone bien couverte par les réseaux GSM/GPRS.

Les forfaits GSM/GPRS sont économiquement plus avantageux que les forfaits satellite lorsque l'on souhaite remonter les positions à une fréquence élevée. Ils sont donc à privilégier si les zones, où l'équipement se déplace, restent bien couvertes par les réseaux GSM/GPRS.

**b- Terminaux satellite :**

La transmission par satellite est plus coûteuse, car elle sollicite une connexion avec appareil de télécommunication. Cependant, il faut privilégier ce type de connexion lorsque que l'on a besoin d'une couverture internationale à tout moment et de bonne qualité. En effet, une géolocalisation via un satellite est possible même en plein désert.

Ces types de terminaux envoient les données à travers un réseau de satellites de télécommunications comme Inmarsat. Même si ces types de canaux sont plus restrictifs au niveau de la quantité de données envoyée, ils peuvent offrir par ailleurs une couverture mondiale sans frais supplémentaires en fonction des satellites et protocoles utilisés. Cela en fait des terminaux idéaux pour la géolocalisation de conteneurs, navires ou véhicules circulant en plein désert.



**Figure2.3** : Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau GSM/GPRS.

Il faut donc bien choisir le type de terminal et de connexion en fonction de la solution en géolocalisation dont vous avez besoin.

#### **IV.4. Types de terminaux existants : [12]**

Les terminaux de géolocalisation existants peuvent être classés en trois catégories, comme citer précédemment le type de service offert par la plateforme est différent selon la manière d'accès à l'information, donc selon le terminal utilisé. Nous y trouvons des data pullers, data pushers et les data loggers, néanmoins certains terminaux peuvent être configurés pour fonctionner dans un mode au choix :

- **Les data loggers:** Ils sont équipés d'un GPS et servent à stocker la position GPS sur un intervalle régulier. Ils peuvent être connectés à un ordinateur pour un transfert de données. Ces terminaux peuvent aussi être utilisés dans le cadre de surveillances ou de suivi de véhicules.
- **Les data pullers :** Ils n'enregistrent pas les positions GPS dans leurs bases, ils se limitent à repérer la position et la communique sur demande à distance. Idéal pour protéger un véhicule contre le vol, le propriétaire peut à tout moment être en communication avec le traceur et recevoir immédiatement par SMS la position GPS du véhicule. Ce système ne nécessite pas d'abonnement, seulement l'emploi d'une carte SIM avec un forfait SMS. Le paiement ne se fait que s'il y a demande de position.
- **Les data pushers:** Ce sont les plus utilisés par les logiciels de géolocalisation pour des applications professionnelles. Ces terminaux envoient les positions vers une plateforme de géolocalisation qui va traiter les données en temps réel. Ils sont munis de traceurs GPS et de modems GSM sous forme de carte SIM.

#### **IV.5. Fonctionnalités offertes par les plateformes de géolocalisation : [W1]**

Les plateformes de géolocalisation professionnelles offrent divers fonctionnalités comme :

- Visualisation de la position de l'ensemble du parc géolocalisé.
- Suivi en temps réel de terminaux.
- Affichage d'un historique des déplacements.
- Création de points d'intérêt.
- Création de zones géographiques (pour le geofencing) et de routes (corridoring).

- Configuration d'alertes automatiques par courriel ou SMS sur des événements.
- Paramétrage d'événements (entrée/sortie de zone, dépassement de vitesse, temps d'arrêt...).
- Guidage sur tournée.
  
- Envoi de commandes vers le terminal et configuration à distance (notamment pour changer la fréquence d'émission de positions).
- Génération de rapports périodiques (temps de conduite, arrêts, vitesses moyennes, zones couvertes...).
  
- Fonds cartographiques variables (cartes classiques, cartographie photographique, cartes de fonds marins, cartes provenant d'un SIG).
  
- Configuration d'alertes sur capteurs distants (via télé relève).
  
- Détection de mouvement.

### **V. Les services mobiles pour entreprise :**

L'exploitation des services mobiles au sein des entreprises est de plus en plus importante, cela lui impose de mettre en œuvre des moyens et technologies de plus en plus sophistiquées. L'automatisation de certaines de ces activités, et le déploiement de moyens sur le plan technologique (choix des terminaux mobiles, technologies déployés ....) et sur le plan des applications proposées aux employés lui permet de répondre aux contraintes de productivité et de rentabilité.

La géolocalisation des véhicules (parc d'automobile) au sein d'une entreprise utilise pour la plupart des cas le système de navigation GPS embarqué dans les véhicules (GPS embarqué, Smartphone embarqué). On peut ainsi situer depuis son ordinateur de bureau les emplacements de chaque automobile, vérifier leurs déplacements et grâce à ces indications, il devient possible d'optimiser leurs affectations géographiques. Cela permet par exemple à une société de surveillance d'arriver plus vite à un lieu d'intervention en choisissant le véhicule le plus proche, de sécuriser le trajet d'un véhicule lors d'une livraison en le suivant à la trace, etc.

#### **V.1. Caractéristiques des applications pour entreprises :**

- beaucoup d'applications hybrides, à la fois mobiles et classiques ;
- peu d'applications mobiles lourdes (utilisation intensive du SMS) ;
- importance stratégique des activités de messagerie (bureau mobile) ;
- majorité de terminaux sophistiqués (Smartphones, PDA, ordinateurs portables).
-

### **a) les gains**

- diminution des coûts et amélioration de l'efficacité ;
- valorisation du personnel ;
- satisfaction du client

### **b) Les freins**

- coût des applications ;
- choix et gestion des types de terminaux mobiles pour les employés (technologies rapidement dépassées) ;
- sécurité et confidentialité des données ;
- changement des habitudes de travail ;
- coût de la formation.

## **VI. Conclusion :**

A travers ce chapitre, nous avons vu que les services mobiles ont connu une évolution remarquable dans le monde, à titre d'exemple le service de géolocalisation qui est devenu un service pratique comme bien pour les professionnels (entreprise, particulier) ainsi que le grand public.

Dans le cas de notre travail nous nous intéressons plus particulièrement à la géolocalisation d'un ou plusieurs véhicules, ceci peut intéresser toute personne ou entreprise ayant à gérer un parc automobile.

CHAPITRE

---

# III

## **Analyse & Conception**

## 1. Introduction :

Dans les chapitres précédents, nous avons défini le domaine de notre système (localisation d'une flotte de véhicules) et les technologies qui participent à sa mise en œuvre. Ce chapitre sera consacré à l'analyse et la conception de notre système.

Pour analyser les besoins du système, on s'est posé deux questions :

- Quels sont les acteurs qui interviennent ?
- Quels sont les objectifs de notre système ?

Dans ce qui suit on présentera les étapes d'analyse et de conception de l'application en faisant appel aux diagrammes UML (Unified Modeling language), qui permet de bien représenter la dynamique d'une application par la série des diagrammes qu'il offre.

## 2. Présentation d'UML:[17]

UML (Unified Modeling Language) est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant : OMT (Object Modeling Technique), Booch et OOSE (Object Oriented Software Engineering).

UML n'est pas une méthode dans la mesure où elle ne présente aucune démarche. A ce titre, c'est un formalisme de modélisation objet. A partir de 1997, il est devenu une norme de l'OMG (Object Management Group), ce qui lui a permis de s'imposer en tant qu'approche de développement objet et être reconnue et utilisée par de nombreuses entreprises.

UML 2.0 définit treize types de diagrammes divisés en trois catégories :

- Diagramme de structure.
- Diagramme d'interaction.
- Diagramme de comportements.

### 2.1. Définition :

UML (Unified Modeling Language traduit en « langage de modélisation objet unifié ») est un langage de description orienté objet qui permet de modéliser une application selon une vision objet. Un objet est décrit par les attributs qui le compose et les traitements, appelé méthodes, qui peuvent lui être appliqués.

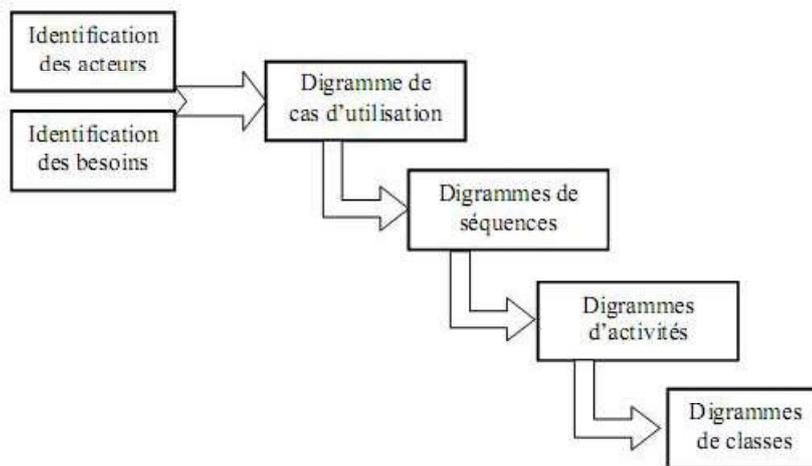
### 2.2. Modélisation avec UML :

UML permet de représenter des modèles, mais il ne définit pas de processus d'élaboration de modèles. Les auteurs d'UML conseillent tout de même une démarche pour favoriser la réussite d'un projet, cette démarche doit être :

- **Une démarche itératif et incrémentale** : Pour comprendre et représenter un système complexe, pour analyser par étapes, pour favoriser le prototypage et pour réduire et maîtriser l'inconnu.
- **Une démarche guidée par les besoins des utilisateurs** : Tout est basé sur le besoin des utilisateurs du système, le but du développement lui-même est de répondre à leur besoin. Chaque étape sera affinée et validée en fonction des besoins des utilisateurs.
- **Une démarche centrée sur l'architecture logicielle** : c'est la clé de voute de succès d'un développement, les choix stratégiques définiront la qualité du logiciel.

### 2.3. La démarche de modélisation avec L'UML :

La démarche de modélisation choisie pour concevoir mon application peut être représentée graphiquement comme suite :



**Figure3.1.** : La démarche adoptée pour la modélisation.

## 3. Analyse :

La partie analyse est divisée en deux parties, analyse fonctionnelle et analyse technique de notre système. Ainsi que la spécification des besoins qui sont des besoins fonctionnels et techniques.

Cette partie comprend l'identification des besoins fonctionnels et non fonctionnels du système, des acteurs et leurs interactions avec le système ainsi que les cas d'utilisation.

### 3.1. Spécification et analyse des besoins :

La spécification des besoins représente la première phase du cycle de développement d'un logiciel. Elle sert à identifier les acteurs réactifs du système et associer à chacun l'ensemble d'actions avec lesquelles il intervient, dans l'objectif de donner un résultat optimal et satisfaisant au client.

Dans ce qui suit on commencera par définir l'objectif du système à réaliser, puis on essaiera d'identifier les besoins fonctionnels et techniques de notre système.

### 3.1.1. Objectif du système :

L'objectif principal du système est de garantir aux utilisateurs de la plateforme un service permettant la localisation et la traçabilité de leurs véhicules en temps réel pour une meilleure gestion de leurs flottes.

Il est donc nécessaire de concevoir une base de données qui doit contenir l'ensemble des données sur les utilisateurs, les employés mobiles, les véhicules ainsi que les coordonnées spatiales des véhicules sur route.

Les principaux modules fonctionnels à développer dans ce projet sont :

- La gestion des utilisateurs.
- La gestion des véhicules et de leurs trajets.
- Le paramétrage des fonctionnalités de la plateforme.
- La gestion des employés mobiles.
- La consultation de la localisation en temps réel via la carte géographique.
- La consultation de l'historique.

### 3.1.2. Identification des acteurs :

Notre système communique avec des acteurs afin de réaliser un ensemble d'opérations. Un acteur représente un ensemble de rôles joués par des entités externes (utilisateur humain, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Il peut consulter et/ou modifier directement l'état du système, en émettant et/ou recevant des messages susceptible d'être porteur des données. L'acteur est associé à un cas d'utilisation, c'est-à-dire qu'il peut interagir avec lui et participer à son scénario. Les acteurs candidats sont :

Les utilisateurs humains directs :

- **Administrateur du système** : responsable de la gestion de la plateforme et de ses utilisateurs. Il s'occupe principalement des tâches suivantes :
  - ✓ Valider l'inscription d'un client.
  - ✓ Mettre à jour les informations et les coordonnées de ses clients : ajouter/ supprimer un utilisateur,

- ✓ Mettre à jour les informations de la flotte : ajouter/supprimer /modifier les caractéristiques des véhicules (marque, type -marque).
- ✓ Contacter le client.

- **Visiteur de la plateforme** : personne qui se connecte au système pour pouvoir s'informer des différentes fonctionnalités qu'offre la plateforme.

L'ensemble d'action qu'un visiteur peut effectuer est :

- ✓ Accéder à l'application.
- ✓ Consulter les différentes fonctionnalités de l'application.
- ✓ Faire une inscription en tant que client.
- ✓ Contacter l'administrateur.

- **utilisateur de la plateforme « user »** : personne inscrite qui se connecte pour pouvoir gérer une flotte de véhicules, à savoir l'état des trajets, le déroulement des missions, le suivi et le tracking de la flotte.

L'application doit lui permettre de :

- ✓ Authentification.
- ✓ Ajout d'un véhicule.
- ✓ Ajout d'un chauffeur.
- ✓ Création de zones.
- ✓ Affectation d'un véhicule à un chauffeur.
- ✓ Affectation d'un véhicule à une zone.
- ✓ Affectation d'un paramètre à un véhicule.
- ✓ Suivre les véhicules en temps réel.
- ✓ Avoir les informations sur la vitesse, les traces et la destination.
- ✓ consulter l'historique des véhicules.

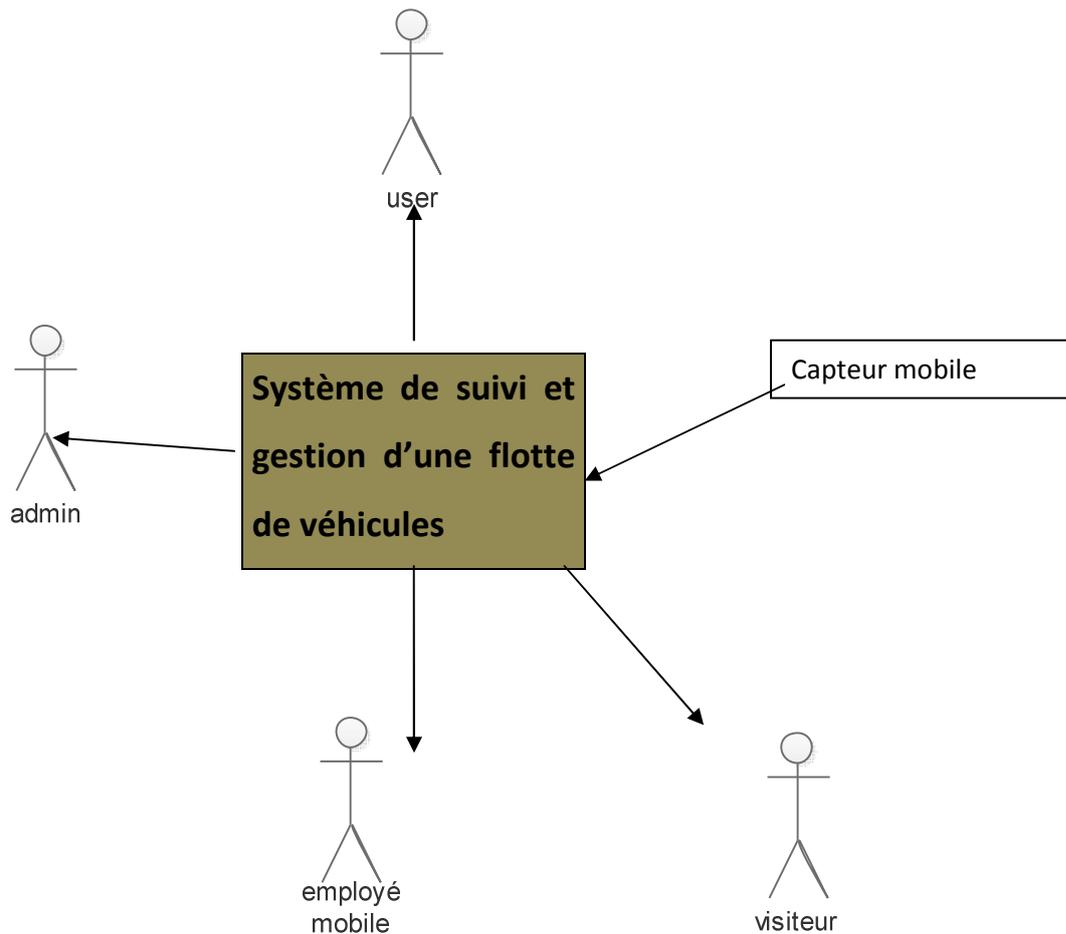
- **Employé mobile** : personne à bord du véhicule, elle avertit le système de l'état de sa tournée en se servant d'un Smartphone qui permet la récupération de la position ainsi que la vitesse du véhicule.

Les autres systèmes connexes qui interagissent aussi directement avec le système :

- **Capteur mobile** : Envoie automatiquement et périodiquement sa localisation au système.

### 3.1.3. Diagramme de contexte :

Le diagramme de contexte est un modèle conceptuel de flux qui permet d'avoir une vision globale des interactions entre le système et les liens avec l'environnement extérieur. Il permet aussi de bien délimiter le champ d'étude. Pour notre cas le diagramme de contexte est donné par la figure suivante :



**Figure3.2.** : diagramme de contexte du système.

### 3.2. Spécifications des tâches :

#### 3.2.1. Définition :

Une tâche est l'ensemble des différentes fonctions qui peut être accédés pas un acteur bien spécifique.

Les acteurs définis précédemment effectuent un certain nombre de tâches, ces tâches sont résumées dans le tableau ci-dessous :

Acteurs	Tâches
<b>Visiteur</b>	T0 : se connecter à la plate-forme (accéder à la page d'accueil) ; T1 : Consulter les informations sur les fonctionnalités de la plate-forme ; T2 : s'inscrire en tant que client.
<b>User</b>	T3 : S'authentifier. T4 : localiser un véhicule sur la Map. T5 : Consulter L'historique des trajets. T6 : consulter les alertes T7 : attribuer une sanction pour l'employé mobile. T8 : gérer les véhicules. T9 : gérer les chauffeurs. T10 : Ajouter zone. T11 : paramétrer l'application. T12 : affectation paramètre/véhicule. T13 : affectation véhicule/zone. T14 : affectation véhicule/chauffeur.
<b>Administrateur</b>	T15 : Ajouter marque véhicule. T16 : Ajouter type marque véhicule. T17 : valider inscription.

**Tableau3.1.** : Spécification des tâches.

### 3.3. Spécification des scénarios :

#### 3.3.1. Définition :

Un scénario est une instance (occurrence) d'un cas d'utilisation, à chaque fois qu'une instance d'un acteur déclenche un cas d'utilisation, un scénario est créé. Ce scénario suivra un chemin particulier dans le cas d'utilisation.

**NB :** Durant notre étude les scénarios seront symboliser par **Si** (**i** : représente le numéro du scénario).

Le tableau ci-dessous récapitule les différents scénarios créés par les cas tâches citées auparavant.

Acteurs	Taches	Scénarios
Visiteur	T01 : consulter les fonctionnalités du système	S01 : accéder à l'application S02 : Choisir dans le menu « fonctionnalité »
	T02 : inscription	S01, S03 : choisir dans le menu « inscription » S04 : remplir un formulaire d'inscription et le soumet.
	T03 : s'authentifier	S01, S05 : Choisir dans le menu « authentification » S06 : remplir un formulaire d'identification et le transmet
	T04 : localiser un véhicule sur la Map	S01, S05, S06 S07 : choisir dans le menu « Map » S08 : insérer le matricule et temps de suivi.
	T05 : Consulter l'historique	S01, S05, S06 S09 : choisir dans le menu « historique » S10 : définir le matricule du véhicule et heure début, heure fin
	T06 : consulter les alertes	S01, S05, S06 S11 : Choisir dans le menu « alerte » S12 : définir un critère de choix pour visualiser la liste des alertes.
	T07 : attribuer une sanction	S01, S05, S06 S13 : identifier l'alerte générée. S14 : identifier l'employé à sanctionné. S15 : associer une observation à l'employé sanctionné.

<b>User</b>	T08 : ajouter un véhicule	S01, S05, S06 S16 : choisir dans le menu « véhicule » S17 : remplir un formulaire d'ajout d'un véhicule et transmettre.
	T09 : ajouter un chauffeur	S01, S05, S06 S18 : choisir dans le menu « chauffeur » S19 : remplir un formulaire d'ajout d'un chauffeur et transmettre.
	T10 : Ajouter zone	S01, S05, S06 S20 : choisir dans le menu « zone » S21 : remplir un formulaire d'ajout d'une zone et transmettre.
	T11 : Paramétrer Alerte	S01, S05, S06 S22 : choisir dans le menu « paramètre » et remplir le formulaire d'ajout de paramètres déclenchant les alertes et soumettre.
	T12 : affecter paramètre/véhicule	S01, S05, S06 S23 : choisir dans le menu « véhicule/paramètre » et affecter à un véhicule les paramètres fixés.
	T13 : affecter zone/véhicule	S01, S05, S06 S24 : choisir dans le menu « véhicule/zone » et affecter à un véhicule une zone.
	T14 : affecter chauffeur/véhicule	S01, S05, S06 S25 : choisir dans le menu « véhicule/chauffeur » et affecter a un véhicule un chauffeur.
<b>Admin</b>	T15 : ajouter marque véhicule	S01, S05, S06 S27 : choisir le menu « marque » remplir le champ marque et soumet.
	T16 : ajouter type marque véhicule	S01, S05, S06 S28 : choisir le menu « type marque » et remplir le formulaire d'ajout et soumet.
	T17 : valider inscription	S01, S05, S06 S29 : choisir dans le menu « utilisateur » S30 : choisir l'inscription à approuver

		et la valider.
--	--	----------------

**Tableau3.2.:** Spécification des scénarios.

### 3.4. Diagramme des cas d'utilisation:

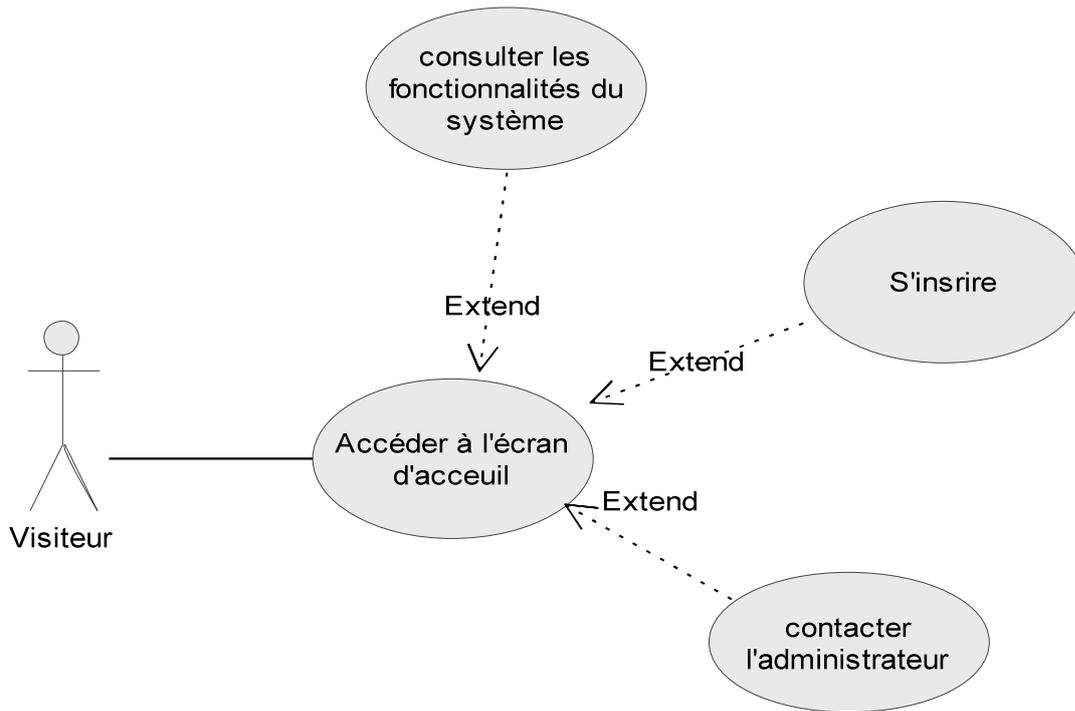
Précédemment nous avons pu identifier les acteurs ainsi que leurs tâches respectives. Ce qui nous donne l'opportunité d'élaborer les diagrammes des cas d'utilisation.

#### 3.4.1. Définition :

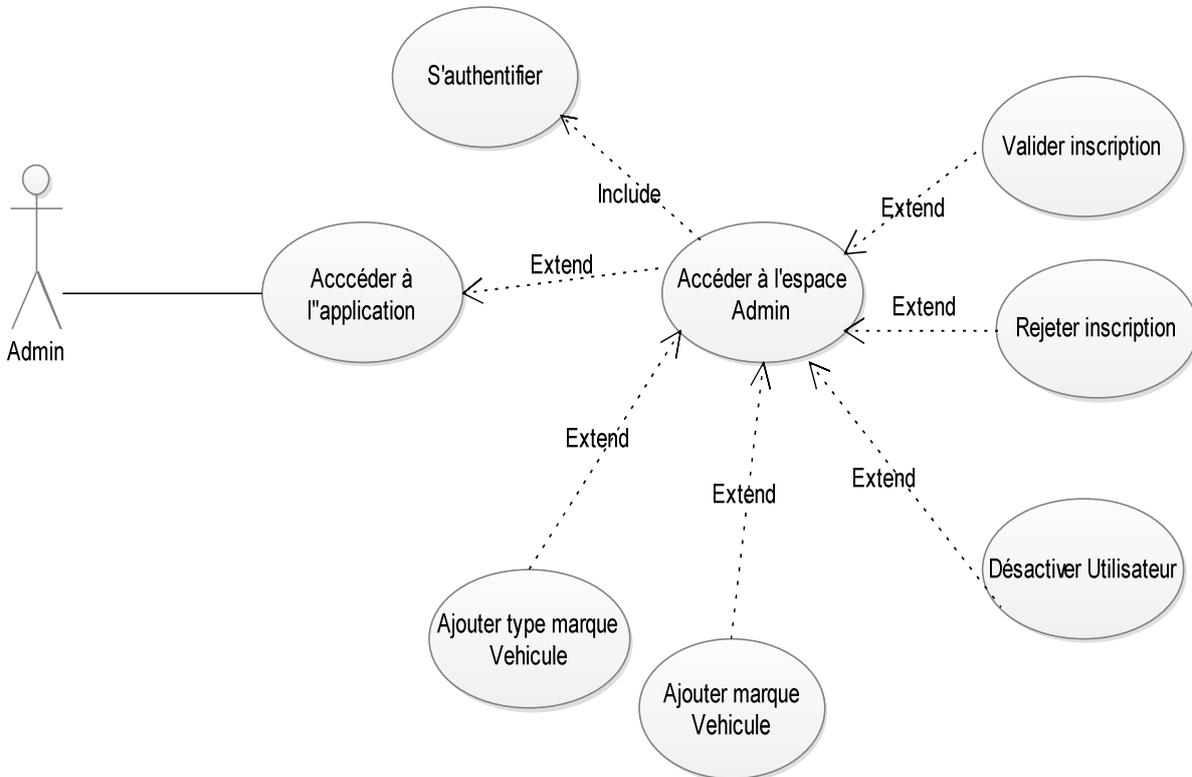
Les diagrammes de cas d'utilisation permettent de mettre en évidence les relations fonctionnelles entre les acteurs et le système étudié. Les cas d'utilisation sont décrits de manière textuelle, agrémentée de quelques diagrammes d'interaction. A ce stade de la modélisation, les interactions représentent les principaux événements qui se produisent dans le domaine de l'application. Plus tard, lors de la conception, ces événements sont traduits en messages qui déclenchent des opérations.

