

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULoud MAMMERI DE TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT INFORMATIQUE



Mémoire fin d'études

EN VUE DE L'OBTENTION DU DIPLOME DE MASTER EN INFORMATIQUE

Option : Réseaux, Mobilité et Système embarqué.

Thème

**Conception et réalisation d'une application de
suivi de véhicule par géolocalisation.**

Dirigé par :

Mr HEMDANI Chabane

Réalisé par :

M^{elle} AMARA Wissam

M^{elle} TAMERT Celia

Promoon 2018/2019

Remerciements

Au terme de la rédaction de ce mémoire, nous estimons qu'il est important d'accorder quelques lignes de reconnaissances à toute personne ayant contribué de prêt ou de loin.

Tout d'abord, nous adressons notre plus profonde gratitude à notre promoteur Mr HEMDANI Chabane, qui a toujours su orienté nos recherches, et pour tout le temps qu'il nous a accordé.

Nous tenons à remercier également les membres de jury d'avoir accepté d'évaluer notre travail.

Nous tenons à exprimer notre sincère reconnaissance envers monsieur AFRIT Yacine pour sa précieuse aide. Nos remerciements vont également à nos amis(es) qui nous ont toujours encouragées que ce soit d'un point de vue moral ou intellectuel. et plus particulièrement à monsieur M'heni MEBARKI.

Dédicaces

A nos chers parents,
A nos frères et sœurs,
A nos familles,
A nos amis(es).

Résumé

Notre travail repose sur la conception d'un système électronique embarqué, destiné à la localisation à distance basée sur les réseaux GSM (Global System for Mobile communication), GPRS (General Packet Radio Service) et GPS (Global Positioning System). Nous avons également conçu une plateforme web temps réel qui est lié à ce système embarqué, il s'agit d'une application de tracking. Durant ce projet on a essayé d'un coté de garantir une grande « autonomie » de fonctionnement du système et d'autre coté de garantir à l'utilisateur final, un contrôle total sur toute la chaîne : surveillance –localisation de sa voiture.

Mots clés: géolocalisation, GSM, GPS, GPRS, antivol, véhicules.

Abstract

This project presents an electronic embedded system for localization based on GSM (Global System for Mobile communication), GPRS (General Packet Radio Service) and GPS (Global Positioning System) networks. We also designed a real time web platform that is related to this embedded system, it is a tracking application. During this project we tried to guarantee a great “autonomy” of system operation, the procedure used in our system provides the owner with a total supervision and localization of his vehicle.

KEYWORDS: localization, GSM, GPS, GPRS, anti-theft device, cars.

Table des matières

Remerciements	i
Dédicaces	ii
Résumé	iii
Abstract	iv
Table des matières	v
Table des figures	ix
Introduction	1
1 Généralités sur les systèmes embarqués :	3
1.1 Introduction :	4
1.2 Définition :	5
1.3 Historique des systèmes embarqués :	5
1.4 La différence entre les systèmes embarqués et notre PC:	7
1.5 Caractéristiques d'un système embarqué :	7
1.6 ARCHITECTURE D'UN SYSTÈME EMBARQUÉ :	8
1.7 les composants matériels essentiels d'un système embarqué:	10
1.7.1 Les mémoires :	10
1.7.2 Les périphériques d'entrée/sortie :	11
1.7.3 Les processeurs :	12
1.7.4 Les systèmes sur puce :	13
1.8 les systèmes d'exploitation pour embarqué	14
1.9 les langages de programmation utilisés pour le développement d'application embarquée :	15
1.10 Exemples d'application des systèmes embarqués:	16
1.11 Conclusion	17
2 Géolocalisation GPS/GSM/GPRS :	18
2.1 Introduction :	19
2.2 La géolocalisation :	20

2.2.1	Définition :	20
2.2.2	Techniques de géolocalisation :	20
2.2.2.1	Géolocalisation par satellite :	20
2.2.2.2	Géolocalisation par GSM :[7]	21
2.2.2.3	Géolocalisation par WIFI :	23
2.2.2.4	Géolocalisation par adresse IP(sur internet):	23
2.2.2.5	Géolocalisation par RFID :	25
2.2.2.6	Combinaison de ces techniques :	25
2.2.2.7	Géolocalisation indoor :	25
2.3	Plateformes logiciels de géolocalisation :	26
2.3.1	Composants :	26
2.3.2	Architecture temps réel :	26
2.4	Principes du GPS :	28
2.4.1	Définition :	28
2.4.1	Historique :	29
2.4.2	Différents segments du GPS:	29
2.4.3	Principe de fonctionnement:	31
2.4.3.1	Principes de triangulation :	31
2.4.3.2	La mesure de distance :	32
2.4.3.3	Le positionnement des satellites :	34
2.5	Le protocole NMEA appliqué aux GPS :	34
2.6	Les moyens de communication des coordonnées géographiques :	35
2.6.1	GSM :	35
2.6.1.1	Le concept cellulaire du GSM :	36
2.6.1.2	Architecture d'un réseau GSM :	36
2.6.1.3	Le service de messages courts de GSM (SMS) :	36
2.7	La géolocalisation GPS/GSM/GPRS :	37
2.7.1	Géolocalisation GPS/GSM/GPRS pour les systèmes de suivi de véhicules :	37
2.8	Conclusion :	38
3	Analyse et Conception du système :	39
3.1	Introduction :	40
3.2	Méthodologie de conception :	41
3.3	Analyse :	42
3.3.1	Le but de notre travail :	42

3.3.2	Spécification des besoins fonctionnels :	44
3.3.3	Spécification des besoins matériels :	44
3.4	Conception	45
3.4.1	Conception matérielle :	45
3.4.1.1	Descriptions des Composants matériels utilisés:	46
3.4.1.1.1	Le module GPS : [16]	46
3.4.1.1.2	Le module GSM SIM900 [17]:	48
3.4.1.1.3	La carte arduino :	48
3.4.2	Conception Logicielle :	49
3.4.2.1	Comportement du système logiciel du Système embarqué dans le véhicule :	50
3.4.2.1.1	Les diagrammes de séquence :	51
3.4.2.2	Comportement du système logiciel de la plateforme du suivi de véhicule :	55
3.4.2.2.1	Diagramme de séquence :	55
3.4.2.2.2	L'architecture web :	57
3.4.2.2.3	L'architecture de communication :	57
3.4.2.2.4	Le niveau donné :	58
3.4.2.3	Algorithme de quelques procédures :	59
3.5	Conclusion :	62
4	Réalisation du système:	63
4.1	Introduction :	64
4.2	Outils de développement utilisés :	65
4.2.1	Outils logiciels :	65
4.2.1.1	Logiciel arduino:	65
4.2.1.2	Framework Laravel :	66
4.2.1.3	Template bracket responsive bootstrap :	66
4.2.1.4	Serveur XAMPP:	67
4.2.1.5	Sublime_text: [20].	67
4.2.1.6	Leaflet:	68
4.2.1.7	Ngrok:	68
4.2.2	Les Langages de programmation :	68
4.2.3	Outils matériels :	69
4.2.4	Implémentation des fonctionnalités et technologies utilisées :	69
4.2.4.1	La partie web :	69
4.2.4.2	Le serveur de communication :	71

❖	Avantage des WebSockets :.....	71
4.3	Réalisation et Tests:	73
4.3.1	Réalisation du système embarqué :.....	73
4.3.1.1	Explication du circuit de notre système :.....	73
4.3.1.2	La programmation :	74
4.3.2	Tests:	75
4.3.2.1	Tests unitaires :	75
4.3.2.1.1	Test du module GPS :	75
4.3.2.1.2	Test du module GSM :	76
4.3.2.1.3	Test de la partie GPRS:.....	76
4.3.2.2	Test de notre système :	77
4.3.2.2.1	Demande de la position du véhicule :	77
4.3.2.2.2	Demande de modification du mot de passe :	79
4.3.3	Quelques Interface de la plateforme web de géolocalisation :	79
4.4	Conclusion:.....	83

Conclusion et perspectives.....	84
--	-----------

Bibliographie

Annexes

Liste des figures

Figure 1.1 : Système de régulation de quantité de vapeur	5
Figure1.2 : Autonetics D-17 guidance computer	6
Figure 1.3 : Architecture typique d'un système embarqué	8
Figure 1.4 : Meilleurs langages pour le développement de systèmes embarqués	15
Figure 2.1 : Repérage d'une position par satellites	21
Figure 2.2 : Principe d'EOTD	22
Figure 2.3 : Principe de la triangulation	22
Figure 1.4 : Principe du Cell ID	23
Figure 2.5 : Technique de localisation par adressage IP	24
Figure 2.6 : Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau satellitaire	27
Figure 2.7 : Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau GSM/GPRS	28
Figure 2.8 : Segment spatial	29
Figure 2.9 : Segment de contrôle	30
Figure 2.10 : Emplacement de segments de Contrôles dans le monde.....	30
Figure 2.11 : Segment utilisateur	31
Figure 2. 12 :La constellation GPS	31
Figure 2.13 :Un satellite GPS	31
Figure.2.14 : principe de positionnement par GPS	32
Figure. 3.1 : Schéma de la méthodologie de conception.....	41
Figure. 3.2 : Fonctionnement général du système de géolocalisation	42
Figure. 3.3 : Fonctionnement du système de géolocalisation GSM/GPS.....	43
Figure. 3.4 : Fonctionnement du système de géolocalisation GPRS/GPS.....	43

Figure.3.5 : Composants matériels utilisés	46
Figure. 3.6: Le module GPS NEO-6MV2	46
Figure. 3.7: Format NMEA.....	47
Figure. 3.8: Connexion entre le module GPS et la carte ARDUINO	47
Figure 3.9: Carte SIM 900 GSM/GPRS shield.....	48
Figure 3.10 : Description des entrées/sorties de la carte Arduino Uno	49
Figure 3.11 : Zoom sur la carte Arduino Uno	49
Figure 3.12 : organigramme global du système.....	50
Figure 3.13 : diagramme des cas d'utilisation du système	51
Figure.3.14 : Diagramme de séquence pour le cas « Démarrage du système	52
Figure.3.15 : Diagramme de séquence pour le cas « Demande de la position du véhicule »	53
Figure.3.16 : Diagramme de séquence pour le cas «changement de mot de passe.»	54
Figure 3.17 : Diagramme général des cas d'utilisation	55
Figure 3.18 : Diagramme de séquence pour la géolocalisation temps réel et affichage du statut d'un véhicule	56
Figure 3.19 : Diagramme de séquence pour consulter l'historique de localisation	57
Figure. 3.20: L'architecture de communication	58
Figure 3.21: Le schéma de la base de données.....	59
Figure 4.1: logiciel de programmation de la carte ARDUINO	65
Figure 4.2 : Vue d'ensemble du logiciel Arduino	66
Figure 4.3 : L'interface de template bracket.....	66
Figure 4.4 : L'interface de xampp	67
Figure 4.5: L'interface de sublime text.....	67
Figure 4.6 : diagramme d'architecture du laravel.....	70

Figure 4.7 : l'ordre d'exécution	72
Figure 4.8 : Connexion entre le module GPS et la carte ARDUINO	73
Figure 4.9 : Connexion entre le module GSM et la carte ARDUINO	73
Figure 4.10 : le circuit final de notre système.....	74
Figure 4.11 : Résultats du test de GPS	75
Figure. 4.12 : Envoi d'un SMS vers le module SIM908.....	76
Figure. 4.13 : Résultats du test de GPRS	76
Figure IV.14 : branchement de notre système de géolocalisation	77
Figure 4.15 : Réception de la position.....	78
Figure 4.16 : Visualisation de la position du véhicule	78
Figure 4.17 : réception de la confirmation du changement du mot de passe	79
Figure 4.18 : interface de sélection du véhicule	80
Figure 4.19 :interface de visualisation de toutes les positions.....	80

Introduction

Introduction

Cadre général et objectifs

Les systèmes de géolocalisation sont couramment utilisés par les entreprises disposant d'un parc automobile important, dans le but d'optimiser les dépenses liées au transport. Ils sont également utilisés par les particuliers, pour les fonctions de sécurité de biens et de personnes, de suivi en temps réel et de contrôle à distance de leurs véhicules. Ces systèmes sont largement utilisés dans le monde entier et se basent principalement sur la technologie GPS et GSM.



Dans ce sens il est donc important de mettre sur pied un nouveau système de géolocalisation capable de répondre aux attentes actuelles des entreprises et des personnels tout en alliant performance et fiabilité.

Contribution

Notre travail consiste à développer un système de suivi de véhicule par géolocalisation, Ce système est composé de deux sous systèmes : un localisateur de véhicule utilisant un récepteur gps pour la localisation et un module GSM/GPRS pour la transmission de la position, et une plateforme web qui doit être capable d'effectuer le suivi en temps réel sur une carte géographique, conserver l'historique de mouvements du véhicule et déterminer la vitesse de ce dernier.

Organisation de la thèse

Notre travail est réparti sur quatre chapitres :

Chapitre 1 : portera sur des généralités sur les systèmes embarqués.

Chapitre 2 : expliquera le principe de la géolocalisation GPS/GSM/GPRS.

Chapitre 3 : présentera les phases d'analyse et de conception de notre système, que ce soit la partie logicielle ou la partie matérielle.

Chapitre 4 : permettra de présenter les outils de développement utilisés, la réalisation et les résultats des tests du système.

Nous terminons notre mémoire par une conclusion générale.

Chapitre 1

Généralités sur les systèmes embarqués

1.1 Introduction :

Les systèmes embarqués sont de plus en plus nombreux et de plus en plus complexes, le rôle des systèmes embarqués est primordial au quotidien.

Un système embarqué se compose d'une partie " matériel " (hard) et d'une partie " logiciel " (soft). Ces deux parties communiquent en permanence en utilisant des composants actifs (qui dialoguent en temps réel avec une base qui alerte l'utilisateur en cas de problème) et passifs (qui attendent que l'utilisateur les interroge pour obtenir une information de leur part).

L'objet de ce chapitre est de présenter les généralités sur les systèmes embarqués. D'abord, nous définissons ces systèmes, nous énumérons ensuite leurs caractéristiques, mais aussi leur Architecture, leur composants matériels essentiels, Systèmes d'exploitation pour système embarqué et les langages de programmation pour le développement d'application embarqué et enfin citer quelque exemple d'application.

1.2 Définition :

En anglais « Embedded system » Les systèmes embarqués, sont des systèmes électroniques et informatiques autonomes, construits pour effectuer des tâches précises. Cette autonomie nécessite qu'ils soient capables d'interagir avec leur environnement et de gérer leurs ressources disponibles (puissance de traitement, capacité de stockage, énergie, etc.). On trouve des systèmes embarqués partout. Au-delà du militaire, de la production de l'énergie, du spatial, de la productique, de tels systèmes sont aussi largement déployés dans les transports : aérien, routier, ferroviaire, naval.

1.3 Historique des systèmes embarqués [1] :

La plupart des machines qui nous simplifient la vie ont besoin d'un système de régulation ou de contrôle pour fonctionner de manière correcte. Ces systèmes de contrôle existent depuis bien avant l'invention des ordinateurs.

Exemple :

Pour maintenir une locomotive à vapeur à une vitesse constante, on a besoin d'un système qui régule la quantité de vapeur envoyée dans les pistons (Figure 1.1).

Si la locomotive ralentit (pente montante...) il faut injecter plus de vapeur, si la locomotive accélère, il faut injecter moins de vapeur. On aimerait que cette tâche se fasse de façon «automatique», c'est-à-dire avec un minimum d'intervention de l'être humain.



Figure 1.1 : *Système de régulation de quantité de vapeur*

Un des premiers exemples de système embarqué date du début des années 1960. Il s'agit de l'ordinateur de bord des vaisseaux spatiaux du programme Apollo, qui a amené N. Armstrong sur la lune. Cet ordinateur contrôlait en temps réel les paramètres de vol et adaptait la trajectoire. Il fonctionnait en mode interactif.

Un autre premier système embarqué est le « Autonetics D-17 guidance computer » de la figure 1.2 qui servait de système de contrôle aux missiles nucléaires américains LGM-30 Minuteman, développé à partir de 1962, il était basé sur des transistors et contenait un disque dur comme mémoire principale.

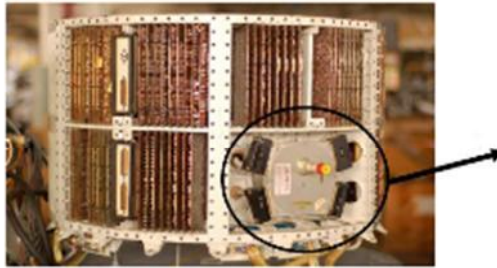


Figure 1.2 : Autonetics D-17 guidance computer

En 1971, Intel produit le premier microprocesseur au monde. Cette puce, le 4004 (4 bits, composé de 2300 transistors), a été conçue suite à une demande de la compagnie japonaise Busicom. Premier circuit générique, personnalisable par logiciel. Le microprocesseur a été un succès immédiat, et son utilisation a augmenté régulièrement au cours de la prochaine décennie.

Une multitude de systèmes embarqués a vu le jour depuis, nous citerons quelques-uns ci-après :

- ✓ 1972 : lancement de l'Intel 8008, premier microprocesseur 8 bits (48 instructions, 800kHz).
- ✓ 1974 : lancement du 8080, premier microprocesseur largement diffusé. 8 bits, (64KB d'espace adressable, 2MHz - 3MHz).
- ✓ 1978 : création du Z 80, processeur 8 bits.
- ✓ 1979 : création du 68000, processeur 16/32 bits.

À partir des années 1990 et suivantes, les principaux pilotes dans les systèmes embarqués ont été les soins de santé pour les appareils de diagnostic intégrés et les systèmes de divertissement, de passer vers les téléphones portables et polyvalents ordinateurs de poche. Un aspect clé qui a changé, c'est que les processeurs 32 bits matures et plus faciles à utiliser des outils de développement ont augmenté la puissance de calcul disponible énormément, alors que bon nombre des contraintes électriques sont tombés à la fabrication du CPU

moderne. Les systèmes embarqués font partie intégrante du système de jeu Wii et le contrôleur Kinect pour la Xbox 360.

Depuis les systèmes se sont diversifiés, ils ont permis l'explosion du marché des «*consumer electronics*» où tout est (devenu) numérique (GSM, électroménager, MP3s, etc.). Ils sont également bien présents dans le domaine industriel pour Contrôle de processus de production, etc.

1.4 La différence entre les systèmes embarqués et notre PC:

Les systèmes embarqués sont dédiés à des tâches spécifiques, tandis que les PC ont des plateformes génériques. Ces systèmes ont de loin moins de ressources que les PC, et doivent souvent fonctionner dans des conditions environnementales extrêmes. Par exemple le cas des capteurs embarqués à bord des satellites. Les systèmes embarqués ont souvent des contraintes d'énergie et des contraintes temps-réel. En effet, les événements temps-réel sont des événements externes au système et ils doivent être traités au moment où ils se produisent (en temps-réel). C'est pourquoi, si un système embarqué utilise un système d'exploitation, il doit probablement utiliser un système d'exploitation temps-réel RTOS¹, plutôt que windows, unix, etc.

1.5 Caractéristiques d'un système embarqué [3] :

Les points suivants permettent de caractériser un logiciel embarqué :

Ciblé : Le domaine d'action d'un système embarqué est limité aux fonctions pour lesquelles il a été créé.

Autonome : Une fois enfouis dans l'application, les systèmes embarqués ne sont (le plus souvent) plus accessibles.

Maintenable dans le temps : Dans la majorité des domaines industriels, et en dehors du logiciel classique, la durée de vie des produits est longue de par des obligations légales ou du moins commerciales qui obligent l'industriel à maintenir le produit pendant une dizaine d'années, cas de l'industrie automobile. Dans le cas d'industries plus sensibles comme l'aéronautique, le militaire ou le spatial, cette durée peut être doublée. Il est donc indispensable que le logiciel embarqué soit maintenable durant toute la durée de vie du produit en cas de découverte de problème de fonctionnement majeur.

¹- RTOS : Real Time Operating System.

Spécifique : L'interface de dialogue avec l'utilisateur est spécifique : dans la majorité des cas, un tel logiciel n'utilise pas les interfaces classiques clavier/souris propres à la micro-informatique. S'ils existent, les périphériques d'affichage sont souvent limités à des panneaux de petite taille de type LCD², et les périphériques d'entrée à quelques boutons poussoirs ou autres composants inhabituels dans l'informatique traditionnelle.

Fiable et sécurisé : le logiciel nécessite une grande fiabilité car il est destiné à un fonctionnement complètement autonome.

Optimisé : le plus souvent, mais ce n'est pas obligatoire, ce logiciel est de petite taille si on le compare aux volumes démesurés atteints par les logiciels classiques multi-usages comme en bureautique. Parmi les raisons d'optimisation :

La nécessité de fiabilité citée précédemment cohabite mal avec un volume démesuré : plus on écrit de lignes de codes, plus on a de chances que celles-ci contiennent des *bogues*.

Et aussi ressources limitées.

Fonctionnement en temps réel : les applications embarquées doivent répondre rapidement aux événements internes ou externes et le résultat peut être néfaste si le système ne réagit pas à l'immédiat à un événement du système.

Environnement : un système embarqué est soumis à des nombreuses contraintes d'environnement et il doit s'adapter et fonctionner avec eux par exemple (La température, L'humidité, Les vibrations...).

Faibles ressources : les systèmes embarqués sont généralement limités par leurs ressources spatiales. Étant utilisés comme composants enfouis dans des systèmes plus complexes et étant conçus pour réaliser des tâches spécifiques, des mémoires de petite taille et des unités de calcul d'une faible puissance sont souvent suffisantes.

² LCD (Liquid Crystal Display) : écran à cristaux liquides.

1.5 ARCHITECTURE D'UN SYSTÈME EMBARQUÉ :

L'architecture d'un système embarqué est illustrée à la figure 1.3:

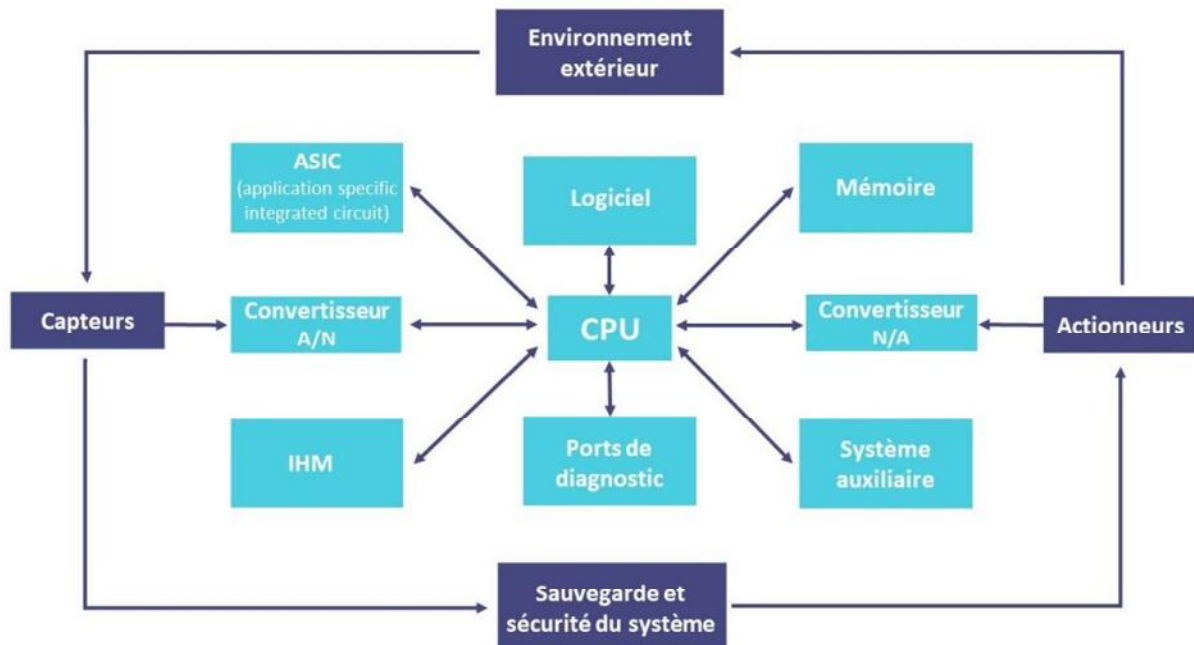


Figure 1.3 : Architecture typique d'un système embarqué

On trouve dans cette architecture:

En entrée des capteurs généralement analogiques (interrupteurs, antennes, etc.) couplés à des convertisseurs analogique/numérique.

En sortie des actionneurs généralement analogiques (LED, etc.) couplés à des convertisseurs analogique/numérique.

Au milieu, on trouve CPU³ (processeur embarqué et ses E/S), Un/des FPGAs⁴ et/ou ASICs⁵ pour jouer le rôle de coprocesseurs. (Accélération matérielle).

Ports de diagnostic utilisés pour la configuration du système et le débogage (USART⁶, JTAG⁷, etc.)

³ CPU : Central Processing Unit.

⁴ FPGA (Field-Programmable Gate Array) : Circuit intégré composé de nombreuses cellules logiques élémentaires reprogrammables après fabrication.

⁵ ASICs (Application-Specific Integrated Circuit) : Circuit intégré configure pour réaliser une tâche spécifique

⁶ USART (Universal Synchronous/Asynchronous Receiver Transmitter) : interface de communication série en mode synchrone ou asynchrone.

⁷ JTAG (Joint Test Action Group) : utilisé au lieu du terme générique Boundary Scan pour désigner le port de teste des cartes électroniques.

Interfaces Homme-Machine (écran LCD, etc.) permettant à l'utilisateur d'interagir avec le système.

Des systèmes auxiliaires qui peuvent compléter le système existant afin de réaliser plus de fonctionnalités.

Sur ce schéma théorique se greffe un paramètre important: le rôle de l'environnement extérieur. Contrairement au PC ronronnant bien au chaud dans un bureau, un système embarqué doit faire face à des environnements plus hostiles. Il doit faire face à un ensemble de paramètres agressifs (Variations de la température, Vibrations, chocs, Eau, feu, radiations...).

