

Projet Fin d'études

Présenté à

Institut Supérieur d'Informatique de Mahdia

En vue de l'obtention

De la Licence en (Technologie Réseaux Informatiques)

Par

Mlle. MERIEM TEKEYA & Mlle. SABRINE HADJ AMMAR

CONCEPTION ET DÉVELOPPEMENT D'UNE APPLICATION MOBILE SOUS LA PLATEFORME ANDROID

Soutenu le 03/06/2013, devant la commission d'examen:

Mr. ANIS HAJ SAID

Président

Mr. OMAR CHEIKHROUHOU

Encadreur

Mr. MOURAD KMIMECH

Rapporteur

Année universitaire 2012/2013

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 frères Ahmed, Med Aziz et Med Ali,

A ma chère sœur et son mari,

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

A mon cher grand-père,

Qui je souhaite une bonne santé.

A ma chère binôme Meriam,

Pour sa entente et sa sympathie.

AAymenTekaya et Melik Ben Salem,

Pour leurs indéfectibles soutiens et leurs patiences infinies.

A mon cher Omar,

Qui m'a aidé et supporté dans les moments difficiles.

A mes chères ami(e)s Imen, Selma, Hamdi, Rabiaa et Housseem,

Pour leurs aides et supports dans les moments difficiles.

A toute ma famille,

A tous mes autres ami(e)s,

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

Sabrina Hadj Ammar

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 frères Anis, Aymen et Sami,

A mes chères Sihem et Oumayma ,

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

A ma chère binôme Sabrina,

Pour sa entente et sa sympathie.

AAymenTekaya et Melik Ben Salem,

Pour leurs indéfectibles soutiens et leurs patiences infinies.

A mon cher Saddam,

Qui m'a aidé et supporté dans les moments difficiles.

A mes chères ami(e)s Imen, Selma, Hamdi, Rabiaa et Housseem,

Pour leurs aides et supports dans les moments difficiles.

A toute ma famille,

A tous mes autres ami(e)s,

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

MariamTekaya

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 exprimons toute notre reconnaissance à **Mr.HajSaid Anis**, d'avoir bien voulu me faire l'honneur de présider le jury de ce mémoire.

Nous adressons nos sincères remerciements à **Mr.Kmimech Mourad**, membre de jury, pour avoir accepté d'être le rapporteur de ce manuscrit. Nous tenons à remercier tout d'abord **Mr.CheikhrouhouOmar** ,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.

Nous tenons aussi à exprimer notre profonde reconnaissance à **Mr. Ben SelemMelik et à Mr. TekayaAymen** pour ses conseils, ses commentaires précieux et le suivi de ce travail.

Sommaire

Introduction générale.....	1
Chapitre 1 Contexte et Problématique.....	3
1. Cadre de projet et problématique.....	4
1.1. Cadre du projet.....	4
1.1.1. Présentation de la société.....	4
1.1.2. Cahier de charge.....	4
1.1.3. Outils utilisées.....	5
1.1.4. Le serveur web.....	5
1.2. Problématique.....	5
2. Etude de l'existant.....	6
3. Objectifs.....	6
Conclusion.....	7
Chapitre 2 Etat de l'art.....	8
1. Internet mobile.....	9
2. Technologie de téléphonie mobile.....	9
2.1. Technologie 3G.....	9

2.2. Technologie 3G+.....	9
2.3. Technologie 4G.....	9
2.4. Smartphone.....	10
3. Système d'exploitation mobile.....	10
3.1. Les différents systèmes d'exploitation sur le marché.....	10
3.2. Partition du marché entre les SE.....	13
4. Le système d'exploitation Android.....	14
4.1. Définition et historique.....	14
4.2. Les versions de l'Android.....	14
4.3. Architecture logicielle.....	15
4.4. Kit de développement.....	17
5. La Géolocalisation.....	18
6. La sécurisation des communications.....	19
6.1. La définition du protocole HTTPS.....	19
6.2. Les objectifs de sécurité assurés.....	19
Chapitre 3 Analyse et Conception.....	21
1. Identification et analyse des besoins.....	22
1.1. Identification des besoins.....	22
1.1.1. Les besoins fonctionnels.....	22
1.1.2. Les besoins non fonctionnels.....	23
1.2. Analyse des besoins.....	23
1.2.1. Cas d'utilisation relatif à l'administrateur.....	23
1.2.2. Cas d'utilisation relatif à visiteur.....	24
1.2.3. La description détaillée des cas d'utilisation.....	25
1.3.1. Scénario du cas d'utilisation "consulter les endroits".....	25
1.3.2. Scénario du cas d'utilisation "gérer la liste des lieux".....	26
2. Conception détaillée.....	30
2.1. Les diagrammes de séquence.....	30
2.1.1. Diagramme de séquence du scénario "gérer la liste des lieux".....	31
2.1.2. Diagramme de séquence du scénario "consultation des endroits".....	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.....	43
2.1. Modèle logique du système d'information.....	43
2.2. Le modèle physique du système d'information.....	43
3. Résultats obtenus.....	44
Conclusion.....	46

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 visiteur.....	25
Figure 6. Cas d'utilisation "consulter les endroits".....	25
Figure 7. Scénario du cas d'utilisation "consulter les endroits".....	26
Figure 8. Cas d'utilisation « gérer la liste des endroits ».....	27
Figure 9. Scénario pour le cas « Ajouter un lieu ».....	28
Figure 10. Scénario pour le cas « modifier un lieu ».....	29
Figure 11. Scénario pour le cas « supprimer un lieu ».....	30
Figure 12. Diagramme de séquence du scénario d'ajout d'un lieu.....	31
Figure 13. Diagramme de séquence du scénario de modification d'un lieu.....	32
Figure 14. Diagramme de séquence du scénario de suppression d'un lieu.....	33
Figure 15. Diagramme de séquence du scénario de consultation des endroits.....	34
Figure 16. Diagramme de classe.....	35
Figure 17. URL Plugin.....	38
Figure 18. Liste des targets.....	39
Figure 19. Android SDK AVD Manager.....	39
Figure 20. Liste des AVD créés.....	40
Figure 21. Récupération du clé MD5 pour Google MAP.....	42
Figure 22. Obtention de la clé API.....	43
Figure 23. Interface d'accueil de l'application.....	44
Figure 24. L'application coté Administrateur.....	45
Figure 25. Partie d'application coté touriste.....	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 endroits ».....	26
Tableau 4 . Cas d'utilisation « Ajouter un lieu ».....	27
Tableau 5. Cas d'utilisation « modifier un lieu ».....	28
Tableau 6. Cas d'utilisation « supprimer un lieu ».....	29

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

De nos jours, l'internet domine le monde en lançant de nouvelles technologies qui ont imposé à l'utilisateur de suivre le rythme de son développement. En plus, le mobile fait une partie intégrante de notre vie et nous voyons même certains de nos compères souffrir du syndrome de monophonie (la peur d'être séparée de son téléphone mobile).

Depuis quelques années, le marché de téléphone portable connaît une évolution technologique, surtout depuis qu'Apple a lancé son i Phone. Il a changé la manière dont nous voyons le téléphone mobile. Cependant, un nouveau concurrent se présente face à l'i Phone : Androïde en 2008. D'après les rapports de Digital Marketing & Media Research Firm eMarketer, Androïde OS dépassera Apple pour atteindre la première loge parmi les OS mobiles d'ici à 2012. En plus, avec l'arrivée d'Internet les activités touristiques ont constitué un changement depuis quelques années ce qui a permis la possibilité d'étendre leurs activités. Les voyageurs utilisent le web pour planifier leurs voyages en effectuant plusieurs opérations telles que la réservation de leur vol et de leur hôtel. Mais dès qu'il se situe dans un endroit inconnue où il a besoin d'information, tel que trouver des place de loisir à proximité (restaurants ou cafétérias), ou communiquer avec son réseau social, etc, il aurait besoin d'un service plutôt sur son mobile.

Le nombre de « mobinautes » et « tablanauts » augmente sans cesse à travers le monde. Actuellement, 500 millions de personnes dans le monde utilisent internet comme service mobile et ce nombre sera triplé à 2013 grâce au Smartphones.

Les Smartphones peuvent être considérés comme des ordinateurs de poche intégrés à un téléphone mobile, un Smartphone permet généralement à l'utilisateur d'installer et d'exécuter des applications plus avancées. Les smartphones sont des téléphones intelligents nous permettant de surfer sur internet, envoyer et recevoir des e-mails, gérer notre agenda, messagerie instantanée, GPS, accès Wi-Fi, jeux, musique, vidéos. Nous pouvons personnaliser nos usages avec le téléchargement d'applications. La plupart des téléphones mobiles vendus sont des Smartphones : iPhone, BlackBerry, HTC, LG, Samsung.

