



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la

Recherche Scientifique

Université Mouloud Mammeri de Tizi-Ouzou

Faculté de Génie Electrique et Informatique

Département d'informatique



Mémoire De fin d'études

En vue d'obtention du diplôme de Master en informatique

Option: Réseaux, Mobilités et Systèmes Embarqués

Thème:

*Systeme de géolocalisation pour les
réseaux mobiles*

Encadré par :

M^r: TALEB Hakim

Membre de Jury composé de :

M^r: SADOU Samir

M^r: DJADEL Hacene

M^r: HAMACHE Arzeki

Réalisé par :

M^{elle}: MERABTENE Amel

M^{elle} : AGRED Nouara

Promotion 2012/2013

Dédicaces

*A mes chers parents,
A mes soeurs Nadia, Farida et
Dady,
A toute la famille AGRED,
A tous mes amis(es),
A tous ceux qui me sont chers, ...
Je dédie ce travail.
Nouara.*

*A mes chers parents,
A mes chers frères et soeurs,
A mes chers(es) amis (es),
A tous ceux qui m'ont aidé de près ou de loin.
Je dédie ce travail.*

Amel.

Résumé

Autrefois, la localisation était utilisée uniquement par les professionnelles comme les marins, et les systèmes satellitaires pouvaient répondre parfaitement à leurs besoins, puisque ces utilisateurs se trouvaient généralement dans de vastes espaces bien desservis par le GPS. Aujourd'hui, les applications de localisation se sont étendues au grand public, notamment avec la vulgarisation des appareils mobiles. Les besoins passent donc par la localisation de son véhicule, celle des personnes à l'intérieur de bâtiments, de gares ou d'aéroports, etc.

L'environnement intérieur présente l'inconvénient d'être moins favorable au GPS, donc trouver d'autres systèmes de substitution s'avère indispensable, et c'est les systèmes basés sur les réseaux locaux : RFID, Infrarouge ou même Wifi, qui ont été retenus.

Les exigences des professionnels ont aussi évolué en terme de sécurité, de contrôle, et d'optimisation des méthodes de travail, de ce fait un bon système répondant à ces exigences est considéré comme une réelle nécessité.

Face aux besoins de trouver sa position dans le magasin, trouver son chemin dans le magasin, nous nous proposons de réaliser un système de géolocalisation indoor par Wifi.

Le système se basera sur la méthode fingerprinting : une méthode basée sur la collecte de puissances à des endroits spécifiques de la zone couverte dans une phase d'apprentissage, les enregistrer dans une base de données, pour les exploiter à déterminer la position du mobile dans une phase de positionnement. L'algorithme des k plus proches voisins sera implémenté pour estimer la position la plus vraisemblable.

Le système respecte une architecture infrastructure centré, les calculs de position seront effectués par un serveur de traitement après qu'il est reçu un rapport de puissance en provenance du mobile souhaitant retrouver sa position.

Une spécification précise des besoins du système de positionnement, avec l'utilisation d'outils professionnels de planification et de conception ont conduit à la

bonne réalisation des différents modules logiciels à l'aide du langage Java et de la plateforme Android. Les tests ont été réalisés en environnement réel à l'aide d'un smartphone Android et les résultats obtenus montrent les performances de la localisation indoor basée sur un réseau Wifi.

Mots-clés : Géolocalisation, Positionnement en intérieur, Wifi, Android, fingerprinting, trilatération, triangulation.

Remerciements

*À dieu, le tout puissant, nous rendons grâce pour nous avoir
donné santé, patience, volonté pour réaliser ce travail.*

*Notre mémoire touche à sa fin, nos remerciements vont
exceptionnellement à notre promoteur Mr H.TALEB pour son
encadrement, son suivi tout au long de l'élaboration de notre projet.*

*Nous tenons, également, à remercier les membres du jury qui nous
font l'honneur d'examiner et de juger notre travail.*

*Enfin, nous souhaitons témoigner notre gratitude à toutes les
personnes qui ont contribué de près ou de loin à l'accomplissement de ce
modeste travail.*

Sommaire

Dédicaces	ii
Résumé.....	iii
Remerciements.....	v
Sommaire.....	vi
Liste des figures.....	x
Liste des tableaux.....	xi
Introduction Générale.....	10
<i>Chapitre I: Etat de l'art de la géolocalisation</i>	
Introduction.....	13
I.1. Généralités sur la géolocalisation.....	13
1.1 Définition.....	13
1.1.2 Terminologie de l'infrastructure des systèmes de géolocalisation	13
1.1.3. Techniques actuelles de géolocalisation	14
1.4. Géolocalisation en intérieur.....	22
2 .Les méthodes et techniques de positionnement en intérieur	30
I.2.1. Méthode géométrique	30
I.2.2. Méthode Statistique.....	32
I.3. Les techniques de Mesure.....	33
I. 3.3. Angle d'arrivée (ou AoA pour Angle of Arrival.....	34
I.4. Collecte de mesure des caractéristiques des signaux.....	34
I.4.1. Cartographie des puissances	35
I.4.2. Modèle de propagation des ondes	36
I.5. Algorithme de calcul de position.....	36
I.6. Applications et services basées sur la géolocalisation.....	37
Conclusion.....	36
<i>Chapitre II : Choix et stratégie de développement</i>	
Introduction.....	39

II.1 Démarche de conduite du projet de développement.....	39
II.2. Informatique mobile et système d'exploitation android.....	40
II.2.1. L'informatique Ubiquiste	40
II.2.2. L'informatique mobile.....	40
2.2.1. Définition :	40
2.2.2. Présentation :	41
2.2.3. Les smartphones	41
II.2.3. Les systèmes d'exploitation Mobiles	41
II.2.3.1. Caractéristiques.....	43
II.2.3.2. Choix de la plateforme Android.....	44
II.3.Objectif et choix techniques.....	45
II.3.1 Objectif	45
II.3.2 Choix techniques.....	45
II.3.2.1. Méthode de localisation.....	45
II.3.2.2. Algorithme de localisation	46
II.3.2.3. Choix de calcul centré mobile	49
II.3.2.4. Architecture générale du système de géolocalisation.....	49
II.3.2.5. Fonctionnement du système	50
Conclusion.....	51
Chapitre III : Analyse & Conception	
Introduction.....	52
III.1. Modélisation du système.....	52
III.1.1. Présentation du langage UML	52
III.1.2. Modélisation avec le langage UML.....	53
III.2. Analyse	54
III.2.1. Identification des besoins	54
III.2.2. Identification des acteurs	55
III.2.3. Les cas d'utilisation	55
III.2.4. Description du cas d'utilisation	55
III.3. Conception.....	56
III.3.1. Le niveau applicatif	56
3.1.1. Le diagramme des cas d'utilisation.....	56
3.1.2. Diagramme de séquence	58

3.1.3. Les diagrammes d'activités.....	59
3.1.4. Les diagrammes de classe	61
III.3.2. Le niveau Données	63
3.2.1. Schéma de la base de données	63
Conclusion.....	64
Chapitre IV: Réalisation	
IV.1. Description de la plateforme matérielle.....	65
IV.2. Environnement de développement logiciel.....	65
IV.2.1. Les outils de développements.....	65
2.1.3 L'IDE Eclipse	68
2.1.4 Le plugin ADT (Android Development Tool) pour eclipse :	69
IV.3. Modules logiciels de l'application.....	74
IV.3.1 Construction de la base de données des mesures (cartographie des puissances):	75
IV.3.2 Module Récupération des signaux wifi :	78
IV.3.3 Module Calcul de position :	78
IV.3.4 Module Interface Android :	78
VI.4 Test du système.....	80
Conclusion	81
Annexe.....	85
Références bibliographiques.....	101

Liste des figures

Figure 1:Principe de positionnement.....	13
Figure 2:structure générale d'un réseau GSM.....	15
Figure 3: Assisted global positioning system (A-GPS).....	17
Figure 4:Localisation par AOA	17
Figure 5: TDOA (Time Difference Of Arrival)	18
Figure 6:Cell-id.....	19
Figure 7:Handover	20
Figure 8:Schéma général d'un système d'identification par radiofréquence	21
Figure 9:TAG (ETIQUETTE).....	22
Figure 10:réseau Wi-Fi avec infrastructure	23
Figure 11:Architecture ad hoc	27
Figure 12:Trilatération.....	31
Figure 13:Triangle	29
Figure 14:Triangulation.....	30
Figure 15:localisation par fingerprint dans un réseau wifi.....	30
Figure 16:Organigramme des étapes de la démarche du projet	31
Figure 17:Le marché des smartphone.....	37
Figure 18:Comparaison entre os mobiles	40
Figure 19:Caractéristiques du système d'exploitation android.	41
Figure 20:Exemple de l'algorithme du K plus proche voisin	42
Figure 21:Architecture du système de positionnement en intérieur.....	47
Figure 22:Démarche adoptée pour la modélisation.....	49
Figure 23:Diagramme de cas d'utilisation Client.....	52
Figure 24:diagramme de séquences du cas d'utilisation « afficher map ».....	55
Figure 25:Diagramme de séquence du cas d'utilisation client.....	56
Figure 26:Diagramme d'activité de cas d'utilisation « afficher map ».....	57
Figure 27:Diagramme d'activité de cas d'utilisation « Se localiser ».....	58
Figure 28:Diagramme de classe « se localiser »	59
Figure 29:structure de la base de données.....	60
Figure 30:Le concept de machine virtuelle java	61
Figure 31:L'émulateur android.....	62
Figure 32:L'environnement de développement eclipse	65
Figure 33:La console DDMS sous eclipse	66
Figure 34:Cycle de vie d'une application android.....	67
Figure 35:Points de mesure réalisés au laboratoire	68
Figure 36:Interface Client Android	70
Figure 37:Plan du département Informatique (Etage 1).....	74
Figure 38:Résultats du scan.....	77

Figure 39:Géo localisation à l'intérieur du laboratoire	80
Figure 40:Relation entre performance et nombre de points d'accès	82

Liste des tableaux

Tableau 1:Table point de référence	62
Tableau 2:Comparaison entre positions réelles et positions générées par le système	79

Introduction Générale

Avec l'avènement du sans fil dans les réseaux informatiques, et le développement rapide des nouvelles technologies de l'information et de la communication notamment, l'apparition de la troisième génération UMTS (Universal Mobile Télécommunication System), plusieurs services basés sur le concept de mobilité ont vu le jour, en particulier le service de la géo-localisation : qui consiste en la capacité d'un système de communication, à déterminer la position géographique d'un terminal.

Le service de géo-localisation a connu un succès grandissant auprès des individus et des industriels, ainsi la mise sur le marché de nombreuses solutions technologiques visant à offrir plus de services et de confort par exemple localisation de biens (traçage des véhicules d'entreprise, avions, navires) et/ou de personnes (localisation des enfants), la publicité ciblée par rapport aux commerces environnants, l'automatisation des forces de vente (aide à la gestion et suivi des collaborateurs mobiles) jusqu'au géomarketing (analyser une population, trouver des adresses autour d'un point donné).

Plusieurs systèmes de géo-localisation ont été proposés à commencer par le système de navigation par satellite le plus répandu : le GPS (Global Positioning System), conçu principalement à des fins militaires l'utilisation du système s'est vu étendre dans d'autres domaines tels que : les transports maritimes, aériens et terrestres, la géologie, les travaux publics, la prospection pétrolière et l'agriculture, etc.

Les opérateurs de téléphonie mobile ont eux aussi la possibilité de localiser leurs abonnés grâce à l'architecture complexe du réseau GSM (Global System for Mobile communications) avec ses structures physiques et ses bases de données.

Ces systèmes offrent une bonne précision de localisation à l'extérieur, dans des endroits dégagés, mais ils restent moins puissants à l'intérieur des bâtiments, ce qui les rend inexploitable. De là, est née la nécessité de trouver des systèmes de substitution

qui franchissent les limites du GPS et combinant efficacité et précision à l'intérieur : aéroport, bâtiments, campus et centres commerciaux, etc.

D'autres technologies comme celles par tags actifs/passifs ou de vision sont autant de moyens pour se localiser en intérieur. On peut citer les systèmes à ultrason, la localisation par infrarouge, par vidéo, la localisation par mesure de champ magnétique ou encore par technologie RFID (Radio Frequency IDentification).

Cependant, ces technologies présentent des points de faiblesse qui peuvent devenir très vite des freins quant à leur déploiement, notamment la complexité d'installation : synchronisation des éléments entre eux, conditions particulières d'installation. Le coût des éléments d'un système gêne parfois aussi le déploiement d'une technologie, et particulièrement lorsque de très nombreux éléments relativement onéreux doivent être installés.

La solution passe donc, par la technologie wifi. Cette technologie présente l'énorme avantage d'être largement développée, présente et déployée pour d'autres services dans les aéroports, les centres commerciaux, les salons d'exposition, les hôpitaux, les gares, les entreprises, etc. Ce qui a suscité un intérêt particulier pour cette dernière au sein de la communauté scientifique, qui tente sans cesse d'exploiter le wifi dans le développement de nouveaux systèmes de positionnement jouissant de meilleures performances en termes de précision et de temps de réponse.

Notre projet consiste donc, en la réalisation d'un système, dont l'objectif est de localiser tout individu détenteur d'appareil mobile android en se basant sur la structure Wifi.

Le présent rapport est organisé en quatre chapitres couvrant l'ensemble des aspects de notre travail.

- Le chapitre un, présente un état d'art du domaine de la géo-localisation. On y explore les principaux concepts concernant la géo-localisation en présentant, dans un premier temps, les systèmes de localisation à grande échelle existants à savoir : le GPS et le GSM. Le contexte de notre travail étant la localisation en indoor, nous ne pouvons recourir aux systèmes de localisation par satellite qui exigent

que leurs éléments (satellite et terminal) soient en visibilité directe. Ainsi, le chapitre expose, dans un second temps, la localisation en intérieur basée sur le réseau RFID, Infrarouge, et Wifi. Une attention particulière est consacrée à la localisation par Wifi où les différentes méthodes et techniques existantes ainsi que les algorithmes développés dans ce domaine sont présentés.

- Le chapitre deux, nous nous basons sur des solutions existantes pour les adapter à notre problématique.
- Le chapitre trois, présente la partie d'analyse et de conception qui est à la base de l'application développée. Il discute l'aspect conceptuel des différents modules du système.
- Le chapitre quatre, présente l'environnement d'implémentation et aborde les aspects techniques liés à la réalisation de l'application. Une phase de test est également observée qui illustre, à travers une série d'images et de capture d'écran, le comportement de l'application à l'exécution et répondre à certaines questions telles : Permet-elle vraiment d'atteindre la précision de positionnement ? Est-elle fiable pour la géolocalisation ? Les résultats obtenus sont-ils un point fort de ce système ?

Enfin, ces chapitres sont précédés par une introduction générale et clôturés par une conclusion générale.

Introduction

Se déplacer est une nécessité vitale pour chaque personne. De tout temps, l'homme a développé des moyens techniques pour faciliter ses déplacements et augmenter son autonomie : depuis la carte et la boussole, les systèmes d'assistance à la navigation ont connu un formidable essor. Grâce à ces systèmes on pourra se localiser en continu, de planifier sa trajectoire puis d'être guidé en considérant les obstacles éventuels.

Dans ce chapitre, il sera question d'introduire le cadre théorique dans lequel s'inscrit notre travail, en définissant dans un premier temps la géolocalisation ainsi que tous les concepts qui gravitent autour, ensuite nous classifierons les systèmes de positionnement les plus conventionnels selon différents critères.

I.1. Généralités sur la géolocalisation

1.1 Définition

La *géolocalisation*, le *géopositionnement* ou le *géoréférencement* sont tous des termes qui désignent un procédé technique permettant de positionner un objet, une personne, etc, sur une carte ou un plan à l'aide de coordonnées géographiques. Ce processus est réalisable grâce aux terminaux mobiles équipés de fonction de télécommunication radio, GPS, GSM ou Wifi. [Clz08]

I.1.2 Terminologie de l'infrastructure des systèmes de géolocalisation

Avant de se lancer dans la géolocalisation proprement dite, il est nécessaire de définir l'architecture d'un système de géolocalisation ainsi que les différents modules qu'on exposera dans ce qui suit : [Las09]

- Un *terminal mobile* (PDA, smartphone, portable...) permet à son porteur de se mouvoir et d'accéder à un système informatique pour demander de l'information

- Un *réseau de communication* permet le transfert des données entre l'utilisateur et le service géo-localisé
- Un *mécanisme de positionnement* absolu ou relatif permettant de connaître implicitement (peut être explicite) la localisation de l'utilisateur
- Un ou plusieurs *serveurs d'application de géolocalisation* qui offre une palette de services à l'utilisateur et est responsable du traitement des requêtes de l'utilisateur : calculs de positions, navigation, etc.
- Un ou plusieurs *serveurs délivrant données* et contenus associés : base de données géographique et méta information associées.

I.1.3. Techniques actuelles de géolocalisation

De nos jours, les techniques de localisation sont multiples. Une bonne maîtrise et connaissance de ces diverses méthodes sont nécessaires afin de dimensionner judicieusement sa propre solution de localisation.

Dans ce qui suit, nous présentons les différentes techniques utilisées à l'extérieur comme à l'intérieur d'un espace donné

1.3.1. Géolocalisation par satellite

L'exemple le plus connu de positionnement est le GPS (Global Positionning System) qui consiste à déterminer la position, par la connaissance des distances entre 3 satellites de la constellation GPS et l'observateur. Ces distances sont calculées à partir de la mesure des temps de parcours des signaux émis par les satellites, dont ce calcul repose sur le principe de la triangulation.

Une constellation de satellite en orbite autour de la terre signifie que :

- Chaque satellite diffuse en permanence des signaux sur l'ensemble des points qui lui sont visibles sur la terre.
- Chaque satellite inclut dans les signaux des messages qui représentent sa position exacte dans l'espace

Un nombre illimité de récepteurs-utilisateur, dotés des propriétés suivantes :

- réception de signaux d'au moins quatre satellites de la constellation.
- mesure de distance qui le sépare de ces quatre satellites.
- Calcule sa propre position en utilisant les équations et les informations transmises par les satellites [Par 96].

La figure suivante montre comment on peut (en principe) se localiser sur la Terre à l'aide de trois satellites seulement :

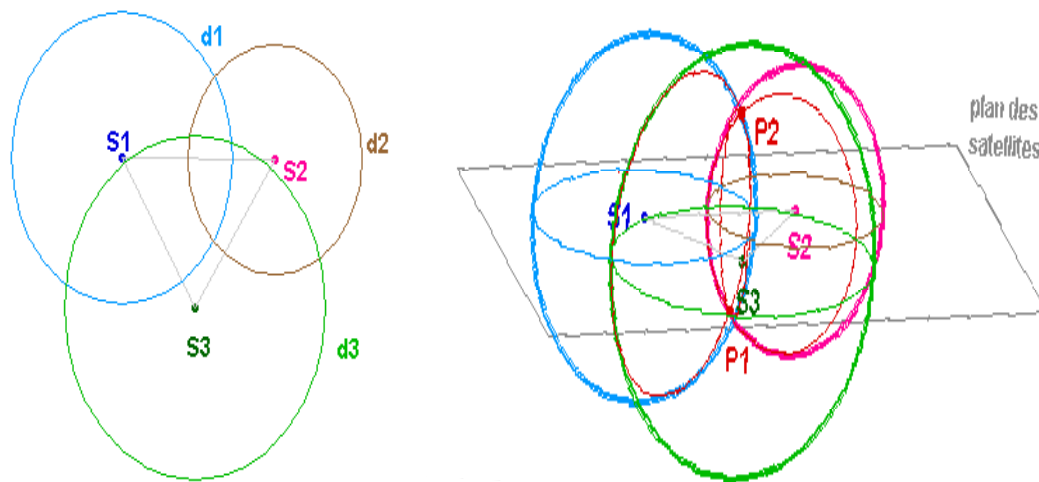


Figure 1:Principe de positionnement

Soit d_1 la distance au satellite S_1 : ce paramètre permet de se localiser sur une sphère de rayon d_1 et de centre S_1 .

Soit d_2 la distance au satellite S_2 : on se trouve également sur une sphère de rayon d_2 et de centre S_2 , donc sur le cercle intersection de ces deux sphères (vu comme une ellipse sur la figure 1).

La distance d_3 à S_3 localise l'observateur à l'intersection des 3 sphères (donc des 3 cercles) ; deux points P_1 et P_2 sont possibles, symétriques par rapport au plan des satellites, mais il n'y a pas d'ambiguïté : l'altitude des satellites GPS est de l'ordre de 20 000 km, donc, si P_1 est au voisinage de la Terre, P_2 se trouve à 40 000 km au-

dessus. Cette méthode permet d'obtenir la longitude, la latitude, mais aussi l'altitude de l'observateur.

Dans la pratique, les mesures sont faites avec 4 satellites ou plus, afin d'augmenter la précision [Cab12].

Le réseau satellite le plus connu est le Global Positionning System mais il existe d'autres systèmes fonctionnant ou en cours de déploiement. On peut relever le Glonass (Russie), le Beidou (Chine), l'IRNSS (Inde) et Galileo (Union européenne). Ce dernier sera opérationnel en 2014 et garantira l'autonomie de l'Union européenne dans le domaine militaire ainsi que des facultés avancées par rapport au GPS ou au Glonass.

Le GPS est composé de 32 satellites tournant autour de la Terre à une vitesse de 1 tour en 12 heures à une altitude entre 20000 et 20500 Km. Ces satellites sont répartis sur 6 orbites différentes. Le but de ce maillage est d'avoir constamment 4 satellites visibles par les navigateurs GPS et ainsi être d'une précision très fiable. L'ESA¹ a annoncé que Galileo possèdera 30 satellites répartis sur 3 orbites différentes. A noté que Galileo sera compatible avec le GPS et le Glonass [Dav].

1.3.2. Géolocalisation par GSM

Le GSM (Global System for Mobile communication) est un standard pour les communications de téléphonie mobile.

Dans un réseau GSM, le territoire est découpé en petites zones appelées cellules d'où le terme la localisation cellulaire qui est une technique situant le terminal mobile par rapport à l'antenne de l'infrastructure réseau à laquelle il est connecté. Il peut s'agir d'un BTS (Base Tranceiver Station) ou même de point d'accès Wifi, dans de telles méthodes de positionnement c'est la portée des antennes qui détermine la précision de localisation.

1.3.2.1. Les équipements du réseau GSM

ESA¹ : est une agence spatiale fondée le 31 mai 1975. Elle est chargée de la coordination des projets spatiaux de 17 pays européens, dont chacun détient *une* voix au conseil.

Le réseau GSM est composé de plusieurs entités, se distinguant par un fonctionnement très spécifique. La figure suivante montre les différentes couches du réseau GSM

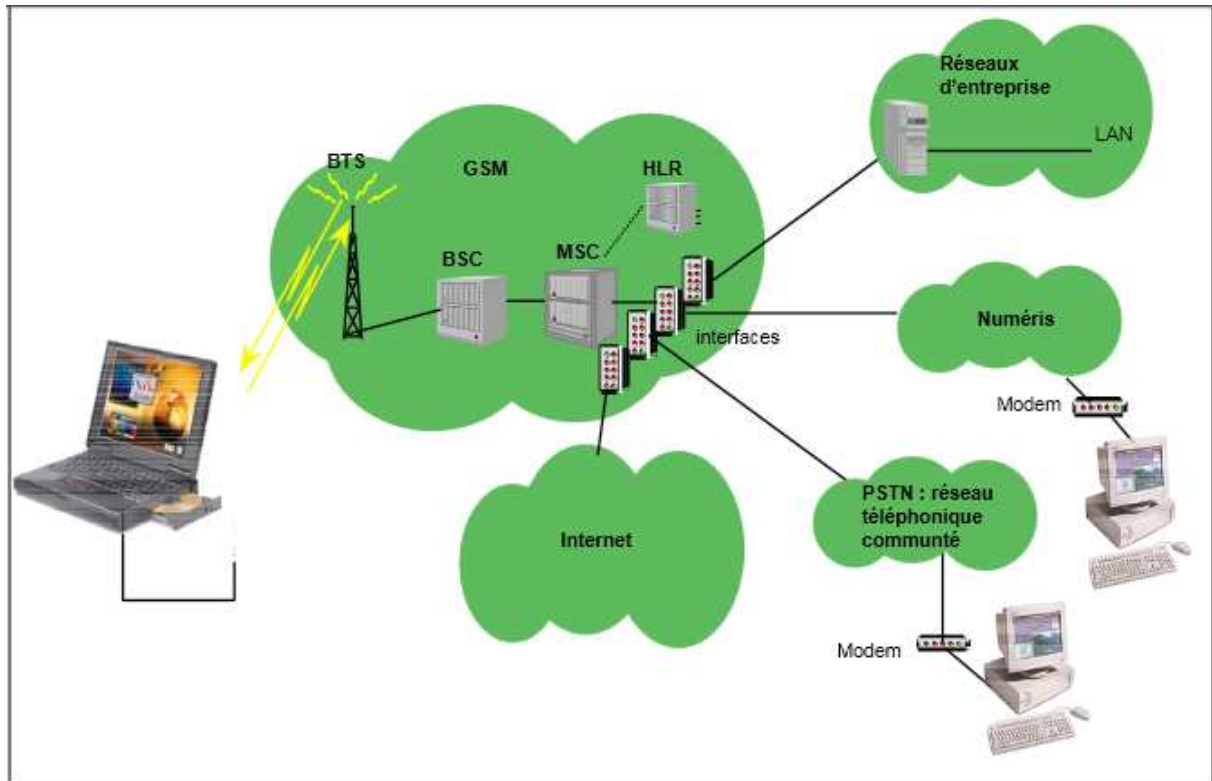


Figure 2:structure générale d'un réseau GSM.

Le réseau peut être divisé en quatre sous systèmes :

- Le téléphone GSM ou station mobile MS : est caractérisée par deux identités :
 - 1) Le numéro d'équipement, IMEI (International Mobile Equipment Identity) mis dans la mémoire du mobile lors de sa fabrication.
 - 2) Le numéro d'abonné IMSI (International Mobile Subscriber Identity) se trouvant dans la carte SIM (Subscriber Identity Module) de l'abonné.
- Le système de communication radio est l'équipement qui assure la couverture de la cellule et comprend :
 - 1) Les stations de transmission de base BTS (Base Transmitter Station)

2) Le contrôleur de stations de base BSC (Base Station Controller) qui gère entre 20 et 30 BTS et possède son registre d'abonnés visiteurs VLR stockant les informations de l'abonné liées à sa mobilité.

- Le commutateur de services mobiles MSC est un autocommutateur qui assure les fonctions de commutation nécessaires en aiguillant les conversations vers la MSC du correspondant ou vers d'autres réseaux (téléphonique, Internet, Numéris ...) à travers des interfaces appropriées.
- Le registre des abonnés nominaux ou HLR (Home Local Register) est une base de données utilisée pour la gestion des abonnés mobiles et contenant deux types d'informations :
 - 1) Les informations d'abonnés, le numéro d'abonné (IMSI)
 - 2) Les informations sur la localisation de l'abonné, permettant aux appels entrant dans le réseau d'être acheminés jusqu'à ce mobile [Mul02].