Cette architecture peut varier selon les systèmes: on peut par exemple, ne pas trouver de systèmes auxiliaires dans de nombreux systèmes embarqués autonome et indépendants. En revanche, l'architecture de base est la plupart du temps composée d'une unité centrale de traitement (CPU), d'un système d'exploitation qui réside parfois uniquement en un logiciel spécifique (ex: routeur). De même l'interface IHM n'est pas toujours existante, mais est souvent utile pour reconfigurer le système ou vérifier son comportement.

1.6 les composants matériels essentiels d'un système embarqué:

1.6.1 Les mémoires :

Il existe plusieurs types de mémoires qu'on peut classer selon différents critères :

- **Les mémoires non volatiles :** Une mémoire non volatile est une mémoire informatique qui conserve ses données en l'absence d'alimentation électrique.

ROM (Read Only Memory) : elle existe en général sous une petite taille dans un système embarqué et elle contient le programme de démarrage du système (bootstrap) qui charge généralement un autre programme (système opératoire) qui est mis au préalable dans la mémoire flash du système et Par convention la ROM n'est accessible qu'en lecture seule et son temps d'accès est plus lent que les RAM.

PROM (Programmable Read Only Memory) : C'est des ROM vierges qu'on peut programmer une seule fois et une fois programmées, il n'est plus possible de les reprogrammer.

EPROM (Erasable Programmable Read Only Memory) : utilisées souvent durant les phases de développement des bootstrap. Ces mémoires peuvent être effacées et réécrites en utilisant des dispositifs spécifiques :

- L'écriture peut se faire sur une ligne spécifique de l'EPROM (en appliquant une tension spécifique (+12 V). L'effacement se fait sur toute la mémoire (en l'exposant aux rayons ultraviolets)
- L'effacement répété finit par endommager la mémoire. Elle n'est pas donc conseillée dans les systèmes finaux.

EEPROM (Electrically Erasable Programmable Read-Only Memory) : ce sont des mémoires non volatiles qu'on peut écrire et effacer en appliquant des tensions électriques, elles contiennent en général des données qui peuvent changer au fil du temps mais qui doivent rester après extinction du système :

- Adresse de la carte réseau
- Paramètres de configuration
- Numéro de série, ...

L'écriture et l'effacement ne nécessite pas de dispositif spécifique comme ce qui est de l'EPROM et peut se faire sur une ligne spécifique et l'effacement répété n'endommage pas l'EEPROM

Les mémoires Flash : les mémoires Flash fournissent une grande densité de sauvegarde comparées aux EEPROM avec un temps d'accès équivalent aux RAMs lentes (DRAM par exemple), elles sont utilisées pour sauvegarder le logiciel (le Firmware) du système embarqué. En pratique, ce logiciel est chargé de la mémoire Flash vers la RAM pour l'exécuter (par le bootstrap se trouvant figé dans la ROM).

L'écriture et l'effacement ne nécessite pas de dispositifs supplémentaire et l'effacement se fait par secteur et pas par ligne (la taille d'un secteur varie entre 16ko et 64ko)

-Il existe deux types de mémoires Flash :

- La NAND-FLASH : grande densité de sauvegarde mais lente
- La NOR-FLASH : petite densité de sauvegarde mais rapide

L'écriture répétée altère la mémoire Flash mais des mécanismes appropriés permettent d'isoler les secteurs défectueux

- **mémoires volatiles (RAMs):** La mémoire volatile est un espace de stockage d'ordinateur qui conserve les données uniquement tant que l'appareil est sous tension.

On retrouve deux types de mémoires volatiles :

Les SRAM (Static RAM) : elle est très rapide avec Petite densité de sauvegarde et coute cher (6 transistors par bit mémoire).

Utilisées surtout dans les mémoires cache et Scratchpad pour exécuter des portions de code les plus fréquentes ou les plus critiques (vecteurs d'interruption, piles et Heap, ...etc.).

Les DRAM (Dynamic RAM) : contrairement a la mémoire SRAM la mémoire DRAM possède une très grande densité de sauvegarde (1 transistor par bit mémoire) et lentes, elles contiennent le programme en exécution et ses données de travail.

1.6.2 Les périphériques d'entrée/sortie :

Tout comme les ordinateurs conventionnels, les systèmes embarqués disposent d'un ensemble de périphériques d'entrées/sorties permettant au processeur de dialoguer avec l'environnement extérieur.

Les périphériques des systèmes embarqués sont plus nombreux et plus variés que les ordinateurs classiques et les périphériques dialoguent avec le processeur à l'aide d'interfaces d'entrées/sortie.

On y retrouve généralement comme interface :

- ✓ Des contrôleurs audio, vidéo, réseau, ...etc.
- ✓ Des ports d'entrées sorties à usage général pour interfacier des dispositifs externes
- ✓ Des ports série pour interfacier les équipements en mode série, USART, USB⁸, SPI⁹, I2C¹⁰, ...etc.
- ✓ Des convertisseurs analogiques/ numériques (CANs) interfaçant des capteurs
- ✓ Des Timers/compteurs pour des applications temps réel

1.6.3 Les processeurs :

Etant donnée la grande variété d'applications embarquées, une grande variété de processeurs est utilisée. Il y a des processeurs à faible puissance, lents, peu coûteux, mais

⁸ USB: Universal Serial Bus.

⁹ SPI: Serial Peripheral Interface.

¹⁰ I2C ou IIC : Inter Integrated Circuit.

aussi des processeurs à haute performance, à usage spécial...etc. Cette section donne un aperçu de certains des types de processeurs disponibles.

➤ **Les microcontrôleurs :**

Le microcontrôleur est comme un mini-ordinateur doté d'un processeur, de RAM, de ROM, de ports série, de timers et des périphériques d'E/S, le tout intégré sur une seule puce. Il est conçu pour effectuer des tâches spécifiques aux applications nécessitant un certain degré de contrôle, telles que la télécommande, écran à LED, montres intelligentes, véhicules, contrôle des feux de circulation, contrôle de température, etc. Il s'agit d'un appareil haut de gamme doté d'un microprocesseur, d'une mémoire, et les ports E/S sur une seule puce. C'est le cerveau d'un système informatique qui contient suffisamment de circuits pour exécuter des fonctions spécifiques sans mémoire externe. Comme il manque de composants externes, la consommation d'énergie est moindre, ce qui le rend idéal pour les appareils fonctionnant sur batteries. En termes simples, un microcontrôleur est un système informatique complet avec moins de matériel externe.

➤ **Les processeurs DSP :**

Les DSPs (Digital Signal Processors) sont des processeurs utilisés dans le traitement du signal. Initialement utilisés pour gérer la carte son d'un ordinateur, les DSPs ont vu leurs utilisations s'accroître considérablement depuis 1985, tout d'abord grâce au développement des télécommunications (téléphonie numérique, puis téléphonie sans fil GSM ...), puis grâce à leurs possibilités de traitement rapide de certaines commandes numériques faisant appel à des algorithmes complexes permettant ainsi le travail en « temps réel ».

Les DSPs ont une architecture optimisée pour traiter une grande quantité de données en parallèle à chaque cycle d'horloge.

➤ **Les processeurs graphiques GPU :**

L'unité de traitement graphique GPU (Graphics Processing Unit) est un processeur dédié au traitement des données graphiques. Ces processeurs remontent aux années 1970, d'abord utilisés à des fins de décryptage, ils ont évolué vers des modèles plus généraux de programmation, et donc ont commencé à apparaître dans d'autres applications de calcul intensif, comme l'instrumentation (accélérateur de particules). Les GPU sont généralement assez gourmands en énergie, et donc aujourd'hui ne sont pas un bon challenge pour les applications embarquées à contraintes énergétiques.

1.6.4 Les systèmes sur puce [3]:

Les systèmes embarqués envahissent jour après jour notre vie quotidienne et professionnelle. Ils couvrent plusieurs domaines tels que la télécommunication, l'avionique, l'automobile, la santé, etc. Grâce à un taux d'intégration de transistors sur la même puce qui n'a pas cessé d'augmenter, un type spécial de systèmes complexes apparaît dans les années 70. Ce sont les systèmes sur puce 1. Un système sur puce (SoC) est un système hétérogène capable d'intégrer plusieurs composants sur une même puce. Parmi les composants qu'on peut trouver dans un système sur puce, on compte les processeurs (hardcore, softcore, DSP, ...), les mémoires (RAM, ROM, flash, ...), les interconnexions qui peuvent suivre plusieurs topologies (bus, crossbar, réseau sur puce, ...), les interfaces externes (USB, Ethernet, PCI, FireWire, ...), les convertisseurs analogiques-numériques, les capteurs et les horloges. Les systèmes sur puce peuvent être statiques ou reconfigurables. Parmi les implémentations les plus utilisées pour un système statique, on trouve les implémentations sur des ASICs (Application-Specific Integrated Circuits). Comme leur nom l'indique, les ASICs sont des circuits personnalisés pour une application spécifique. Ils ont comme avantages la réduction du nombre de composants sur le circuit imprimé par rapport aux circuits génériques, la confidentialité qu'offre la personnalisation du circuit et l'optimisation du circuit en termes de performances et de consommation d'énergie. Cependant, les ASICs ont plusieurs inconvénients à cause du fait qu'ils soient dédiés seulement à une seule application qui ne peut pas être modifiée après la fabrication, et du coût de conception élevé de ces circuits qui ont aussi un long temps de mise sur le marché au cours duquel plusieurs erreurs de conception peuvent se produire. Comme alternative apportant souvent des solutions acceptables à ces problèmes, les circuits programmables et reconfigurables sont utilisés. Les systèmes reconfigurables sur puce peuvent intégrer les mêmes composants supportés par un SoC traditionnel à la différence qu'ils sont implémentés sur du matériel reconfigurable. De ce fait, ils offrent une grande flexibilité par rapport aux SoCs classiques. La reconfiguration permet à ces systèmes d'être reconfigurés un nombre illimité de fois, et offre aux concepteurs la possibilité d'ajouter de nouvelles fonctionnalités et d'effectuer des modifications sur un système après sa fabrication. La reconfiguration dynamique est un type spécial de la reconfiguration, qui permet la modification d'un système au cours de l'exécution en introduisant le concept du matériel virtuel. Ce type de reconfiguration permet aux systèmes de s'adapter à des changements au cours de l'exécution. Ces changements peuvent être liés aux

préférences des utilisateurs, aux contraintes de qualité de service, de surface, de consommation d'énergie, etc.

Dans ce contexte, les FPGAs (Field Programmable Gate Arrays) représentent une solution idéale pour implémenter la reconfiguration dynamique du fait de leur grande flexibilité par rapport aux SoCs traditionnels. La reconfiguration dynamique partielle (RDP) est un type de reconfiguration supporté par certains FPGAs et permet de reconfigurer une partie du FPGA sans perturber le fonctionnement du reste. Ce type de reconfiguration permet par exemple de remplacer un module matériel, réalisant une tâche de l'application par un autre. Cela permet de cibler de larges applications en utilisant un nombre limité de ressources. Un tel mécanisme permet d'améliorer la performance globale en faisant se chevaucher l'exécution et la reconfiguration des différentes portions du système.

Dans les stratégies de conception actuelles, un système embarqué est généralement intégré sur un support silicium unique constituant ainsi un système complet intégré sur une puce SoC (System on a Chip).

1.7 les systèmes d'exploitation pour embarqué[2]

Il existe une multitude de systèmes d'exploitation à embarquer parmi ces systèmes on cite :

VxWorks et pSOS : VxWorks est aujourd'hui le noyau temps réel le plus utilisé dans l'industrie. VxWorks inclut en natif un support TCP/IP et une interface de programmation intégrant les *sockets*.

QNX : QNX est un noyau temps réel de type Unix très intéressant. Il est parfaitement conforme à Posix, permet de développer directement sur la plate-forme cible et intègre l'environnement graphique *Photon*, proche de *X Window System*. Conscient de la percée de Linux dans le monde de l'embarqué, QNX Software s'en est rapproché en mettant à disposition la majorité des outils GNU sur la plate-forme QNX. Autre avantage non négligeable : QNX peut être utilisé gratuitement pour des applications non commerciales. Il peut occuper une très faible empreinte mémoire et, de ce fait, peut être utilisé dans des environnements très réduits comme les téléphones portables GSM Timeport de chez Motorola.

μC/OS (micro-C OS) et μC/OS II : μC/OS, développé par le Canadien Jean J. Labrosse, est destiné à des environnements de, très petite taille comme des microcontrôleurs de type Motorola 68HC11. Il est maintenant disponible sur un grand nombre de processeurs et peut intégrer des protocoles standards comme TCP/IP (μC/IP).

Windows CE : Annoncé avec fracas par Microsoft comme le système d'exploitation embarqué *qui tue*, Windows CE et ses cousins comme Embedded Windows NT n'ont pour l'instant pas détrôné les systèmes embarqués traditionnels. Victime d'une réputation de fiabilité approximative, Windows CE est pour l'instant cantonné à l'équipement de nombreux assistants personnels (PDA, *Personal Digital Assistant*).

Nucleus : Noyau RT avec support de TCP/IP, une interface graphique, un navigateur web et serveur HTTP.

LynxOS : Système RT conforme à la norme POSIX¹¹.

eCos : Noyau RT, faible empreinte mémoire, basé sur Linux, support de TCP/IP. Licence proche GPL¹². Largement utilisé (automobile, imprimante, lecteur MP3). Support des processeurs x86, PowerPC, SH3 Hitachi ou StrongARM.

1.8 les langages de programmation utilisés pour le développement d'application embarquée [4] :

Il faut noter que plusieurs langages de programmation se veulent dédiés à l'embarqué. Parmi ces langages se trouvent Ada et le langage assembleur, ce dernier restant encore un choix approprié pour les systèmes soumis à des contraintes sévères de temps réel.

Des langages proches de la machine comme le C et dans une moindre mesure le C++ sont aussi utilisés ; ce qui justifie leur position dans le dernier sondage.

C et C++ confortent également leur place respective dans le dernier classement de l'IEEE des meilleurs langages pour les systèmes embarqués. On voit aussi d'autres langages tels qu'Arduino, Haskell, D, LabVIEW et VHDL qui sont bien classés (figure I.2).

¹¹ POSIX (Portable Operating System Interface X): famille de normes techniques de standardisation des interfaces

¹² GPL (General Public License): est une licence qui fixe les conditions légales de distribution des logiciels libres du projet GNU.

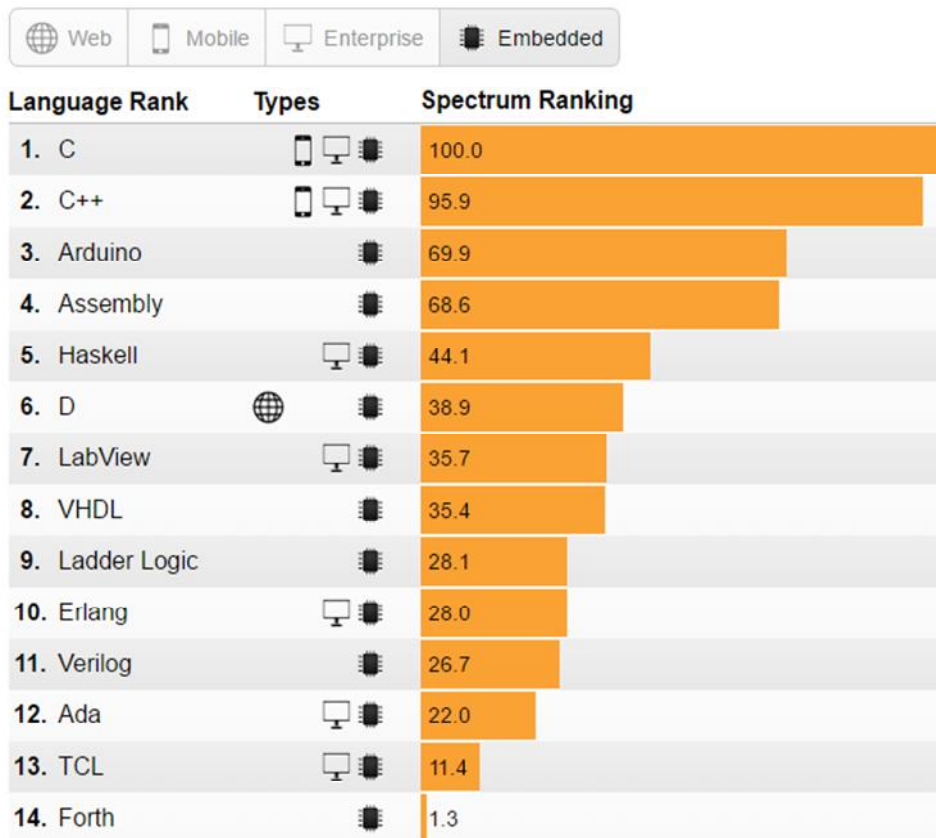


Figure 1.4 : Meilleurs langages pour le développement de systèmes embarqués

1.9 Exemples d'application des systèmes embarqués:

Les systèmes embarqués «animent» ou même «habitent» de très nombreux objets et/ou systèmes. Et leur périmètre ne cesse de croître. Ci-après, trois exemples d'application : Voitures et Transports ; Aide à la personne ; Quartiers, Bâtiments et Maisons.

Voitures et transports :

C'est dans ce domaine que s'applique une très grande quantité d'innovations. Il suffit d'observer l'expression des besoins et les souhaits de l'automobiliste : il exige de sa voiture, sécurité, fiabilité, communicabilité, personnalisation, assistance, services, sobriété, respect de l'environnement et distraction. Il souhaite aussi de nombreuses fonctions automatisées : créneaux simplifiés par exemple, contrôle de vitesse et de distance entre véhicules, communication actualisée avec l'infrastructure routière... Autant d'exigences, autant de systèmes embarqués composites, inter-opérants, à fiabilité et capacité accrues. Les spécialistes de l'électronique, de l'informatique embarquées et de la communication mobile savent d'ores et déjà réaliser tout ça, ou alors ils sont en train de le développer.

Outre la voiture, les projets innovants faisant appel aux systèmes embarqués se portent également sur le transport professionnel : fret et logistique sur route (avec le suivi temps réel de flotte et l'optimisation des tournées); trains, métro, tramway, bus (de l'informatisation des aiguillages aux systèmes vidéo de sécurité des passagers en passant par les fonctions intelligentes d'économie d'énergie).

Aide à la personne :

Assistance et sécurité : les systèmes embarqués ont un rôle-phare à jouer avec, et autour de l'Homme, dans sa sphère professionnelle comme personnelle. Au chapitre de la mobilité, le citoyen se veut joignable à chaque instant avec les téléphones portables. Ce terminal, fer de lance de l'embarqué grand public, a pris en quelques années une grande importance : il a su dépasser sa fonction première téléphonique et empiler les applications : photo, vidéo numériques, lecteur de musique, terminal de présence, GPS, outil de téléchargement et d'échange de fichiers, boîte de réception et d'envoi de mails, messagerie instantanée, terminal de consultation de réseaux sociaux, terminal de paiement sécurisé, de paiement du transport, d'assistance...etc.

Il y a aussi toutes les applications professionnelles, et celles des PDAs : connexion mobile et sécurisée au système d'information de l'entreprise, pour les commerciaux itinérants, les techniciens, les collaborateurs nomades et les télétravailleurs.

QUARTIERS, BATIMENTS ET MAISONS :

Les infrastructures publiques, les quartiers, les bâtiments comme les maisons elles-mêmes font l'objet d'une nouvelle attention, sous la double contrainte coûts et efficacité, en rapport avec le développement durable.

En France de très nombreuses mesures, notamment issues du Grenelle de l'Environnement, vont tenter de limiter la dépense énergétique en obligeant à la modernisation mais aussi en encourageant à l'investissement et à la recherche de moyens, appareils et systèmes d'information ayant pour objet la mesure permanente, le contrôle, la maîtrise et l'optimisation des ressources principales consommées par les habitats : électricité, chauffage, climatisation, eau...

1.10 Conclusion

Ce chapitre nous a permis de voir que les systèmes embarqués ont certaines particularités en ce qui concerne leur architecture et leurs systèmes d'exploitation, et cela leur permet de réaliser une tâche bien précise tout en respectant un bon nombre de contraintes. Devenus indispensables dans divers domaines.

Le prochain chapitre sera consacré à introduire les concepts de géolocalisation.

Chapitre 2

Géolocalisation GPS/GSM/GPRS

2.1 Introduction :

La géolocalisation est un procédé permettant de positionner un objet (une personne, un véhicule, ...) sur un plan ou une carte à l'aide de ses coordonnées géographiques.

Derrière ce mot compliqué se cache donc une notion assez simple : en donnant une origine et des coordonnées, tout le monde est capable de retrouver un objet en utilisant les technologies de géolocalisation.

Là où les choses se compliquent, c'est qu'il faut que la personne à qui on veut situer l'objet doit connaître le lieu d'origine, il faut donc avoir un langage commun, compréhensible de chacun (une carte par exemple).

On veut pouvoir situer cet objet partout dans le Monde de la même façon, d'où la nécessité de s'entendre sur des coordonnées absolues, acceptées par tous (exemple latitude et longitude).

Mais la plus grosse difficulté qu'a dû surmonter l'Homme, c'est d'arriver à situer en temps réel un objet en mouvement. D'où des problématiques telles que : comment détecter un objet, comment récupérer des informations sur sa position (en communiquant avec lui ou contre son gré), comment ensuite le situer précisément et rapidement sur une carte.

Pour cela dans ce chapitre et dans le cadre de notre projet, nous allons définir la géolocalisation, nous détaillerons ensuite le système de positionnement par satellite le plus connu, le GPS. Les données GPS devant être transmises à distance, nous allons voir les réseaux utilisés notamment le GSM. Ce chapitre porte aussi sur les outils de localisation et leur utilisation pour le suivi des véhicules.

2.2 La géolocalisation :

2.2.1 Définition :

La géolocalisation est un procédé permettant de positionner un objet sur un plan ou une carte à l'aide de ses coordonnées géographiques. Cette opération est réalisée à l'aide d'un terminal capable d'être localisé et de publier ses coordonnées géographiques (latitude/longitude). Les positions enregistrées peuvent être stockées au sein du terminal et être extraites postérieurement, ou être transmises en temps réel vers une plateforme logicielle de géolocalisation. La transmission temps réel nécessite un terminal équipé d'un moyen de télécommunication de type GSM, GPRS, UMTS, radio ou satellite lui permettant d'envoyer les positions à des intervalles réguliers. Ceci permet de visualiser la position du terminal au sein d'une carte à travers une plateforme de géolocalisation le plus souvent accessible depuis internet. [5]

2.2.2 Techniques de géolocalisation :

2.2.2.1 Géolocalisation par satellite :

La géolocalisation par satellite consiste à calculer, grâce aux signaux émis par une constellation de satellites prévue à cet effet, la position actuelle sur la face terrestre d'un terminal équipé d'une puce compatible. Cette position est traduite en termes de latitude, longitude et parfois altitude et peut alors être représentée physiquement sur une carte. Le réseau satellite de positionnement le plus connu est le GPS (Global Positioning System). Ce dernier utilise un système de guidage par satellite. Les satellites forment un maillage dans le ciel et servent de repères aux navigateurs GPS pour déterminer une position grâce à des coordonnées géographiques (latitude et longitude). [6]

Ce système est conçu de sorte à ce qu'il y ait toujours au minimum quatre satellites identifiables par les navigateurs GPS. Equipé d'une puce électronique, le boîtier GPS embarqué dans un véhicule capte les signaux émis par un réseau de satellites et peut ensuite calculer sa position géographique de 10 à 15 mètres près. En général, un boîtier GPS met à jour sa position toutes les secondes, voire plus (jusqu'à 100 Hz).

La figure ci-dessous illustre le Repérage d'une position par satellites.



Figure 2.1 : Repérage d'une position par satellites.

2.2.2.2 Géolocalisation par GSM : [7]

Avec cette technique de géolocalisation, on trouve la position d'un terminal GSM¹ en se basant sur certaines informations relatives aux antennes GSM auxquels ce dispositif est connecté. 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. Il existe plusieurs techniques de géolocalisation par GSM :

- **Le différentiel temps** : aussi appelée EOTD², Le principe est simple, le téléphone émet des signaux vers les antennes environnantes. La plus proche renvoie ce signal; le calcul entre le temps d'émission et de réception du signal se fait par un serveur externe qui calcule la localisation du téléphone dans le réseau.

La figure ci-dessous illustre le principe d'EOTD.

¹GSM: Global System for Mobile communication.

²EOTD: Enhanced Observed Time Difference.

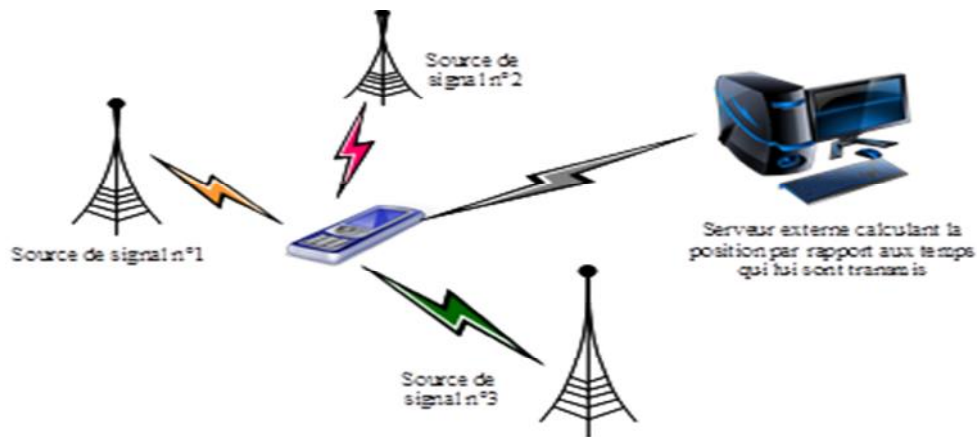


Figure 2.2 : Principe d'EOTD

- **La triangulation** : Le portable est en liaison permanente avec l'antenne de sa cellule, et "écoute" également les cellules voisines. Aussi peut-il déterminer sa distance par rapport à trois antennes relais en fonction de l'intensité des signaux radioélectriques qu'il reçoit. Ces informations sont transmises à un terminal externe qui calcule sa position géographique. Pour cela, le téléphone doit avoir reçu un programme spécifique sur sa carte SimToolkit³.