- **La relation d'inclusion (include) :** Elle indique que le cas d'utilisation source contient aussi le comportement décrit dans le cas d'utilisation destination. Cette relation permet de décomposer des comportements et de définir les comportements partageables entre plusieurs cas d'utilisations.
- **La relation d'extension (Extend) :** Elle indique que le cas d'utilisation source ajoute son comportement au cas d'utilisation destination. L'extension peut être soumise à des conditions.

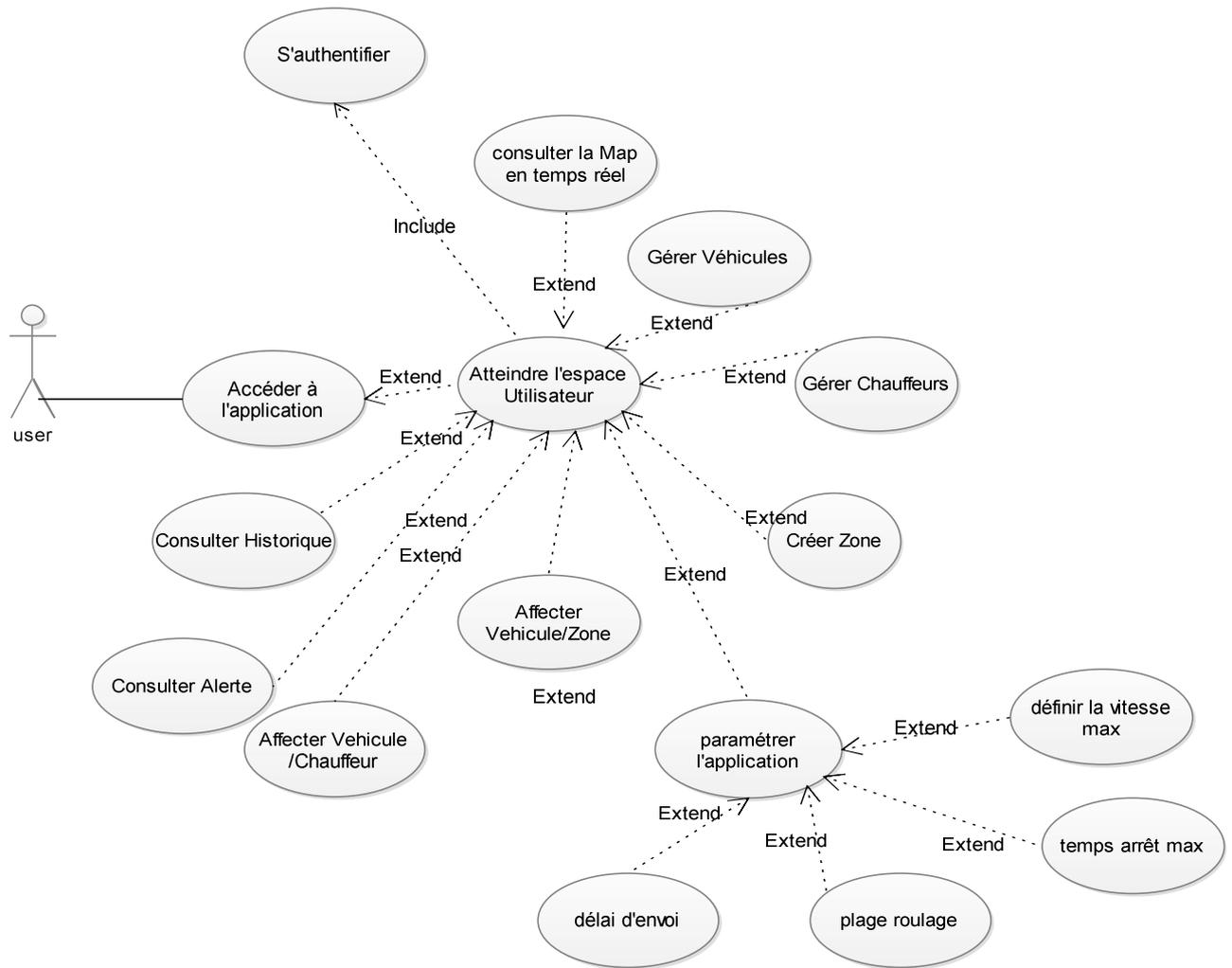
#### 3.4.2. Au niveau de l'application web :



**Figure 3.3.** : Diagramme de cas d'utilisation « Acteur : Visiteur ».

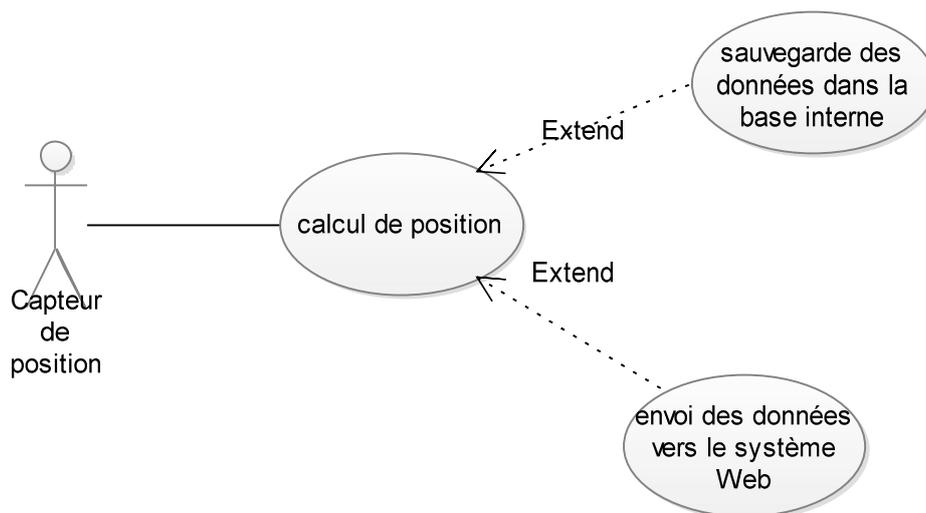


**Figure3.4.:** Diagramme de cas d'utilisation « Acteur : Administrateur ».



**Figure3.5.** : Diagramme de cas d'utilisation « Acteur : User ».

### 3.4.3. Au niveau de l'application mobile :



**Figure3.6.** : Diagramme de cas d'utilisation « Acteur : Capteur mobile ».

## 3.5. Description des cas d'utilisations :

Nous procéderons aux descriptions des cas d'utilisation dans notre système :

### 3.5.1. Au niveau de l'application web :

**Cas d'utilisation** : inscription

**Scenario** : S01, S03, S04

**Rôle** : visiteur

**Résumé : (objectif)**

- Se connecter à l'application
- avoir des droits d'accès en tant qu'utilisateur.

**Pré conditions**

- le visiteur doit être connecté à l'application.

**Description :**

1. L'utilisateur lance l'application.
2. Le système affiche la page d'accueil.
3. L'utilisateur choisi dans le menu « inscription ».
4. Le système affiche une page contenant un formulaire d'inscription ;
5. L'utilisateur remplis le formulaire et le transmet ;
6. Le système vérifie l'intégrité des données :
  - Si les informations sont correctes, le système accepte l'inscription et lui

retourne un message d'acceptation.

- Si non le système affiche un message d'erreur.

**Post-Conditions**

- une instance inscription est créée.

**Tableau3.3. :** Description de cas d'utilisation « Inscription ».

**Cas d'utilisation :** authentification

**Scenario :** S01, S05, S06

**Rôle :** tous les acteurs (sauf visiteur)

**Résumé : (objectif)**

- Se connecter à l'application
- Vérifier si l'utilisateur est autorisé à accéder à l'application. S'il est autorisé, le système vérifie ses privilèges.

**Pré conditions**

- Le login et le mot de passe de l'utilisateur doivent être connus par le système

**Description :**

- 1- L'utilisateur lance l'application.
- 2- Le système affiche l'écran d'accueil.
- 3- L'utilisateur choisi dans le menu « authentification »
- 4- Le système affiche une page contenant un formulaire d'identification ;
- 5- L'utilisateur saisi son nom d'utilisateur et mot de passe et soumet le formulaire ;
- 6- Le système vérifie ces informations et détermine s'il a le droit d'accéder à l'application.

- Si l'utilisateur n'est pas autorisé à accéder à l'application, indiquer un message d'erreur
- Si non il le dirige vers son espace de travail.

**Post-Conditions**

- une instance de Login est créée.
- Les droits de l'utilisateur sont récupérés.

**Tableau3.4. :** Description de cas d'utilisation « Authentification ».

<p><b>Cas d'utilisation :</b> localiser un véhicule sur la Map en temps réel.</p> <p><b>Scenario:</b> S01,S05,S06, S07, S08</p> <p><b>Rôle :</b> utilisateur</p> <p><b>Résumé :</b></p> <ul style="list-style-type: none"><li>• Se connecter l'application</li><li>• Visualiser la position du véhicule en tems réel.</li></ul> <p><b>Pré condition :</b></p> <ul style="list-style-type: none"><li>• L'utilisateur doit accéder à son espace.</li></ul> <p><b>Description :</b></p> <ol style="list-style-type: none"><li>1- l'utilisateur choisi dans le menu « temps réel GPS»</li><li>2- le système affiche la page Map.</li><li>3- l'utilisateur choisi le véhicule à suivre et la date début du suivi et soumet.</li><li>4- le système charge la carte et les positions du véhicule.</li></ol> <p><b>Post-Conditions :</b></p> <p>position du véhicule récupérée et affichée sur la carte.</p>
--

**Tableau3.5. :** Description de cas d'utilisation « localiser un véhicule sur la Map en temps réel ».

<p><b>Cas d'utilisation :</b> Ajouter véhicule</p> <p><b>Scenario :</b> S01, S05, S06, S16, S17</p> <p><b>Rôle :</b> Utilisateur</p> <p><b>Résumé :</b></p> <ul style="list-style-type: none"><li>• Se connecter l'application</li><li>• Ajouter un nouveau véhicule.</li></ul> <p><b>Pré condition :</b></p> <ul style="list-style-type: none"><li>• L'utilisateur doit accéder à son espace.</li></ul> <p><b>Description :</b></p> <ol style="list-style-type: none"><li>1- l'utilisateur choisi dans le menu « Véhicule »</li><li>2- le système lui affiche la page Véhicule.</li></ol>
--

- 3- l'utilisateur remplit le formulaire d'ajout d'un véhicule et soumet.
- 4- le système retourne un message pour confirmer l'ajout du véhicule si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau véhicule  
sinon, le système le retourne à son ancien formulaire.

**post -condition :**

- Une instance de véhicule est créée et enregistrée.

**Tableau 3.6.** : Description de cas d'utilisation « Ajouter Véhicule ».**Cas d'utilisation :** Ajouter chauffeur**Scenario :** S01, S05, S06, S18, S19**Rôle :** Utilisateur**Résumé :**

- Se connecter l'application
- Ajouter un nouveau chauffeur.

**Pré condition :**

- L'utilisateur doit accéder à son espace.

**Description :**

- 1- l'utilisateur choisi dans le menu « chauffeur »
- 2- le système lui affiche la page Chauffeur.
- 3- utilisateur remplit le formulaire d'ajout d'un Chauffeur et soumet.
- 4- le système retourne un message pour confirmer l'ajout du chauffeur si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau chauffeur  
sinon, le système le retourne à son ancien formulaire.

**post -condition :**

- Une instance de chauffeur est créée et enregistrée.

**Tableau 3.7** : Description de cas d'utilisation « Ajouter Chauffeur ».**Cas d'utilisation :** Ajouter zone**Scenario :** S01, S05, S06, S20, S21**Rôle :** Utilisateur**Résumé :**

- Se connecter l'application
- Ajouter une nouvelle zone.

**Pré condition :**

- L'utilisateur doit accéder à son espace.

**Description :**

- 1- l'utilisateur choisi dans le menu « zone »
- 2- le système lui affiche la page Zone.
- 3- L'utilisateur rempli le formulaire d'ajout d'une zone et soumet.
- 4- le système retourne un message pour confirmer l'ajout de la zone.  
si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant la nouvelle zone.  
sinon, le système le retourne à son ancien formulaire.

**post -condition :**

- Une instance de zone est crée et enregistrée.

**Tableau3.8. :** Description de cas d'utilisation «Ajouter Zone».

**Cas d'utilisation :** paramétrer l'application.

**Scenario :** S01, S05, S06, S22, S23

**Rôle :** Utilisateur

**Résumé :**

- Se connecter l'application
- Définir les paramètres pour l'application.

**Pré condition :**

- L'utilisateur doit accéder à son espace.
- un véhicule, un chauffeur et une zone doivent être créés.

**Description :**

- 1- l'utilisateur choisi dans le menu « paramètre »
- 2- le système lui affiche la page Paramètre.
- 3- L'utilisateur rempli le formulaire d'ajout d'un paramètre et soumet.
- 4- le système retourne un message pour confirmer l'ajout du paramètre  
si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau paramètre.  
sinon, le système le retourne à son ancien formulaire.

**Post -condition :**

- une instance paramètre est crée et enregistrée.

**Tableau 3.9. :** Description de cas d'utilisation «Paramétrer l'application».

**Cas d'utilisation** : affecter paramètre/véhicule

**Scenario** : S01, S05, S06, S24

**Rôle** : Utilisateur

**Résumé** :

- Se connecter à l'application.
- Affecter des paramètres à un véhicule.

**Pré condition** :

- L'utilisateur doit accéder à son espace.
- L'application doit être paramétrée, un véhicule doit être créé.

**Description** :

- 1- l'utilisateur choisi dans le menu « véhicule/paramètre »
- 2- le système lui affiche la page Véhicule/Paramètre.
- 3- l'utilisateur remplit le formulaire d'affectation de paramètres à un véhicule et soumet.
- 4- le système retourne un message pour confirmer l'affectation si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau paramètre.  
sinon, le système le retourne à son ancien formulaire.

**Post -condition** :

- une instance paramètre/véhicule est créée et enregistrée.

**Tableau3.10** : Description de cas d'utilisation « Affecter Paramètre/Véhicule »

**Cas d'utilisation** : affecter zone/véhicule

**Scenario** : S01, S05, S24

**Rôle** : Utilisateur

**Résumé** :

- Se connecter à l'application.
- Affecter une zone à un véhicule.

**Pré condition** :

- L'utilisateur doit accéder à son espace.
- Une zone, un véhicule doivent être créés

**Description** :

- 1- l'utilisateur choisi dans le menu « véhicule/zone »
- 2- le système lui affiche la page Véhicule/zone
- 3- utilisateur remplit le formulaire d'affectation de la zone à un véhicule.
- 4- le système retourne un message pour confirmer l'affectation

si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau paramètre.  
sinon, le système le retourne à son ancien formulaire.

**Post -condition :**

une instance zone/véhicule est créée et enregistrée.

**Tableau3.11 :** Description de cas d'utilisation « Affecter Zone/Véhicule ».

**Cas d'utilisation :** affecter chauffeur/véhicule

**Scenario :** S01, S05, S25

**Rôle :** Utilisateur

**Résumé :**

- Se connecter à l'application.
- Affecter un chauffeur à un véhicule.

**Pré condition :**

- L'utilisateur doit accéder à son espace.
- Un chauffeur, un véhicule doivent être créés

**Description :**

- 1- l'utilisateur choisi dans le menu « véhicule/chauffeur »
- 2- le système lui affiche la page Véhicule/Chauffeur
- 3- utilisateur rempli le formulaire d'affectation d'un chauffeur à un véhicule.
- 4- le système retourne un message pour confirmer l'affectation  
si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant le nouveau paramètre.  
sinon, le système le retourne à son ancien formulaire.

**Post -condition :**

une instance Chauffeur /véhicule est créée et enregistrée.

**Tableau 3.12:** Description de cas d'utilisation « Affecter Chauffeur/Véhicule ».

**Cas d'utilisation :** Consulter Historique

**Scenario :** S01, S05, S09, S10

**Rôle :** Utilisateur

**Résumé :**

- Se connecter à l'application.
- Consulter Historique.

**Pré condition :**

- L'utilisateur doit accéder à son espace.

**Description :**

- 5- l'utilisateur choisi dans le menu « historique GPS »
- 6- le système lui affiche la page Historique.
- 7- utilisateur saisie le matricule, heure début, heure fin
- 8- le système retourne une ligne de la table historique correspondant à l'entée précédente.

**Post -condition :**

une instance historique est créée.

**Tableau 3.13. :** Description de cas d'utilisation «consulter Historique ».

**Cas d'utilisation :** Ajouter marque véhicule

**Scenario :** S01, S05, S06, S27

**Rôle :** administrateur

**Résumé :**

- Se connecter l'application
- Ajouter marque véhicule.

**Pré condition :**

- L'utilisateur doit accéder à son espace.

**Description :**

- 1- l'utilisateur choisi dans le menu « marque »
- 2- le système lui affiche la page Marque Véhicule
- 3- L'utilisateur rempli le formulaire d'ajout d'une marque et soumet.
- 4- le système retourne un message pour confirmer l'ajout de la zone.  
si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant la nouvelle zone.  
sinon, le système le retourne à son ancien formulaire.

**post -condition :**

- Une instance marque véhicule est créée et enregistrée.

**Tableau3.14 :** Description de cas d'utilisation « Ajouter marque Véhicule ».

**Cas d'utilisation :** Ajouter type-marque véhicule

**Scenario :** S01, S05, S06, S28

**Rôle :** administrateur

**Résumé :**

- Se connecter l'application

- Ajouter type-marque véhicule.

**Pré condition :**

- L'utilisateur doit accéder à son espace.

**Description :**

- 1- l'utilisateur choisi dans le menu « type-marque »
- 2- le système lui affiche la page Type-Marque Véhicule
- 3- L'utilisateur remplit le formulaire d'ajout de type-marque et soumet.
- 4- le système retourne un message pour confirmer l'ajout de la zone.  
si l'utilisateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant la nouvelle zone.  
sinon, le système le retourne à son ancien formulaire.

**post -condition :**

Une instance type- marque véhicule est créée et enregistrée

. **Tableau 3.15:** Description de cas d'utilisation « Ajouter type-marque véhicule ».

**Cas d'utilisation :** valider inscription

**Scenario :** S01, S05, S06, S29, S30

**Rôle :** administrateur

**Résumé :**

- Se connecter l'application.
- Valider inscription.

**Pré condition :**

- L'administrateur doit accéder à son espace.
- Le client doit avoir fait une inscription.

**Description :**

- 1- l'administrateur choisi dans le menu « utilisateur »
- 2- le système lui affiche la page Utilisateur.
- 3- L'administrateur choisi l'inscription à valider et approuve.
- 4- le système retourne un message pour confirmer la validation de l'inscription.  
si l'administrateur confirme l'ajout alors :  
le système met à jour la BDD en ajoutant la nouvelle inscription.  
sinon, le système le retourne vers le menu précédant.

**post -condition :**

Une instance inscription est créée et enregistrée

**Tableau 3.16 :** Description de cas d'utilisation « Ajouter type-marque véhicule ».

**3.5.2. Au niveau de l'application mobile :**

**Cas d'utilisation :** envoi de données.

**Scenario :**

**Rôle :** capteur de position.

**Résumé :**

- Envoi et sauvegarde des données dans la base de données interne au système ou transmission vers une plateforme distante.

**Pré condition :**

- Application lancé.
- Position calculé.

**Description :**

1- Le système vérifie la connectivité :

\*Si la connectivité est inexistante

les données seront sauvegardées dans la base de données interne

- si la connexion s'établie :

les données seront transmises de la base de données interne vers l'application distante.

\*sinon, les données seront transmises vers le système communicant jusqu'à arrêt de l'application.

**post -condition :**

Une instance données crée et enregistrée

**Tableau 3.17 :** Description de cas d'utilisation «Envoi des données ».

#### 4. Conception :

La conception est la phase la plus complexe du projet, surtout avec la fréquence des innovations technologiques. Dans cette phase une nouvelle vue du modèle fait son apparition. Cette vue exprime les modules et les exécutables physiques sans aller à la réalisation concrète du système.

##### 4.1. Les diagrammes de séquences :

Un diagramme de séquence représente des échanges de messages entre objets. Il permet de mieux visualiser la séquence des messages par une lecture de bas en haut. L'axe vertical représente le temps, et l'axe horizontal représente les objets qui collaborent. Une ligne verticale en pointillé est attachée à chaque objet et représente sa ligne de vie.

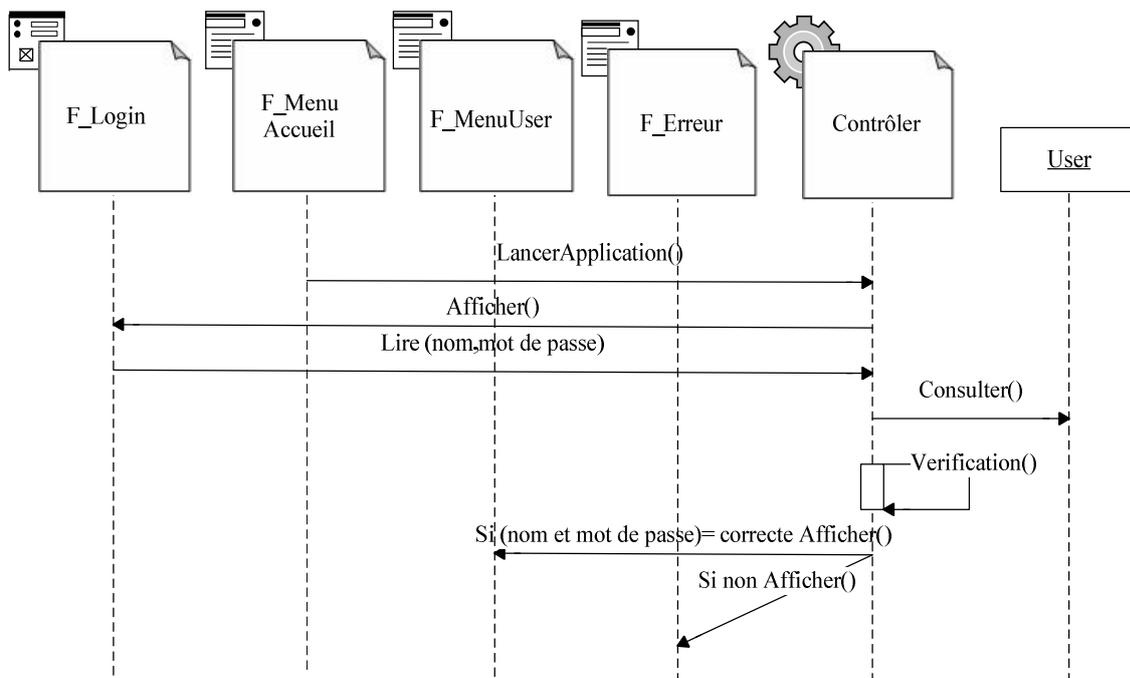


Figure3.7.: Diagramme de séquences de cas d'utilisation « authentification ».

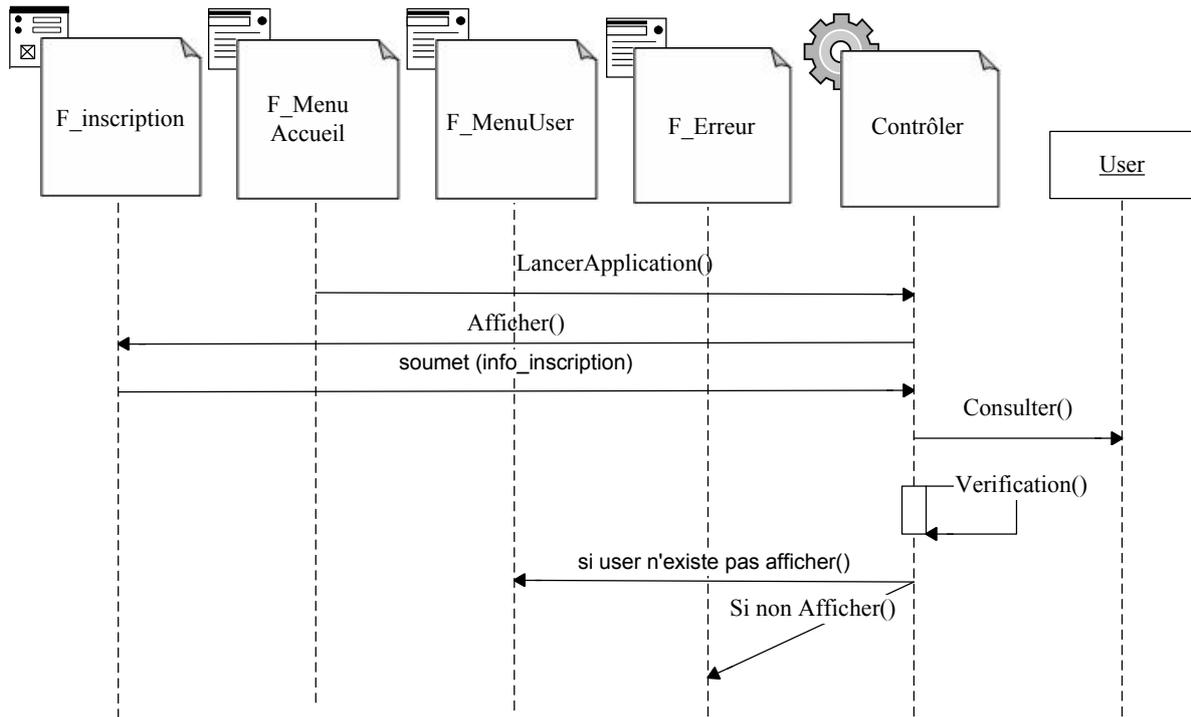


Figure3.8. : Diagramme de séquences de cas d'utilisation «inscription »