1.3.2.2. Méthodes de positionnement sur GSM

a) A-GPS (Assisted Global Positioning System)

Dans cette technique le mobile reçoit les signaux d'au minimum trois satellites, il estime le temps d'arrivée et envoie ces informations aux BTS via connexion radio.

Les traitements seront effectués par les BTS ce qui représente une bonne approche pour minimiser l'exploitation des ressources de la station mobile. [Tis 99].

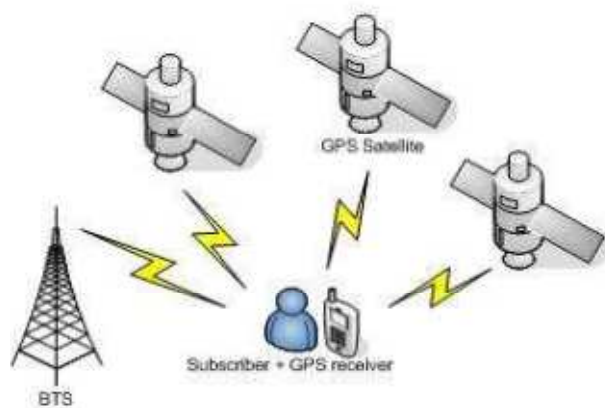


Figure 3: Assisted global positioning system (A-GPS)

b) Angle d'arrivée (Angle of Arrival – AOA) :

Dans ce cas, la position du mobile est déterminée par triangulation, en mesurant l'angle d'arrivée du signal d'au moins trois BTS.

Cette méthode peut être appliquée dans un environnement sans obstacles.

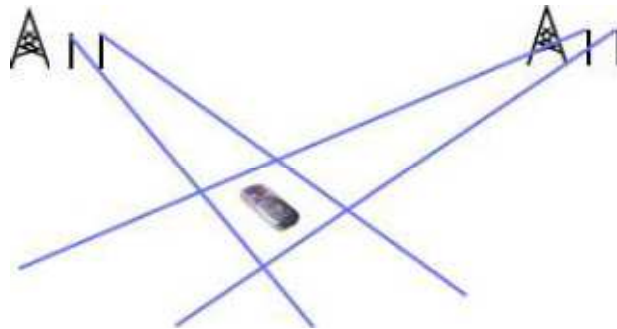


Figure 4: Localisation par AOA

c) TOA ou triangulation réseau (Time Of Arrival) et TDOA (Time Difference Of Arrival) :

TOA localise l'utilisateur dans un cercle qui a pour centre le BTS et dont le rayon est la distance qui sépare le BTS de la station mobile. Etant donné que la vitesse² d'une onde radio est connue, TOA calcule le temps de transmission entre la station mobile et le BTS et vice versa, il est donc possible d'estimer la distance qui sépare la station mobile du BTS.

TDOA situe la station mobile à l'intersection de trois BTS, par mesure du temps de transmission entre la station mobile et les BTS simultanément. [Tis 99].

² Vitesse : se propageant à 300 000 km/s dans l'air

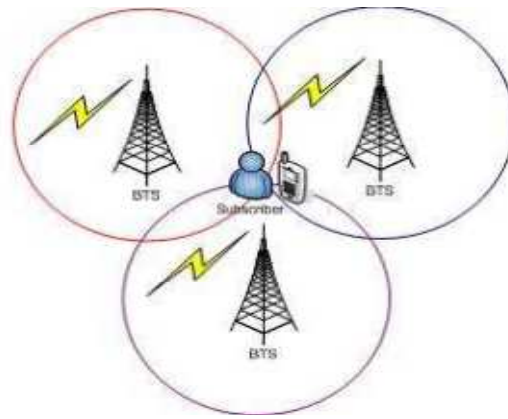


Figure 5: TDOA (Time Difference Of Arrival)

Cell-ID :

Cette méthode est basée sur le réseau GSM, détermine la position de l'utilisateur en récupérant l'identifiant de la cellule dans laquelle il est enregistré. Un centre de positionnement traite les informations et renvoie la position à la station mobile.

Dans cette méthode la précision dépend de la portée des cellules ainsi que sur l'environnement (rural ou urbain). [Tis 99].

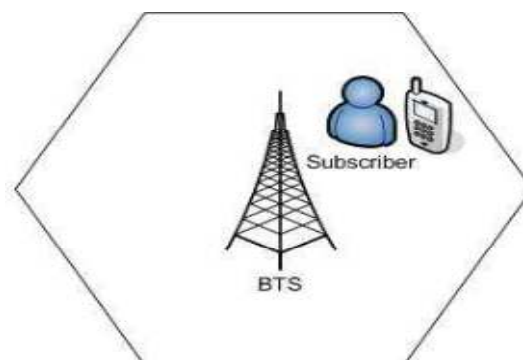


Figure 6:Cell-id

d) Transfert intercellulaire : Handover

Le *Handover*, également appelé *itinérance (roaming)* permet à un terminal mobile de changer de cellule sans perdre la connexion au réseau. Ainsi, le client peut se déplacer tout en continuant d'utiliser son équipement mobile.

Le *handover horizontal* concerne le passage d'une cellule à une autre du même type (ex : GSM vers GSM). Tous les systèmes actuels incluent le handover horizontal.

Le *handover vertical* permet le passage d'une cellule d'un type à une autre de type différent (ex : GSM vers Wi-Fi). [Dew11].

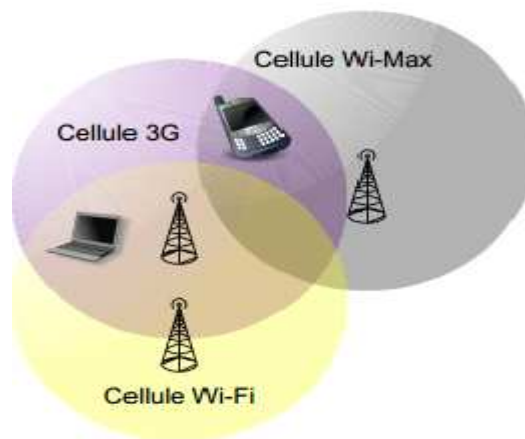


Figure 7:Handover

1.3.3. Géolocalisation par Géocodeur

Le géocodage permet d'affecter des coordonnées géographiques (la longitude et la latitude) à une adresse. Ces données géographiques vont permettre de positionner chaque adresse sur une carte numérique. Ce principe de géocodage est automatisé à l'aide d'un géocodeur qui peut être sous forme de logiciel ou de service en ligne.

L'intérêt de cette technique est de pouvoir traiter géographiquement l'ensemble des informations de la base de données. Ainsi, on peut regrouper plus facilement l'entièreté de ses informations clients, prospects et concurrents sur une même carte.

Cependant, la technique de géocodage a comme inconvénient son manque de précision. [TVi09].

1.3.4. Géolocalisation par adresse IP (sur internet)

Cette méthode permet de déterminer la position géographique d'un ordinateur ou de n'importe quel terminal connecté à internet en se basant sur son adresse IP. Les adresses IP sont gérées par l'IANA³, une organisation qui s'occupe de découper les blocs d'adresses IP disponibles et de les distribuer de façon très contrôlée aux pays qui en demandent. Toutes ces attributions étant très bien documentées, il est possible de savoir dans quel pays se trouve un terminal connecté à internet grâce à son adresse IP.

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. On peut citer comme exemple Lookup City (http://www.maxmind.com/app/lookup_city) qui est un outil en ligne effectuant cette géolocalisation [TVi09].

1.4. Géolocalisation en intérieur

Afin d'obtenir des précisions de l'ordre du mètre ou voire meilleures, plusieurs technologies de proximité ont été explorées. La proximité de tous les éléments du système (mobile, stations de bases du réseau) permettent d'atteindre une précision métrique. L'émergence des nouveaux réseaux sans fil est une des solutions à envisager pour se localiser à l'intérieur des bâtiments.

1.4.1. Géolocalisation par RFID

L'identification par radiofréquence ou RFID (Radio Frequency Identification) permet d'identifier à distance des objets ou des individus, à l'arrêt ou en mouvement, et d'échanger avec eux des données en fonction des applications envisagées. Cette identification est rendu possible grâce à une étiquette émettant des ondes radio attachées à ce dernier

L'IANA³ : Internet Assigned Numbers Authority) est une organisation américaine dont le rôle est la gestion des noms de domaines de niveau supérieur (TLD), de l'espace d'adressage IP d'Internet, et des autres ressources partagées de numérotation requises soit par les protocoles de communication sur Internet, soit pour l'interconnexion de réseaux à Internet.

1.4.1.1. Composants et fonctionnement d'un système RFID

Un système RFID est composé d'étiquettes, de lecteurs, d'encodeurs et de middleware. Ce dernier joue le rôle d'intermédiaire entre le système (les composants physique) et le système d'information de l'entreprise. [Hau]

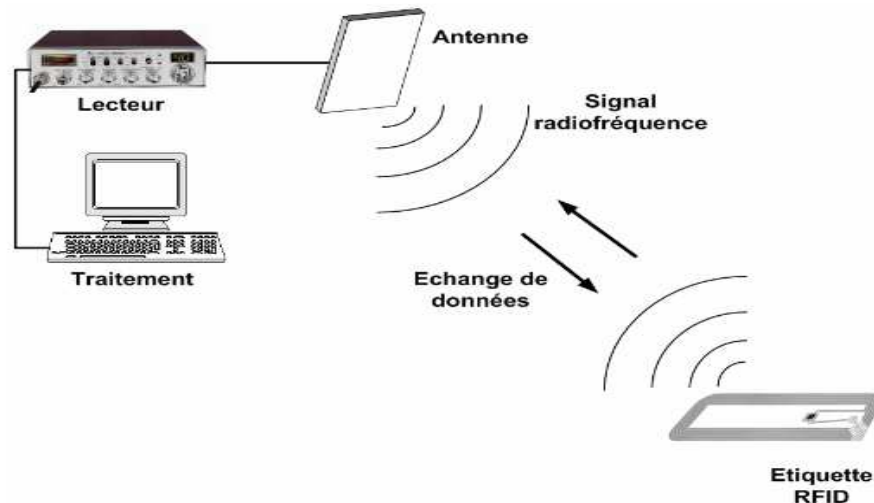


Figure 8:Schéma général d'un système d'identification par radiofréquence

- a) *Le lecteur* : également appelé *scanner*, qui envoie une onde électromagnétique porteuse d'un signal en direction des objets à identifier ou à contrôler. En retour, le lecteur reçoit l'information renvoyée par ces objets.
- b) *Etiquette* : en anglais « tag » fixée sur ces objets, qui réagit à la réception du signal envoyé par le lecteur en renvoyant vers ce dernier l'information demandée.
- c) *un ordinateur* : de stockage et de traitement des informations recueillies par le lecteur.

On appelle (RFID Tag Ou RFID Label) l'encapsulation d'une puce (chip) abritant un numéro de série ou une suite de données et une petite antenne reliée à cette dernière. Le format Des données Inscrites sur les étiquettes est standardisé à l'initiative

d'EPC Global (Electronic Product Code). Ces tags sont collés ou incorporés à des objets ou sur des produits. [Hau]

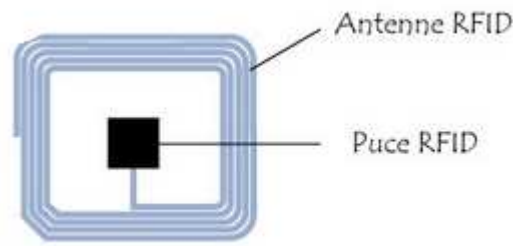


Figure 9:TAG (ETIQUETTE)

Dans un système RFID on retrouve deux types de localisation :

- 1) *Localisation passive* : un lecteur RFID est placé sur certains points bien définis (Pass Navigo par exemple). Dans ce cas, la position de l'objet est liée à celle du lecteur.
- 2) *Localisation active* : La puce émet d'elle même un signal à une ou plusieurs antennes proches. On utilise alors les techniques CellID, AoA, ToA, DToA vues dans le GSM. [Dew11]. Et par conséquent, on distingue :
 - a) *les puces passives* (beaucoup plus répandues) qui envoient des informations à la sollicitation d'un lecteur qui en s'approchant crée un champ électromagnétique.
 - b) *Les puces actives* dotées de micro batteries capables de fournir une alimentation électrique et donc d'envoyer des informations par elles mêmes.

On retrouve, donc ce type de puces sur des types d'objets extrêmement variés (étiquettes antivol, clés de voitures, passes pour les garages ou les transports en commun, passeports, ...).

1.4.2. Géolocalisation par INFRAROUGE

Le système Active Badge est l'un des premiers systèmes de localisation en intérieur, proposé par AT&T entre 1989 et 1992 [Cam12].

Dans ce type de système, le mobile doit être doté d'un tag émettant des signaux infrarouges toutes les 10 secondes. Les récepteurs installés au plafond de chaque pièce de la zone à couvrir sont reliés entre eux pour former un réseau permettant de détecter le tag actif, on parle de localisation par zone. Cette technologie était adoptée autrefois parce qu'elle était peu coûteuse et une bonne précision a été obtenue lors de son exploration. En revanche, elle présente d'énormes inconvénients liés aux ondes radio qui sont perturbés par la présence de lumière et des obstacles : murs, meubles lourds, etc, ce qui réduit la portée du système. Le fonctionnement de ce système et l'exploitation des données remontant de capteurs Infrarouge est semblable à celle du système RFID (Radio Frequency Identifier).

1.4.3. La localisation par vidéo

La vidéo est un mécanisme permettant de détecter la présence d'un élément, de le localiser dans une scène. La localisation est effectuée grâce à des transformations entre l'image de la scène et les angles de vues de la caméra [SBFC03].

La détection d'intrusion est possible avec cette technologie, grâce aux techniques de reconnaissance de contours, un objet est repérable sur une image. Il est possible de suivre le déplacement de ce contour tant qu'il reste dans le champ de vision de la caméra.

Ce système est utilisé notamment en robotique, la nouvelle génération a gagné en autonomie grâce aux systèmes de vision. Ces robots peuvent se repérer dans l'espace et donc se déplacer. La faiblesse de ce système réside dans sa portée limitée, elle se retrouve restreinte à une seule pièce dans les environnements indoor. De plus le problème d'identification est à soulever : Ce problème n'est pas négligeable car les applications requièrent, en plus de la position d'un mobile, un identifiant permettant de le distinguer. Or avec cette technologie, différencier deux objets mobiles n'est pas simple. Lorsqu'un objet masque un autre pendant un bref instant, le système de détection par vidéo conclut qu'il n'y a qu'un seul objet dans la scène.

1.4.4. Localisation par Wi-Fi

1.4.4.1. Définition :

Wi-Fi signifie **Wireless Fidelity**. C'est un standard de réseau local sans fil obéissant à la norme 802.11b (et maintenant 802.11g qui est compatible avec la précédente). Il s'agit d'un réseau haut débit dont la couverture se veut "large " à l'instar de la téléphonie mobile et est prévu pour une connexion à Internet [Say 09].

1.4.4.2. La Technologie Wi-Fi norme IEEE 802.11:

La norme **802.11** s'attache à définir les couches basses du modèle OSI pour une liaison sans fil utilisant des ondes électromagnétiques, c'est-à-dire :

- *La couche physique* : définit la modulation des ondes radioélectriques et les caractéristiques de la signalisation pour la transmission des données.
- *La couche liaison de données* : constituée de deux sous-couches : le contrôle de la liaison logique (Logical Link Control, ou LLC) et le contrôle d'accès au support (Media Access Control, ou MAC). Elle définit l'interface entre le bus de la machine et la couche physique, notamment une méthode d'accès proche de celle utilisée dans le standard Ethernet et les règles de communication entre les différentes stations.

1.4.4.3. Différentes norme de wifi

La norme IEEE⁴ 802.11 est en réalité la norme initiale offrant des débits de 1 ou 2 Mbps. Des révisions ont été apportées à la norme originale afin d'optimiser le débit (c'est le cas des normes 802.11a, 802.11b et 802.11g, appelées normes 802.11 physiques) ou bien préciser des éléments afin d'assurer une meilleure sécurité ou une meilleure interopérabilité. La logique aurait voulu un ordre alphabétique.

IEEE⁴ : 'Institute of Electrical and Electronics Engineers: est un comité de l'IEEE qui décrit une famille de normes relatives aux réseaux locaux (LAN) et métropolitains (MAN) basés sur la transmission de données numériques par le biais de liaisons filaires ou sans fil

1.4.4.4. Les équipements Wi-Fi

Il existe différents types d'équipements pour la mise en place d'un réseau sans fil Wifi

- **Les adaptateurs sans fil ou cartes d'accès :** En anglais *wireless adapters* ou *network interface controller*, noté NIC. Il s'agit d'une carte réseau à la norme 802.11 permettant à une machine de se connecter à un réseau sans fil. Les adaptateurs WiFi sont disponibles dans de nombreux formats (carte PCI, carte PCMCIA, adaptateur USB, carte compactflash, ...). On appelle **station** tout équipement possédant une telle carte.
- **Les points d'accès :** Notés **AP** pour **Access point**, parfois appelés bornes sans fil, permettant de donner un accès au réseau filaire (auquel il est raccordé) aux différentes stations avoisinantes équipées de cartes WiFi. Cette sorte de hub est l'élément nécessaire pour déployer un réseau centralisé en mode infrastructure. Certains modèles proposent des fonctions de modem ADSL et comprennent plus ou moins de fonctions comme un pare-feu.

1.4.4. 5. Architecture wifi :

Wi-Fi est fondé sur une architecture cellulaire. Cette architecture peut s'apparenter à celle utilisée dans la téléphonie mobile, où des téléphones mobiles utilisent des stations de base pour communiquer entre eux. Il est composé de :

- ❖ Un ou plusieurs **point d'accès** auxquels un certain nombre de station de base équipée de carte Wi-Fi s'associent pour changer des données. Son rôle est d'unifier tout le réseau et à servir de pont entre les stations du réseau et un réseau extérieur.
- ❖ **La taille du réseau** dépend de la zone de couverture du point d'accès, aussi appelée cellule. Cette zone peut varier dans le temps, car le fait d'utiliser les ondes radio ne permet pas de couvrir constamment une même zone.
- ❖ La **cellule** constitue l'architecture de base de Wi-Fi, appelée **BSS** (Basic Service Set), ou ensemble de services de base.

La norme 802.11b définit deux modes de fonctionnement : un mode avec infrastructure, et un mode sans infrastructure (ou ad hoc)

Mode avec Infrastructure :

Les nœuds du réseau communiquent entre eux via un *point d'accès*, aussi appelé *base*, qui peut être relié à un réseau filaire.

L'ensemble formé par le point d'accès et les stations situées dans sa zone de couverture qui constitue une cellule.

Ainsi, une base peut communiquer directement avec les stations mobiles se trouvant dans sa cellule. Bien entendu, les stations mobiles ne peuvent communiquer entre elles, et sont obligées de passer par la base pour toute communication avec d'autres stations de la cellule.

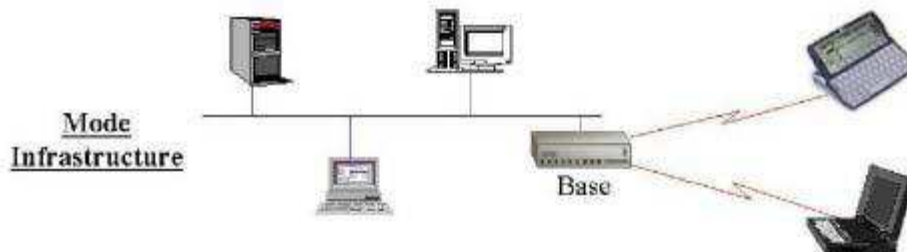


Figure 10:réseau Wi-Fi avec infrastructure

Dans ce mode :

- Le débit est partagé par les stations (minimum des débits des stations divisé par le nombre de stations).
- Les communications entre deux stations d'un BSS doivent passer par le point d'accès.
- Un ESS = ensemble de BSS connectés par un système de distribution (DS)
- Le DS le plus utilisé est Ethernet mais ça peut être un autre réseau Wi-Fi.
- Si les BSS se recouvrent, il peut y avoir un service de mobilité (802.11f).

Mode sans infrastructure

Pour ces réseaux là, les unités mobiles se connectent les unes aux autres à travers une interface radio afin de constituer un réseau point à point (**Figure 11**) :

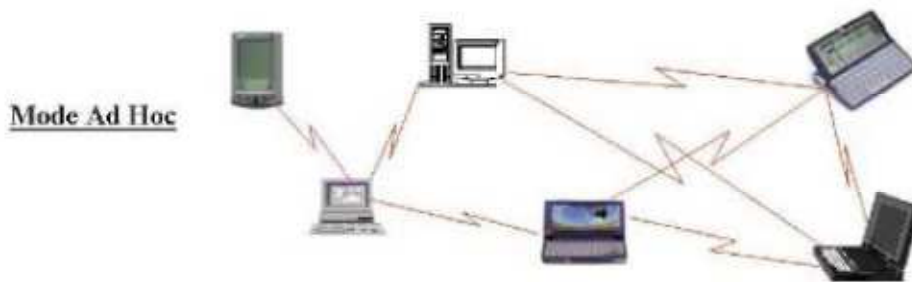


Figure 11:Architecture ad hoc

1.4.4.6. Caractéristiques d'une cellule Wi-Fi

- **SSID (Service Set Id) :**
 - Analogue à un nom de réseau, attribué à une ou plusieurs cellules Wi-Fi.
 - Configuré au niveau des points d'accès et des stations en mode infrastructure et seulement au niveau des stations pour le mode ad-hoc.
 - Il est envoyé en clair, mais peut aussi être caché, pour se connecter il faut le spécifier.
 - Si SSID=Any, il accepte toutes les connexions même si on ne spécifie pas le nom de réseau.
- **Sécurité :**
 - Optionnelle dans Wi-Fi.
 - Utilise une clé secrète partagée (symétrique) par le point d'accès et les stations.
 - Utilise l'authentification et le chiffrement des données.
- **Canal de transmission :**
 - 14 canaux pour 802.11b et g, 8 pour 802.11 a .
 - En mode infrastructure, seul le point d'accès peut définir le canal.
 - En mode ad-hoc, la 1ère station définit le canal.
- **Débit :**
 - 1, 2, 5.5, 11, 54 M bit/s.
 - Lorsqu'une station s'éloigne du point d'accès, son débit chute.
 - Plus le débit est faible, plus la zone de couverture est importante.
 - Une zone de couverture faible est un paramètre de sécurité.

- **Puissance :**
 - Définit la portée en émission des ondes.
 - La portée augmente avec la puissance.

2 .Les méthodes et techniques de positionnement en intérieur :

Les méthodes de géolocalisation peuvent être divisées en trois catégories [Say09] :

- **Méthodes géométriques** : utilisant les théorèmes géométriques sur les relations dans les triangles en particulier,
- **Méthodes statistiques** : méthode d'empreinte radio (radio fingerprinting) par exemple,
- **Méthodes hybrides** : utilisant une combinaison de méthodes géométriques et statistiques

I.2.1. Méthode géométrique

Ce sont les méthodes qui se basent sur les théorèmes géométriques pour le calcul des distances et/ou des angles.

2.1.1. Trilatération

Cette méthode permet de positionner un Objet Mobile en 2D à l'aide de trois Points de Références. Elle consiste à mesurer les distances entre les nœuds de références et le nœud que l'on cherche à positionner en utilisant une méthode à base de temps d'arrivée et une à base de mesure de puissance de signal. On trace ensuite un cercle autour de chaque Point de Référence dont le rayon est égal à la distance mesurée entre le Point de Référence et l'Objet Mobile. La position de l'Objet Mobile se situe à l'intersection de ces trois cercles.

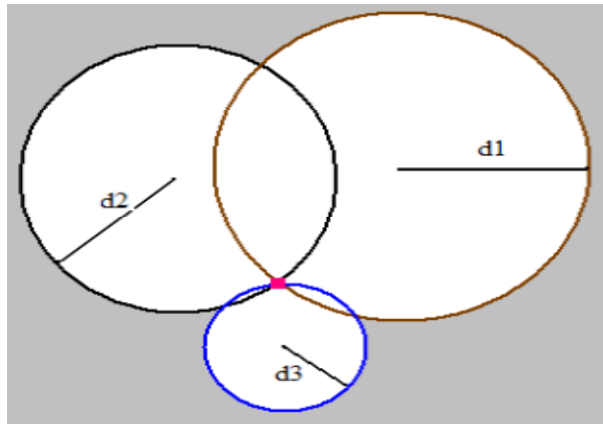


Figure 12:Trilatération

2.1.2. Triangulation

Elle utilise les propriétés géométriques du triangle pour pouvoir positionner un nœud à partir de deux nœuds de référence. Il utilise pour cela : **[Dou]**

- la loi des sinus
- le théorème d'Al Kashi

Soit un triangle ABC, dans lequel on utilise les notations usuelles exposées sur la figure13 : d'une part α , β et γ pour les angles et, d'autre part, a, b et c pour les cotes respectivement opposés à ces angles.

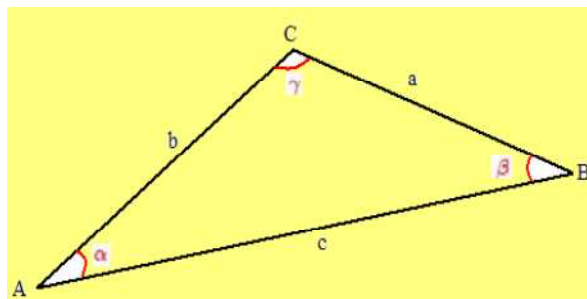


Figure 13:Triangle

A partir de ces relations et de deux nœuds de référence, on peut obtenir la position d'un troisième nœud (bateau par exemple (figure14)). Dans le cas du positionnement d'un bateau, on utilise en effet, la distance AC entre les deux balises puis on mesure les angles A et C. Ensuite, il existe plusieurs solutions pour arriver à déterminer les deux distances AB et BC qui nous manquent.

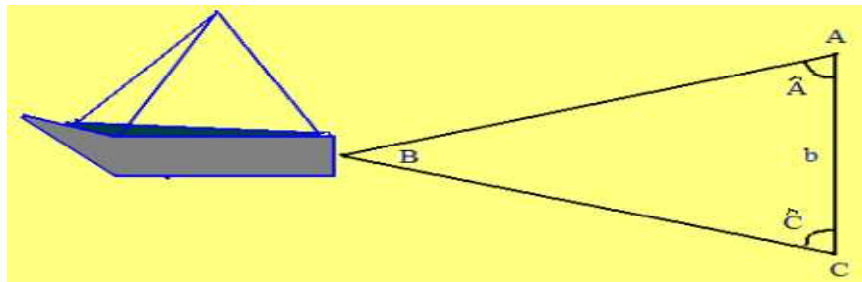


Figure 14: Triangulation

I.2.2. Méthode Statistique

Ce sont les méthodes qui consistent à effectuer des mesures « références » puis à comparer les mesures réalisées pour l'Objet Mobile avec ces mesures de références pour en déduire la position de ce dernier. [Say09]

Ces approches comprennent donc deux étapes : une étape de *calibrage* ou d'apprentissage et l'étape *d'estimation de la position*. Une des méthodes les plus efficaces est la méthode d'empreinte radio (en anglais RF *fingerprinting*), qui se base sur les mesures des puissances reçues (RSSI pour Received Signal Strength Indicator) par l'objet mobile à partir de balises radios (comme les points d'accès WiFi).