La figure ci-dessous illustre le principe de la triangulation.

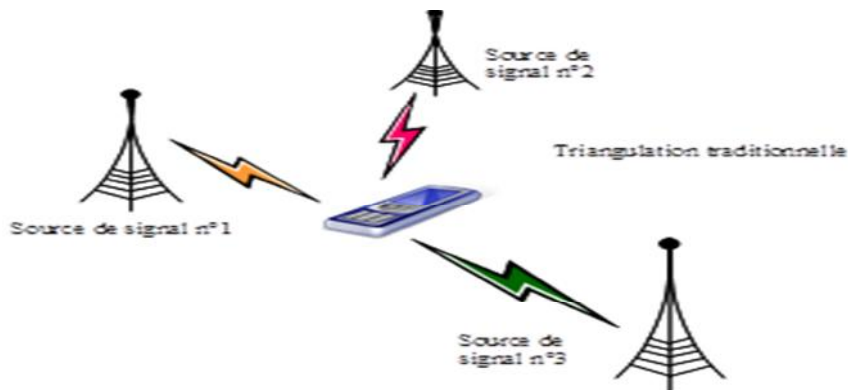


Figure 2.3 : Principe de la triangulation

³Simtoolkit : un ensemble d'applications installées sur les téléphones permettant d'accéder rapidement aux services fournis par l'opérateur mobile.

- **L'identification des cellules ou <<Cell-ID>>** : C'est la plus simple des méthodes. Les téléphones mobiles communiquent à chaque instant avec une antenne relais qui couvre une cellule. L'opérateur repère de quelle antenne relais le téléphone mobile dépend et relève ses coordonnées. Dans les zones où les antennes sont nombreuses et rapprochées, la localisation sera plus précise (la précision peut varier de 100 à 700 mètres) que dans les zones rurales où il y a peu d'antennes (la précision peut s'étendre jusqu'à 10 kilomètres). Cette localisation ne demande aucun changement des terminaux mais seulement une amélioration logicielle.

La figure ci-dessous illustre le principe du Cell ID.

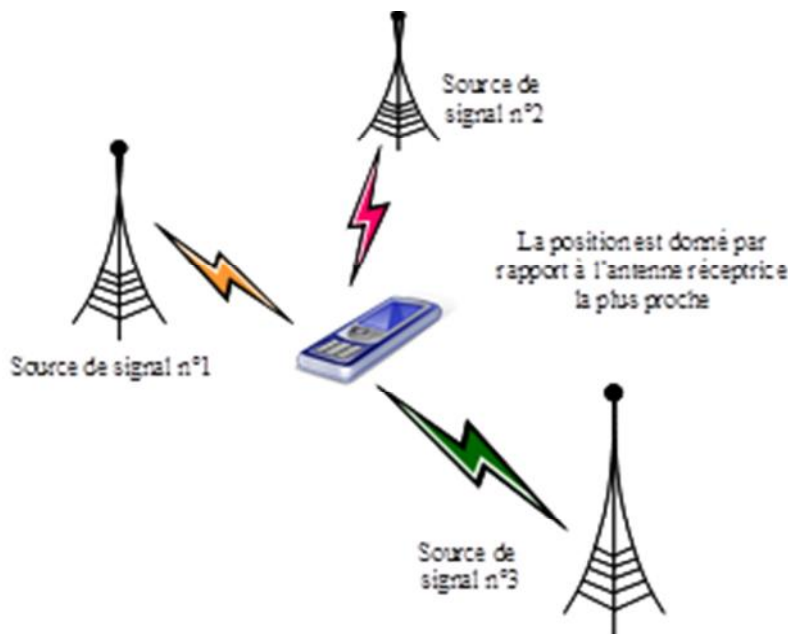


Figure 2.4 : Principe du Cell ID

2.2.2.3 Géolocalisation par WIFI :

Tout comme la géolocalisation par GSM, un terminal Wifi peut être localisé grâce à la méthode Cell ID. Dans ce cas-ci, c'est l'identifiant des bornes Wifi qui est détecté. Des bases de données reprenant ces identifiants et leur position géographique sont communiquées par des entreprises privées et d'autres communautés. [8]

2.2.2.4 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'IANA⁴, 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. On peut même obtenir un niveau de précision de l'ordre de la ville en se basant sur la distribution des adresses IP faite par les fournisseurs d'accès à internet.

Il existe plusieurs sites pour géolocaliser, qui utilise la technique d'adressage IP.

Voici un exemple qui a été réalisé par le site **Hostip.info**. [9]

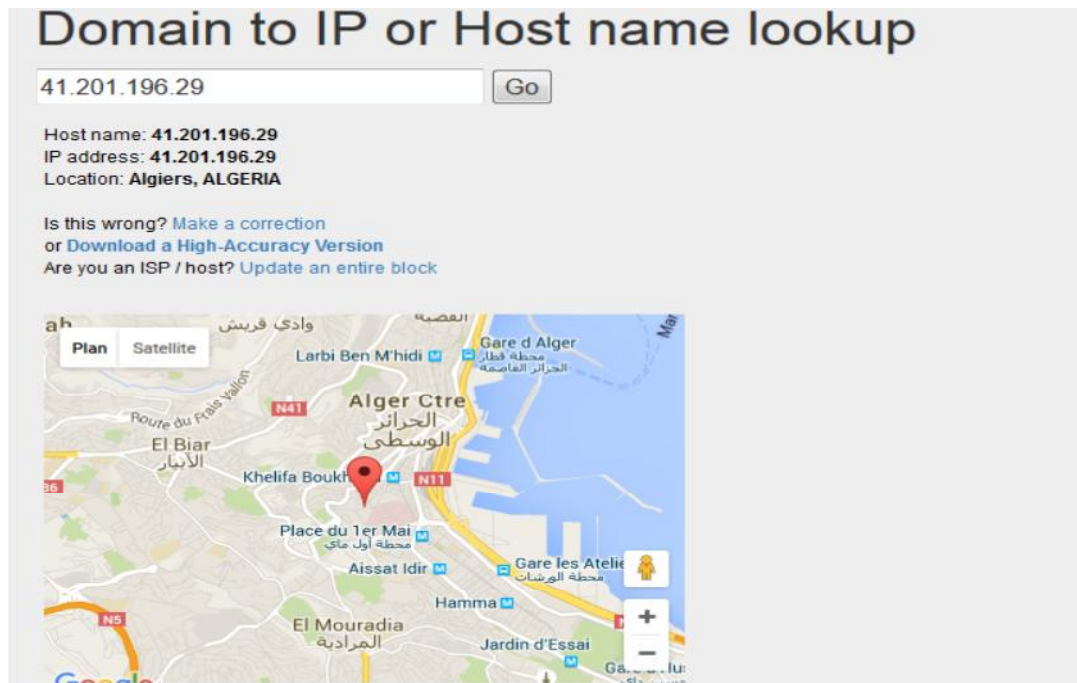


Figure 2.5 : Technique de localisation par adressage IP.

- **Hostip.info** est un site communautaire qui génère une base de données de géolocalisation.

Les données sont issues des membres qui en s'inscrivant et en soumettant leurs propre adresses postales et adresse IP, renseignent la base. Celle-ci compte déjà plus de 8 millions d'entrées.

Le site propose une API de type REST qui se veut simple et facile à intégrer.

Le service fournit différents types de données comme les noms des villes et les Coordonnées géographiques (latitude, longitude).

⁴IANA: Internet Assigned Numbers Authority

2.2.2.5 Géolocalisation par RFID :

Le RFID⁵ 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. [8]

2.2.2.6 Combinaison de ces techniques :

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, plus complexe en milieu rural. Il existe 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. [8]

2.2.2.7 Géolocalisation indoor :

La géolocalisation indoor ou la géolocalisation d'intérieur est une technique qui permet de localiser en temps réel des biens ou des personnes dans des espaces fermés tels que les centres commerciaux, aéroports, hôpitaux, usines, complexes militaires ou industriels, etc...

Grâce aux nouveaux outils de mobilité : Smartphones mobiles, tablette tactile durcie... et aux nouvelles technologies : Wifi, Bluetooth ou Ultra Wide Band et RFID , les utilisateurs peuvent se repérer facilement à l'intérieur d'un bâtiment. Ces informations peuvent être utilisées également par les propriétaires de sites pour suivre le parcours des clients en temps réel et interagir avec eux en leur proposant des services et déclencher des actions de marketing.

Les systèmes de géolocalisation indoor permettent de connaître avec une précision plus ou moins grande le positionnement d'une personne ou d'un objet dans un espace ou un lieu dans lequel l'accès aux satellites et les données GPS ne sont pas disponibles.

⁵RFID : Radio Frequency Identification.

2.3 Plateformes logiciels de géolocalisation :

2.3.1 Composants :

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 sur le terrain (sens interdits, restrictions pour les poids lourds, vitesses autorisées...).

2.3.2 Architecture temps réel :

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 :

1. Le terminal détermine sa position géographique grâce à une des techniques de géolocalisation citées précédemment (de préférence GPS, GSM et/ou Wifi);
2. Il envoie ces données vers une plateforme logicielle soit par le réseau GSM/GPRS soit par un réseau satellitaire de type Inmarsat⁶;

⁶Inmarsat : pour *IN*ternational *MA*rtime *SA*Tellite *OR*ganisation, est une société britannique du secteur des télécommunications spécialisée dans la téléphonie par satellite.

3. 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 télérelève (trafic routier, autonomie du véhicule, points à visiter etc.), des calculs d'itinéraires ou de tournées peuvent par exemple être générés.
4. 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) nous recensons deux principaux moyens de transmission : le réseau GSM/GPRS et le réseau satellitaire. Les schémas suivant présentent les architectures types illustrant ces deux modes de transmission :

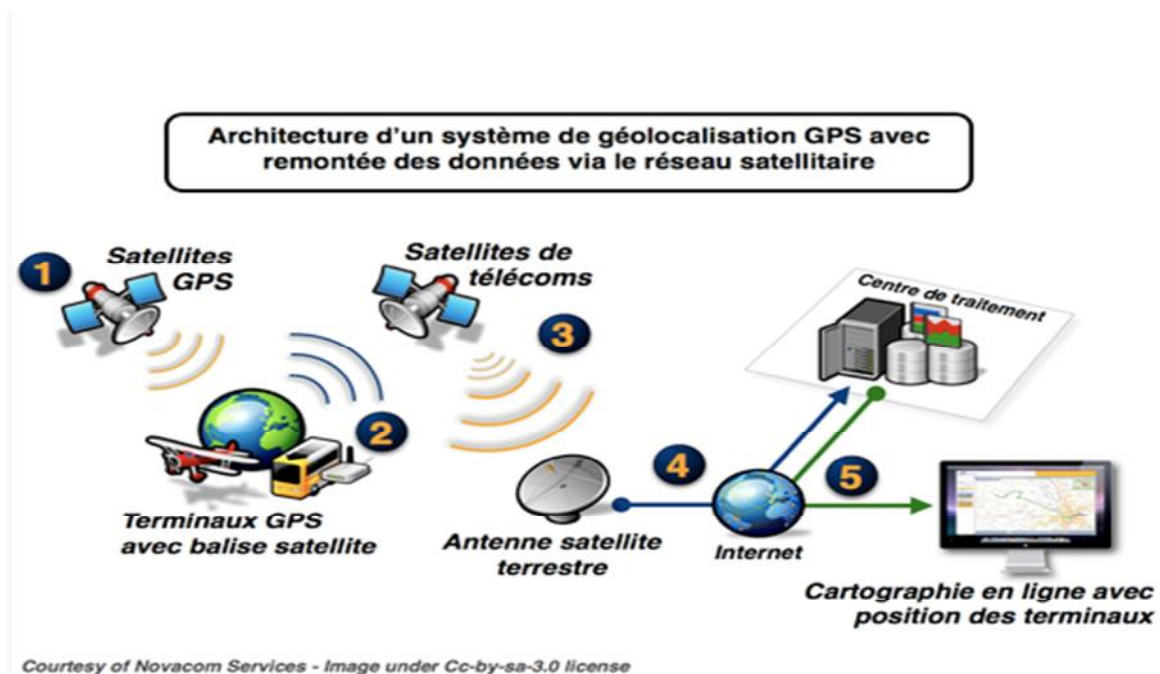


Figure 2.6 : Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau satellitaire.

Les satellites de télécommunications sont utilisés pour transmettre des informations d'un point à l'autre de la Terre, notamment les communications téléphoniques, la transmission de données. Le satellite de télécoms peut, à tout moment, recevoir et transmettre des messages à n'importe quel émetteur ou émetteur-récepteur installés à terre qui se trouve dans la zone de couverture géographique délimitée à portée visuelle du satellite.

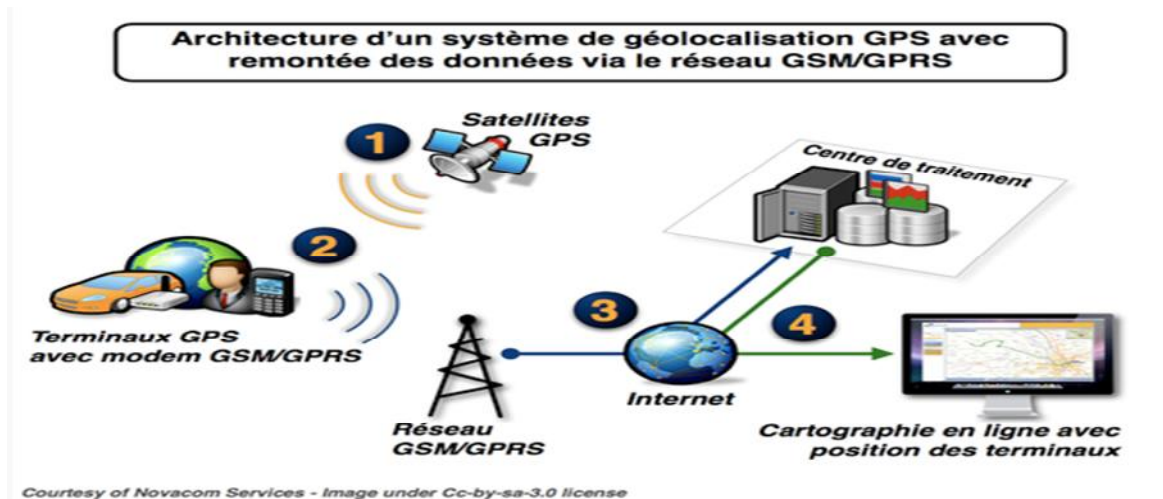


Figure 2.7 : Architecture d'un système de géolocalisation GPS avec remontée des données via le réseau GSM/GPRS.

Ce moyen de transmission nécessite un boîtier disposant d'un modem GSM/GPRS⁷ ou 3G/UMTS⁸, ainsi que d'une carte SIM de n'importe quel opérateur. Le boîtier nécessite d'être sous couverture GSM/GPRS pour pouvoir envoyer les données vers un serveur distant. Ce dernier traite les données et les met à disposition des utilisateurs. Les utilisateurs par la suite lisent les données depuis leur PC et/ou Smartphones en se connectant au serveur, via une connexion internet. Ce moyen est utilisé lorsque l'objet ou la personne à géolocaliser reste dans une zone bien couverte par les réseaux GSM/GPRS.

2.4 Principes du GPS :

Le GPS est devenu aujourd'hui une avancée technologique que beaucoup d'entre nous utilisent. Il faut noter que notre système repose sur la géolocalisation GPS/GSM, pour cela dans cette partie on va exposer la technologie GPS et son fonctionnement.

2.4.1 Définition :

Le Système de positionnement mondial est un système de géolocalisation par Satellite. C'est un réseau de 24 satellites actuellement en fonctionnement, Développé à l'origine par les USA pour des besoins militaires (opéraonnel fin 1993) .Il a été mis à la disposition des applications civiles par la suite. Il permet de déterminer les coordonnées géographiques de n'importe quel point situé à la surface du globe. Sa précision peut atteindre 1 mètre .Le GPS s'utilise en association avec une carte pour se repérer et se positionner.

⁷GPRS: General Packet Radio Service.

⁸UMTS : Universal Mobile Télécommunications System.

⁹NMEA: National Marine Electronics Association.

2.4.1 Historique :

Historiquement, les premières informations GPS ont été utilisées par les marins, militaires de surcroît. Il n'est donc pas surprenant que le protocole de communication adopté par le GPS soit le protocole NMEA⁹. Ce protocole a en effet été conçu à l'origine pour faciliter les échanges de données entre les différents instruments de bord sur les bateaux, et ce bien avant l'existence des GPS. [10]

2.4.2 Différents segments du GPS:

Lorsque nous parlons de GPS, nous avons tendance à penser systématiquement et exclusivement à l'appareil qui sert à donner une position. Cependant, le système GPS ne se limite pas à cet instrument, puisqu'il se compose de trois éléments distincts, appelés segments ou secteurs. Le premier segment, composé de satellites, est appelé « segment spatial ». Le deuxième segment qui comprend des stations de contrôle s'appelle « segment de contrôle ». Le dernier, correspondant aux récepteurs GPS, est dit « segment utilisateur ».

- a. **Segment spatial** : Le segment spatial est constitué d'une constellation de 24 satellites opérationnels en orbites quasi circulaires autour de la Terre à une hauteur d'environ 20 000 km, dont les satellites sont répartis sur 6 plans orbitaux inclinés 55° par rapport à l'équateur. Ces satellites transmettent des signaux unidirectionnels, lesquels donnent le positionnement et l'heure du réseau GPS. [11]

La figure ci-dessous illustre le segment spatial.



Figure 2.8: Segment spatial.

- b. **Segment de contrôle** : Le segment de contrôle se compose de stations réparties à travers le monde, elles vérifient l'état des satellites lors de leur passage au-dessus d'elles ; Ces stations transmettent ensuite leurs données à la station principale de Colorado Springs c'est là que les paramètres décrivant l'orbite des satellites, et la qualité des horloges embarquées sont estimés, la vérification de la santé des satellites et la détermination d'un repositionnement éventuel sont également contrôlés. Cette information est alors renvoyée à trois stations de chargement qui

transmettent l'information aux satellites grâce à la répartition uniforme des stations de contrôle, tous les satellites GPS sont captés à 92% du temps.

La figure ci-dessous illustre le segment de contrôle.



Figure 2.9 : Segment de contrôle.

Le segment de contrôle comprend cinq stations de poursuite situées à Colorado Springs (Colorado, US), Hawaii, Ascension Island (South Atlantic), Diego Garcia (Indian Ocean) et Kwajalein (North Pacific).



Figure 2.10. Emplacement de segments de Contrôles dans le monde.

- c. **Segment utilisateur** : Le segment utilisateur comprend les récepteurs GPS, qui reçoivent les signaux des satellites et qui calculent, sur la base des informations transmises la position tridimensionnelle et la référence temporelle de l'utilisateur.

La figure ci-dessous illustre le segment utilisateur.



Figure 2.11 : Segment utilisateur.

2.4.3 Principe de fonctionnement:

Le GPS est un système avec 24 satellites circulant à 20 200km d'altitude, ils se répartissent sur 6 orbites distinctes à raison de 4 satellites par orbite. Ces satellites émettent en permanence sur deux fréquences, datés précisément grâce à leur horloge atomique et par un message de navigation qui inclut les éphémérides permettant le calcul de la position des satellites ainsi que les informations sur leur horloge interne. Ainsi un récepteur GPS qui capte d'au moins 4 satellites peut en calculant les temps de propagation de ces signaux entre les satellites et lui, connaître sa distance par rapport à ceux-ci et par, trilatération situer précisément en trois dimensions n'importe quel point placé en visibilité des satellites GPS. [6]

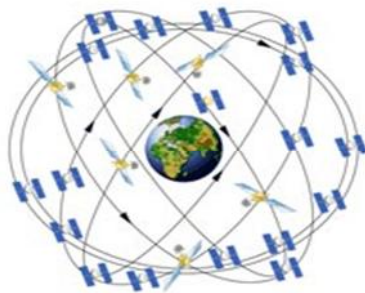


Figure 2. 12:La constellation GPS .



Figure 2.13 :Un satellite GPS .

2.4.3.1 Principes de triangulation :

Le principe du positionnement GPS repose sur le principe de triangulation. La détermination d'un lieu géographique est basée sur l'intersection de trois sphères dans l'espace. Chaque sphère est définie par son centre correspondant à la position d'un satellite et par son rayon qui est la distance entre le centre et le récepteur GPS. Un satellite émet une onde électromagnétique de vitesse connue. Le récepteur détermine le temps mis par cette onde pour l'atteindre donc la distance qui le sépare du satellite. Le récepteur sait alors qu'il se trouve sur une sphère centrée sur le satellite.

En recoupant les informations de 2 satellites, le lieu géométrique du récepteur devient un cercle.

Avec 3 satellites, l'intersection se réduit à un point (le second point étant incohérent puisque se trouvant dans l'espace). [12]

La figure suivante illustre le principe de positionnement par GPS.

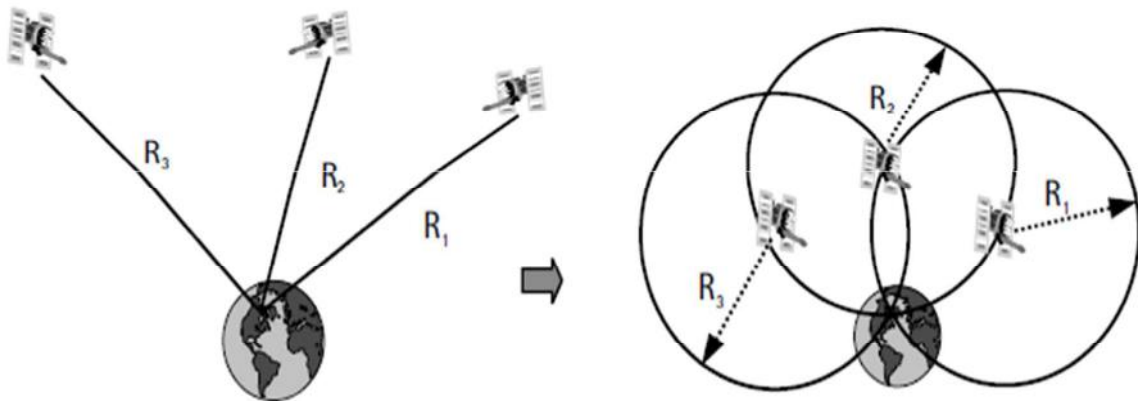


Figure. 2.14 : principe de positionnement par GPS

2.4.3.2 La mesure de distance :

Toute la démonstration précédente reposait sur l'hypothèse que l'on connaissait exactement la distance séparant le satellite du récepteur. Pour calculer cette distance, le principe est le suivant: le satellite envoie un signal vers le récepteur, celui-ci détermine le temps de transmission de ce signal et ainsi peut déduire la distance le séparant du satellite

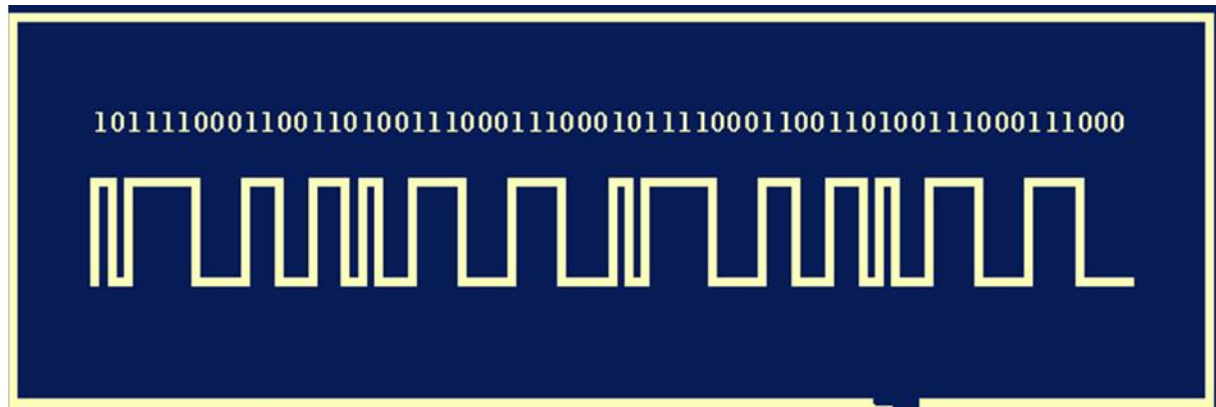
Suivant la formule :

Distance = vitesse * temps

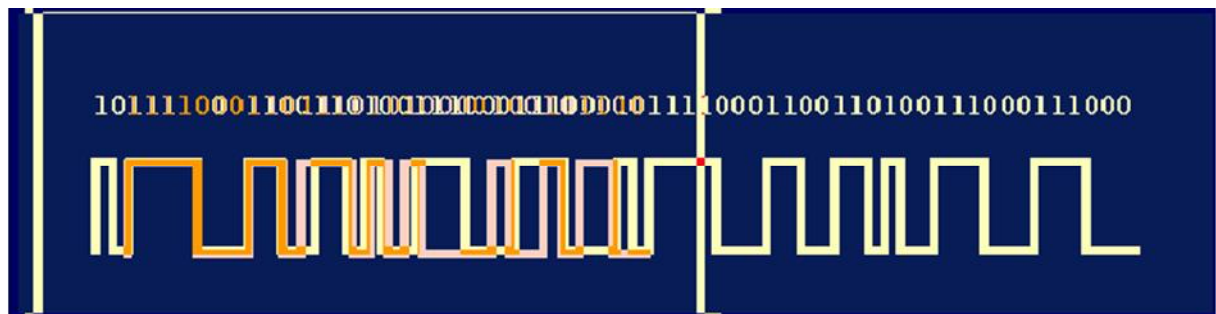
La célérité des ondes transmises est proche de celle de la lumière c'est-à-dire 300 000 km/s.