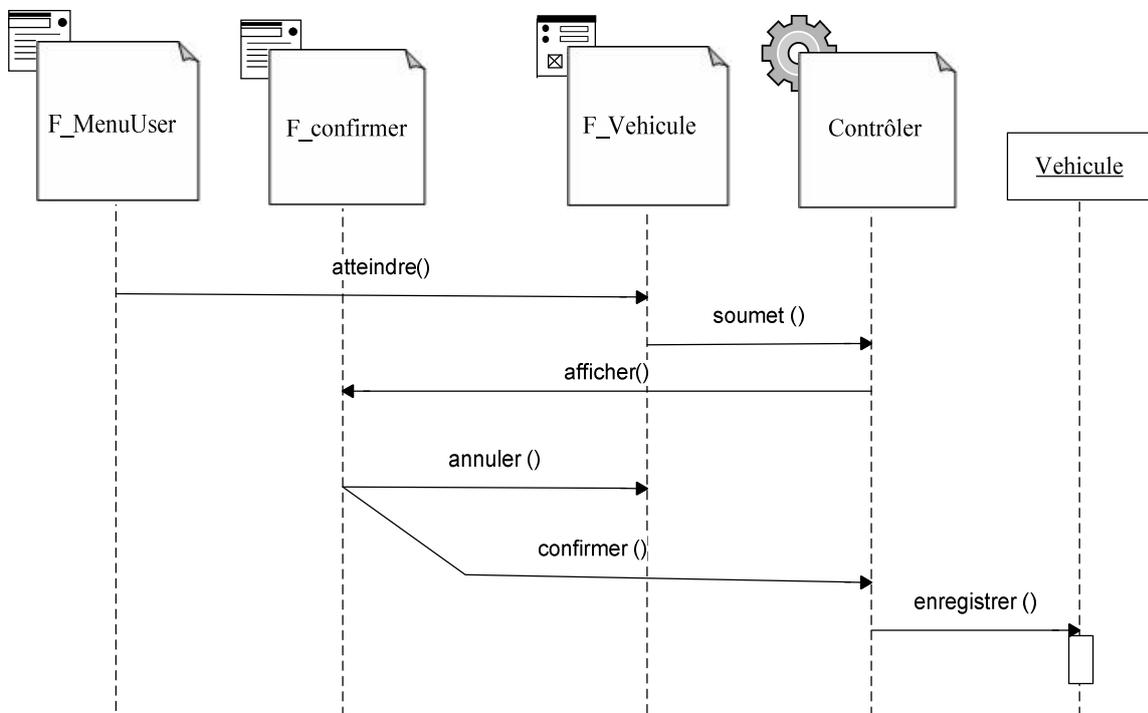
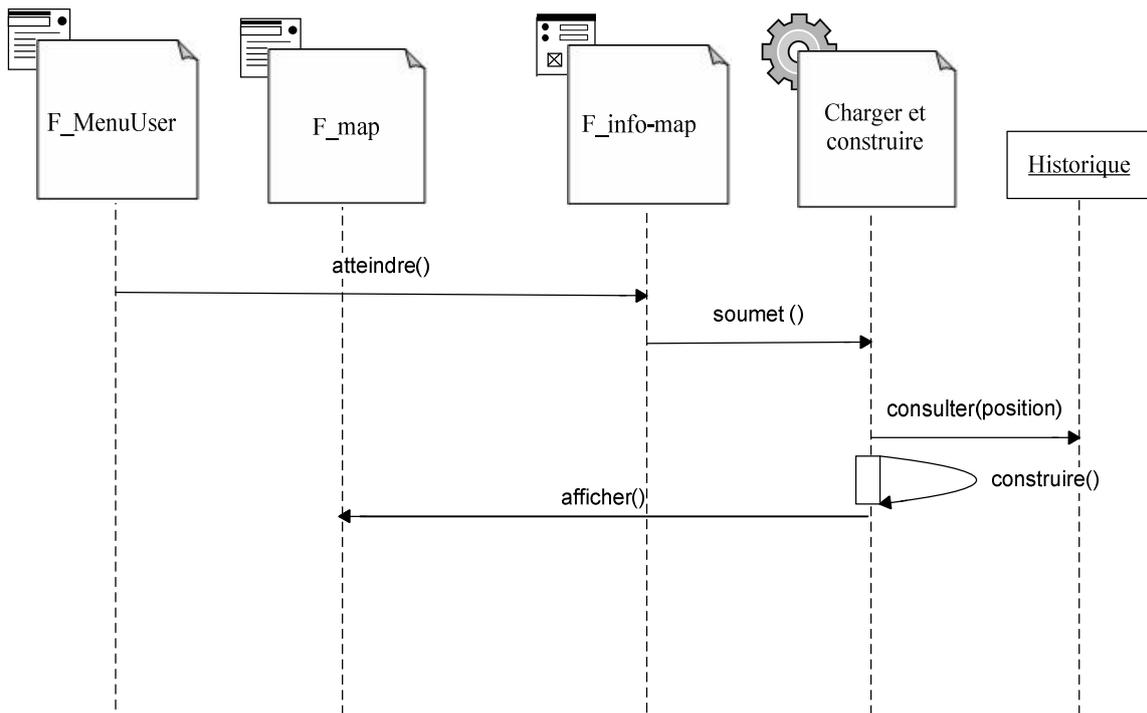
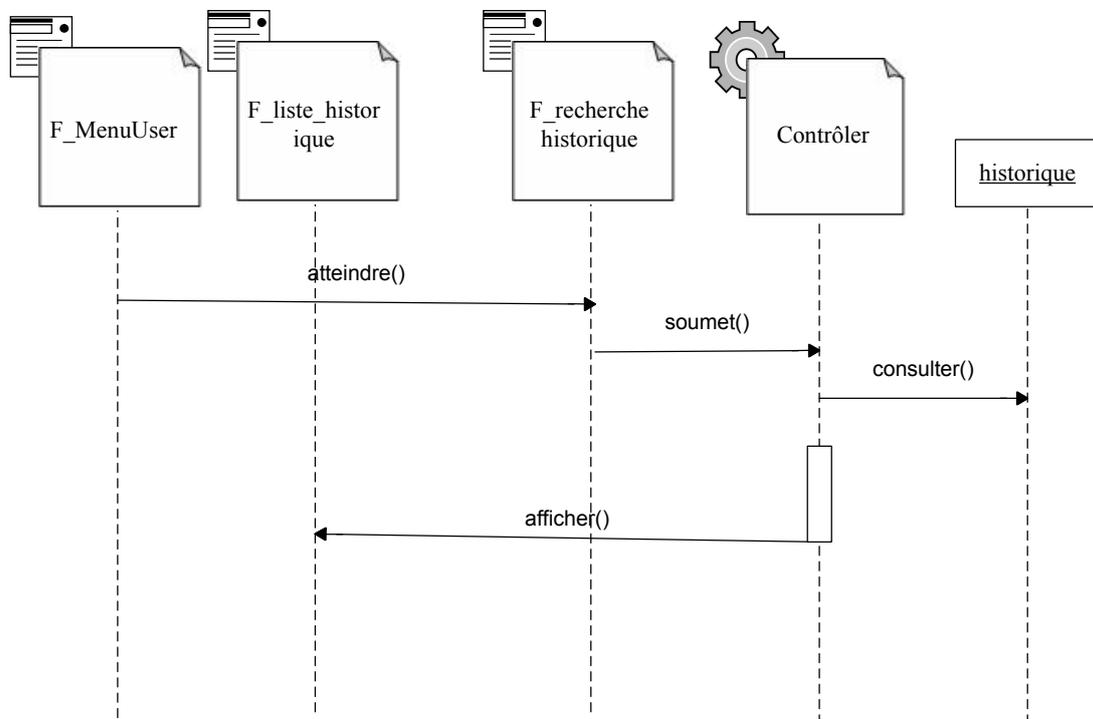


Figure3.9 : Diagramme de séquences de cas d'utilisation « Ajouter véhicule ».



**Figure3.10:** Diagramme de séquences de cas d'utilisation « Localiser un véhicule sur la Map en temps réel ».



**Figure 3.11:** Diagramme de séquences de cas d'utilisation « Consulter Historique ».

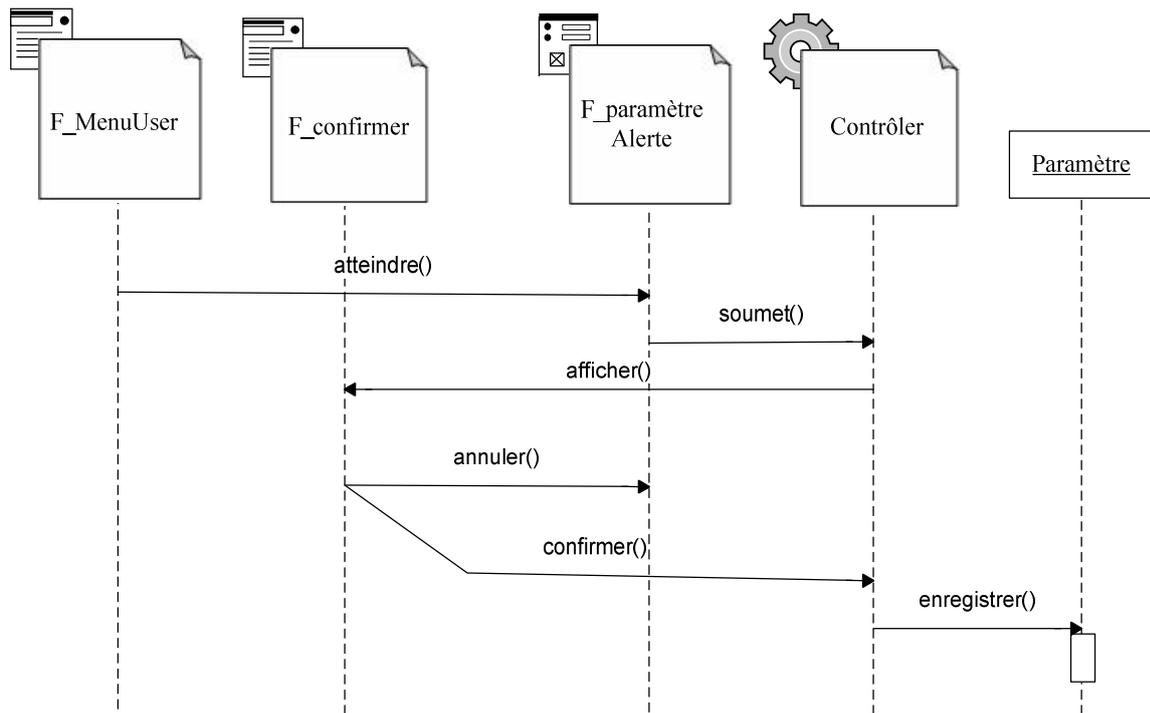


Figure3.12 : Diagramme de séquences de cas d'utilisation « Paramétriser l'application ».

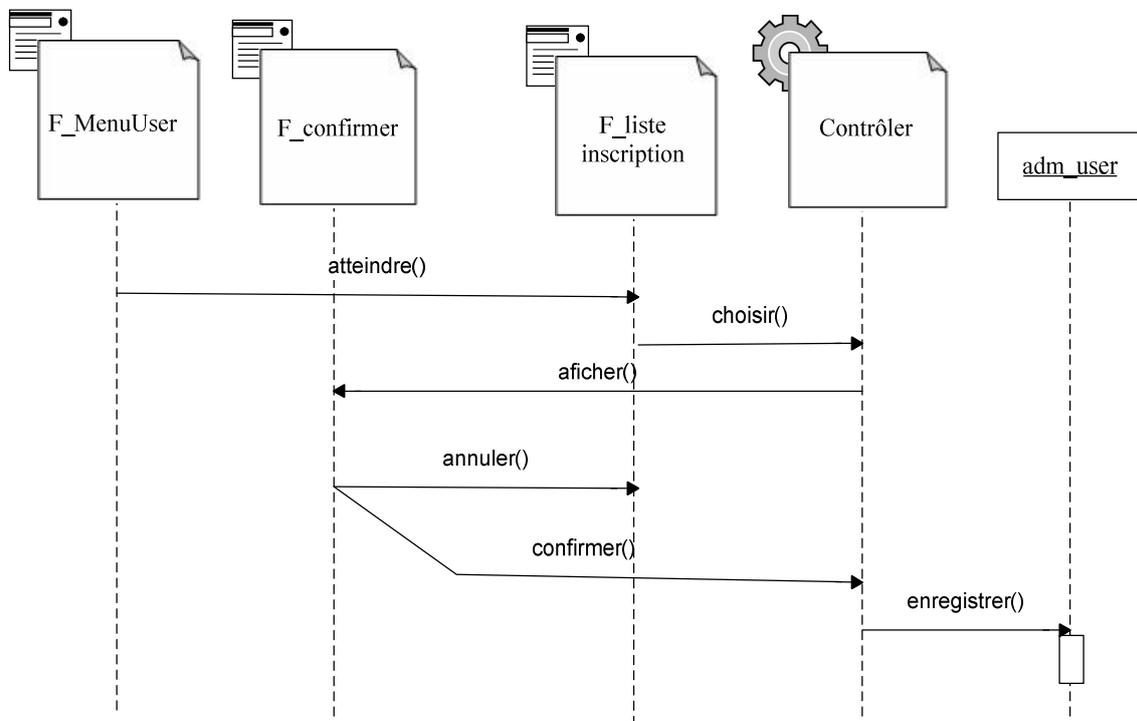


Figure3.13 : Diagramme de séquences de cas d'utilisation « Valider inscription ».

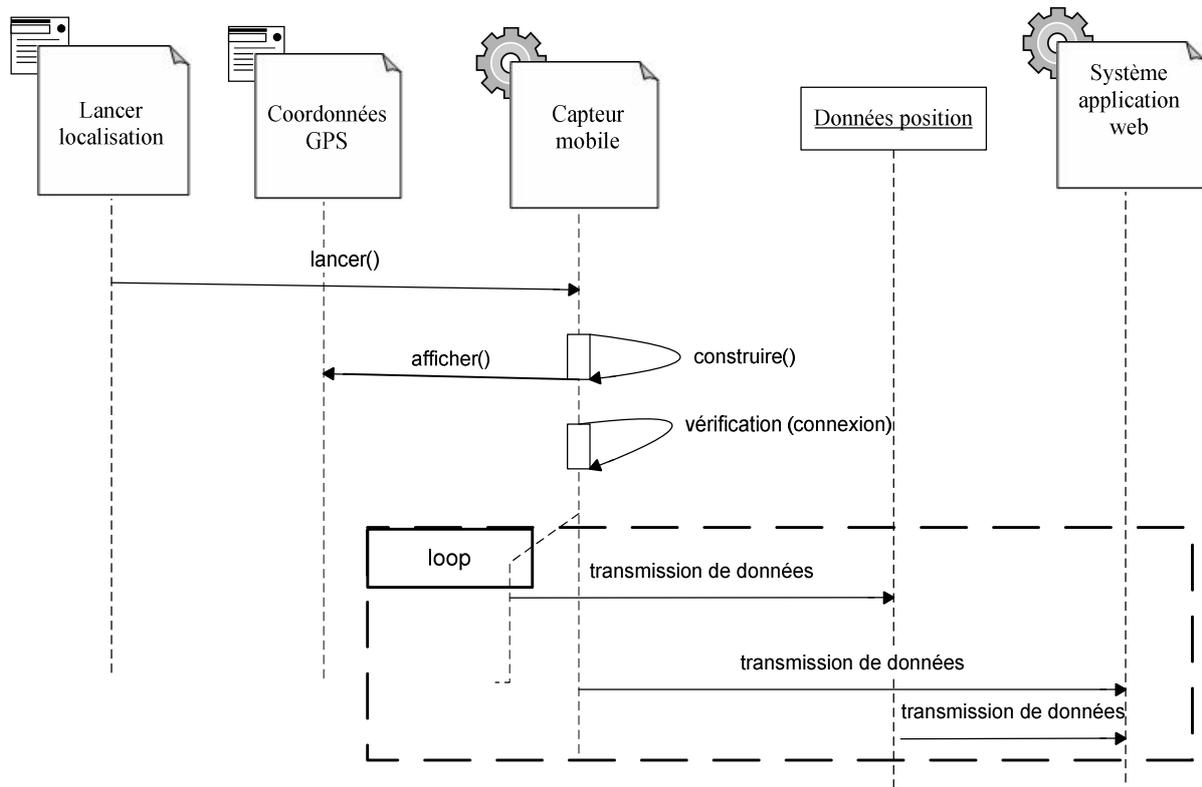
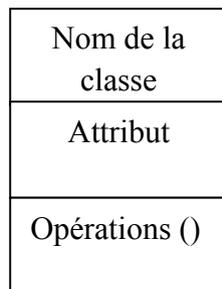


Figure3.15 : Diagramme de séquences de cas d'utilisation « envoi des données ».

## 4.2. Diagrammes de classes :

La classe est un ensemble d'objet ayant les mêmes caractéristiques

- **Représentation graphique des classes :**



Un diagramme de classe représente la structure statique d'un système, il contient des classes et leurs associations et éventuellement des objets. L'intérêt majeur des diagrammes de classe est de modéliser les entités d'un système.

L'ensemble des classes est partitionné en quatre composantes :

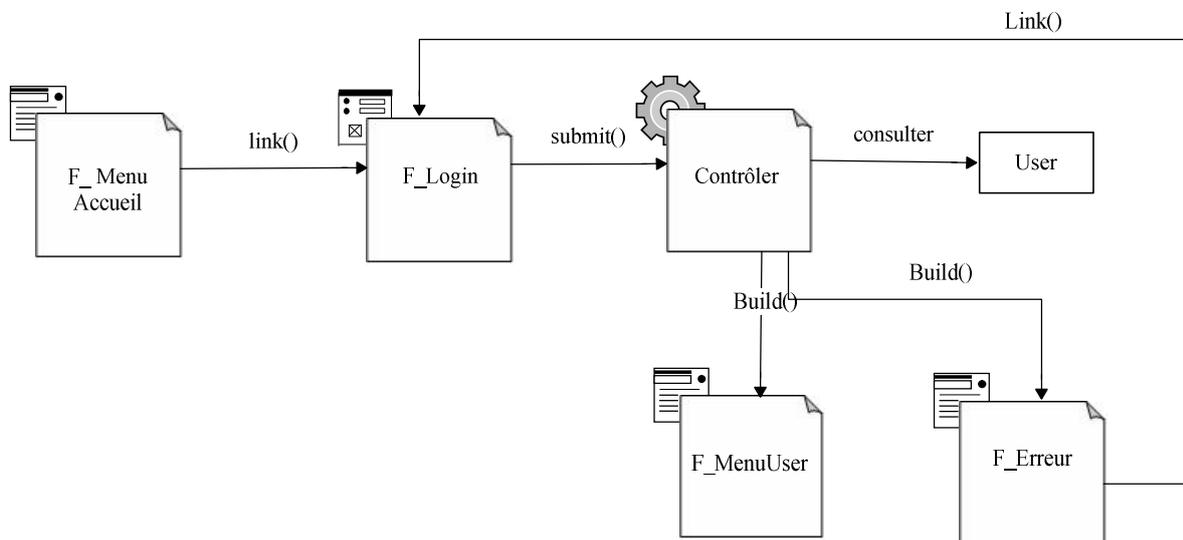
- **Classes entités** : elles permettent de modéliser toutes les informations que l'on veut gérer.
- **Les classe acteurs** : elles représentent les rôles attribués.
- **Les classes processus** : elles permettent de répertorier les activités organisées pour accomplir les missions.
- **Autres processus** : cette catégorie regroupe toutes les classes qui ne sont ni acteur ni processus.

### **Stéréotype association :**

- « **Link** » : est une association entre page entre une page client et une autre page client ou serveur. Elle représente un pointeur entre ces pages. A un lien correspond une balise ancre HTML.
- Soumet « **Submit** » : est une association qui se trouve toujours entre un formulaire et une page serveur. Les valeurs des champs du formulaire sont soumises au serveur qui les traite par l'intermédiaire de pages serveur.

- « **Build** » : est une association orientée entre les pages client et les pages serveur. Elle indique quelle page serveur est responsable de la création de la page client. Une page serveur peut construire plusieurs pages client, mais une page client n'est construite que par une et une seule page serveur.
- « **Redirect** » : est une association unidirectionnelle qui relie deux pages client ou serveur.

Après les diagrammes de séquence des différents cas d'utilisations représentés, nous allons passer aux diagrammes de classes généraux qui représentent une vue logique des objets.



**Figure3.16** : diagramme de classes général du cas d'utilisation « authentification »

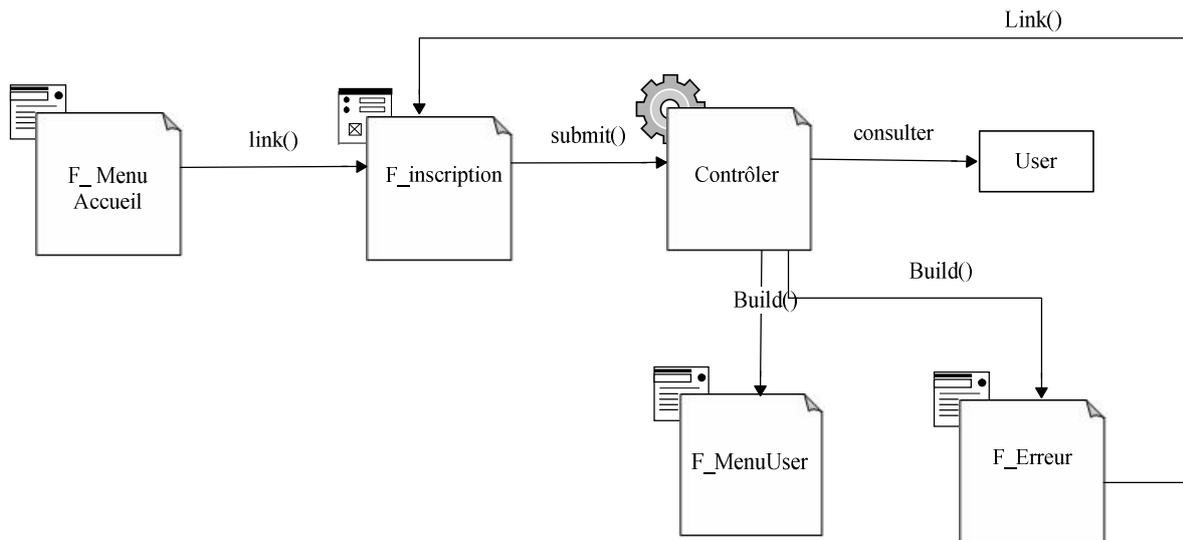


Figure3.17 : diagramme de classes général du cas d'utilisation « inscription »

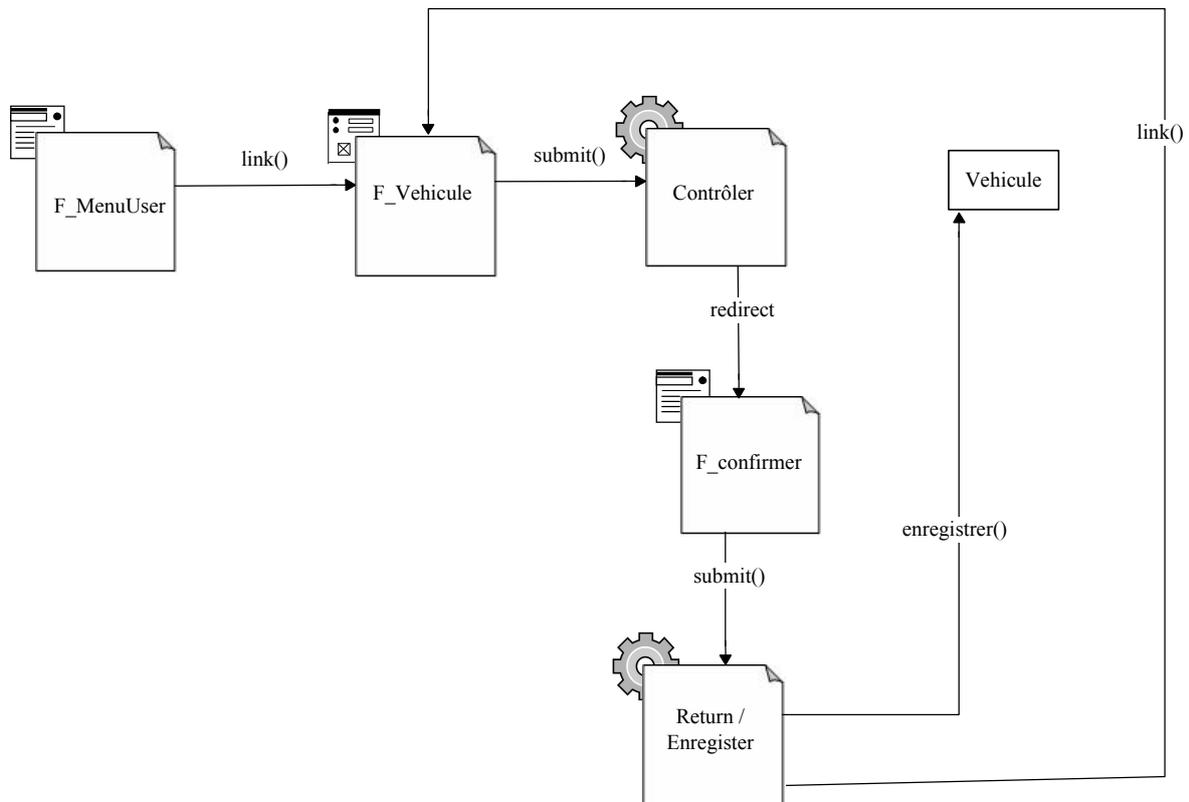


Figure3.18 : diagramme de classes général du cas d'utilisation «Ajouter véhicule».

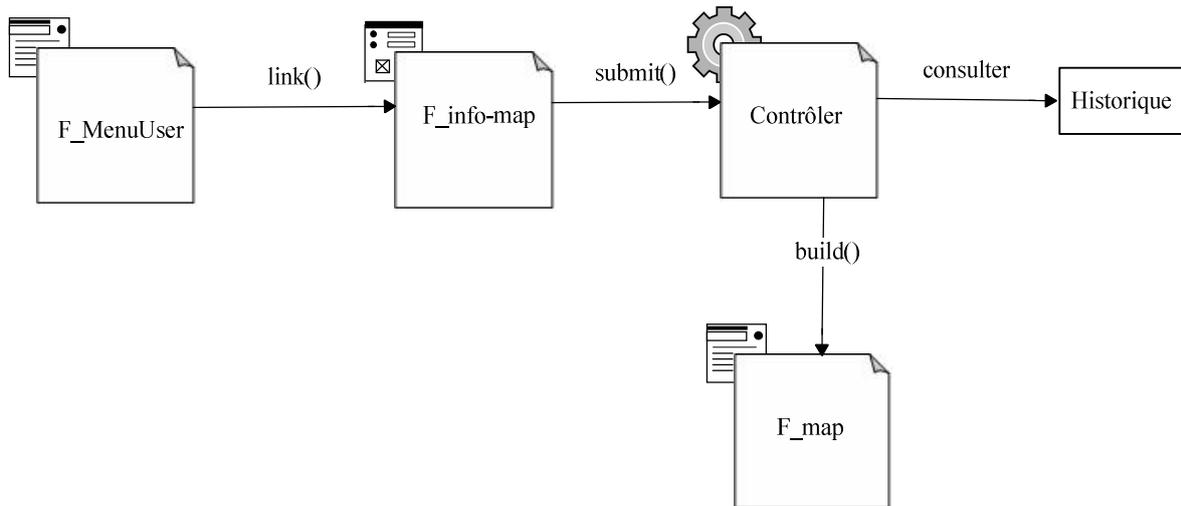


Figure3.19 : Diagramme de classes général du cas d'utilisation « Localiser un véhicule sur la Map »

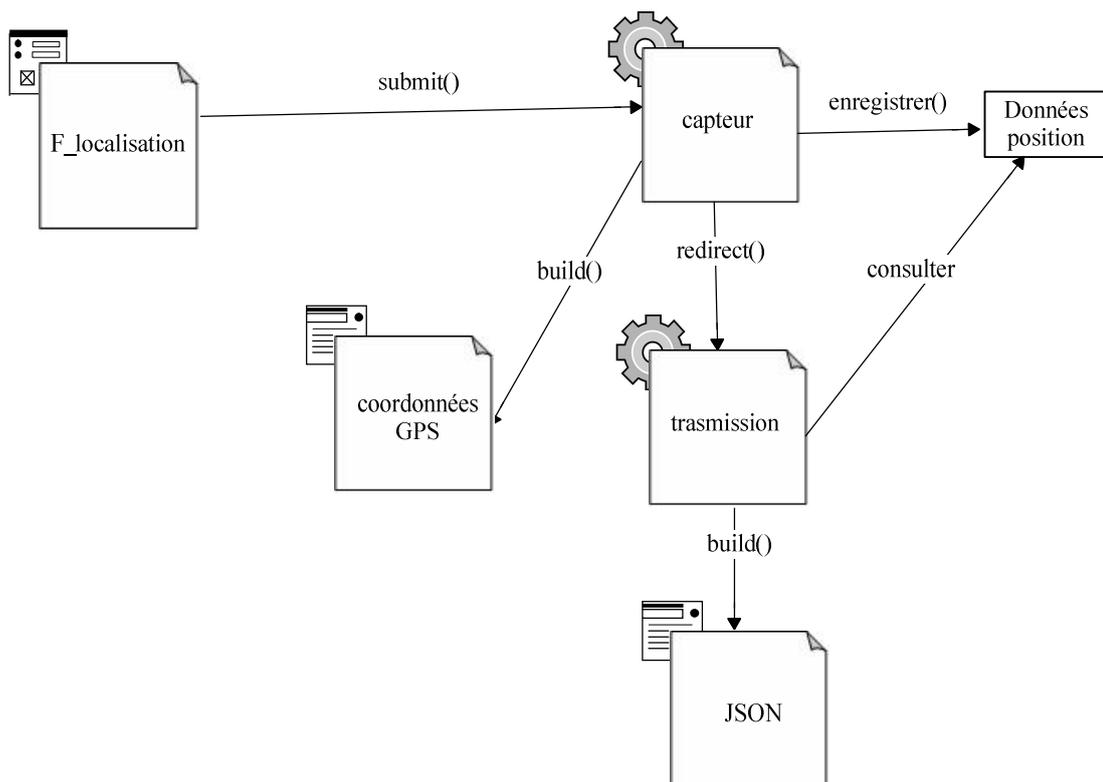
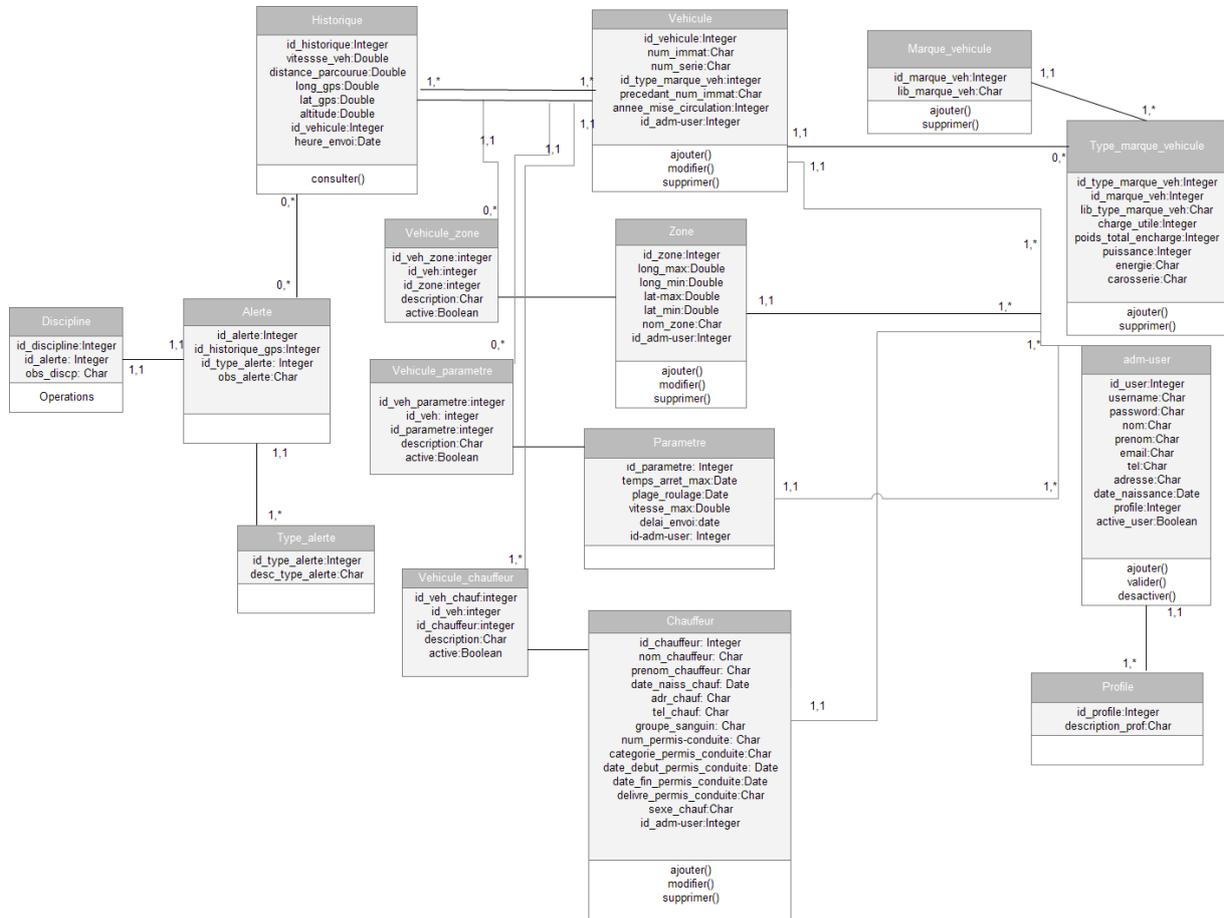


Figure3.20 : Diagramme de classe de cas d'utilisation « envoi de données ».