De plus, l'industrie touristique se situe aux premières loges pour bénéficier de cette révolution de l'internet mobile et des téléphones intelligents. En effet, « Les technologies mobiles ont profondément modifiés notre rapport à l'information mais surtout son accessibilité. Pour le touriste, ce qui prévaut c'est un mode de consommation de l'information en situation. Les

recherches locales sur mobile ont par exemple été multipliées par 15 entre 2007 et 2008 » déclare Vincent Herman, Fondateur de Nexence.

Associé à la géolocalisation, le tourisme mobile propose aux utilisateurs de téléphone portable de se situer sur une carte dans la ville où on se trouve. Le service comprend non seulement la géolocalisation en temps réel, mais également, des informations sur la ville et ses centres d'intérêts (musées, parcs, monuments, événements culturels ou artistiques du moment, etc...) : un vrai guide touristique. L'originalité est qu'il permet à l'utilisateur de personnaliser son guide selon sa localisation.

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, l'ISIM@ offre à ses étudiants une opportunité de préparer un projet de fin d'études dans l'industrie. Dans ce cadre, nous avons réalisé notre projet de fin d'études au sein de la société Smart Vision pendant la période du 18 février 2013 au 11 mai 2013. Ce projet vient compléter notre formation scientifique et technologique et améliorer nos compétences dans un environnement industriel réel.

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

Smart Vision est une société de services informatiques spécialisée dans le développement des nouvelles technologies de l'information et de la communication. L'originalité de l'approche de Smart Vision consiste dans sa proximité avec ses clients européens par le biais de son antenne commerciale en France et de sa compétitivité grâce à la structure et ses couts et à son équipe d'ingénierie comptant 45 ingénieurs et techniciens basée en Tunisie.

La société Smart Vision a vu le jour le 04 Janvier 2006, prenant comme siège un des bureaux de la Technopole de Sousse.

Smart Vision a développé une expertise dans les domaines suivants :

- Web 2.0 et applications e-commerce.
- Applications mobiles et embarquées.
- Conseils et formation à l'intégration des systèmes

1.1.2. Cahier de charge

Cette application a pour but de satisfaire le besoin de :

- 1) Visiteur (client) qui peut consulter les pages publiques.
- 2) L'administrateur qui gère le site en ligne.

Tous les internautes peuvent accéder au site qu'il est publique et ne requiert pas une authentification.

Un client peut gérer son profil et aussi sa localisation.

L'administrateur peut gérer les listes des visiteurs, leurs catégories ainsi que la localisation de leurs endroits.

1.1.3. Outils utilisées

Pour créer 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 web 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 classique 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].

1.1.4. Le serveur web

C'est un protocole de transfert de fichier (essentiellement au format http) localisé grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web.

- **WampServer** : C'est une plate-forme de développement Web sous Windows pour des applications Web dynamique à l'aide de serveur Apache2, du langage script PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données [2].

Parmi ces avantages, nous pouvons citer :

- Il est gratuit.
- Son utilisation est simple, on s'habitue rapidement avec son interface.
- Il dispose de nombreuses fonctionnalités avancées.

Son seul inconvénient est la complexité pour envoyer des mails avec le logiciel.

1.2. Problématique

La vulgarisation d'internet a permis aux professionnels du tourisme d'étendre leurs activités et de développer différents concepts. Le tourisme électronique a définitivement trouvé sa place sur le web. Toujours dans le but de satisfaire et de faciliter la vie des consommateurs de produits touristiques, un nouveau concept est né : le tourisme mobile ou m-tourisme.

Associé à la géolocalisation, le tourisme mobile propose aux utilisateurs de téléphone portable de se localiser sur une carte ou sur un plan à l'aide de ses coordonnées, géolocalisation réel ou en différé. La géolocalisation fait désormais partie de fondamentaux dans le tourisme. Les touristes ont besoin de ce type d'entrée pour préparer leur séjour : itinéraire, les endroits à visiter (musées, restaurants, magasins...) à proximité des lieux de séjour. De ce fait, le développement d'une application mobile servant comme guide touristique est très prometteuse.

Pour bénéficier d'une telle application, il faut avoir accès à la technologie internet 4G. Sur l'échelle nationale, la technologie internet 4G est lancée avec l'opérateur Tunisie Télécom à partir de 2013.

2. Etude de l'existant

L'internet participe aujourd'hui notablement à l'évaluation, au choix, aux décisions spontanées d'achat de produits au de prise d'informations touristiques. En effet, l'e-tourisme a permis à ce secteur de profiter de la possibilité d'hébergement de réservation, de paiement ainsi que de diffusion des spots publicitaire.

Les technologies mobiles font parti de la nouvelle mutation de tourisme. En effet, avec cette technique, nous aurons la possibilité en temps réel d'accéder aux informations et services touristiques grâce à un système de communication sans fil, basé sur les téléphones mobiles. Le but de notre travail est de faciliter l'accès au données des emplacements en utilisant la possibilité de géolocalisation et la création de base de données des lieux (cafés, restaurants, parking, ... etc). Cette application permet aux utilisateurs de consulter et d'afficher les endroits les plus proches de leurs emplacement actuels ainsi que de créer des comptes utilisateurs permettant d'enregistrer les lieux plus visités et les favoris.

3. Objectifs

L'application T_Guide à développer doit fournir les services suivants :

- Permet l'utilisateur soit visiteur, soit administrateur de s'inscrire au sein de l'application.
- Permet aux utilisateurs de gérer leurs comptes.
- Visualiser la position actuelle de touriste sur une Map et tous les endroits au coin.

- Consulter les informations détaillées relatives à l'endroit déjà sélectionné.
- Gérer la liste des favoris des endroits.
- Consulter la liste des promotions relatives à l'endroit au coin.
- Gérer la liste des commerces pour les propriétaires.
- Gérer la liste de promotions offertes par les propriétaires.
- Réserver une place dans un endroit choisi.
- Confirmer les demandes de réservation déjà délivré par les touristes.
- 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.

Chapitre 2
Etat de l'art

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 Error: Reference source not found.

2. Technologie de téléphonie mobile

2.1. Technologie 3G

Nommée aussi "technologie de troisième génération", elle est devenue disponible au public depuis 2002 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 [4]..

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 Error: Reference source not found.

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*) Error: Reference source not found.

3. 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](#)) Error: Reference source not found.

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

4. 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 Error: Reference source not found.

4.1. 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 (*Intarnetwork 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 ¹	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

4.2. Partition du marché entre les SE

D'après la publication d'IDC (*International Data Corporation*), la plateforme Android de Google a 39,5% du marché mondial des *Smartphones*, contre 15,7% pour les iPhone d'Apple. Loin devant Symbian de Nokia, qui représenterait 20,9% du marché à la fin de l'année, d'après l'institut IDC.

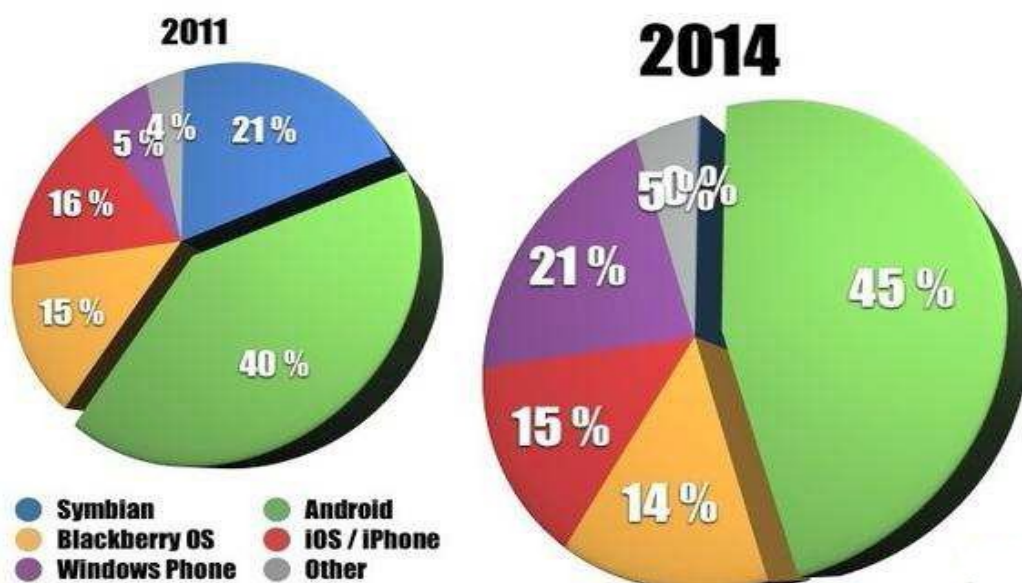


Figure 1 . Les parts de marché des systèmes d'exploitation mobile pour les années 2011 et 2014

En 2015, la plateforme Android de Google devrait avoir 45,4% du marché mondial des *Smartphones*, contre 15,3% pour les iPhone d'Apple, 13,7 % pour BlackBerry OS et 20,9 % pour Windows Phone7, d'après l'institut IDC.