2.2.1. Empreinte radio

La technique dite du "RF fingerprinting", se fait en deux phases :

- 1) **Phase de formation** : appelée aussi *calibrage*, consiste à relever les caractéristiques (fingerprint) des signaux aux différents endroits du bâtiment, et les enregistrer dans une base de données pour pouvoir être exploitées par la suite dans une phase de positionnement. (figure15)

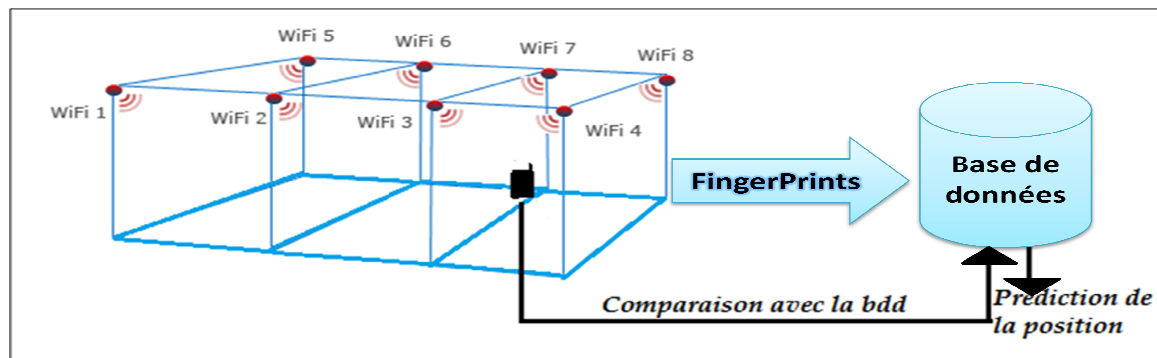


Figure 15:localisation par fingerprint dans un réseau wifi

2) **Phase de positionnement** : appelée aussi *localisation*, consiste à déterminer la position de l'Objet Mobile en comparant les RSS mesurés par l'Objet Mobile avec les RSS sauvegardés dans la base de données (appelée carte radio ou map). Plus le nombre de balises radios augmente, plus la précision s'améliore ; plusieurs mesures successives de puissance doivent être réalisées pour une meilleure évaluation de la position, les ondes radios étant affectées par plusieurs facteurs.

I.3. Les techniques de Mesure

La plupart des systèmes de localisation pour les réseaux sans fil reposent sur au moins une de ces techniques. Afin de calculer les distances ou les angles, plusieurs paramètres des communications sans fil sont utilisés.

I.3.1. Temps d'Arrivée (ou ToA pour Time of Arrival) :

Cette technique se base sur le temps de propagation du signal.

- ✓ La source envoie un signal daté au récepteur qui date à son tour son heure d'arrivée,
- ✓ Le système de géo-localisation se basera sur ces informations pour en déduire la distance émetteur-récepteur en supposant le plus souvent que la propagation se fait en ligne directe.
- ✓ Une synchronisation complète entre l'émetteur et le récepteur pour un calcul précis.

I.3.2. Différentiel de temps d'Arrivée(ou TDoA pour Time Difference of Arrival) :

- ✓ Le principe est similaire au ToA, en revanche, pas de forte synchronisation entre l'émetteur-récepteur,
- ✓ En utilisant une source quelconque qui ne date pas son émission. Le signal est alors reçu par les récepteurs,
- ✓ Le système de géo-localisation détermine la position de la source en fonction de la différence des temps d'arrivée sur les récepteurs.

Cette technologie exige de même une horloge très précise et très bien synchronisée au niveau des récepteurs.

I. 3.3. Angle d'arrivée (ou AoA pour Angle of Arrival)

- ✓ La position du mobile est déterminée par triangulation,
- ✓ En mesurant l'angle d'arrivée du signal d'au moins trois nœuds distincts,
- ✓ Cela nécessite cependant un matériel spécifique car capable d'évaluer l'incidence.
- ✓ Cette méthode peut être appliquée dans un environnement sans obstacle.

I. 3.4. Puissance du signal reçu (RSS) :

- ✓ Appuyé sur des modèles de propagation du signal et sur les mesures d'énergie d'un signal reçu en plusieurs points fixes, les distances entre le mobile et les références sont estimées.
- ✓ La position du mobile est ensuite calculée, par exemple par triangulation. Cette technique dépend de l'environnement dans lequel le signal est propagé.
- ✓ La précision de localisation est d'autant plus fine que la modélisation de la propagation du signal dans un milieu est détaillée.

I.4. Collecte de mesure des caractéristiques des signaux

La collecte de données établit une base de données des observations physiques du réseau sans-fil, par des mesures de puissance du signal, de rapport signal sur bruit ou toute autre grandeur mesurable permettant de quantifier le signal. [Las09]

Il ya deux modèles :

1. Cartographie des puissances
2. Modèle de propagation des ondes

I.4.1. Cartographie des puissances

Basée sur l'enregistrement de mesure de puissance des signaux reçus dans une base de données qui contient les mesures en relation avec leur position en formant un point de référence. Par comparaison, de nouvelles valeurs mesurées avec le contenu de la base de données, on pourra déterminer la localisation d'un terminal mobile. Alors que, le système RADAR propose une comparaison déterministe, et celles, basées sur des méthodes probabilistes.

Approche Déterministe

Cette approche constitue la base de données de référencement avec une information de puissance moyenne.

La géo-localisation est ensuite effectuée par comparaison des mesures en temps réel avec les informations contenues dans la base de données. Tous les systèmes de géo-localisation déterministes stockent au moins les coordonnées physiques et les moyennes des mesures de puissance pour chaque point d'accès du système.

La base de donnée obtenue est composée d'une suite d'enregistrements de la forme $(p, ss_1, \dots, ss_N)$ relatifs chacun à un point de référence. Ainsi que, peut également contenir d'autres informations relatives à la qualité du signal ou au positionnement des terminaux mobiles.

Où :

P : sont les coordonnées géographiques du point.

$(ss_1, \dots, ss_N)$: est l'ensemble des moyennes de puissance mesurées pour les N point d'accès du système.

Approche Probabiliste :

Les approches probabilistes consistent à utiliser un ensemble de valeurs de SS pour chaque point de référence. Les valeurs sont représentées par une distribution. Généralement, une distribution suivant une loi normale est choisie.

L'idée de l'utilisation de plusieurs mesures est d'absorber les variations inévitables des mesures par la quantité. De cette façon, les mesures utilisées sont plus fiables car la redondance des échantillons de mesures réduit les effets des interférences en produisant un ensemble de mesures réparties selon leurs probabilités d'être observées.

I.4.2. Modèle de propagation des ondes

La seconde famille de techniques de géo-localisation s'appuie sur la modélisation de la propagation des ondes radio et, plus particulièrement de l'atténuation. Cette dernière permet la détermination de la distance entre le terminal mobile dont on doit déterminer la position et les différents points dont les coordonnées sont connues. La position peut ensuite être résolue analytiquement grâce à la trilatération.

I.5. Algorithme de calcul de position

Le serveur de calcul met en œuvre plusieurs algorithmes de calcul de la position du mobile, qui se séparent en deux catégories : les algorithmes «simples », qui calculent la position en fonction des seules données instantanées, et ceux qui tiennent compte du parcours antérieur effectué par le mobile pour affiner la position calculée ou sélectionner la plus pertinente possibilité parmi plusieurs.

Certains de ces algorithmes utilisent une cartographie des puissances, d'autres un modèle de propagation, ou encore une combinaison des deux. [TVi09]

I.5.1. Algorithmes instantanés

- RADAR [Eve07], qui utilise une cartographie des puissances. Il sélectionne, dans l'espace des puissances du signal, le point le plus proche des puissances des signaux reçus du mobile.
- Une adaptation de la formule de Friis [Las09]. Le calcul est effectué par multilatération, en prenant comme points de référence les coordonnées des Point d'accès.
- FBCM (Friis-Based Calibrated Model), qui est une modification dynamique de la formule de Friis, adaptée à un bâtiment donné grâce à des points de calibration (plutôt qu'à un milieu hétérogène générique). Le calcul est effectué de la même façon, par multilatération.
- FRBHM (FBCM and Reference-Based Hybrid Model) basique, qui combine une cartographie des puissances avec le FBCM. Comme pour RADAR, le point le plus proche dans l'espace des puissances est sélectionné, puis il est modifié à l'aide de la formule d'atténuation obtenue par le FBCM pour le bâtiment où se déroulent les expérimentations.

I.5.2. Algorithmes tenant compte du parcours :

- RADAR⁵. [Eve07]
- FRBHM avec prise en compte de la topologie
- FRBHM continu, dont le fonctionnement est similaire au précédent, à la différence que les points les plus pertinents sont tous altérés par FBCM.

I.6. Applications et services basées sur la géolocalisation

La possibilité de connaître la position de l'utilisateur permet de développer différentes applications. En réalité leur nombre peut être très grand et les possibilités illimitées. [MC03]

- **Le service itinéraire** : Les abonnés peuvent demander les directions rue par rue (en texte) pour leurs déplacements à pied, en voiture, à bicyclette, en bus, en métro ou en train.
- **Les services de proximité** : permettent aux abonnés de rechercher et/ou de repérer dans leur voisinage des points d'intérêt public tels que les parkings, arrêts de bus, musées, monuments, services publics, etc.
- **Le service de trafic** : La carte de trafic routier aide les abonnés à éviter l'encombrement en affichant l'information à jour du trafic routier sur des cartes prédéfinis.
- **Service de secours** : Ce service permet à un abonné en difficulté de faire appel en cas d'urgence à un service capable de le localiser et de lui apporter l'aide nécessaire. Cette aide peut couvrir les besoins suivants : dépannage auto ; urgence médicale ; police secours.
- **La Recherche** : On se trouve dans une région inconnue, on décide de se rendre dans un restaurant dont on connaît le nom (ou le début de nom). La recherche permet alors de retrouver l'adresse exacte de ce lieu s'il est contenu dans la base de données. Après avoir retrouvé l'endroit recherché, il est possible de consulter l'itinéraire.

RADAR⁵ : est un système qui utilise les ondes radio pour détecter la présence ainsi que déterminer la position et la vitesse d'objets tels que les avions, les bateaux, ou encore la pluie.

Conclusion

Le long de ce chapitre, nous avons étalés les différentes techniques de localisation. Plusieurs d'entre elles ne resteront que des essais sans une vraie rentrée dans le monde technologique. Nonobstant cela, la diversité de ces dernières permet à la société scientifique de migrer vers de nouvelles techniques plus perfectionnées.

En passant vers la description de la nouvelle approche symbolique, nous avons préparé le terrain pour une étude plus approfondie. Ayant apprivoisé les concepts théoriques de celle-ci, nous allons traiter les choix adoptés pour cette application dans le chapitre succédant à travers, un processus de développement, et spécifications des besoins.

Introduction

Le chapitre précédent nous a servi d'un tour d'horizon des différents procédés de localisation en intérieur. Ainsi, nous consacrons à présent le chapitre en cours à la présentation des fondements qui serviront comme base au développement de notre système de localisation.

Il a été nécessaire dans un premier temps d'effectuer ses choix techniques: méthode et algorithme de localisation, et ceci en respectant les besoins attendus de notre système en terme de fonctionnement et de performances. Dans cette partie, nous soulèverons les différentes phases du processus de développement et la démarche à suivre.

II.1 Démarche de conduite du projet de développement

Une bonne conduite du projet nécessite plusieurs critères. La garantie de cette dernière s'effectue, à travers une bonne spécification des besoins, une minutie des enjeux et une estimation des gains potentiels.

Ce projet de fin d'étude comporte plusieurs parties, permettant d'atteindre l'objectif tracé. Nous allons étaler cela en utilisant l'organigramme suivant :

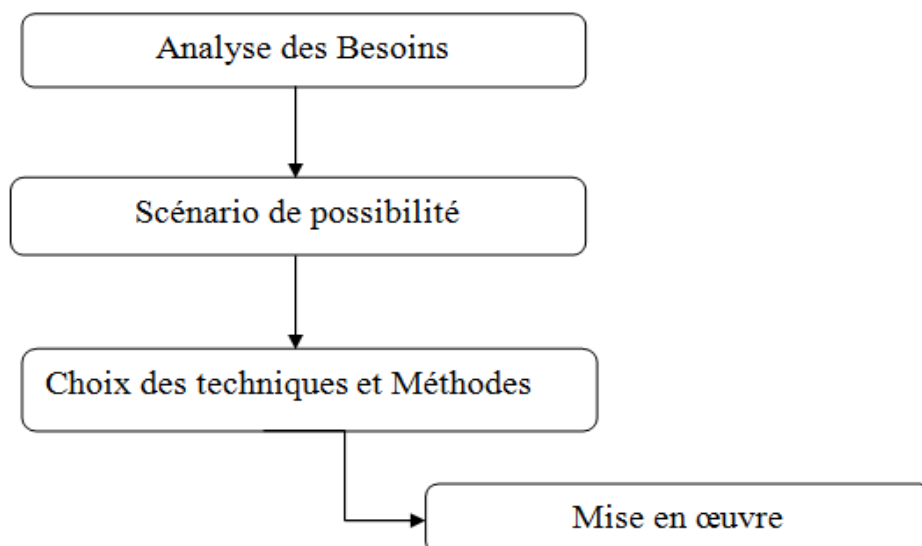


Figure 16: Organigramme des étapes de la démarche du projet

La géolocalisation par wifi connaît un essor considérable dû au fait que les réseaux wifi sont présents et très déployés, mais ils sont également soutenus par le développement très rapide de l'informatique mobile.

II.2. Informatique mobile et système d'exploitation android

II.2.1. L'informatique Ubiquiste

Le terme ubiquitous computing (informatique ubiquitaire) a été introduit par WEISER M. [Wei91]. Il désigne une vision de l'ordinateur dont l'ancrage dans le quotidien serait tel qu'il rendrait continuellement des services indispensables sans que personne ne remarque sa présence.

Elle peut être définie aussi comme étant la tendance à l'informatisation et la miniaturisation des dispositifs électroniques en les intégrant dans tout objet du quotidien.

Cette informatique s'est rapidement développée grâce au progrès technique des systèmes de communication (réseaux sans fil) et de la microélectronique.

L'informatique ubiquitaire a donc pour but de rendre accessible toutes sortes de services, n'importe où, tout en masquant l'ordinateur. Cette volonté d'affranchir l'utilisateur des contraintes actuelles d'utilisation d'un ordinateur (être assis devant un clavier, un écran, etc.) lui rend sa liberté d'actions, notamment sa liberté de mouvement. L'ubiquité permet donc souvent la mobilité.

II.2.2. L'informatique mobile

2.2.1. Définition :

La mobilité caractérise ce qui peut se mouvoir ou être mû, ce qui peut changer de place, de position. Cette notion intuitive de l'activité humaine s'est traduite par la création de dispositifs accompagnant la personne dans ses déplacements, tout en lui offrant des services de qualité.

2.2.2. Présentation :

L'informatique mobile s'inscrit sous l'approche globale de l'informatique ubiquitaire, elle constitue un outil de communications très puissant pour les entreprises et pour le grand public.

Les supports qui lui ont été développés ont repris l'industrie sans fil. Il s'agissait des entités mobiles communicantes, de petites tailles permettant de se connecter aux différents types de réseaux tant qu'ils disposent de périphériques dédiés. Les Smartphones en font partie.

2.2.3. Les smartphones

Un Smartphone est un téléphone « intelligent », appelé aussi ordiphone, dédié aux communications mobiles, disposant d'un système d'exploitation et adoptant des applications tierces développées par le fabricant, par l'opérateur ou par un éditeur de logiciel.

Ce téléphone fournit un module de radiocommunication (pour la voix et l'échange de données), des fonctionnalités bureautiques (agenda, calendrier, navigation web, email, messagerie instantanée, GPS, etc.) ainsi que, des fonctionnalités multimédia (photo, musique, vidéo et jeux).

En effet, les Smartphones sont dotés d'un système d'exploitation(SE) qui gère leurs fonctionnalités.

II.2.3. Les systèmes d'exploitation Mobiles

Actuellement, il existe plusieurs SEs pour mobiles, on retrouve : Windows Mobile, Palm OS, Symbian, BlackBerry qui sont des systèmes propriétaires. De plus, il y a les plateformes libres au code source ouverts comme Moblin.org, Ubuntu MID Edition, Android, etc.

Chaque type de SE a ses avantages et ses limitations.

Aujourd'hui, le marché de la téléphonie mobile est partagé entre cinq grands acteurs [Gar 11] :

- **Nokia** et son système **Symbian**, encore leader mais en franche dégringolade, au point de s'être associé récemment à Microsoft.
- **Google** et son système **Android**, qui explose littéralement les compteurs et prend la seconde place.
- **Research in Motion**, plus connue pour ses **Blackberry**, un peu à la traîne
- **Apple** et son système **iOS**.
- **Microsoft** et son système **Windows Phone**, qui a encore du chemin à faire.

Company	2010 Units	2010 Market Share (%)	2009 Units	2009 Market Share (%)
Symbian	111,576.7	37.6	80,878.3	46.9
Android	67,224.5	22.7	6,798.4	3.9
Research In Motion	47,451.6	16.0	34,346.6	19.9
iOS	46,598.3	15.7	24,889.7	14.4
Microsoft	12,378.2	4.2	15,031.0	8.7
Other Oss	11417.4	3.8	10432.1	6.1
Total	296,646.6	100.0	172,376.1	100.0

Source: Gartner (February 2011)

Figure 17: Le marché des smartphone

Une étude menée par le magazine PCWorld comparant le degré de développement de : Windows Mobile, Android et iOS, est résumée dans le tableau suivant:



	Windows Phone 7	iOS (iPhone)	Android
Developer	Microsoft	Apple	Google
Copy/Paste	✗	✓	✓
Multitasking	✗	✓	✓
Flash Support	✗	✗	✓
Silverlight Support	✗	✗	✗
HTML5 Support	✗	✓	✓
Unified Inbox	✗	✓	✓
Exchange Support	✓	✓	✓
Threaded Email	✗	✓	✓
Visual Voicemail	✗	✓	✓
Video Calling	✗	✓	✓ Third Party App
Universal Search	✗	✓	✓
Internet Tethering	✗	✓	✓
Removable Storage	✗	✗	✓
Facebook Integration	✓	✗ (Third Party App)	✓ (Third Party Integration)
Twitter Integration	✗	✗ (Third Party App)	✓ (Third Party Integration)
Folders	Hubs	✓	✓
Apps Organization	Alphabetical	Customizable	Customizable
App Store	1,000+ Apps	300,000+ Apps	90,000+ Apps
Microsoft Office Support	Built-In	Third Party App	Third Party App
Widgets	Tiles on Home Screen	✗	✓
Media Sync	Zune Software Mac & PC	iTunes Mac & PC	Direct File Transfer + Third Party Software
X-Box Live Integration	Built-In	Via Third Party App	Via Third Party App

Figure 18: Comparaison entre os mobiles

Tous ces avantages constituent de bonnes raisons pour choisir Android dans le développement de notre application.

II.2.3.1. Caractéristiques

Le système d'exploitation Android est doté d'un ensemble de caractéristiques intrinsèques faisant de lui un terrain d'application lucratif. Les principales caractéristiques sont résumées dans le tableau :

Framework	Framework Java pour le développement d'application pour la plateforme Android
Machine virtuelle Dalvik	Machine virtuelle spécialement développée pour Android. Cette machine virtuelle permet d'exécuter les applications java développées avec le Framework.
Navigateur web	Navigateur web basé sur le moteur de rendu Webkit
Graphique	Librarie graphique 2D, librarie graphique 3D basé sur OpenGL ES 1.0. Accélération matériel possible.
Stockage	Base de données SQL : SQLite est utilisé pour le stockage des données
Média	Android supporte les formats audio/video /image suivants : MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF
Connectivité	gsm, edge, 3G, bluetooth, wifi
Support Matériel	Android est capable d'utiliser Camera, GPS, accéléromètre
environnement de développement	Android possède un environnement de développement complet contenant : un émulateur, un débogueur, un analyseur de mémoires et de performances et un plugin edipse.

Figure 19:Caractéristiques du système d'exploitation android.

II.2.3.2. Choix de la plateforme Android

Android est un système ouvert, contrairement à ses deux concurrents : Microsoft et Apple. C'est un point extrêmement important car, cela signifie que la communauté de développeurs qui contribue à son développement est naturellement plus disposée à investir sur un système ouvert que sur un système rigide et soumis à des cahiers des charges trop contraignants.

De plus, Google est sans cesse en train d'innover et de proposer de nouvelles fonctionnalités gratuites, tant sur les applications web que pour les applications mobiles. Il permet d'intégrer ces dernières dans des applications personnalisées. Tous ces avantages expliquent le nombre croissant d'applications dans l'Android Market qui selon une étude menée par Lookout et dévoilée au Mobile World Congress de Barcelone, est estimé à 127% au cours des six derniers mois. À ce rythme, l'Android Market a dépassé l'App Store à la mi-2012 [bus11].

Après avoir découvert les possibilités de développement qu'offre la plateforme Android, nous consacrons cette partie du chapitre à la présentation de l'architecture générale du système afin de spécifier les différentes entités intervenantes ainsi que, les techniques de localisation adoptées.

II.3.Objectif et choix techniques

II.3.1 Objectif

Notre objectif était de comprendre le fonctionnement du géopositionnement et de réaliser ce dossier de veille⁶ coroner par le développement d'une application.

Cette application s'est portée sur la localisation Indoor dont, le concept est d'aider un client dans sa vie de tous les jours à faire ces courses dans un hypermarché. Nous désirons l'aider à trouver son chemin en le localisant à l'aide de son appareil mobile. Et à partir de sa position, lui donner le chemin le plus cohérent pour récupérer tous les produits dont il a besoin.

En fonction de ce scénario, et de l'analyse précédente des différentes techniques de géolocalisation, la technique la plus adaptée pour réaliser ce scénario est la géolocalisation en Wifi. Il suffit de placer des points d'accès wifi dans l'hypermarché et de repérer le client à l'aide d'un téléphone portable muni du wifi.

II.3.2 Choix techniques

II.3.2.1. Méthode de localisation

Notre choix s'est porté sur le *fingerprinting*, une des techniques de localisation qui ne nécessitent pas d'informations temporelles pour fonctionner. En effet la relation distance – puissance reçue, est très dépendante de l'environnement, ce qui peut fausser les résultats dans un environnement où les signaux ne sont pas stables : par conséquent les méthodes de localisation géométrique n'ont pas été retenues.

Dans le cadre des réseaux sans fil de type 802.11, l'information disponible est une information de puissance du signal. Cette information est obtenue à partir des balises émises par les points d'accès. Aucune modification des équipements ne doit

Veille⁶ : Cette technique a pour objectif de s'informer de ce qui existe d'innovant en terme de produit et de démarche dans le domaine auquel appartient le projet que l'on souhaite réaliser.

être effectuée, car cette information est présente par défaut. Elle permet au client de détecter tous les points d'accès qui le couvrent par leur portée.

Cette technique de localisation utilise une base de données qui réalise un mapping entre des positions de l'espace et des puissances de signal qu'on peut capter en ces positions. Elle est astucieuse dans le sens où elle prend bien en compte tous les phénomènes de propagation.

L'inconvénient majeur vient de la constitution de cette base de données. Plusieurs questions viennent à l'esprit notamment sur la densité de cette base de données en fonction de l'environnement (c'est à dire le nombre de points de mesures), faut-il respecter un maillage strict ? Quelles sont les informations à garder pour constituer cette base de données ?

Pour rechercher la position, le *fingerprinting* propose plusieurs algorithmes de calcul présentant chacun des avantages et des inconvénients, notre choix pour l'algorithme des k plus proche voisins sera justifié dans ce qui suit.

II.3.2.2. Algorithme de localisation

Les performances de l'algorithme de localisation constituent un critère de premier choix.

La complexité et le temps de calcul est aussi un critère de sélection très important : l'algorithme des K plus proches voisins a une complexité⁷ de l'ordre de $O(n*m+m*K+(3*K+1))$ alors que celle de l'algorithme de Bayesienne⁸ [Ham05] a une complexité de l'ordre de $O(2m)$. Ce qui nous conduit à dire que, les méthodes probabilistes nécessitent beaucoup plus de temps de calcul que la méthode des K plus proche voisins.

Complexité⁷ : étudie formellement la *quantité de ressources* (en temps et en espace) nécessaire pour la résolution de *problèmes* au moyen de l'exécution d'un algorithme. Il s'agit donc d'étudier la difficulté intrinsèque de problèmes posés mathématiquement.

Bayesian⁸ : Terme qui a commencé d'être utilisé vers les années 1950. Ce terme fait référence à Thomas Bayes (1702 — 1761).

❖ Présentation de l'algorithme de la valeur moyenne

C'est l'algorithme le plus basique des algorithmes de localisation [Ham05], il se base sur le niveau de la puissance reçu. Il se déroule en deux phases :

1. **Phase de calcul** : durant laquelle la moyenne des puissances reçues à des points de références de coordonnées connues est calculée, et enregistrement des informations collectées dans une base de données.
2. **Phase de positionnement** : Dans cette phase, le niveau de signal est mesuré en temps réel par le mobile, et transmis à un module de traitement. Une distance Euclidienne est calculée entre les puissances reçues et celles stockées au niveau de la base de données. Le point de la base de données correspondant à la distance euclidienne minimale correspond à la position la plus vraisemblable du mobile.

Une distance est définie par la formule suivante : $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$.

Pour calculer la distance on définit :

$\mathbf{S}_m = \{\mathbf{SS}_{m0}, \mathbf{SS}_{m1}, \dots, \mathbf{SS}_{mn-1}\}$: le tuple représentant le niveau de puissance reçu des différents AP en un point dont on ne connaît pas les coordonnées. Tel que $\mathbf{SS}_{m0}$ représente la valeur moyenne du niveau de puissance reçu de l'AP0.

Soit $\mathbf{S}_i = \{\mathbf{SS}_{i0}, \mathbf{SS}_{i1}, \dots, \mathbf{SS}_{in-1}\}$ un tuple de la base de données représentant les niveaux de puissance mesurés dans une phase de calibration, $\mathbf{SS}_{i0}$ est la valeur moyenne du niveau de puissance reçu de l'AP0.