Il reste donc à déterminer le temps de transmission du signal. Pour cela, le récepteur et le satellite émettent au même moment une trame pseudo-aléatoire identique (appelée ainsi car elle est générée par des équations très complexe, la rendant ainsi unique). Une fois que cette trame sera reçue par le récepteur, celui-ci pourra la décaler dans le temps de façon à la faire coïncider avec celle qu'il a générée, la mesure du temps de transmission est déduite de ce procédé, et ainsi on peut connaître la distance séparant le récepteur du satellite.[9]

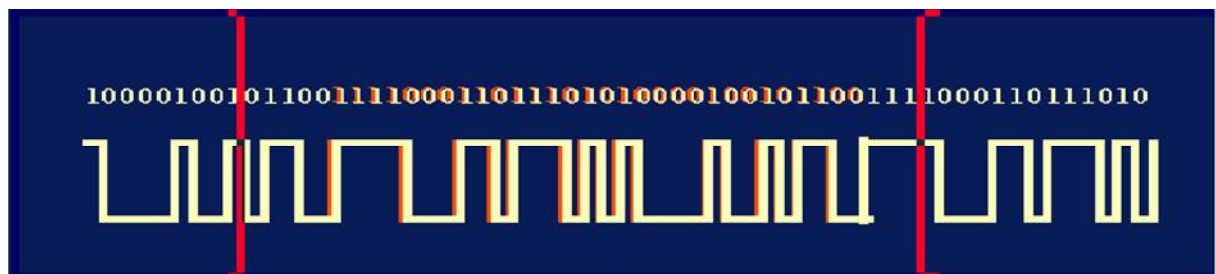
Trame générée



Trame générée + trame reçue



Trame reçue décalée de façon à coïncider avec la trame générée: obtention du dt

Décalage entre le satellite et le récepteur = t

En clair, les satellites du réseau GPS sont munis d'une horloge atomique d'une précision meilleure que 10^{-9} secondes. A un moment que nous appellerons t un satellite GPS envoie un signal horaire vers le récepteur du GPS qui le réceptionne à l'instant t_1 . Le signal se déplace comme dit précédemment à la vitesse de la lumière c'est à dire 300 000km/s, il est donc facile de savoir la distance d_1 qui sépare le récepteur GPS au satellite qui a envoyé ce signal : $(t_1 - t) / 300\,000 = d_1$

Le récepteur GPS se trouvera par définition sur le pourtour de la sphère ayant un diamètre d centrée sur ce premier satellite. En répétant cette opération trois fois on obtient le principe de la triangulation.

Les horloges des satellites sont très précises, en effet il s'agit d'horloges atomiques au Césium. Cependant les horloges des récepteurs sont beaucoup moins précises, nous verrons que cela conduira à utiliser un satellite supplémentaire.

2.4.3.3 Le positionnement des satellites :

Pour obtenir un positionnement exacte, il y a un autre détail à régler : la position des satellites pour calculer précisément la distance séparant un satellite du point à déterminer, il faut que ce dernier connaisse parfaitement sa position dans l'espace. C'est le rôle des stations de contrôle : Le satellite renvoie sa position théorique à la station de contrôle, qui calcule alors l'erreur de position commise par ce dernier afin de lui renvoyer la valeur de cette erreur. Le satellite peut donc informer le récepteur de l'erreur qu'il doit prendre en compte dans ses calculs.

2.5 Le protocole NMEA appliqué aux GPS :

La norme NMEA a été créée pour faciliter les échanges de données entre équipements électroniques de navigation. Il est donc normal que les GPS y fassent appel puisque les premiers utilisateurs de ces appareils ont été les marins.

La norme utilisée par les récepteurs GPS est la version NMEA 0183 qui précise que la transmission des données a lieu sous forme série asynchrone avec 8 bits de données, 1 bit de stop et aucune parité, à la vitesse de 4800 bits par seconde. Ces données fonctionnent par envoi de messages ou trames, toutes organisées de la façon suivante.

Une trame commence toujours par le caractère \$(dollar) suivi de deux lettres qui identifient le type d'émetteur (**GP** dans le cas d'un GPS). Viennent ensuite trois autres lettres qui identifient la trame pour l'émetteur concerné. Dans le cas des GPS, la trame **RMC** est la plus courante (et même la seule pour les appareils les plus simples) mais on trouve aussi **GGL**, **GSA**, **GGA**, etc.

Les différents champs de données viennent ensuite, séparés chacun par des virgules. L'ordre et la taille de ces champs dépendent de la trame concernée et il n'existe donc pas de règle commune à toutes les trames. Le caractère * (astérisque) suit quant à lui immédiatement la dernière donnée (sans virgule de séparation donc) et précède une somme de contrôle ou checksum de 2 caractères qui est calculée en réalisant un OU exclusif de tous les caractères du message de \$ à * compris. Voici donc, par exemple, à quoi ressemble une trame **RMC** délivrée par un GPS :

```
$GPRMC,114630.32,A,4315.0426,N,00603.2734,W,0.20,307.53,310508,,E*54
```

Si une donnée est absente d'un champ, l'emplacement de ce dernier doit tout de même être présent. On rencontre alors deux virgules successives. Les différentes trames se suivent sans ordre défini par la norme puisque elles sont clairement identifiables par leur en-tête (les cinq lettres qui suivent le caractère \$). [10]

2.6 Les moyens de communication des coordonnées géographiques :

Comme on l'a souligné précédemment, une fois la position d'un objet connue, on peut soit la sauvegarder, soit la visualiser directement sur l'équipement si celui-ci dispose d'un écran ou alors la communiquer à distance via un réseau de communication tel que les réseaux mobiles GSM, GPRS et UMTS. Dans ce qui suit, nous allons voir les services offerts par ces réseaux qui permettent la transmission des informations d'un récepteur GPS.

- **Le GSM :** Le « Global System for Mobile communication » est le système cellulaire standard de la deuxième génération des réseaux mobiles. Adopté par plus de 180 pays, aujourd'hui, plus d'un milliard d'utilisateurs sont équipés d'une solution GSM. Parmi les services qu'offre ce réseau, le service SMS « Short Message Service » qui permet à un utilisateur de composer un message textuel à partir de son terminal mobile et de l'envoyer à un destinataire possédant également un téléphone mobile. Un peu plus loin, nous allons nous intéresser à l'architecture de ce réseau, mais aussi décrire le service SMS qui est utilisé pour communiquer les coordonnées GPS notamment par notre application.
- **Le GPRS :** Le « *General Packet Radio Service* » est une extension de l'architecture du standard GSM, elle s'appuie sur la technique de transmission par paquets et permet ainsi d'accéder aux services internet avec des débits théoriques maximums de l'ordre de 171,2 kbit/s (en pratique jusqu'à 114 kbit/s). Grâce au mode de transfert par paquets, les transmissions de données n'utilisent le réseau que lorsque c'est nécessaire. Le standard GPRS permet donc de facturer l'utilisateur au volume échangé plutôt qu'à la durée de connexion, ce qui signifie notamment qu'il peut rester connecté sans surcoût. [13].
- **L'UMTS :** « Universal Mobile Telecommunications System » désigne une technologie retenue comme norme pour les systèmes de télécommunications mobiles dits de troisième génération. La 3G permet de profiter d'un très haut débit pour la transmission de données. Elle ouvre ainsi la porte à des applications et services nouveaux. Elle permet en particulier un accès internet performant et confortable et un transfert en temps réel des contenus multimédia tels que les images, le son et la vidéo.[9].

2.6.1 GSM :

Ce standard pour les communications de téléphonie mobile permet de transmettre numériquement non seulement la voix, mais aussi des données informatiques (SMS).

2.6.1.1 Le concept cellulaire du GSM :

Le principe du système cellulaire est de diviser le territoire en de petites zones, appelées cellules, au centre de chacune d'elles se situe une station de base (antenne d'émission) à laquelle on associe un certain nombre de fréquences.

La station de base communique par radio avec un mobile en lui allouant un canal de communication dès qu'il en a besoin, le terminal doit évidemment se trouver dans la zone couverte par la station. Etant donné que le nombre de fréquences accordées au système GSM est restreint (il utilise les bandes de fréquences 900 MHz et 1800 MHz), l'opérateur est obligé de réutiliser les mêmes fréquences sur des cellules suffisamment éloignées de telle sorte que deux communications utilisant la même fréquence ne se brouillent pas. [13]

2.6.1.2 Architecture d'un réseau GSM :

Le réseau mobile se décompose en trois parties distinctes :

BSS : (base station subsystem) permet l'accès aux réseaux GSM.

NSS : (Network switching subsystem) chargé de la commutation et routage.

OSS : (operation support subsystem) chargé de l'administration de notre réseau. [13]

2.6.1.3 Le service de messages courts de GSM (SMS) :

Le service de messages courts offert par le réseau GSM (SMS, pour Short Messages Service) permet à un utilisateur de composer un message textuel d'au plus 160 caractères (codés à l'aide d'ASCII 7 bits sur 140 octets) à partir de son terminal mobile et de l'envoyer à un destinataire possédant également un téléphone radio-mobile GSM ou à une entité extérieure au réseau GSM appelée SME (Short Message Entity).

Un exemple de SME est l'application " Mobile Banking " de la plate-forme ASP par exemple.

Les messages courts sont désignés dans ce rapport par SM, pour Short Messages, mais sont plus couramment désignés par Texto ou SMS. Ces messages sont soit transmis directement au terminal destinataire du message (si celui-ci est allumé), soient stockés dans le serveur de message courts (SMSC, pour SMS Center) par lequel ils transitent.

Pour une communication vocale, le canal de contrôle est utilisé pour initier la

Communication qui est ensuite transmise sur un canal de trafic. Les SMS, par contre, sont acheminés directement sur un canal de contrôle.

Un message SMS peut être transmis de deux façons :

- En mode PDU (Protocol Description Unit) qui est une suite de caractères hexadécimaux qui codifient le SMS ;

- En mode TEXT qui est une représentation sous forme de texte des données qui composent le SMS.

Différents types d'alphabets sont utilisés pour passer du mode PDU au mode TEXT. Par exemple le téléphone portable détermine automatiquement quel type d'alphabet utiliser pour afficher en mode TEXT, sur son écran, les données d'un éventuel SMS reçu.

2.7 La géolocalisation GPS/GSM/GPRS :

Les systèmes de géolocalisation GPS/GSM reposent sur le principe suivant : Une demande (un appel ou un SMS) d'information est adressée par le biais du réseau GSM à l'aide d'un terminal GSM. Le dispositif à localiser, équipé d'un récepteur GPS, reçoit la demande et calcule en temps réel sa position et la renvoie sous forme de message SMS par le réseau GSM. L'affichage de la position du récepteur, et donc des coordonnées géographiques, se fera sur

L'écran du terminal ayant émis la demande (ou sur un autre) à l'aide d'un logiciel approprié. [14].

2.7.1 Géolocalisation GPS/GSM/GPRS pour les systèmes de suivi de véhicules :

Un système de géolocalisation permet de localiser un véhicule sur une carte en utilisant ses coordonnées géographiques (latitude et longitude).

Le véhicule est suivi en temps réel et les informations sont accessibles en permanence depuis un équipement informatique (PC, tablette ou Smartphone) connecté à Internet. Grâce à un dispositif de géolocalisation, il est possible de connaître à tout moment la position exacte d'un véhicule mais aussi la vitesse à laquelle il roule, le nombre de kilomètres parcourus ou encore sa consommation de carburant.

Pour géolocaliser un véhicule, trois éléments sont nécessaires :

- **Un émetteur** (boîtier qui doit se trouver dans le véhicule à géolocaliser)
- **Un réseau** (satellite, Wifi, antennes de téléphonie mobile, RFID...)
- **Un logiciel de réception** des données et de cartographie (il peut être installé sur un ordinateur, une tablette ou un Smartphone).

2.8 Conclusion :

Dans ce chapitre nous avons introduit la notion de géolocalisation puis nous avons présenté la plateforme de géolocalisation, en précisant ses différents composants ainsi que son fonctionnement.

Par la suite nous avons abordé les techniques de géolocalisation en se focalisant principalement sur la technologie GPS et son fonctionnement.

Dans le chapitre suivant, nous allons présenter la conception de notre application en spécifiant son objectif et les moyens utilisés pour sa réalisation.

Chapitre 3

Analyse et Conception du système

3.1 Introduction :

Aujourd'hui un grand nombre d'entreprises se trouvent dans l'obligation de mettre en place des moyens de déplacement pour leurs employés, afin de transporter les marchandises et assurer des services rapides et permanents. Ainsi que nous avons pu constater, les employeurs se confrontent à deux problèmes majeurs, le premier est quand les véhicules sont utilisés hors services, ou hors zones de travail pour des intérêts personnels, ce qui en résulte un gaspillage d'énergie et l'exposition des véhicules aux pannes, et le deuxième, les situations de vols de voitures, une criminalité qui est en hausse constante.

Face à ses deux problèmes ainsi que d'autres moins graves, il s'avère que les employeurs se trouvent dans la nécessité d'utiliser les nouvelles technologies et techniques pour en remédier. Ils se penchent alors de plus en plus sur les systèmes de localisation qui sont devenus des composants indispensables dans les voitures modernes. Ainsi notre travail consiste à développer un système qui exploite les techniques de positionnement par GPS pour pouvoir localiser un véhicule et utilise le réseau GSM/GPRS pour communiquer la position.

Le présent chapitre sera consacré à la spécification des besoins du projet et à sa modélisation. Pour cela nous allons exposer le contexte du projet et analyser l'aspect matériel et logiciel du travail à réaliser. Nous allons ensuite procéder à la résolution du problème dans la partie conception à travers la conception matérielle et logicielle. Cela va nous permettre de modéliser le système, c'est à dire le représenter de manière simplifiée et abstraite.

3.2 Méthodologie de conception :

Une méthodologie de conception correspond à un ensemble de méthodes que doit suivre le concepteur afin de réaliser son logiciel, dans notre cas, nous allons suivre un processus de conception typique [15], celui-ci consiste en la spécification des besoins, la conception, le codage et les tests de validation.

Dans ce contexte, un langage de modélisation est souvent utilisé. En ce qui nous concerne, pour la modélisation de la partie logicielle, nous allons nous inspirer du support graphique du langage UML afin de mieux expliquer les caractéristiques structurelles et comportementales de notre système. Notre méthodologie peut être décrite par le schéma de la figure 3.1:

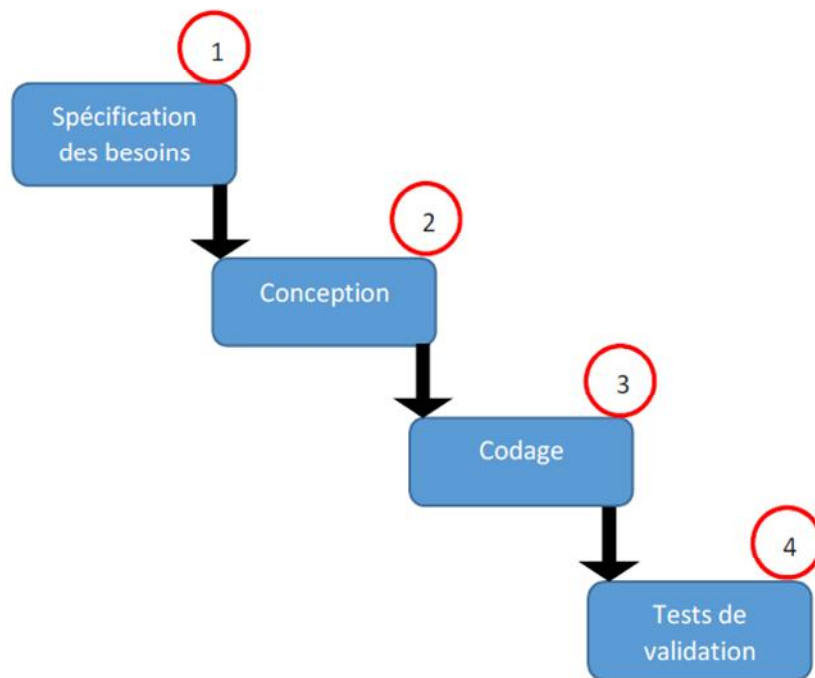


Figure. 3.1 : Schéma de la méthodologie de conception.

3.3 Analyse :

3.3.1 Le but de notre travail :

Notre travail consiste à concevoir et développer un système qui se compose de deux sous systèmes comme illustré dans la figure 3.2 :

- Un Système à embarquer dans un véhicule afin de pouvoir le localiser à n'importe quel moment et à n'importe quel instant.
- Une plateforme de suivi du véhicule qui permet de récupérer les positions du véhicule et ses informations (la vitesse) en temps réel.

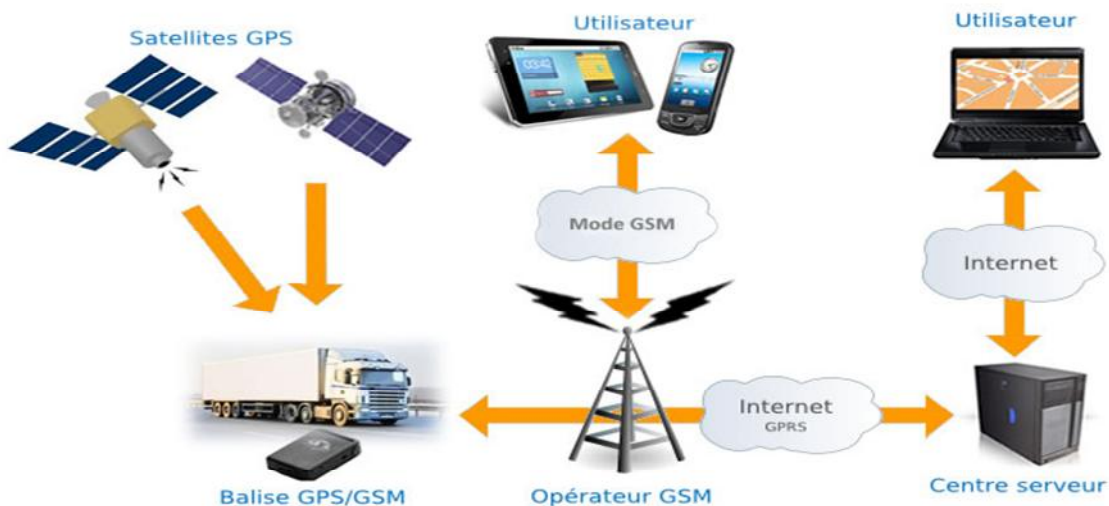


Figure. 3.2 : Fonctionnement général du système de géolocalisation.

On va suivre le véhicule en deux méthodes :

- ❖ la première est la localisation sous demande qui consiste à n'envoyer les coordonnées qu'en cas de demande de l'utilisateur par SMS. Comme on peut le voir dans la figure 2.3, via le réseau mobile GSM, l'utilisateur envoie un SMS vers le numéro de la carte SIM du système (1), ce dernier calcule sa position (2) et communique cette position sous forme d'un message (3), le message contient deux coordonnées : la latitude et la longitude, mais aussi un lien vers Google maps pour afficher la position sur une carte. Une fois le message reçu, l'utilisateur peut visualiser l'endroit où se trouve son véhicule, en utilisant un logiciel de cartographie(4).

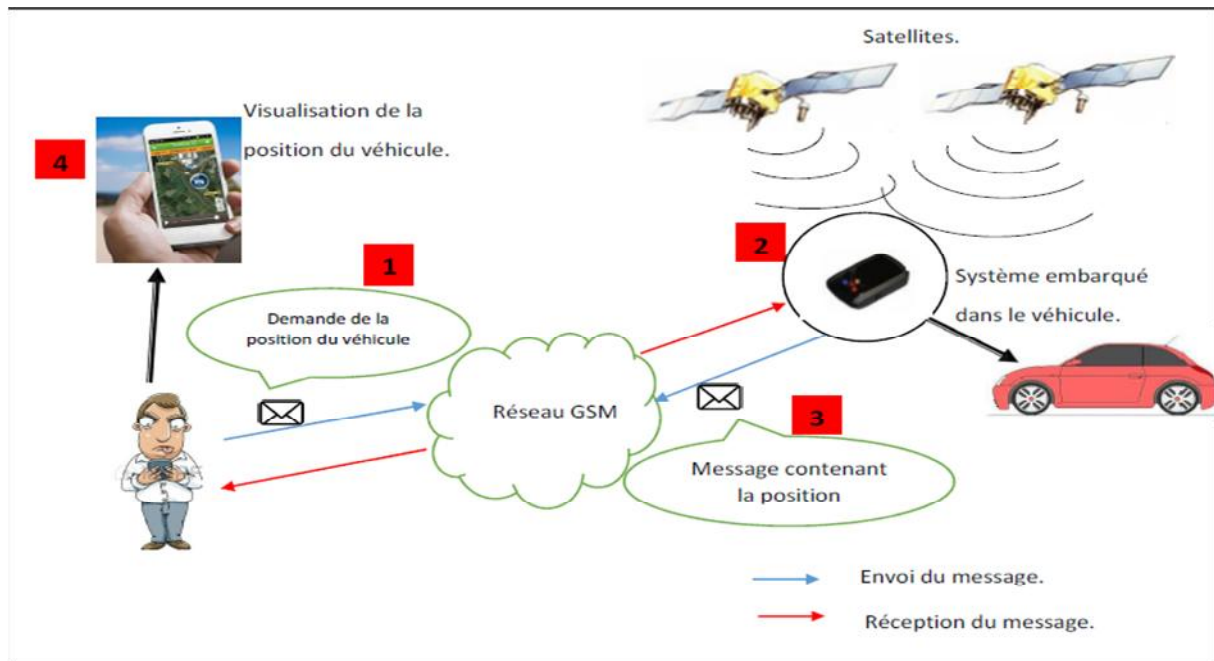


Figure. 3.3 : Fonctionnement du système de géolocalisation GSM/GPS.

- ❖ la deuxième est le tracking qui envoie périodiquement (chaque une demi-heure) la position du véhicule à une API qu'on a développée pour recevoir les données qui seront ensuite envoyées au serveur central qui loge une base de données pour les afficher dans la plateforme de suivi du véhicule. Comme on peut le voir dans la figure 3.4, le système installé dans le véhicule détermine la position du véhicule, la position sera transférée à un serveur connecté au réseau internet, et enfin l'utilisateur peut visualiser la position dans sa plateforme de suivi de véhicule.



Fig. 3.4 : Fonctionnement du système de géolocalisation GPRS/GPS.

3.3.2 Spécification des besoins fonctionnels :

L'objectif principal de l'application développée est de permettre à l'utilisateur de géolocaliser l'endroit où se trouve son véhicule en temps réel. En plus de la localisation de son véhicule, l'utilisateur du système peut visualiser l'historique de localisation de sa voiture, procéder à la configuration du système comme par exemple changer le mot de passe.

- ✓ S'authentifier dans l'application embarquée dans un véhicule.
- ✓ Localiser en temps réel le véhicule :
 - envoyer un SMS au système gps/gsm/gprs.
 - Le système renvoie un SMS qui contient un lien Google maps et deux coordonnées la latitude et la longitude.
 - le système envoie les coordonnées géographiques à un serveur distant.
 - Afficher l'endroit où se trouve le véhicule dans une plateforme de suivi de véhicule.
- ✓ Voir le statut du véhicule (arrêt ou en marche) dans la plateforme web développée.
- ✓ Modifier le mot de passe d'authentification à l'application embarquée.
- ✓ Visualiser l'historique de localisation du véhicule dans la plateforme de suivi du véhicule.

3.3.3 Spécification des besoins matériels :

Afin de réaliser notre système, on doit utiliser un ensemble de composants matériels en interaction, chacun réalisant une tâche précise.

Pour cela:

- Le système doit pouvoir se localiser de manière précise, chose qui est permise par les systèmes de positionnement par satellite, le GPS est un système de géolocalisation accessible et entièrement opérationnel mais aussi le plus répandu, d'où le choix d'utilisation d'un récepteur GPS.
- Pour pouvoir recevoir et envoyer les messages, le système doit disposer d'un module de communication radio, sur une longue distance, le choix se tourne alors vers un réseau mobile, mais celui-ci doit assurer la meilleure couverture possible, et être accessible par le grand public, d'où le choix d'utilisation d'un module GSM.
- Pour contrôler les modules cités ci-dessus, traités les différentes informations et assurer le bon fonctionnement du système, on a besoin d'une carte à microcontrôleur et nous allons utiliser arduino.

- Nous aurons besoin d'envoyer un SMS à notre système pour demander la position du véhicule ou changer le mot de passe, d'où l'utilité d'utilisation d'un Smartphone.
- Le système embarqué dans le véhicule envoie périodiquement les positions à un serveur distant pour les afficher dans la plateforme qu'on a développée. Pour cela nous aurons besoin d'un pc.

3.4 Conception

Dans cette partie, nous allons choisir les composants matériels, les présenter en détail et voir leurs circuits d'interconnexion, puis on définira le fonctionnement du système logiciel qui permettra d'implémenter les différentes tâches vues lors de l'analyse.

3.4.1 Conception matérielle :

Il existe des équipements intelligents dit de technologies M2M, Ces équipements permettent l'échange d'informations et l'exécution d'opérations sans intervention humaine, dans notre cas nous allons utiliser un module GPS et un module GSM et qui présentent beaucoup d'avantages, notre choix porte alors sur le module SIM900 et module GPS UBLOX NEO6M. Le choix de ses composants est du à :

- Leur disponibilité sur le marché national.
- Leur Prix relativement faibles.
- Très large utilisation de ses composants dans le domaine des systèmes embarqués.

