

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Licence en Informatique

Thème

**Conception et Développement d'une
Application Android
Géo localisation des médecins conventionnés**

Réalisé par :

- MALTI Ahmed Choukri
- MESSAOUD Mohammed

Présenté le 07 Septembre 2014 devant la commission d'examinations composée de

- Mr Smahi Ismal..... (Encadreur)
- Mr Boudefla Amine (Examineur)
- Mr Merzoug Mohammed (Examineur)

Dédicace

Je dédie ce projet :

*Rien n'est aussi beau à offrir que le fruit d'un labeur
Qu'on dédie du fond du cœur à tous ceux qu'on aime
Et qu'on remercie en exprimant la gratitude et la
Reconnaissance durant toute notre existence.
Je dédie avec grande joie, ce modeste travail :*

*A ma défunte mère paix à son âme, qui malgré son absence a toujours été la dans mon cœur
Et le sera à jamais*

A mon très cher père et ma deuxième mère qui ont toujours été la pour moi durant mon cursus. Vous étés pour moi un exemple de courage et de sacrifice continu, que cet humble travail témoigne mon affection, mon éternel attachement et qu'il appelle sur moi votre continue bédiction.

*A ma chère épouse qui m'a épaulé pour que je puisse atteindre mes objectifs
A mon cher frère, mes chères sœurs
A mes adorables petits neveux : Yasmine, Yacine et Sirine et mon beau-frère
A mes oncles, mes tantes, cousins et cousines
A toute la famille MALTI
ET ma belle famille*

*A tous mes professeurs qui m'ont encadré durant tout mon cursus
A toutes les personnes qui m'ont soutenu
A tous mes aimables collègues et amis
A tous ceux qui m'aiment*

Malti Ahmed Choukri

Dédicace

Je dédie ce projet :

A ma chère mère,

A mon cher père,

*Qui n'ont jamais cessé, de formuler des prières à mon égard, de me soutenir
et de m'épauler pour que je puisse atteindre mes objectifs.*

A mes sœurs Fatima et Nacera

Pour ses soutiens moral et leurs conseils précieux tout au long de mes études.

A mes chères neveux Mohammed et Marwa,

A toute ma famille,

A tous mes amis,

A tous ceux que j'aime et ceux qui m'aiment.

Messaoud Mohammed

Remerciements

Nous sommes très reconnaissants envers tous ceux qui, par leurs compétences scientifiques et leurs qualités humaines, ont contribué au bon déroulement de ce mémoire.

*Nous adressons nos sincères remerciements à **Mr. Boudefla Amine** et **Mr. Merzoug Mohammed** membres du jury, pour avoir accepté d'être le rapporteur de ce manuscrit.*

*Nous tenons à remercier **Mr. Smahí Ismaíl**, pour ses valeureux conseils et pour la confiance et la sympathie qu'il nous a accordée en acceptant de nous encadrer et qu'il nous a témoignée au cours de ce projet de Fin d'études.*

Sommaire

Introduction générale.....	5
Chapitre 1 Contexte et Problématique	7
1. Cadre de projet et problématique	8
1.1. Cadre du projet.....	8
1.1.1. Présentation de la société	8
1.1.2. Cahier de charge.....	10
1.1.3. Outils utilisées	10
1.2. Problématique	10
2. Etude de l'existant.....	11
3. Objectifs	11
Conclusion.....	12
Chapitre 2 Systemes d'exploitationmobile -ANDROID-	13
1. Internet mobile.....	14
2. Technologie de téléphonie mobile	14
2.1. Technologie 3G.....	14
2.2. Technologie 3G+.....	14
2.3. Technologie 4G.....	14
2.4. Smartphone	15
3. Système d'exploitation mobile.....	15
3.1. Les différents systèmes d'exploitation sur le marché	15
3.2. Partition du marché entre les SE	18
4.Le système d'exploitation Android	19
4.1. Définition et historique	19
4.2. Les versions de l'Android	19
4.3. Architecture logicielle	24
4.4. Kit de développement	25
5. La Géolocalisation	26
6. La sécurisation des communications	27
6.1. La définition du protocole HTTPS.....	27
6.2. Les objectifs de sécurité assurés.....	28
Chapitre 3 Analyse et Conception.....	29
1. Identification et analyse des besoins.....	30
1.1. Identification des besoins	30
1.1.1. Les besoins fonctionnels	30
1.1.2. Les besoins non fonctionnels	30
1.2. Analyse des besoins	31
1.2.1. Cas d'utilisation relatif à l'administrateur.....	31
1.2.2. Cas d'utilisation relatif à l'assuré.....	32

11.3. La description détaillée des cas d'utilisation	32
1.3.1. Scénario du cas d'utilisation "consulter les medecin a proximite"	32
2. Conception détaillée	33
2.1. Les diagrammes de séquence	33
2.1.1. Diagramme de séquence du scénario "consultation des medecin a proximite"	33
3. Diagramme de classe.....	35
Chapitre 4 Réalisation	37
1. Environnement de développement.....	38
1.1. Installation des outils.....	38
1.2. Paramétrage d'Eclipse et installation du plugin ADT.....	38
1.3. Téléchargement des différents composants d'Android SDK.....	39
1.4. Création d'un AVD	40
1.5. Bibliothèques externes	41
1.6. Fichier source d'un projet	41
1.6.1. Java.....	41
1.6.2. XML	41
1.7. Intégration de Google Map	42
2. Persistance de donnée	42
Modèle logique du système d'information.....	42
3. Résultats obtenus	43
Conclusion.....	43

LISTE DES FIGURES

Figure 1 . Les parts de marché des systèmes d'exploitation mobile pour les années 2011 et 2014.....	13
Figure 2. La part de chaque version d'Android.....	15
Figure 3. Architecture logicielle de l'Android.....	15
Figure 4. Diagramme de cas d'utilisation relatif à l'administrateur	24
Figure 5. Diagramme de cas d'utilisation relatif à assuré.....	25
Figure 6. Cas d'utilisation "consulter les medecin a proximité"	25
Figure 7. Diagramme de séquence du scénario de consultation des medecin a proximité ..	34
Figure 8. Diagramme de classe	35
Figure 9. URL Plugin.....	38
Figure 10. Liste des targets	39
Figure 11. Android SDK AVD Manager	39
Figure 12. Liste des AVD créés	40
Figure 13. Récupération du clé MD5 pour Google MAP	42
Figure 14. Obtention de la clé API.....	43
Figure 15. Interface d'accueil de l'application.....	44
Figure 16. Interface de la Recherche.....	45
Figure 17. Interface choix du resultat	46
Figure 18. Liste medecin et leur géo localisation.....	46

Liste des tableaux

Tableau 1 <i>Une comparaison entre les systèmes d'exploitation mobile.</i>	12
Tableau 2. <i>Les versions de l'Android</i>	14
Tableau 3 . Cas d'utilisation « consulter les mesdecins a proximité ».....	26

LISTE DES ABREVIATIONS

WAP: *Wireless Application Protocol*

UMTS: *Universal Mobile Telecommunications System*

HSDPA: *High Speed, Downlink Packet Access*

LTE: *Long Term Evolution*

WIMAX: *Worldwide Interoperability for Microwave Access*

GPS: *Global Positioning System*

OS: *Operating System*

IOS: *Internetwork Operating System*

RIM: *Research In Motion*

VB: *Visual Basic*

IDC: *International Data Corporation*

OHA: *Open Handset Alliance*

SDK: *Software Development Toolkit*

API: *Application Program Interface*

AVD: *Android Visual Device*

HTML: *Hypertext Markup Language*

HTTPS: *Hypertext Transfer Protocol Secured*

HTTP: *Hypertext Transfer Protocol*

XML: *eXtensible Markup Language*

SGML: *Standard Generalized Markup Language*

SGBD: *System Management Databases*

Introduction générale

"Les mobiles sont aussi différents de l'internet que la télé l'a été de la radio", a annoncé -Tomi Ahonen-, meilleur auteur technologique et médiatique à succès, consultant de stratégie et orateur motivationnel.

Les téléphones portables sont devenus les premiers médias de masse dans le monde (on compte 6,8 milliards d'abonnés au téléphone mobile). Il n'est pas un PC plus bête, mais un téléphone portable est considéré comme un autre support à générer des formes médiatiques.

Il a subi une évolution à une vitesse surprenante, passant du premier téléphone portable inventé par Dr Martin Cooper -un directeur général de Motorola- en 1973 qui a été la première personne à faire un appel depuis son téléphone portable, aux ordiphones, PDA et Smartphones qui disposent d'un système d'exploitation adoptant des applications tierces qui leur sont dédiées.

L'invention du premier PDA au monde, Le PenPad conçu par Apple, était dans le but de pouvoir prendre des notes, gérer son agenda, ses adresses, effectuer des calculs, etc, sans avoir à s'encombrer d'un ordinateur portable ou d'un bloc notes.

Aujourd'hui ces périphériques ont atteint une puissance de calcul, une taille mémoire ainsi qu'un débit nécessaire pour faire tourner des applications aussi diverses que variées qui vont de l'Outlook mobile jusqu'aux applications de navigation GPS.

Les plates-formes de distribution de ces applications sont en plein essor, Windows Phone 7 à son magasin d'applications, l'iPhone à l'App Store, Android à son Market, etc.

Ce qui ne cesse d'inciter beaucoup de développeurs à l'élaboration des petits logiciels très prisés en profitant des multiples apports des plateformes présentes sur le marché et des diverses innovations technologiques (Wifi, GPS, GPRS, 3G, 4G etc.).

En effet, la géo localisation du GPS des « téléphones intelligents » est très utile aux applications comme les annuaires, portails et autres outils permettant de trouver ce que l'on cherche autour d'un lieu, répondant par la fin aux besoins quotidiens de la communauté.

C'est dans ce cadre que s'inscrit notre projet de fin d'étude, intitulé « Médecins Conventionnés >> dont l'objectif est de concevoir une application dédiée au téléphone mobile, doté de la plateforme Android, permettant au assurés la géo localisation des médecins conventionnés avec la CNAS.

Le travail effectué a fait l'objet de quatre chapitres. Le premier chapitre présente le cahier de charge. Le deuxième chapitre présente les technologies utilisées pour le développement de notre application. Le troisième chapitre détaille la phase d'analyse et de conception à l'aide des diagrammes d'UML. Le quatrième chapitre porte sur la réalisation, expose l'environnement de développement ainsi que les résultats. Enfin, nous présentons une synthèse ainsi que les perspectives en raison d'améliorer les performances de l'application.

Chapitre 1

Contexte et Problématique

1. Cadre de projet et problématique

1.1. Cadre du projet

1.1.1. Présentation de la société

Tout acquis, informations ou connaissances, ne peuvent être vérifiés et concrétisés que sur le plan pratique et professionnel. Ainsi, dans ce cadre, nous avons réalisé notre projet au sein de la caisse nationale de l'assurance sociale (CNAS). Ce projet vient compléter notre formation scientifique et technologique et améliorer nos compétences dans un environnement réel.

Ce chapitre a pour objectif de situer notre sujet dans son contexte général. Nous commençons par une présentation de la société d'accueil « CNAS ». Ensuite, nous décrivons brièvement le sujet à traiter et les objectifs à atteindre, ainsi que la méthodologie de travail adoptée.

La CNAS est un établissement public à gestion spécifique en application de l'article 49 de la loi n° 88-01 du 12 janvier 1988, elle est dotée de la personnalité morale et de l'autonomie financière, et réputée commerçante dans ses relations avec les tiers.