5. Le modèle conceptuel des données :



Les règles de gestion :

- Un type véhicule correspond à plusieurs véhicules.
- Un véhicule a un seul type.
- Une marque véhicule possède 1 ou plusieurs type véhicule.
- Un type appartient à une seule marque.
- Un véhicule est conduit par un chauffeur ou plusieurs qui peuvent être activé.
- Un chauffeur conduit un seul véhicule.
- Un véhicule possède un seul paramètre qui peut être activé.
- Un paramètre est affecté à 1 ou plusieurs véhicules.
- Un véhicule possède 1 ou plusieurs lignes historiques.
- Une ligne historique appartient à un seul véhicule.

- Un véhicule peut rouler dans 1 ou plusieurs zones qui peuvent être activé.
- Une zone est affectée à 1 ou plusieurs véhicules.
- Une ligne historique peut déclenchée une alerte zone, alerte paramètre.
- Une discipline est infligée à un chauffeur suite à une infraction qui génère une alerte à un moment donné.
- Un utilisateur possède un seul profile.
- Un profile possède plusieurs utilisateurs.

## 6. Modèle logique de données :

### Abréviations utilisées sur les contraintes:

- **PK:** *Primary Key* (clef primaire)
- **FK:** *foreign key* (clef étrangère)

#### 1. Zone de circulation du véhicule (Nom de Table : ZONE)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
id_ZONE	NOT NULL	integer	<b>PK</b>	
Lat_min		Double		
Lat_max		Double		
Long_min		Double		
Long_max		Double		
Id_adm-user		Integer	<b>FK</b>	adm-user
Nom_zone	NOT NULL	Text		

#### 2. Alerte du véhicule (Nom de Table : ALERTE )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
Id_alerte	NOT NULL	integer	<b>PK</b>	
Id_hist_gps		Integer	FK	Historique
Id_type_alerte		Integer	FK	Type_alerte
Obs_alerte		Text		

#### 3. Discipline de l'employé mobile (Nom de Table : Discipline)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_discipline</b>	NOT NULL	integer	<b>PK</b>	
Id_alerte		Integer	FK	Alerte
Id_adm-user		Integer	FK	adm-user
Obs_alerte		Text		

#### 4. Paramètres de l'application (Nom de Table : Paramètre )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_parametre</b>	NOT NULL	integer	<b>PK</b>	
Temps arrêt_max		Time		
Plage_roulage		Time		
Vitesse_max		Double		
Délai_envoi		Time		
Id_adm-user		Integer	FK	adm-user

#### 5. Historique du véhicule (Nom de Table : Historique)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_Historique</b>	NOT NULL	integer	<b>PK</b>	
Vitesse_veh		Double		
Distance_parcouru		Double		

Long_gps	NOT NULL	Double		
Lat_gps	NOT NULL	Double		
Altitude_gps	NOT NULL	Double		
Id_vehicule	NOT NULL	Integer	FK	Vehicule
Heure_envoi	NOT NULL	Time		

## 6. Véhicule (Nom de Table : Véhicule )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_vehicule</b>	NOT NULL	integer	<b>PK</b>	
Num_immatriculation	NOT NULL	VARCHAR(15)		
Num_serie		VARCHAR(25)		
Id_marque_veh	NOT NULL	Integer	FK	Marque_vehicule
Id_type_marque_veh	NOT NULL	Integer	FK	Type_marque_veh
Precedent_num_immat		VARCHAR(15)		
Annee_mise_circulat		Integer		
Id_adm-user	NOT NULL	integer	FK	Adm-user

## 7. Chauffeur du véhicule (Nom de Table : Chauffeur )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_chauffeur</b>	NOT NULL	integer	<b>PK</b>	
Nom_chauffeur		VARCHAR(25)		
Prenom_chauffeur		VARCHAR(25)		

Date_naissance_chauf		Date		
Adresse_chauffeur		VARCHAR(50)		
Tel_chauffeur		VARCHAR(20)		
Email_chauffeur		VARCHAR(50)		
Groupe_sanguin		VARCHAR(5)		
Num_permis_conduite		VARCHAR(20)		
Categorie_permis_conduite		VARCHAR(20)		
Date_debut_permis_conduite		Date		
Date_fin_permis_conduite		Date		
Delivre_permis_conduite		VARCHAR(25)		
Sexe_chauffeur		VARCHAR(5)		
Id_adm-user	NOT NULL	integer	FK	adm-user

8. Utilisateur de la plateforme (Nom de Table : Adm\_user )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_user</b>	NOT NULL	integer	<b>PK</b>	
Username	NOT NULL	VARCHAR(50)		
Password	NOT NULL	VARCHAR(50)		
Nom	NOT NULL	VARCHAR(50)		
Prenom	NOT NULL	VARCHAR(50)		
e-mail	NOT NULL	VARCHAR(100)		
Tel	NOT NULL	VARCHAR(20)		

Profile	NOT NULL	integer	FK	Profile
Active_user		Boolean		
Date_naissance		Date		
Adresse		VARCHAR(100)		

9. Profile des utilisateur (**Nom de Table** : Adm\_profile )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_profile</b>	NOT NULL	integer	<b>PK</b>	
Description_profile	NOT NULL	VARCHAR(20)		

10. Affectation véhicule /zone (**Nom de Table** : véhicule/zone )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_veh_zone</b>	NOT NULL	Serial	<b>PK</b>	
Id_vehicule	NOT NULL	Integer	FK	Vehicule
Id_zone	NOT NULL	Integer	FK	Zone
Description		VARCHAR(15)		
Active		Boolean		

11. Affectation véhicule /paramètre (**Nom de Table** : véhicule/paramètre)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_veh_parametre</b>	NOT NULL	Serial	<b>PK</b>	
Id_vehicule	NOT NULL	Integer	FK	Vehicule

Id_parametre	NOT NULL	Integer	FK	Parametre
Description		VARCHAR(15)		
Active		Boolean		

## 12. Affectation véhicule /chauffeur (Nom de Table : véhicule/chauffeur )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_veh_chauffeur</b>	NOT NULL	Serial	<b>PK</b>	
Id_vehicule	NOT NULL	Integer	FK	Vehicule
Id_chauffeur	NOT NULL	Integer	FK	Chauffeur
Description		VARCHAR(15)		
Active		Boolean		

## 13. Type\_marque\_vehicule (Nom de Table : type\_marque )

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
<b>Id_type_marque_veh</b>	NOT NULL	Integer	<b>PK</b>	
Id_marque_veh	NOT NULL	Integer	FK	Marque_vehicule
Libelle_type_marque_veh	NOT NULL	VARCHAR(25)		
Charge_utile		Integer		
Poids_total_encharge		Integer		
Puissance		Integer		
Energie		VARCHAR(15)		
Carrosserie		VARCHAR(25)		

14. Marque\_vehicule (Nom de Table : marque\_vehicule)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
Id_marque_veh	NOT NULL	integer	PK	
Libelle_marque_veh	NOT NULL	VARCHAR(25)		

15. Type\_alerte (Nom de Table : Type\_alerte)

Nom Attributs	NULL ?	Type Attributs	Contraintes	Table Référencée
Id_type_alerte	NOT NULL	integer	PK	
Description_type_alerte	NOT NULL	VARCHAR(50)		

**7. Conclusion :**

Dans ce chapitre nous avons présenté une démarche pour développer notre application, En se basant sur la méthode UML, en commençant par l'analyse et la spécification des besoins et les divers cas d'utilisations, et en phase de conception nous avons élaboré les diagrammes de séquences et de classe, et le modèle conceptuelle des données (MCD), et enfin nous avons défini les tables de notre base de données.

CHAPITRE

---

IV

*Implémentation  
&  
Mise En Œuvre*

## I. Introduction :

L'implémentation est le résultat de la conception, son importance est crucial pour le traitement du système sous forme de composants pour chaque cas d'utilisation, et traduire les méthodes et objets de cette application en des classes et du code source.

## II. Description générale du fonctionnement du système :

L'architecture du système développé est décrite dans la figure ci-dessous :



**Figure 4.1** : architecture matérielle du système.

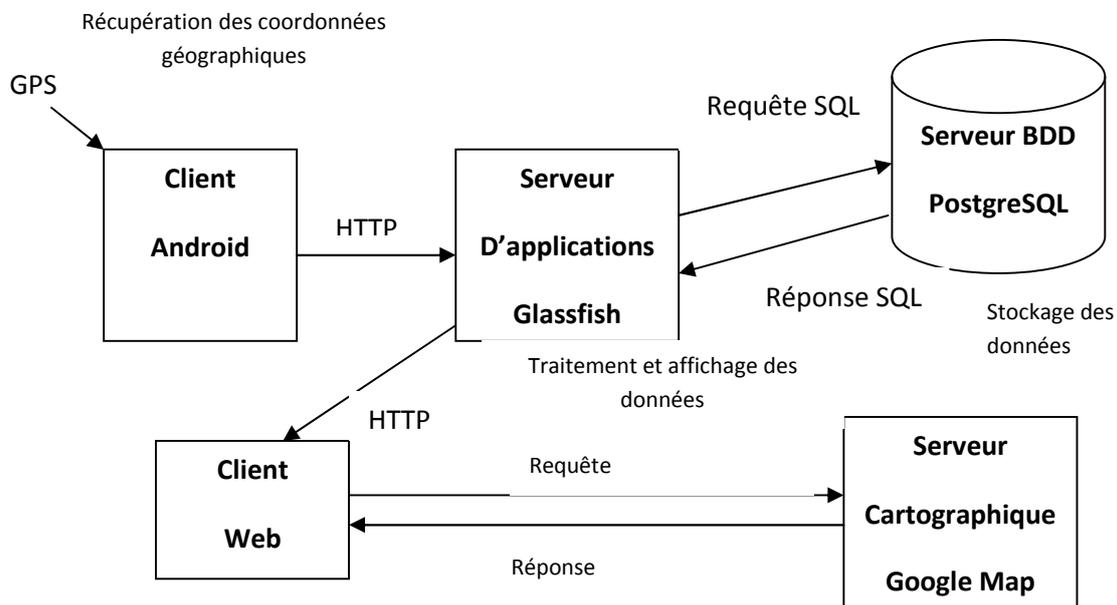
La figure ci-dessus représente l'architecture matérielle du système futur, les éléments de géolocalisation par le système GPS sont :

- l'appareil de suivi (Smartphone) qui sert à calculer les informations de navigation (coordonnées géographiques, vitesse ...) relative au véhicule.
- le système GPS utilisé par le terminal pour ces calculs.
- un réseau GSM/GPRS que le terminal utilisera pour transmettre les données aux serveurs.
- un ensemble de serveurs, il se compose d'un serveur de base de données, d'un serveur d'application et d'un serveur cartographique.
- un réseau informatique pour la diffusion des données (Internet).

L'architecture de l'application mise en œuvre est une architecture multi niveaux. On peut diviser notre système en deux sous systèmes : un système chargé de la récupération des données et un autre système chargé du stockage et des traitements de ces données.

Cela nous conduit à une implémentation double car ces deux systèmes sont hétérogènes, fonctionnant sur des plateformes différentes, à distance ; l'environnement de travail de chaque sous système fonctionne avec des technologies et outils différents, ce qui rend son implémentation assez délicate.

Pour bien illustrer chaque environnement, ainsi que les outils, langages utilisés nous avons opté à diviser cette section en deux parties ; partie web et partie mobile.



**Figure 4.2** : Architecture logicielle du système.

### III. Choix des outils et technologies utilisés :

Une étude comparative des différents outils technologiques était établie afin d'opter pour celui qui conviendra le plus aux spécifications du projet, à savoir les langages de développement, l'outil de cartographie.

#### a. Langage de programmation :

Une attention particulière a été mise sur le langage de programmation, puisque ce choix va orienter le choix des outils utilisés par la suite. En comparant essentiellement les performances, la disponibilité des bibliothèques et la portabilité du langage, en Conclusion notre choix est opté pour le langage JAVA.

#### b. Plateforme mobile :

Aujourd'hui les deux plateformes mobiles les plus utilisées sont Android de Google et IOS de Apple, bien qu'il existe d'autres alternatives pour produire des applications embarquées comme J2ME par exemple. On comparant ces deux OS en abouti a la conclusion que l'utilisation du système Android est meilleure et cela pour plusieurs raisons :

- Android est la plateforme mobile la plus utilisée.
- Le SDK et l'émulateur d'Android sont gratuits, à l'inverse de ceux de l'iPhone. En plus, le système lui-même est gratuit et open source.
- Programmation Android est basée sur XML et JAVA, deux langages déjà très populaires, à l'inverse de iOS, dont le langage de programmation est spécifique (Objective C).
- L'environnement de développement d'Android est intégré aux IDE à travers des plugins.

#### **IV. Environnement et les outils de développement :**

Il existe un panorama de technologies et de nombreuses solutions techniques pour mettre en œuvre un outil de gestion d'une flotte de véhicules, dans ce qui suit nous allons citer les technologies et outils auxquelles nous avons eu recours

##### **a. Environnement matériel :**

Pour la réalisation de notre application ; nous avons travaillé sur un environnement avec les caractéristiques suivantes :

- PC Portable : Intel® Pentium® CPU P6200 @2.13GHz, 2.00 Go de mémoire vive, Windows 7.
- Smartphone RAINBOW version 8 QUAD CORE, 1.3 GHZ, 1GB de RAM ; version Android 4.4.2. (KitKat).

##### **b. Partie mobile :**

Cette partie sera divisée en deux parties ; la première partie sera consacrée à la description de l'environnement de travail et les outils de développement de notre application mobile. La deuxième partie contiendra les détails techniques d'implémentation des différentes fonctionnalités offertes par l'application mobile.

##### **i. Environnement cible :**

Disposant d'un Smartphone RAINBOW version 8 QUAD CORE, de 1GB de RAM ; version Android 4.4.2. (KitKat). Notre application est destinée pour tous les appareils équipés d'un système Android 4.4.2, et toutefois, notre application, est utilisable sur les versions antérieures d'Android.

## ii. Environnement logiciel :

Avant d'entamer le développement, il est nécessaire de préparer l'environnement de travail. Pour notre cas, nous avons choisi de faire notre développement sur la plateforme Eclipse. Pour la préparer, nous avons effectué différentes tâches que nous décrivons dans ce qui suit :

- Installation d'éclipse (comme pour le développement d'applications java classique).
- Installation SDK (software development Kit) Android qui va contenir tous les "outils" nous permettant de "construire" des application sous Android.
- Le plugin "Android pour Eclipse" ADT (Android Development Tools), qui adapte Eclipse au développement d'applications sous Android.

Après la préparation du logiciel de travail éclipse et pour pouvoir développer une application dans de bonnes conditions, il faut bien savoir choisir son environnement de développement selon les besoins, de ce fait, nous avons développé une application pour mobile en JAVA, qui va être compilé en fichier DEX par le **Dalvik VM** (voir l'annexe) ensuite déployé sous une extension **APK** que nous avons installé sur un mobile fonctionnant sous le système **Android**, pour cela nous avons choisi **Eclipse IDE** qui nous permettra de réaliser ce travail.

### 1. Eclipse :

Eclipse est un IDE qui permet de programmer dans différents langages grâce à ses nombreux plug-ins et notamment le plug-in d'Android. Une interface spécifique permet de gérer des fichiers java et de compiler ses programmes.les fichiers sont organisés selon une arborescence qui correspond aux paquetages java définis. L'analyse syntaxique permet de mettre en valeur les mots clés dans les fichiers java. Eclipse dispose aussi d'un système d'auto complétion des fonctions, de détection des erreurs syntaxiques en temps réel sans oublier un système de débogage permettant d'exécuter ses programmes pas à pas.

### 2. Eclipse IDE :

L'IDE Eclipse est un environnement de développement permettant d'écrire, compiler, déboguer et déployer des programmes. Il est écrit en java. Par ailleurs il existe un grand nombre de modules pour étendre l'IDE Eclipse.

Pour le développement de notre application, nous avons choisi Eclipse LUNA (Son interface principale est donnée dans la figure1 ci-dessous :

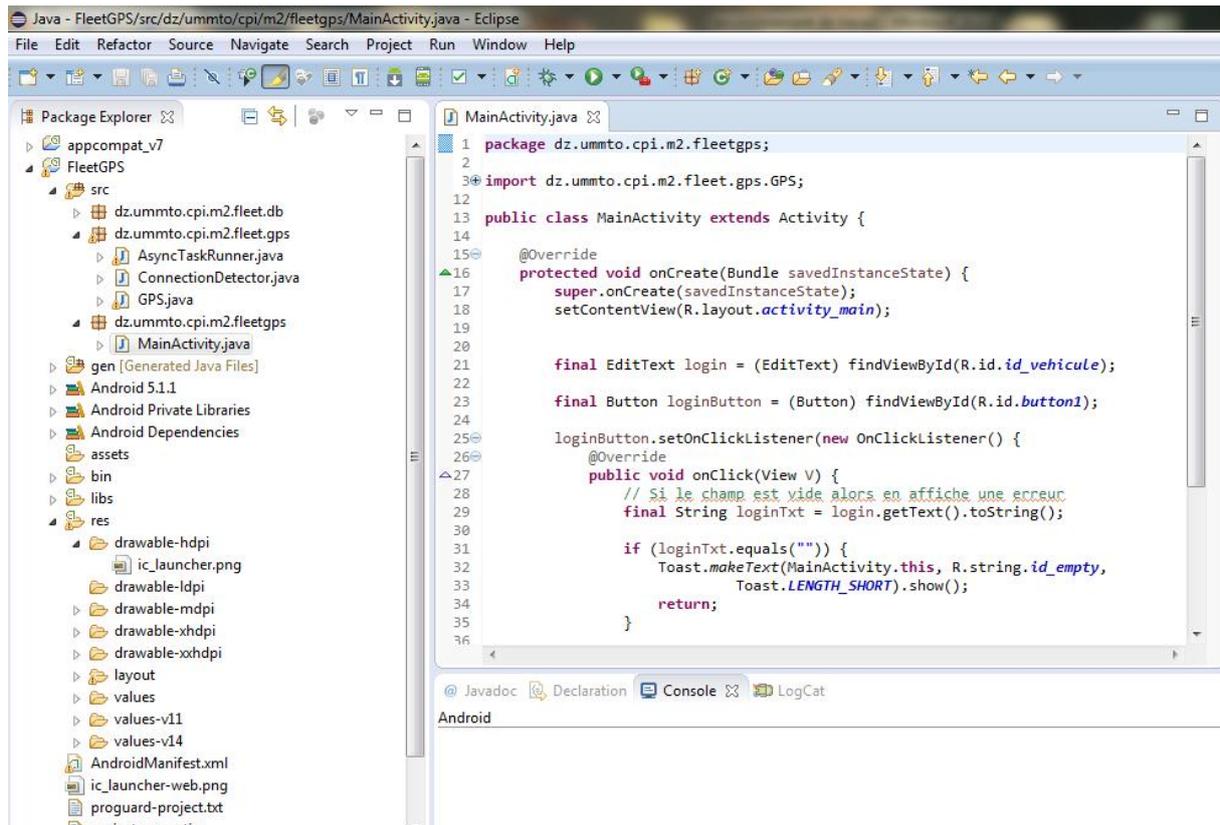


Figure4.2. : Interface principale d'éclipse.

### 3. Le plugin ADT :

Afin de développer sous Android, nous avons installé le plugin Android qui rajoutera à Eclipse les fonctionnalités spécialisées dans le développement sous Android. Le contenu du projet ADT (Android Development Tools) est donné dans l'annexe A.

### 4. Le SDK Android :

Le SDK (Software Development kit) est un ensemble d'outils de développement qui permet de créer des nouvelles applications. Son installation se fait en lançant le SDK-Manager.exe, téléchargé du site officiel d'Android, la fenêtre suivante s'ouvre :

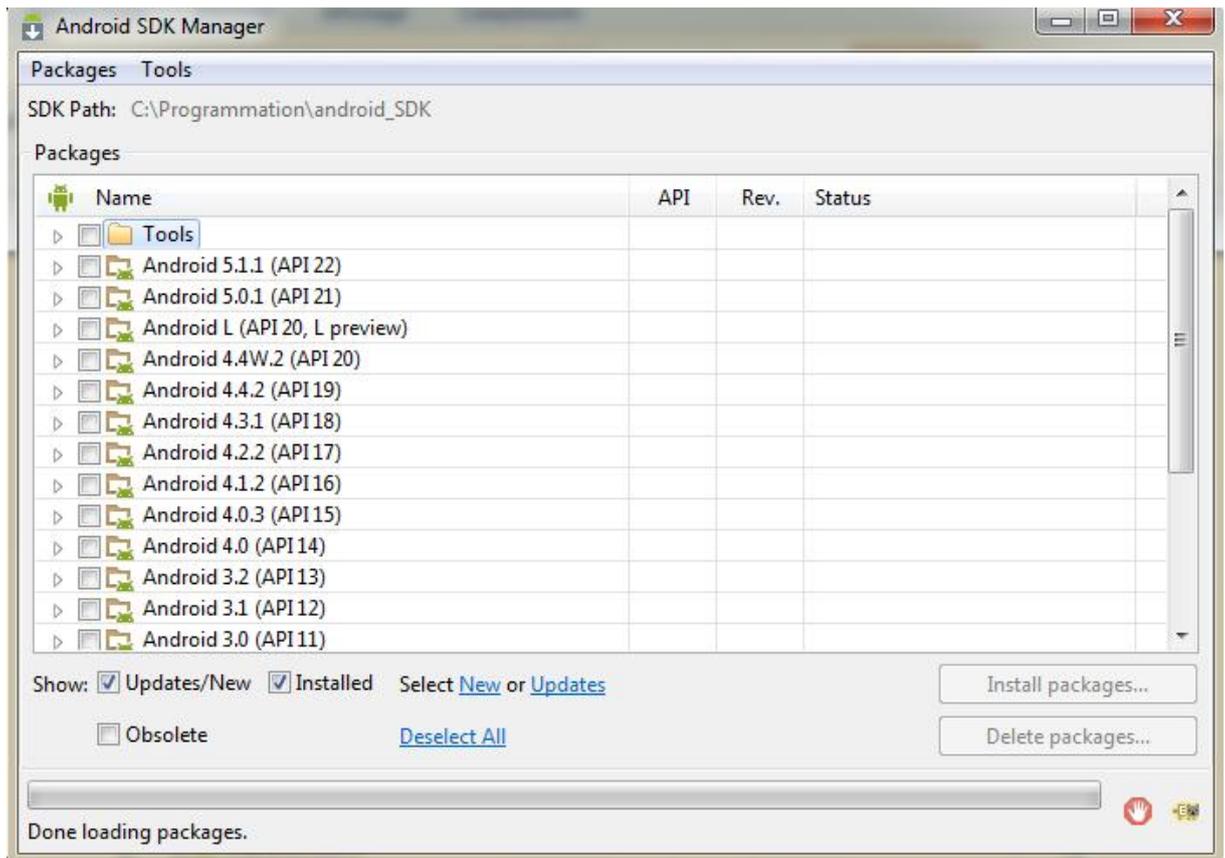


Figure4.3. : Choix de la version de SDK.