Nous utilisons aussi un microcontrôleur ATMEL de référence ATmega328 de la famille AVR, Celui-ci dispose :

- d'une bonne puissance de calcul
- de différentes interfaces séries (USART, SPI, USB, I2C).
- RAM suffisamment large pour éviter d'avoir recours à des mémoires externes.

Les composants (le microcontrôleur, le sim900 et le GPS) disposant tous les trois d'interfaces série de type USART, la communication est ainsi assurée par cette interface (Figure.3.5).



Figure.3.5 : Composants matériels utilisés.

3.4.1.1 Descriptions des Composants matériels utilisés:

3.4.1.1.1 Le module GPS : [16]

Le module GPS est le composant principal dans notre projet de suivi du véhicules, on l'utilise avec deux méthode : le tracking et la localisation sous demande .la première va identifier et remonter l'information à une période constante. Tandis que la localisation sous demande consiste à n'envoyer l'information qu'en cas de demande de l'utilisateur.

Le module Gy-NEO6MV2 illustré dans la figure 3.6 permet de connaître la position en temps réel. Il communique avec le microcontrôleur via le protocole UART. Le récepteur GPS est équipé de :

- EEPROM pour une sauvegarde permanente de la configuration.
- Une antenne acve de 25x25mm.
- Batterie rechargeable pour la sauvegarde.
- Port de communication UART (TTL).



Figure. 3.6: Le module GPS NEO-6MV2

Ce module fonctionne avec une alimentation 5 volts, et sa vitesse de transmission des données via l'interface série est de 9600 Baud. Le module GPS envoie les données relatives à la position en temps réel au format NMEA. Le format NMEA, comme montré dans

la Figure.3.7 est une chaîne de caractères composée de plusieurs mots séparés par des virgules.

```
$GPRMC,123519.000,A, 7791.0381,N, 06727.4434,E,022.4,084.4,230394,003.1,W*5A
$GPRMC,HHMMSS.SSS,A,latitude,N,longitude,E,speed,angle,date,MW,W,CMD
```

Figure. 3.7: Format NMEA.

La première ligne donne un exemple de chaîne de caractères composée de plusieurs mots séparés par des virgules et la seconde ligne de la figure.III.7 donne la signification de chaque mot.

Le tableau suivant décrit les champs les plus importants et dont on aura besoin pour notre projet:

Mot	Signification
\$GPRMC	Le nom de la chaîne
Speed	La vitesse KM/h
Latitude	La latitude du véhicule
Longitude	La longitude du véhicule
Date	La date de la réception

Communication avec la carte :

Le module GPS communique avec l'ARDUINO (Fig. 3.8) en série alors pour permettre la communication série sur les broches 3 et 4 de la carte arduino nous incluons la bibliothèque SoftwareSerial dans le programme.

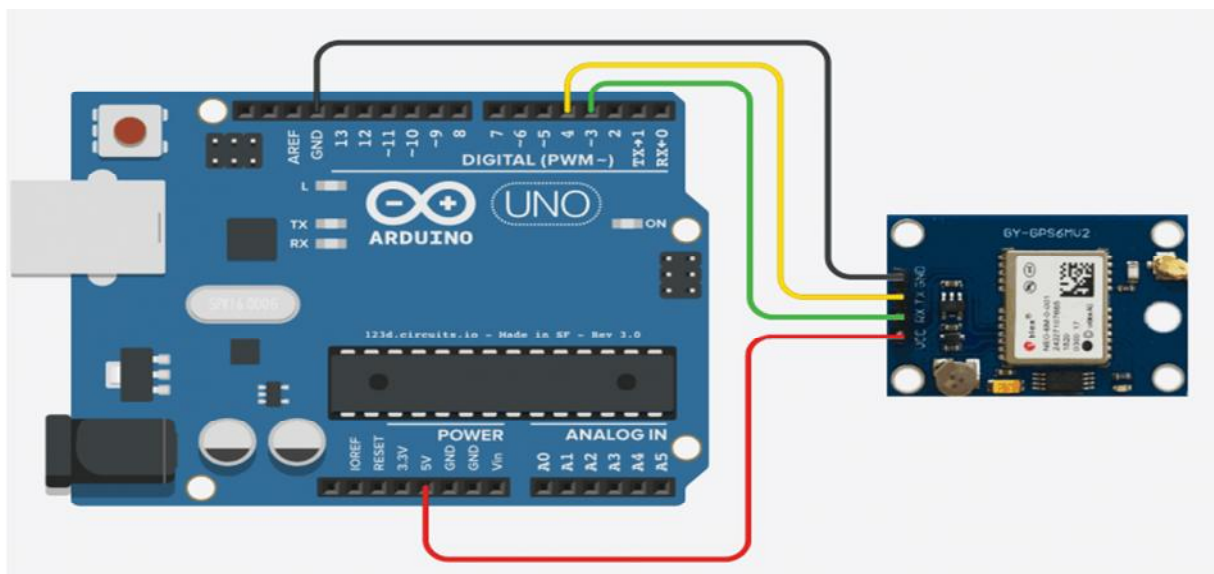


Figure. 3.8: Connexion entre le module GPS et la carte ARDUINO

3.4.1.1.2 Le module GSM SIM900 [17]:

Le module SIM900 de SIMCOM (Figure. 3.9) permet d'envoyer et recevoir des SMS, des données, des communications vocales depuis le réseau mobile et un accès direct à internet via le GPRS. Il est contrôlé via les commandes AT depuis une carte à microcontrôleur.



Figure 3.9: Carte SIM 900 GSM/GPRS shield

Les commandes AT:

Le microcontrôleur commande le module GSM en utilisant des commandes AT. Les commandes AT (ou commandes de Hayes) constituent un langage de commande développé afin de commander les modems, ce langage est développé à l'origine pour le modem Hayes Smart modem 300. Ce jeu de commandes s'est ensuite retrouvé dans tous les modems produits. Toutes les commandes doivent être précédées de "AT". Par exemple, ATA signifie "répondre". Certaines commandes, comme ATA, sont communes à tous les modems, d'autres sont variables selon le modèle spécifique, comme l'affichage de la configuration, et peuvent varier selon l'équipement. Il faut donc se reporter à la documentation de son modem.

3.4.1.1.3 La carte arduino :

Le modèle UNO de la société ARDUINO est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATmega328. L'ATmega328 est un microcontrôleur 8bits de la famille AVR dont la programmation peut être réalisée en langage C /C++.

La carte Arduino Uno est constituée de 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 6 broches d'entrées analogiques, d'une connectique USB, d'une connectique d'alimentation, d'un port ICSP et d'un bouton RESET. [18]

La description de toutes les connectiques de la carte Arduino UNO est présentée dans les deux images ci-dessous :

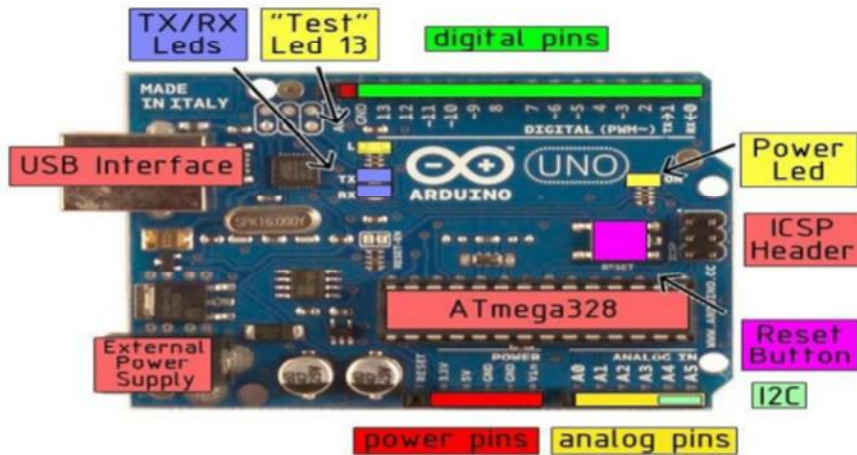


Figure 3.10 : Description des entrées/sorties de la carte Arduino Uno

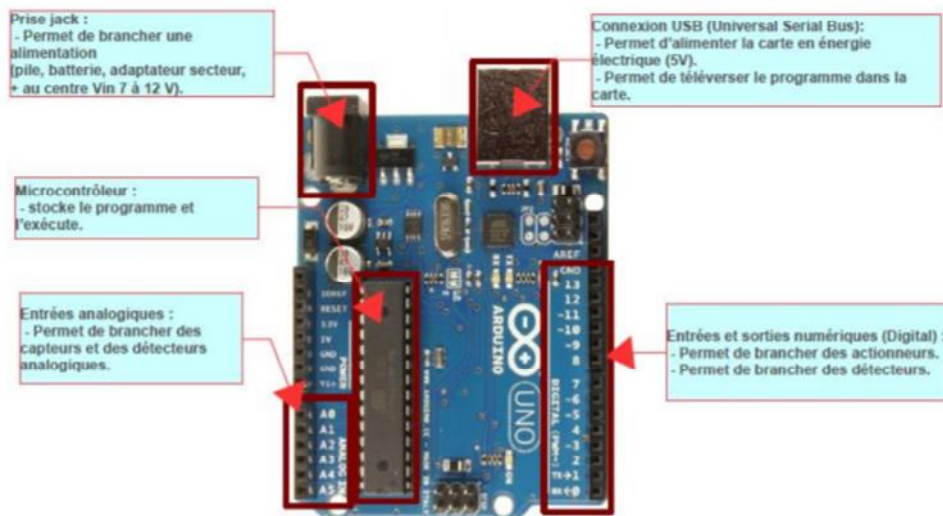


Figure 3.11 : Zoom sur la carte Arduino Uno

3.4.2 Conception Logicielle :

Comme on l'a souligné dans la partie Analyse, notre système est constitué de deux sous-systèmes :

- Sous-système embarqué dans un véhicule.
- Une plateforme de suivi de la flotte.

Le fonctionnement global de notre système est décrit par l'organigramme de la figure 3.12.

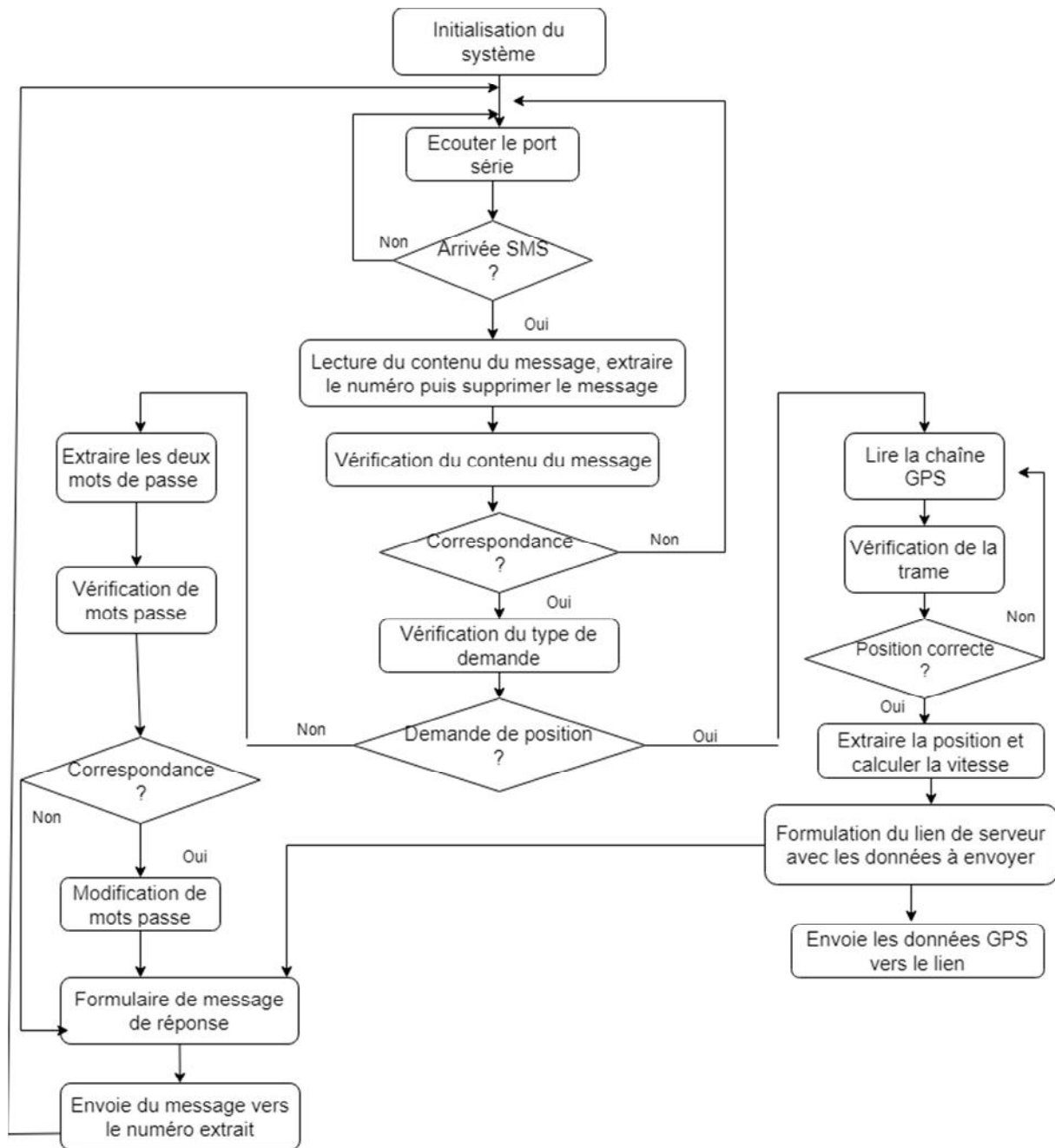


Figure 3.12 : organigramme global du système

Pour une description détaillée de notre système ,Nous allons utiliser trois diagrammes du langage UML(diagramme de cas d'utilisation,diagramme de séquence et diagramme de classe).

3.4.2.1 Comportement du système logiciel du Système embarqué dans le véhicule :

L'utilisateur du système peut visualiser la position de son véhicule et procéder à la configuration du système, il pourra changer par exemple son mot de passe, la figure 3.13 donne le diagramme des cas d'utilisation de notre système.

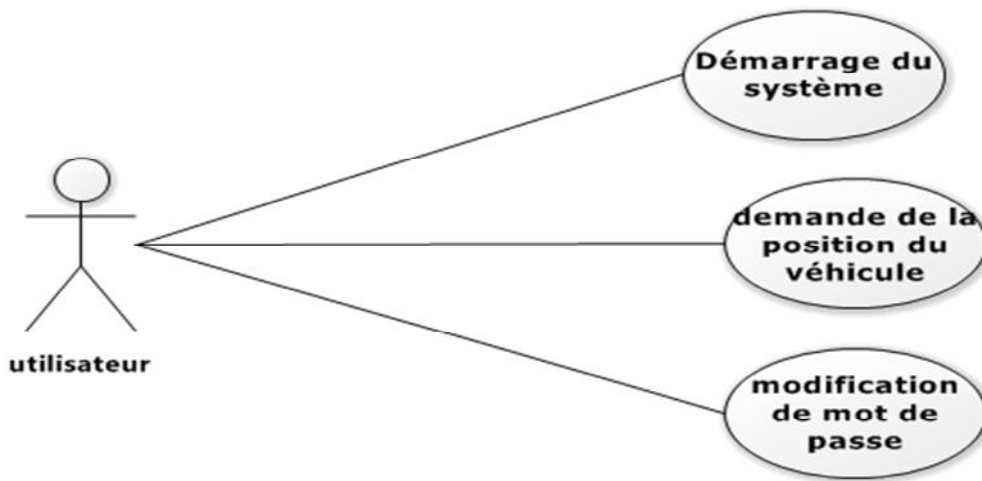


Figure 3.13 : diagramme des cas d'utilisation du système.

Afin de mieux structurer notre système nous allons procéder à son abstraction en regroupant les tâches commune dans un seul module, ainsi nous allons utiliser les modules suivants :

Initialisation : Ce module permet la configuration initiale des composants à utiliser.

Message : Ce module regroupe les procédures de traitement de message.

Motdepasse : Ce module comporte les procédures de modification de mot de passe.

Base : Ce module contient le point de lancement et terminaison de l'application, il fait la liaison entre les différents modules, et écoute l'arrivée des événements et leur contrôle.

Commande : Ce module disposera des différentes procédures permettant la Communication entre le microcontrôleur et les modules GPS et GSM via la liaison série

Position : Ce module contiendra toutes les procédures permettant de recevoir et traiter la trame NMEA depuis le SIM900.

3.4.2.1.1 Les diagrammes de séquence :

Les diagrammes suivants vont nous permettre de voir avec précision l'interaction entre les différents modules du système logiciel dans les différents cas d'utilisation définis précédemment :

- ***Le cas d'utilisation « démarrage du système »***

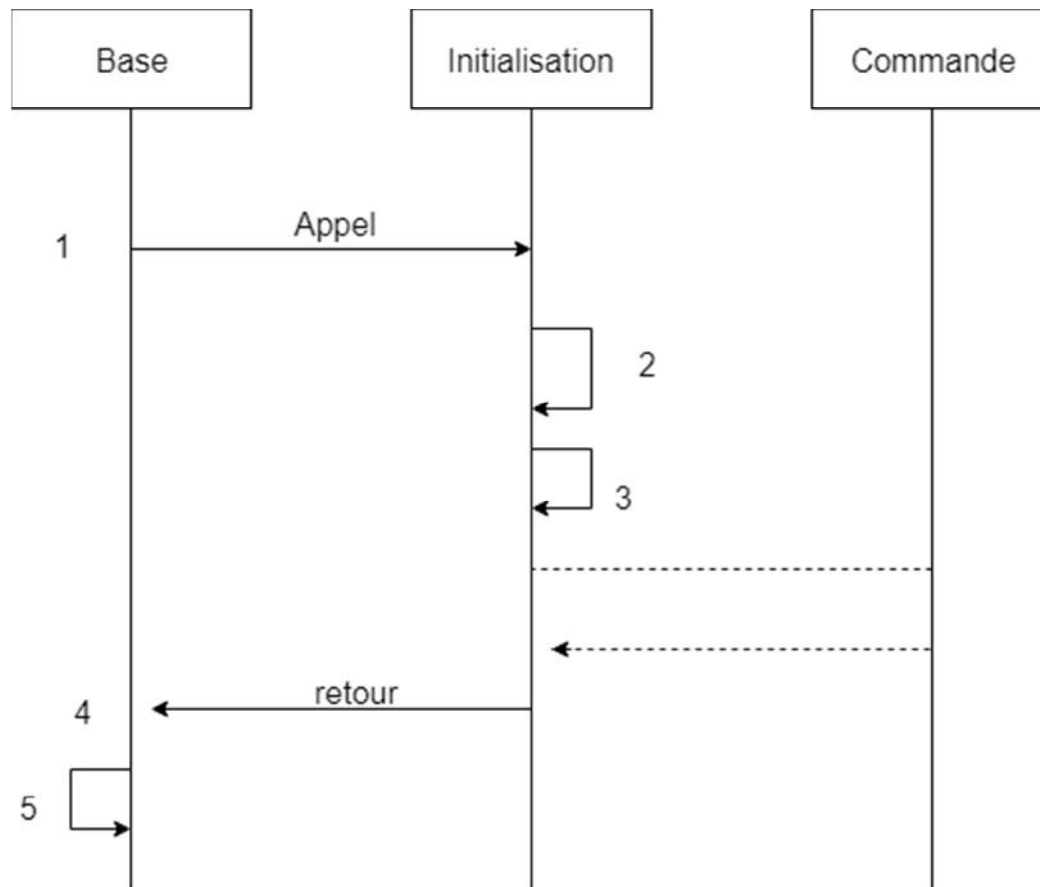


Figure.3.14 : Diagramme de séquence pour le cas « Démarrage du système »

Description :

- (1) Lors de la mise en marche du système, le module <<**base**>> fait appel au module <<**initialisation**>>.
- (2) Dans le module initialisation on fait appel à la procédure d'initialisation du GSM.
- (3) Toujours dans le module initialisation on fait appel à la procédure d'initialisation du GPS.
- (4) On revient au module <<**Base**>>.
- (5) Le programme principale scrute le port série et attend un événement si cet événement correspond a l'arrivée d'un message on appelle le module message sinon le programme boucle.

Plusieurs procédures de différents modules font appelle au module commande afin d'envoyer les commandes vers les modules SIM900 et GPS.

- **Le cas d'utilisation « demande de la position d'un véhicule »**

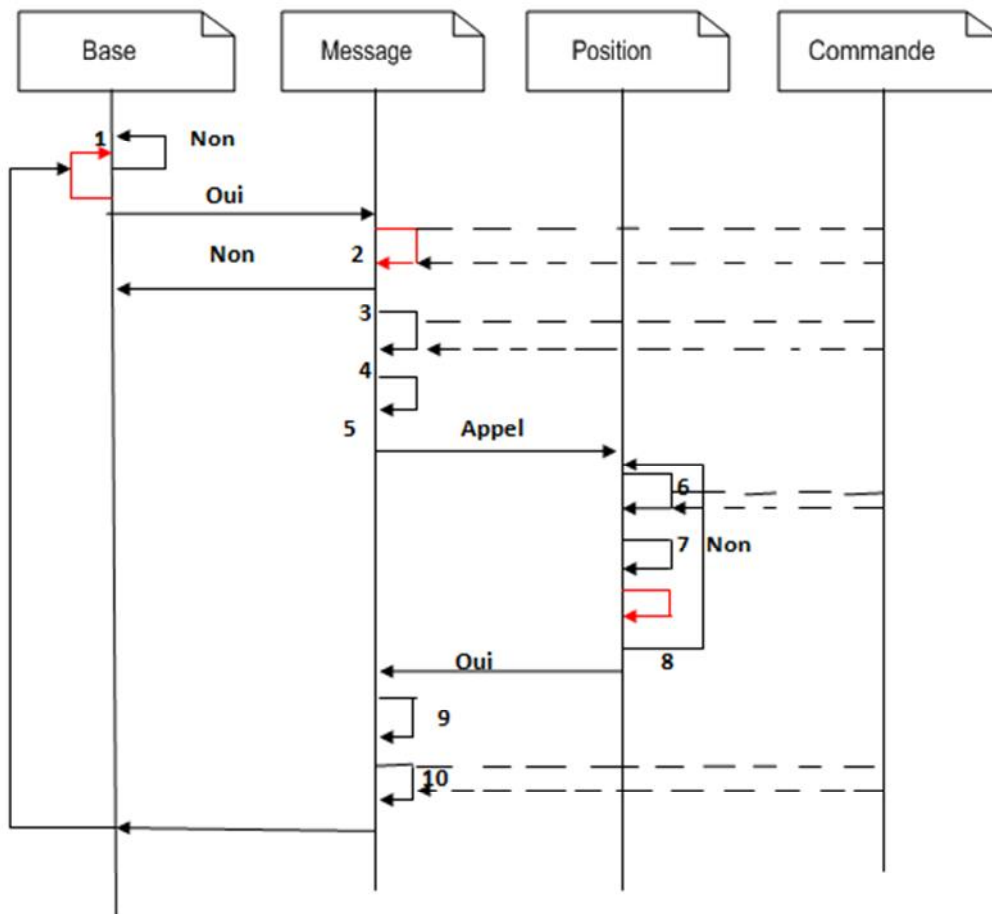


Figure.3.15 : Diagramme de séquence pour le cas « Demande de la position du véhicule »

Description :

- (1) Le module « **Base** » scrute l'arrivée d'un nouvel évènement, dès qu'il s'agit d'un message il fait appel au module « **Message** ».
- (2) On vérifie le contenu du message et on le compare à un mot de passe sauvegardé dans l'EEPROM de l'arduino si ils sont différents, on revient à (1).
- (3) On fait appel à une procédure pour en extraire le numéro de téléphone de l'émetteur et le sauvegardée en mémoire.
- (4) Le module <<**Message**>> supprime le message.
- (5) On fait appelle au module <<**Position** >>.
- (6) On appelle la procédure qui récupère la trame NMEA.
- (7) On extrait la latitude, la longitude et la vitesse et on les sauvegarde.
- (8) On vérifie la validité de ses données si elles ne sont pas valides on revient a (6), sinon on revient au module <<**Message**>>.
- (9) On construit le message de réponse à l'utilisateur.
- (10) On envoie ensuite le message construit vers le numéro de téléphone sauvegardé.