Attributions de la CNAS :

- Gérer les prestations des assurances sociales (maladie, maternité, invalidité, décès) ainsi que des accidents du travail et maladies professionnelles (AT/MP)
- Gérer les allocations familiales pour le compte de l'Etat .
- Assurer le recouvrement, le contrôle et le contentieux du recouvrement des cotisations destinées au financement des prestations.
- Attribuer un numéro d'immatriculation national aux assurés sociaux et aux employeurs .
- Contribuer à promouvoir la politique de prévention des AT/MP et gérer le fonds de prévention AT/MP .
- Gérer les prestations dues aux personnes bénéficiaires des accords bilatéraux de sécurité sociale .
- Exercer le contrôle médical des bénéficiaires .
- Entreprendre des actions en vue de faire bénéficier les travailleurs et leurs ayants droit de prestations collectives, sous forme de réalisations à caractère sanitaire et social .
- Gérer le fonds d'aide et de secours .
- Conclure des conventions avec les prestataires de soins .

- Assurer l'information des bénéficiaires et des employeurs .

Les Bénéficiaires :

- Les travailleurs salariés, quel que soit le secteur d'activité
- Les apprentis
- Les bénéficiaires des emplois d'attente
- Les étudiants
- Les stagiaires de la formation professionnelle
- Les handicapés
- Les moudjahiddines (anciens combattants)
- Les titulaires d'avantages de sécurité sociale (pensionnés et rentiers)
- Les bénéficiaires de l'allocation forfaitaire de solidarité (personnes malades ou âgées et inactives)

Il faut ajouter les ayants droit qui sont :

- Le conjoint
- Les enfants mineurs
- Les filles inactives non mariées
- Les ascendants à charge

Prestation

- Les soins de santé et les médicaments sont pris en charge à 80 % et dans certains cas à 100 % (malades chroniques notamment)
- L'indemnisation des arrêts de travail pour maladie représente 50 % du salaire pendant les 15 premiers jours. Elle est portée à 100 % du salaire au-delà du 16ème jour.
- La durée maximale de cette indemnisation est de trois ans.
- Les prestations de l'assurance maternité sont prises en charge à 100 %; la femme travailleuse bénéficie d'un congé de maternité de 98 jours.
- Le montant minimum des pensions d'invalidité est égal à 75 % du SNMG.
- Au décès de l'assuré, il est servi un capital décès à ses ayants droit.
- Les risques professionnels donnent lieu à une couverture à 100 % pour les soins et les arrêts de travail.
- Des rentes sont versées en cas de séquelles corporelles de l'accident.
- Des rentes sont servies aux ayants droit en cas d'accident mortel.

- Des retraités, assurés sociaux et à leurs ayants droit; bénéficient de consultations médicales gratuites, dans le cadre de l'application de la convention signée entre les médecins traitants et la Sécurité sociale.

1.1.2. Cahier de charge

- Cette application à pour but de satisfaire les besoins des assurés qui peuvent utiliser l'application et requiert une authentification.
- Un assuré peut consulter le profil d'un médecin et aussi sa localisation.

1.1.3. Outils utilisées

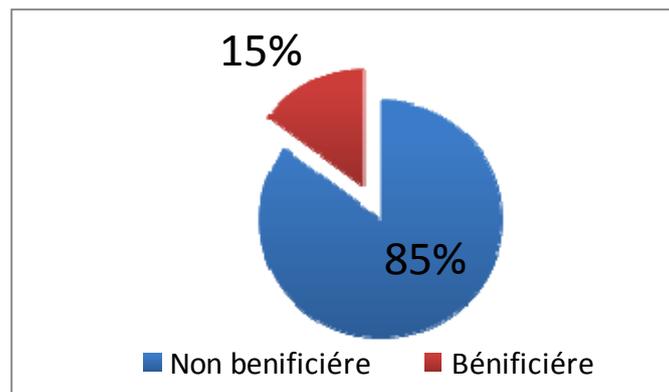
Pour crée une application, nous avons utilisé Eclipse qui est la version la plus récente du projet de niveau supérieur, et open source qui nous permet de créer des applications mobiles utilisant des API standard pour les plates-formes qui nous intéressent.

Il contient ce dont vous avez besoin pour créer des applications basées sur la technologie Eclipse, y compris les environnements intégrés de développement (IDE), et les applications clientes riches en utilisant la plate-forme Eclipse Rich Client (RCP). L'Eclipse fournit édition Java de qualité supérieure avec la compilation incrémentielle, la prise en environnement de développement (PDE), le code source complet de la plate-forme Eclipse, et bien plus encore .

1.2. Problématique

- Les assurés sont mal informés sur les services et prestations de la CNAS, à titre d'exemple les retraités ont le droit a une prestation d'un médecin traitant conventionné avec la CNAS ; Mais la majorité trouvent des difficultés à localiser ces médecins, généralistes ou spécialistes.

Le nombre totale des assurés retraités est de 80587, seulement 11980 soit 15% sont bénéficière de cette conventions.



- On perçoit plus de 150 médecins généralistes et 18 spécialistes, répartie à travers toute la wilaya de Tlemcen.
- L'amélioration de la qualité de services est un challenge que tout acteur dans le domaine professionnel cherche à atteindre. Afin d'y parvenir, il est primordial de proposer de nouvelles technologies d'informations et de communications.
- Notre objectif à travers ce projet est de proposer une solution répondant aux besoins des assurées et les aider dans leur quête.

Pour bénéficier d'une telle application, il faut avoir accès à la technologie du GPS et un Smartphone connecté à internet.

2. Etude de l'existant

Les technologies mobiles font parti de la nouvelle mutation pour propager l'information. En effet, avec cette technique, nous aurons la possibilité d'accéder aux informations et au service de géo localisation grâce à un système de communication sans fil, basé sur les téléphones mobiles. Le but de notre travail est de faciliter la recherche des médecins conventionnés et de leurs emplacements en utilisant la possibilité de géo localisation. Cette application permet aux utilisateurs de consulter et d'afficher les médecins conventionnés les plus proches de leurs emplacements actuels.

3. Objectifs

L'application "Médecins Conventionnés" à développer doit fournir les services suivants :

- Visualiser la position actuelle sur une Map.
- Consulter la liste des médecins conventionnés.
- Consulter la localisation et les informations détaillées relatives à un médecin sélectionné.
- Localisation des médecins conventionnés a proximité de l'endroit de l'assuré.
- Permettre à l'administrateur de gérer la liste des endroits qui se trouvent dans la base de données avec leurs promotions s'ils existent.

Conclusion

Ce chapitre constitue une présentation générale du sujet ainsi que les objectifs à atteindre en développant cette application.

Notre objectif est de proposer une solution répondant aux besoins des assurées et les aider à géo localiser les médecins conventionnés avec la CNAS.

Chapitre 2
Systemes d'exploitation
mobile –Android-

Avant d'entamer le développement de l'application, nous allons présenter quelques éléments d'initiation en liaison avec notre projet (Internet mobile, Android, ...etc). Pour cela, nous commençons par définir l'Internet mobile, le système d'exploitation mobile en détaillant celui d'Android, système d'exploitation avec lequel on a développé notre application. Enfin, nous définissons la sécurisation des communications et des protocoles.

1. Internet mobile

L'internet mobile est différent de l'internet fixe à travers les ordinateurs. Cette distinction se constitue dans l'ergonomie des sites web qui doit respecter l'affichage sur des petits écrans. La technologie WAP (*Protocole d'application sans fil*) a permis d'avoir un service Internet pour les mobinautes. Mais en raison des ressources limitées des ces appareils à cette période ont empêché la diffusion de cette technologie.

Maintenant avec les nouvelles normes de technologie de téléphone mobile, le nombre de personnes dans le monde utilisant l'Internet comme service mobile atteint 2 milliards.

2. Technologie de téléphonie mobile

2.1. Technologie 3G

Nommée aussi "technologie de troisième génération", elle est devenue disponible au public depuis 2013 et elle se base sur la norme de communication UMTS (*Système de télécommunications mobiles universelles*). Elle peut atteindre un débit égal à 2 Mbps (244Ko/s) à partir d'un lieu .

2.2. Technologie 3G+

La 3G+ est une technologie qui permet d'échanger les données de façon plus rapide et dans des tailles plus importants grâce à l'association simultanée des systèmes HSDPA (*High Speed, Downlink Packet Access*) jusqu'à 3 à 5 fois plus rapide que les technologies précédentes. La 3G+ amène une meilleure connexion Internet en mobile.

2.3. Technologie 4G

La 4G est le terme utilisé pour désigner la prochaine vague de technologies mobiles haut débit qui seront utilisés pour remplacer les réseaux 3G actuels. Les deux principaux

prétendants sont LTE (*Long Term Evolution*) et WiMAX (*Worldwide Interoperability for Microwave Access*).

2.4. Smartphone

C'est un téléphone mobile disposant des performances proches de l'ordinateur. Par rapport aux téléphones standards, les Smartphones ont généralement des écrans plus larges et des processeurs plus puissants. La saisie des données se fait par le biais d'un écran tactile ou d'un clavier. Il fournit des fonctionnalités basiques comme : l'agenda, le calendrier, la navigation sur le web, la messagerie instantanée et ainsi le GPS (*Système de localisation du véhicule par satellites*).

Il dispose d'un OS (système d'exploitation) embarqué pour l'exploitation de ces capacités : mémoire, le processeur, le capteur, etc.

3. Système d'exploitation mobile

Le système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Pour le faire, il faut qu'il soit non seulement robuste mais suffisamment flexible pour effectuer des tâches qui dépassent le champ que l'on connaît dans la micro-informatique. Cela revient à la richesse du monde mobile. En plus, le Smartphone comporte beaucoup plus de défis que les stations de travail fixes. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface.

3.1. Les différents systèmes d'exploitation sur le marché

Il existe sur le marché des dizaines de systèmes d'exploitation différents : Symbian OS de Nokia, ios d'Apple, BlackBerry OS de RIM, Windows Phone de Microsoft, Palm webOS, Bada de Samsung et Android de Google.

Symbian OS

Le Symbian OS est développé par la société éponyme qui est une propriété exclusive de Nokia. Bien que cette plateforme soit créée par la participation de plusieurs fabricants tels que Samsung ou Sony Ericsson, ce système est fortement connoté Nokia, ce qui est un frein à son adoption par d'autres constructeurs. Il est récemment passé en open source. C'est un système libre, open source se base sur un noyau Symbian.

Ios

IOS (*Internetwork Operating System*), qui était nommé iPhone OS, se trouve non seulement sur les différentes générations de iPhone mais également sur d'autres produits de Apple iPad et iPod touch. Il est dérivé de Mac OS X dont il partage les fondations : kernel, les services Unix et Cocoa. Pour Apple, le succès est considérable : début 2009, il n'y avait pas moins de 5 millions de téléchargements par jour. Donc, il s'agit du concurrent numéro un pour Android.

BlackBerry OS

Le système d'exploitation BlackBerry est la plate-forme exclusive mobile développé par RIM (*Research In Motion*) exclusivement pour ses Smartphones BlackBerry et les appareils mobiles. RIM utilise ce système d'exploitation pour soutenir des fonctions spécialisées, notamment le trackball de la marque, molette, le trackpad et l'écran tactile.