D'après une autre étude menée par IDC, on a montré que l'OS Android confortera sa place de leader du marché des systèmes d'exploitation mobiles à l'horizon 2015. Ainsi, il s'octroiera plus de 45% de part de marché. A l'inverse, l'OS Symbian pourrait quasiment disparaître du paysage en 2015, date à laquelle le système de Nokia ne conserverait que 0,2% de part de marché.

IDC indique enfin que le marché mondial des *Smartphones* devrait connaître une croissance en 2015, soit plus du double de ce qui est espéré pour l'année 2011 (472 million).

5. Le système d'exploitation Android

5.1. Définition et historique

Android est un système d'exploitation Open Source pour Smartphones Error: Reference source not found, 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.

5.2. Les versions de l'Android

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

Version	Nom	API Level	Distribution
1.5	Cupcake	3	0.2%
1.6	Donut	4	0.5%
2.1	Eclair	7	4.2%
2.2	Froyo	8	15.5%
2.3 – 2.3.2	Gingerbread	9	0.3%
2.3.3 – 2.3.7		10	60.3%
3.1	Honeycomb	12	0.5%
3.2		13	1.8%
4.0 – 4.0.2	Ice Cream	14	0.1%
4.0.3 – 4.0.4	Sandwich	15	15.8%
4.1	Jelly Bean	16	0.8%

Tableau 2. Les versions de l'Android

C'est les dernières statistiques qui datent du janvier 2013 concernant la répartition des différentes versions Android.

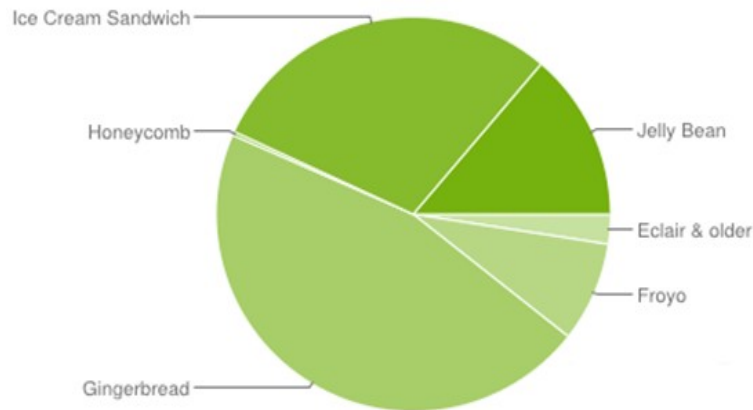


Figure 2. La part de chaque version d'Android

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 2.3 qui est toujours *leader* avec 45,6%, on peut s'apercevoir aussi que la version Android 4.0 prend une bonne proportion du marché avec 29%. Les autres versions restent anecdotiques avec 1,3 % pour Honeycomb (Android 3.x), 2,2 % pour Eclair (Android 2.1) et 0,2 % pour Donut (Android 1.6).

5.3. Architecture logicielle

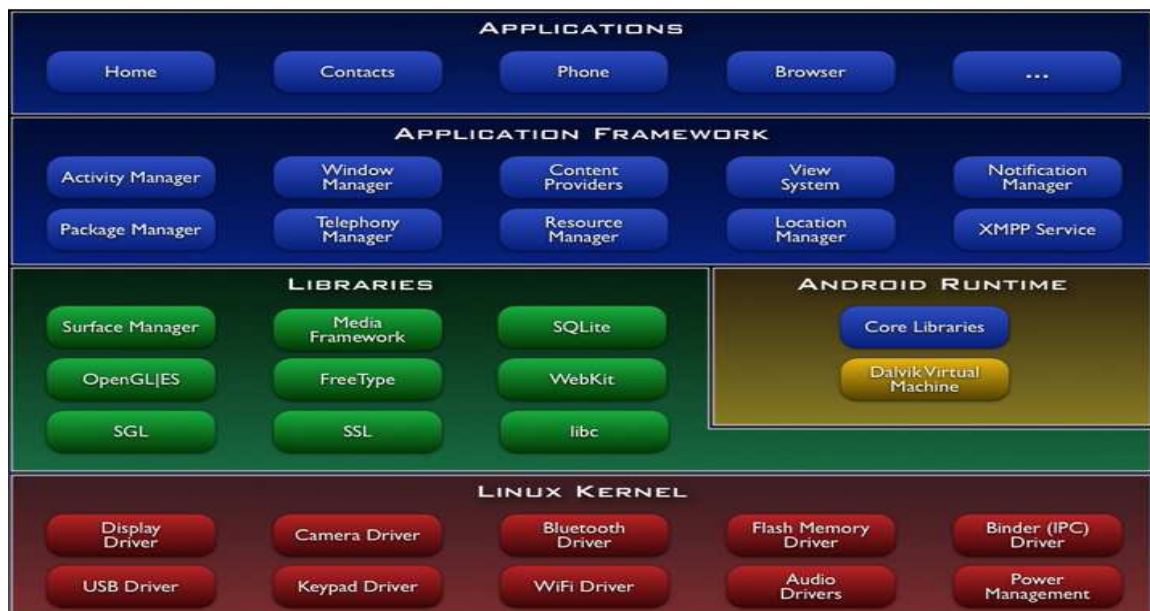


Figure 3. Architecture logicielle de l'Android

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 librairies natives constituant les couches bases du système. Ces librairies 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.

5.4. 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.

6. 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 Error: Reference source not found.

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 trouve 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.

7. 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*).

7.1. La définition du protocole HTTPS

"*HyperText Transfer Protocole Secured*" (HTTPS) est un protocole de transfert hypertexte sécurisé qui a été développé par Netscape. Reference source not found.

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.

7.2. 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 comme Android et le protocole HTTPS.

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 visiteur.

Administrateur :

Cette application permet à l'administrateur de :

- Ajouter des nouveaux lieux,
- Modifier des lieux où de données relatifs aux lieux,
- Supprimer des lieux,
- Affiche les lieux de disponibilités,
- Mettre les contacts à jour,
- Afficher les promotions et gérer la réservation en ligne.

Visiteur :

Les fonctions offertes pour un visiteur sont :

- S'authentifier,
- Consulter de liste des lieux,
- Savoir la localisation les lieux les plus proches,
- Consulter la liste des endroits favoris,
- Ajouter, supprimer et modifier la liste des favoris,
- Consulter la liste des promotions offertes,
- Réserver une place sélectionnée.

1.2.1. 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 (identification).
- L'ergonomie des interfaces de l'application mobile et web.

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 ; que ce soit un administrateur (Figure 4), ou un visiteur (Figure 5).

1.1.1. Cas d'utilisation relatif à l'administrateur

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

- Gérer les catégories des visiteurs.
- Gérer la localisation des endroits.
- Gérer la liste des promotions : une fois l'administrateur possède un lieu dans la base de données il peut ajouter des promotions relatives à leur lieu touristiques. Tout en précisant une description de la promotion ainsi que sa période de validité.