- Le cas d'utilisation « modification de mot de passe » :

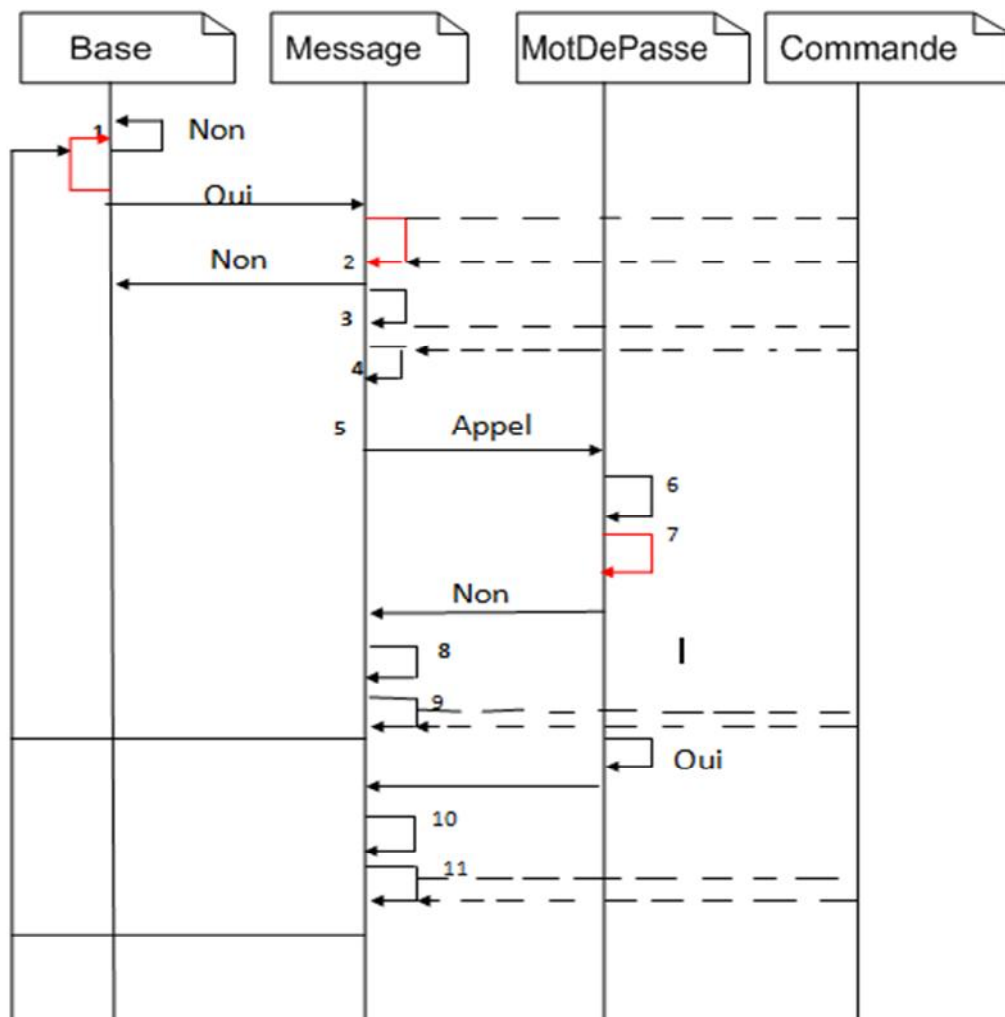


Figure.3.16 : Diagramme de séquence pour le cas «changement de mot de passe.»

Description :

(1)(3)(4) sont les mêmes étapes que le cas<<demande de position de véhicule>>.

(2) : On vérifie le contenu du message s'il s'agit d'une commande de modification de mot de passe, si non on revient à la scrutation.

(5) : S'il s'agit d'une commande de modification de mot de passe on fait appel au module <<MotDePasse>>.

(6) : On extrait l'ancien mot de passe et le nouveau mot de passe du contenu et on les sauvegarde.

(7) : une procédure de test vérifie l'exactitude de l'ancien mot de passe si il n'est pas correct on revient au module <<Message>>. Sinon on remplace l'ancien mot de passe par le nouveau

(8) : Construire le message d'erreur.

(10) Après modification du mot de passe, retour au module « **Message** » où on construit le Message de confirmation en réponse à l'utilisateur.

(9)(11) : On envoie ensuite le message construit vers le numéro de téléphone sauvegardé.

3.4.2.2 Comportement du système logiciel de la plateforme du suivi de véhicule :

Dans cette partie, on va décrire la conception de notre plateforme de suivi de véhicule (Figure 3.17), l'architecture web et l'architecture de communication.

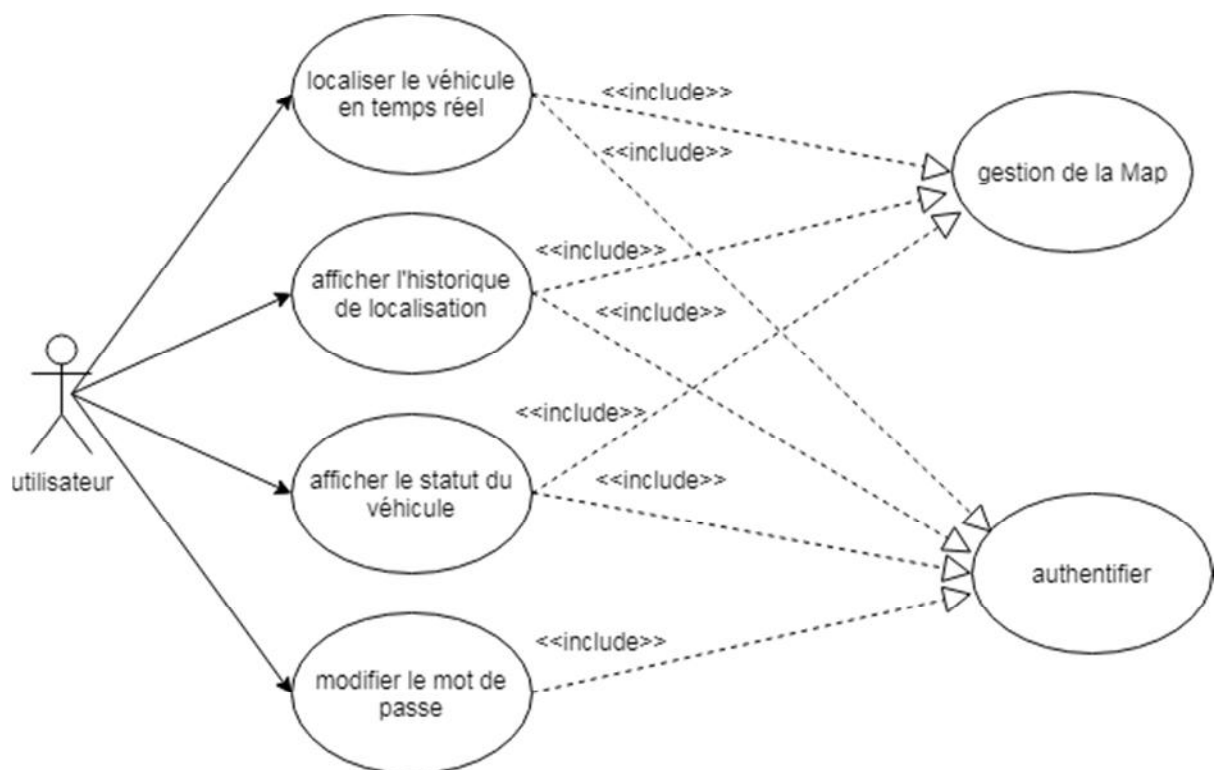


Figure 3.17 : Diagramme général des cas d'utilisation.

3.4.2.2.1 Diagramme de séquence :

Les diagrammes suivants vont nous permettre de voir avec précision les différents cas d'utilisation définis précédemment.

- **Les cas d'utilisation <la géolocalisation en temps réel et afficher le statut du véhicule >> :**

Pour la géolocalisation en temps réel, une fois que l'utilisateur est authentifié, il demande au système l'affichage de l'interface de géolocalisation. Le système répond par une liste de véhicule, L'utilisateur sélectionne un véhicule puis le système affiche une carte avec

un marqueur représentant le véhicule sélectionné et son statut marche<<avec un marker en couleur bleu>> ou arrêt<<avec un marker en couleur rouge>>

La Figure 3.18 présente le diagramme de séquence pour la géolocalisaon temps réel et affichage du statut d'un véhicule.

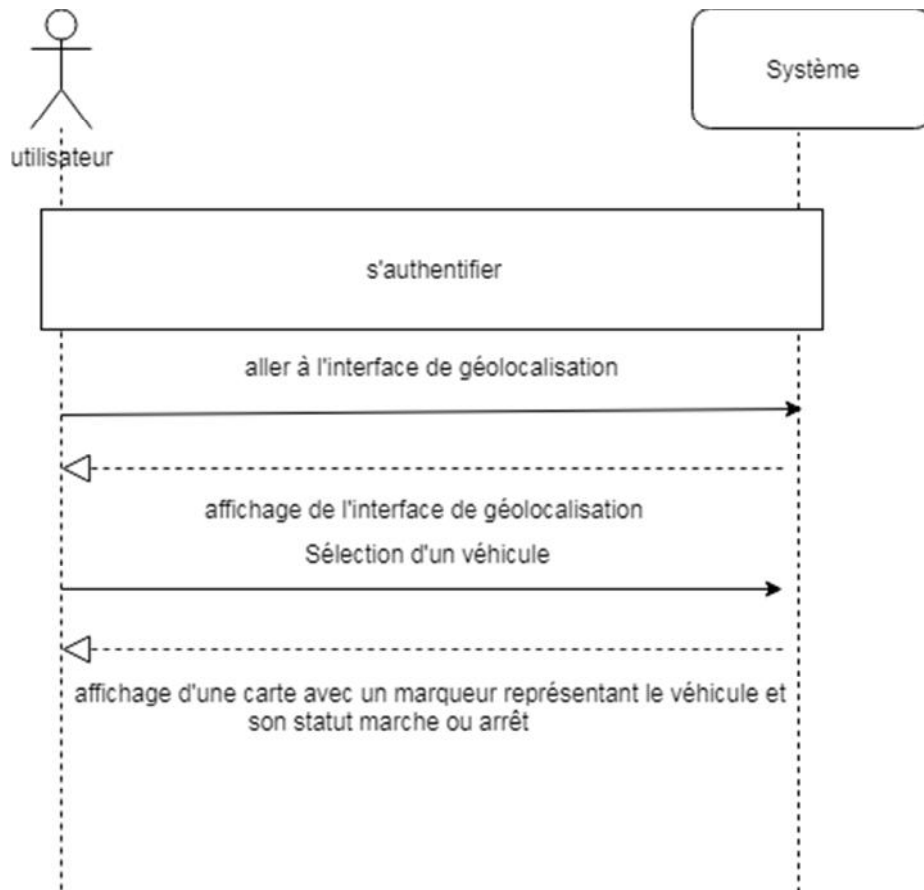


Figure 3.18 : Diagramme de séquence pour la géolocalisation temps réel et affichage du statut d'un véhicule.

- **Le cas d'utilisation <historique de localisation> :**

Une fois que l'utilisateur est authentifié. Il se présente devant l'interface de la géolocalisation puis il clique sur le véhicule et fixe un intervalle temporel pour la consultation de l'historique. la position du véhicule s'affiche sur une carte géographique.

La Figure 3.19 présente le diagramme de séquence pour consulter l'historique de localisation.

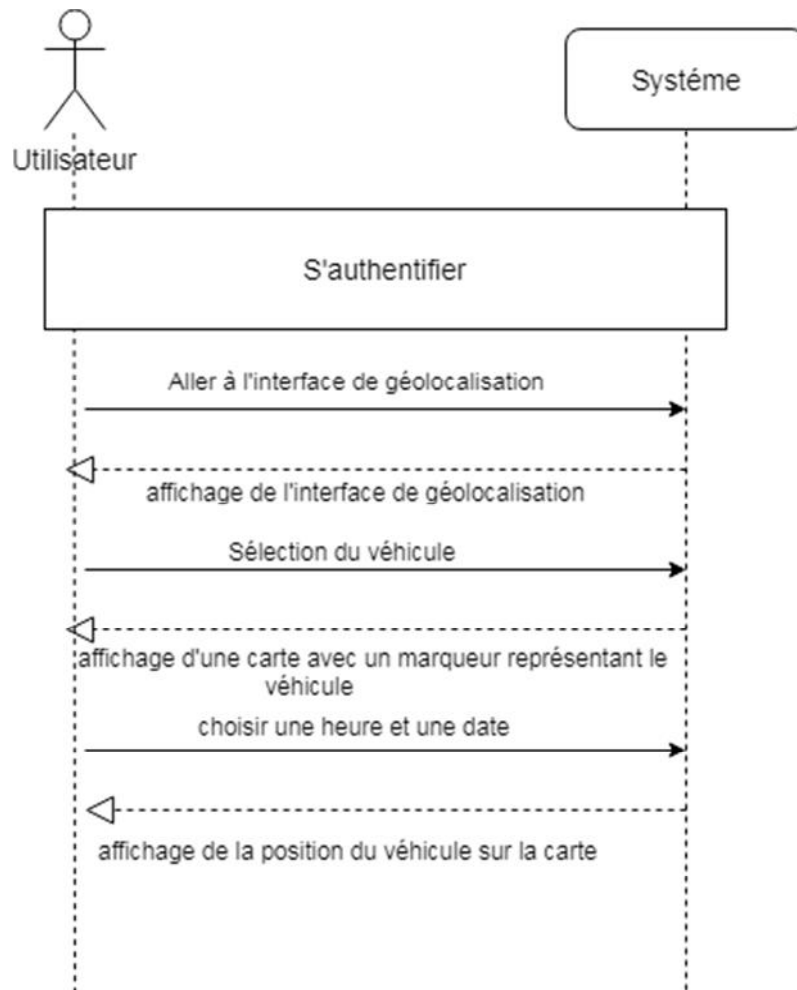


Figure 3.19. Diagramme de séquence pour consulter l'historique de localisation

3.4.2.2.2 L'architecture web :

Notre applicaon web est bi en évi dement une archi tecture à 3 niveaux classiques comprenant :

L'administrateur : accédant à l'application depuis son navigateur et aux fonctionnalités de l'application au travers d'une interface administrateur.

Un serveur de données : qui va garantir la persistance des données sensibles ou importantes et permettre aux clients d'utiliser l'application de façon optimale.

Un serveur web : dont le rôle est d'assurer l'affichage et le rafraîchissement de l'interface utilisateur, de présenter et d'assurer le bon fonctionnement des fonctionnalités de l'application mais également de faire le lien entre le client et le serveur de données en assurant l'échange de données entre ces deux entités.

3.4.2.2.3 L'architecture de communication :

L'architecture de communication est également une architecture réseau à trois niveaux faisant intervenir quasiment les mêmes entités que dans l'architecture web.

Seul le serveur d'application diffère de l'architecture précédente et se compose d'un serveur WebSocket assurant une communication full-duplex (bidirectionnelle) entre le navigateur et le serveur web

Cela signifie que les utilisateurs de notre site peuvent obtenir des mises à jour sur les événements importants se produisant sur le serveur sans le demander explicitement.

Les descriptions ci-dessus nous permettent d'élaborer un modèle de fonctionnement général de l'application, décrit par la Figure 3.20.

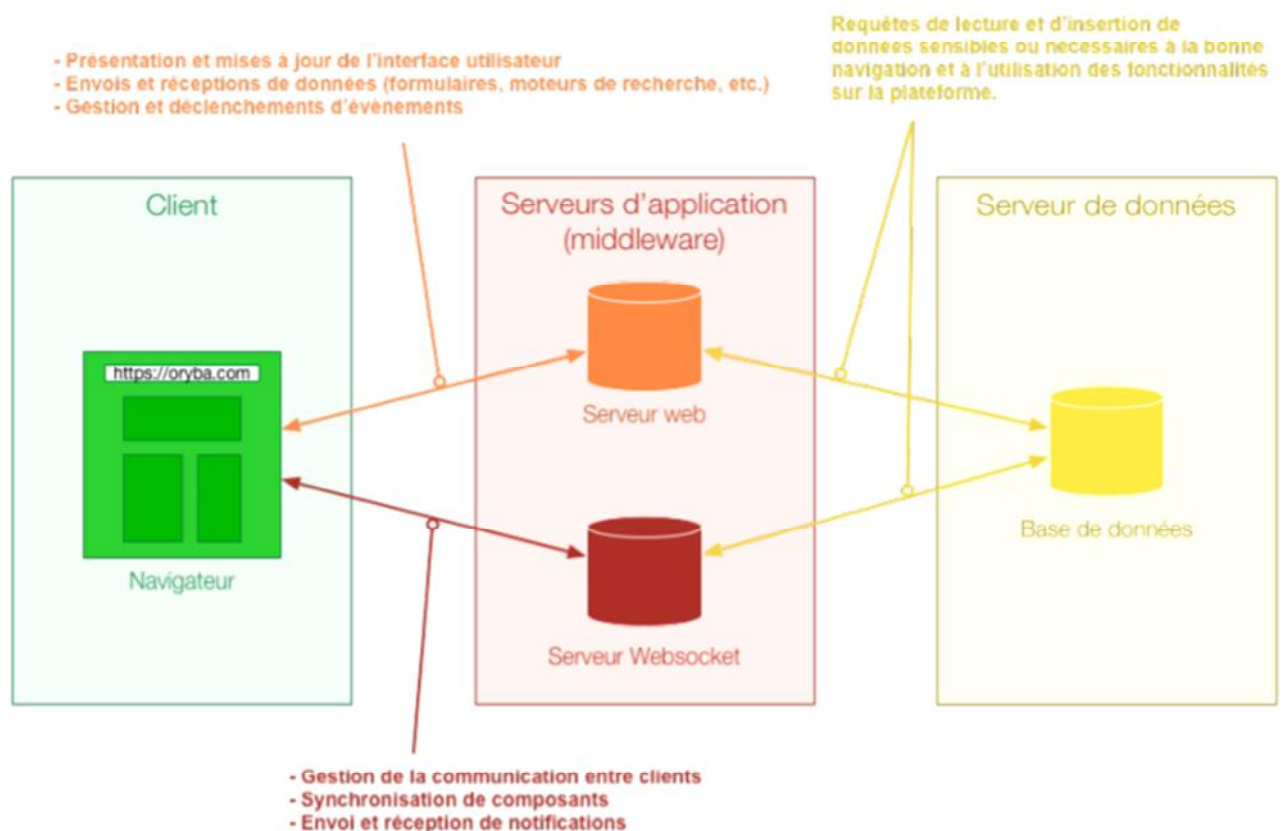


Figure. 3.20: L'architecture de communication.

3.4.2.2.4 Le niveau donné :

Schéma de la base de données :

Le schéma conceptuel de la base de données est représenté par le diagramme de la figure 3.21:

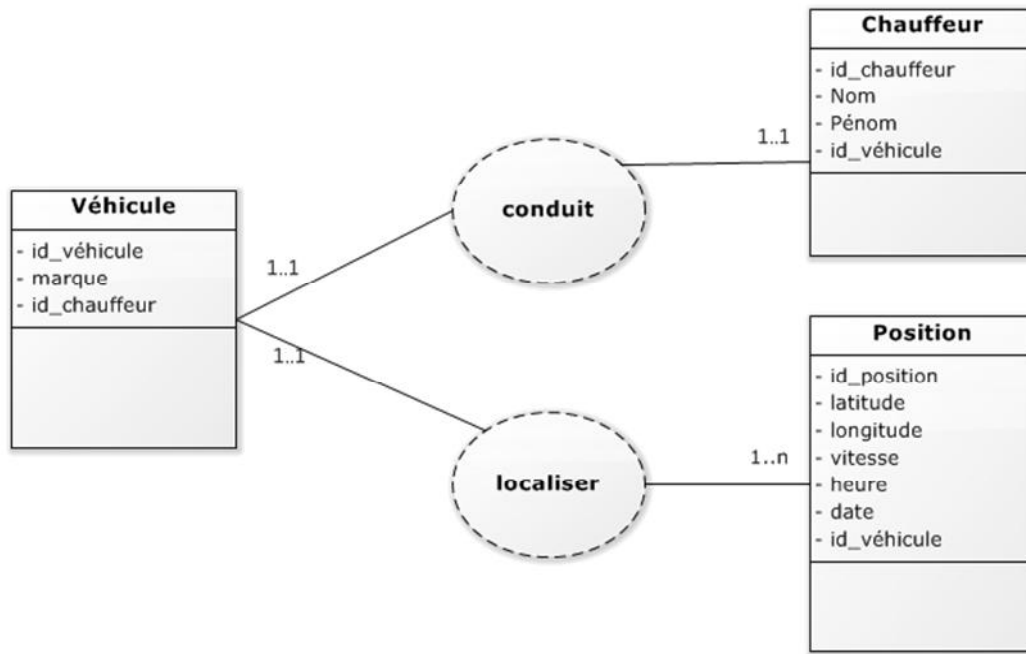


Figure 3.21: Le schéma de la base de données.

Notre base de donnée est constitué de 4 tables que nous allons détailler par la suite. Ces quatre Tables son liées entre elles avec les relations suivantes :
Un et un seul véhicule peut être localisé par une position à la fois.

Les tables :

L'application dans tout ces cas d'utilisation manipule les données stockées dans la Base De Données, et elle contient les tables suivantes :
Les clés primaires sont soulignées, les clés étrangères sont précédées par #.

Véhicule (id_véhicule, Marque, Immatriculation, #id_chauffeur).

Chauffeur (id_chauffeur, Nom, prénom, #id_véhicule).

Position (id_position, latitude, longitude, vitesse, heure, date, #id_véhicule).

3.4.2.3 Algorithme de quelques procédures :

Nous avons utilisé les commandes at pour l'envoi et la réception des SMS, et nous les avons utilisé également pour l'envoi des données à notre serveur.

- **La procédure Traitement trame ():**

La procédure sert à calculer la latitude, la longitude et la vitesse et Cela à partir de la trame reçues. Pour limiter la consommation de ressources, nous allons utiliser une bibliothèque GPS qui permet d'éviter toute dépendance obligatoire en virgule flottante et ignore presque tous les champs GPS clés.

Procédure Traitement trame ():**Début**

Traiter la trame en provenance du GPS.

Faire des tests de contrôle.

Extraire la position et calculer la vitesse.

Fin.

- **La procédure Envoie chaîne(chaine de caractère)**

La procédure Envoie_chaine(chaine de caractère)**Début**

Envoyer la chaine de caractère au TXD du port série.

Fin.

- **La procédure Envoie Message()**

La procédure Envoie_Message()**Début**

Sélectionner le format du message en mode texte.

Commande at qui permet la lecture du message.

Extraction du numéro de téléphone à contacter

Envoyer le message.

Fin.

- **La procédure modifier mot de passe ()**

Procédure modifier_mot_de_passe ()

nouveau_mot_de_passe: chaine de caractères//récupérer le nouveau mot de passe saisi.

ancien_mot_de_passe :chaine de caractères//récupérer l'ancien mot de passe saisi .

initial_mot_de_passe: chaine de caractères//récupérer le mot de passe initialisé sur la

EEPROM de l'arduino.

Début

Pour(i=0 ;i<taille du tableau ;i++)faire

```
    If (ancien_mot_de_passe[i]== initial_mot_de_passe[i]){  
  
    initial_mot_de_passe[i]<= nouveau_mot_de_passe[i] ;  
  
    EEPROM <=intial_mot_de_passe[i] ;  
  
    }  
  
    fait  
  
Fin
```

- **La procédure Envoie_Message_au_serveur()**

La procédure Envoie_Message_au_serveur()

Début

Activer le module SIM900 en mode GPRS.

Faire communiquer le module GSM avec l'opérateur,

Démarrer le service HTTP.

Envoyer les coordonnées au serveur.

Terminer la connexion.

Fin.

3.5 Conclusion :

Dans ce chapitre, nous avons modélisé les deux parties de notre système. On a choisi les composants matériels nécessaires et déterminé les différents modules logiciels qui vont assurer le bon fonctionnement du système. Suite au travail accompli au niveau de cette partie, il reste les étapes d'implémentation et des tests qui vont faire l'objet du prochain chapitre.

Chapitre 4

Réalisation du système

4.1 Introduction :

Dans le chapitre précédent, nous avons présenté les différentes étapes de conception et d'analyse. L'étape suivante est consacrée à la réalisation, où nous détaillerons l'environnement matériel et logiciel de notre travail, ainsi que la mise en œuvre et les résultats des tests effectués, et enfin nous citerons des perspectives.

Nous avons travaillé sur une carte arduino uno, nous avons utilisé le module GPS neo6mv2 et un module GSM SIM900 de SIMCOM.

L'application permettra de suivre le véhicule par géolocalisation, le GPS placé dans la voiture permettra d'avoir les coordonnées de son emplacement et le GSM permettra de les envoyer par SMS en cas de demande et également les envoyer périodiquement à un serveur distant. L'utilisateur pourra suivre l'historique de localisation de son véhicule ainsi que sa vitesse de déplacement.

4.2 Outils de développement utilisés :

4.2.1 Outils logiciels :

Pour le développement de notre application, nous avons été amenées à utiliser une multitude d'outils et langages de programmation que nous présentons dans ce qui suit.

4.2.1.1 Logiciel arduino:

Le logiciel arduino présenté dans la figure 4.1 est un environnement de développement (IDE) open source et gratuit, téléchargeable sur le site officiel arduino . [18]
On l'a utilisé pour le développement du code embarqué.



Figure 4.1: logiciel de programmation de la carte ARDUINO

L'IDE Arduino présenté dans la figure 4.2 permet :

- ✓ d'**éditer un programme** : des croquis (*sketch* en Anglais),
- ✓ de **compiler ce programme** dans le langage « machine » de l'Arduino,
- ✓ de **téléverser** le programme dans la mémoire de l'Arduino.
- ✓ de **communiquer** avec la carte Arduino grâce au terminal.

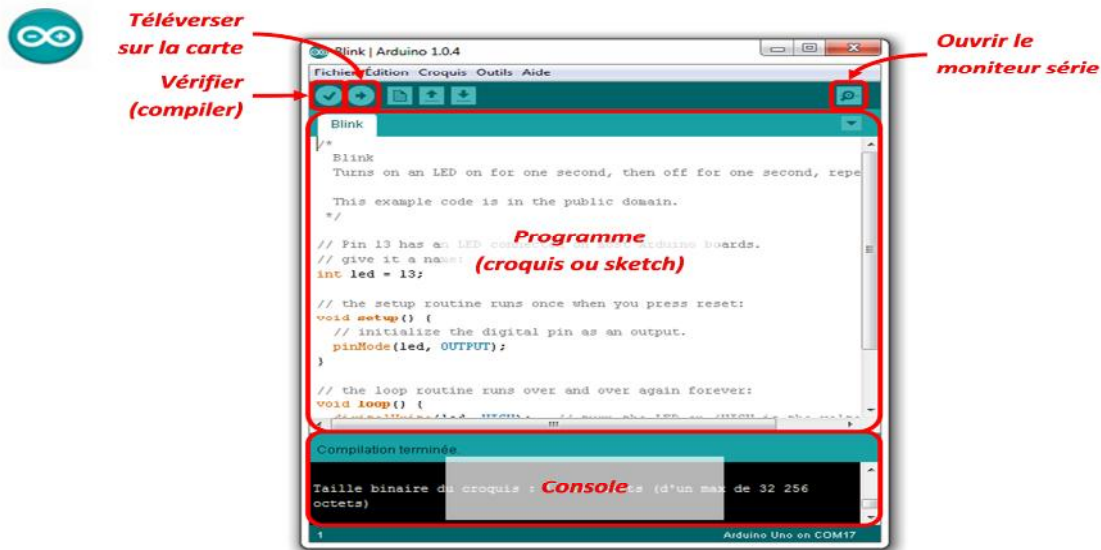


Figure 4.2 : Vue d'ensemble du logiciel Arduino.