La distance euclidienne entre $\mathbf{S}_m$ et $\mathbf{S}_i$ est calculée comme suit :

$$d(\mathbf{S}_m, \mathbf{S}_i) = (\mathbf{SS}_{m0} - \mathbf{SS}_{i0})^2 + (\mathbf{SS}_{m1} - \mathbf{SS}_{i1})^2 + \dots + (\mathbf{SS}_{mn-1} - \mathbf{SS}_{in-1})^2$$

Équation 1: La distance euclidienne

Mais cet algorithme appliqué à la cartographie des puissances peut produire des distances euclidiennes égales pour des points éloignés, ou dans d'autres cas il donne carrément des résultats erronés, pour cela l'algorithme du K plus proche voisin vient combler et corriger les erreurs du premier.

❖ *Présentation de l'algorithme du K plus proche voisin*

L'algorithme des **K** plus proches voisins choisit **K** localisations qui ont les distances Euclidienne les plus faibles par rapport au point qu'on veut localiser. Ensuite, les coordonnées recherchées du point courant sont une combinaison linéaire des coordonnées des **K** points de références choisis. Le coefficient **i** de cette combinaison linéaire est d'autant plus grand que la distance euclidienne entre le point courant et le point de référence **i** est faible.

K est un entier positif.

$SSx = \{SSx_0, SSx_1, \dots, SSx_{n-1}\}$ est la valeur moyenne mesurée dans une position inconnue.

n : est le nombre de Point d'Accès qu'on utilise dans le système.

- Chercher les K enregistrements dont les valeurs moyennes des signaux

$SSi = \{SSi_0, SSi_1, \dots, SSi_{n-1}\}$ ont les distances Euclidienne les plus faibles avec SSx

- Li est un vecteur, dans notre cas à deux composantes, de coordonnées (x_i, y_i) donc ce qu'on obtient c'est les coordonnées du point dont on connaît les niveaux de puissances reçus des différents Points d'Accès.

$$Li = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$$

- La position finale (x_f, y_f) est déterminée par la formule suivante:

$$\begin{pmatrix} x_f \\ y_f \end{pmatrix} = \frac{\sum_{i=1}^K \frac{1}{d(SSi, SSx) + d_0} * Li}{\sum_{i=1}^K \frac{1}{d(SSi, SSx) + d_0}}$$

- d_0 est une constante faible car quand on est dans un point de référence et donc la distance est nulle on va diviser par zéro. Pour cela on prend $d_0 = 0.001$ car on va considérer les niveaux de puissances en pourcentage par exemple 100% est considéré 1 et 30% considéré 0.3. Ainsi d_0 est faible devant l'ordre de grandeur des distances de telles sorte que ça n'influe pas les coordonnées qu'on va choisir pour le point recherché.

Pour la mesure effectuée dans le premier tableau et dont on ne connaît pas les coordonnées, on représente dans le deuxième tableau, après avoir fait un calcul de distance euclidienne, les K nœuds les plus proches.

Position	AP ₀	SS ₀	AP ₁	SS ₁	
inconnue	...	-52	...	-49	...

Position	AP ₀	SS ₀	AP ₁	SS ₁	
0	...	-50	...	-49	...
1	...	-52	...	-50	...
2	...	-55	...	-54	...
...	...	...	...	...	...
...	...	-80	...	-70	...

Figure 20:Exemple de l'algorithme du K plus proche voisin

Les lignes colorées en gris sont celles qui sont plus proche par rapport à la mesure effectuée dans le premier tableau qui représente le point dont veut localiser et dont on possède une information concernant le niveau de signal reçu des différents AP.

II.3.2.3. Choix de calcul centré mobile

Le calcul de la position des mobiles peut se faire selon deux approches différentes : soit par l'infrastructure ou par l'appareil mobile, dans notre cas nous avons choisi l'approche centré mobile.

Le modèle « **Client centré** » chaque client reçoit la puissance des signaux des points d'accès et il utilise ces informations pour en déduire sa position actuelle. Il permet d'augmenter la taille du système de localisation car, toutes les opérations sont réalisées sur les stations mobiles. Cependant, un système de localisation basée sur un tel modèle doit toujours faire face à la difficulté de la taille mémoire des stations mobiles. Comme la procédure de localisation est effectuée sur la machine cliente, on a besoins d'une taille mémoire relativement grande pour que le temps de localisation ne soit pas trop long. En effet, plus on stocke de données propres à la carte de couverture, plus la localisation est rapide.

II.3.2.4. Architecture générale du système de géolocalisation

- *Les mobiles*, qui sont des équipements dotés d'une carte Wi-Fi et d'un système Android.

- *Les points d'accès*, diffusent un signal qui permet une couverture du mobile dans une zone où il veut se positionner.
- *Base de données*, nécessaire à l'enregistrement de la cartographie des puissances réalisée dans une phase de calibrage.

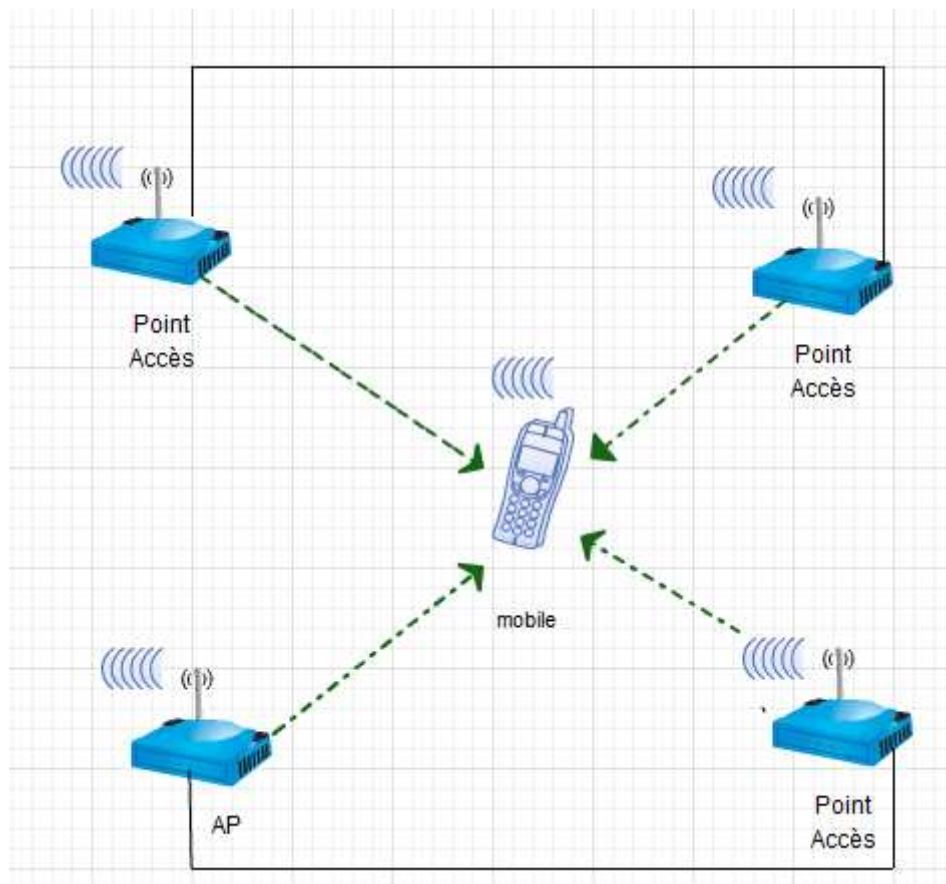


Figure 21: Architecture du système de positionnement en intérieur

II.3.2.5. Fonctionnement du système

Le traitement consiste en l'extraction de l'information essentielle au calcul de la position du mobile, représentée par le rapport de puissance des signaux captés.

Dans ce cas de figure, les éléments de l'infrastructure émettent des informations, que le mobile écoute et dont il se sert pour déduire sa position.

Ces informations peuvent tout simplement être les balises Wi-Fi (beacons) émises par de simples points d'accès (AP).

Le calcul de position se fait par application de la méthode *fingerprinting*, et ceci en interrogeant la base de données afin de consulter les informations cartographiques.

Conclusion

Ce chapitre a été consacré à l'identification et à l'extraction des informations pertinentes pour le positionnement des mobiles dans l'environnement indoor. En choisissant technologie, technique et algorithme qui sont nécessaires à la bonne compréhension du sujet afin de savoir la démarche à suivre et la planification des tâches pour la bonne conduite du projet.

Dans le prochain chapitre, nous passerons à la conception de l'application, à travers différents diagrammes, permettant une meilleure compréhension des fonctionnalités du système.

Introduction

Dans ce chapitre, sera question de présenter le processus de conception qui est utilisé pour mettre en place une solution logicielle. Il exploite conjointement, à travers des diagrammes, une représentation graphique qui vienne consolider la représentation textuelle. En effet, leur utilisation simultanée permet de construire une solide base de la spécification, l'analyse et la documentation de la conception.

Pour ce faire, il est important d'identifier les différents acteurs qui interagissent avec le système, ainsi que leurs besoins. En se basant sur les résultats de spécification, qui conduiront à une description analytique du système cible.

Pour la modélisation du système, nous avons opté pour le formalisme UML vu qu'il est utilisé dans de nombreux domaines d'activité, et qu'il offre une représentation claire.

III.1. Modélisation du système

La modélisation consiste à créer une représentation simplifiée d'un problème, un concept et le simuler. Elle comporte deux composantes :

- L'analyse, c'est-à-dire l'étude du problème,
- la conception, soit la mise au point d'une solution au problème

III.1.1. Présentation du langage UML

UML (*Unified Modeling Language*) est un langage de modélisation formel et normalisé, né de la fusion de plusieurs méthodes existantes **[Roq 02]**. Il permet de modéliser informatiquement un ensemble d'éléments cohérents du monde réel en un ensemble d'entités informatiques. Ces entités informatiques sont appelées objets. Ces objets sont décrits par des vues statiques et dynamiques, incluant un ensemble de

diagrammes, qui collaborent pour représenter diverses projections d'une même représentation d'un système d'objets.

UML possède 9 diagrammes de modélisation, repartis sur trois axes du niveau conceptuel :

- ✓ Statique
- ✓ Dynamique
- ✓ Fonctionnel

III.1.2. Modélisation avec le langage UML

UML permet de représenter des modèles [Uml8], mais il ne définit pas de processus d'élaboration de modèles. Les auteurs d'UML conseillent tout de même une démarche pour favoriser la réussite d'un projet, cette démarche met en œuvre les suivants :

- ✓ **Une démarche itératif et incrémentale** : pour modéliser un système complexe, il vaut mieux s'y prendre en affinant son analyse par étapes et favoriser le prototypage dans le but de mieux maîtriser la part d'inconnu et d'incertitudes qui caractérisent ces systèmes.
- ✓ **Une démarche guidée par les besoins des utilisateurs** : Tout est basé sur le besoin des utilisateurs du système, le but du développement lui-même est de répondre à leur besoin. Chaque étape sera affinée et validée en fonction des besoins des utilisateurs.
- ✓ **Une démarche centrée sur l'architecture logicielle** : Une architecture adaptée est la clé de voûte du succès d'un développement. Elle décrit des choix stratégiques qui déterminent en grande partie les qualités du logiciel (adaptabilité, performances, fiabilité...).

La démarche de modélisation choisie pour concevoir notre application peut être représentée graphiquement comme suit :

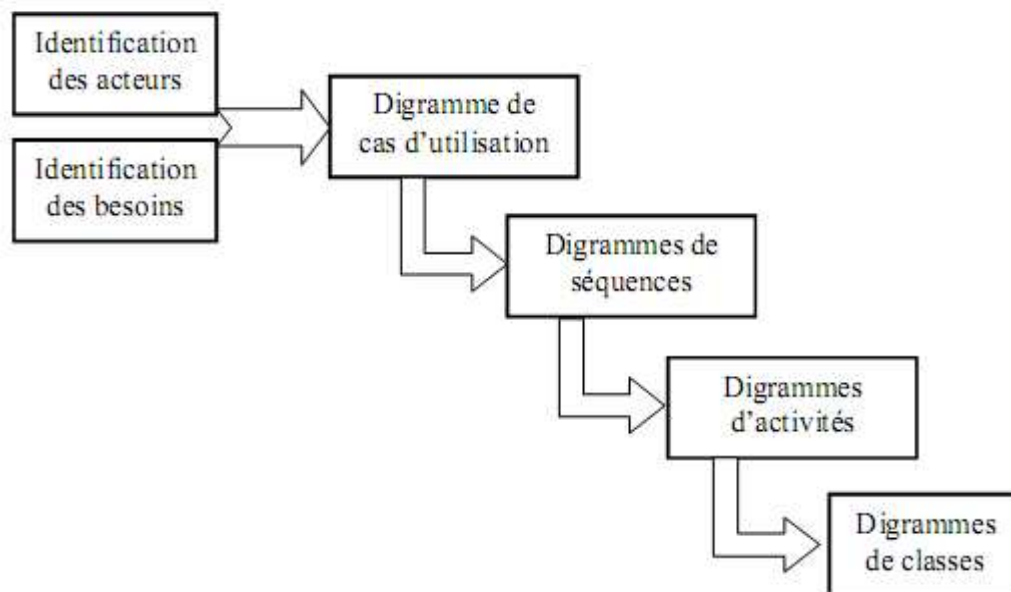


Figure 22:Démarche adoptée pour la modélisation

III.2. Analyse

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

III.2.1. Identification des besoins

L'objectif principal de l'application est de permettre à l'utilisateur de se repérer dans un endroit indoor.

En fait, quand les utilisateurs pénètrent dans un endroit inconnu, ils ont un sentiment de faiblesse, étant donné leur incertitude et la méconnaissance de leur position courante, ainsi que le manque de repérage.

Ce besoin d'être situé est satisfait grâce à notre application. Plus exactement, notre système offre l'opportunité aux gens de se positionner dans un local inexploré sans aucune contrainte.

Les fonctionnalités offertes par l'application sont les suivantes :

- ✓ se localiser,
- ✓ afficher la map.
- ✓ rechercher les points d'accès

III.2.2. Identification des acteurs

Définition: Un *acteur* représente un ensemble de rôles joué par des entités externes (utilisateur humain, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. [Ham05]

Dans le cas de notre application, nous avons un acteur qui interagit avec le système qui est :

- ✓ **User** : sont les utilisateurs détenteurs d'appareils mobiles couvèrent par les signaux diffusés par le réseau wifi (point d'accès), et ceci grâce à une application cliente installée sur leurs appareils.

III.2.3. Les cas d'utilisation

Définition : Un *cas d'utilisation* (**user case**) représente un ensemble de séquence d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. Un cas d'utilisation modélise un service rendu par le système.

III.2.4. Description du cas d'utilisation

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

Cas d'utilisation : **Se Localiser.**

Rôle : utilisateur

Description :

- 1- L'utilisateur lance l'application après avoir cliquer sur l'icone.
- 2- Le système affiche la page d'accueil.
- 3- L'utilisateur clique sur le bouton « map ».
- 4- Le système affiche l'interface « map »
- 5- L'utilisateur clique sur le bouton « se localiser ».
- 6- Le système construit et affiche la position « géo localisation ».

III.3. Conception

Le processus de conception de notre projet se caractérise par deux niveaux : le niveau applicatif et le niveau donnée.

- Le niveau applicatif s'appuie essentiellement sur quelques diagrammes de l'extension du langage de modélisation UML, à cet effet, nous avons adopté la démarche suivante:
 - ✓ Après l'identification des acteurs ainsi que les cas d'utilisation qui sont mis en œuvre par ces acteurs, le diagramme de cas d'utilisation est élaboré.
 - ✓ Chaque cas d'utilisation se traduit sous forme graphique à l'aide d'un diagramme de séquence et un diagramme d'activité.
 - ✓ Une identification des classes est fournie par la synthèse des diagrammes de séquence, ainsi le diagramme de classe sera élaboré.
- Le niveau donné concerne l'organisation conceptuelle, logique et physique des données manipulées. Durant, la partie analyse nous avons pu identifier les données nécessaires et indispensables au bon fonctionnement de l'application. A travers, la conception du niveau applicatif nous allons dégager les classes significatives, dès lors on peut élaborer la conception de la base de données.

III.3.1. Le niveau applicatif

3.1.1. Le diagramme des cas d'utilisation

Lors de la phase d'analyse nous avons pu identifier les acteurs ainsi que les cas d'utilisation associés à ces derniers. Ce qui nous donne l'opportunité d'élaborer le diagramme des cas d'utilisation.

Définition : Les diagrammes de cas d'utilisation permettent de représenter un ensemble de cas d'utilisation, d'acteurs et leurs relations. Ils présentent la vue statique des cas d'utilisation d'un système. Ils sont particulièrement importants dans l'organisation et la modélisation des comportements d'un système.

Suite à la présentation du système, nous pouvons dégager les besoins fonctionnels que nous présentons par ce qui suit et que nous exposons par des diagrammes des cas d'utilisation.

La représentation de ces diagrammes obéira aux conventions suivantes :

- ✓ Des flèches intitulées "**uses**" pour indiquer une relation entre un acteur externe et notre système.
- ✓ Une flèche continue dirigée d'un cas d'utilisation vers un autre intitulée "**includes**" signifiant que le premier cas d'utilisation utilise le deuxième pour atteindre un but nécessaire à son scénario.

Le client mobile :

Capte les signaux diffusés par les points d'accès (AP) wifi, qui lui permettent de se localiser sur une carte géographique représentant l'intérieur du bâtiment.

Les services offerts sont mis en valeur à travers le diagramme de cas d'utilisation illustré par la figure suivante :

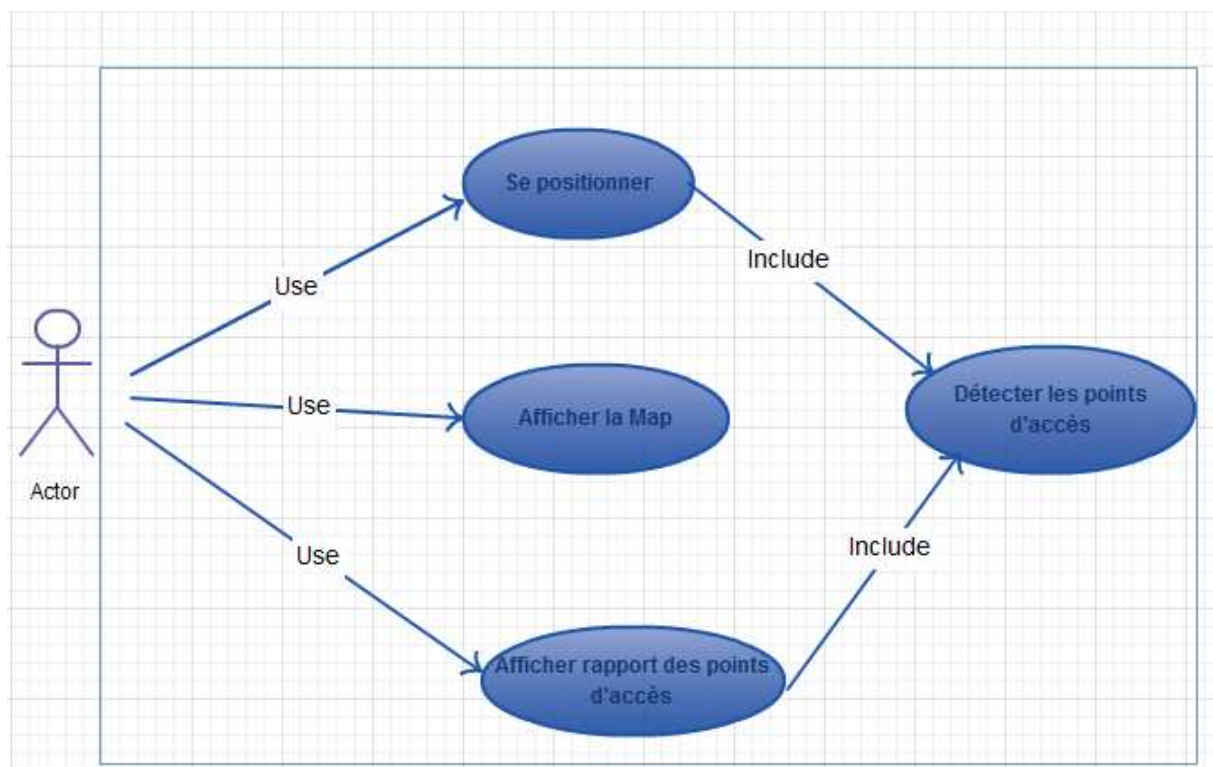


Figure 23:Diagramme de cas d'utilisation Client

3.1.2. Diagramme de séquence

Un *diagramme de séquence* permet de spécifier les interactions qui existent entre un groupe d'objet selon un point de vue temporel, on y met l'accent sur la chronologie des envois de message. Le diagramme de séquence est toujours de haut vers le bas, il illustre l'ordre dans lequel les messages sont envoyés entre les objets. Il peut servir à illustrer un cas d'utilisation.

Les composants d'un diagramme de séquence sont les suivants :

Les objets : ils apparaissent dans la partie supérieure, ce qui facilite l'identification des classes qui participent à l'interaction.

Les messages : ils sont représentés par des flèches directionnelles au-dessus de ces flèches figurent un texte nous informant du message envoyé entre les objets.

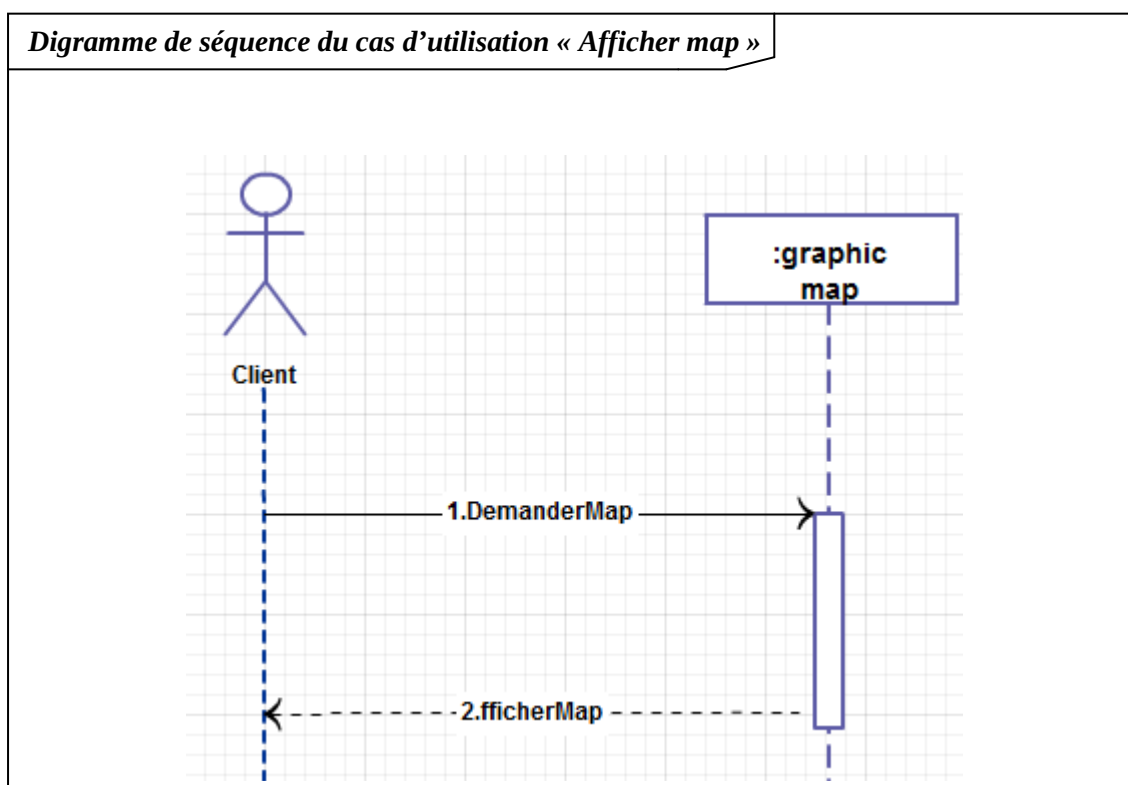


Figure 24:diagramme de séquences du cas d'utilisation « afficher map »

Diagramme de séquence du cas d'utilisation Client

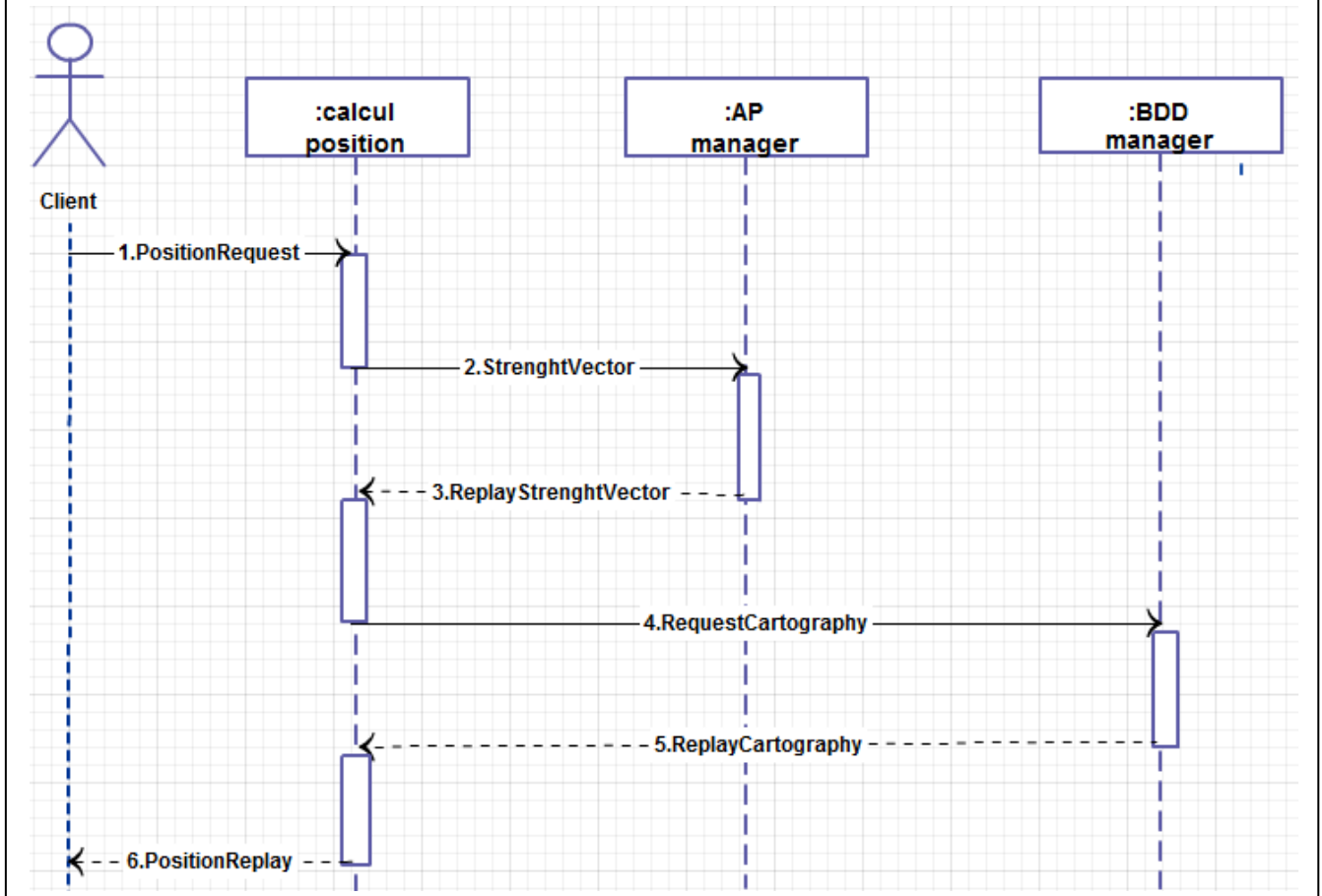


Figure 25:Diagramme de séquence du cas d'utilisation client

3.1.3. Les diagrammes d'activités

Un diagramme d'activités : apporte un point de vue complémentaire à l'aspect dynamique de la modélisation. Il offre un pouvoir d'expression très proche des langages de programmation objets. Il est donc bien adapté à la spécification détaillée des traitements en phase de réalisation. Un diagramme d'activités se concentre plutôt sur les activités entre les objets, c'est-à-dire, il met en évidence l'activité qui a lieu dans le temps, donc les opérations transmises entre les objets.