Windows Phone

Windows Mobile, WiMo pour les intimes, est l'OS (système d'exploitation) mobile de Microsoft. C'est une évolution de Windows Pocket PC, ancêtre de Windows CE. Cet OS a réussi au fil des années à s'octroyer une part de marché honorable. Son succès est dû à son affiliation à la famille d'OS Windows, ultra-dominante sur le bureau. Un autre avantage souvent cité est la facilité de développement apportée grâce à l'environnement cliquodrome de Visual Studio qui a su faire venir au développement mobile les développeurs VB (*Visual Basic*).

Palm webOS

Il y a quelques années, Palm a même cédé aux sirènes de Windows Mobile en proposant certains de ses appareils sous l'OS de Microsoft. Palm avait cessé d'innover et devrait réagir face aux assauts d'Apple et de Google.

	Symbian OS	Ios	Blackberry	Windows Phone	Palm webOS	Android
Langage de programmation	C++	Objective-C	Java	C, C++	HTML, CSS, JavaScript, JSON, etc.	Java
	gratuit	Intégré à Xcode	Gratuit	Gratuit	Gratuit	Gratuit
Disponibilité de l'environnement de développement	Carbide.C++	Xcode	JDE	Visual Studio, eMbeddeb VC++	Eclipse, CodeWarrior, PocketStudio, HB++	Eclipse, Netbeans
Multiplate-forme de déploiement	Samsung	iPhone, iPod touch, iPad	Blackberry seulement	Windows Mobile, Windows CE	Palm OS Palm, Windows Mobile	Andoid seulement
Cout d'outils de développement	gratuit	Gratuit	Gratuit	Gratuit	Gratuit et commercial	Gratuit
Magasin en Ligne	Ovi Store	App Store	App World	Windows Market Place	App catalog	Android Market
Open source	Oui	Non	Oui	Non	Non	Oui
Constructeur	Nokia	Apple	RIM	Microsoft	Palm	Google

Tableau 1. Une comparaison entre les systèmes d'exploitation mobile

3.2. Partition du marché entre les SE

D'après la publication d'IDC (*International Data Corporation*), la plateforme Android de Google avoir 84,7% du marché mondial des *Smartphones*, contre 11,7% pour les iPhone d'Apple. Le reste du marché a savoir 3.6% et partagé entre Blackberry et Windows phone.

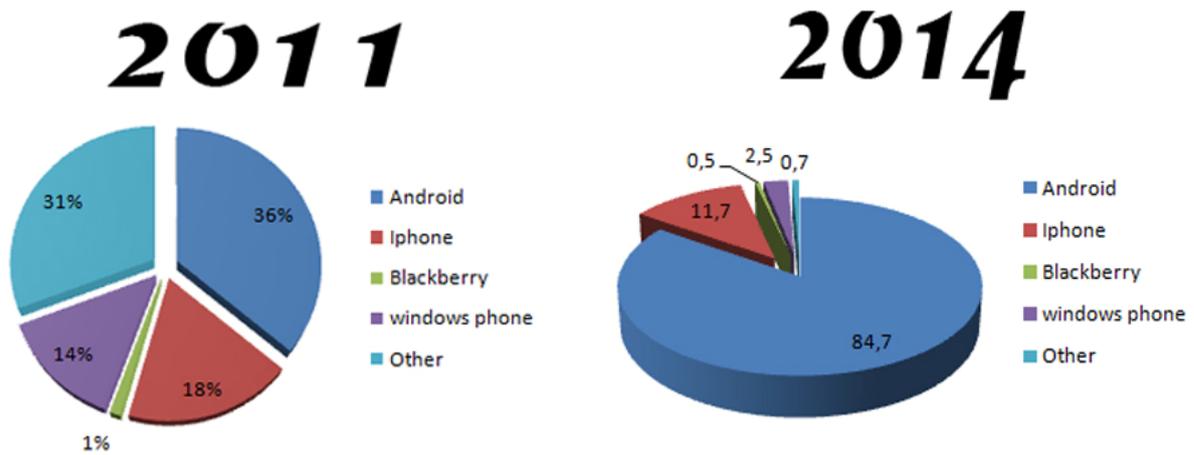
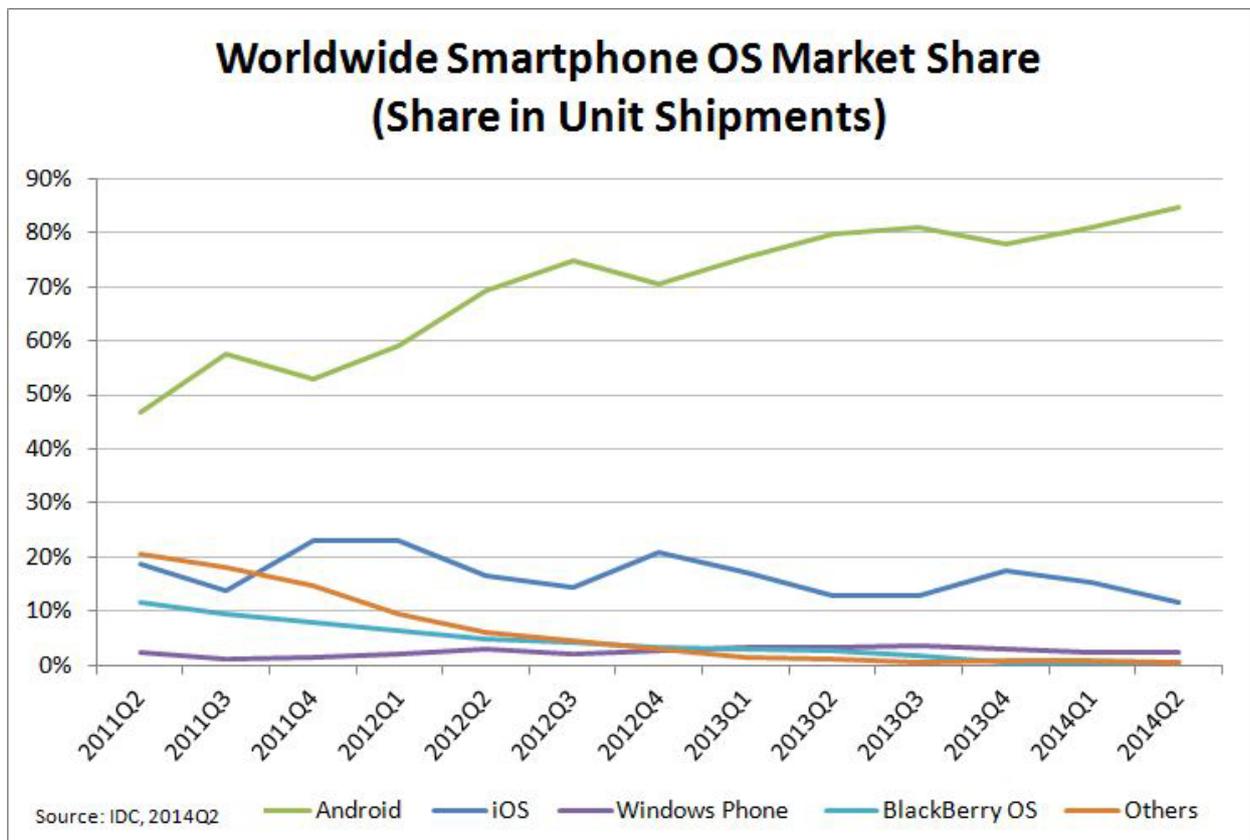


Figure 1 . Les parts de marché des systèmes d'exploitation mobile pour les années 2011 et 2014



Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q2 2014	84.7%	11.7%	2.5%	0.5%	0.7%
Q2 2013	79.6%	13.0%	3.4%	2.8%	1.2%
Q2 2012	69.3%	16.6%	3.1%	4.9%	6.1%
Q2 2011	36.1%	18.3%	1.2%	13.6%	30.8%

Source: IDC, 2014 Q2

4. Le système d'exploitation Android

4.1. Définition et historique

Android est un système d'exploitation Open Source pour Smartphone, PDA (*Personal Digital Assistant*) et terminaux mobiles conçu par Android, une startup rachetée par Google, et annoncé le 15 novembre 2007. Le terme Android fait référence au nom « androïde » qui désigne un robot construit à l'image d'un être humain.

Afin de promouvoir ce système ouvert, Google a su fédérer autour de lui une trentaine de partenaires réunis au sein de l'OHA (*Open Handset Alliance*). En fait, plus de 50 entreprises ont participé à l'OHA, Qualcomm, y compris, Broadcom, HTC, Intel, Samsung, Motorola, Sprint, Texas Instruments et le japonais KDDI transporteurs sans fil et NTT DoCoMo.

Le T-Mobile G1, a été annoncé le 23 Septembre 2008, et a été le premier Smartphone Android OS pour être officiellement introduit sur le marché.

4.2. Les versions de l'Android

La répartition des différentes versions Android est représentée dans le tableau suivant :

Version	Dernière révision	Nom de code	Date de sortie	Caractéristiques (uniquement celles "Open Source" liées à l'AOSP)	Version du noyau
1.0	1.0	Apple pie	fin 2007	Version connue uniquement ou presque des développeurs car c'est la version du SDK distribuée avant la sortie du premier téléphone Android	
1.1	1.1	<i>Bananas split</i>	22 octobre 2008	Ou <i>Beta</i> , version incluse dans le premier téléphone, le HTC G1/Dream	
1.5	mai 2010	<i>Cupcake</i>	30 avril 2009	Nouvelles fonctionnalités et mises à jour de l'interface graphique	Basée sur le noyau Linux 2.6.27
1.6	mai 2010	<i>Donut</i>	15 septembre 2009	Nouvelles fonctionnalités et mises à jour de l'interface graphique	Basée sur le noyau Linux 2.6.29
2.0	2.1, mai 2010	<i>Eclair</i>	26 octobre 2009	Nouvelles fonctionnalités et mises à jour de l'interface graphique	Basée sur le noyau Linux 2.6.29

Version	Dernière révision	Nom de code	Date de sortie	Caractéristiques (uniquement celles "Open Source" liées à l'AOSP)	Version du noyau
2.2.x	2.2.3, 2011	Froyo ¹³	20 mai 2010	Vitesse améliorée, nouvelles fonctionnalités et mises à jour de l'interface graphique	Basée sur le noyau Linux 2.6.32
2.3.x	2.3.7, 2012	Gingerbread ^{13,14}	6 décembre 2010	Dernière version dédiée uniquement aux smartphones. Cette version est parfois utilisée sur de petites tablettes.	Basée sur le noyau Linux 2.6.35
3.x.x	3.2, 2012	Honeycomb ¹⁵	22 février 2011	Réservé aux tablettes tactiles et aux téléviseurs connectés ¹⁶ , cette mise à jour comprend de nombreux changements dans l'interface	
4.0.x	4.0.4, 2012	Ice Cream Sandwich	19 octobre 2011	Cette nouvelle version, fortement inspirée d'Honeycomb, unifiée ¹⁸ pour smartphones, tablettes et Google TV apporte de nombreux changements	Basée sur le noyau Linux 3.0.1
4.1.x	4.1.2, 2012	Jelly Bean ¹⁹	9 juillet 2012	Il ajoute un système de notification améliorée, la reconnaissance vocale sans connexion internet , et le « Project Butter » qui augmente la fluidité d'Android;	Basée sur le noyau Linux 3.0.31

Version	Dernière révision	Nom de code	Date de sortie	Caractéristiques (uniquement celles "Open Source" liées à l'AOSP)	Version du noyau
4.2.x	4.2.2, février 2013	Jelly Bean	13 novembre 2012	Nouvelle interface de l'appareil photo, d'un système multi-compte uniquement sur tablette, et de <i>Type Gesture</i> permettant d'écrire avec le clavier rien qu'en glissant le doigt.	Basée sur le noyau Linux 3.4.0
4.3.x	4.3.1, 24 juillet 2013	Jelly Bean	24 juillet 2013	Support du Bluetooth SMART (en) basse consommation et ajout de la norme AVRCP (en) 1.3, gestion multi-utilisateur plus poussée, support d' OpenGL ES 3.0 , nouvelle interface de l'appareil photo, mises à jour de sécurité et SlimPort (en) .	Basée sur le noyau Linux 3.4.0
4.4.x	4.4.4, 19 juin 2014	KitKat	31 octobre 2013	Consommation en ressource moins élevée nécessitant moins de RAM, nouvelles icônes plus soignées, la barre du bas et celle de statut deviennent transparentes sur certains menus et changent de couleur en fonction du contenu affiché.	Basée sur le noyau Linux 3.4.0

Tableau 2. Les versions de l'Android

C'est les dernières statistiques qui datent du janvier 2014 concernant la répartition des différentes versions Android.