4.2.1.2 Framework Laravel :

Laravel est un framework PHP qui propose des outils pour construire une application web. Le créateur de Laravel, Taylor Otwell, a simplement regroupé les meilleures bibliothèques pour chaque fonctionnalité nécessaire pour la création d'un site web. Il a aussi rajouté des fonctionnalités exclusives, comme le moteur de template Blade. Dans notre application on a utilisé Laravel 5. [19]

4.2.1.3 Template bracket responsive bootstrap :

Il offre une conception d'interface utilisateur simple et moderne, facile à utiliser, ainsi qu'une présentation entièrement réactive. On l'a utilisé pour créer notre site web. La figure 4.3 présente l'interface du template qu'on a utilisé.

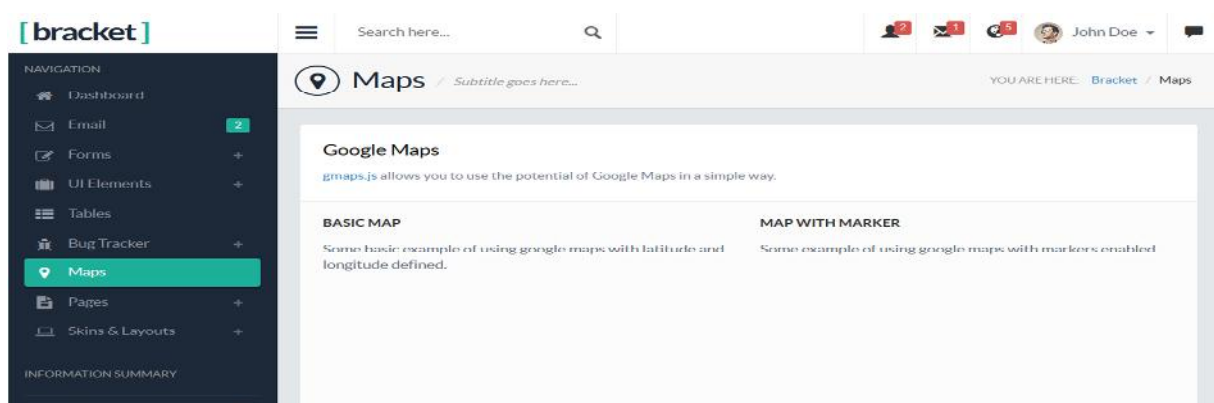


Figure 4.3 : L'interface de template bracket.

4.2.1.4 Serveur XAMPP:

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur Web local, un serveur FTP et un serveur de messagerie électronique.

Il s'agit d'une distribution de logiciels libres (**X** (cross) **A**pache **M**ariaDB **P**erl **P**HP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus [20]. Nous avons utilisé xampp car il contient la plus part des outils nécessaires pour reproduire le comportement d'un serveur web. La figure 4.4 présente l'interface du serveur xampp.



Figure 4.4 : L'interface de xampp.

4.2.1.5 Sublime_text: [20]

Sublime Text (figure 4.5) est un éditeur de texte générique codé en C++ et Python, disponible sur Windows, Mac et Linux. On l'a utilisé pour le développement du code de la plateforme web.

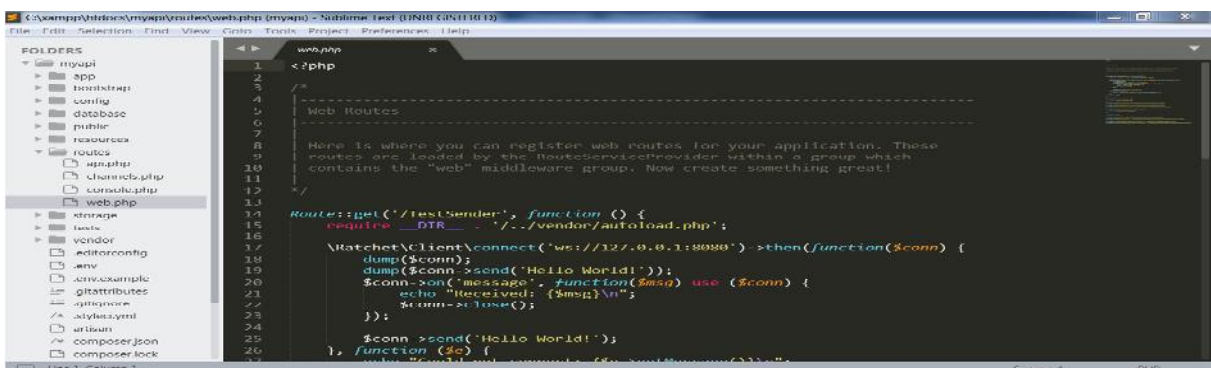


Figure 4.5: L'interface de sublime text.

4.2.1.6 Leaflet:

Très souvent utilisé pour remplacer les API de Google Maps, leaflet est une solution plus récente, mais déjà bien réputée et bien appréciée pour sa simplicité. Cette bibliothèque JavaScript Open Source a été pensée pour des cartes interactives et mobiles. [21]

4.2.1.7 Ngrok:

C'est un logiciel couplé à un service web qui va vous permettre de créer un tunnel à partir d'Internet vers un port de notre choix sur la machine en local (TCP ou UDP). Il suffit de l'installer (Mac, Linux, Windows), Le service fournit une URL publique qu'on avait utilisé pour envoyer les coordonnées GPS à partir du GSM à notre serveur.

4.2.2 Les Langages de programmation :

- ✓ **Language c** : est un langage de programmation impératif généraliste, de bas niveau. Inventé au début des années 1970 pour réécrire UNIX, C est devenu un des langages les plus utilisés, encore de nos jours. De nombreux langages plus modernes comme C++, C#, Java et PHP ou Javascript ont repris une syntaxe similaire au C et reprennent en partie sa logique. C offre au développeur une marge de contrôle importante sur la machine (notamment sur la gestion de la mémoire) et est de ce fait utilisé pour réaliser les « fondations » (compilateurs, interpréteurs...) de ces langages plus modernes. [20]
- ✓ **Php** : est un langage utilisé dans des applications web pour écrire des scripts html le but premier de ce langage est de permettre aux développeurs web de concevoir rapidement des sites aux pages dynamiques .c'est un langage exécuté cote serveur, à la différence de JavaScript et html, donc le code de l'utilisateur n'apparaît pas dans la source de sa page web, seul les informations qu'il souhaite afficher seront visible par les visiteurs.
- ✓ **Javascript** : JavaScript est un langage de script développé par Netscape. Les scripts intégrés aux balises HTML d'une page web, s'exécutent dans le navigateur web sans faire appel aux ressources du serveur permettant d'effectuer des contrôles de saisie pour valider les formulaires, ouvrir ou fermer des fenêtres ou encore de gérer des éléments graphiques.

4.2.3 Outils matériels :

La réalisation et la mise en œuvre de circuit de notre application nécessite les composants matériels suivants :

- **carte arduino** : le microcontrôleur qui a servi comme unité de contrôle du Système.
- **Le SIM900** : le module GSM qui nous a permis d'envoyer et recevoir des SMS
- **Le GPS Neo6mv2** : le module GPS qui permet de capter la position du véhicule.
- **Le lab d'essai** : le lab d'essai nous a permis de réaliser et de tester l'interfaçage de nos modules au niveau matériel.
- **L'alimentation stabilisée et régulée** : l'alimentation stabilisée et régulée a été utilisée pour répondre aux besoins des diverses tensions (9V).
- **Un pc** : le pc a été utilisé pour visualiser les positions envoyées périodiquement par le système embarqué dans le véhicule.
- **Un Smartphone** : il nous a permis d'envoyer des SMS à notre système pour demander la position du véhicule ou changer le mot de passe.

4.2.4 Implémentation des fonctionnalités et technologies utilisées :

Dans le chapitre précédent, nous avons décrit les différents choix qui ont été faits autour de la conception de l'architecture générale de la plateforme de suivi du véhicule.

Nous allons détailler dans la partie suivante la technologie utilisée pour élaborer les différentes fonctions.

4.2.4.1 La partie web :

➤ Le serveur web et la base de données :

Comme énoncé précédemment, l'architecture web est composée d'un serveur web et d'un serveur de données. Nous avons donc choisi d'utiliser :

- **Un serveur de base de données relationnelle MySQL** pour son faible coût, sa rapidité et sa simplicité.
- **Un serveur web Apache** pour les requêtes web et le lien avec la base de données.

➤ Laravel un framework pour un site dynamique :

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. Laravel est distribué sous licence MIT, avec ses sources hébergées sur GitHub. [20]

Dans une application Laravel à usage général, l'architecture typique est la suivante. Sur la base du diagramme d'architecture illustré dans la figure 4.6, on peut constater que chaque demande d'un client est traitée en fonction de la configuration de sa route.

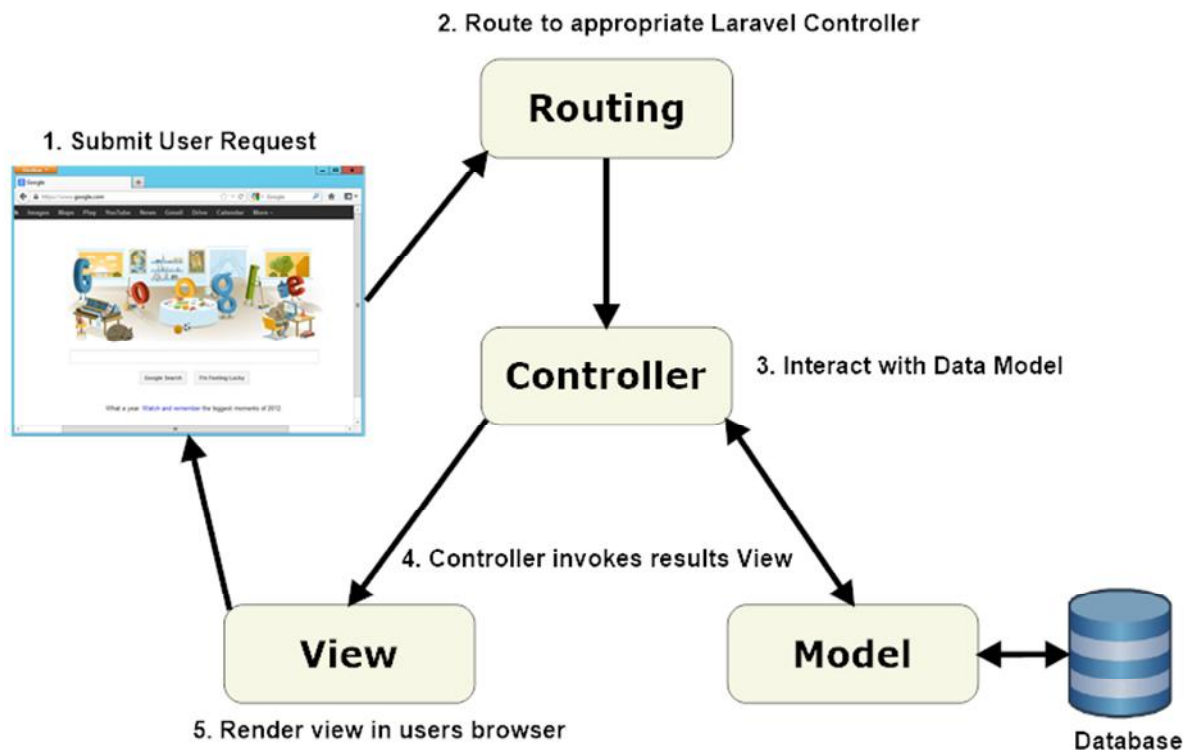


Figure 4.6 : diagramme d'architecture du laravel.

La seule différence majeure entre l'architecture de laravel et une autre application typique est son routage, d'autres composants tels que Controller, View et Model ont le même concept que n'importe quelle application typique. On présente ci-dessous les composants du laravel :

Contrôleur : Un contrôleur est responsable de la définition de la logique d'application (traitement de la demande). Les contrôleurs sont généralement stockés dans le répertoire `app / Http / Controllers`.

Modèle : le modèle correspond à une table d'une base de données. C'est une classe qui étend la classe **Model** qui permet une gestion simple et efficace des manipulations de données et l'établissement automatisé de relations entre tables.

Vue : la vue est soit un simple fichier avec du code HTML, soit un fichier utilisant le système de template **Blade** de Laravel.

Routing : Le routage joue le rôle le plus important dans le rendu des applications laravel. Toute URL demandée à l'utilisateur doit être soumise à des itinéraires spécifiques. Toutes les routes de Laravel sont définies dans le fichier routes.php. L'emplacement de routes.php est défini sous `app \ Providers \ RouteServiceProvider.php`

4.2.4.2 Le serveur de communication :

➤ **Web en temps réel avec les WebSockets: [22]**

Afin de concevoir une application permettant d'afficher en temps réel les positions du véhicule envoyé par le système embarqué, nous avons dû nous pencher sur la conception d'un serveur de communication capable d'établir une connexion fiable et efficace entre l'API qui permet de récupérer les positions de la voiture et notre serveur, nous avons utilisé un serveur **RATCHET** qui implémentent le protocole websocket qui apporte une solution pour l'établissement d'un canal de communication full-duplex (bidirectionnel) basé sur une connexion TCP (Transport Control Protocol) permettant à des clients de communiquer grâce à leur navigateur avec le serveur mais aussi au serveur d'envoyer des messages ou notifications aux clients sans pour autant que ces clients n'aient à envoyer une nouvelle requête d'actualisation au serveur.

Le protocole WebSocket a été normalisé en 2011 par l'IETF (Internet Engineering Task Force) dans la RFC 6455 définissant la mise en œuvre et le fonctionnement du protocole. Ainsi, le protocole comprend deux phases :

- La phase de « Handshake » pour initialiser une connexion WebSocket entre un client et un serveur et créer le canal de communication entre ces entités.
- L'envoi de données entre le client et le serveur à travers le canal de communication précédemment créé.

❖ Avantage des WebSockets :

Les WebSockets ont les avantages suivants :

- Nécessitent un faible délai de transit (*latency*) : pas de nouvelle connexion TCP à créer comme c'était le cas par requête HTTP

- N'ont besoin que d'un petit surdébit (*overhead*) : seulement 2 octets par message pour un message texte, au lieu des centaines parfois nécessaires dans les en-têtes HTTP
- Résultent en une diminution de trafic : puisque les clients n'ont plus besoin de faire de scrutation (*polling*), les messages sont envoyés seulement lorsqu'il y a des données

En fait, un des gros problèmes résolu avec les WebSockets est que le serveur puisse initier des envois. De plus, l'idée de la spécification est qu'à terme elle soit implantée de façon native dans tous les navigateurs modernes, rendant ainsi la technologie du temps réel accessible technologiquement et efficace.

❖ Serveur RATCHET :

Ratchet est une bibliothèque PHP faiblement couplée offrant aux développeurs des outils pour créer des applications bidirectionnelles en temps réel entre les clients et les serveurs via WebSockets.

Avec le protocole http habituel, chaque page Web chargée côté client lancera un nouveau script PHP sur le serveur, chargera des ressources (telles qu'une connexion à une base de données), exécutera le code, fermera les ressources et renverra la sortie (HTML) au client, et fermera la connexion.

WebSockets et l'utilisation de PHP dans Ratchet sont donc un peu différentes. Un seul script est exécuté et les connexions restent ouvertes. [22]

Le diagramme ci-dessous illustre l'ordre d'exécution commençant par un client demandant une page Web sur un site:

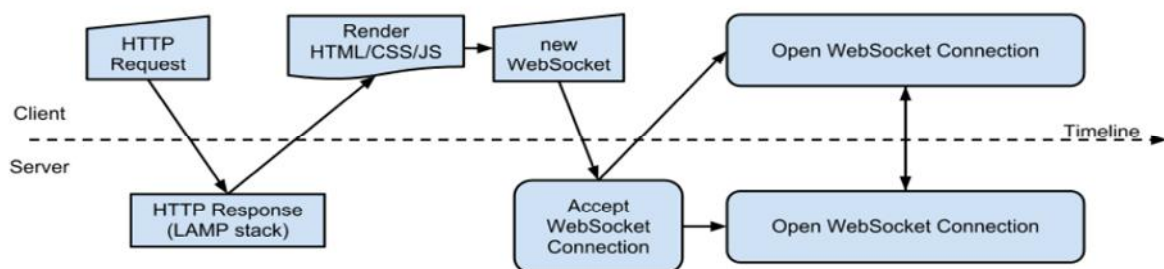


Figure 4.7 : l'ordre d'exécution.

Comme on peut le constater, une fois la page Web chargée, une connexion WebSocket est rétablie dans l'application Ratchet. Si tout se passe bien, une connexion reste ouverte et le serveur ou le client peut envoyer des données à l'autre à tout moment.

4.3 Réalisation et Tests:

4.3.1 Réalisation du système embarqué :

4.3.1.1 *Explication du circuit de notre système :*

Le module GPS fonctionne avec une alimentation 5 Volt, il communique avec la carte arduino en série. Pour permettre la communication série sur les broches 2 et 3 de la carte, nous avons inclus la bibliothèque <<software série>> dans le programme pour créer un port série logiciel vu que la carte arduino uno dispose d'un seul port hardware qui est utilisé pour communiquer avec le module SIM900. Le tableau de la figure 4.8 montre la connexion entre le module GPS et la carte ARDUINO.

Branchement :

Module GPS NEO-6M	Câblage vers Arduino UNO
VCC	5V
RX	D3
TX	D 2
GND	GND

Figure 4.8 : Connexion entre le module GPS et la carte ARDUINO

Le module GSM est également alimenté par 9 volts et sa vitesse de transmission des données avec la carte ARDUINO est de 19200 Baud. Il communique avec la carte arduino en série via le port hardware de la carte (tx, rx). Le tableau de la figure 4.9 montre la connexion entre module GSM et la carte ARDUINO.

Branchement :

Module GSM	Câblage vers Arduino UNO
VCC	5V
RX	TX
TX	RX
GND	GND

Figure 4.9 : Connexion entre le module GSM et la carte ARDUINO

Arduino est utilisé pour contrôler l'ensemble du processus avec un module récepteur GPS et GSM. Le circuit de la figure 4.10 montre le circuit final de notre système il est réalisé à l'aide du logiciel ISIS Proteus.

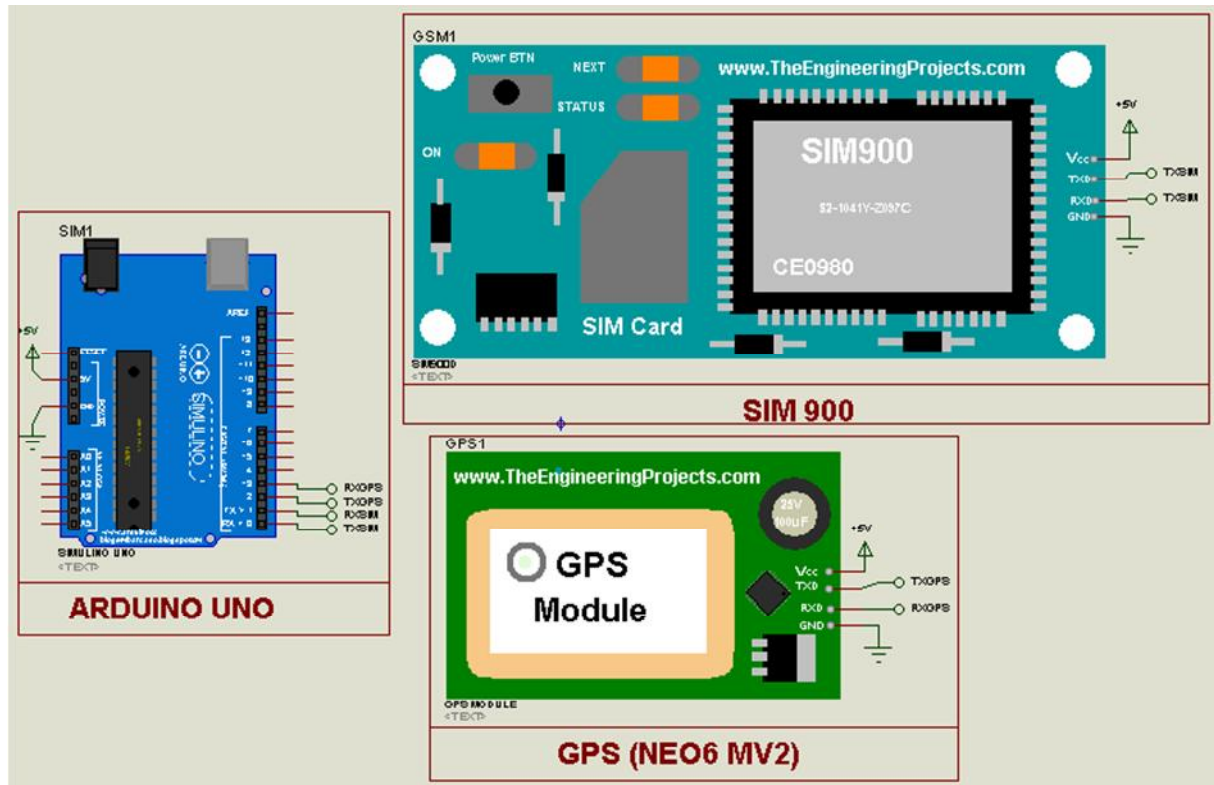


Figure 4.10 : le circuit final de notre système.

4.3.1.2 La programmation :

- ❖ On a inclut la bibliothèque de communication série software pour permettre la communication série sur les broche (2,3) de la carte ARDUINO avec le module GPS.

```
#include <SoftwareSerial.h>

SoftwareSerial ss(2,3); // RX, TX
```

- ❖ On a définis la vitesse de transmission à 9600 avec le GPS et à 19200 avec le GSM.

```
Serial.begin(19200);

ss.begin(9600);
```

- ❖ On a inclus la bibliothèque EEPROM.h, elle nous a permet de persister un mot de passe dans l'EEPROM de l'arduino.

```
#include<EEPROM.h>
```

- ❖ Pour interagir avec le module SIM900, nous avons utilisé des commandes AT, les commandes ci-dessous nous ont permis de définir le mode texte et lire les SMS.

```
Serial.print("AT+CMGF=1\r");
```

```
Serial.print("AT+CNMI=2,2,0,0,0\r");
```

- ❖ Afin d'envoyer la position par SMS, voici une petite partie du code qui nous a permis d'extraire le numéro de téléphone.

```
if(data == '+'){RcvdCheck = 1;}
if((data == 'C') && (RcvdCheck == 1)) {RcvdCheck = 2;}
if((data == 'M') && (RcvdCheck == 2)) {RcvdCheck = 3;}
if((data == 'T') && (RcvdCheck == 3)) {RcvdCheck = 4;}
```

- ❖ La commande AT ci-dessous sert à activer le mode GPS du module SIM 900.

```
Serial.println("AT+CGATT=1");
delay(100);
```

- ❖ La commande AT ci-dessous sert à initialiser le service http.

```
Serial.println("AT+HTTPINIT");
delay(2000);
```

4.3.2 Tests:

4.3.2.1 Tests unitaires :

4.3.2.1.1 Test du module GPS :

Nous avons testé le module GPS en demandant la trame NMEA, nous avons utilisé la bibliothèque <<TinyGPS++.h> le résultat est présenté sur la figure 4.11.

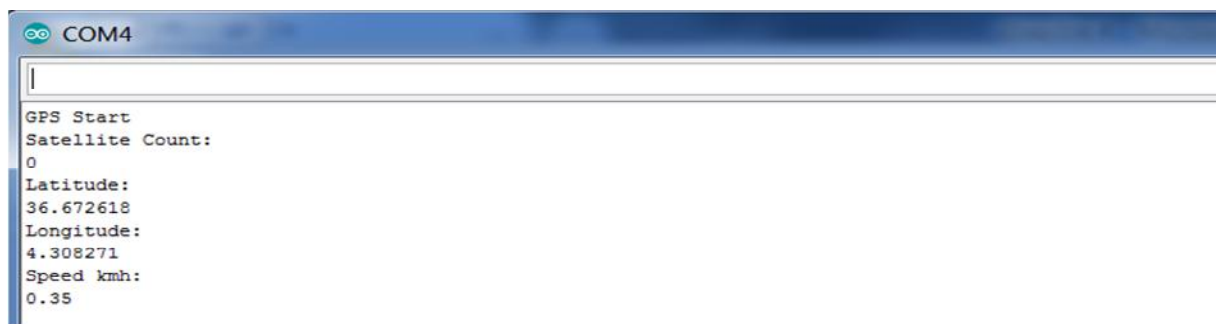


Figure 4.11 : Résultats du test de GPS.

4.3.2.1.2 Test du module GSM :

Nous avons testé le module GSM en envoyant le message « *celui-ci est un message de test du SIM900* », le module GSM a récupéré le numéro de téléphone de l'émetteur ainsi le contenu du message. Le résultat de l'envoi est présenté sur la figure 4.12.



Figure. 4.12. Envoi d'un SMS vers le module SIM908.

4.3.2.1.3 Test de la partie GPRS:

Au final, nous avons testé la partie GPRS du module, le résultat de l'envoi est présenté sur la figure 4.13.



Figure. 4.13. Résultats du test de GPRS.

4.3.2.2 Test de notre système :

La figure 4.14 montre le branchement final de notre système, nous avons injecté le code de notre application dans le microcontrôleur arduino, on l'a relié avec le module SIM900 et le module GPS via la liaison série USART et on a alimenté le tout à l'aide d'un chargeur 9v.