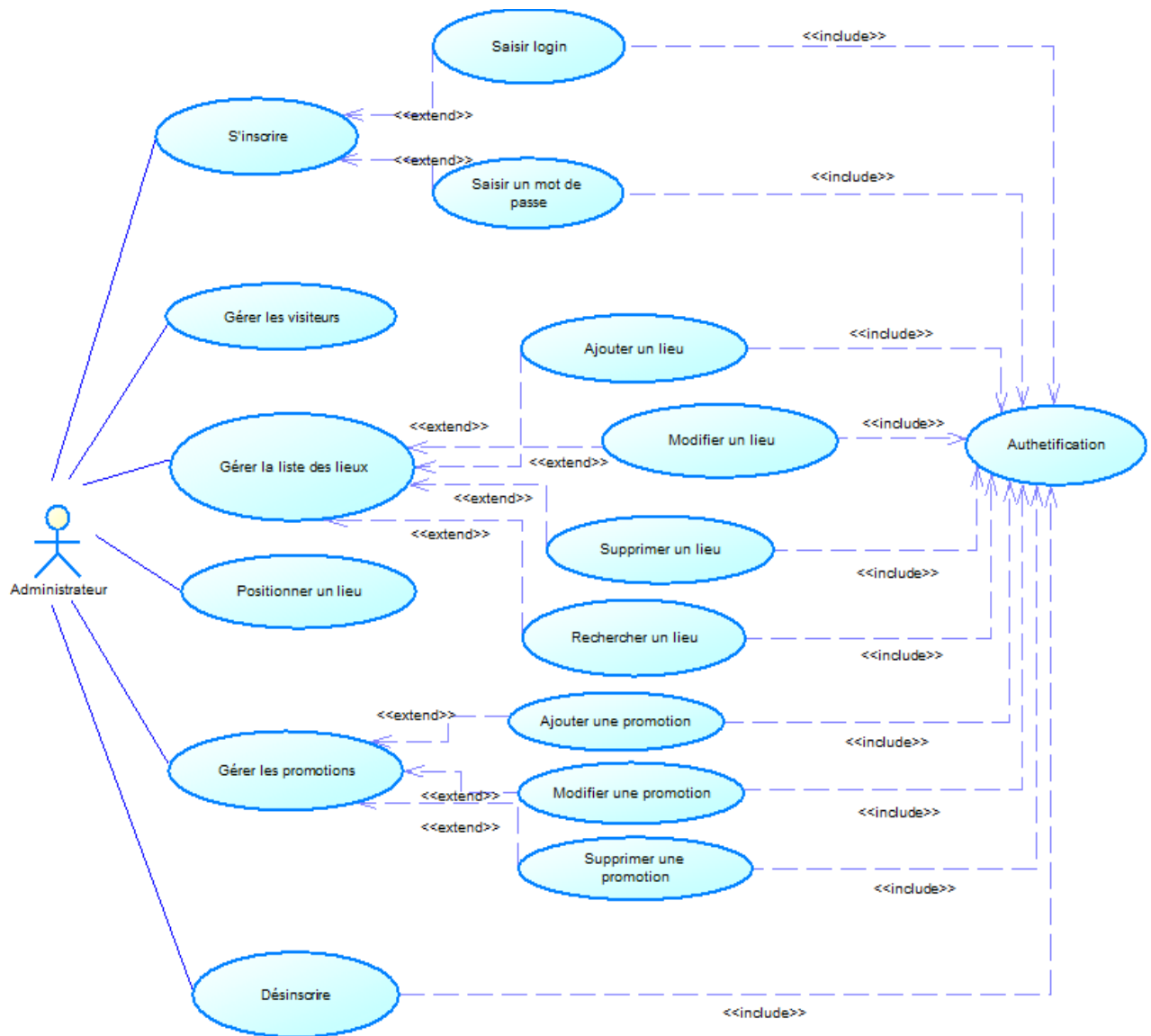


Figure 4. Diagramme de cas d'utilisation relatif à l'administrateur

1.2.1. Cas d'utilisation relatif à visiteur

Le visiteur peut :

- Consulter la liste des endroits : l'application permet de visualiser les restaurants, les cafétérias, et les hôtels les plus proches de l'utilisateur. En effet, l'utilisateur peut visualiser sa position récupérée à travers le GPS et l'afficher sous la carte. Aussi, ces endroits sont représentés par des icônes.
- Consulter les promotions : le client peut profiter des promotions données par l'administrateur. En fait, selon l'endroit où se trouve l'utilisateur les promotions sont affichées dans une liste.

- Gérer la liste des endroits favoris : cette fonction permet à l'utilisateur de créer une liste de favoris contenant les endroits préférés. Elle permet aussi de supprimer des endroits appartenant à cette liste.

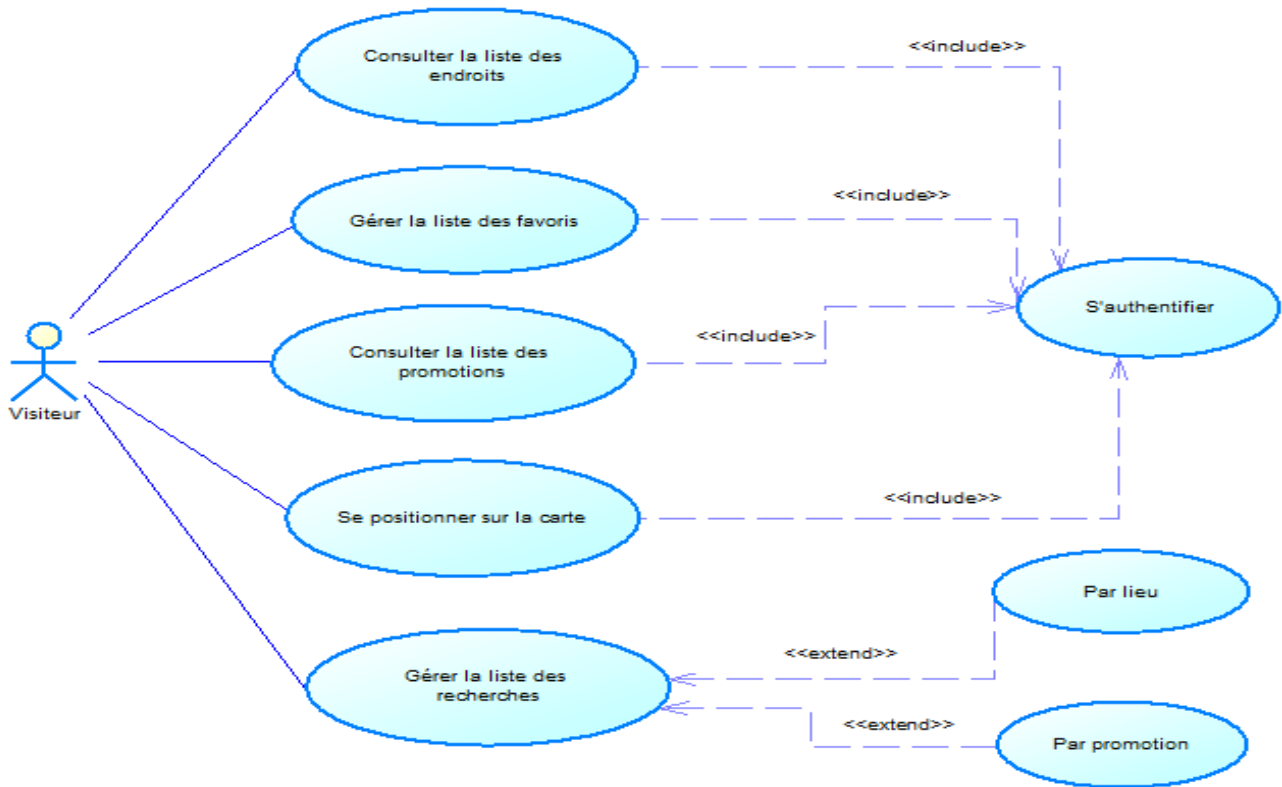


Figure 5. Diagramme de cas d'utilisation relatif à visiteur

1.3. La description détaillée des cas d'utilisation

1.1.1. Scénario du cas d'utilisation "consulter les endroits"

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

La Figure 6, présente le cas d'utilisation « consulter les endroits » de l'acteur visiteur.



Figure 6. Cas d'utilisation "consulter les endroits"

Pré-condition	L'utilisateur peut effectuer une consultation en tant que propriétaire ou client.
Scénario	La position de l'utilisateur sera affichée sur la carte géographique. Ainsi que les endroits les plus proches de l'utilisateur ().

Tableau 3. Cas d'utilisation « consulter les endroits »

Le digramme de scénario de ce cas d'utilisation est présenté dans la figure 7.

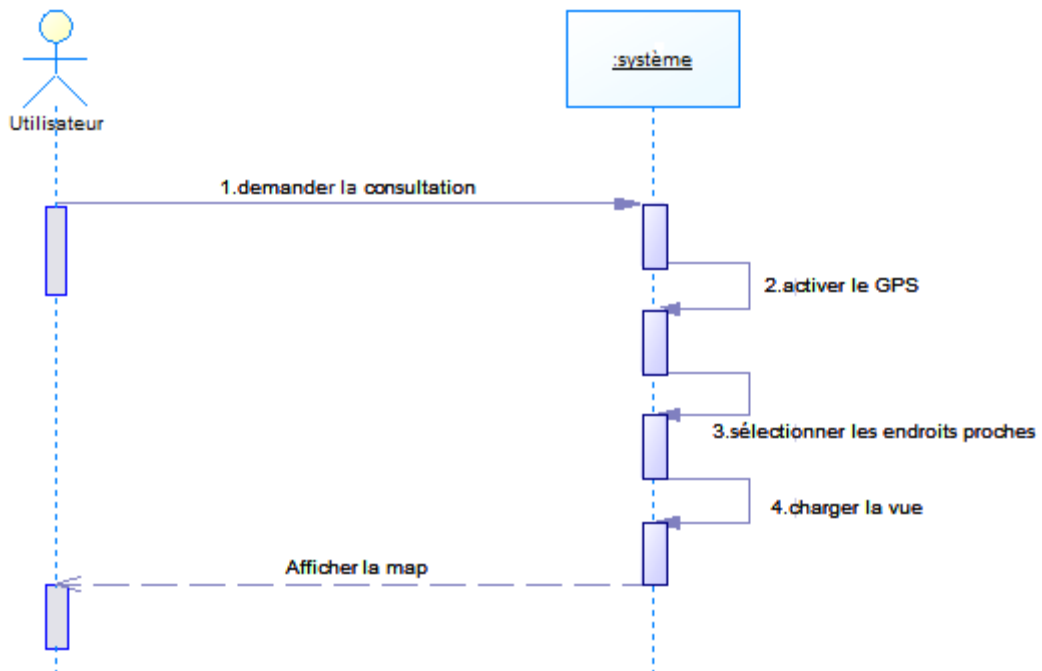


Figure 7. Scénario du cas d'utilisation "consulter les endroits"

1.2.1. Scénario du cas d'utilisation "gérer la liste des lieux"

Le cas d'utilisation « Gérer la liste des lieux » de l'acteur administrateur, englobe trois scénarios (Figure 8):

- Ajouter un lieu,
- Modifier un lieu,
- Supprimer un lieu.