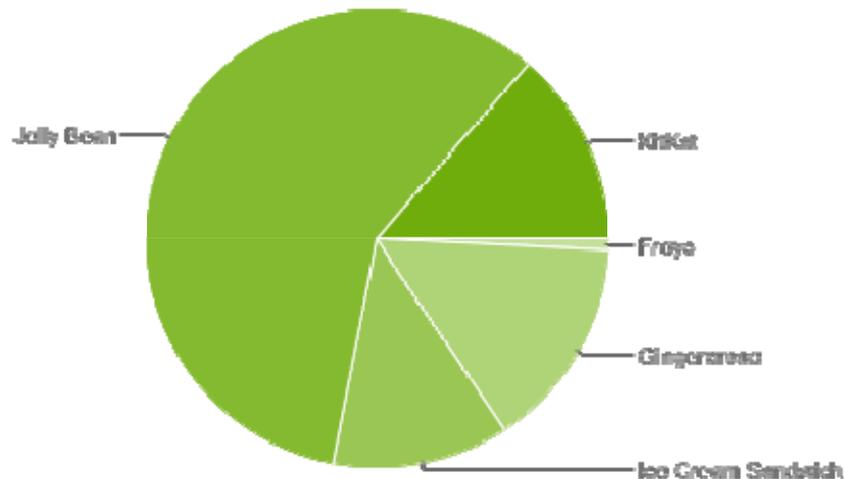


Figure 2. La part de chaque version d'Android

La figure 2 est basée sur le nombre d'appareils Android qui ont accédé au Google Play, dans un délai de 14 jours se terminant à la date de la collecte des données ci-dessous.

Nous avons la grande proportion de terminaux qui sont équipés par la version Android Jelly Bean qui est toujours *leader* avec 54%, on peut s'apercevoir aussi que la version Android Kitkat prend une bonne proportion du marché avec 20,9 %. Les autres versions partagent la part restante.

Version	Nom de code	Date de sortie	API level	%
2.2	Froyo	20 mai 2010	8	0,7 %
2.3.x	Gingerbread	6 décembre 2010	9-10	13,6 %
4.0.x	Ice Cream Sandwich	19 octobre 2011	14-15	10,6 %
4.1.x	Jelly Bean	9 juillet 2012	16	26,5 %
4.2.x	Jelly Bean	13 novembre 2012	17	19,8 %
4.3.x	Jelly Bean	24 juillet 2013	18	7,9 %
4.4.x	KitKat	31 octobre 2013	19	20,9 %

4.3. Architecture logicielle

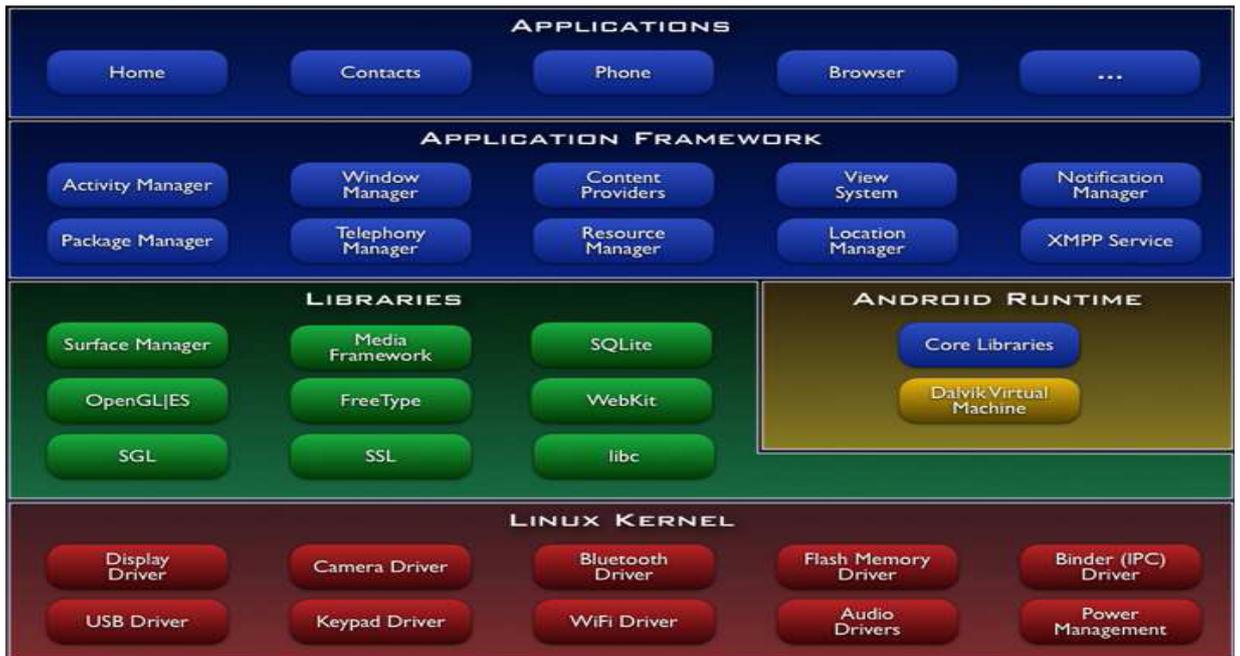


Figure 3. Architecture logicielle de l'Android

La Figure 3 illustre l'architecture logicielle de l'Android en lui composé en plusieurs couches.

La plate-forme Android est composée de différentes couches:

- **Linux Kernel**

L'Androïde se fonde sur la version 2,6 de Linux pour des services système de noyaux tels que le système de sécurité, la gestion de mémoire, la gestion de processus industriel, le réseau et le modèle de pilote. Le noyau agit sur la couche d'abstraction entre le matériel et le logiciel.

- **Librairies**

Au-dessus du noyau kernel, proprement dit, se loge un ensemble de bibliothèques natives constituant les couches bases du système. Ces bibliothèques sont écrites en C/C++ et utilisées par les différentes composantes du système Android.

- **Android Runtime**

L'Android inclut un ensemble de bibliothèques de base qui fournit la plupart des fonctionnalités disponibles dans les bibliothèques de base du langage de programmation Java. Les applications s'exécutent chacun dans son propre processus.

Une application sous Android s'exécute dans son propre processus, avec son propre instance de machine virtuelle Dalvik. Ce dernier exécute des fichiers avec l'extension ".dex" qui est

optimisé pour une empreinte mémoire minimale. Le VM Dalvik s'appuie sur le noyau Linux pour les fonctionnalités de base telles que la gestion de la mémoire de bas niveau.

- **Application Framework**

Les développeurs ont un accès complet à l'API. L'architecture d'application est conçue pour rendre la réutilisation des composants plus simple. En fait, chaque application peut publier ses capacités et d'autres applications peuvent alors faire usage de ces capacités. Toutes les applications sous-jacentes sont un ensemble des services et des systèmes.

- **Applications**

Android sera livré avec un ensemble d'applications de base, dont un client de messagerie, le programme de SMS, calendrier, cartes, navigateur, Contacts, et d'autres. Toutes les applications sont écrites en utilisant le langage de programmation Java.

4.4. Kit de développement

Exploiter une nouvelle plate-forme n'est jamais été une chose aisée. C'est pourquoi Google fournit, en plus du système d'exploitation, un kit de développement (*Software Development Toolkit* ou SDK). Ce SDK est un ensemble d'outils qui permet aux développeurs et aux entreprises de créer des applications. Il est disponible gratuitement sur le site de Google.

Le SDK Android est composé de plusieurs éléments afin d'aider les développeurs à créer et à maintenir des applications :

- Des outils ;
- Des exemples de code ;
- De la documentation ;
- Des API (interfaces de programme d'application).

Les outils

Le SDK est livré avec un certain nombre d'outils couvrant différents aspects du cycle de développement d'une application Android. Le kit de développement propose une boîte à outils complète pour les tâches de compilation, de débogage, de génération de code AIDL et de la signature de l'application, etc.

L'émulateur Android : c'est un téléphone virtuel qui permet de tester les applications qui sont entrain de se développer. Il est lancé par la commande "emulator ". Celle-ci prend en paramètre l'image AVD (*Android Virtual Device*) qui sera montée en mémoire. Il a des

limitations par exemple : il n'est pas capable de supporter le Bluetooth ainsi qu'il ne permet pas le teste des applications de réalité augmentée.

Les API

Android offre plusieurs API (*Application Program Interface*) tel que :

- **Google Maps** : intègre et contrôle l'affichage d'une carte dans une interface graphique de l'application.
- **Géolocalisation** : permet d'accéder au service de localisation du système, de choisir le fournisseur en fonction des critères et de préciser la position actuelle du téléphone (latitude, longitude, vitesse, etc.).

Les exemples de code

Le kit de développement est accompagné d'un certain nombre d'exemples illustrant les possibilités du SDK Android. Parmi ces exemples, on peut citer : un jeu du serpent et le projet qui couvre l'utilisation de plusieurs exemples de l'API Android comme les alarmes, les notifications et les menus.

La documentation

La documentation du SDK Android est scindée en deux parties bien distinctes :

- Le guide du développeur qui est disponible en HTML (*Hypertext Markup Language*) dans le répertoire du SDK qu'on vient d'installer ;
- La documentation des API au format javadoc est également située dans le répertoire docs et accessible grâce au chemin débutant du répertoire d'installation.

5. La Géolocalisation

Parmi les fonctionnalités les plus appréciées sur les plates-formes mobiles modernes, la géolocalisation permet de réaliser des applications innovantes. Grâce à Google Maps notamment, elle est au cœur d'Android.

Le service de géolocalisation récupère les coordonnées de l'utilisateur et les envoie pour la réalisation un service informatique. A chaque fois, il demande l'autorisation de l'utilisateur avant la réalisation de cette opération. Seules les informations concernant la latitude et la longitude sont envoyées.