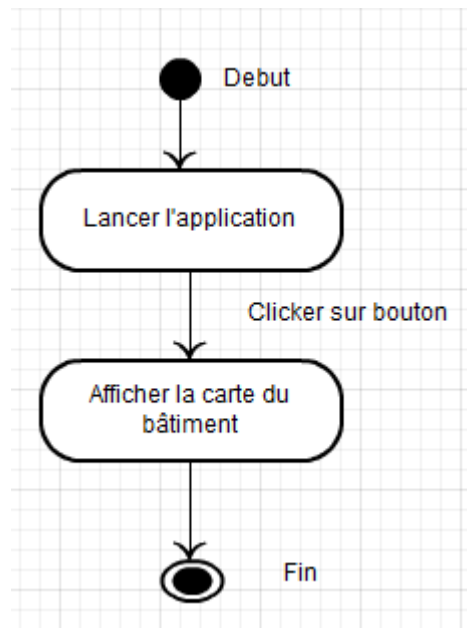


Figure 26:Diagramme d'activité de cas d'utilisation « afficher map ».

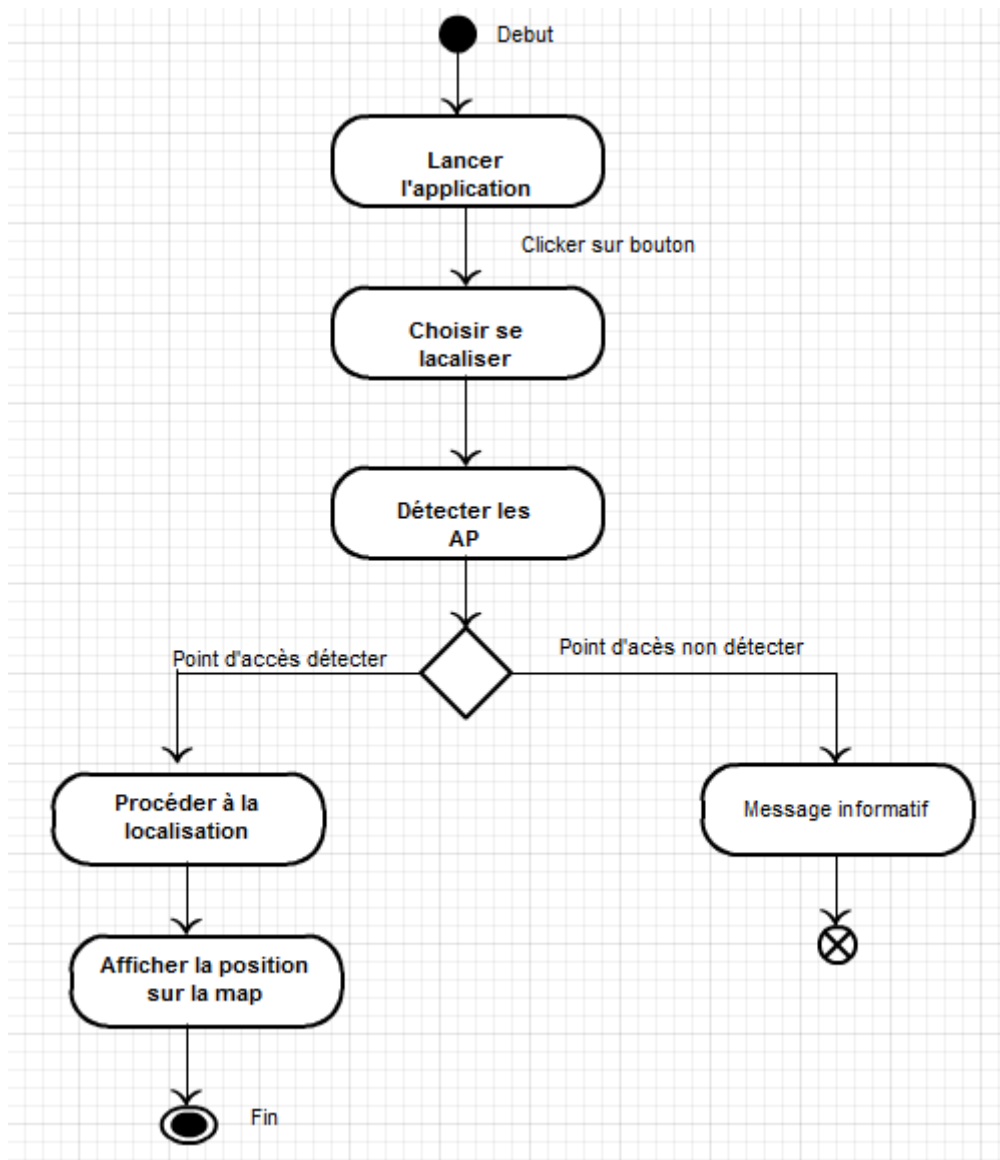


Figure 27:Diagramme d'activité de cas d'utilisation « Se localiser ».

3.1.4. Les diagrammes de classe

Dans cette partie, nous allons passer à la modélisation de l'aspect statique de notre application, c'est-à-dire nous allons modéliser l'intérieur de notre système. Pour ce faire, nous allons utiliser les diagrammes de classes.

Le diagramme de classe est le modèle le plus utile. Il offre un formalisme pour représenter les concepts usuels de l'orienté objet. Le modèle des états et le modèle d'interaction permettent de représenter la dynamique des objets. Alors que les diagrammes précédents (diagrammes de séquences et diagrammes d'activités)

montrent le système du point de vue dynamique, le diagramme de classe en montre la structure interne. Il contient principalement des classes reliées par des associations et chaque classe contient des attributs et des opérations.

Après avoir élaboré les diagrammes de séquences et les diagrammes d'activités pour les cas d'utilisation déjà décrit, nous allons élaborer des diagrammes de classes détaillés pour chacun d'eux.

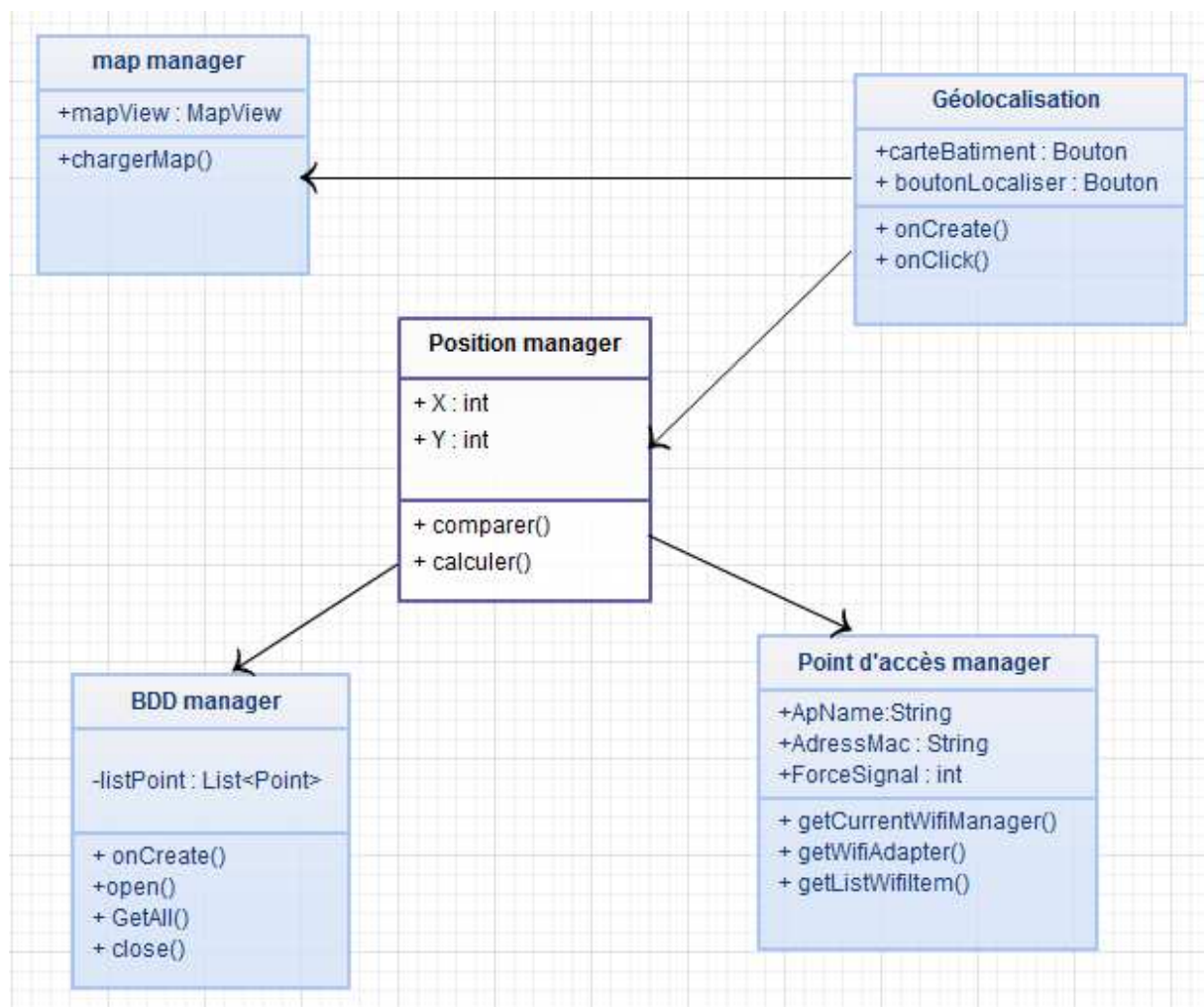


Figure 28:Diagramme de classe « se localiser »

III.3.2.Le niveau Données

3.2.1. Schéma de la base de données

Notre application de localisation nécessite la mise en œuvre d'une base de données, contenant un ensemble de points d'accès avec certaines informations. Ces informations sont nécessaires au bon fonctionnement du système.

Le schéma conceptuel de la base de données est représenté par la figure suivante :

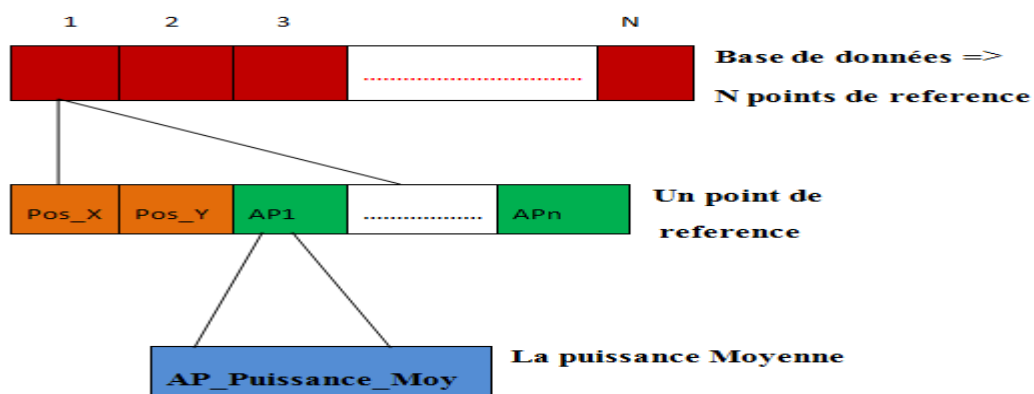


Figure 29:structure de la base de données

L'application dans tout ces cas d'utilisation manipule les données stockées dans la Base De Données, et elle contient la table suivante :

Table Point Référence:

Champs	Signification	Type	Clé
ID	l'identifiant unique pour chaque point d'accès auto-incrémenté (ici il s'agit d'un String mais nous avons fait en sorte que l'auto-incrémentation se fasse quand même).	VARCHAR	Primaire (Pk)

Référence	C'est la position sur un plan à 2 Dimension (X,Y)	VARCHAR
AP1	Représente la moyenne-puissance captée en Décibel	VARCHAR
AP2	Représente la moyenne-puissance captée en Décibel	VARCHAR
AP3	Représente la moyenne-puissance captée en Décibel	VARCHAR
Salle	La pièce où se situe le point d'accès	VARCHAR

Tableau III.1 : Table point de référence

Conclusion

La phase de conception sert à identifier les différents objets, qui nous aident à certifier les fonctionnalités souhaitées. Tout en donnant, une description complète de ces diagrammes, ce qui permet une meilleure compréhension du système.

Dans le chapitre qui suit nous allons nous intéresser à la réalisation de ce qui a été conçu, pour enfin mettre sur pied le système final.

Introduction

L'implémentation constitue la dernière étape dans le processus de réalisation de notre système de localisation. Durant cette étape, nous avons procédé au codage, au rassemblement et intégration des différents modules, avant de conclure avec une série de test.

Nous exposons dans ce qui suit, l'environnement matériel et logiciel ainsi que la démarche suivie au cours du développement des différents modules logiciels.

IV.1. Description de la plateforme matérielle

Notre environnement matériel se caractérise par l'utilisation de deux PC de marque HP Compaq CQ71 et Toshiba dont les caractéristiques se rapprochent :

- Un microprocesseur : Pentium® Dual Core CPU T4200 @2.00Ghz 2.00Ghz.
- Une mémoire vive de 4Go.
- Un disque dur de 300Go.

Un Smartphone équipés du système android de marque Sony Ericson

IV.2. Environnement de développement logiciel

Afin de réaliser notre système de géo localisation en intérieur décrit tout au long de ce mémoire, plusieurs outils sont nécessaires :

VI.2.1. Les outils de développements

Les applications android sont développées en **Java** : une technologie présentée pour la première fois en 1995 par Sun Microsystems (rachetée ensuite par Oracle).

2.1.1 Le langage de programmation Java

Java **[Bou 08]** est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C++.

- Approche objets

La programmation orienté objet est définie comme un paradigme de programmation informatique basé sur l'interaction de briques appelées objets, qui sont eux-mêmes des abstractions d'objets réels, via leurs relations et les propriétés qui leur sont accordées.

Ce paradigme assure:

- une modularité des programmes résolvant le problème de la complexité des codes.
- une vitesse d'exécution.
- intégration des traitements aux données
- réduire distorsions entre système informatique et monde réel
- encapsulation
- une souplesse de réutilisation et d'évolution du code.

Parmi les caractéristiques Java qui ont contribué au succès de ce langage, on retrouve la propriété de portabilité : le code des applications développées dans ce langage est indépendant de la machine sur laquelle il s'exécute. Le code passe par une phase de compilation pour générer du « byte code », pour pouvoir être exécuté par la JVM qui le transformera en code natif pour la machine.

Il est donc possible d'exécuter des programmes Java sur tous les environnements qui possèdent une Java Virtual Machine, il s'agit du principe « Compile Once, run everywhere ».

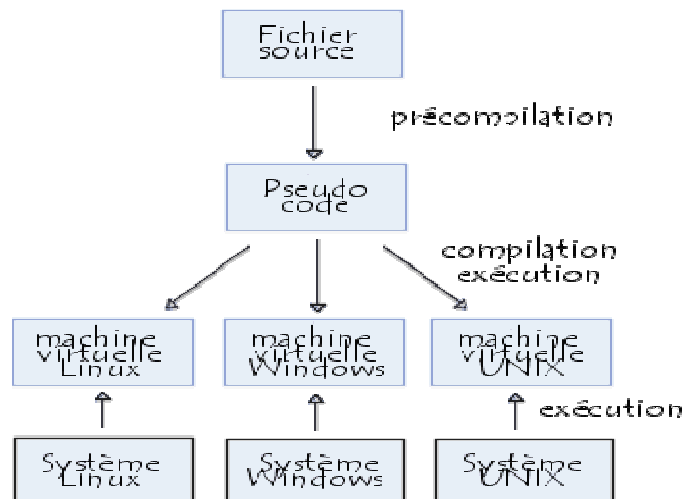


Figure 30:Le concept de machine virtuelle java

2.1.2 Le SDK android (Standard Development Kit)

Google a créé sa propre machine virtuelle Java pour appareils mobiles android à ressources limitées en termes de mémoire et de processeur : Dalvik de Dan Bornstein.

Puisque l'environnement n'est pas une JVM classique, nous ne pouvons pas utilisés directement les outils Java habituels pour le développement. Pour ce faire Google offre un kit de développement (sdk) téléchargeable gratuitement.

Il s'agit de la couche « Application Framework » présentée dans la figure 23(Architecture Android). Ce SDK contient une partie du noyau Java enrichit de fonctionnalités android : on y retrouve l'ensemble des interfaces de développement (Application Programming Interface – API) que Google met à disposition.

- L'émulateur android :

Pour tester les applications développées on est pas obligé d'être en possession d'un smartphone android, puisque Google a développé son propre émulateur.

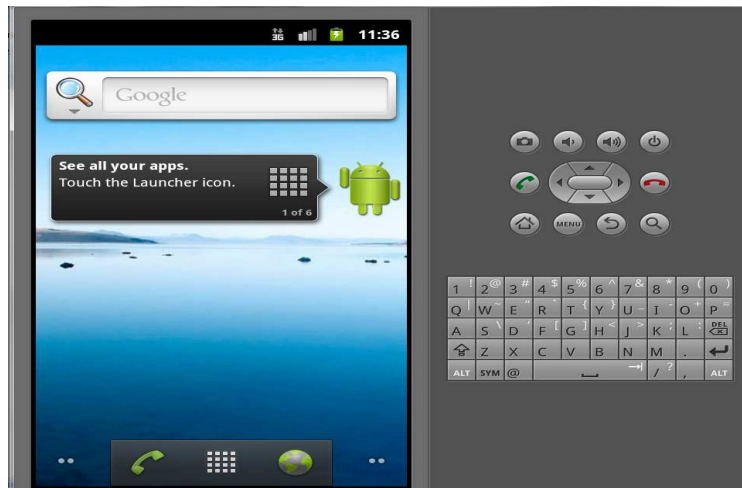


Figure 31:L'émulateur android

Cet émulateur reproduit toutes les caractéristiques matérielles et logicielles d'un smartphone : on peut lui créer un espace de stockage, simuler des appels ou messages, simuler le GPS...etc On peut alors estimer que notre application se comportera de la même manière sur l'émulateur que sur un smartphone.

Le SDK est fournit avec encore d'autres outils afin de favoriser la productivité on peut citer :

- ✓ **apkbuilder** : permet de créer des « livrables », il s'agit d'un paquet prêt à être installé sur un appareil
- ✓ **mksdcard** permet de créer une « fausse » carte SD sur l'émulateur.
- ✓ **Android Debug Bridge (adb)** : un outil permettant de gérer l'émulateur depuis un terminal, il permet aussi de travailler sur un vrai téléphone branché à l'ordinateur à l'aide d'un port USB.

Afin de coder et tester les applications, Google a aussi prit conscience que le développement demande un environnement permettant d'accroître la productivité en automatisant les phases simples et ainsi permettre de se concentrer sur la valeur ajoutée d'une application. C'est pour cette raison que l'IDE Eclipse a été retenu.

2.1.3 L'IDE Eclipse

Eclipse est un est un environnement de développement, développé par IBM en 2004, initialement destiné à Java, mais par sa gestion de plugins il est

aujourd'hui possible d'utiliser Eclipse avec presque toutes les technologies. Lui-même écrit en Java, il est disponible pour toutes les plateformes. Eclipse est libre (sous licence Eclipse Public Licence) et donc gratuit.

Pour le développement de notre application, on a choisi Eclipse INDIGO (Son interface principale est donnée dans la figure IV.3).

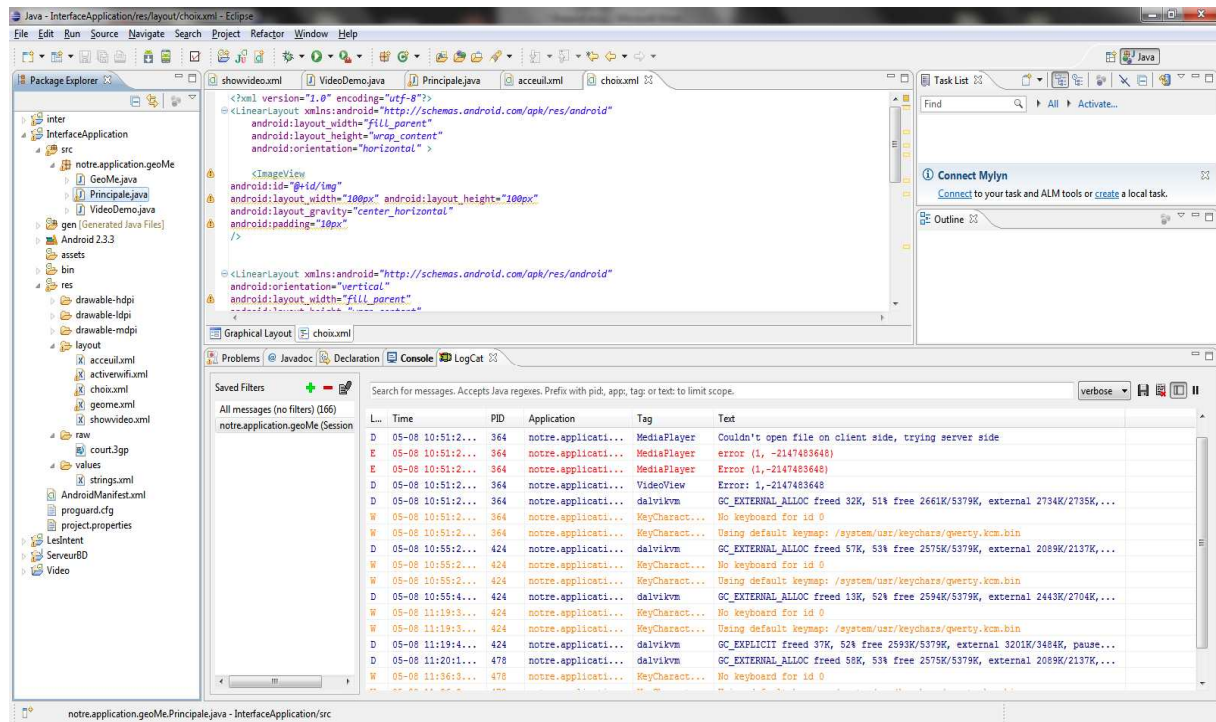


Figure 32:L'environnement de développement eclipse

2.1.4 Le plugin ADT (Android Development Tool) pour eclipse :

Google a développé son propre plugin : Android Development Tool (ADT) adapté à eclipse mettant à la disposition des développeurs un ensemble d'outils et de fonctionnalités Android. Il est disponible en téléchargement libre à l'adresse suivante <<http://developer.android.com/sdk/eclipse-adt.html>. />

Ce plugin étend les capacités d'Eclipse et permet entre autre la création facile d'interfaces graphiques, le débogage avec la vue Dalvik Debug Monitor System (DDMS) ou encore l'export d'un APK (paquet prêt à être installé).

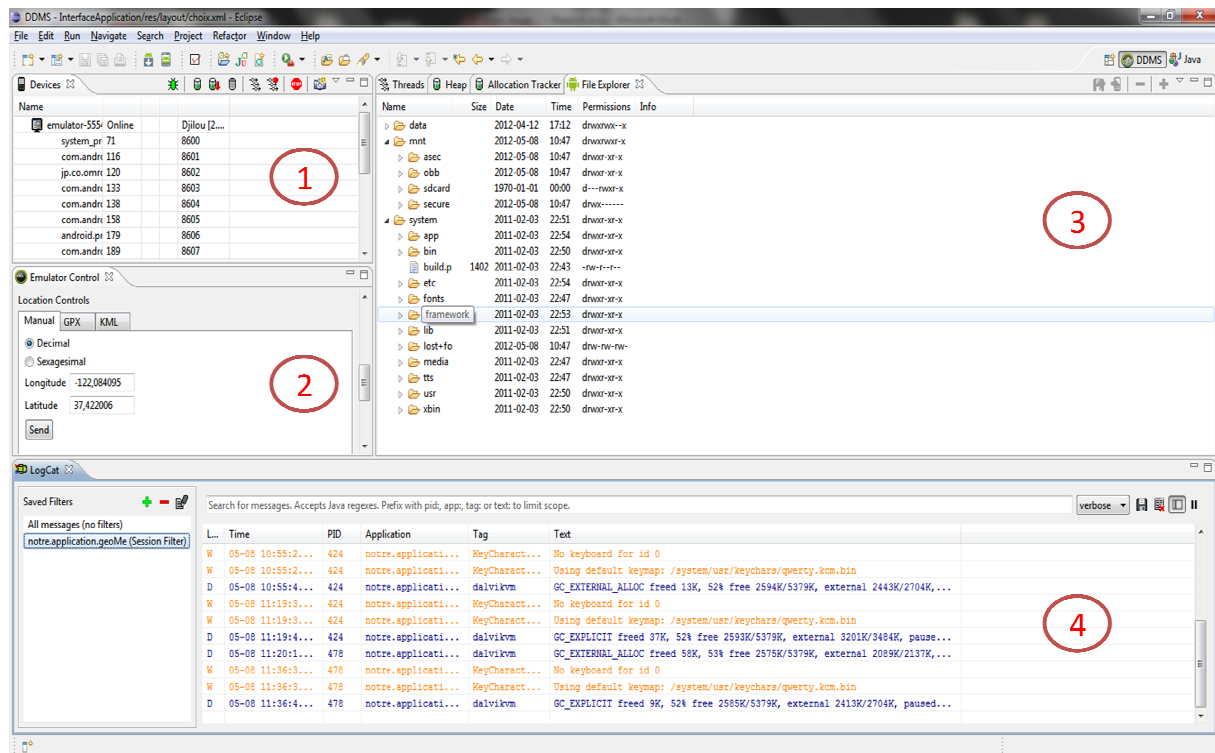


Figure 33:La console DDMS sous eclipse

Le plugin permet une gestion complète de l'émulateur :

- Dans la zone 1 de la figure on retrouve l'état du système avec les processus en cours d'exécution.
- L'onglet Emulator control de la zone 2 permet de simuler des appels ou message entrants, la position du téléphone en saisissant des coordonnées GPS ou en téléchargeant des fichiers KML (Keyhole Markup Language : un langage basé sur le formalisme XML et destiné à la gestion de l'affichage de données géo-spatiales comme Google Earth et GoogleMaps).
- Dans la zone 3 on aperçoit les différents onglets permettant de gérer : les allocations dynamiques, les threads en cours d'exécution, un explorateur depuis lequel nous pouvons ajouter ou retirer des fichiers. Et enfin la zone 4 qui représente le log du téléphone, cette vue permet de suivre le comportement de l'application.