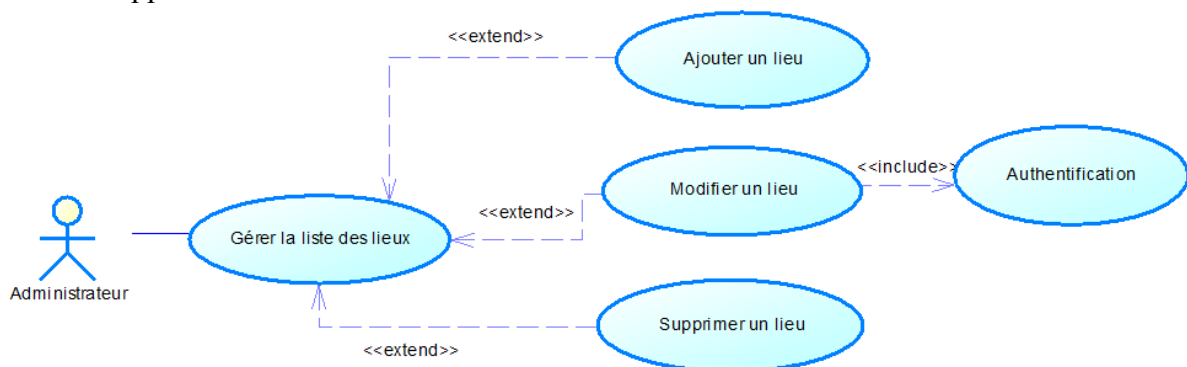


Figure 9. Cas d'utilisation « gérer la liste des endroits »

Pour ce cas d'utilisation, nous traitons le cas de restaurant.

Ajouter un lieu

Cas d'utilisation « gérer la liste des lieux »		
Ajouter un lieu	Pré-condition	L'administrateur doit s'authentifier tout d'abord au sein de l'application.
	Scénario	L'administrateur sélectionne le bouton d'ajout d'un lieu. Remplit le formulaire en choisissant le type de lieu. Pour le cas d'un restaurant, l'administrateur doit sélectionner la spécialité, la cuisine, l'ambiance et leurs descriptions. Une fois l'administrateur valide l'ajout d'un lieu, il est ajouté à la base de données (Figure 9).

Tableau 4. Cas d'utilisation « Ajouter un lieu »

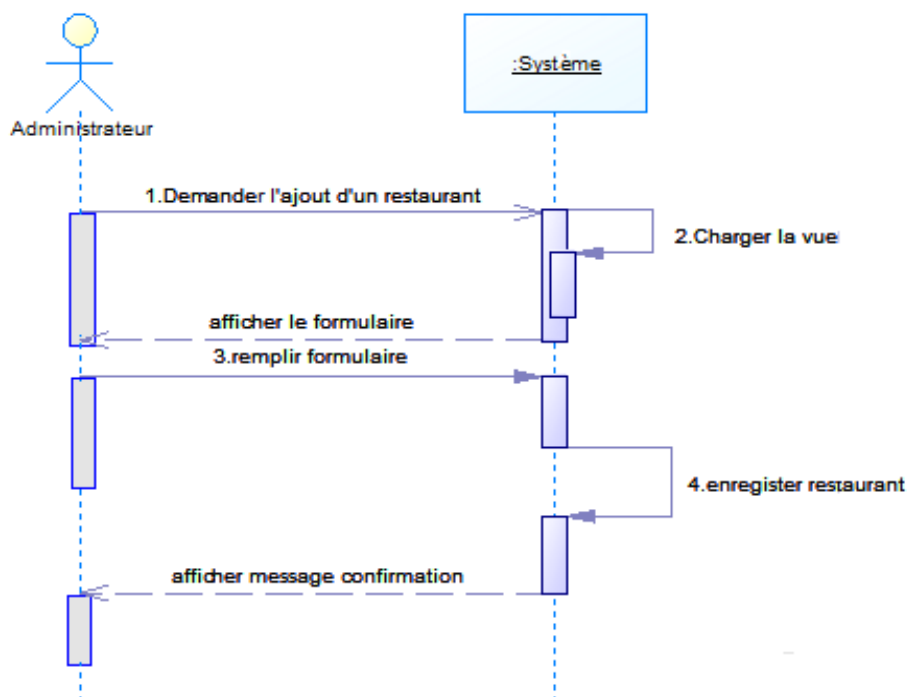


Figure 9. Scénario pour le cas « Ajouter un lieu »

Modifier un lieu

Cas d'utilisation " gérer la liste des lieux "

Modifier un lieu	Pré-condition	L'administrateur doit être s'authentifier tout d'abord au sein de l'application.
	Scénario	L'administrateur sélectionne le restaurant désiré qui existe dans sa liste des lieux. Le même formulaire d'ajout mais il est rempli par les données de ce restaurant. L'administrateur peut modifier les données et les valider s'il le veut. Donc le mis à jour sera prie en considération est sauvegardé dans la base de données (Figure 10).

Tableau 5. Cas d'utilisation « modifier un lieu »

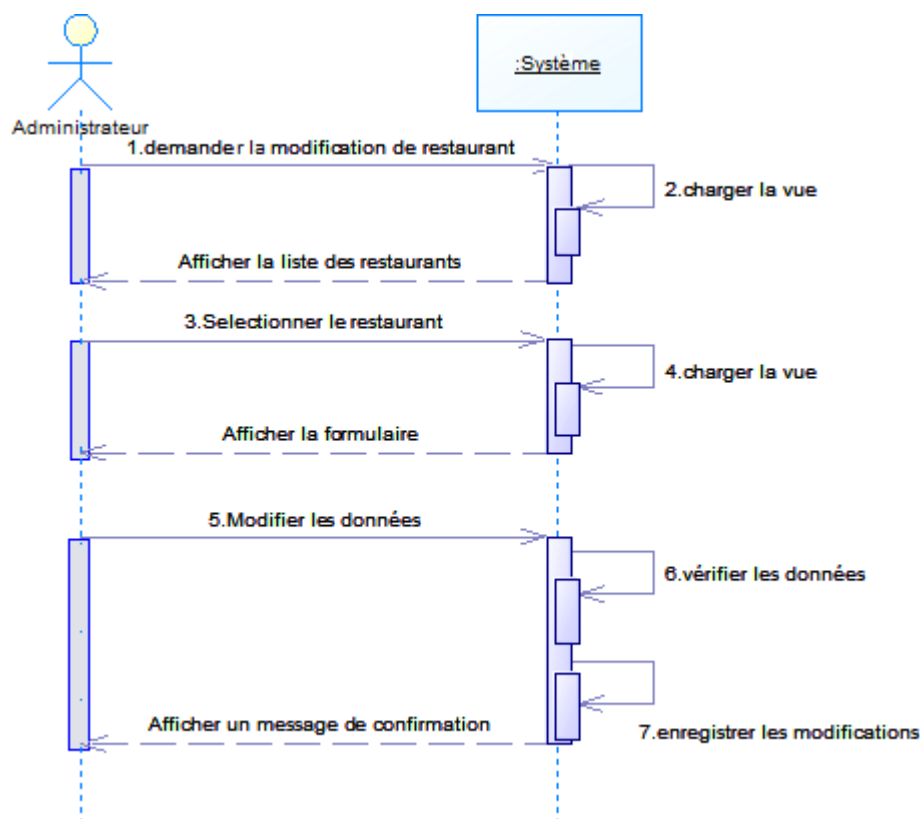


Figure 10. Scénario pour le cas « modifier un lieu »

Supprimer un lieu

Cas d'utilisation " gérer la liste des lieux "		
Supprimer un lieu	Pré-condition	L'administrateur doit s'authentifier tout d'abord au sein de l'application.
	Scénario	L'administrateur sélectionne le restaurant désiré qui existe dans sa liste des lieux. Toutes les informations concernant le restaurant sont affichées. Puis l'administrateur peut valider ou annuler la suppression du lieu (Figure 11).