La localisation du mobile se fait selon plusieurs technologies comme :

- **GPS** : il s'effectue par la réception de signaux provenant de plusieurs satellites qui se trouvent en orbite. Le téléphone mobile équipé d'un GPS permettra de transmettre sa position via un réseau SMS, GPRS, Edge ou UMTS.
- **Internet** : La précision de la localisation par adresse IP sur le réseau internet se situera au niveau d'un pays, d'une ville ou d'un quartier selon l'opérateur (national ou local). Cependant, au sein d'un réseau ADSL d'un même opérateur, la géolocalisation peut être très précise (adresse ou bâtiment par exemple) si les lieux des connexions sont enregistrés dans une base de données.
- **Wifi** : La localisation est similaire au cas du réseau GSM ou IP, par les cellules émettrices, avec une précision inférieure à 100 mètres. Une triangulation entre plusieurs antennes Wifi peut donner la position avec une précision d'environ 5m par l'analyse de la puissance du signal radio reçu de l'appareil.
- **GSM** : il est basé sur le code unique de la carte SIM. La connexion au réseau est autorisée après une identification à une cellule composant le réseau GSM. La précision dépend de l'étendue de la cellule, de 250 mètres en zone urbaine à 10 km en zone rurale.

6. La sécurisation des communications

Pour assurer une communication sécurisée entre le client et le serveur, il est obligatoire d'utiliser des protocoles de cryptage. On trouve aussi des protocoles de communication qui garantissent la sécurisation du canal de transport des données comme HTTPS (*HyperText Transfer Protocole Secured*).

6.1. La définition du protocole HTTPS

"*HyperText Transfer Protocole Secured*" (HTTPS) est un protocole de transfert hypertexte sécurisé qui a été développé par Netscape.

Il correspond à une version sécurisée du http (*HyperText Transfer Protocole*). Le HTTPS répond aux différents problèmes de confidentialité que le protocole http a connus.

L'idée principale de HTTPS est de créer un canal sécurisé sur un réseau non sécurisé et d'assurer une protection raisonnable contre les oreilles indiscrettes à condition que les suites de chiffrement adéquates soient utilisées et que le certificat de serveur soit vérifié et approuvé.

6.2. Les objectifs de sécurité assurés

Le protocole HTTPS fournit les objectifs de sécurité suivants :

- L'authentification en permettant l'assurance de l'identité du programme, de la personne ou de l'entreprise avec laquelle on communique.
- La confidentialité des données échangées : Il est impossible d'espionner les informations échangées.
- L'intégrité des données échangées : Il est impossible de truquer les informations échangées.
- La spontanéité : la connexion de client avec le serveur est transparente.

Conclusion

Durant ce chapitre, nous avons présenté les technologies utilisées et leurs fonctionnalités.

Android et iOS "dominent le monde" - La domination d'Android sur le marché mondial des smartphones ne se dément pas. Ainsi sur les 287,8 millions de terminaux livrés au 1er trimestre 2014, 81,1% tournaient sous Android selon IDC. Et sur le 2e trimestre, le rapport de force s'est encore accentué au profit des deux plateformes dominantes : 96% des terminaux livrés dans le monde, soit 301,3 millions d'unités, tournaient sous Android et iOS, dont 84,7% pour le seul Android.

Malgré une progression de ses ventes, Apple abandonne des parts de marché. Cependant, avec un positionnement sur le haut de gamme, l'iPhone reste toujours très rentable pour la firme. D'après IDC, la part de marché d'iOS sur le haut de gamme est considérable, de l'ordre de 84,6%, contre seulement 19,82% pour Android. L'OS Google domine en revanche sur l'entrée de gamme, laissée de côté par Apple, puisque 58,6% des androphones livrés sur la période appartiennent à ce segment. Du côté de Windows Phone, la ventilation par catégories est relativement similaire.

Chapitre 3

Analyse et Conception

Pour une meilleure compréhension du travail effectué, nous présentons dans ce chapitre l'étape d'analyse et de conception. En effet, pour réaliser une bonne conception de l'application, il faut faire une étude approfondie des exigences du marché du travail. Dans ce chapitre, nous commencerons par une étude préliminaire qui consiste à repérer les besoins fonctionnels et non fonctionnels. Par suite, nous avons élaborés une modélisation claire de ce qui a été établi au cours de cette étude. Aussi, ce chapitre sera dédié pour la conception architecturale de notre application.

1. Identification et analyse des besoins

1.1. Identification des besoins

Les besoins de l'application se divisent en des besoins fonctionnels et non fonctionnels.

1.1.1. Les besoins fonctionnels

Un acteur joue le rôle d'une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. L'acteur représenté dans notre application pour l'utilisateur qui peut être soit un administrateur ou un assuré.

Administrateur :

Cette application permet à l'administrateur de :

- Ajouter des nouveaux médecins (informations et localisation).
- Modifier les coordonnées GPS d'un médecin où les données relatives à ces informations.
- Supprimer des médecins.

L'assuré :

Les fonctions offertes pour un assuré sont :

- Consulter de liste des médecins conventionnés,
- Géolocalisation des médecins et consultation de leur coordonnées.
- Possibilité de localiser les médecins proximités de l'emplacement de l'assuré.

1.1.2. Les besoins non fonctionnels

Les besoins non fonctionnels décrivent les objectifs liés aux performances du système et aux contraintes de son environnement. Ses exigences techniques sont souvent exprimées sous forme d'objectifs spécifiques que doit atteindre le système.

Les besoins non fonctionnels de cette application sont les suivants :

- Assurer une cohérence et une crédibilité aux informations stockées dans la base de données.
- La confidentialité des données personnelles (Medecins).
- L'ergonomie d'interface de l'application mobile.

1.2. Analyse des besoins

Un cas d'utilisation est un ensemble de scénarios. Les diagrammes de cas d'utilisation présentent des acteurs qui interagissent avec des cas d'utilisation. Ces diagrammes sont faciles à interpréter même si nous ne connaissons pas la notion. Pour traiter ces diagrammes, nous avons pu les classer selon la nature de l'utilisateur.

1.2.1. Cas d'utilisation relatif à l'administrateur

Nous pouvons trouver plusieurs cas d'utilisation de l'application pour l'administrateur :

- Gérer la localisation des médecins.
- Gérer les informations des médecins.

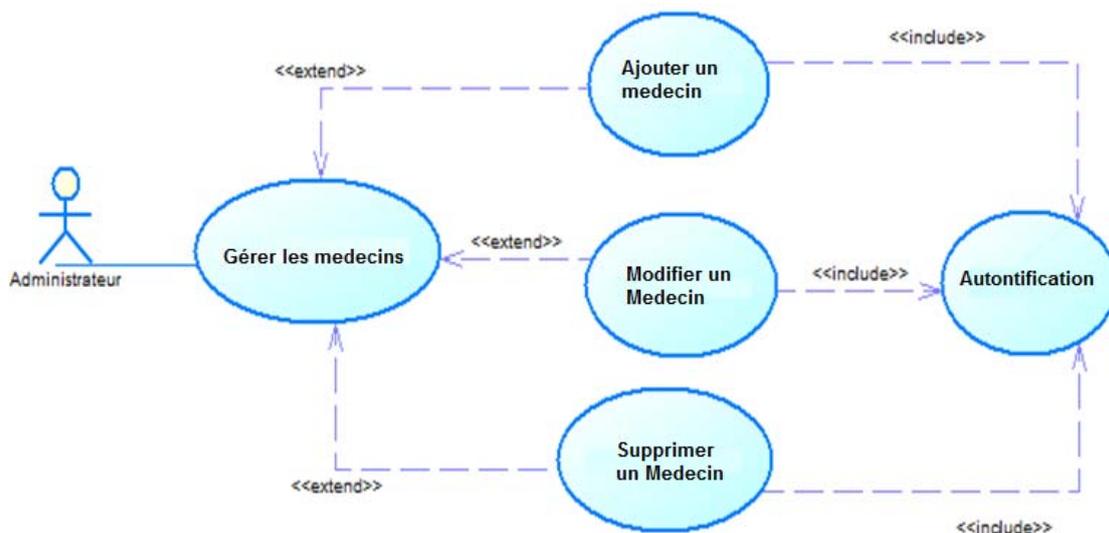
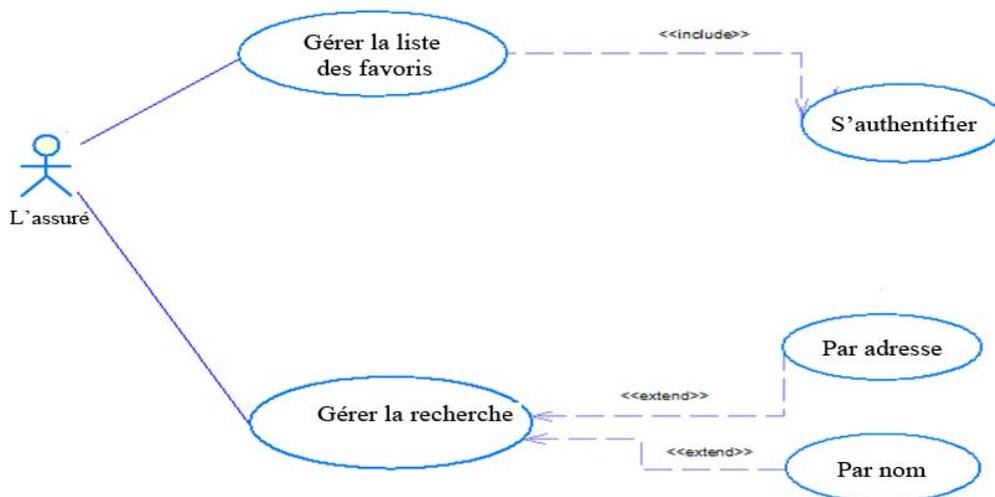


Figure 4. Diagramme de cas d'utilisation relatif à l'administrateur

1.2.2. Cas d'utilisation relatif à l'assuré

L'assurée peut :

- Consulter la liste des médecins : l'application permet de visualiser la localisation des médecins conventionnés (généralistes ou spécialistes) avec la CNAS les plus proches de l'assuré. En effet, l'assuré peut visualiser sa position récupérée à travers le GPS et l'afficher sur la carte.



• Figure 5. Diagramme de cas d'utilisation relatif à l'assuré

1.3. La description détaillée des cas d'utilisation

1.3.1. Scénario du cas d'utilisation "consulter les médecins a proximité"

Dans cette section, nous détaillons les principaux cas d'utilisation de l'assuré et nous traitons son scénario.

La Figure 6, présente le cas d'utilisation « consulter les médecins a proximité » de l'acteur assuré.

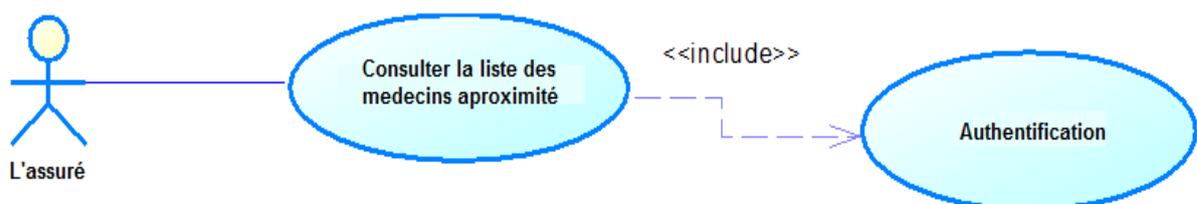


Figure 6. Cas d'utilisation "consulter les médecins a proximité"

Cas d'utilisation " consulter les médecins a proximité "	
Pré-condition	L'assuré peut effectuer une consultation d'un médecin
Scénario	Les médecins les plus proches de l'assuré seront afficher sur une map.