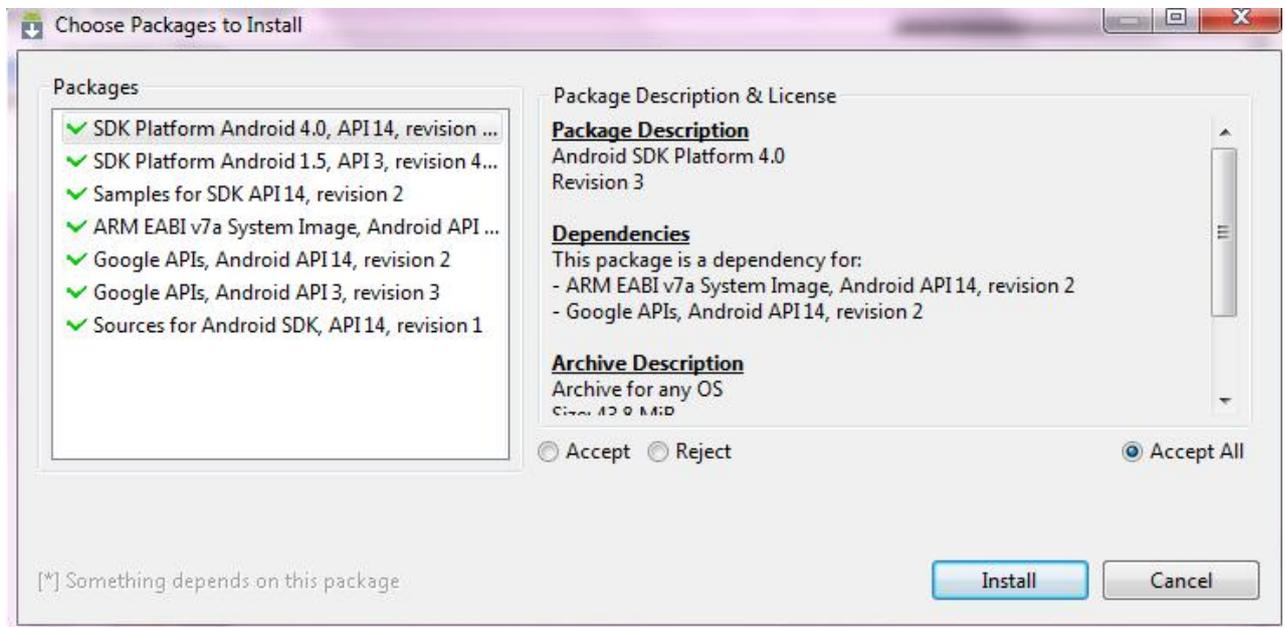


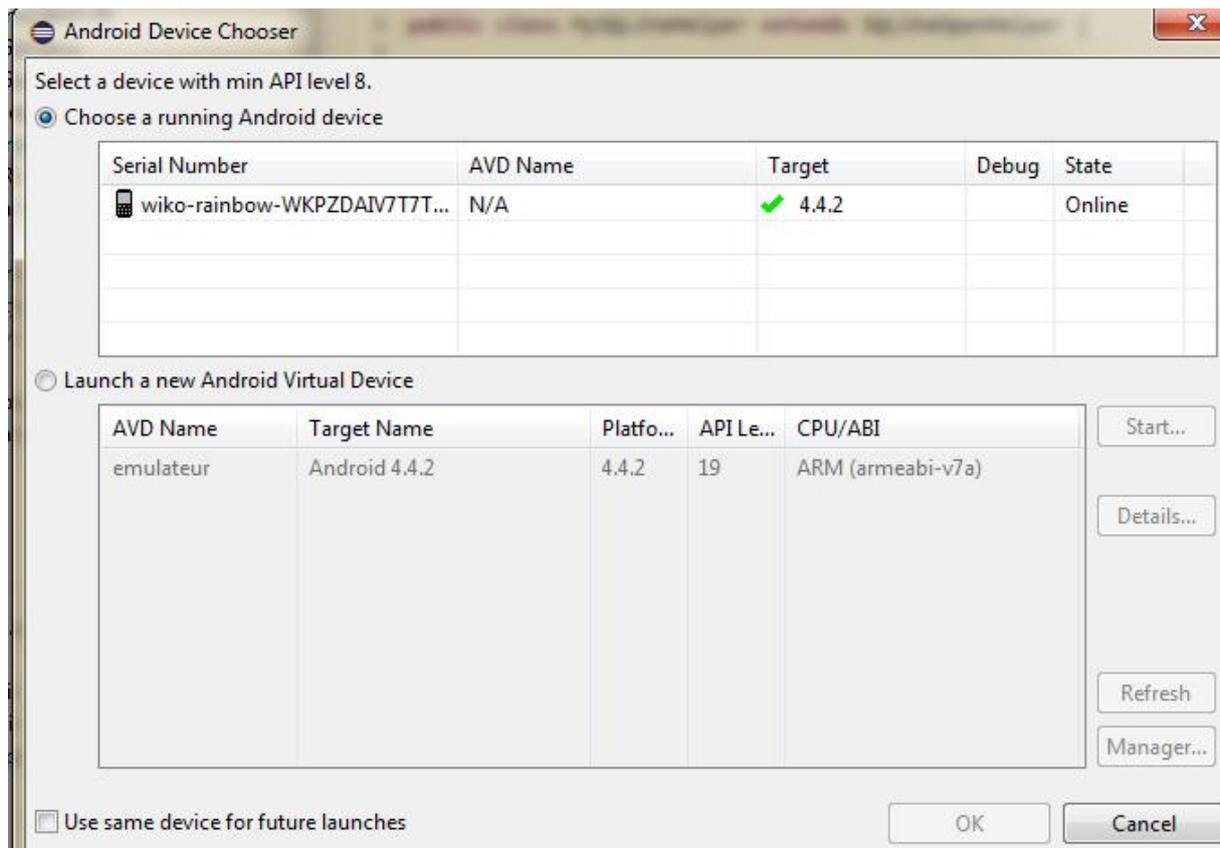
Figure4.3. : Installation des packages.

## 5. L'Emulateur Android :

Le SDK Android inclut un émulateur d'appareil mobile, c'est un dispositif mobile virtuel qui s'exécute sur l'ordinateur et permet de développer et de tester des applications Android sans l'aide d'un dispositif physique.

Lorsque l'émulateur est lancé, il nous permet d'interagir avec le périphérique mobile émulé. Nous utilisons également le pointeur de la souris pour "toucher" l'écran tactile et nous pouvons utiliser certaines touches du clavier pour invoquer certaines touches sur l'appareil.

L'émulateur Android imite toutes les fonctionnalités matérielles et logicielles d'un dispositif mobile typique, tel que la lecture des fichiers audio et vidéo, stockage des données. Sauf qu'on ne peut accéder à la carte réseau chose qui est indispensable pour le fonctionnement de notre application pour cela nous n'avons pas utilisé cet outil on est amené à exécuter l'application directement sur le Smartphone physique en le reliant à la machine avec un câble USB et faire le choix lors du lancement de l'exécution comme le montre la figure suivante :



**Figure4.4.** : Effectuer le choix lors de lu lancement de l'exécution.

## 6. Système d'exploitation Android :

L'Android est un système d'exploitation open source pour Smartphones, PDA et les terminaux mobiles conçu par Android qui est une startup rachetée par Google.

Le Système d'exploitation Android est fondé sur un noyau Linux. Il comporte une interface spécifique développée en java. Les programmes sont exécutés via un interpréteur JIT (Just in time).

Parmi les caractéristiques de notre système d'exploitation adopté, une machine virtuelle « Dalvik » optimisé pour les appareils mobiles, un navigateur intégré basé sur l'open source « WebKit », aussi une transition graphique optimisée et alimenté par une bibliothèque graphique 3D (OpenGL ES 2.0).

De plus, Android contient l'outil « SQLite » pour stocker des données structurées, il soutien des médias pour l'audio, la vidéo et des formats d'images telles que : MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF.( Plus de détails dans l'**annexe A**.)

## 7. SQLite database browser :

Le **SQLite database browser** permet de modifier ou de visualiser les bases de données sous Android stockées dans les répertoires de la forme **data/data/<package>/database**. On peut avoir plusieurs bases de données par application, cependant chaque fichier créé l'est selon le mode **MODE\_PRIVATE**, par conséquent les bases ne sont accessibles qu'au sein de l'application elle-même.

Dans notre application on n'a qu'une seule base de données nommée **HistoriqueGps.db** contenant une seule table nommée **HistoriqueGps** (id, matricule, latitude, longitude, altitude, vitesse, accuracy)

### iii. Implémentation de notre application :

#### a- Partie stockage :

Pour l'implémentation de cette partie nous avons choisi de sauvegarder les données sous forme de fichiers plats, on utilisant le SGBD SQL ite. SQLite est un moteur de base de données relationnel gratuit, qui est facilement portable, car le moteur et les données sont contenus dans un même fichier, donc c'est un excellent choix pour sauvegarder des données dans un Smartphone.

Pour gérer les fichiers SQLite, on crée une classe qui hérite de SQLiteOpenHelper qui inclut les méthodes :

onCreate() : code exécuter à la création des fichiers SQLite.

onUpgrade() : code exécuté lors de la mise à jour du schéma relationnel de la base de données.

### b- Partie transfert des données :

Pour l'implémentation de cette partie on a utilisé la bibliothèque **ksoap2-android-assembly-3.4.0-jar-with-dependencies.jar** et **java-json.jar**, pour ce faire :

- Nous avons copié les deux archives dans le répertoire libs.

Pour appeler un Service Web SOAP, une bibliothèque est nécessaire vu qu'il n'existe pas un tel outil dans le SDK d'Android. Il s'agit de la bibliothèque kSoap2 qui est copiée dans le répertoire libs. Elle permettra également de parser la réponse du service. La capture d'écran ci-dessous montre l'utilisation de la bibliothèque :

```
public final static String URL = "http://192.168.1.4:8080/WebServiceLocalisationGps/WebServiceLocalisationGps?wsdl";
public static final String NAMESPACE = "http://webService.gps.ummo.dz/";
public static final String SOAP_ACTION_PREFIX = "";
private static final String METHOD = "localisationGps";
//
private String matricul;
```

L'URI (Uniform Resource Identifier) du fichier WSDL est une information importante à connaître. Elle est nécessaire pour configurer les clients de ce web service. (Les web services seront plus détailler dans l'Annexe B).

- **Le format JSON** : (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. 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.

### c- Partie traitement

Le traitement le plus important dans notre système c'est bien évidemment la récupération des coordonnées GPS en utilisant la puce GPS du Smartphone.

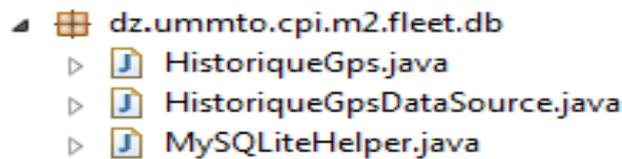
Dans cette section, nous allons faire une brève description des différents packages implémentant notre application, ainsi les différentes classes qui la constituent.

Pour faciliter le développement et la modification prochaine de notre application nous avons

choisi de la structurer en trois packages séparés.

➤ **Le Package dz.ummtto.cpi.m2.fleet.db :**

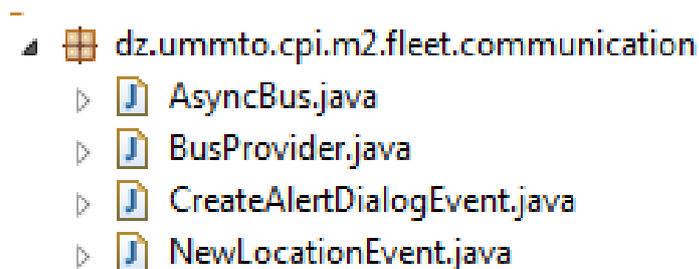
Ce package contient les classes java utilisées pour sauvegarder les données dans une base de données interne à l'application.



**Figure4.5.** Package dz.ummtto.cpi.m2.fleet.db

➤ **Le Package dz.ummtto.cpi.m2.fleet.communication:**

Ce package contient les classes java qui permettent le test de la connectivité ainsi que le transfert des coordonnées GPS en utilisant des requête HTTP sous format JSON vers un web service.



**Figure4.6:** Package dz.ummtto.cpi.m2.fleet.communication

➤ **Le Package dz.ummtto.cpi.m2.fleetgps:**

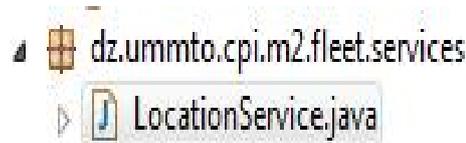
Ce package contient la classe MainActivity, classe principale de notre système.



**Figure4.7:** Package dz.ummtto.cpi.m2.fleetgps.

➤ **Le Package dz.ummto.cpi.m2.fleet.service :**

Ce package contient la classe java qui permet la récupération des coordonnées GPS en un intervalle de temps.



**Figure4.8. :** Package dz.ummto.cpi.m2.Fleet. Service.

➤ **La récupération des données géographiques:**

Pour pouvoir récupérer la position du véhicule qui est exprimé en termes de latitude, longitude, altitude, le Smartphone dispose d'une puce GPS intégrée qui permet la récupération des signaux satellites. Cela se fait en activant la fonctionnalité GPS du mobile, mais dans notre cas, on voudrait évidemment exploiter ces données d'où une certaine configuration ainsi qu'une implémentation bien spécifique, pour ce faire on :

Implémenter une classe GPS.java ou il sera spécifié :

- La source (le provider) qui va permettre la localisation du véhicule.
- Les interfaces LocationManager qui permet l'accès aux services de localisation du système et LocationListener qui est utilisé pour la réception des notifications du LocationManager lorsque la position change.
- Cette dernière interface va implémenter obligatoirement les méthodes suivantes :  
OnLocationChanged().  
OnProviderDisabled().  
OnProviderEnabled().  
OnStatusChanged().

#### **iv. Fonctionnement de l'application :**

##### **1. Présentation générale de la structure de l'application :**

L'application que nous avons développée comporte deux activités, une principale nommée **MainActivity.java** et une sous-activité nommée **GPS.java** que nous avons relié par les Intents. Pour ce faire, une modification au niveau du Manifest est effectuée comme on peut le voir dans la figure ci-dessous :

```

17
18
19② <application
20     android:allowBackup="true"
21     android:icon="@drawable/ic_launcher"
22     android:label="@string/app_name"
23     android:theme="@style/AppTheme" >
24② <activity
25     android:name=".MainActivity"
26     android:label="@string/app_name" >
27② <intent-filter>
28     <action android:name="android.intent.action.MAIN" />
29
30     <category android:name="android.intent.category.LAUNCHER" />
31 </intent-filter>
32
33
34 </activity>
35 <activity
36     android:name="com.master.m2.cpi.fleet.gps.GPS"
37     android:label="@string/title_activity_gps" />
38 </application>
39
40 </manifest>

```

**Figure4.9.** : Le fichier MANIFEST.

D'après la Figure on voit bien la présence de nos deux activités mise entre les balises `<activity></activity>`.

une autre modification est aussi importante dans le Manifest , et c'est l'ajout des permissions :

```

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>

```

**Figure4.9.** : Les permissions dans le Manifest.

Les figures ci-dessous représentent une vue après l'exécution de l'activité principale. Après clic sur l'icône de l'application on aura cette interface :



**Figure4.10** : écran principale de l'application

La figure représente l'activité qui a pour rôle d'afficher l'écran principale au lancement de l'application. On lance l'application à travers un clique sur le bouton «Start Location » qui apparait sur l'interface. On fait entrer le matricule du véhicule qu'on souhaite géolocaliser dans le champ (le champ est obligatoire). L'envoi des données se fait en arrière plan.

### **c. Partie web :**

Le langage de programmation que nous avons choisi pour le développement de notre application Web est le langage JAVA, développé par SUN Microsystems, il est disponible pour les principales plates formes du marché (LINUX, Windows ou autres) et il est totalement gratuit.

JAVA possède de nombreuses caractéristiques :

- ✓ Simple du faite que sa syntaxe soit basée sur celle du C++, mais dépouillée de tous les mécanismes complexes, redondants et inutiles (pointeurs,...).
- ✓ Performant, puissant, java est une plate forme de développement comportant une bibliothèque de classes très riche et de nombreux outils d'interfaces de programmations applicatifs (API).
- ✓ Interprété, portable et indépendant des architectures matérielles : Cette caractéristique est un avantage primordiale pour java face à des applications transmises par un réseau et exécutées sur des machines hétérogènes. Un programme

java est successivement compilé pour fournir un code intermédiaire indépendant de la plate forme d'exécution (le byte code) simple et rapide à traduire en langage machine.

✓ Riche : Un des aspects importants de l'environnement java est la richesse de ses librairies.

Java permet de développer de nombreux sortes de programmes dont :

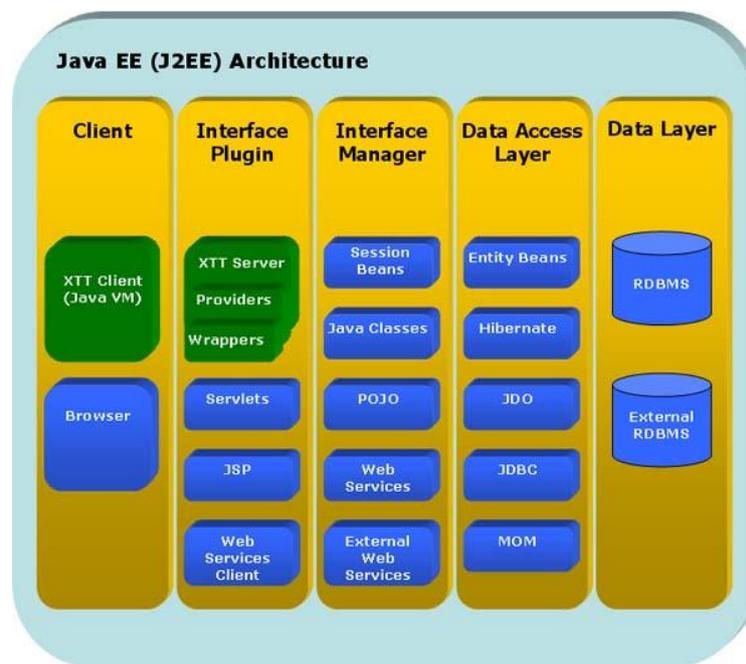
- ✓ Des applications, sous forme de fenêtre ou de console ;
- ✓ Des applets, qui sont des programmes JAVA incorporés à des pages web ;
- ✓ Servlets et jsp pour le développement d'applications web ;

Notre choix s'est porté pour le développement en JEE (Java Entreprise Edition) : développement des applications Web en JAVA grâce aux servlets et jsp.

### i. Présentation de l'application web :

Pour la réalisation proprement dite de notre système Web, on a opté pour la technologie JEE donc nous pouvons illustrer son architecture en couche dans ce qui suit.

D'après cette figure on peut dire que notre application est une application client/serveur multi-tiers. Elle est constituée essentiellement de trois grandes parties :



**Figure4.11.** : Architecture JEE.

- **Une partie stockage :** cette partie contient une base de données qui nous permet de stocker et sauvegarder nos données.

- **Une partie traitement** : dont laquelle on pourra effectuer les différents traitements sur nos données, cette dernière sera décomposée selon l'architecture JEE utilisées aux couches suivantes :
  
- **La couche [JDBC]** : (Java DataBase Connectivity) cette couche gère la connexion avec la (ou les) base(s) de données. Ici on utilisera la notion de pool de connexion. Un pool de connexion est un ensemble de connexions avec la base de données déjà instanciées. Cela permet aux requêtes de s'exécuter plus rapidement. On peut venir connecter plusieurs couches JPA sur la couche JDBC si nécessaire.
  
- **La couche [JPA]** : (Java Persistence Annotation ) est une couche d'abstraction de la couche JDBC. Elle permet notamment de faire du Mapping Relationnel-Objet (ORM, Object-Relationnal Mapping en anglais) qui consiste à modéliser la base de données sous forme d'objets pour une manipulation plus simple à travers le code Java (requêtes pré-écrites, gestion des liens entre les tables,...). Généralement la couche JPA contient une classe (entité) par table, des contrôleurs (fonctions de base implémentées) et des gestionnaires d'exceptions.
  
- **La couche [DAO]** : (Data Access Objet) est la couche Java qui nous permet l'accès aux données, elle découpée en deux sous couches Cette couche représente l'intelligence de l'application. Elle est composée d'un ensemble d'interfaces locales (local) et distantes (remote). Les DAO permettent d'accéder aux objets et proposent des méthodes de CRUD (Create, Read, Update, Delete). Un EJB (Entreprise Java Bean) sera piloté à partir d'une autre application distante ou locale (client EJB).
  - Une sous couche qui sera implémentée avec les deux couches précédentes dans un conteneur EJB, elle est chargée de gérer les opérations avec la couche JPA de cet EJB.
  - Une sous couche web service qui sera implémentée dans un conteneur web, elle est chargée d'exposer à des clients distants l'interface de l'EJB.
  
- **Web Services** : Cette couche a pour but de définir des services qui pourront être appelés selon le protocole SOAP. Ainsi les informations pourront circuler entre les applications sous forme de messages XML. Cela peut servir à faire communiquer deux applications qui peuvent être codées dans deux langages différents, en local ou à distance.

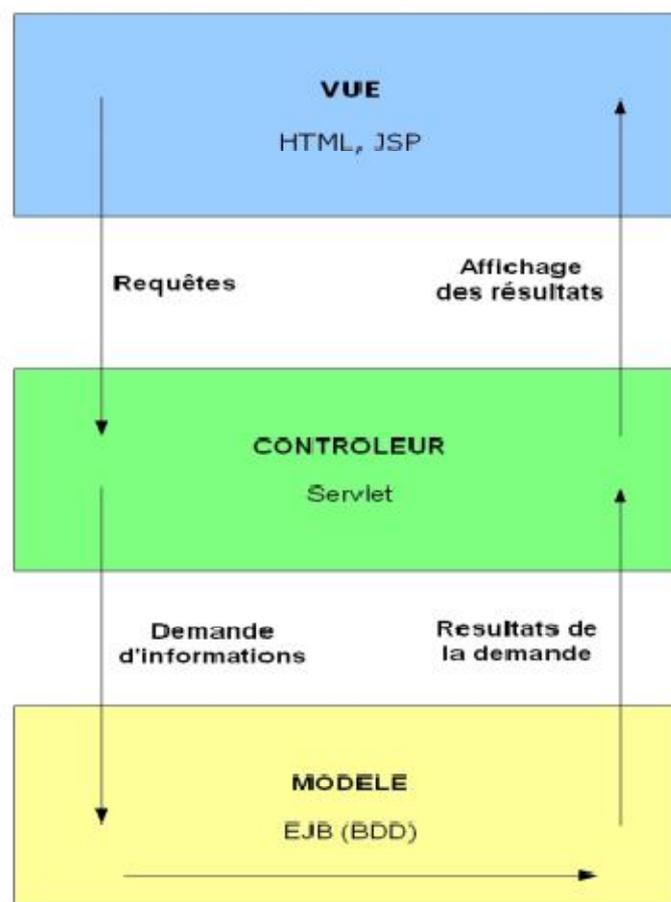
- **Une partie présentation** : Comme son nom l'indique, cette partie s'occupe de la présentation graphique du système pour les utilisateurs; elle est implémentée par des clients java.

## ii. Présentation du modèle MVC (Model-View-Controller) :

Une architecture MVC cherche à séparer trois choses : le Modèle, les Vues et les Contrôleurs. Les contrôleurs permettent de répondre aux actions de l'utilisateur. Chaque contrôle est associé à une vue, cette dernière permet de présenter l'information retournée à l'utilisateur. L'information renvoyée est dépendante des actions d'entrées de l'utilisateur (capturées par les contrôleurs, nous venons de le dire). Les liens (les traitements) sont réalisés par le modèle (le logique métier).

Si nous cherchons à mettre en œuvre une architecture MVC via un environnement JEE, on peut donc faire les rapprochements suivants :

- Un contrôleur est implémenté sous forme de servlet Java.
- Le modèle consiste en l'implémentation du logique métier d'une application web.
- Chaque vue est implémentée sous forme d'une page html ou page jsp.



**Figure4.12.** : Modèle MVC.

### iii. Langages de programmation : [19]

#### 1. *Les Servlets*

Une servlet est un programme java qui fonctionne sur un serveur Web compatible J2EE et dont le rôle est d'apporter une réponse à une requête. Elle reçoit une requête du client, elle effectue des traitements et renvoie le résultat. La technologie des servlets n'est qu'un ensemble de classes. Pour fonctionner convenablement elles ont besoin d'une machine virtuelle JAVA et de l'ensemble des autres classes intégrées à l'API standard du langage JAVA. Une servlet est une application développée dans un contexte client-serveur.

#### 2. *Les Jsp*

JSP « Java Server Pages » est un fichier contenant du code HTML et des fragments de code Java exécutés sur le moteur de Servlets Comparable aux langages côté serveur de type PHP, ASP, ... Les JSP permettent d'introduire du code Java dans des tags prédéfinis à l'intérieur d'une page HTML. Elles permettent donc de mélanger la puissance de Java côté serveur et la facilité de mise en page d'HTML côté client. Les fichiers JSP possèdent par convention l'extension .jsp .

#### 3. *HTML/XHTML*

HTML (Hyper Text Markup Language/eXtensible HyperText Markup Language) est un langage de balisage (dit aussi langage de marquage) qui permet de structurer le contenu des pages web dans différents éléments. Historiquement, les langages de balisage sont issus du langage SGML (Standard Generalized Markup Language) créé en 1986 pour structurer des contenus très divers. Ce langage s'est révélé trop complexe pour être appliqué tel quel au Web, d'où la nécessité d'en créer une version allégée respectant les mêmes principes essentiels.

Le HTML est un langage de description (et non un langage de programmation) qui va nous permettre de décrire l'aspect d'un document, d'y inclure des informations variées (textes, images, sons, animations) et d'établir des relations entre ces informations grâce aux liens hypertextes. HTML est un langage qui définit la structure logique d'un document web. Il a été comme un langage de structuration logique et utilisé comme langage de présentation et de mise en page.

Les pages produites avec ce langage portent le nom de page HTML. Elles sont en général stockées sur un serveur Web, qui les fournira à la demande. Du côté client, un navigateur est chargé de l'interprétation de pages HTML et de leur affichage. On dit donc que le navigateur affiche des pages web.

## 4. JavaScript

C'est un langage de scripts principalement utilisé dans les pages web interactives.

JavaScript fait ce que HTML ne peut pas. Il permet de traiter les saisies de l'utilisateur faites au clavier ou avec la souris et d'en réagir avec des sorties à l'écran ou des modifications dynamiques dans la page Web affichée. L'utilisateur introduit donc quelques valeurs, JavaScript fait les calculs et affiche le résultat sans qu'il y ait d'échanges entre le client et le serveur web.

## 5. CSS (Cascading Style Sheets) :

Il s'agit d'un langage de programmation directement complémentaire qui fut développé spécialement pour le HTML. Les CSS permettent de disjoindre le contenu et la forme d'une page HTML ainsi la taille des pages diminue et les traitements sur les fichiers deviennent aisés puisque il y aura beaucoup moins de tri à faire entre contenu et mise en forme.

## 6. Le langage de requêtes (JPQL)

Le langage JPQL (Java Persistence query language) est un langage de requête dont la grammaire est définie par la spécification J.P.A. (Java Persistence API) Il est très proche du langage SQL dont il s'inspire fortement mais offre une approche objet. On trouve également le nom d'EJBQL dans la littérature Java, il s'agit du nom donné à ce langage dans la norme EJB2.

### iv. Les Framework : [20]

- un **Framework** est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture). Un Framework se distingue d'une simple bibliothèque logicielle principalement par :
  - son caractère générique, faiblement spécialisé, contrairement à certaines bibliothèques ; un Framework peut à ce titre être constitué de plusieurs bibliothèques chacune spécialisée dans un domaine. Un Framework peut néanmoins être spécialisé, sur un langage particulier, une plateforme spécifique, un domaine particulier : Reporting, mapping, etc.
  - le cadre de travail (traduction littérale de l'anglais : Framework) qu'il impose de par sa construction même, guidant l'architecture logicielle voire conduisant le développeur à respecter certains patterns ; les bibliothèques le constituant sont alors organisées selon le même paradigme.

## 1. EJB :

La technique Enterprise JavaBeans (EJB) est une architecture de composants logiciels côté serveur pour la plate-forme de développement JEE.

Cette architecture propose un cadre pour créer des composants distribués (c'est-à-dire déployés sur des serveurs distants) écrit en langage de programmation Java hébergés au sein d'un serveur applicatif permettant de représenter des données (EJB dit *entité*), de proposer des services avec ou sans conservation d'état entre les appels (EJB dit *session*), ou encore d'accomplir des tâches de manière asynchrone (EJB dit *message*). Tous les EJB peuvent évoluer dans un contexte transactionnel.

## 2. JSF :

JavaServer Faces (abrégé en JSF) est un framework Java basé sur la notion de composants et destiné au développement d'applications Web. JSF est techniquement comparable à Swing ou SWT sauf qu'ils sont utilisés pour le développement d'application bureau, en JSF et en Swing l'état d'un composant (représenté par un objet java) est enregistré lors du rendu de la page, pour être ensuite restauré au retour de la requête.

Une page JSF est une page xhtml (ou jsp) liée aux Managed Bean via le langage EL.