❖ Les composants d'une application Android

Une application Android est composée d'éléments de base :

✓ Activities (Activités en Français)

Une activité est la composante principale pour une application Android. Elle représente l'implémentation et les interactions de vos interfaces.

✓ Services

Un service, à la différence d'une activité, ne possède pas de vue mais permet l'exécution d'un algorithme sur un temps indéfini. Il ne s'arrêtera que lorsque la tâche est finie ou que son exécution est arrêtée.

Il peut être lancé à différents moments :

- Au démarrage du téléphone.
- Au moment d'un événement (arrivée d'un appel, SMS, mail, etc...).
- Lancement de votre application.
- Action particulière dans votre application.

✓ Broadcast and Intent Receivers

Un Broadcast Receiver comme son nom l'indique permet d'écouter ce qui se passe sur le système ou sur votre application et déclencher une action que vous aurez prédéfinie. C'est souvent par ce mécanisme que les services sont lancés.

✓ Content providers

Les "content providers" servent à accéder à des données depuis votre application. Vous pouvez accéder :

- Aux contacts stockés dans le téléphone.
- A l'agenda.

- Aux photos.
- Ainsi que d'autres données depuis votre application grâce aux content providers.

❖ Cycle de vie d'une application Android

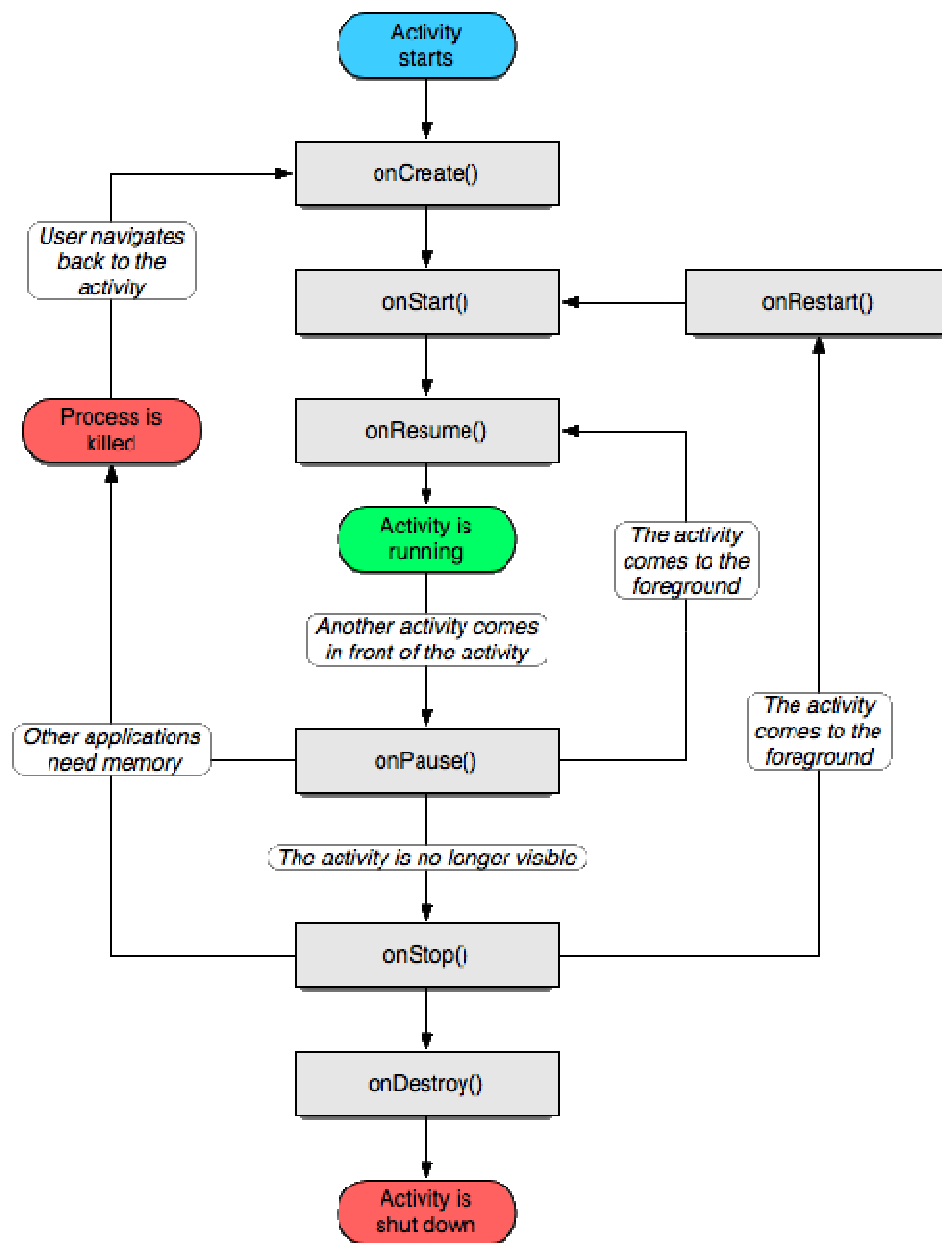


Figure 34: Cycle de vie d'une application android

❖ Méthodes correspondant au cycle de vie :

➤ onCreate

Cette méthode est appelée à la création de votre activité (Activity). Elle sert à initialiser votre activité ainsi que toutes les données nécessaires à cette dernière.

Quand la méthode **OnCreate** est appelée, on lui passe un Bundle en argument. Ce Bundle contient l'état de sauvegarde enregistré lors de la dernière exécution de votre activité.

➤ onStart

Cette méthode signifie que l'activité va devenir visible (début du passage au premier plan).

Si votre activité ne peut pas aller en avant plan quelque soit la raison, l'activité sera transférée à **OnStop**.

➤ onResume

Cette méthode est appelée après **OnStart** (au moment où votre application repasse en foreground). à la fin de l'appel à la méthode **onResume** votre application se trouve au premier plan et reçoit les interactions utilisateurs.

➤ onPause

Si une autre activité passe au premier plan, la méthode onPause est appelée sur votre activité.

Afin que vous sauvegardiez l'état de l'activité et les différents traitements effectués par l'utilisateur.

A ce stade, votre activité n'a plus accès à l'écran, vous devez arrêter de faire toute action en rapport avec l'interaction utilisateur (désabonner les listeners). Vous pouvez

par contre continuer à exécuter des algorithmes nécessaires mais qui ne consomment pas trop de CPU.

➤ onStop

Appelée quand votre activité n'est plus du tout visible quelque soit la raison. Dans cette méthode vous devez arrêter tous les traitements et services exécutés par votre application.

➤ onDestroy

Appelée quand votre application est totalement fermée (Processus terminé). Toutes les données non sauvegardées sont perdues

Android permet de séparer le code métier de l'application du code des interfaces graphiques grâce au formalisme XML :

VI. Le formalisme XML :

EXtensible **M**arkup **L**anguage est un langage informatique de balisage générique. Il sert essentiellement à stocker/transférer des données de type texte Unicode structurées en champs arborescents.

Sous Android, et grâce à ce langage, les interfaces sont décrites dans ce format spécial, et Android les convertit automatiquement en objets Java qui seront par la suite disponibles comme tout autre objet du code de l'application. Il offre ainsi plus de souplesse de développement, facilite les modifications du code et assure la séparation entre la présentation et le comportement des objets.

IV.3. Modules logiciels de l'application

Afin de simplifier le développement, nous avons décidé de scinder le système en sous modules logiciels remplissant chacun des fonctionnalités bien spécifiques comme suit :

1. Création de la base de données
2. Module de récupération de signaux Wifi ainsi que leurs caractéristiques
3. Module de recherche de la position géographique
4. Interface Client Android

La modularité d'une telle architecture facilite la maintenance à long terme, l'extensibilité du système et aussi la réutilisation des différentes parties dans d'autres systèmes.

Notre Application a été développée pour la version 2.3.3 d'API level 10 d'Android baptisé Eclair. Toutefois, elle est utilisable sur la version 4.0.3 d'API level 15 et sur les versions antérieures d'Android.

IV.3.1 Construction de la base de données des mesures (cartographie des puissances):

La première phase de développement consiste en la création de la cartographie des puissances, et qui constituera un support pour les autres modules.

La base de données est construite à partir de mesures réalisées en environnement réel. Plusieurs points de mesure ont été réalisés au niveau du département Informatique : *Laboratoire de Recherche*, et sont représentés sur la figure suivante par des points bleue. Les points d'accès présents lors des mesures sont localisés à l'aide des points rouges.

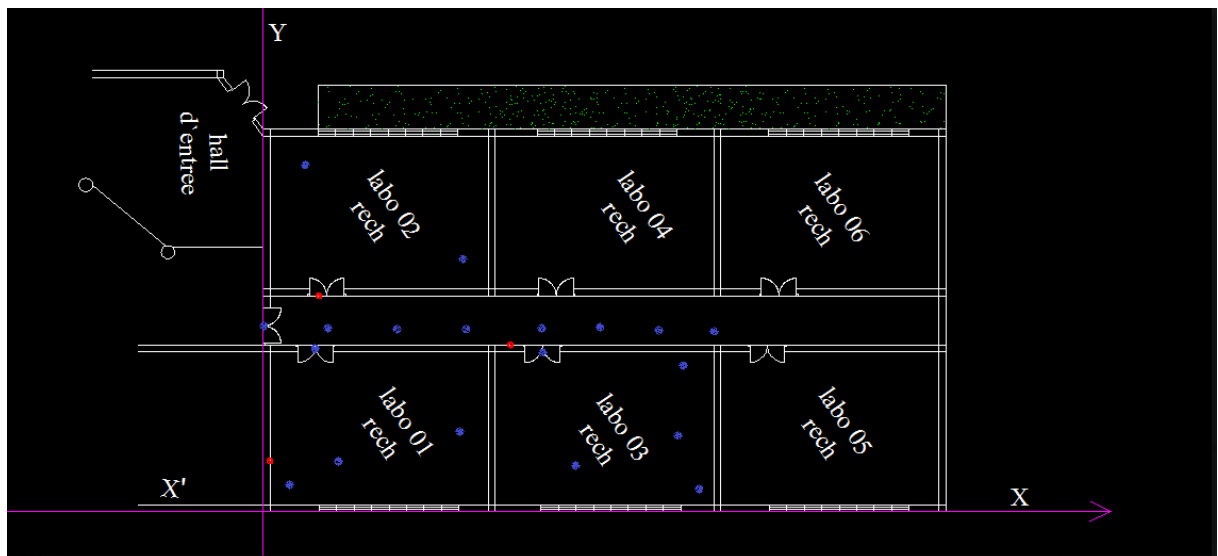


Figure 35:Points de mesure réalisés au laboratoire

La prise de mesure a été effectuée à l'aide du mobile Sony Ericson, doté d'un système d'exploitation Android à différents points de la zone à couvrir.

Pour chaque point de mesure, on a enregistré des mesures de puissance à des intervalles de temps réguliers : pendant toute la durée de la mesure, on réalise des déplacements lents dans toutes les directions sur une zone de 1m² autour du point de test tout en modifiant l'orientation du récepteur. Ceci permet d'obtenir un ensemble représentatif du canal sur une zone d'1 m² environ. A chaque point de mesure, plusieurs échantillons sont enregistrés. La valeur moyenne des échantillons est stockée dans la base de données.

Il est important de noter que l'on ne possède de mesure que pour les points d'accès qui couvrent le point de mesure, c'est-à-dire les points d'accès dont le niveau de puissance est supérieur au seuil de détection de la carte réceptrice.

Nous avons choisit de prendre deux points de mesure par bureau ce qui est largement suffisant pour déterminer avec précision la position du mobile.

Les puissances de signaux relevés aux différents endroits du bâtiment (les points de références) pendant la phase de calibration sont enregistrées au niveau de la base de données cartographique.

Un enregistrement de cette base représente un point de référence. Ce dernier est caractérisé par ses coordonnées géographiques dans l'espace à deux dimensions ainsi que les puissances captées par chaque point d'accès de la zone de couverte.

Pour l'implémentation de notre base de données, notre choix s'est porté sur le SQLite.

❖ Présentation :

SQLite est un système de gestion de bases de données relationnelles (SGBDR) bien connu, il est :

- ✓ open-source ;
- ✓ conforme aux standards ;
- ✓ léger ;
- ✓ mono tiers.

Il a été implémenté sous la forme d'une bibliothèque C compacte incluse dans Android.

Étant implémenté sous forme de bibliothèque et non exécuté dans un processus distinct, chaque base de données SQLite fait partie intégrante de l'application qui l'a créée. Cela réduit les dépendances externes, minimise la latence et simplifie le verrouillage des transactions et la synchronisation.

SQLite a une réputation de grande fiabilité et il est le SGBDR choisi par de nombreux appareils électroniques, notamment beaucoup de lecteurs MP3 et de smartphones. Léger et puissant, il diffère des moteurs de bases de données conventionnels par son typage faible des colonnes, ce qui signifie que les valeurs d'une colonne ne doivent pas forcément être d'un seul type. Chaque valeur est typée individuellement par ligne.

La conséquence en est que la vérification de type n'est pas obligatoire lors de l'affectation ou de l'extraction des valeurs des colonnes d'une ligne.

Toutes ces caractéristiques font de SQLite un bon candidat pour notre projet. **[RM12]**

VII.2 Module Récupération des signaux wifi :

Le deuxième module développé pour le smartphone android englobe la fonctionnalité de lecture des informations détectées par la carte wifi du terminal mobile. Il transmet les informations nécessaires à la localisation telles que : l'identifiant (ESSID) ou l'adresse MAC, le niveau du signal (RSSI), la fréquence et le canal de transmission au module Recherche de position.

Ce module est déclenché comme processus de fond dès que l'application est installée sur le mobile.

VII.3 Module Calcul de position :

Ce module est développé en java, il recherche la localisation du mobile à partir des informations reçus par le module Récupération des signaux et en consultant le module de base de données. La position la plus vraisemblable du mobile est déterminée en appliquant l'algorithme des k plus proches voisins.

VII.4 Module Interface Android :

C'est le module développé pour le smartphone Android , il représente l'interface de l'application, il invite l'utilisateur à choisir parmi les différentes options du menu :



Figure 36:Interface Client Android

- ❖ Carte du Batiment : Consulter le plan du Batiment qui est une vue où l'utilisateur peut avoir la totalité de l'architecture.

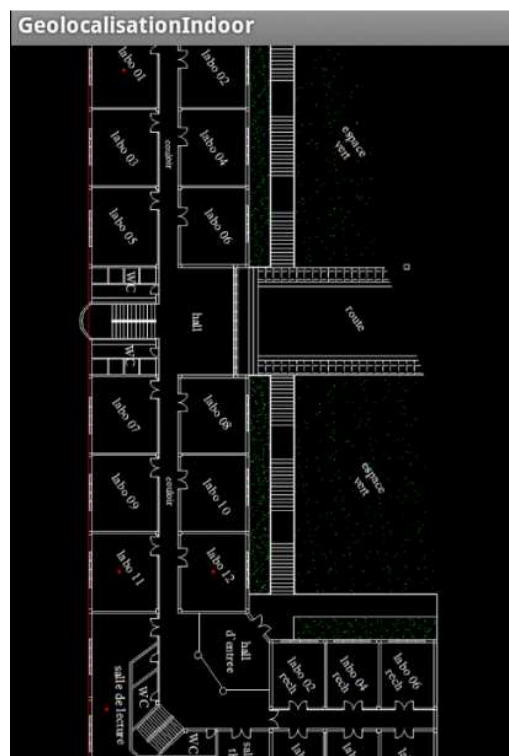


Figure 37:Plan du département Informatique (Etage 1)

❖ Rechercher Ap : permet d'afficher les points d'accès détectés ainsi que leurs caractéristiques (NomAp, Force du signal, AdresseMac)

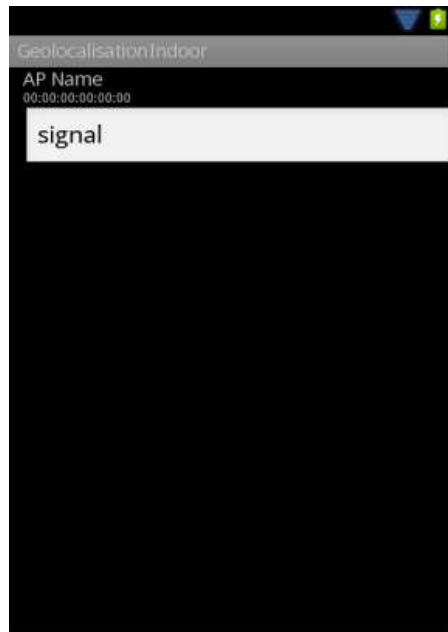


Figure 38: Résultats du scan

❖ Se Localiser : Permet de se localiser sur la carte pour retrouver la position à la demande de l'utilisateur

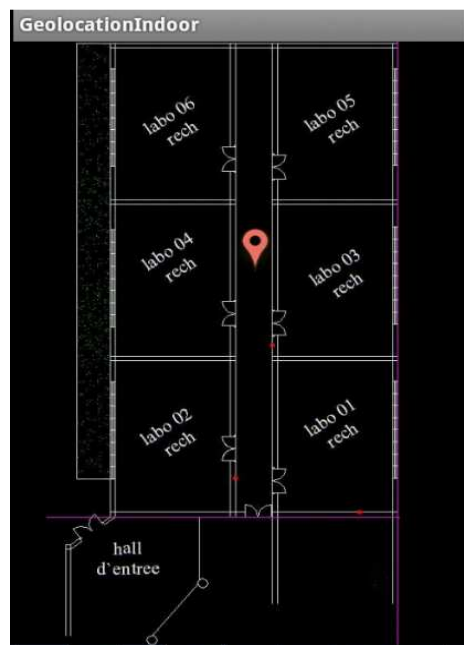


Figure 39: Géo localisation à l'intérieur du laboratoire

VIII. Test du système :

Des tests unitaires ont été réalisés pour valider chaque module développé .Le tout a été intégré pour construire le système de positionnement global.

Après avoir charger l'application Android sur le Smartphone les tests peuvent commencer.

Une série de mesures a été menée au sein du bâtiment, dans ce qui suit un tableau récapitulatif des positions réelles et celles générées par le système de localisation implémentant la méthode fingerprinting avec l'algorithme des K plus proches voisins :

Position réelle	Position générée par le système	Nombre d'AP détectés	Erreur
(2, 5)	(1.00, 6.72)	3	3,45
(5,6)	(0.59, 5.39)	2	5,62
(8,0)	(0.49, 5.38)	1	10,05
(8,7)	(2.91, 5.62)	2	5,78
(1,4)	(1.03, 5.54)	3	5,75
Moyenne :			6.13

Tableau : Comparaison entre positions réelles et positions générées par le système

Nous remarquons que dans 80% des cas, les positions générées par le système et les positions réelles se trouvent dans la même pièce.

Ceci est influencé en partie par les niveaux de puissances reçues, en effet plus ces derniers sont faibles et plus l'écart se fait ressentir.

L'écart entre les deux positions : réelle et générée par le système est estimé à 6,13mètres, ce qui constitue une précision acceptable dans un environnement intérieur.

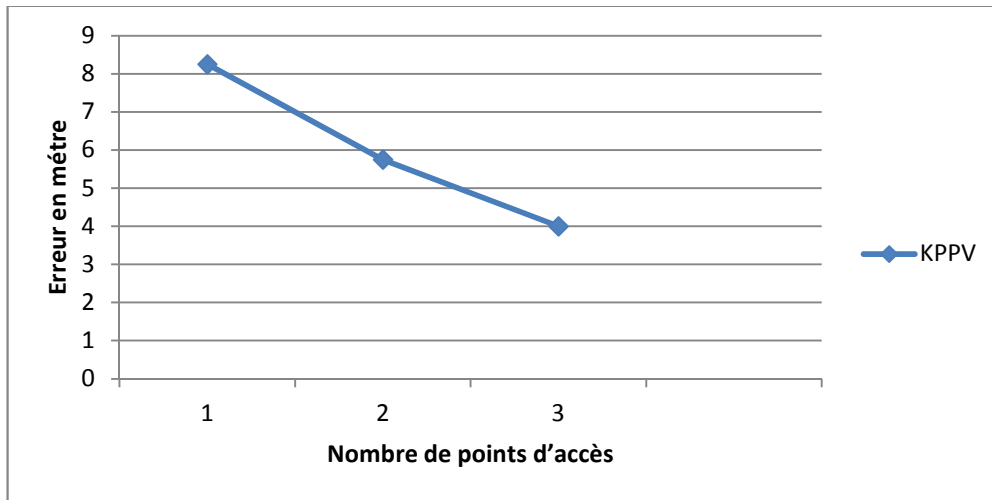


Figure 40:Relation entre performance et nombre de points d'accès

Comme on peut le constater dans le graphe ci-dessus, quand le nombre de points d'accès augmente le taux d'erreur diminue ce qui montre que la performance de la localisation est étroitement liée au nombre de points d'accès.

Conclusion

Dans ce chapitre, nous avons décrit les résultats obtenus suite à l'étape de codage. Nous avons testé certaines fonctionnalités du système et l'aboutissement est jugé appréciable et encourageant, ce qui nous mène à continuer nos travaux et nos efforts dans ce domaine.

Ces résultats pourront être utiles pour des évolutions ou encore des optimisations.

Conclusion générale

Les systèmes de géopositionnement ont un avenir fleurissant, et de nombreuses technologies seront sûrement très utilisées, et sont déjà de plus en plus développées par les entreprises et centres de recherches. Les exemples présents dans la littérature se sont multipliés. Ce qui prouve l'intérêt que l'on porte à ces systèmes. On peut prévoir que dans quelques années, les systèmes de géopositionnement en extérieur par satellite ne seront plus les seuls démocratisés et de nombreuses techniques de géopositionnement, notamment en intérieur, seront utilisées par le grand public. En effet, l'amélioration du positionnement et de la mobilité ouvre de nombreux horizons, pour le multimédia et pour le confort du grand public, des entreprises ou des lieux publics, etc.

Ce modeste travail s'inscrit dans le contexte de la géo-localisation des utilisateurs mobiles dans un environnement indoor. L'utilisation des réseaux Wifi, conçus initialement pour répondre à des besoins de communication, s'est rapidement élargie pour intégrer des besoins de localisation. Pour ce faire, nous avons étudiés, la localisation en extérieur ainsi qu'en intérieur, en présentant leurs principes d'une façon générale, ensuite, une description des techniques et méthodes utilisées dans ce domaine. Après une exposition de la méthode de localisation basée sur l'approche adoptée, dans le cadre de notre travail, ce projet décrit les différentes étapes menées pour aboutir à ce système, en le vérifiant avec des tests qui répondent à la problématique posée dans ce mémoire. Les résultats des tests conduits ont été assez satisfaisants et encourageant. Néanmoins, le développement d'un tel système a nécessité beaucoup d'efforts, du fait que le domaine de la géolocalisation et notamment en intérieur est un domaine récent et donc les travaux réalisés par la communauté scientifique restent expérimentaux.

Bien que nos objectifs initiaux, ceux de découvrir le concept de géolocalisation, les systèmes et les méthodes existantes et arriver à terme dans la réalisation d'un système de localisation en intérieur dans le but soit atteint, il y'a toujours place aux améliorations, qui permet, à nos successeurs intéressés dans ce domaine, d'investir

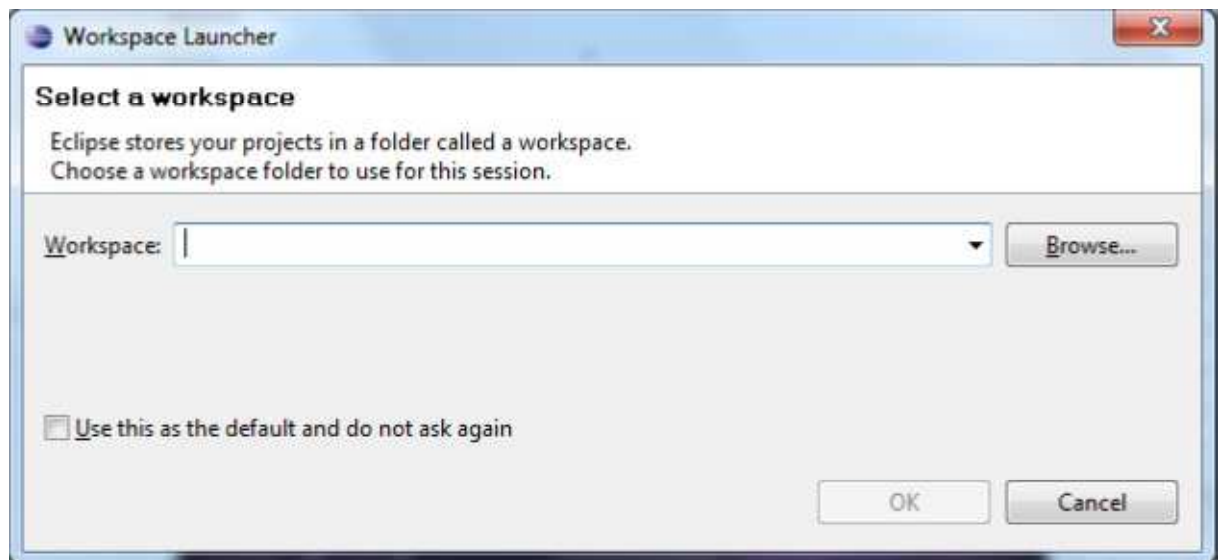
leur temps et leurs connaissances, afin de contribuer à un progrès dans d'autres voies de la localisation en intérieur, en essayant d'atteindre la précision de la localisation en extérieur qu'on a de nos jours et ceci en ajoutant des points de références à la cartographie des puissances. Ainsi que, dessiner le trajet que l'utilisateur pourra suivre pour atteindre sa destination.

On estime que l'occasion de déployer nos connaissances et nos capacités pour la réalisation de ce projet furent une expérience très enrichissante sur le plan personnel, et surtout sur le plan professionnel. On a eu l'honneur de contribuer à un projet très intéressant et novateur, qui mène un bon pas vers le monde professionnel. Au niveau plus personnel, l'expérience de travailler en équipe, nous a permis d'enrichir notre personnalité et d'acquérir encore plus d'indépendance.