Tableau 6. Cas d'utilisation « supprimer un lieu »

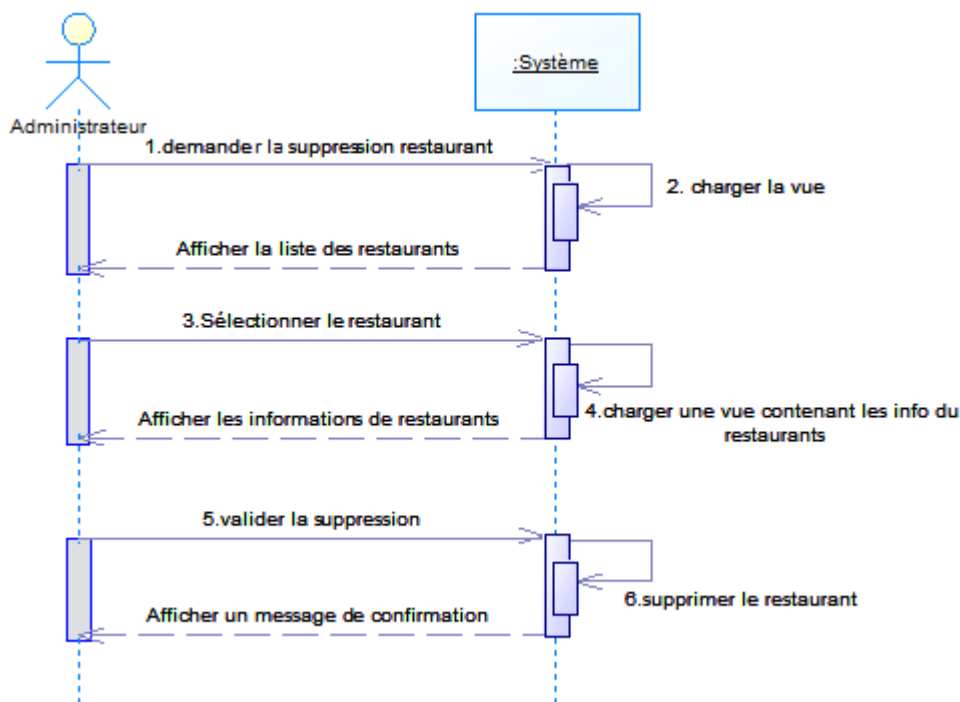


Figure 11. Scénario pour le cas « supprimer un lieu »

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, la 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 envois des messages. Nous allons présenter le diagramme de séquence pour deux cas d'utilisation : « gérer la liste des lieux » et « consultation des endroits ».

2.1.1. Diagramme de séquence du scénario "gérer la liste des lieux"

- **Ajouter un lieu**

L'administrateur a la possibilité d'ajouter son lieu dans la base de données des places visible pour l'utilisateur de l'application. En fait, la [figure 12] montre qu'il faut remplir le formulaire en précisant le type de lieu soit restaurant, soit cafétéria ou parking aussi d'autre informations tel que l'adresse et une description. Après la vérification des données, ils seront sauvegardés dans la base de données à travers le contrôleur "c_restaurant" dans ce cas.

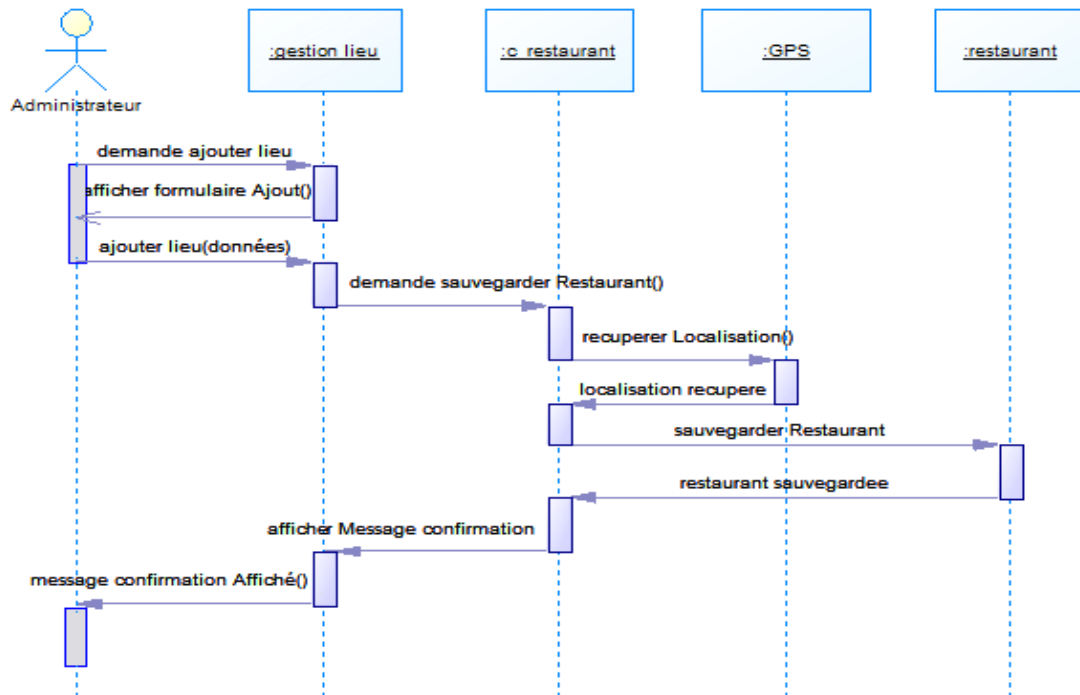


Figure 12. Diagramme de séquence du scénario d'ajout d'un lieu

- **Modifier un lieu**

D'après la figure 13, l'administrateur possède aussi la possibilité de modifier les informations relatives au lieu "restaurant". Le même formulaire d'ajout est affiché mais cette fois remplis par les données déjà extraites de base de données.

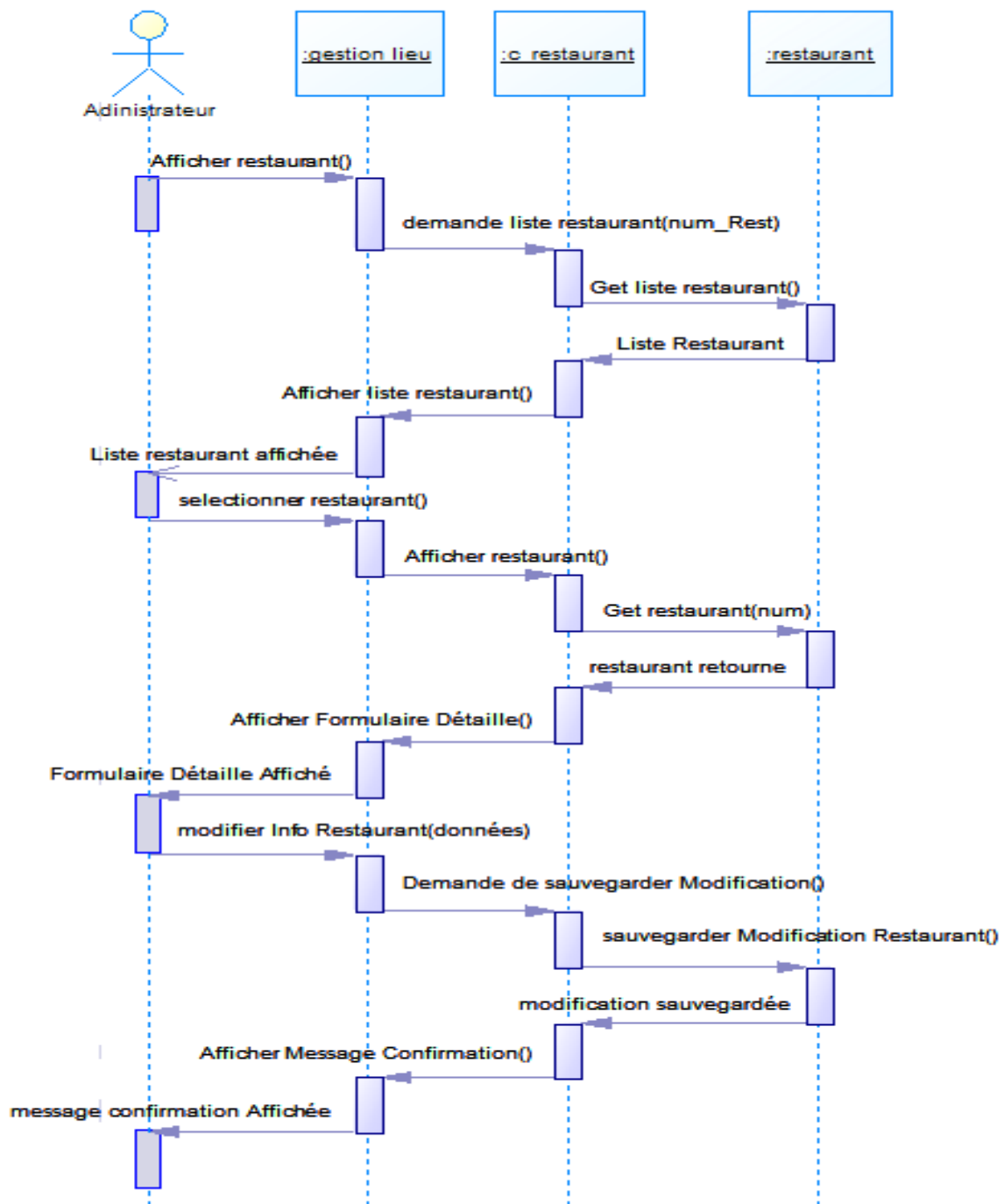


Figure 13. Diagramme de séquence du scénario de modification d'un lieu

- **Supprimer un lieu**

Une fois le restaurant est ajouté, l'administrateur peut le trouver dans la liste de ces lieux. Donc il peut sélectionner un et décider de le supprimer de sa liste. En fait, une liste contenant les lieux s'affiche et il sélectionne un. Puis, les informations détaillées de ce lieu seront affichées et il prend sa décision de le supprimer ou non. C'est bien illustré par la figure 14.