JSF est principalement constitué de :

- Un ensemble d'APIs pour la représentation et la gestion des composants, de leur état, des évènements, de la validation des entrées et la conversion des sorties, l'internationalisation et l'accessibilité ainsi que la navigation inter-vues.
- Deux jeux de composants standards (affichage de texte, saisie de texte, tables, zone à cocher, etc.): html et core.
- Deux bibliothèques de balises JSP (une pour chaque jeu de composants) pour permettre l'utilisation de JSP pour la construction de vues JSF.
- Un modèle évènementiel côté serveur.
- Les Managed-Beans : qui forment la couche contrôle de JSF.
- Unified Expression Language: abrégé en EL ou langage d'expressions unifié pour JSF et JSP 2.0. Il permet de lier les composants aux managed-beans.

L'objectif de JSF est de faciliter le développement des interfaces utilisateurs des applications web, or les deux jeux de composants standards de JSF (html et core) s'avèrent limités et insuffisants pour le développement d'applications d'entreprise. Des jeux de composants additionnels qui offrent de nouveaux composants plus riches sont indispensables pour le développement en JSF, Primefaces en offre un qui a prouvé son efficacité.

### **3. EclipseLink**

EclipseLink est un Framework JPA 2.0 open source de mapping objet-relationnel pour les développeurs Java. Il fournit une plateforme puissante et flexible permettant de stocker des objets Java dans une base de données relationnelle et/ou de les convertir en documents XML. EclipseLink est dérivé du projet TopLink de la société Oracle. Il supporte un certain nombre d'API relatives à la persistance des données et notamment la JPA.

### **4. PrimeFaces :**

PrimeFaces est un framework open source, qui peut être assimilé à une suite de composants d'interfaces riches pour JSF.

JSF apporte des éléments très basiques d'UI par contre PrimeFaces va plus loin et propose plus de 100 composants, plus de 35 thèmes, des composants mobiles et bien plus encore.

C'est actuellement de loin la suite pour JSF la plus populaire, ainsi qu'un outil graphique très populaire partout pour le développement de Java Web.

Ces bibliothèques complémentaires :

- PrettyFaces
- OmniFaces (bibliothèque utilitaire pour JSF)
- PrimeFaces Extensions.

### **5. JAAS :**

Le service d'authentification et d'autorisation de Java (JAAS) est un ensemble de paquets qui permet à des services d'authentifier et imposer des contrôles d'accès sur des utilisateurs. Il implémente une version en technologie Java du framework standard PAM (Pluggable Authentication Module) et supporte les autorisations basés sur les utilisateurs.

## **v. Outils de développement :**

### **1. IDE NetBeans :**

NetBeans est un environnement de développement intégré (EDI), placé en open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License) et GPLv2. En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C++, JavaScript, XML, Groovy, PHP et HTML de façon native ainsi que bien d'autres (comme Python ou Ruby) par l'ajout de greffons. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web). Conçus en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Development Kit JDK est requis pour les développements en Java.

NetBeans constitue par ailleurs une plate forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate forme.

## **2. Serveur d'application JEE GlassFish :**

« Serveur d'application : Un serveur d'applications est un logiciel d'infrastructure offrant un contexte d'exécution pour des composants applicatifs. »

GlassFish est le nom du serveur d'applications Open Source Java EE 5 et qui sert de fondation au produit Sun Java System Application Server de Sun Microsystems. Sa partie Toplink persistance provient d'Oracle. C'est la réponse aux développeurs Java désireux d'accéder aux sources et de contribuer au développement des serveurs d'applications de nouvelle génération de Sun.

## **3. SGBD PostgeSQL :**

« Un SGBD (Système de Gestion de Base de Données) permet de définir les structures physiques et logiques qui constituent une base de données, et qui servent au stockage d'important volumes d'information qu'il est ensuite possible d'interroger rapidement et accessibles par des utilisateurs simultanés. »

PostgreSQL est un système de gestion de bases de données relationnelles objet (ORDBMS) fondé sur *POSTGRES, Version 4.2™*. Ce dernier a été développé à l'université de Californie au département des sciences informatiques de Berkeley. POSTGRES est à l'origine de nombreux concepts qui ne seront rendus disponibles au sein de systèmes de gestion de bases de données commerciales que bien plus tard. PostgreSQL est un descendant Open Source du code original de Berkeley. Il supporte une grande partie du standard SQL tout en offrant de nombreuses fonctionnalités modernes :

- requêtes complexes ;
- clés étrangères ;
- triggers ;
- vues
- intégrité transactionnelle ;
- contrôle des versions concurrentes (MVCC ou multiversion concurrency control).

De plus, PostgreSQL peut être étendu par l'utilisateur de multiples façons. En ajoutant, par exemple :

- de nouveaux types de données ;
- de nouvelles fonctions ;
- de nouveaux opérateurs ;
- de nouvelles fonctions d'agrégat ;
- de nouvelles méthodes d'indexage ;
- de nouveaux langages de procédure.

- Et grâce à sa licence libérale, PostgreSQL peut être utilisé, modifié et distribué librement, quel que soit le but visé, qu'il soit privé, commercial ou académique.

#### d. Implémentation des différentes parties de l'application :

##### i. Partie stockage :

Pour l'implémentation de cette partie, Nous avons créé avec le SGBD PostgreSQL une base de données [geolocalisation\_GPS] contenant les différentes tables de chapitre conception.

Voici un exemple de script de création d'une des tables :

```
CREATE TABLE gps_vehicule
] (
  num_immatriculation character varying(15) NOT NULL,
  num_serie character varying(25),
  id_marque_veh integer,
  id_type_marque_veh integer,
  precedent_num_immatric character varying(15),
  annee_premiere_mise_circulation integer,
  id_vehic integer NOT NULL,
  id_client integer,
  CONSTRAINT pk_vehic PRIMARY KEY (id_vehic),
  CONSTRAINT fk_client FOREIGN KEY (id_client)
    REFERENCES adm_user (id_user) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk_type_marque_veh FOREIGN KEY (id_marque_veh, id_type_marque_veh)
    REFERENCES gps_type_marque_vehicule (id_marque_veh, id_type_marque_veh) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
-)
```

Figure4.13. : Script de création de la table « véhicule ».

Après avoir créé cette base de données, on a établie une connexion NetBeans /PostgreSQL à l'aide de pilote JDBC.

##### ii. Partie Traitement

Dans cette section, nous allons faire une brève description des différents projets NetBeans implémentant cette partie, ainsi les différentes couches JEE qui la constituent.

###### ➤ le projet GeolocalisationGps :

Ce projet est un projet JEE de type Entreprise Application, c'est lui qui nous permet le déploiement de nos services au sein d'un serveur Glassfish.

Ce dernier est composé de deux modules :

- Un module de type web application qui implémente notre système.
- Un module EJB dont dépend notre système.

La figure suivante est une capture d'écrans montrant ce projet NetBeans.



Figure4.14. : Le projet GeolocalisationGps.

### ➤ Le module GeolocalisationGPS-ejb :

C'est un projet de type EJB Module de la catégorie Java EE des projets NetBeans.

Ce module construit un composant logiciel distribué (c'est-à-dire déployé sur le serveur distant Glassfish), il implémente les couches : [JDBC], [JPA/Hibernate], Ejb [DAO] de notre application.

Afin que notre module EJB, puisse exploiter les données de notre base de données, en assurant leurs persistances, on a configuré ça sous forme des fichiers XML qui ont été créés dans ce module.

### ➤ Ajout d'une ressource JDBC au serveur Glassfish :

On a ajouté une ressource JDBC au serveur Glassfish afin qu'il puisse accéder à la base de données. Ce qui génère le fichier de configuration [glassfish-resources.xml] dans le répertoire « Server Resources » du module EJB.

### ➤ Création d'une unité de persistance :

L'unité de persistance [persistence.xml] configure la couche JPA : elle indique l'implémentation JPA utilisée, ainsi elle nous permet d'indiquer le type de SGBD à gérer, ce fichier est créé dans le répertoire « Configuration Files » du module EJB.

### ➤ Création des entités JPA :

Pour donner une image objet a notre base de données, on a crée plusieurs packages dans le répertoire « Source Packages » du module EJB, ces package implémentent la couche JPA de l'architecture précédente, ils contiennent des entités JPA.

Chaque entité de ces packages encapsule les lignes d'une table de la base de données, donc il y'a autant d'entités dans ces packages que de tables dans la base de données.

**iii. Partie présentation :**

Pour illustrer l'échange de données entre le client web et notre système, ainsi que la communication entre ce système web et le système mobile décrit précédemment nous avons réalisé une application web qui interagit avec l'application mobile et permet l'exploitation et la gestion des données récupérées, en les déployant à travers divers interfaces.

À travers les interfaces présentées ci-dessous, nous visons à donner une vue générale de notre application conçue.

**e. Interfaces de l'application web :**

Nous allons présenter, dans ce qui suit, les principales interfaces de notre application web en illustrant leur fonctionnement.

**i. Interfaces communes :**

**a- Page Accueil :**

C'est la première page qui apparaît à l'utilisateur dans le navigateur lors de la connexion au site. Elle constitue la porte d'entrée pour accéder à toutes les pages de notre application.



**Figure4.15. :** Interface « Accueil ».

Cette page est destinée à tous ceux qui visitent notre application. Elle comporte un menu comme toutes les pages de cinq onglets (Accueil, Présentation, Inscription, authentification), au centre de la page il y a un slide qui comporte des images correspondante au sujet qui est bien la géolocalisation.

**b- Page fonctionnalité :**

Cette page présente les fonctionnalités que la plateforme peut offrir à ses clients.



**Figure 4.16. :** Interface « fonctionnalités de la plateforme ».

**c- Page inscription :**

La page inscription permet une inscription de toute personne qui accède à la plateforme, l'inscription se fait en remplissant le formulaire illustrer dans la figure ci-dessous et valider en cliquant sur le bouton « valider ».

Une fois l'inscription faite, il reste à quelle soit valider par l'administrateur du site.

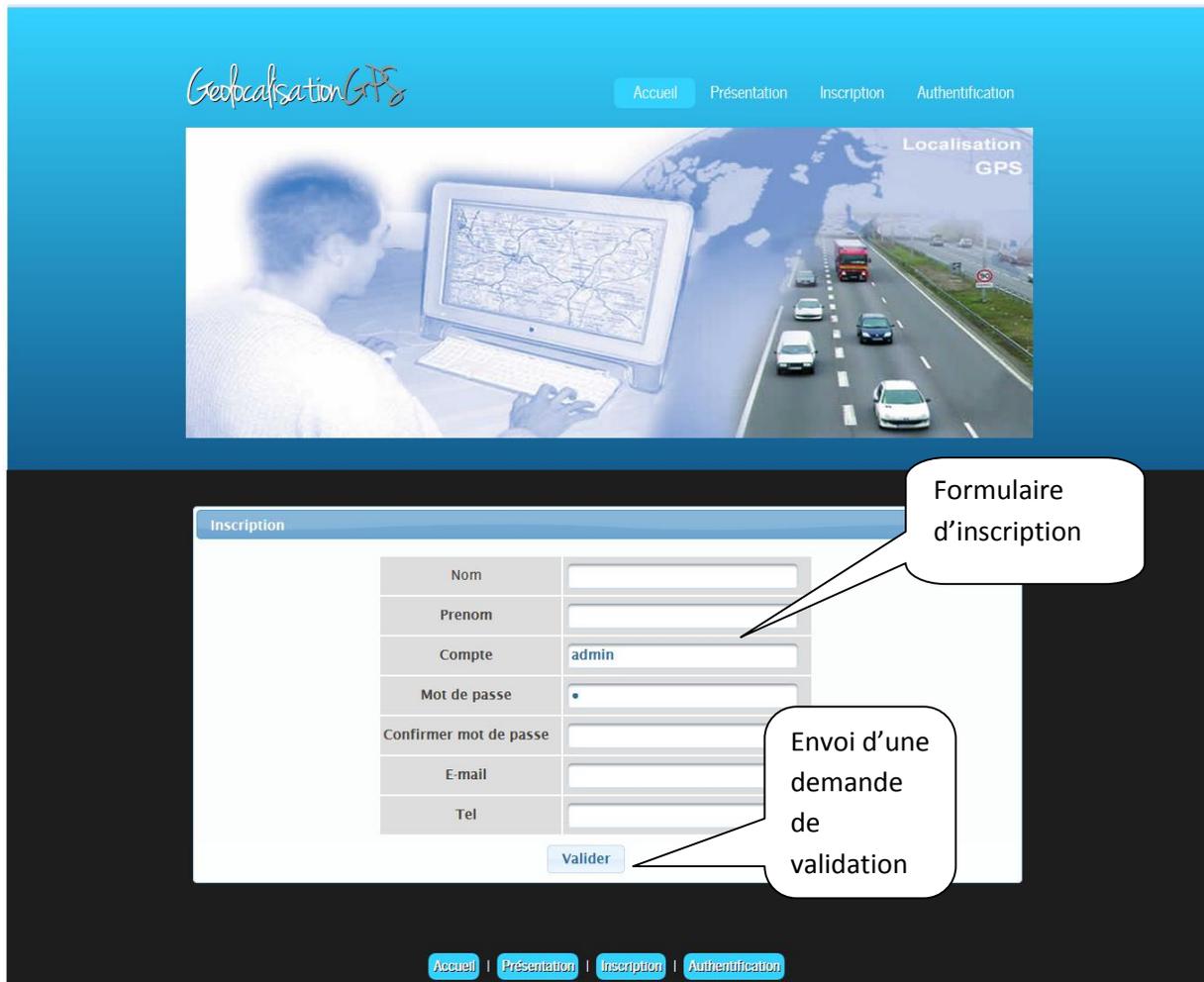


Figure 4.17. : Interface « Inscription »

**d- Page Authentification :**

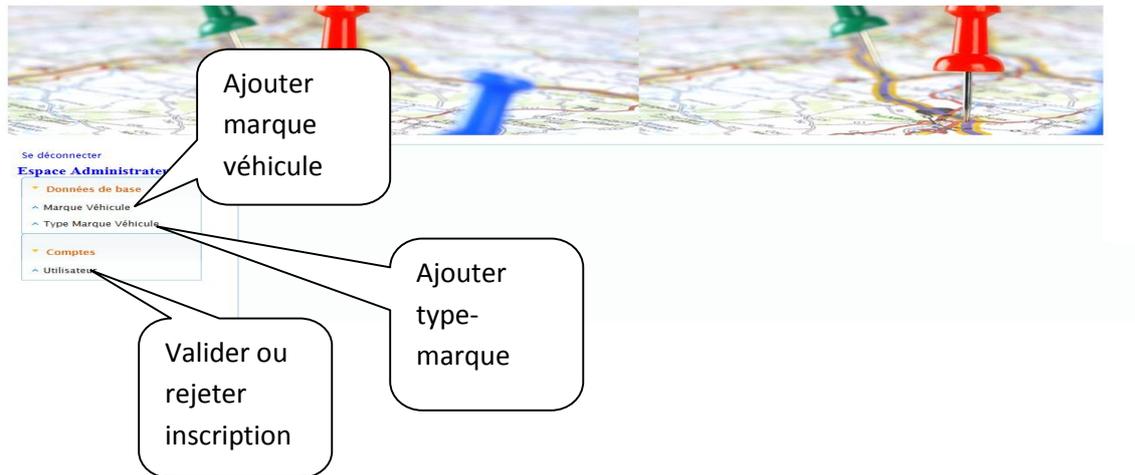
La page ci-dessous donne accès à l’authentification de l’administrateur ou du client.



Figure4.18. : Interface « authentification »

ii. **Espace Administrateur :**a- **Page d'accueil :**

Une fois que l'administrateur à saisi son nom d'utilisateur ainsi que son mot de passe, le système le dirige vers son espace.



**Figure4.19.** Interface « espace administrateur »

**b- Page marque véhicule :**

L'administrateur peut ajouter un type de véhicule en accédant à la page illustré ci-dessous.

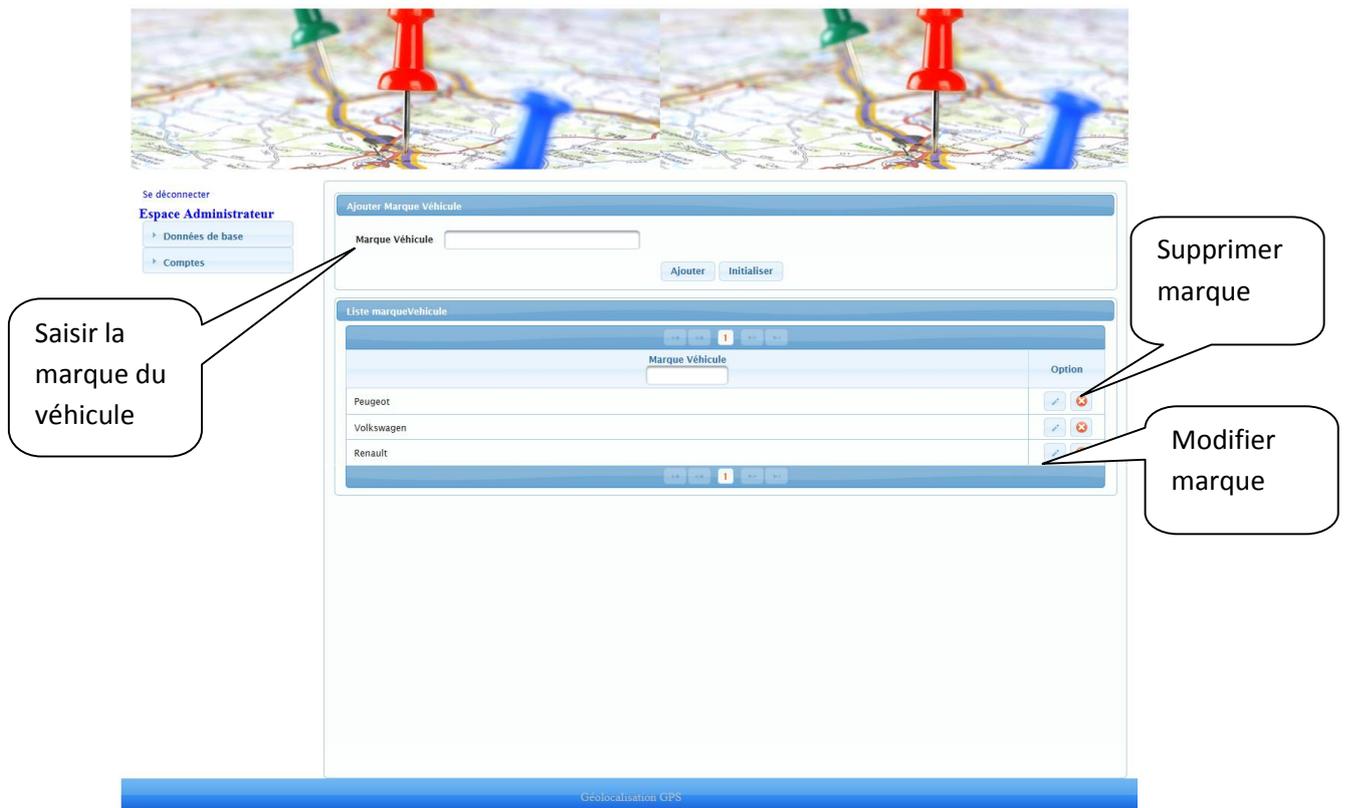
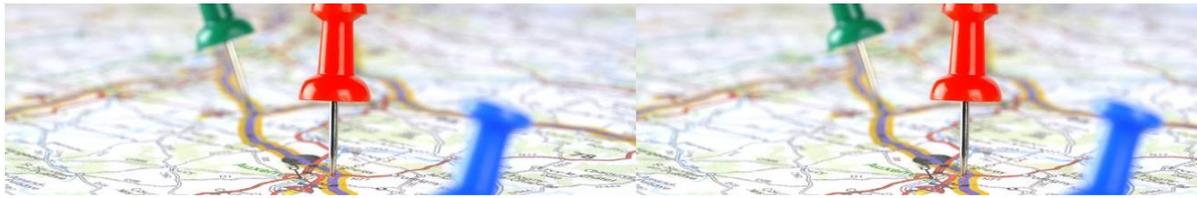


Figure4.20. : interface « ajouter marque véhicule »

**c- Page utilisateur :**

Une fois qu'une inscription est faite, on incrémente la liste des utilisateurs. L'administrateur a le droit de valider ou refuser une inscription.

Cette opération se fait à l'aide du champ « état », l'administrateur change la valeur de ce champ par rapport à la validation ou le refus d'une inscription. Aussi il peut à tout moment désactiver ou activer un client.



Se déconnecter

**Espace Administrateur**

- Données de base
- Comptes

Liste Utilisateurs

Nom	Prenom	Date de naissance	Compte	Adresse	E-mail	Téléphone	Etat	Option
aggoun	karim		admin				Activer	[Action icons]
aggoun	karim		user				Activer	[Action icons]
lynda	meghenez		geolocalisation		lyndam90@live.fr		Déactiver	[Action icons]
lynda	meghenez		admin				Déactiver	[Action icons]

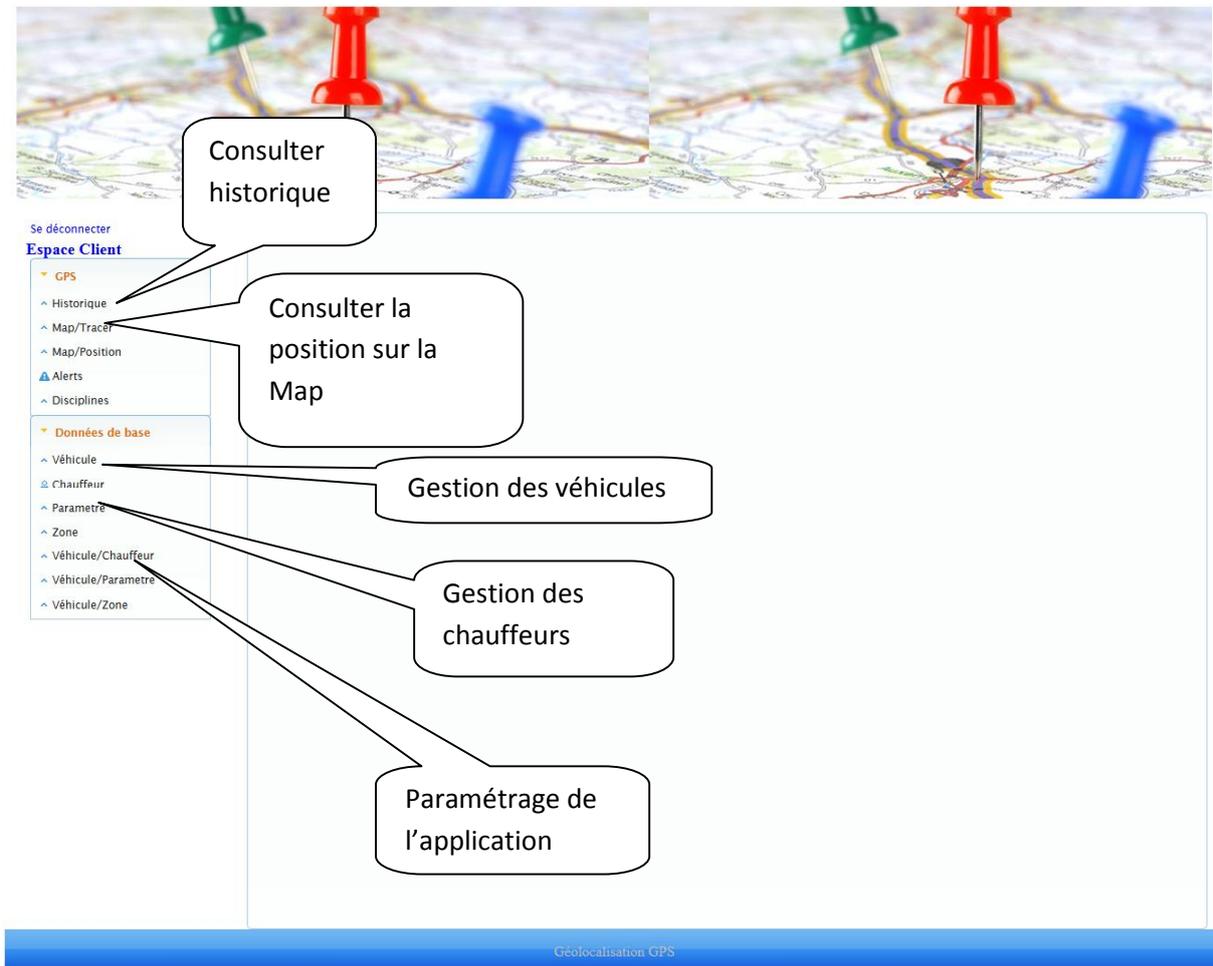
Le champ « état » pour valider ou pas une inscription

Geolocalisation GPS

Figure4.21. : Interface « gestions des inscriptions »

**iii. Espace client :**

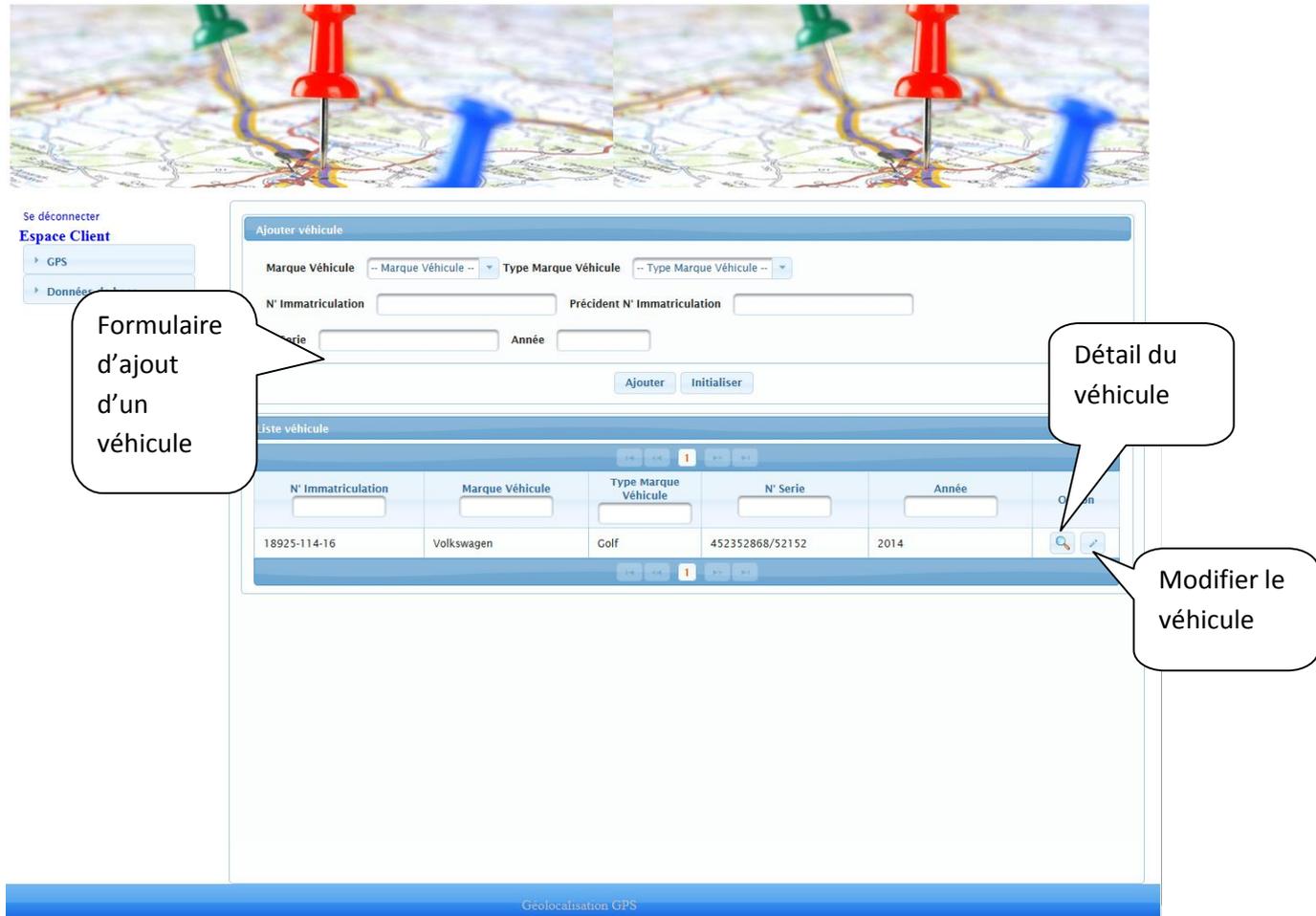
Une fois que le client a saisi son nom d'utilisateur et son mot de passe le système le dirige vers son espace.