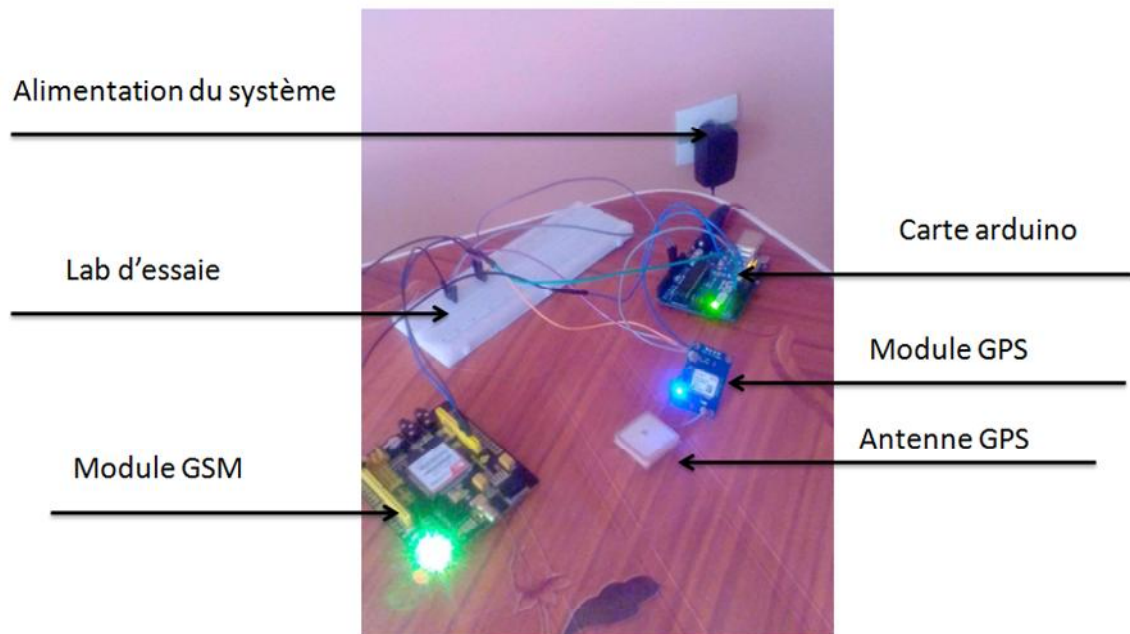


Figure 4.14 : branchement de notre système de géolocalisation.

4.3.2.2.1 Demande de la position du véhicule :

Pour tester notre système, nous avons envoyé un SMS contenant un mot de passe <<TRACK>> prédéfini dans l'EEPROM de notre uc pour savoir la localisation exacte de notre véhicule comme le montre la figure 4.15, le système répond par un SMS contenant les coordonnées géographiques de la position du véhicule sous forme d'un lien google maps.



Figure 4.15 : Réception de la position.

Suite à la réception de la position, nous avons visualisé la position de la voiture sur google maps en cliquant sur le lien reçu, la capture de la figure 4.16 illustre la localisation de notre voiture :

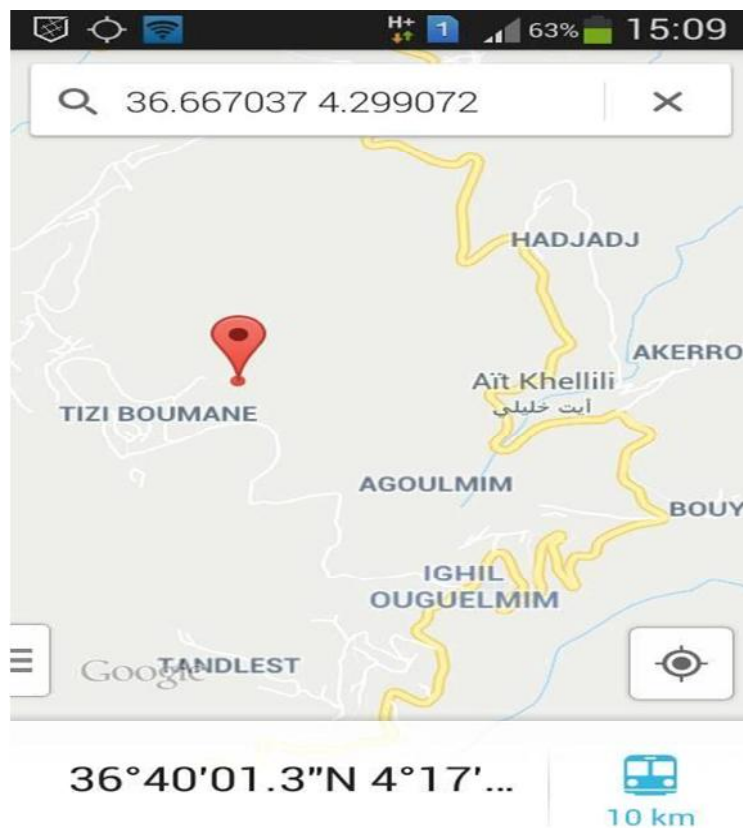


Figure 4.16 : Visualisation de la position du véhicule.

4.3.2.2 Demande de modification du mot de passe :

Nous avons testé la tâche de modification de mot de passe, pour cela nous avons envoyé la commande correspondante. Comme le montre la figure 4.17, nous avons envoyé la commande <<#ancien_mot_de_passe nouveau_mot_de_passe>>, après vérification de l'exactitude de l'ancien mot de passe, le système répond par un message de confirmation de la modification de l'ancien mot de passe <<TRACK>> par le nouveau <<GEOLO>>. En cas d'erreur le système par un message d'erreur.

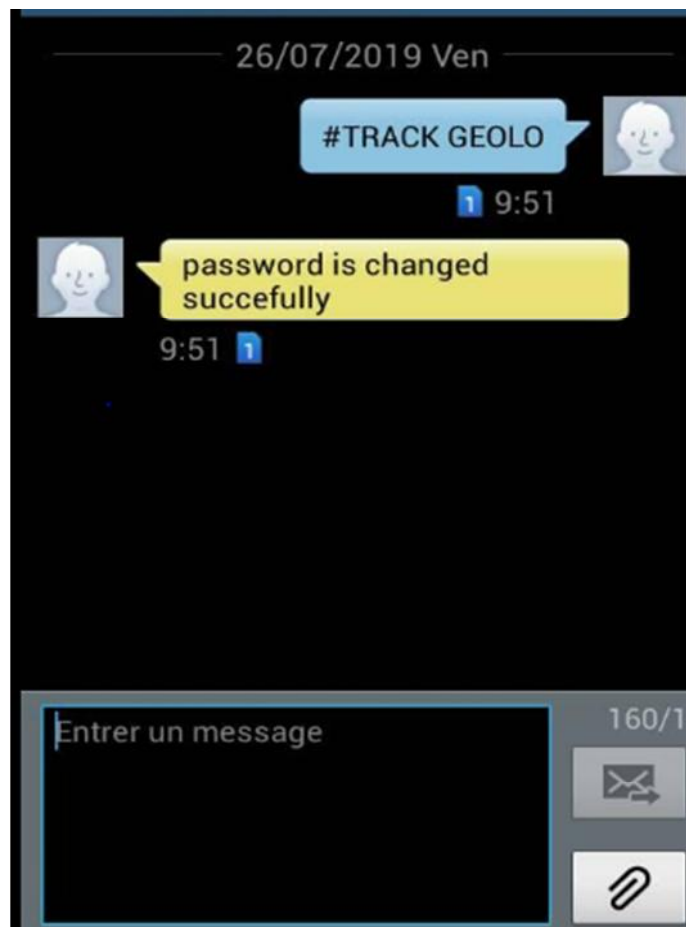


Figure 4.17: réception de la confirmation du changement du mot de passe.

4.3.3 Quelques Interface de la plateforme web de géolocalisation :

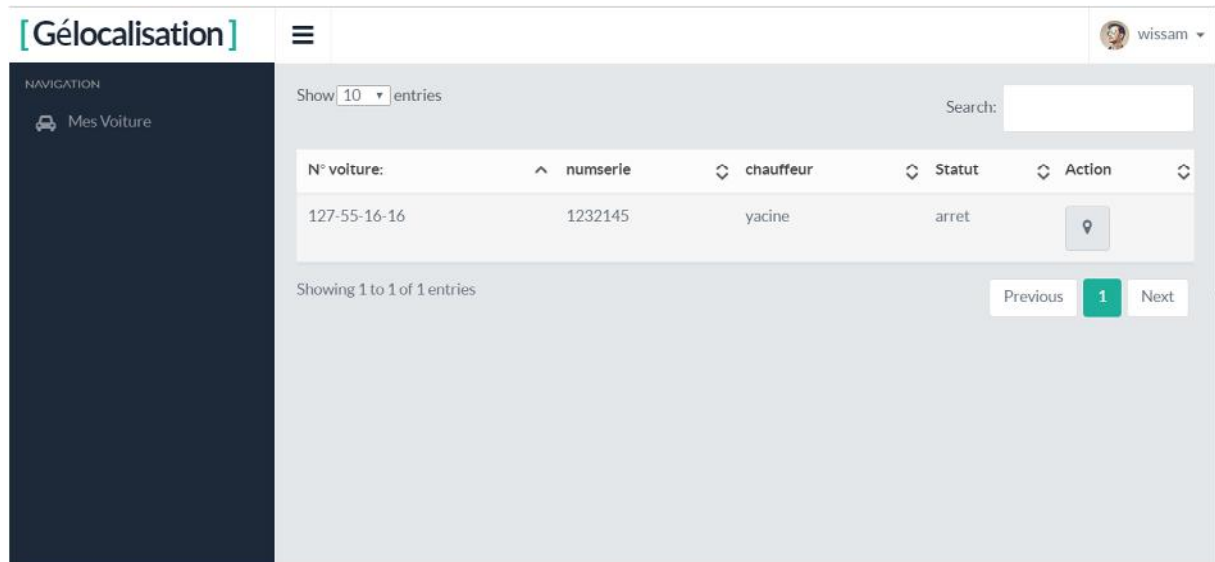


Figure 4.18: interface de sélection du véhicule

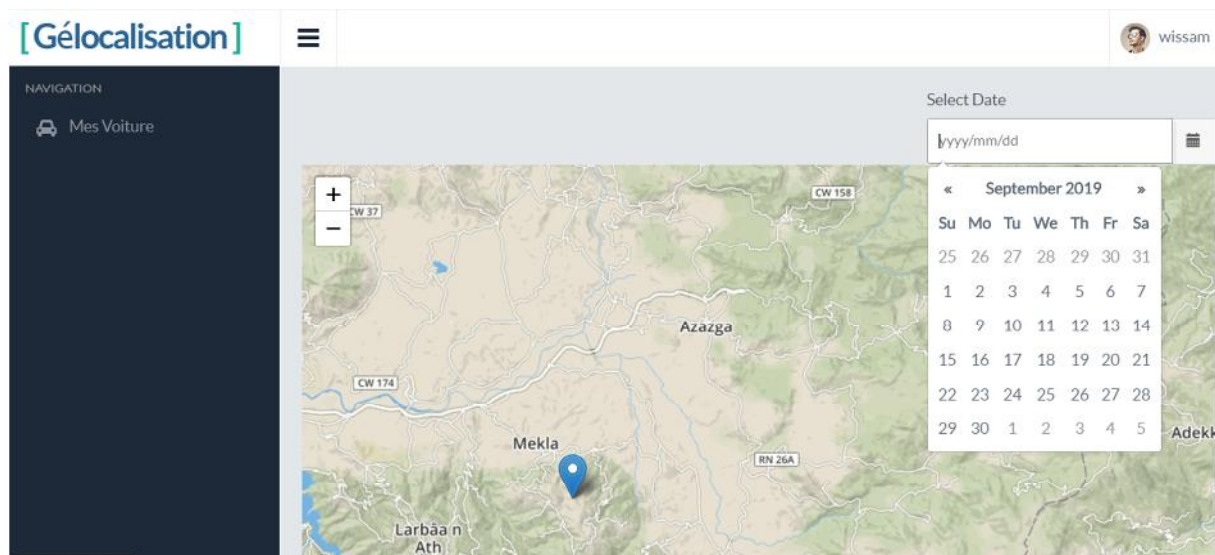


Figure 4.19: interface de visualisation de toutes les positions

4.4 Conclusion:

Au cours de ce dernier chapitre, nous avons présenté les différents Outils de développement utilisés, la réalisation du système embarqué (connexions et programmation de l'arduino). Ainsi que les figures présentant les résultats des tests effectués sur notre système, et enfin, la description de notre application en présentant quelques interfaces essentielles.

Conclusion et perspectives

Conclusion et perspectives

L'objectif du présent travail étant de concevoir et développer un système permettant de localiser un véhicule. Cette application permet à l'utilisateur la localisation en temps réel de son véhicule équipé d'un boîtier de localisation. Pour ce faire, on a eu recours à différentes technologies et outils de localisation et de communication avec la base de données distante, jugés nécessaires pour aboutir à l'objectif de l'application développée.

Notre application ainsi réalisée permet de suivre notre véhicule par deux méthodes : le tracking et la localisation sous demande.

Apports et perspectives

Ce projet nous a permis d'acquérir des nouvelles connaissances à propos le matériel utilisé : GPS, GSM et aussi la carte arduino et de maîtriser la nouvelle technologie RATCHET et de se familiariser avec le Framework LARAVEL qui seront certainement utiles dans nos futures vies professionnelles. Ce projet nous a permis aussi d'acquérir les bases de la géolocalisation et savoir les avantages de cette dernière, plus précisément les avantages apportés aux entreprises :

- ✓ Elle minimise le risque de vols ou ses conséquences.
- ✓ Elle permet de localiser quasiment en temps réel le véhicule dans le cadre d'une mission, ce qui permet une prise de décision rapide pour tous les aspects liés à ces voyages professionnels.
- ✓ Optimiser les temps de transport.

Enfin, on ne peut sans doute pas affirmer que notre travail est complet car plusieurs améliorations peuvent être apportées. Mais nous espérons au moins qu'on a réussi à réaliser une simple application fonctionnelle qui satisferait les besoins des futurs utilisateurs et qui conviendrait à leurs attentes.

Parmi ces améliorations on cite :

- ✓ Développer une application de géolocalisation android qui utilise une technologie temps réel.
- ✓ Des alertes peuvent être mises en place pour avertir les conducteurs des comportements de conduite dangereux.
- ✓ Intégration de la fonction géo-repérage ou gardiennage virtuel permettant de configurer des alertes au cas où un véhicule ne respecterait pas sa trajectoire.

Bibliographie

Bibliographie :

- [1] : Embedded Systems: Hardware, Design and Implementation , Krzysztof iniewski , Wiley-Blackwell ,11 janvier 2013
- [2] : Linux embarqué 2ème édition, pierre ficheux, Editions EYROLLE, 29 septembre 2005
- [4] : Introduction aux systèmes embarqués temps réel - Fondamentaux et études de cas: Conception et mise en œuvre, Emmanuel Grolleau, Dunod, 24 octobre 2018.
- [6] : Tout Savoir Sur : La Géolocalisation, nouvelle arme des marketeurs, Muriel Glatin et Maïke Strudthoff , Editions Kawa , 3 mars 2014.
- [9] : Arduino« Applications avancées - Claviers tactiles, télécommande par Internet, géolocalisation, applications sans fil..», Christian Tavernier, Dunod, octobre 2012.
- [10] : Jean-Baptiste HENRY, Ingénieur Géomètre-Topographe «Service régional de traitement d'Image et de télédétection ».
- [14] : La gestion de flotte de véhicules par GPS et GSM, Stéphane SOLEILHAVOUP, technologie 165, janvier-février 2010.
- [15] : Architecture logicielle et méthodologie de conception embarquée sous contraintes temps réel pour la radio logicielle, Thèse de Doctorat, Noël Bertrand TCHIDJO MOYO, Université de rennes 1, 2011.

Sites web:

[3]: <https://tel.archives-ouvertes.fr/tel-00852361/document>

[5] : CLAUDE GUEDAT, OLIVIER BRETTE, « systèmes de géolocalisation », cours, janvier 2010.

[7] : <http://www.hospital.info/index.html>, consulté le 23 juin 2019.

[8] : <http://www.geolocalisation-vehicule.be>

[11] : https://fr.wikipedia.org/wiki/Global_Positioning_System

[12] : <https://lewebpedagogique.com/isneiel/files/2017/06/GPS.pdf>

[13] : http://www.memoireonline.com/07/10/3741/m_Evolution-technologique-de-la-3G1.html

[16] : <https://www.openimpulse.com/blog/products-page/product-category/gy-neo6mv2-gps-module/>

[17]: https://wiki.keyestudio.com/Ks0142_keyestudio_SIM900_GSM/GPRS_Module_Shield

[18] : <https://www.arduino.cc/en/main/software>

[19] : <https://laravel.com/>

[20] : <https://fr.wikipedia.org/wiki/>

[21] : <https://leafletjs.com>

[22] : <http://socketo.me/>

Annexes

Annexes A

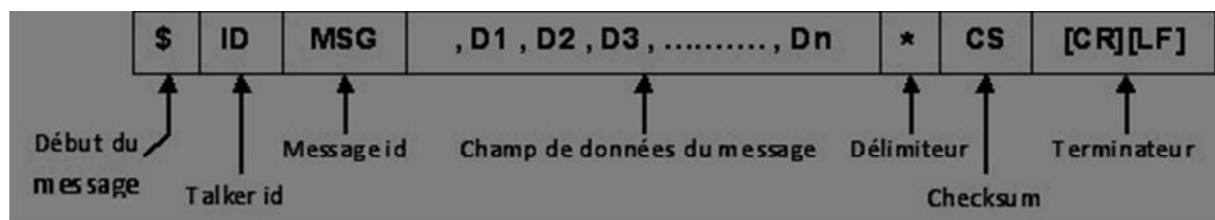
Le standard NMEA.

NMEA (National Marine Electronics Association) est une association américaine denfabricants d'appareils électroniques maritimes, basée à Severna Park au Maryland (États-Unis d'Amérique). Elle est à l'origine de nombreux standards et en particulier du Standard NMEA-0183.

NMEA 0183 est une spécificaon pour la commu ni cation entre équi pements n a i rs , dont les équipements GPS. Sous ce standard, toutes les données sont transmises sous la forme des caractères ASCII, tous imprimables, ainsi que les caractères [CR] Retour Chariot et [LF] Retour à la ligne. Les données sont transmises sous forme de trames.

Dans ce qui suit, le Standard NMEA est défini comme étant le protocole de transmission des données entre les équipements électroniques liés au GPS.

1- Format de la trame NMEA 0183 :



Chaque trame commence par le caractère \$

Suivi par un groupe de 2 lers pour l'idenfiant du récepteur . ID

Exemples :

- ✓ GP pour Global Positioning System.
- ✓ LC Loran-C receiver.
- ✓ OM Omega Navigation receiver.
- ✓ II Integrated Instrumentation (eg. AutoHelm Seataalk system).

Puis un groupe de 3 lers pour l'idenfiant de la trame (MSG).Exemples :

- ✓ GGA : pour GPS Fix et Date.
- ✓ GLL : pour Positionnement Géographique Longitude - Latitude.
- ✓ GSA : pour DOP et satellites actifs.
- ✓ GSV : pour Satellites visibles.
- ✓ VTG : pour Direction (cap) et vitesse de déplacement (en noeuds et Km/h).
- ✓ RMC: pour données minimales exploitables spécifiques.

Suivent ensuite un certain nombre de champs séparés par une "virgule". La virgule permet la dé-concaténation des données dans le programme de traitement des données, calculateur, navigateur.

Et enfin un champ optionnel dit checksum précédé du signe *, qui représente le « OU exclusif » de tous les caractères compris entre \$ et * (sauf les bornes \$ et *), certaines trames exigent le checksum.

Suit la fermeture de la séquence avec un [CR][LF].

Une trame comporte au maximum 82 caractères.

Annexes B

Les protocoles SMS-SUBMIT et SMS-DELIVER du réseau GSM.

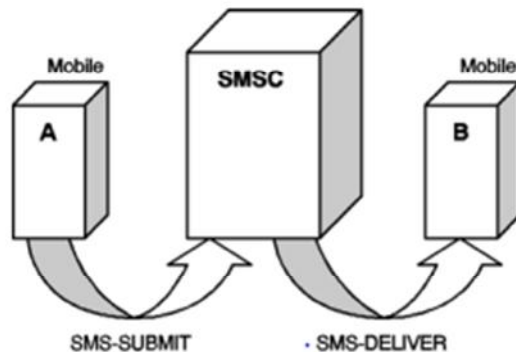


Figure 3

Lorsqu'un mobile A envoie un SMS à un mobile B, le message est acheminé vers le centre de messagerie (SMSC) par le protocole SMS-SUBMIT. Une fois le message traité par le SMSC, il est délivré au mobile B, par le protocole SMS-DELIVER (figure 3).

Les protocoles SMS-SUBMIT et SMS-DELIVER sont des PDU, il en existe d'autres, d'importance moindre, qui permettent de signaler des éventuelles erreurs d'acheminement :

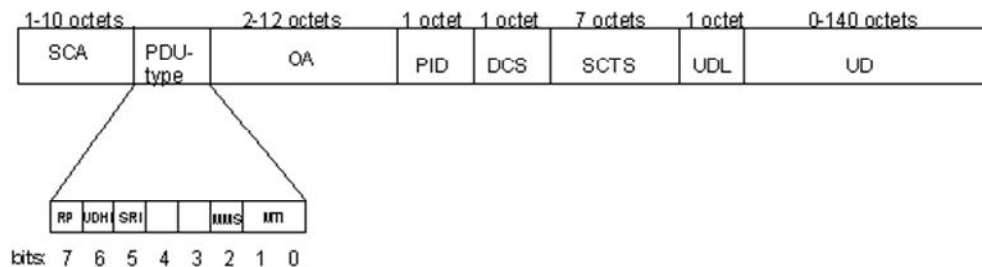
- SMS-DELIVER-REPORT : le cas échéant, il indique une défaillance lors du transfert du SMS par le SMSC au destinataire ;
- SMS-SUBMIT-REPORT : le cas échéant, il indique une défaillance lors du transfert du SMS par le mobile au SMSC ;
- SMS-STATUS-REPORT : le SMSC envoie un rapport d'état au mobile émetteur du SMS ;
- SMS-COMMAND : le mobile envoie une commande au SMSC.

Les trames codant le SMS sont différentes suivant le type de PDU mis en œuvre. Dans la partie qui va suivre nous nous contenterons de décrire les deux principaux protocoles : SMS-SUBMIT et SMS-DELIVER.

SMS-SUBMIT

Téléphone GSM vers SMSC. La taille maximale de la trame d'un SMS-SUBMIT est de 173 octets. Le champ le plus important en terme de taille est le champ qui codifie le corps du

message qui peut atteindre 140 octets. Les deux autres champs indispensables sont SCA qui codifie l'adresse du SMSC et DA qui codifie l'adresse de l'émetteur.



La taille maximale de la trame d'un SMS-SUBMIT est de 173 octets. Le champ le plus important en termes de taille est le champ qui codifie le corps du message qui peut atteindre 140 octets.

SCA : (Service Center Address) Adresse du centre de messagerie, sa taille dépend de la longueur du numéro du SMSC utilisé.

Type de PDU : Le champ PDU toujours codé sur 1 seul octet, il a pour fonction principale de définir s'il s'agit d'un SMS-DELIVER ou d'un SMS-SUBMIT. Dans le cas du SMS-SUBMIT les deux premiers bits de ce champ sont positionnés à « 01 ».

MR: (Message reference) chaque message envoyé par le mobile au SMSC est identifié par un numéro compris entre 0 et FFhex (Référence du Message).

DA: (Destination Address) indique l'adresse du destinataire.

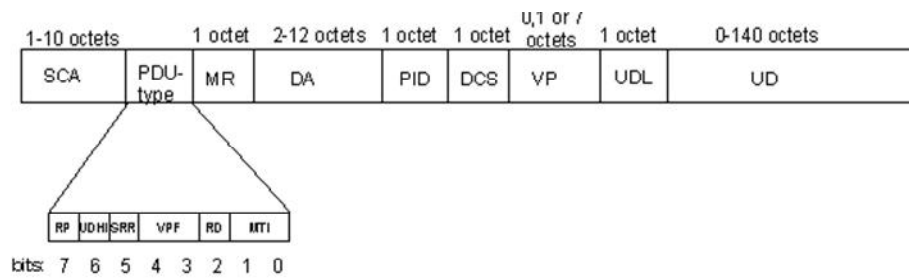
PID: (Protocol Identifier) le champ PID indique à quel type de service télématique est destiné le message. Dans le cas du réseau GSM ce champ sera toujours positionné à 00hex c.à.d. que la trame est traitée comme un message court.

DCS(Data Coding Scheme) le champ DCS indique de quelle manière est codé le champ UD qui correspond au corps du message. Dans la pratique, tous les bits sont positionnés à zéro c.à.d. que le champ UD est codé avec l'alphabet GSM (avec l'alphabet GSM chaque caractère est codé sur 7 bits).

VP : (Validity Period) permet d'indiquer au SMSC la durée de validité du SMS.

UDL : (User Data Length) et UD : (User Data) : UDL contient la taille en octets utilisés pour codifier le message dans UD.

SMS-DELIVER



SCA : Service Centre Adresse

Adresse du centre de messagerie, il indique quel est le SMSC qui a traité le SMS.

PDU : Protocol Data Unit

Les deux premiers bits de ce champ sont positionnés à « 00 » pour indiquer qu'il s'agit d'un SMS-DELIVER.

OA : Originator Address

Indique l'adresse de l'émetteur.

PID : Protocol Identifier, DCS: Data Coding Scheme

Ces champs sont identiques à ceux présentés dans la partie SMS-SUBMIT.

SCTS : Service Centre Time Stamp

Le champ SCTS indique au destinataire la date et l'heure à laquelle le SMS est arrivé au SMSC.

UDL : User Data Length, UD : User Data

Ces champs sont identiques à ceux présentés dans la partie SMS-SUBMIT