Tableau 3. *Cas d'utilisation « consulter les médecins a proximité»*

2. Conception détaillée

Ayant dégagé les différents acteurs et énuméré les objets nécessaires lors de l'étape de l'analyse et après avoir décrit la conception générale de l'application, on doit désormais détailler la conception de l'application en décortiquant les différentes classes d'une telle application et en définissant en détails notre base de données.

Pour cela, le langage de modélisation UML a été adopté afin de réaliser la modélisation des différents modules ainsi que leurs interactions et les différentes vues statiques et dynamiques du système.

2.1. Les diagrammes des séquences

Pour notre application, nous allons élaborer les principaux diagrammes de séquence pour déterminer la dynamique du système. En fait, ces diagrammes permettent de représenter des collaborations entre objets selon un point de vue temporel, on y met l'accent sur la technologie des envoies des messages. Nous allons présenter le diagramme de séquence pour le cas d'utilisation : « consultation des médecins a proximité ».

2.1.1. Diagramme de séquence du scénario recherche médecins a proximité

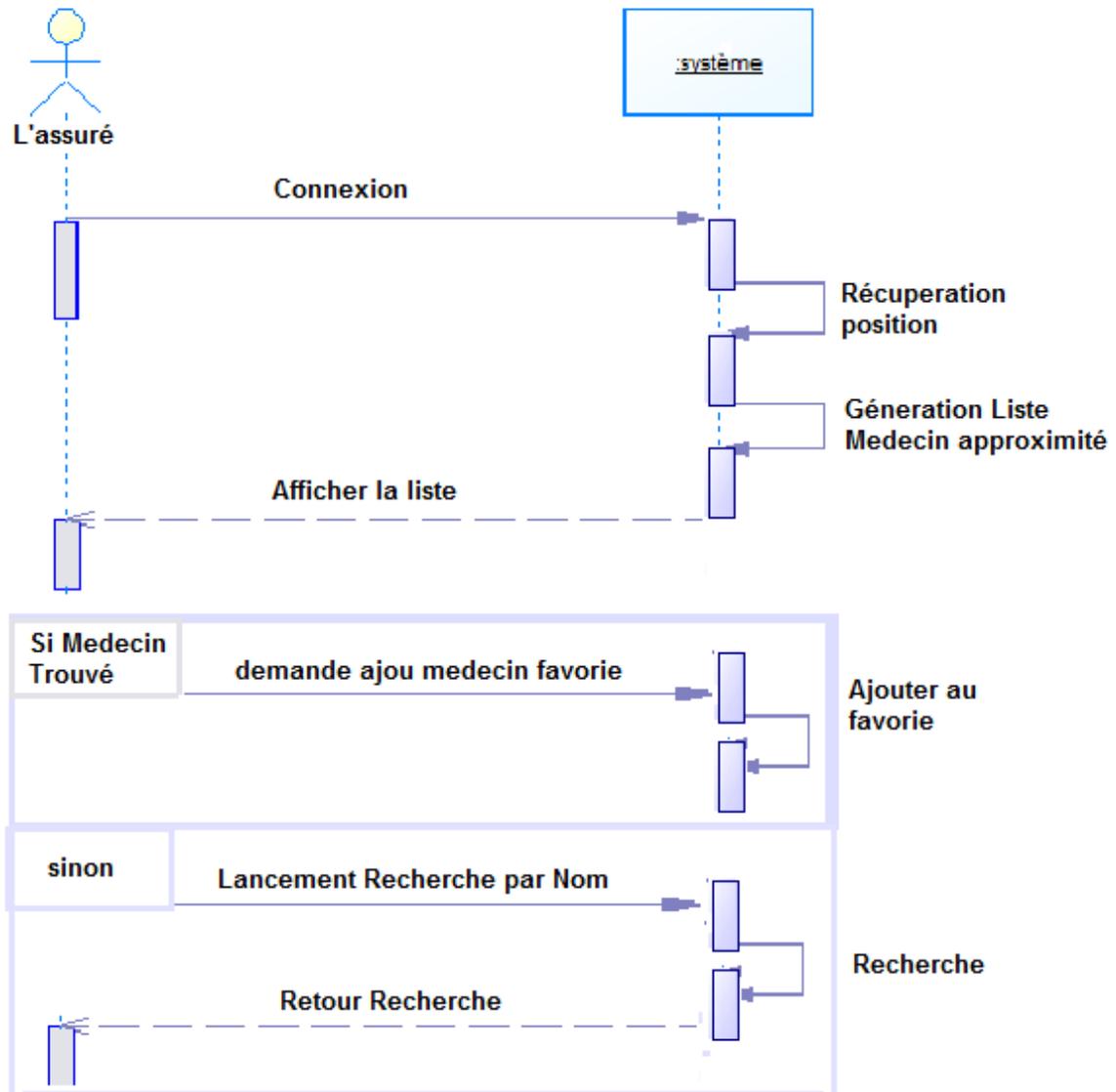


Figure 7. Diagramme de séquence du scénario recherche médecins a proximité

3. Diagramme de classe

Le diagramme de classe est une description statique du système focalisé sur le concept de classe et d'association.

Une classe représente un ensemble d'objets qui possèdent des propriétés similaires et des comportements communs décrivant en terme d'attributs et d'opérations. Une association consiste à présenter les médecins entre les instances de classe.

Durant cette section, nous allons présenter les diagrammes de classes entités à notre application.

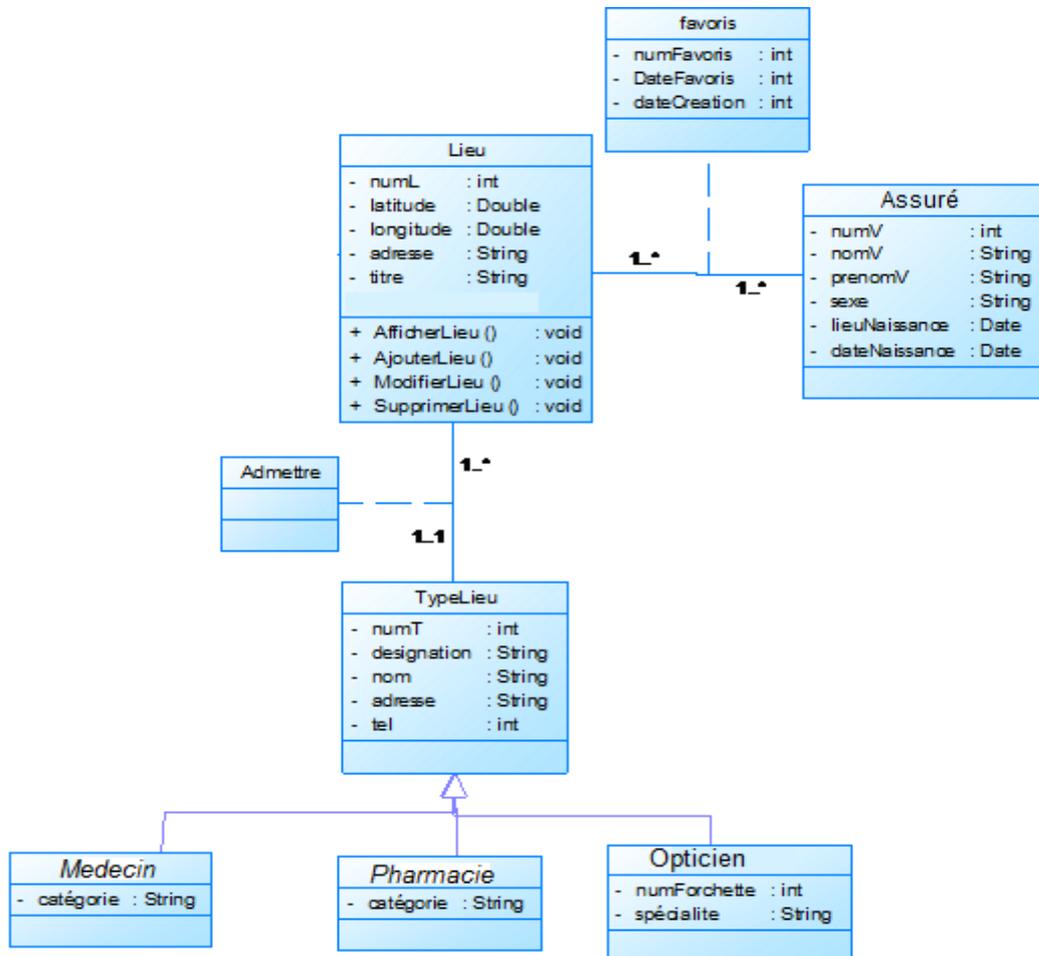


Figure 8. Diagramme de classe

Ce diagramme comporte un ensemble de classes définissant les entités du monde réel du système d'informations d'une application de localisation des médecins conventionnés. Ce diagramme est composé des classes et de classes d'associations.

Conclusion

A travers ce chapitre, nous avons présenté la conception de l'application. Nous avons présenté une vue globale. Puis, nous avons présenté la conception détaillée de l'application à travers les diagrammes de classe et de séquence.

Chapitre 4

Réalisation

Après avoir décortiqué la partie conception, nous présenterons la partie réalisation de notre application. Nous procéderons dans ce chapitre à une spécification de l'environnement logiciel de développement, et nous terminons ce chapitre par présenter et décrire quelques interfaces de notre application.

1. Environnement de développement

Pour créer une application Android, on doit installer les outils suivants :

- ✓ Android Software Development Kit (SDK),
- ✓ Eclipse Version: 4.2.2 (Juno) ,
- ✓ Le compilateur Java (JDK 7).

1.1. Installation des outils

- Nous avons installé SDK 7.
- Nous avons dé zippé le contenu du fichier Android SDK sous le chemin C:\android-sdk\.
- Nous avons dé zippé la version Eclipse sous le chemin C:\eclipse\.

1.2. Paramétrage d'Eclipse et installation du plugin ADT

- Nous lançons Eclipse puis dans le menu « Help » on clique sur « Install new software ». Une fenêtre s'ouvre alors et nous configurons les dépôts Google Android pour avoir le plugin AVD intégré à notre IDE.
- Nous cliquons sur le bouton « Add » en haut à droite de la fenêtre
- Nous renseignons les champs suivants :
 - Name : Android (ou autre).
 - Location : <http://dl-ssl.google.com/android/eclipse/>.