**Figure 4.22.** Interface « espace Client »

**a- Page véhicule :**

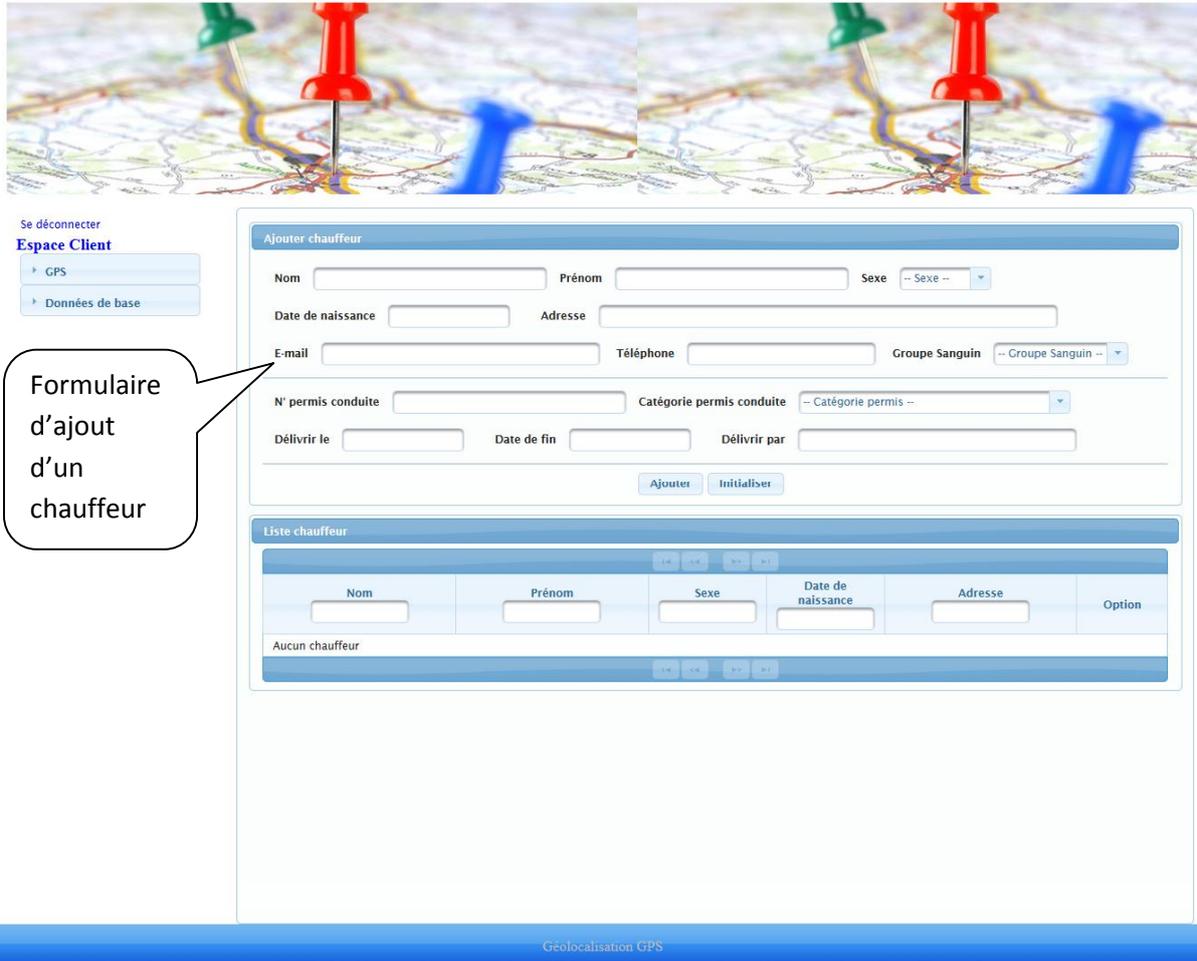
Dans la page ci-dessous un client peut ajouter un véhicule, il demandé de remplir un formulaire d'ajout de véhicule et l'ajouter ainsi il sera afficher dans la liste des véhicules.



**Figure4.23. :** Interface «Ajout d'un véhicule »

**b- Page Chauffeur :**

Dans la page ci-dessous un client peut ajouter un chauffeur, il est demandé de remplir un formulaire d'ajout d'un chauffeur et l'ajouter ainsi il sera afficher dans la liste des chauffeurs



**Figure4.24. :** Interface «Ajout d'un Chauffeur »

**c- Page Historique :**

Le client peut faire une recherche dans la liste historique en entrant le matricule du véhicule ainsi que la date recherchée, la page ci-dessous lui affiche toute les entrées historiques qui correspondent à sa recherche.

Se déconnecter  
Espace Client  
GPS  
Données de base

Rechercher historique

Matricule: 18925-114-16    Date début: 06/01/2015 00:00:00    Date Fin: 07/02/2015 00:00:00    Rechercher

Matricule	Vitesse	Neure Envoyer	Lat	Long	Long	Option
18925-114-16	15.0	05/06/2015 00:00:00	36.6997	3.0577	56.0	[Icon]
18925-114-16	26.0	05/06/2015 00:10:00	36.6998	3.0578	60.0	[Icon]
18925-114-16	58.0	05/06/2015 00:20:00	36.6999	3.0579	67.0	[Icon]
18925-114-16	45.0	05/06/2015 00:30:00	36.6999	3.058	45.0	[Icon]
18925-114-16	45.0	05/06/2015 00:40:00	36.69	3.0581	45.0	[Icon]

Rechercher l'historique d'un véhicule avec tel matricule de date début jusqu'à date fin

**Figure4.25.** Interface « Recherche et consultation Historique »

**d- Page alerte :**

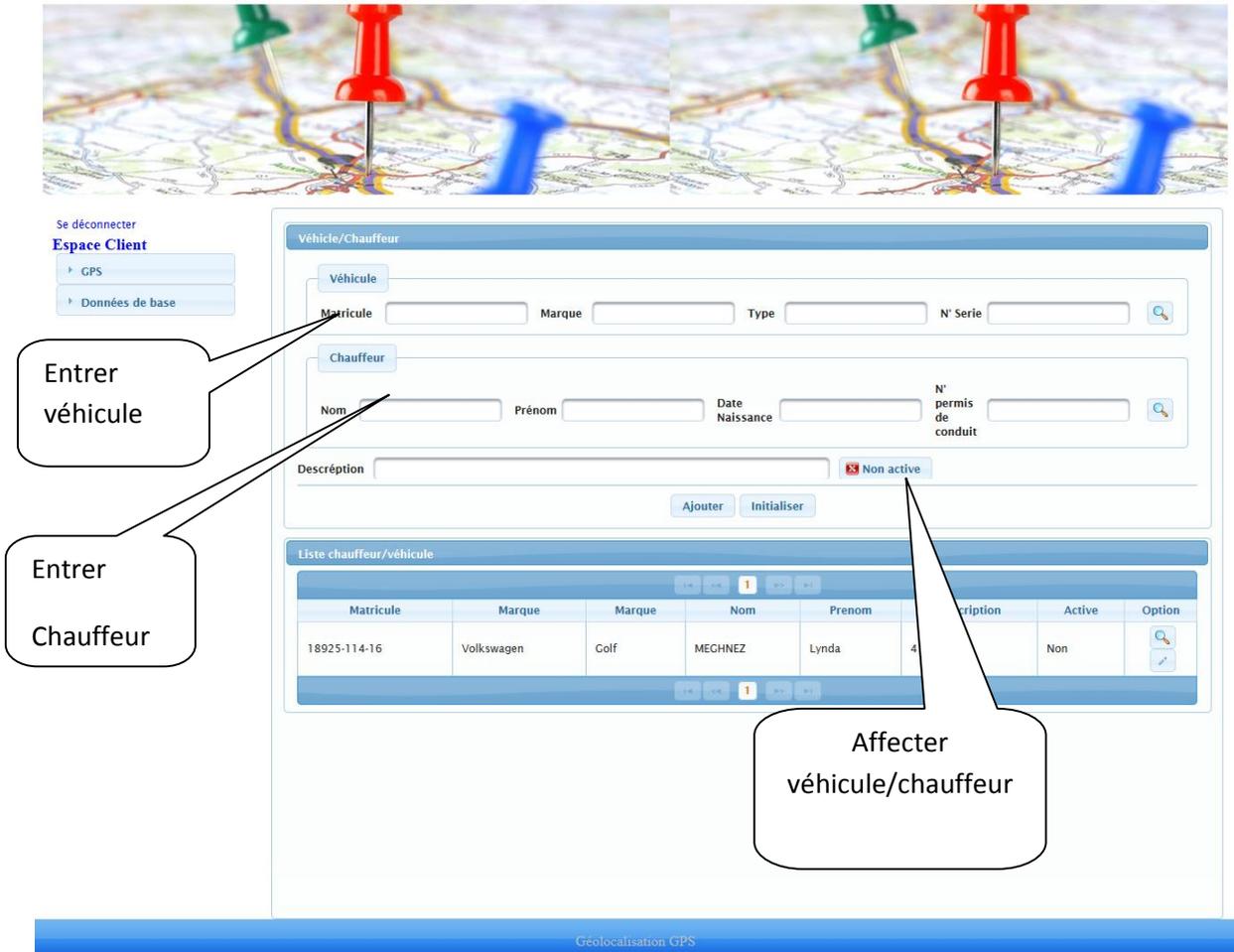
Le client peut consulter les différentes alertes enregistrées par le système suite à un non respect à l'un des paramètres définis.

Alert	N° Immatriculation	Vitesse	Neure Envoyer	Option
Hors Zone	18925-114-16	15.0	05/06/2015 00:00:00	
Hors Zone	18925-114-16	26.0	05/06/2015 00:10:00	
Hors Zone	18925-114-16	0.0	29/06/2015 23:48:45	
Hors Zone	18925-114-16	0.0	29/06/2015 23:48:54	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:04	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:14	
Hors Zone	18925-114-16	26.0	05/06/2015 00:10:00	
Hors Zone	18925-114-16	0.0	29/06/2015 23:48:45	
Hors Zone	18925-114-16	0.0	29/06/2015 23:48:54	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:04	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:14	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:24	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:34	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:44	
Hors Zone	18925-114-16	0.0	29/06/2015 23:49:54	

**Figure 4.26.** Interface « Consulter les alertes »

**e- Page affectation véhicule/Chauffeur :**

Après ajout d'un véhicule et d'un chauffeur on peut affecter a une véhicule un chauffeur pour une période donnée, le bouton « active » nous permet de faire cette opération.il est « non active » par défaut.



**Figure4.27. :** Interface « affecter véhicule/chauffeur »

**f- Page map/tracer :**

L'utilisateur peut visualiser la trace des déplacements des véhicules sur la Map.

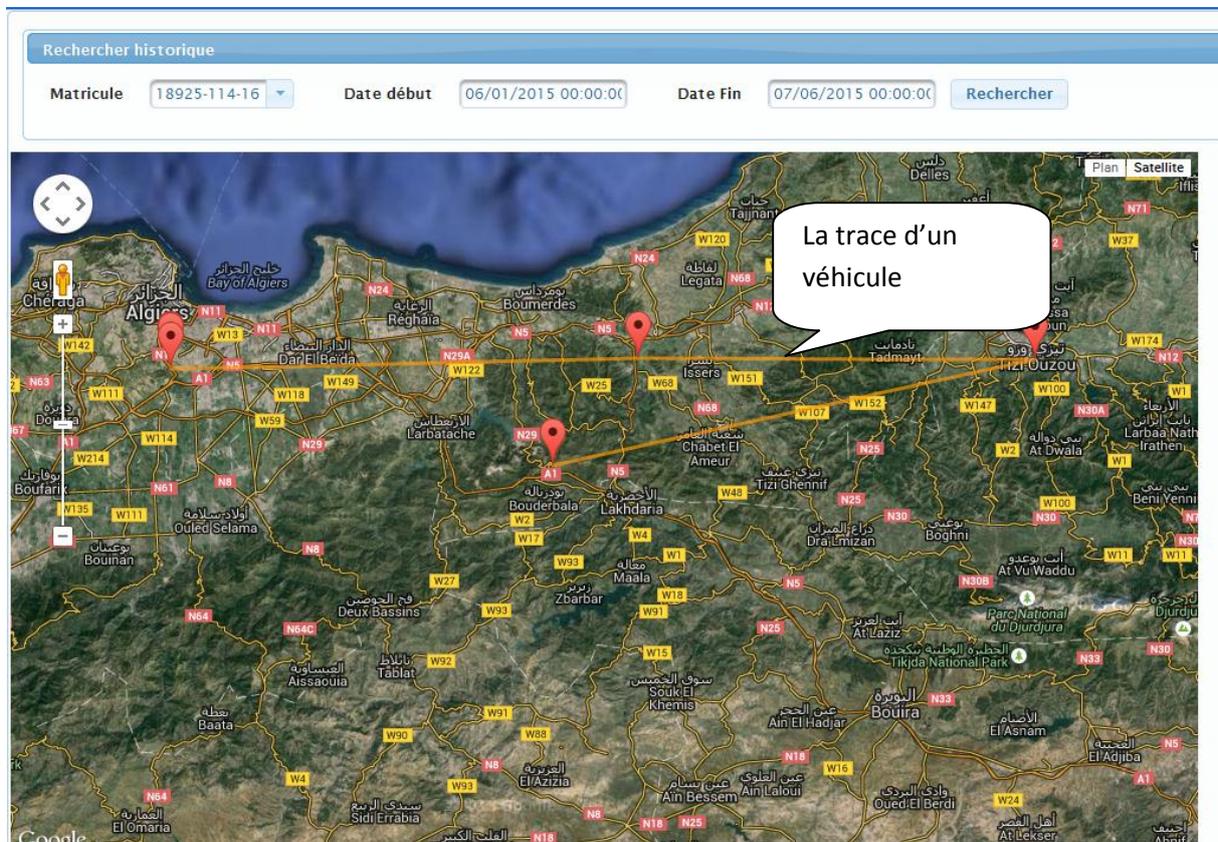


Figure4.22. : Interface « Map/tracer ».

**g- Page map/position :**

La page ci-dessous affiche la position du véhicule à instant donné.

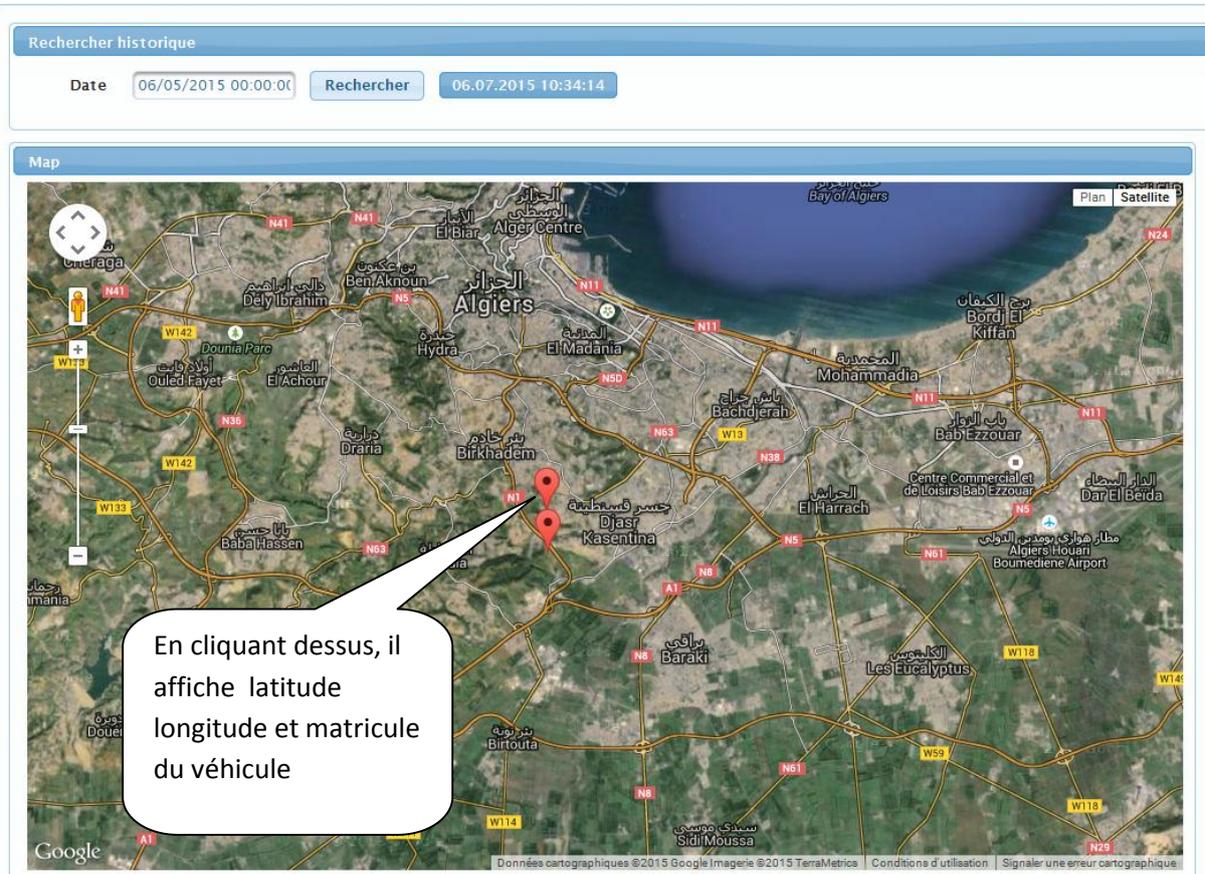


Figure4.23. : Int

.....erface « Map/position ».

## **V. Conclusion**

Dans ce chapitre nous avons présenté les outils utilisés pour le développement de notre application, ensuite nous avons donné une description du fonctionnement de notre application en présentant ces différentes interfaces et les différents traitements que peut faire l'utilisateur.

# *Conclusion Générale*

## Conclusion Générale

---

L'élaboration de notre travail était dans le but de concevoir un outil de géolocalisation en temps réel d'une flotte de véhicules, cet outil se compose d'une application dédiée aux terminaux mobiles disposant de la plateforme Android pour la récupération des données géographiques des véhicules, et une application web de gestion et de suivi des tournées de ces derniers. Cet outil permet une localisation en temps réel de chaque véhicule embarquant un Smartphone et on a installé l'application mobile et disposant d'une connexion au réseau GPRS/3G pour l'envoi des données collectées vers un serveur de cartographie. Pour ce faire, on a eu recours à différentes technologies et outils de localisation et d'orientation jugés nécessaires pour aboutir à l'objectif de notre application.

L'étude réalisée sur le domaine de la géolocalisation m'a permis de bien comprendre plusieurs aspects techniques et fonctionnels tournants autour de ce concept, et comme expérience pour la conduite d'un projet, ce travail m'a permis :

- Une acquisition des méthodes et techniques d'analyse et de conception ;
- se familiariser avec plusieurs langages à savoir : JAVA, SQL, XML, Android ;
- enrichir les connaissances sur les web services, comment les développer, les déployer,
- se familiariser avec les environnements de développement orienté web, à savoir : Netbeans, PostgreSQL pour les bases de données et Eclipse pour le développement mobile, Sqlite pour les fichiers plats et Edraw Max, Visio pour les diagrammes UML.

Enfin, il est impossible d'affirmer que ce travail est complet d'où plusieurs améliorations peuvent être apportées.

En guise de perspectives il est envisageable d'enrichir cette application avec de nouvelles fonctionnalités, ainsi que d'améliorer les interfaces utilisateurs. Les possibilités d'évolution de ce projet sont nombreuses allant par exemple l'ajout d'alertes comme dépassement temps roulage, dépassement temps arrêt max, ajouter la fonctionnalité d'alerter le point cible en étant assez proche de lui.

# *Annexe*

Dans cette annexe nous allons exposer certaines notions sur le système Android et son architecture.

- **Plateforme Android :**

Android est un système d'exploitation, basé sur un noyau linux, destiné aux appareils mobiles (Smartphone, PDA, tablettes, etc.).

**Android** est développé par un consortium : l'**Open Handset Alliance**. L'Open Handset Alliance regroupe beaucoup de sociétés liées aux nouvelles technologies (Intel, HTC, Motorola, Garmin,.....), mais le principal contributeur est **Google**.

Android est à l'heure actuelle le système d'exploitation pour Smartphones et tablettes le plus utilisé.

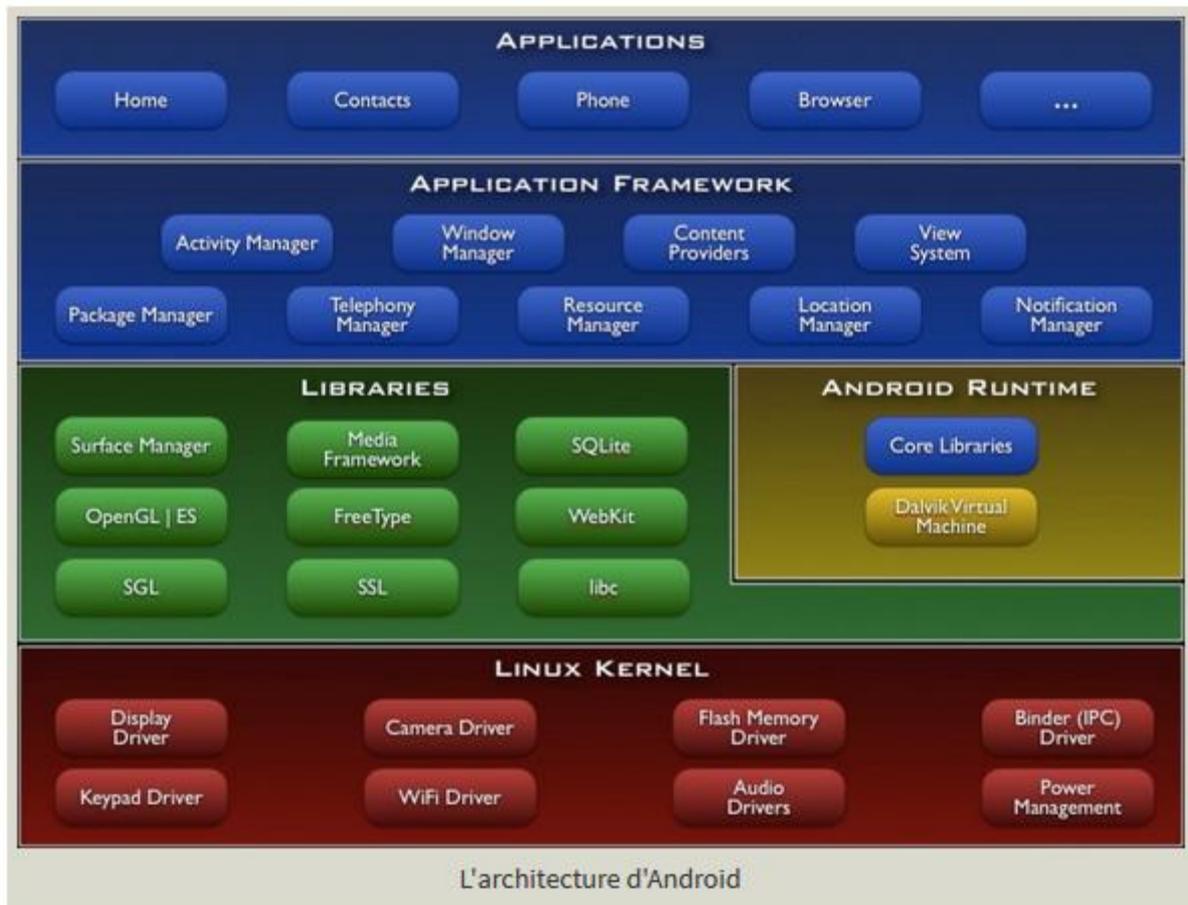
Android en tant que plate-forme :

- elle est innovante car toutes les dernières technologies de téléphonie y sont intégrées : écran tactile, accéléromètre, GPS, appareil photo numérique etc.
- en tant que développeur vous n'avez pas à connaître un langage peu utilisé ou spécifique : le développement sur la plateforme Android est en effet réalisé en langage Java, un des langages de programmation les plus répandus, juste il faut savoir qu'un programme fonctionnant sur la machine virtuelle java développée par Sun (JVM) ne fonctionnera pas sous Android. Car ce dernier étant destiné à des appareils mobiles ayant peu de puissances (par rapport à un ordinateur classique), Google a développé sa propre machine virtuelle : **Dalvik**.
- elle est ouverte, la plate-forme Android est fournie sous licence Open source, permettant à tous les développeurs – et constructeurs- de consulter les sources et d'effectuer les modifications qu'ils souhaitent.

- **Architecture d'Android :**

La figure suivante schématise l'architecture d'Android. Ce schéma provient du site d'Android destiné aux développeurs.

## Annexe A

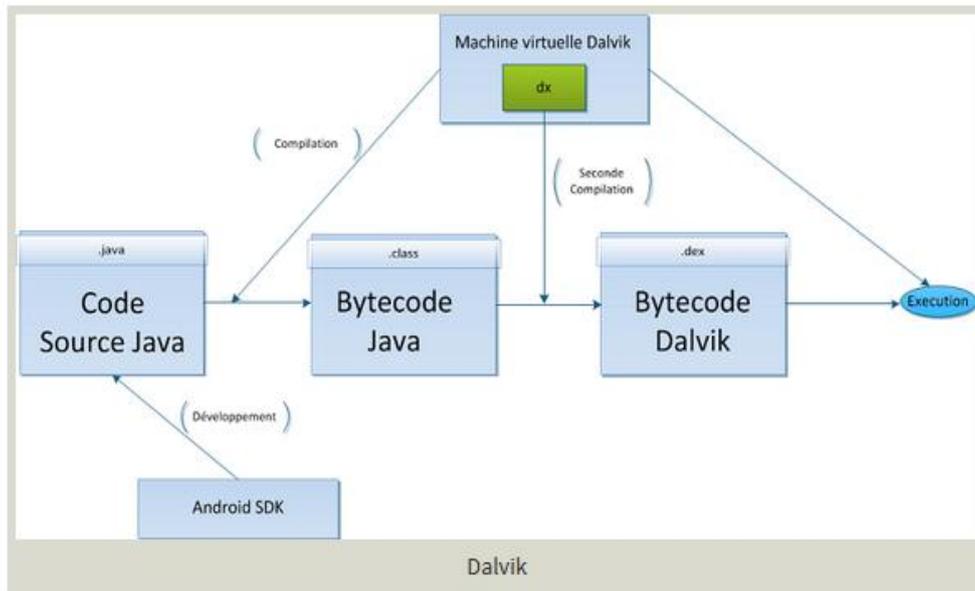


On peut y observer toute une pile de composants qui constituent le système d'exploitation. Le sens de lecture se fait de bas en haut, puisque le composant de plus bas niveau (le plus éloigné des utilisateurs) est le noyau Linux et celui de plus haut niveau (le plus proche des utilisateurs) est constitué par les applications.

Le système d'exploitation d'Android se base sur Linux. Si on veut être plus précis, c'est le noyau (« *kernel* » en anglais) de Linux qui est utilisé. Le noyau est l'élément du système d'exploitation qui permet de faire le pont entre le matériel et le logiciel. Par exemple, les pilotes Wifi permettent de contrôler la puce Wifi. Quand Android veut activer la puce Wifi, on peut imaginer qu'il utilise la fonction « `allumerWifi()` », et c'est au constructeur de spécifier le comportement de « `allumerWifi()` » pour sa puce. On aura donc une fonction unique pour toutes les puces, mais le contenu de la fonction sera unique pour chaque matériel.