Annexe

Créer un projet Android avec Eclipse

1. Lancer Eclipse
2. Choisir un répertoire de travail



3. Indiquer à Eclipse où se trouve le SDK Android : Windows -> Preferences puis dans la rubrique Android désigner l'emplacement du SDK par le bouton "Browse", terminer par "Apply" puis "OK"

4. Quand 'Eclipse a démarré aller dans le menu : File -> New ->Android project

New Android Project

Project name must be specified

Project name:

Contents

- ☒ Create new project in workspace
- ☐ Create project from existing source
- ☒ Use default location

Location:

- ☐ Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API Level
☐ Android 2.1-update1	Android Open Source Project	2.1-update1	7
☐ Google APIs	Google Inc.	2.1-update1	7

Properties

Application name:

Package name:

☒ Create Activity:

Min SDK Version:

Working sets

☐ Add project to working sets

Working sets:

5. Cliquer sur Finish, le projet est créé :
On y trouve les répertoires :

- *src* pour les sources : ce répertoire contient un paquetage dont le nom est celui qui a été choisi au début dans lequel se trouve la classe pour l'activité dont le contenu est :

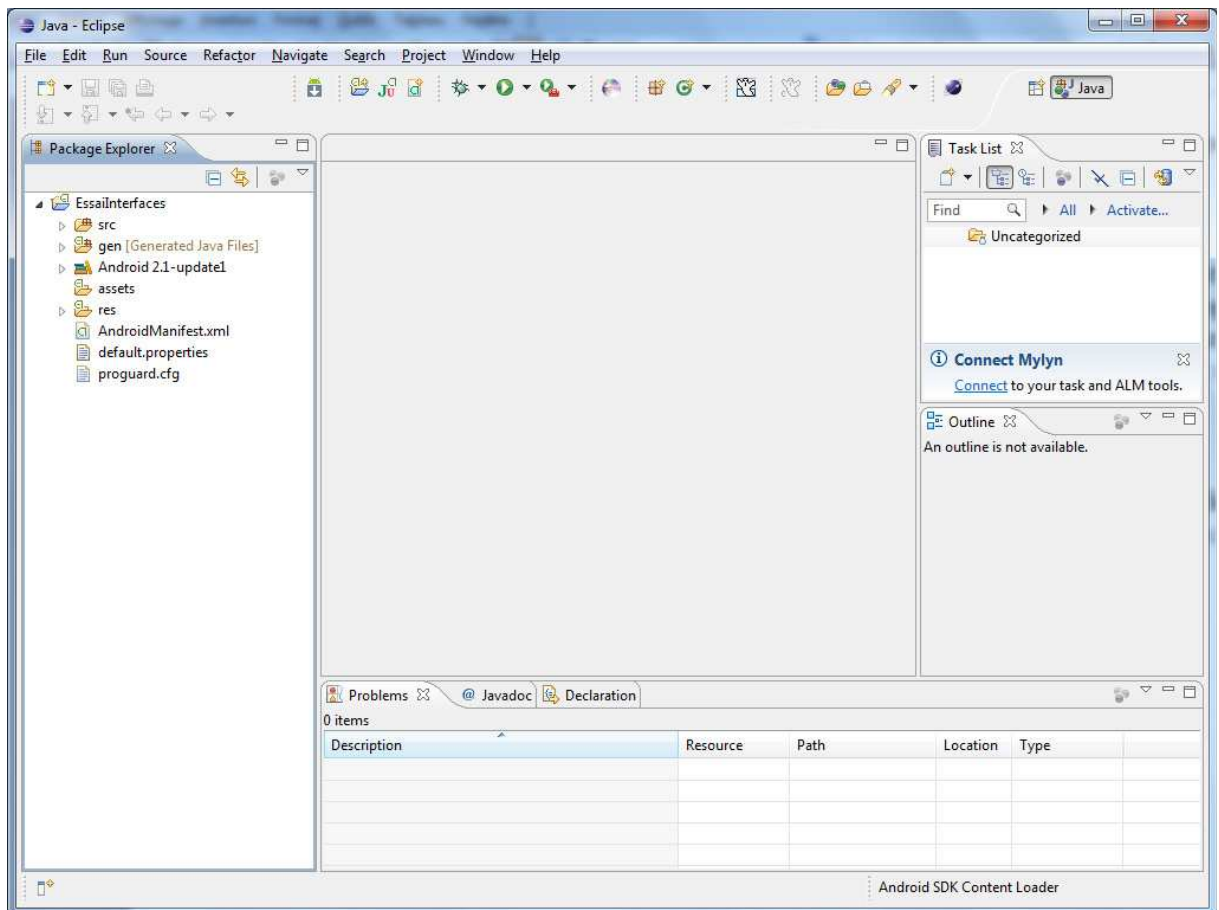
```
package nom_du_paquetage;  
  
import android.app.Activity;
```

- *gen* pour les fichiers générés automatiquement par Eclipse : contient un paquetage dont le nom est celui qui a été choisi au début dans lequel se trouve la classe R qui permet l'accès aux ressources.
- *res* pour les ressources : il contient 5 sous répertoires :
 - ✓ drawable-hdpi
 - ✓ drawable-ldpi
 - ✓ drawable-mdpi
 - ✓ layout
 - ✓ values

Les 3 *premiers* sont réservés aux images respectivement en haute/basse/moyenne définition. Au départ on y trouve l'icône de l'application (le robot vert d'android). Cette image est référencée par la balise `application android:icon="@drawable/icon"` dans le fichier `AndroidManifest.XML` et est associée à la valeur `public static final int icon=0x7f020000` dans la classe R.

Le 4^{ème} contient le fichier `main.xml` qui décrit l'interface principale de l'application.

Le 5^{ème} contient les ressources de type valeurs qui peuvent être référencées soit dans les autres fichiers xml (comme c'est le cas de la chaîne de caractères appelée "hello" dans le fichier `main.xml`) ou dans le code. Au départ il contient un fichier `"strings.xml"` dans lequel sont définies deux chaînes (celle mise dans le `TextView` et le nom de l'application).



Eclipse crée également un manifeste (AndroidManifest.XML) qui décrit l'application. Ce fichier peut être modifié directement ou par le biais d'interfaces présentées dans des onglets et correspondant aux divers types d'informations qu'il contient (versions, permissions, ...).

6. A partir de là on peut commencer à créer sa propre application en :

- o modifiant le fichier main.xml pour définir son interface
- o modifiant le fichier de l'activité pour y mettre le code propre à l'application
- o éventuellement modifiant de fichier AndroidManifest.XML en particulier si l'application a besoin d'autorisations particulières
- o modifiant le fichier strings. XML si on doit définir d'autres valeurs
- o ajoutant des ressources dans les sous répertoires du répertoire rep si nécessaire.

Compiler un projet Android avec Eclipse

Si dans le menu "projet" d'Eclipse "Build Automatically" est coché, le projet sera recompilé à chaque enregistrement de l'un des fichiers du projet.

Attention : modifier un fichier sans l'enregistrer ne provoquera pas de recompilation et c'est toujours l'ancienne version qui sera exécutée !

Si vous décochez cette case il faudra lancer la compilation par le menu "projet" en choisissant "Build Project".

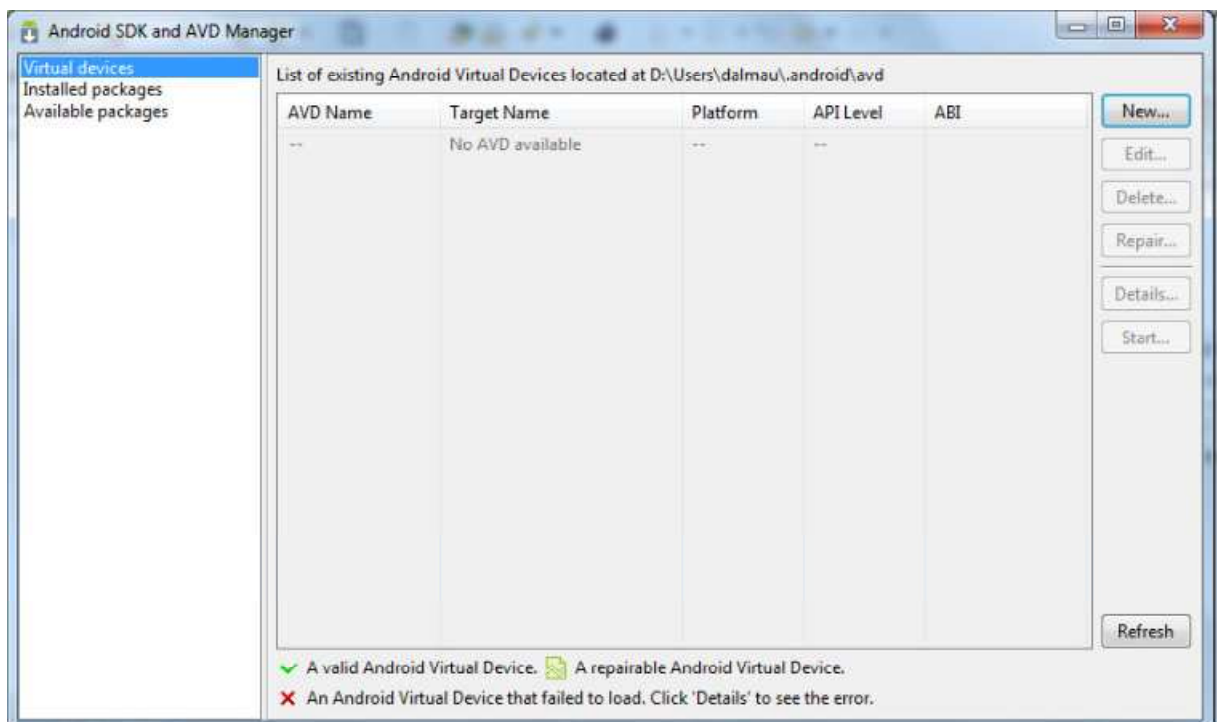
Créer un terminal virtuel pour tester les applications en simulation

L'exécution de votre application peut se faire :

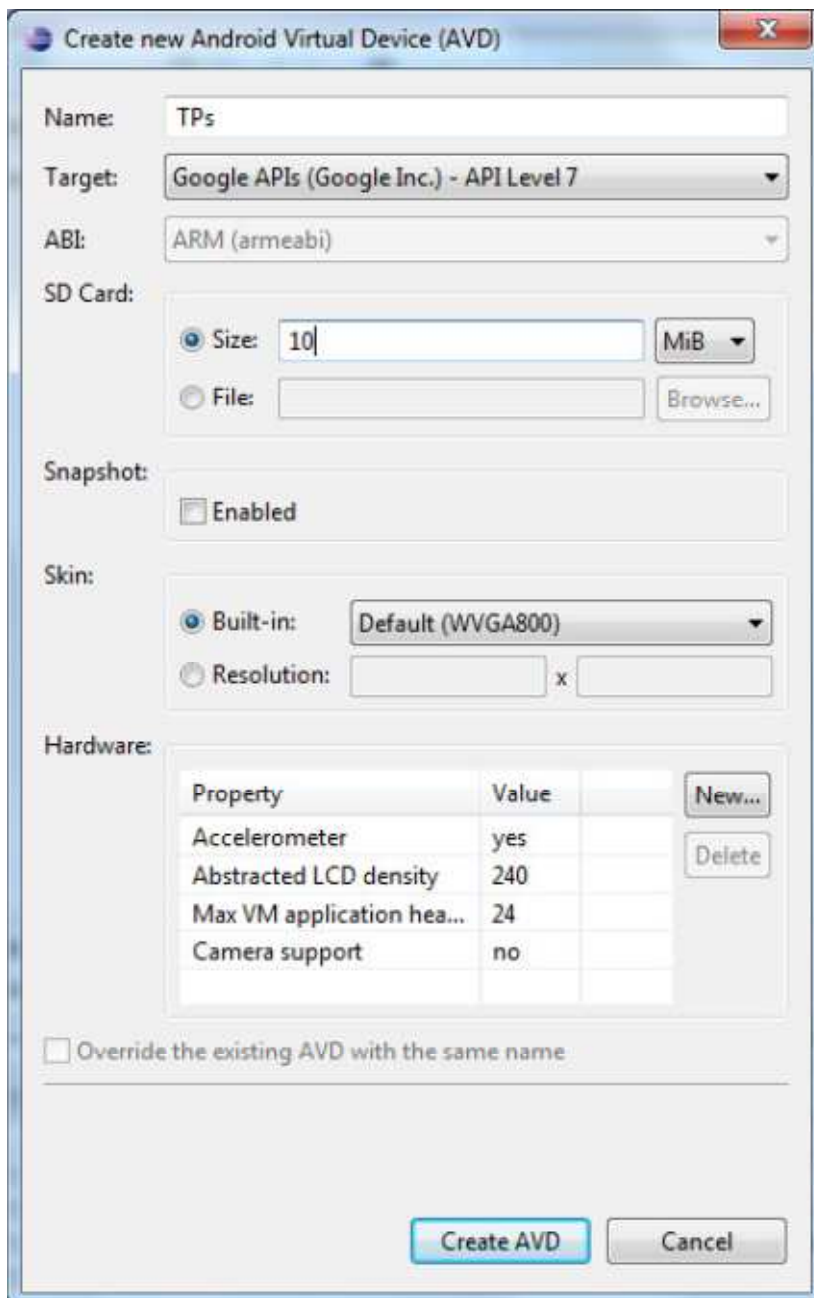
- sur un véritable terminal Android
- sur un simulateur

Pour pouvoir utiliser un simulateur il faut créer un terminal virtuel. On procède de la façon suivante :

Choisi le menu "Windows" puis "Android sdk and AVD Manager". La fenêtre suivante apparaît :



Cliquer sur bouton "new" pour faire apparaître cette fenêtre :



Choisir un nom pour votre terminal virtuel ;

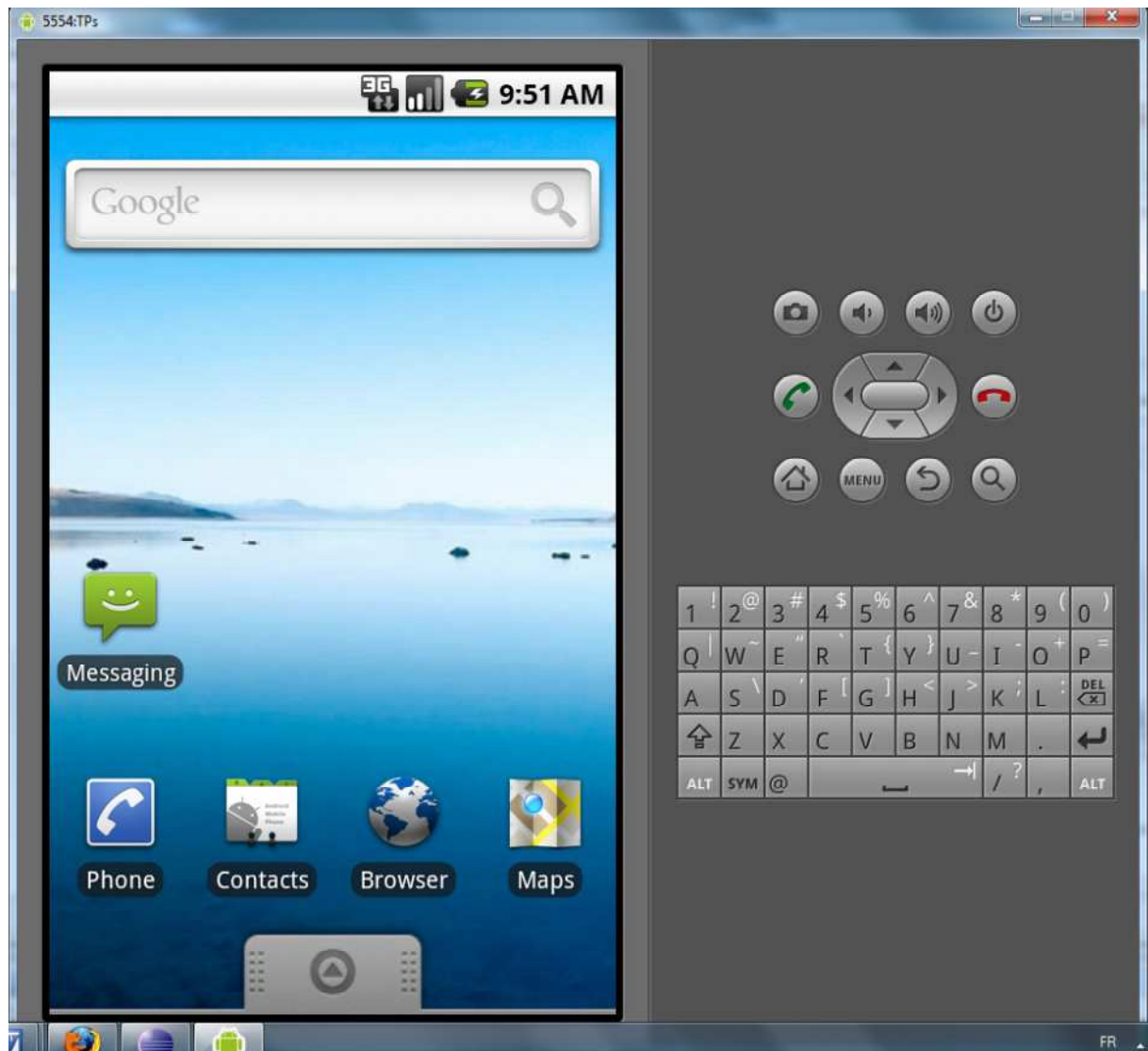
Choisir le sdk (comme lors de la création de l'application Android ou Google API) ;

Taille de la carte SD (10Mo suffisent bien) ;

Capteurs simulés : on peut ajouter des capteurs par le bouton "New" ;

Terminer par "Create AVD" ;

Votre terminal est créé, il pourra être utilisé pour tous vos tests. Voir la figure



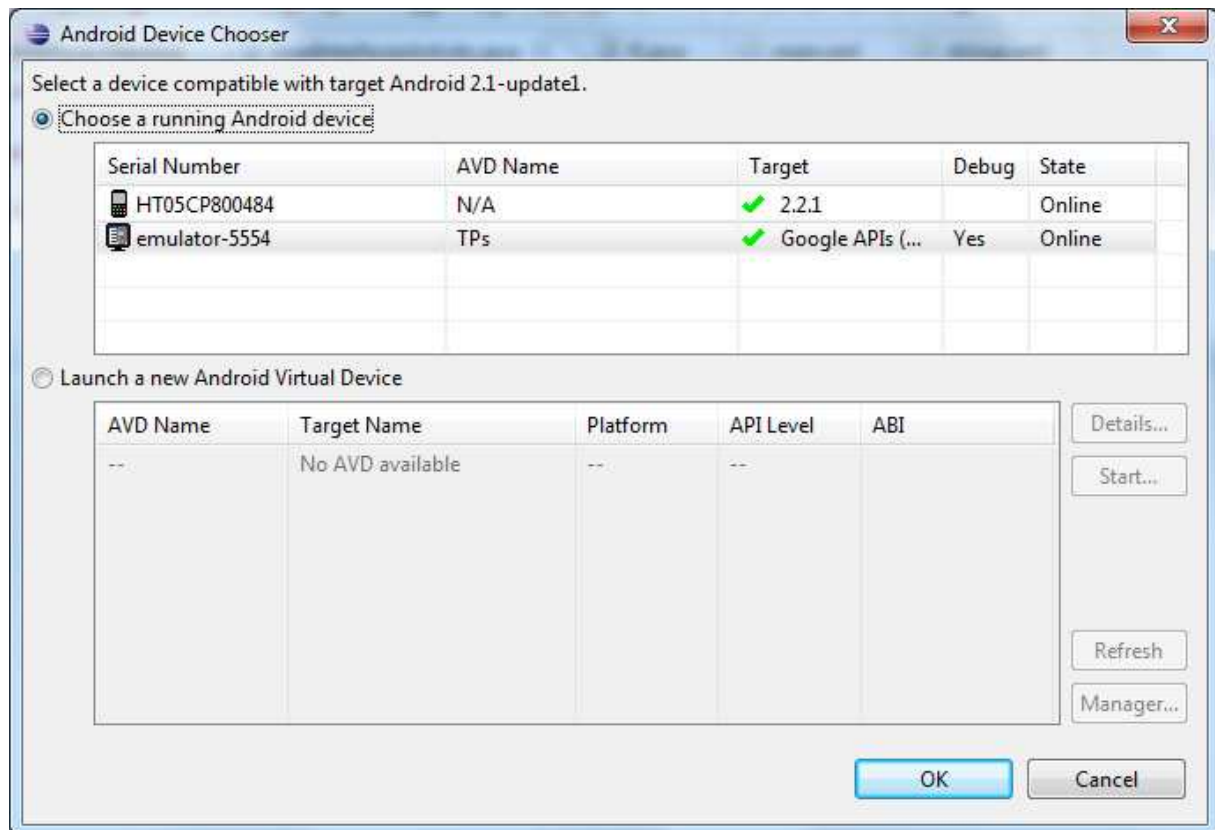
Exécuter un projet Android avec Eclipse

L'exécution peut se faire :

- sur un véritable terminal Android
- sur un simulateur

Le lancement de l'application est obtenu en faisant un clic droit sur le nom de l'application dans la fenêtre "Package Explorer" (à gauche), puis en choisissant dans le menu contextuel "Run as" puis "AndroidApplication". Le lancement se fera sur le terminal physique s'il y en a un connecté ou sur le terminal virtuel sinon.

Si plusieurs terminaux sont disponibles une fenêtre vous permettra d'indiquer celui que vous voulez utiliser (voir ci-dessous).

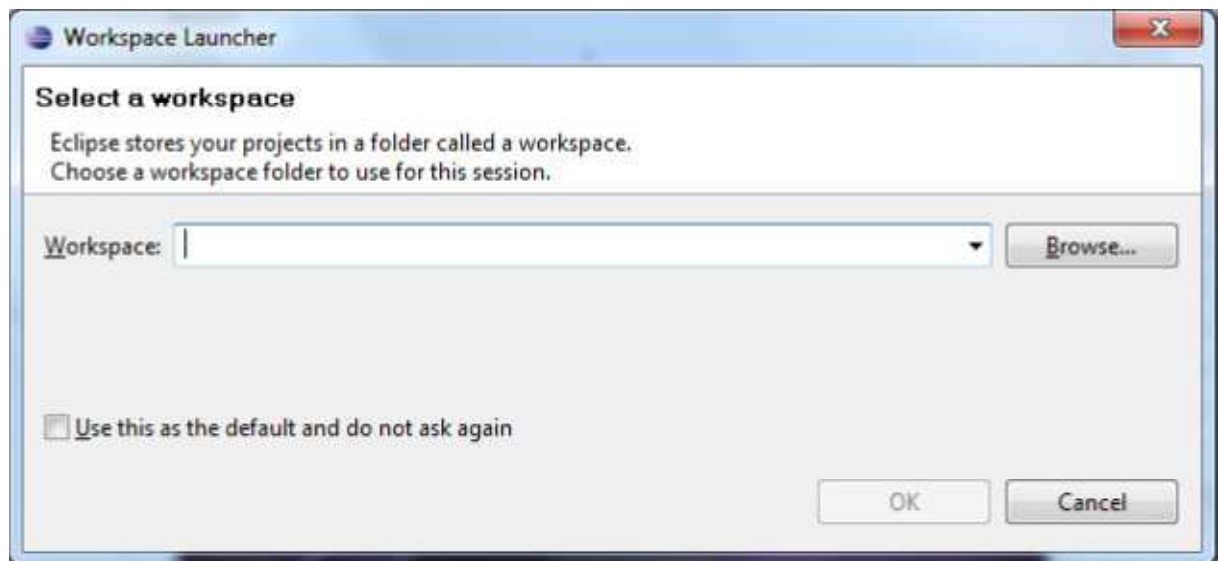


Le terminal virtuel apparaît à l'écran et s'initialise (c'est plutôt long) puis votre application y démarre. L'onglet "Console" en bas de la fenêtre d'Eclipse permet de suivre l'évolution de ce lancement de l'application.

Une fois lancé il est inutile de fermer ce terminal. Si vous modifiez votre application vous pouvez en exécuter la nouvelle version sur ce même terminal en refaisant "Run as" puis "Android Application".