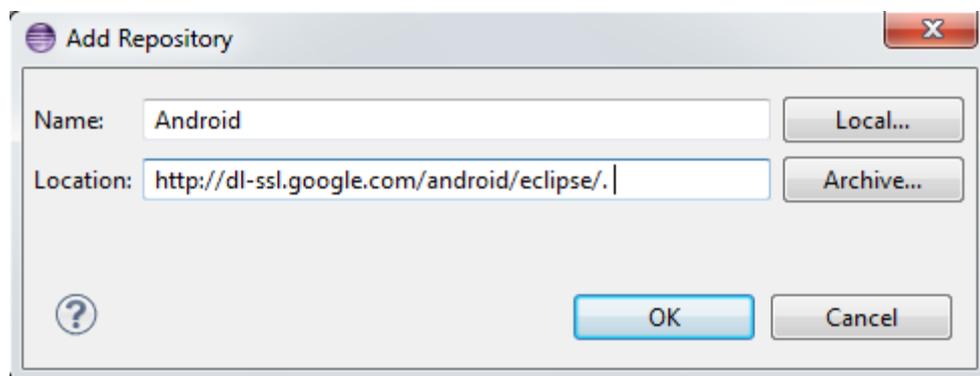


Figure 9. URL Plugin

- Après rafraîchissement, Eclipse affiche les plugins disponibles. Nous sélectionnons le « Android Development Tools ».
- Nous validons les étapes suivantes et nous redémarrons Eclipse.
- Maintenant que Eclipse reconnaît les plugins, sous Windows-> Préférences->Android nous indiquons le chemin d'Android SDK : C:\android-sdk\ dans notre cas :

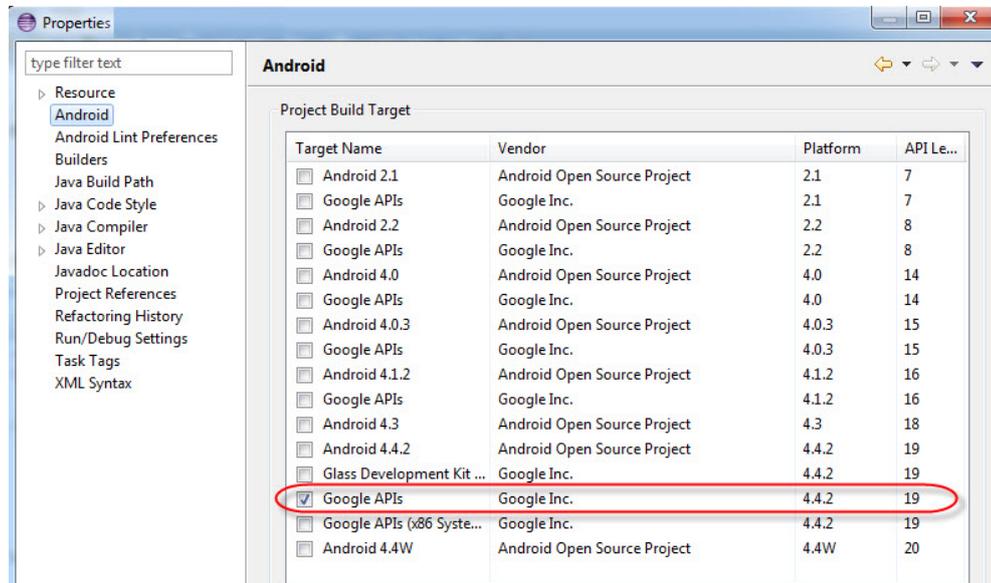


Figure 10. Liste des targets

1.3. Téléchargement des différents composants d'Android SDK

- Sous C:\android-sdk\ nous lançons SDK Setup.exe.
- Sous « Available Packages », nous trouvons la liste des packages et Archives du Android SDK.
- Nous choisissons les différents packages et nous validons.

Une fois ceci terminé, nous nous retrouvons avec les différentes versions d'Android installées sur le système comme le montre la capture suivante :

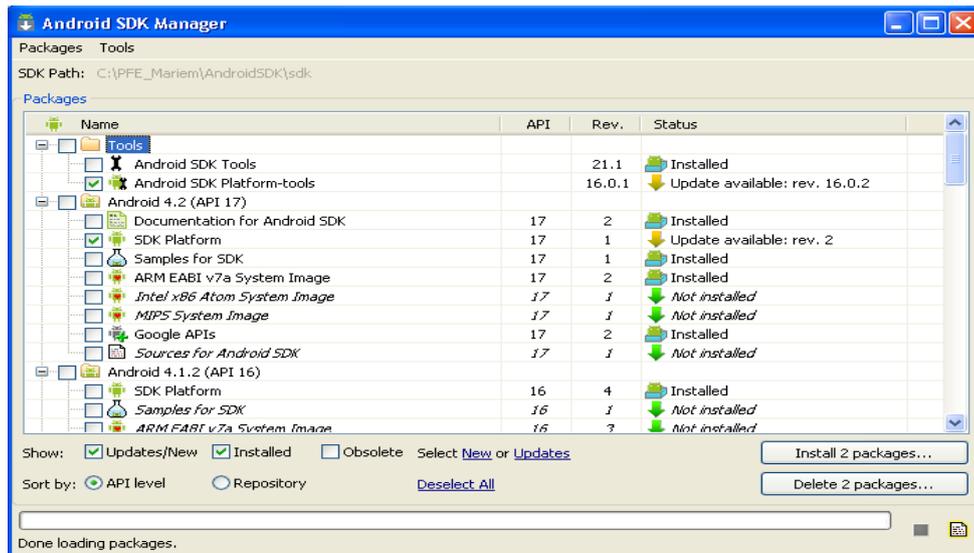


Figure 11. Android SDK AVD Manager

1.4. Création d'un AVD

Afin de tester notre application, nous allons utiliser l'émulateur Android. Il faudra donc créer un Android Virtual Device (AVD). Un AVD décrit les paramètres systèmes et les composants de notre émulateur.

Pour créer un AVD:

- Nous lançons Eclipse,
- Nous allons sous « Window > Android SDK and AVD Manager »,
- Nous sélectionnons « Virtual Device » dans le panneau à gauche,
- Nous cliquons sur « New ». La boîte de dialogue « Create New AVD » apparaîtra,
- Nous tapons le nom de notre AVD, « Medecin » par exemple,
- Nous choisissons la cible (the target). La cible est la version de la plateforme Android SDK que nous avons téléchargé,
- Nous ignorons les autres champs pour le moment et nous cliquons sur « Create AVD ».

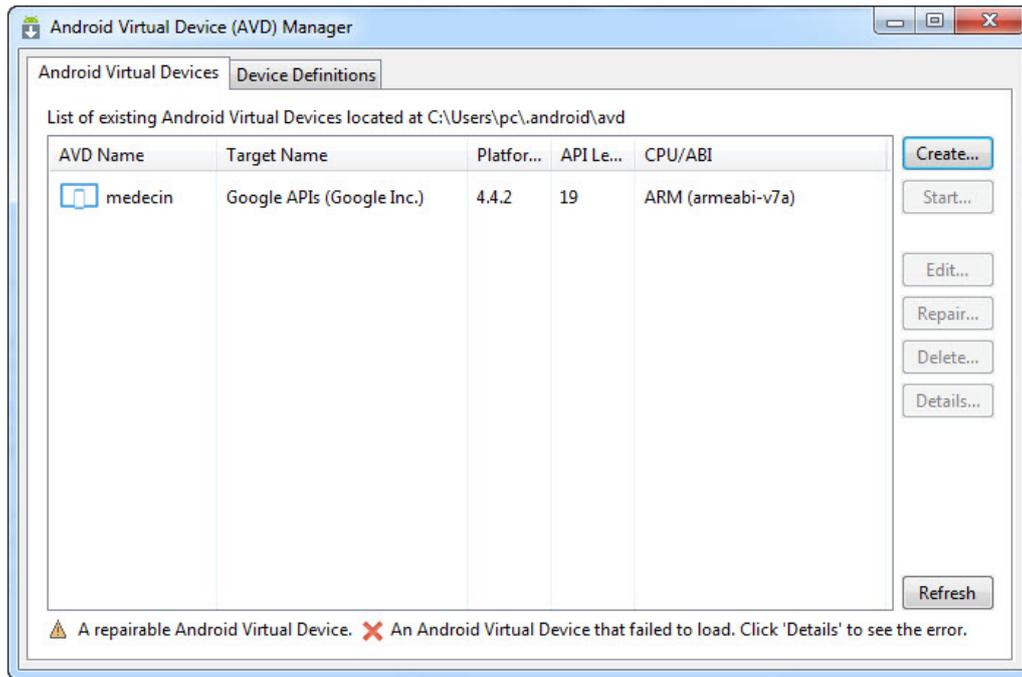


Figure 12. Liste des AVD créés

1.5. Fichier source d'un projet

On a deux types de fichier source:

1.5.1. Java

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de réaliser des logiciels compatibles avec de nombreux systèmes d'exploitation. Java offre aussi la possibilité de réaliser des programmes pour téléphones mobiles et assistants personnels PDA. Il possède plusieurs caractéristiques qui le rendent parmi les meilleurs outils de développement des logiciels tels que : simple, distribué, interprété, indépendant de l'architecture, portable, robuste, sûr, dynamique, multithreadé.

1.5.2. XML

Le XML (*eXtensible Markup Language*) est dérivé du SGML (*Standard Generalized Markup Language*), tout comme le HTML (*Hypertext Markup Language*) [9]. Il est couramment utilisé comme format générique pour l'échange de données entre serveurs et applications, dans les processus de communication entre couches applicatives ou pour le stockage de données complexes. Le XML a été largement adopté dans tous les secteurs d'industrie et par la majorité des langages de programmation.

1.6.Intégration de Google Map

Pour intégrer Google Map sous une application Android, on a besoin de :

➤ **MD5**

Pour créer votre md5, il faut trouver où se situe votre fichier debug.keystore.

- Si vous utilisez **Eclipse**, vous pouvez obtenir le chemin de la façon suivante : « Windows > Prefs > Android > Build ».

- Une fois le chemin connu, il faut lancer une console Windows (Démarrer -> Exécuter puis taper cmd). Puis dans la console, taper la commande suivante :

keytool -list -alias androiddebugkey -keystore<chemin_vers_le_fichier_debug>.keystore -b storepass android -keypass android.

Pour pouvoir générer votre clé, il vous suffit de saisir le md5 obtenu comme le montre la figure suivante :



Figure 13. Récupération de la clé MD5 pour Google MAP

➤ **Key Api**

Après l'insertion de MD5, on doit avoir un compte Google.

Et voici notre clé :

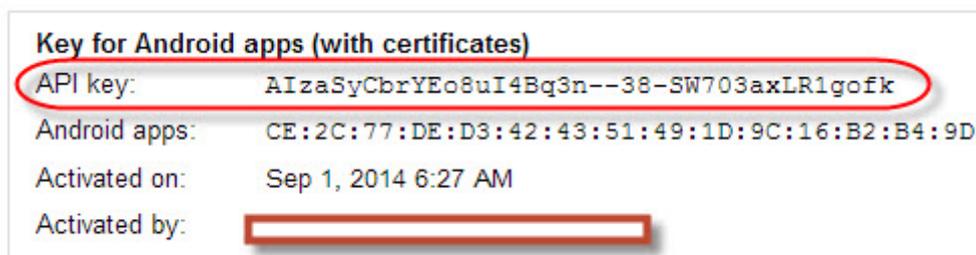


Figure 14. Obtention de la clé API

2. Persistance de donnée

Modèle logique du système d'information

Le modèle logique est une représentation du système d'information de l'application indépendamment des supports physique d'implantation des objets persistants.

A fin de déduire ce modèle de système d'information, nous choisissons le modèle relationnel comme un modèle logique.

Le passage du modèle objet vers le modèle relationnel a fournit le schéma relationnel suivant.

- L'assurée (numV, nom, Prénom, sexe, lieuNaissance, dateNaissance),
- Lieu (numL, latitude, longitude, adresse, titre, #numFavoris),
- Favoris (#numV, #numL, numFavoris, DateFavoris, dateCreation),
- TypeLieu (numT, désignation, nom, adresse, tel).