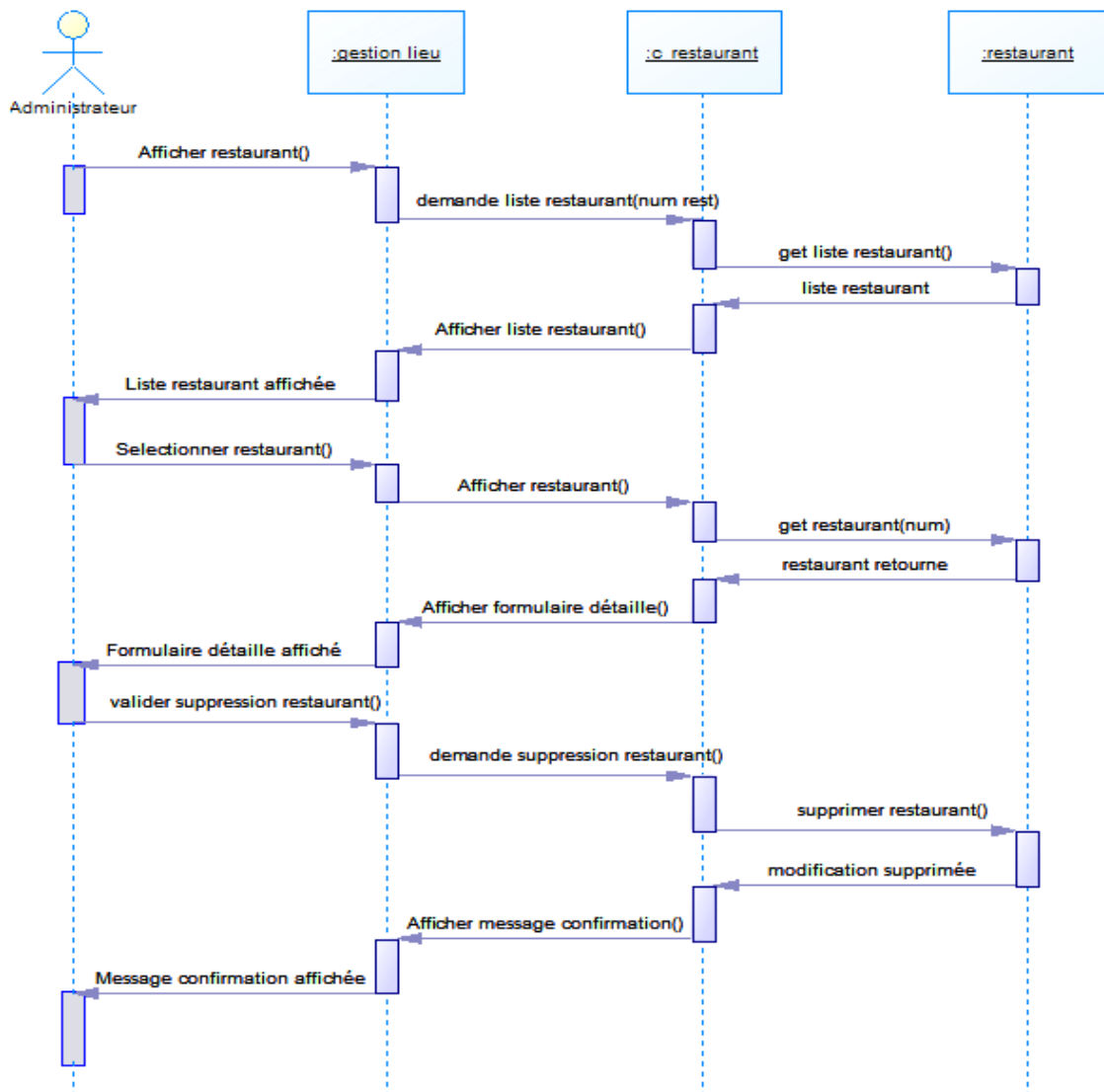


Figure 14. Diagramme de séquence du scénario de suppression d'un lieu

2.1.2. Diagramme de séquence du scénario "consultation des endroits"

L'utilisateur peut découvrir les endroits au coin tel que restaurant et cafétéria. En fait, le contrôleur "c_GPS" permet de récupérer la position actuelle du touriste à travers le GPS et l'afficher sur une map de Google. Puis envoyer ces coordonnées au serveur pour récupérer celles des lieux les plus proches et les afficher sur la carte à travers l'interface "gestion_lieu".

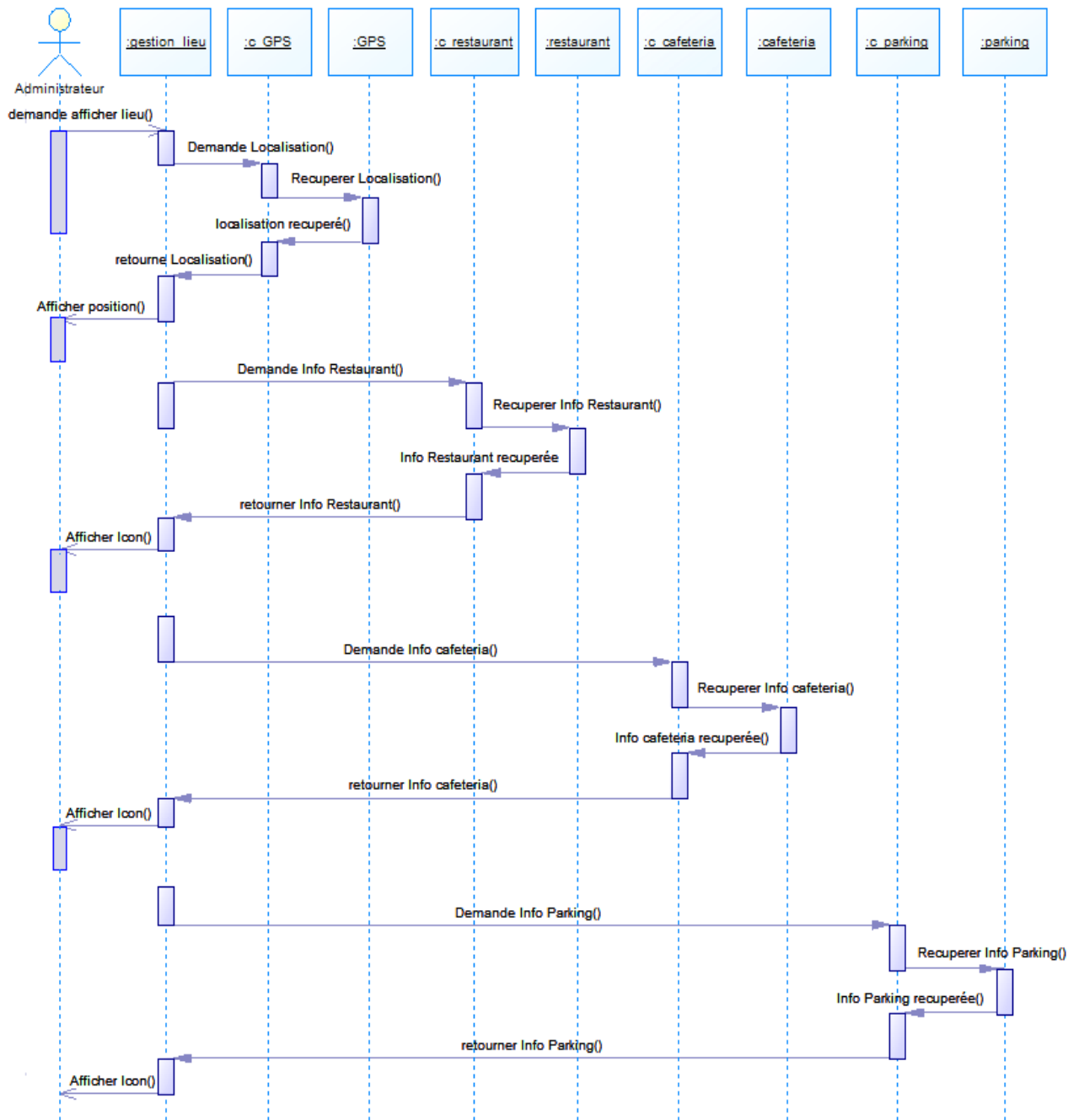


Figure 15. Diagramme de séquence du scénario de consultation des endroits

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 liens entre les instances de classe.