La version du noyau utilisée avec Android est une version conçue spécialement pour l'environnement mobile, avec une gestion avancée de la batterie et une gestion particulière de la mémoire. C'est cette couche qui fait en sorte qu'Android soit compatible avec tant de supports différents.

- **Le moteur d'exécution d'Android:** La figure suivante est un schéma qui indique les étapes nécessaires à la compilation et à l'exécution d'un programme



La version de Java qui permet le développement Android est une version réduite amputée de certaines fonctionnalités qui n'ont rien à faire dans un environnement mobile. Par exemple, la bibliothèque graphique Swing n'est pas supportée, on trouve à la place un système beaucoup plus adapté. Mais Android n'utilise pas une machine virtuelle Java ; une machine virtuelle tout étudiée pour les systèmes embarqués a été développée, et elle s'appelle « Dalvik ». Cette machine virtuelle est optimisée pour mieux gérer les ressources physiques du système. Elle permet par exemple de laisser moins d'empreinte mémoire (la quantité de mémoire allouée à une application pendant son exécution) ou d'utiliser moins de batterie qu'une machine virtuelle Java.

On voit bien que le code Java est ensuite converti en bytecode Java. Ce bytecode Java ne peut être lu que par une machine virtuelle Java, mais que Dalvik n'était pas une machine virtuelle Java. Il faut donc procéder à une autre conversion à l'aide d'un programme qui s'appelle « dx » qui s'occupe de traduire les applications de bytecode Java en bytecode Dalvik, qui, lui, est compréhensible par la machine virtuelle.

- **Application Android :**

- **Vue générale :**

Une application Android est un assemblage de composants liés grâce à un fichier de configuration. Dans ce qui suit nous allons découvrir **une vue générale** de ces composants à savoir :

- ✓ Les activités ;
- ✓ Les vues et contrôles ;
- ✓ Les ressources ;
- ✓ Le fichier de configuration appelé également **manifest**.

## Annexe A

---

### ➤ **Activités :**

Une Activité représente un écran de l'application. Une application peut avoir une ou plusieurs activités (par exemple pour une application de messagerie on pourrait avoir une Activité pour la liste des contacts et une autre pour l'éditeur de texte). Chaque Activité est implémentée sous la forme d'une classe qui hérite de la classe.

- Une activité est composée de deux volets :
- la logique de l'activité et la gestion du cycle de vie de l'activité qui sont implémentés en Java dans une classe héritant de `Activity` ;
- l'interface utilisateur, qui pourra être définie soit dans le code de l'activité soit de façon plus générale dans un fichier XML placé dans les ressources de l'application.

### ➤ **Les vues et contrôles :**

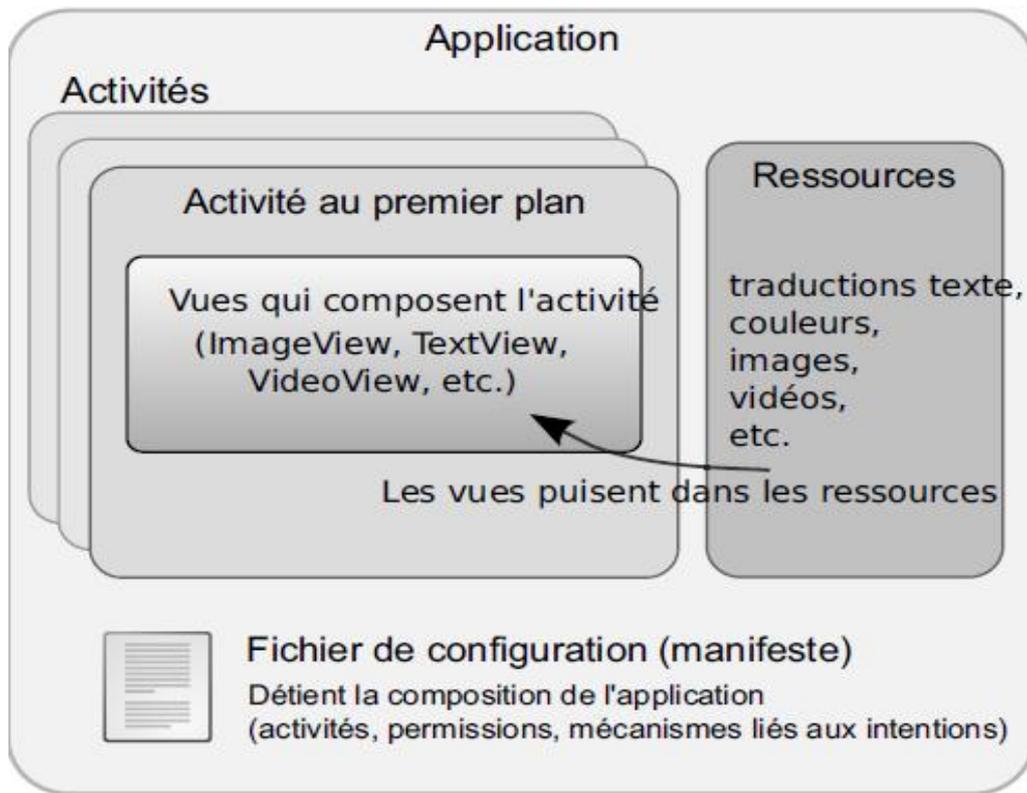
Les vues sont les éléments de l'interface graphique que l'utilisateur voit et sur lesquels il pourra agir. Les vues contiennent des composants, organisés selon diverses mises en page (les uns à la suite des autres, en grille...). Quant aux *contrôles* (boutons, champs de saisie, case à cocher, etc.) sont eux-mêmes un sous-ensemble des vues (nous y reviendrons ultérieurement). Ils ont besoin d'accéder aux textes et aux images qu'ils affichent (par exemple un bouton représentant un téléphone aura besoin de l'image du téléphone correspondante). Ces textes et ces images seront puisés dans les fichiers *ressources* de l'application.

### ➤ **Le fichier de configuration :**

Le fichier de configuration de l'application. C'est un fichier indispensable à chaque application qui décrit entre autres :

- ✓ Le point d'entrée de notre application (quel code doit être exécuté au démarrage de l'application);
- ✓ Quels composants constituent ce programme ;
- ✓ Les permissions nécessaires à l'exécution du programme (accès à Internet, accès à l'appareil photo, accès au wifi pour notre cas.).

Voici une figure qui représente la composition d'une application Android (vue générale):



Composition d'une application

### **Vue détaillée :**

Plus une application est complexe, plus le nombre de pièces utilisées sera grand. Voici donc d'autres pièces principales du « puzzle » Android :

- ✓ Services.
- ✓ Fournisseurs de contenu.
- ✓ Gadgets.
- ✓ Objets Intent .
- ✓ Récepteurs d'Intents.
- ✓ Notifications.

#### ➤ **Services :**

Le *service* est un composant qui fonctionne en tâche de fond, de manière invisible. Ses principales utilisations sont la mise à jour de sources de données ainsi que d'activités visibles et le déclenchement de notifications.

#### ➤ **Le fournisseur de contenu :**

## Annexe A

---

Permet de gérer et de partager des informations. Un même fournisseur permet d'accéder à des données au sein d'une application et entre applications.

➤ **Le gadget :**

Est un composant graphique qui s'installe sur le bureau Android. Le calendrier qui affiche de l'information ou le lecteur audio qui permet de contrôler la lecture de fichiers sont deux exemples de gadgets que l'on trouve souvent sur un écran d'accueil.

➤ **L'objet Intent :**

Il permet de diffuser des messages en demandant la réalisation d'une action. L'accès aux autres applications et au système étant restreinte par le modèle de sécurité Android, ces objets permettent aux applications de fournir ou demander des services ou des données. La transmission se fait à travers tout le système et peut cibler précisément une activité ou un service.

➤ **Récepteur d'Intents :**

Il permet à une application d'être à l'écoute des autres afin de répondre aux objets Intent qui lui sont destinés et qui sont envoyés par d'autres composants applicatifs.

➤ **Notification :**

Une notification signale une information à l'utilisateur sans interrompre ses actions en cours.

• **Notion de Filtres (d'Intent) :**

Un objet **Intent** peut mentionner explicitement un composant cible. Si cette information n'est pas renseignée, Android doit choisir le meilleur composant pour y répondre. Ceci est mené à bien via une comparaison de l'objet **Intent** avec les filtres des différentes applications cibles. Les filtres ne se manipulent généralement pas via l'API mais sont paramétrables grâce à la `<intent-filter>` du fichier de configuration ou bien d'une manière dynamique.

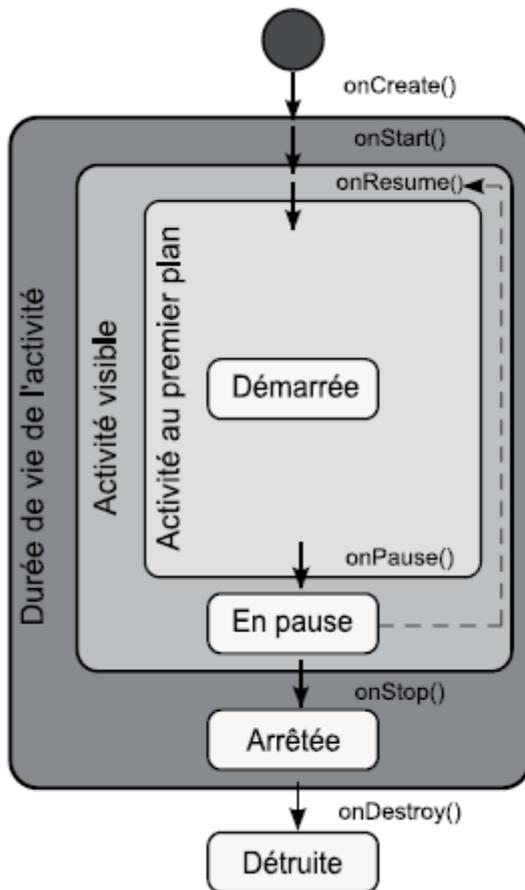
• **Permissions :**

Certaines opérations sont réalisables à condition d'en obtenir la permission. Ces actions sont de plusieurs formes :

- ✓ opérations pouvant entraîner un surcoût (connexion, échange de données, envoi de SMS par exemple) ;
- ✓ utilisation de données personnelles (accès à vos contacts, à votre compte Google, exploitation de vos informations linguistiques entre autres) ;
- ✓ accès au matériel du téléphone (prise de clichés, écriture sur la carte mémoire...) ;
- ✓ accès au service wifi d'Android.

Ces permissions devront être déclarées pour leur utilisation dans le fichier de configuration qui décrit l'application. À l'installation de l'application, l'utilisateur disposera d'un récapitulatif de toutes les permissions demandées pour que l'application fonctionne. Et ainsi il pourra soit continuer, soit interrompre l'installation en connaissance de cause.

## Cycle de vie d'une activité :



Les états principaux d'une activité sont les suivants :

- active (*active*) : activité visible qui détient le focus utilisateur et attend les entrées utilisateur. C'est l'appel à la méthode `onResume`, à la création ou à la reprise après pause qui permet à l'activité d'être dans cet état. Elle est ensuite mise en pause quand une autre activité devient active grâce à la méthode `onPause` ;
- suspendue (*paused*) : activité au moins en partie visible à l'écran mais qui ne détient pas le focus. La méthode `onPause` est invoquée pour entrer dans cet état et les méthodes `onResume` ou `onStop` permettent d'en sortir ;
- arrêtée (*stopped*) : activité non visible. C'est la méthode `onStop` qui conduit à cet état.

## Projet ADT :

Un projet basé sur le plugin ADT est décomposé de la manière suivante:

- `src/`: les sources Java du projet
- `libs/`: bibliothèques tierces
- `res/`:
  - `res/drawable`: ressources images
  - `res/layout`: description des IHMs en XML
  - `res/values`: chaînes de caractères et dimensions

## Annexe A

---

- gen/: les ressources auto générées par ADT
- assets/: ressources brutes (*raw bytes*)
- bin/:
  - bin/classes: les classes compilées en *.class*
  - bin/classes.dex: exécutable pour la JVM Dalvik
  - bin/myapp.zip: les ressources de l'application
  - bin/myapp.apk: application empaquetée avec ses ressources et prête pour le déploiement

### SQLite :

SQLite est un moteur de base de données libre qui implémente la plupart des spécifications du standard SQL92.

La principale différence entre SQLite et PostgreSQL ou MySQL tient au fait qu'une base de données entière avec toutes ses tables est stockée dans un seul et unique fichier. Une autre particularité est que l'accès à ce fichier de données ne nécessite aucun serveur de base de données. A cela on peut rajouter qu'il est rapide (deux fois plus que PostgreSQL et MySQL pour des opérations courantes), peu gourmand en mémoire, et peut gérer des bases de données de grandes dimensions.

Comment accéder à la base de données :

Voici la méthode pour accéder à la base de données sous Android :  
DDMS -> FileExplorer ->data ->data ->nom\_package->databases

## Annexe B

---

Dans cette annexe nous allons exposer le concept Web service et les protocoles qu'il utilise.

### **Définition :**

Au sens large, les services web sont des systèmes logiciels, permettant l'interopérabilité entre plusieurs systèmes logiciels (agents) sur un réseau informatique.

Plus spécifiquement, lorsque nous utilisons comme base les normes du W3C, l'interface du système est définie par un langage lisible par un ordinateur (WSDL). D'autres systèmes logiciels vont communiquer avec le service Web selon sa description en utilisant le langage SOAP, généralement en utilisant XML pour sérialiser les messages et HTTP comme protocole réseau.

Les Web Services rendent disponibles à plus grande échelle les intergiciels (middleware) traditionnels.

### **Le concept des web services :**

Le concept des Web Services s'articule actuellement autour des trois acronymes suivants:

- WSDL : Web Services Definition Language pour la description des Services Web.
- SOAP : Simple Object Access Protocol pour les appels de services à distance par échange de messages XML.
- UDDI : Universal Description, Discovery and Integration, pour le référencement des services.

### **L'intérêt des services web :**

Les Services Web comportent de nombreux avantages, ils sont utilisables à distance via n'importe quel type de plate-forme, ils peuvent servir au développement d'applications distribuées et sont accessibles depuis n'importe quel type de clients. Les services web appartiennent à des applications capables de collaborer entre elles de manière transparente pour l'utilisateur.

### **Architecture des Web Services :**

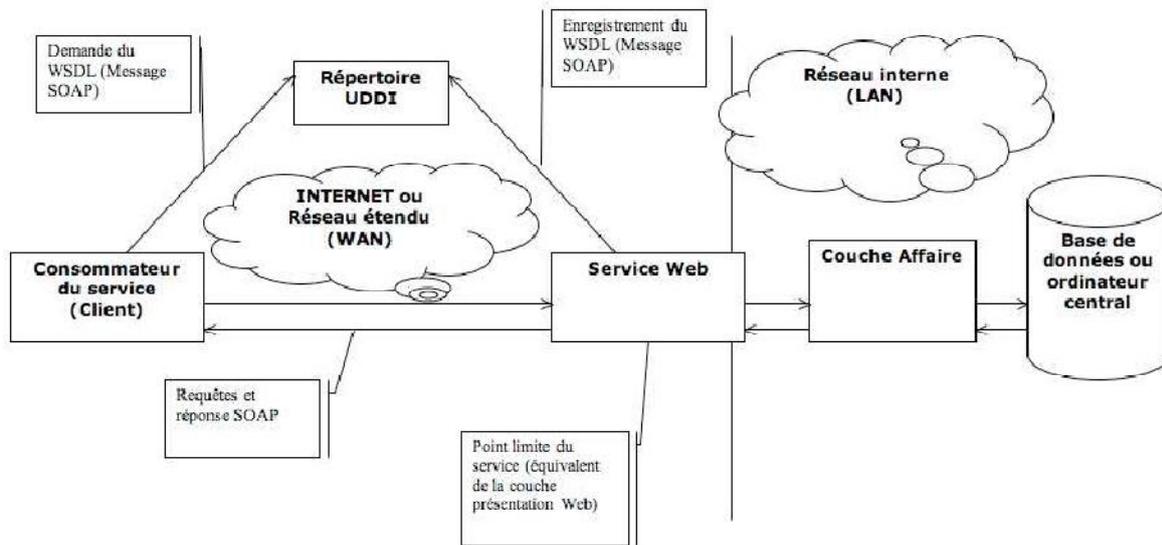
Jusqu'ici, l'accès via Internet à une ressource applicative ou à une base de données s'effectuait par l'envoi d'une requête s'appuyant sur des langages de script (PHP, JSP,...).

Il s'agissait donc d'un dialogue entre une couche de *présentation* reposant sur HTML (protocole http) et des applications installées sur un serveur distant.

Avec les Web Services, un dialogue est désormais instauré entre applications qui peuvent être installées sur des machines distantes, et ceci grâce à des standards XML.

En effet, afin de dialoguer via Internet, ces applications doivent « parler » le même langage, langage basé sur le XML. Comme illustrer dans la figure ci-dessous.

## Annexe B



L'architecture de référence des services web se base sur les trois concepts suivants :

- **Le fournisseur de service** : c'est le propriétaire du service.
- **Le client (ou le consommateur de service)** : c'est un demandeur de service. D'un point de vue technique, il est constitué par l'application qui va rechercher et invoquer un service.
- **L'annuaire des services** : c'est un registre de descriptions de services offrant des facilités de publication de services à l'intention des fournisseurs ainsi que des facilités de recherche de services à l'intention des clients.

Les interactions de base qui existent entre ces trois éléments sont les opérations de *publication*, de *recherche*, d'*invocation* et de *lien* (bind).

### Les applications des Services web :

L'application des services web est multiple, autant dans les domaines du **B2C**, **B2B** que pour des domaines de gestion, par exemple gestion de stock, gestion commerciale, etc...

- **B2C (Business to Consumer)** : Qualifie une application, un site Internet destiné au grand public.
- **B2B (Business to Business)** : Qualifie une application, un site Internet destiné au commerce de professionnel à professionnel.

## Protocoles :

Les Services Web se basent actuellement sur trois protocoles : **SOAP**, **WSDL**, **UDDI**, dans ce qui suit, nous allons les détailler.

### **SOAP (Simple Object Access Protocol) :**

Au sein des protocoles liés aux Services Web, SOAP fait figure de pièce centrale. En effet, SOAP définit un protocole simple destiné à l'échange d'informations.

Son objectif est de permettre la normalisation des échanges de données au sein d'architectures distribuées orientées objet.

SOAP constitue un standard simple et neutre puisqu'il :

- N'impose pas l'utilisation d'une API particulière.
- N'impose aucun modèle de programmation.

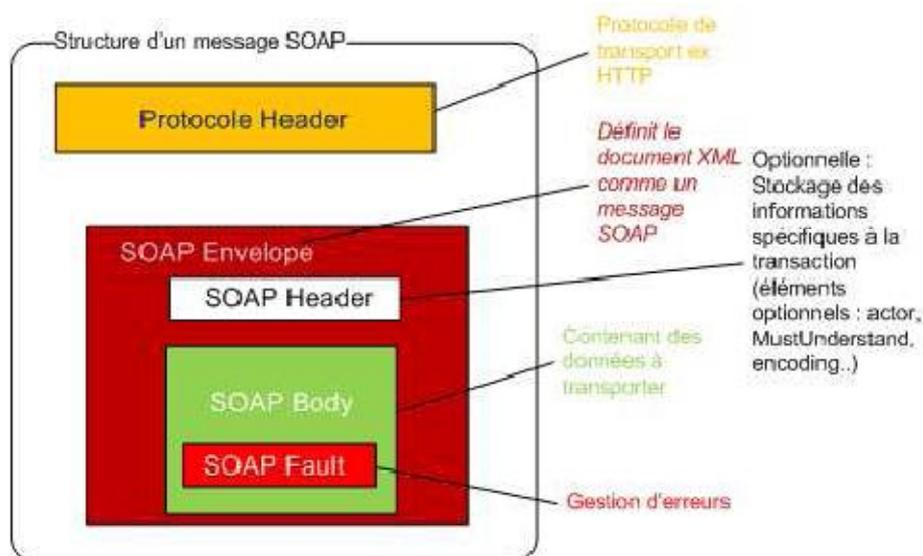
De plus, SOAP est fondé sur un standard : le langage XML.

### **Structure d'un message SOAP :**

Tout d'abord un message SOAP est un document XML qui doit avoir la forme suivante :

- La structure de l'enveloppe SOAP qui définit une structure générale décrivant le contenu, le destinataire, et la nature du message.
- Les règles d'encodage SOAP qui définissent le mécanisme de sérialisation utilisé pour échanger des objets.
- SOAP RPC qui définit, pour les utilisations synchrones, une convention de représentation des appels et des réponses des procédures distantes.

Un message SOAP est composé des parties suivantes :



### **WSDL (Web Service Description Language) :**

## Annexe B

---

Le WSDL est un langage dérivé d'XML permettant de fournir les spécifications nécessaires à l'utilisation d'un Service web en décrivant les méthodes, les paramètres et les types de retour. Le WSDL est aussi l'équivalent de IDL (Interface Definition Language) pour la programmation distribuée (CORBA). Ce langage permet de décrire de façon précise les services Web, en incluant des détails tels que les protocoles, les serveurs, les ports utilisés, les opérations pouvant être effectuées, et les formats des messages d'entrée et de sortie.

### **UDDI (Universal Description Discovery and Integration):**

UDDI est une spécification définissant la manière de publier et de découvrir les Services Web sur un réseau. Ainsi, lorsque l'on veut mettre à disposition un nouveau service, on crée un fichier appelé **Business Registry** qui décrit le service en utilisant un langage dérivé d'XML suivant les spécifications UDDI.

Les informations qu'il contient peuvent être séparées en trois types :

- **Les pages blanches** incluant l'adresse, le contact et les identifiants relatifs aux Services Web.
- **Les pages jaunes** identifiant les secteurs d'affaires relatifs aux Services Web.
- **Les pages vertes** donnant les informations techniques.

En utilisant l'API **UDDI** l'utilisateur peut alors stocker ces informations dans un nœud (node) UDDI. Elles sont ensuite répliquées de nœud en nœud (principe assez proche dans l'idée des répliqués de DNS).

Une fois ceci est fait, le Service Web peut alors être connu de tous ceux qui le recherchent.

### **URI :**

Un URI ( Uniform Resource Identifier) est la façon d'identifier un point de contenu sur le web, que ce soit un fichier texte, audio, vidéo. L'URI la plus connue est l'adresse d'une page web par exemple : [http:// : www.developer.com](http://www.developer.com) . cette adresse URL (Uniform Resource Locator) est l'une des sous catégorie d'URI.

---

# *Bibliographie*

## Bibliographie

- [1] **Stephane DURAND**, 2003. « Amélioration de la précision de la localisation différentielle temps réel par mesure de phase des systèmes GNSS: étude détaillée des équations d'observation et du problème de résolution des ambiguïtés entière », l'observatoire de Paris, 2003.
- [2] **GNING El Hadji Amadou** « Localisation garantie d'automobiles. Contribution aux techniques de satisfaction de contraintes sur les intervalles » Spécialité : Technologies de l'information et des systèmes, université de technologies Compiègne.
- [3] **Borhane Eddine Boulhila**, 2014. « Conception et mise en place d'un système FleetTracking » Ingénieur : Concepteur Système et Réseaux à l'école supérieure Des Ingénieurs Limoges-France ETM Ibnrochd Alger-Algérie.
- [4] **Maher Ben Jemaa, Fatma Bouzid et Saber Ghribi**, article : Conception d'un Logiciel de Localisation d'un mobile. Ecole Nationale d'Ingénieurs de Sfax, Tunisie.
- [5] Article N°31 SUPPLY CHAIN MAGAZINE, 2009, informatique embarquée, garder la maîtrise de sa flotte.
- [6] **K.Bouaphanh, F.Harmand**, article : enjeux, possibilités et dérives de la géolocalisation.
- [7] « Implémentation d'une solution de géolocalisation des véhicules pour le module open TMS d'open ERP, à l'aide d'open GTS. », Rapport.
- [8] **Jinane SAYAH**, Contribution à la modélisation, à la simulation et à l'évaluation d'applications nomades à intelligence répartie – Application à l'assistance aux voyageurs aveugles dans les transports publics et les pôles d'échanges, Le 18 décembre 2009
- [9] **A.Antonin, B.Mihai-Bogdan, B.Sanaa, B.Pierrick, B. Stephanie**, article : système de géolocalisation ; Culture Générale des Télécommunications ; Janvier 2010
- [10] **Benjamin Savouré**, La téléphonie mobile : technologies, acteurs et usages, mémoire Master II recherche « droit des médias », 2005.
- [11] **Patrick DESSALLE**, 2005. Conception et réalisation d'une plateforme de déploiement de services géolocalisés, Mémoire de fin d'études pour l'obtention du grade d'Ingénieur Civil Informaticien. Université Libre de Bruxelles.
- [12] **Mokhtari ACHRAF**, 2012. « Conception et implémentation d'une application sous système Android pour la télégestion de flotte de véhicules ». Projet de fin d'étude en Télécom, Ecole Marocaine Des Sciences de l'Ingénieur.
- [13] **Siaka Diakite**, 2010, « Webmapping, gestion de flotte : développement d'une application de gestion spatiale de flotte et maintenance des mobiles véhicules », IASIG.
- [14] « guide de la géolocalisation des salariés, droits et obligations en matières de géolocalisation des employés par dispositif de suivi GSM/GPRS », Rapport.

[15] **Grady Booch, James Rambough et Ivar Jacobso** « Le guide de l'utilisateur UML », Edition Eyrolles, 2001.

[16] : Pascal Roques, « **UML par la pratique** », Edition EYROLLES 2eme édition 2004

[17] : UML 2 en action, Edition EYROLLES ,2007.

[18]ANNICK FRON ; « **Architecture réparties en JAVA, RMI, COBRA, JMS, Sockets, SOAP, services web** ».

[19] : Saleh, « **les langages de développement web** », Edition HERMES SCIENCE, 1996.

[20] : Jean Michel Doudoux, « **Développons en Java** », Version 0.90, 2006.

[21] : Pierre-Yves Gibello, « **java 2 entreprise édition** », Edition ARMAND COLIN 2004.

[22] « **programmation Android, de la conception au déploiement avec le SDK Google Android 2** », Edition EYROLLES 2009.

[23] Paul-Andrés, « **développement Web et JEE** » Master Miage Nantes, université de Nantes.

[24] « les web services : définitions, technologies, acteurs, impacts sur les entreprises et problèmes » Rapport.

[w1]: <http://www.wikipédia.com/géolocalisation>

[w2] :<http://www.wikipédia.com/>

[w3] <http://www.eclipse.org/downloads/>

[w4]: <http://developer.android.com/index.html>.

[w5]: <https://dl-ssl.google.com/android/eclipse/>.

[w6]: <http://www.mti.epita.fr/blogs/2010/08/03/introduction-a-la-programmation-sous-android/>.

[w7]: [www.javafr.com](http://www.javafr.com).

[w8]: <http://java.sun.com>

[w9]: <http://commentçamarche.com>

[w10]: <http://sqlite.com>

[w11]: <http://www.netbeans.org/download/>.

# Résumé

---

## **Résumé :**

Le présent travail s'inscrit dans le cadre du projet de fin d'étude au département d'informatique de l'université mouloud Mammeri, en vue d'obtention du diplôme de Master en informatique. Notre travail consiste à mettre en place une application répartie pour l'alerte, la localisation et le suivi des véhicules en tournées.

Notre mission globale se divise en deux volets. Le premier volet consiste à mener une étude approfondie de l'aspect métier du projet, ainsi que des technologies qui permettront de le réaliser. Le deuxième volet est la réalisation concrète du projet. Les travaux menés dans ce volet consistent en la conception et l'implémentation de deux applications web et mobile, sur une plateforme répartie client/serveur, déployés respectivement sur un poste de contrôle et sur un terminal mobile.

Le projet fait appel à diverses technologies et répond amplement aux besoins exprimés par les entreprises voulant optimiser et rentabiliser l'emploi de leurs flottes de véhicules.

## **Mot clés :**

Suivi des véhicules, tracking, alerte, géolocalisation, services géolocalisés, GPS, GPRS, GSM plateforme logicielle de géolocalisation, système de gestion d'une flotte des véhicules.