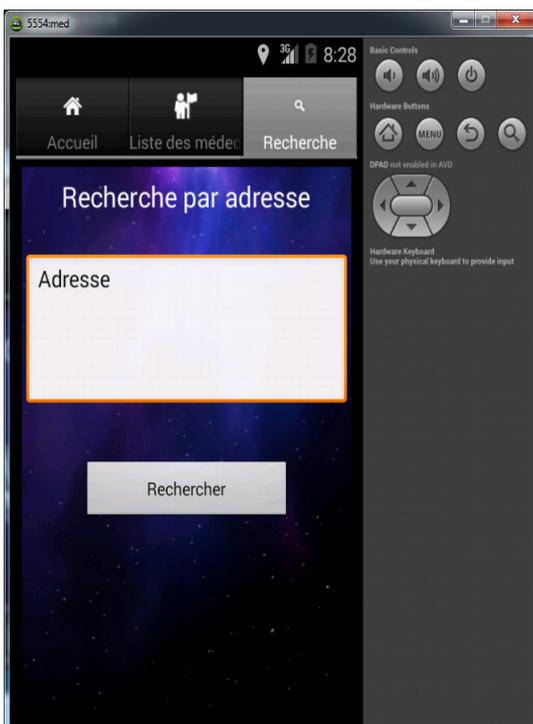
3. Résultats obtenus

Notre application est développée de façon incrémentale et itérative. Comme première itération, nous avons commencé par la réalisation de cas d'utilisation "consultation des médecins" puis le cas d'utilisation "s'inscrire" suivie des autres. Dans cette partie, on va présenter quelques résultats obtenus.



Figure 15. Interface d'accueil de l'application

Interface de la recherche



Résultat de la recherche

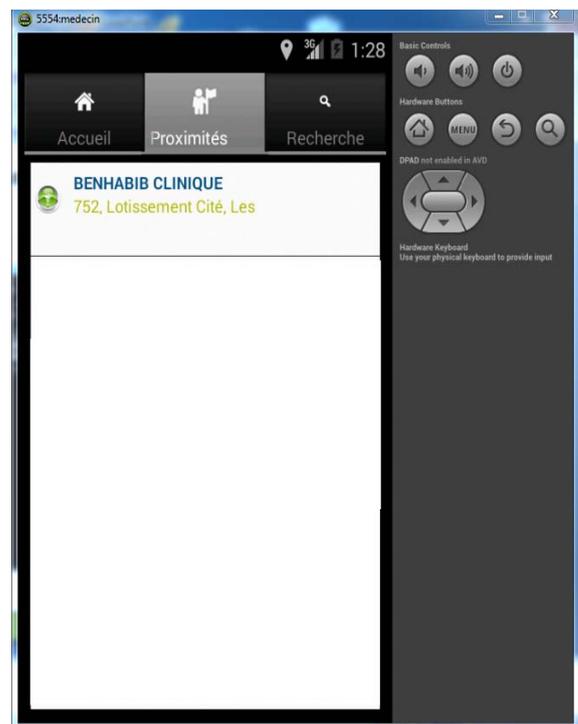


Figure 16. Recherche dans l'application

Pour cette interface, l'utilisateur peut rechercher un médecin grâce à son adresse
En cas de succès, l'interface du résultat s'affiche.

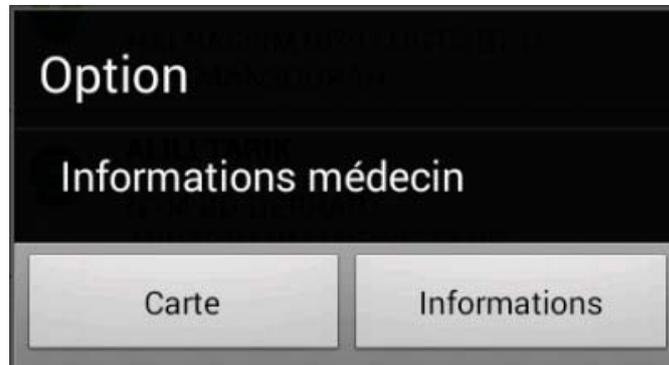
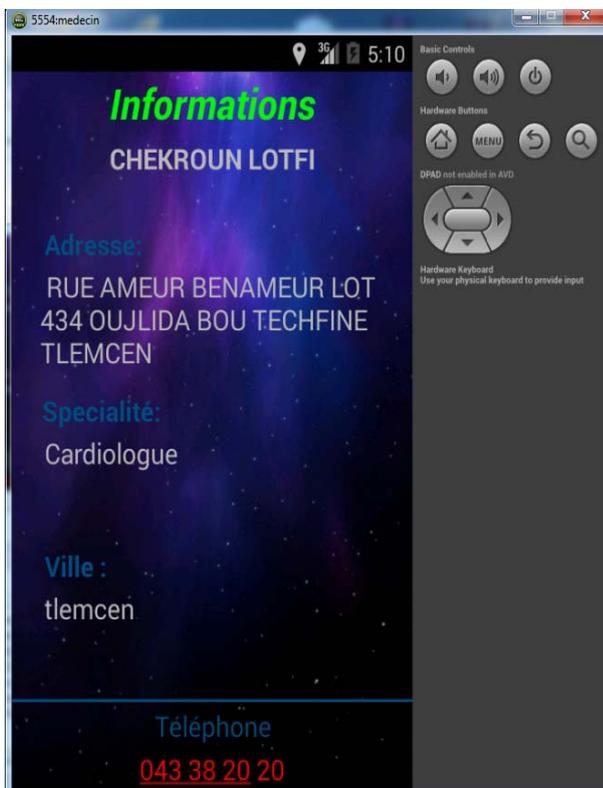
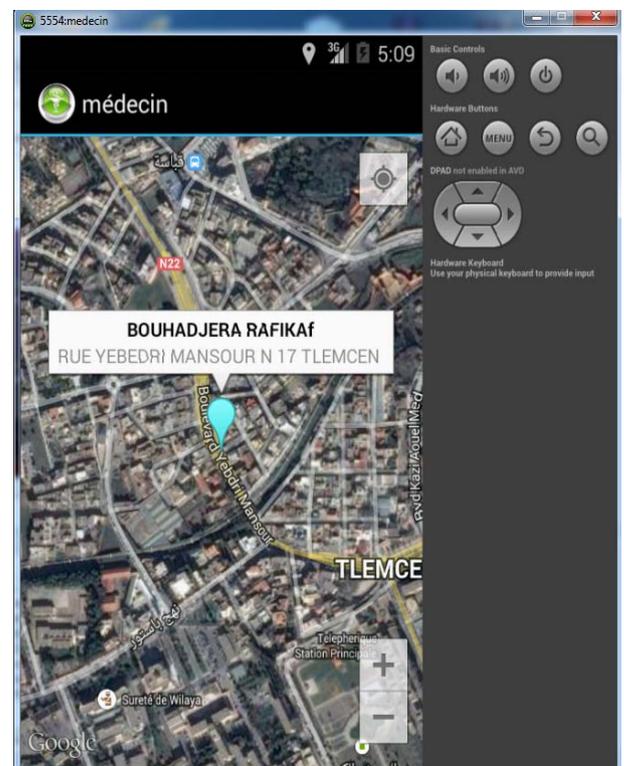


Figure 17. Interface de choix

En cliquant sur le médecin sélectionné de l'interface du résultat de la recherche on obtient l'interface du choix qui permet d'affiché soit les informations du médecin, soit l'afficher sur une carte géographique.



Informations sur un médecin



Géo localisation d'un médecin

Figure 18. Informations et géo localisation d'un médecin

La figure 18 illustre les interfaces représentent les principales activités de l'application comme la consultation des endroits des médecins et leurs informations.

Liste des Médecins Conventionnés

Carte des Médecins conventionnés

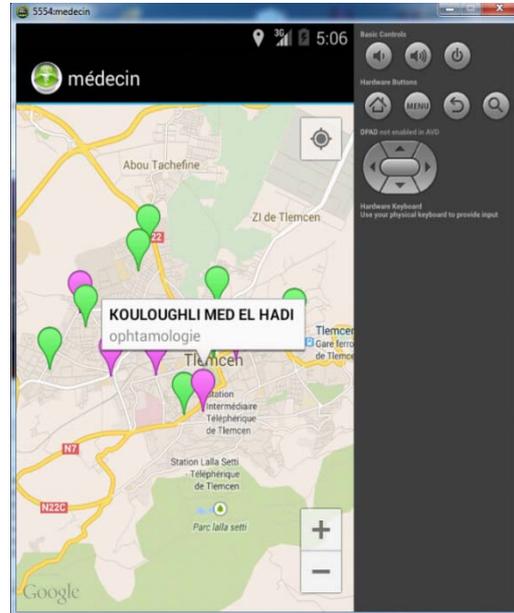
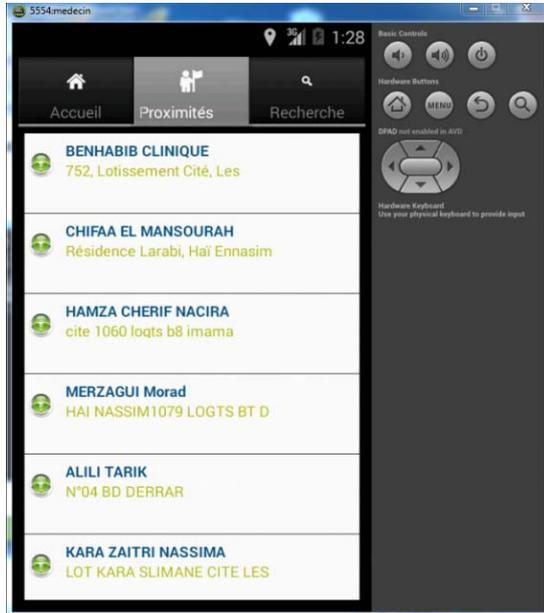


Figure 19. Liste des médecins et leur Géo localisation

Conclusion

Au bout de notre cursus en licence informatique, nous avons été chargés de réaliser un projet de fin d'année. Notre travail s'est basé sur le développement d'un programme sur les technologies mobiles (Smartphone). Ceci nous a amené à découvrir une nouvelle plateforme de développement et à enrichir notre savoir et notre expérience.

Notre projet a permis de concevoir et de réaliser une application mobile servant de guide pour trouver les médecins conventionnés généralistes et spécialistes..

En premier lieu, nous avons commencé par une présentation générale du problème à aborder en allant de l'exposition du cahier de charge, à l'étape de l'analyse et de spécification pour finir par la fixation des besoins de l'application en montrant les différents cas d'utilisation. Ensuite, nous avons entamé la conception de l'application en donnant un aperçu sur les principaux diagrammes UML. Dans l'étape de réalisation, nous avons sélectionné les technologies les plus adaptées à notre choix technique, pour finir par une illustration des différentes interfaces graphiques de notre application qui pourra servir comme guide pour les assurés

Dans le cadre de ce projet, nous avons eu l'opportunité de nous familiariser avec des technologies récentes telles qu'Android. Un autre aspect de ce projet était de pouvoir travailler sur les logiciels libres tels que l'IDE Eclipse.

Bibliographie

- [1]. www.eclipse.org
- [2]. www.latribune-online.com
- [3]. www.lemag.pro.bouyguestelecom.fr
- [4]. www.lemag.pro.bouyguestelecom.fr
- [5]. Fr.wikipedia.org
- [6]. Fr.wikipedia.org
- [7]. Fr.wikipedia.org/wiki/Android
- [8]. Fr.wikipedia.org
- [9]. www.linternaute.com
- [10]. <http://www.zdnet.fr/actualites/chiffres-cles-les-os-pour-smartphones-39790245.htm>