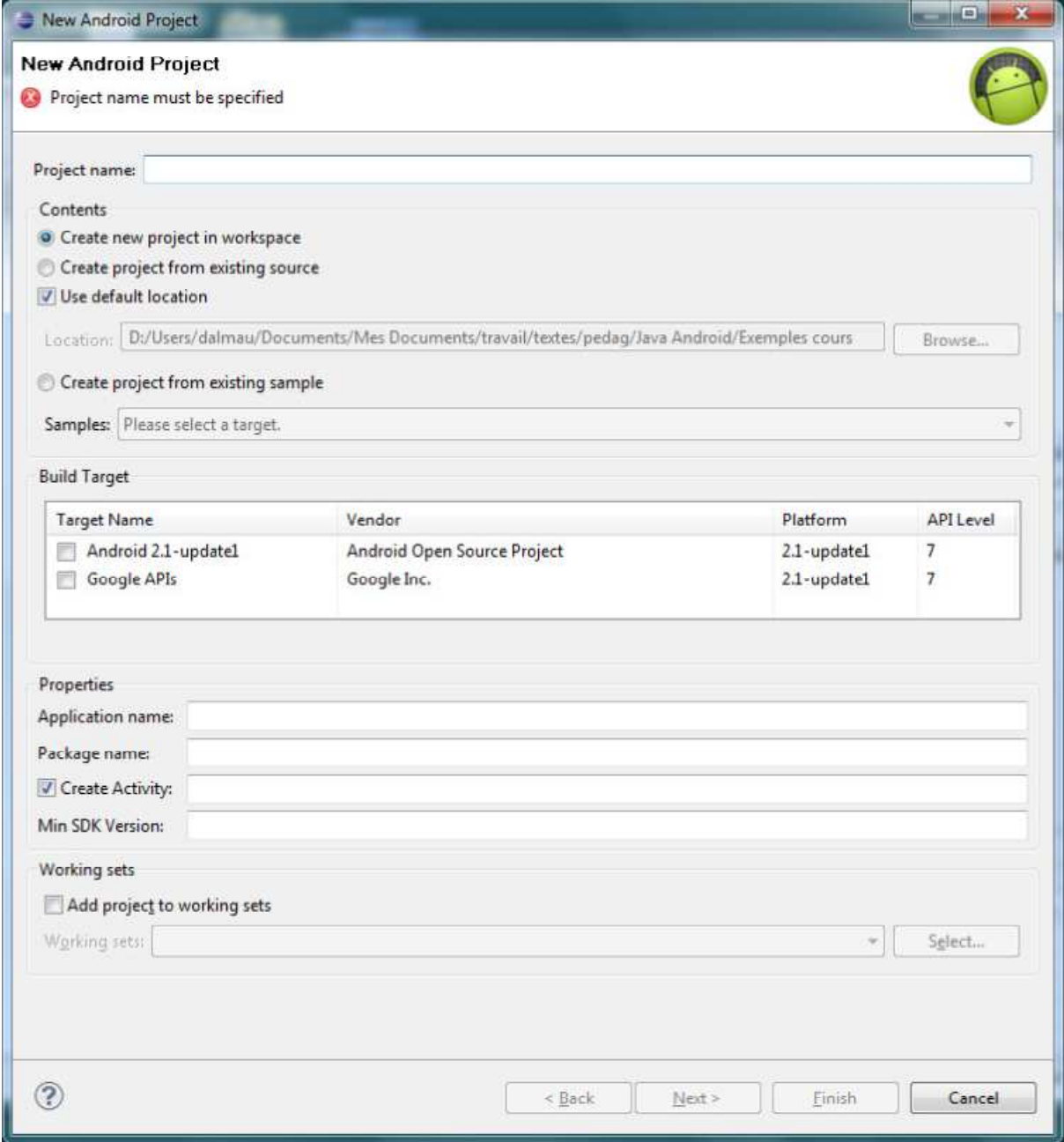
Créer un projet Android avec Eclipse

1. Lancer Eclipse
2. Choisir un répertoire de travail



3. Indiquer à Eclipse où se trouve le SDK Android : Windows -> Preferences puis dans la rubrique Android désigner l'emplacement du SDK par le bouton "Browse", terminer par "Apply" puis "OK"

4. Quand 'Eclipse a démarré aller dans le menu : File -> New -> Android project



New Android Project

Project name must be specified

Project name:

Contents

- ☒ Create new project in workspace
- ☐ Create project from existing source
- ☒ Use default location

Location:

☐ Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API Level
☐ Android 2.1-update1	Android Open Source Project	2.1-update1	7
☐ Google APIs	Google Inc.	2.1-update1	7

Properties

Application name:

Package name:

☒ Create Activity:

Min SDK Version:

Working sets

☐ Add project to working sets

Working sets:

5. Cliquer sur Finish, le projet est créé :
On y trouve les répertoires :

- *src* pour les sources : ce répertoire contient un paquetage dont le nom est celui qui a été choisi au début dans lequel se trouve la classe pour l'activité dont le contenu est :

```
package nom_du_paquetage;  
  
import android.app.Activity;
```

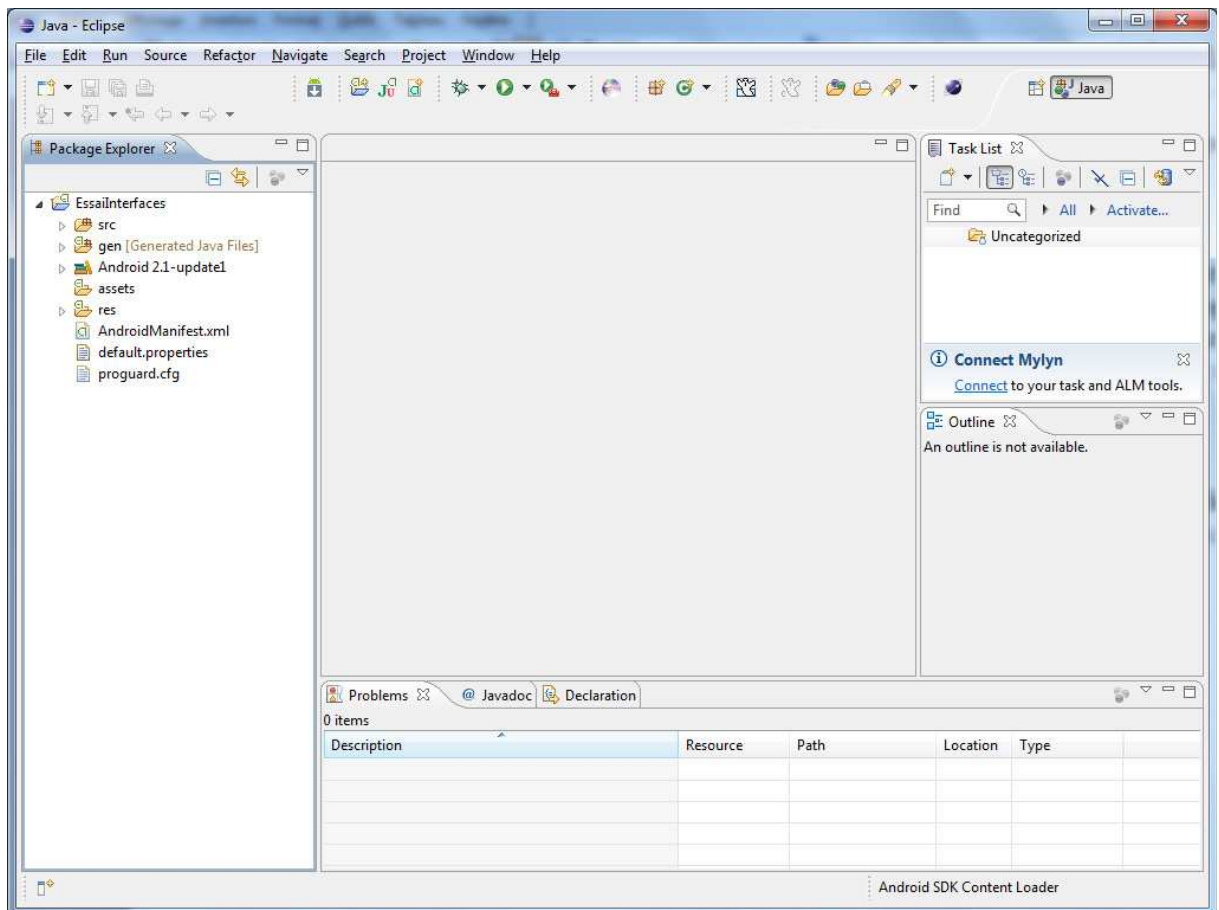
- *gen* pour les fichiers générés automatiquement par Eclipse : contient un paquetage dont le nom est celui qui a été choisi au début dans lequel se trouve la classe R qui permet l'accès aux ressources.
- *res* pour les ressources : il contient 5 sous répertoires :

- ✓ drawable-hdpi
- ✓ drawable-ldpi
- ✓ drawable-mdpi
- ✓ layout
- ✓ values

Les 3 *premiers* sont réservés aux images respectivement en haute/basse/moyenne définition. Au départ on y trouve l'icône de l'application (le robot vert d'android). Cette image est référencée par la balise `application android:icon="@drawable/icon"` dans le fichier `AndroidManifest.XML` et est associée à la valeur `public static final int icon=0x7f020000` dans la classe R.

Le 4^{ème} contient le fichier `main.xml` qui décrit l'interface principale de l'application.

Le 5^{ème} contient les ressources de type valeurs qui peuvent être référencées soit dans les autres fichiers xml (comme c'est le cas de la chaîne de caractères appelée "hello" dans le fichier `main.xml`) ou dans le code. Au départ il contient un fichier `"strings.xml"` dans lequel sont définies deux chaînes (celle mise dans le `TextView` et le nom de l'application).



Eclipse crée également un manifeste (AndroidManifest.XML) qui décrit l'application. Ce fichier peut être modifié directement ou par le biais d'interfaces présentées dans des onglets et correspondant aux divers types d'informations qu'il contient (versions, permissions, ...).

6. A partir de là on peut commencer à créer sa propre application en :

- o modifiant le fichier main.xml pour définir son interface
- o modifiant le fichier de l'activité pour y mettre le code propre à l'application
- o éventuellement modifiant de fichier AndroidManifest.XML en particulier si l'application a besoin d'autorisations particulières
- o modifiant le fichier strings. XML si on doit définir d'autres valeurs
- o ajoutant des ressources dans les sous répertoires du répertoire rep si nécessaire.

Compiler un projet Android avec Eclipse

Si dans le menu "projet" d'Eclipse "Build Automatically" est coché, le projet sera recompilé à chaque enregistrement de l'un des fichiers du projet.

Attention : modifier un fichier sans l'enregistrer ne provoquera pas de recompilation et c'est toujours l'ancienne version qui sera exécutée !

Si vous décochez cette case il faudra lancer la compilation par le menu "projet" en choisissant "Build Project".

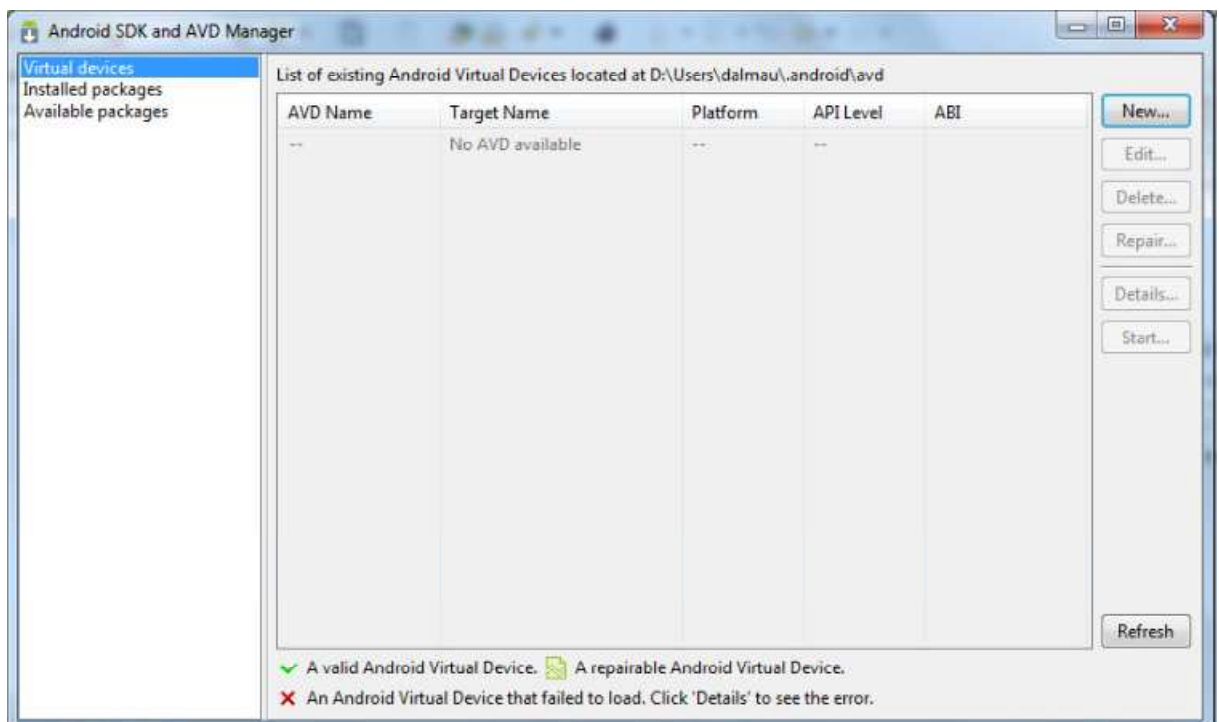
Créer un terminal virtuel pour tester les applications en simulation

L'exécution de votre application peut se faire :

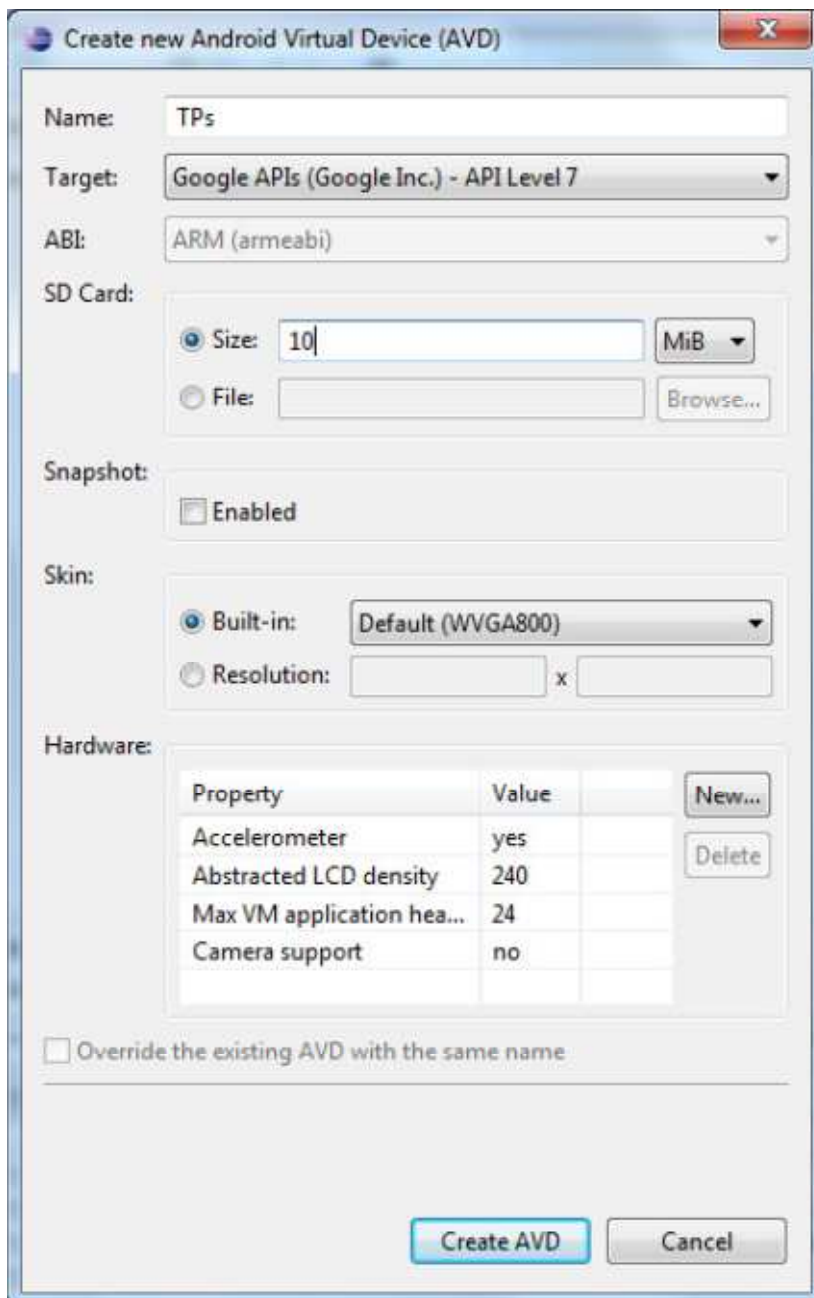
- sur un véritable terminal Android
- sur un simulateur

Pour pouvoir utiliser un simulateur il faut créer un terminal virtuel. On procède de la façon suivante :

Choisi le menu "Windows" puis "Android sdk and AVD Manager". La fenêtre suivante apparaît :



Cliquer sur bouton "new" pour faire apparaître cette fenêtre :



Choisir un nom pour votre terminal virtuel ;

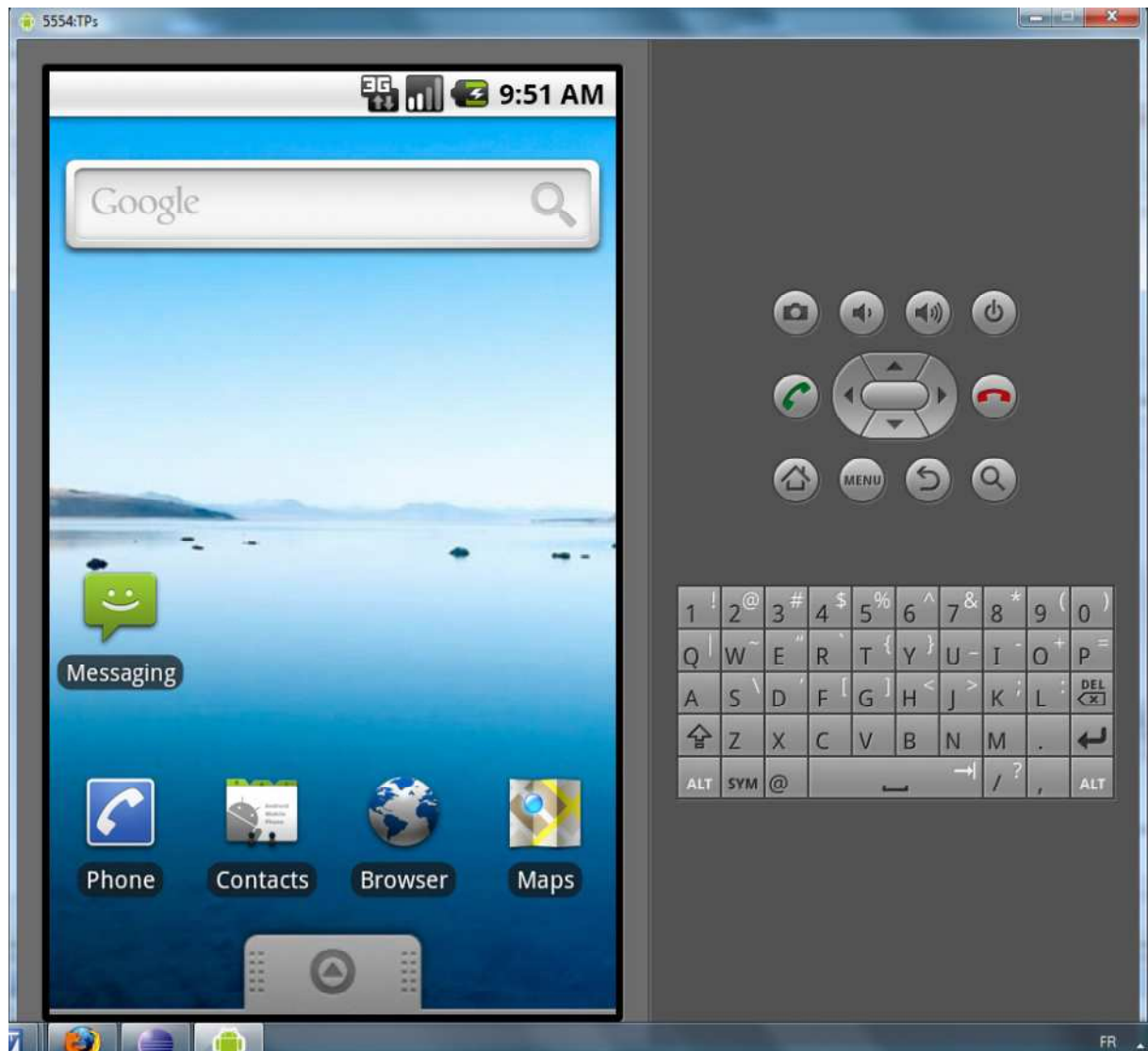
Choisir le sdk (comme lors de la création de l'application Android ou Google API) ;

Taille de la carte SD (10Mo suffisent bien) ;

Capteurs simulés : on peut ajouter des capteurs par le bouton "New" ;

Terminer par "Create AVD" ;

Votre terminal est créé, il pourra être utilisé pour tous vos tests. Voir la figure



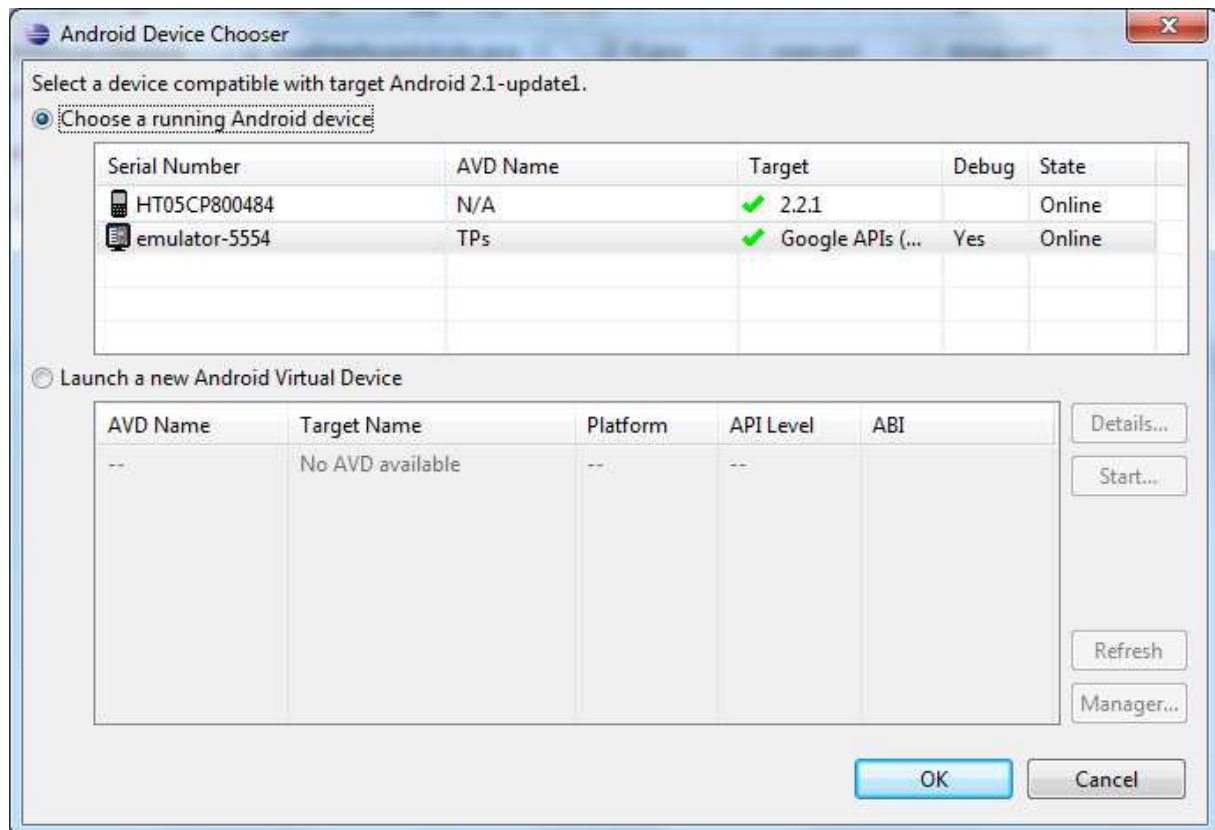
Exécuter un projet Android avec Eclipse

L'exécution peut se faire :

- sur un véritable terminal Android
- sur un simulateur

Le lancement de l'application est obtenu en faisant un clic droit sur le nom de l'application dans la fenêtre "Package Explorer" (à gauche), puis en choisissant dans le menu contextuel "Run as" puis "AndroidApplication". Le lancement se fera sur le terminal physique s'il y en a un connecté ou sur le terminal virtuel sinon.

Si plusieurs terminaux sont disponibles une fenêtre vous permettra d'indiquer celui que vous voulez utiliser (voir ci-dessous).



Le terminal virtuel apparaît à l'écran et s'initialise (c'est plutôt long) puis votre application y démarre. L'onglet "Console" en bas de la fenêtre d'Eclipse permet de suivre l'évolution de ce lancement de l'application.

Une fois lancé il est inutile de fermer ce terminal. Si vous modifiez votre application vous pouvez en exécuter la nouvelle version sur ce même terminal en refaisant "Run as" puis "Android Application".

Références bibliographiques

- [Bou08] J. BOUGEAULT, *"Java - La maîtrise 2ème édition - Java 5 et 6»*, 2 édition, Éditions Eyrolles, ISBN10 : 2212122500, ISBN13 : 9782212122503, 2008.
- [Bus11] *"l'Android Market grossit trois fois plus vite que l'App Store"* (businessmobile.fr) Par : ZDNet France. Publié le 17 février 2011
- [Cab12] Cabri-géomètre et Cabrijava et Genevière Tulloue, *"localisation par satellite G.P.S (1)"*, 2012
- [Cam12] AT&T Laboratories Cambridge. The Active Badge System. WebSite, consulté le: 10/05/2012 < The Active Badge System/>
- [Clz 08] M. CYPRIANI, F. LASSABE, S. ZIRARI, P. CANALDA, F. SPIES, *"Open Wireless Positioning System Version 0.8"*, Rapport Technique no RT2008-02 THÈME 4, Décembre 2008
- [Dou] N.DOUMA, M.ELHAMMOUTI. « Localisation dans un réseau de capteur », WebSite, < <http://projets-gmi.iup.univ-avignon.fr/cahierdecharge.pdf> > consulté le: 31/03/2013.
- N.Dewaele, « État de l'art des systèmes de localisation », Centre de

- [Dew11] Marne-la-Vallée, 13 mai 2011.
- [Eve07] F.EVENNOU « Techniques et technologies de localisation avancées pour terminaux mobiles dans les environnements indoor », Thèse de doctorat Université Joseph Fourier, Janvier 2007.
- [Gar11] "*Gartner Says Worldwide Mobile Device Sales to End Users Reached 1.6 Billion Units in 2010; Smartphone Sales Grew 72 Percent in 2010*", UK, February 9, 2011.
<<http://www.gartner.com> />, Consulté le : 20/05/2013
- [Hau] J-P.HAUET, « L'identification par radiofrequence (RFID) – techniques et perspectives radio frequency identification – technologies and perspectives ».
- [Ham05] A.HAMMAMI, «Etude et mise en place d'un système de gestion de parking automobile utilisant les technologies de communication sans fil», PFE 2005
- [Las09] F. LASSABE "*Géolocalisation et prédiction dans les réseaux Wi-Fi en intérieur*" Thèse de doctorat, Laboratoire d'Informatique de l'Université de Franche-Comté, avril 2009
- [MC03] A.MEDERREG, A.CAVALLI, M.MACKAYA, R.CASTANET, «Test de services basés sur la localisation» Article GRES Février 2003, Fortaleza-CE-Brésil

- [Mul02] J-P .MULLER, « Le réseau GSM et le mobile », version 07/2002
- [Par96] BW. PARKINSON, Jr. JJ. SPILKER, "*Global positioning system: theory and applications*". American Institute of Aeronautics and Astronautics, 1996.
- [Ray09] R.ROSSO, « Le développement du positionnement par satellites Réalités industrielles », MAI2009.
- [RM12] R.MEIER « Bases de données et fournisseurs de contenu », Android 4 Pearson France 2012.
- [Roq02] P. ROQUES , F. VALLEE, "*UML en action*», 2e édition, De l'analyse des besoins à la conception en Java», Eyrolles , 2002 (2e édition).
- [Say09] J. SAYAH, "*Contribution à la modélisation, à la simulation et à l'évaluation d'applications nomades à intelligence répartie – Application à l'assistance aux voyageurs aveugles dans les transports publics et les pôles d'échanges*", Thèse de doctorat, Université Paris_est Ecole doctorale mathématique et STIC, décembre 2009.
- [SBFC03] D. Schulz, W. Burgard, D. Fox, and Armin B. Cremers. "*People tracking with a mobile robot using sample-based joint probabilistic data association filters*". February 2003.
- [Tis99] J. TISAL, "*Le réseau gsm. l'évolution gprs : une étape vers umts*", troisième ed. Dunod, 1999.

- [TVi09] P-D.TRUER, Olivier VIATOUR « Geopositionnement par WiFi Indoor », UNIVERSITE de FRANCHE-COMTE, jeudi 10 décembre 2009
- [Uml8] Note de Cours UML, <www.math-info.univ-paris5.fr/~bouzy/Doc/UML-NotesCours.pdf>, consulté 15/05/2013.
- [Wei91] M.WEISER, "*The computer of the 21st century*", Scientific American, vol. 265, no 3, 1991.