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

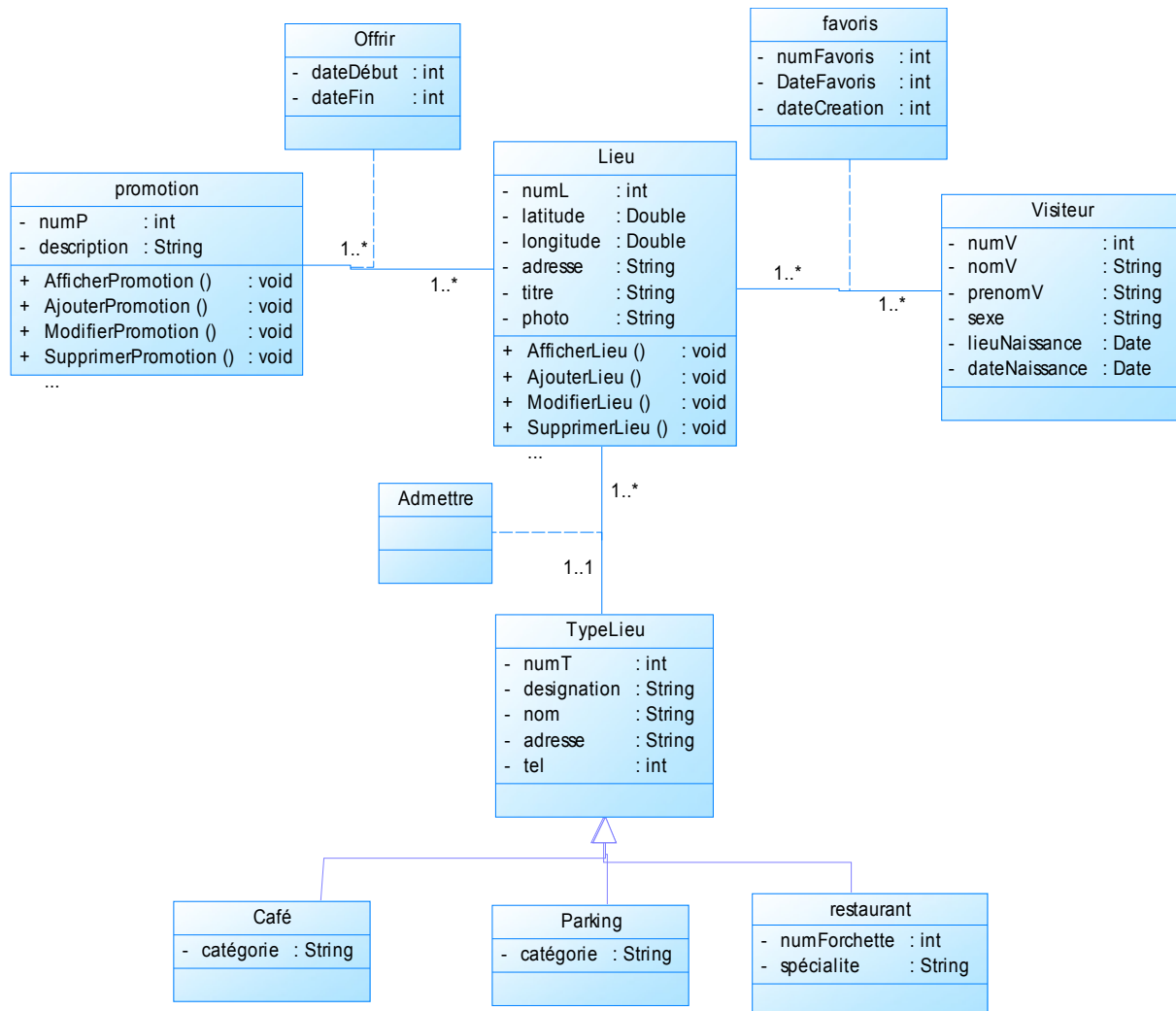


Figure 16. Diagramme de classe

Ce diagramme comporte un ensemble de classes définissant les entités du monde réel du système d'informations d'une application de guide touristique. 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/>.

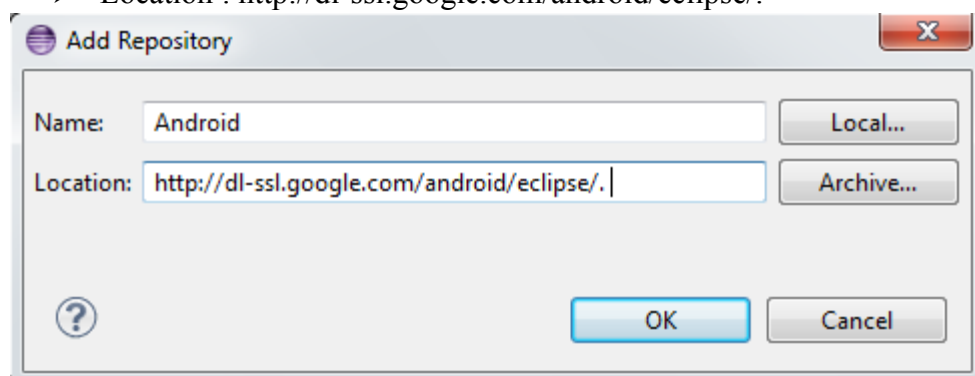


Figure 17. URL Plugin

- Après rafraichissement, Eclipse affiche les plugins disponibles. Nous sélectionnons le « Android DDMS » et 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 :

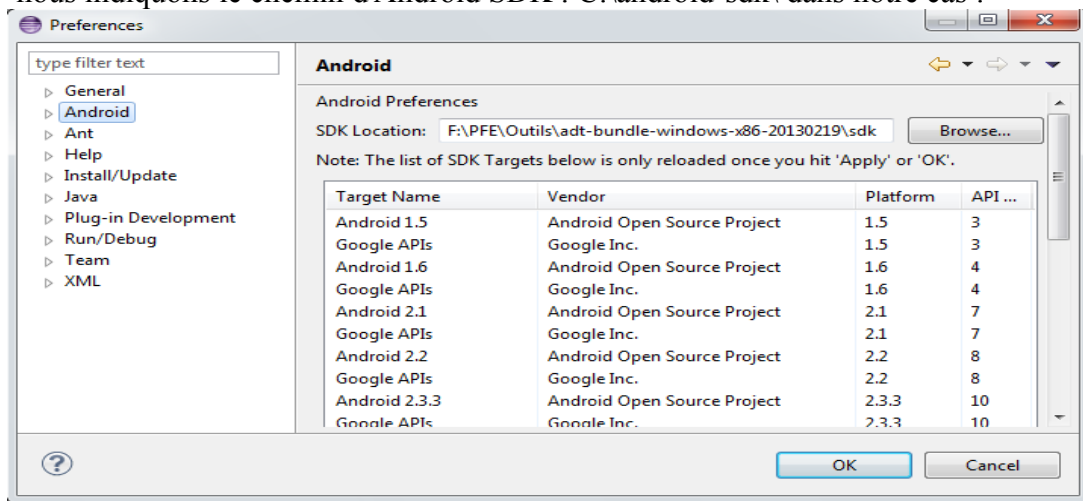


Figure 18. Liste des targets

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 :

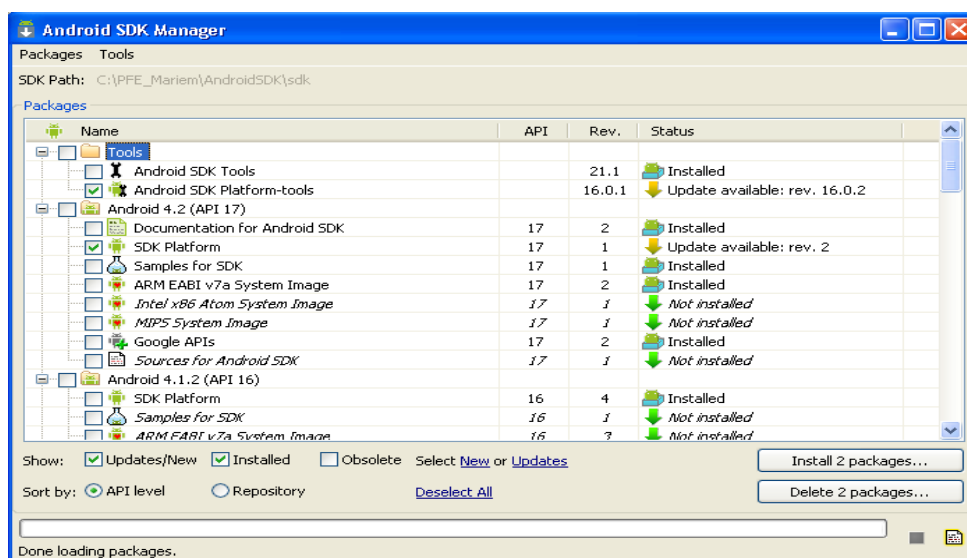


Figure 19. Android SDK AVD Manager

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, « hello_avd » 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 ».

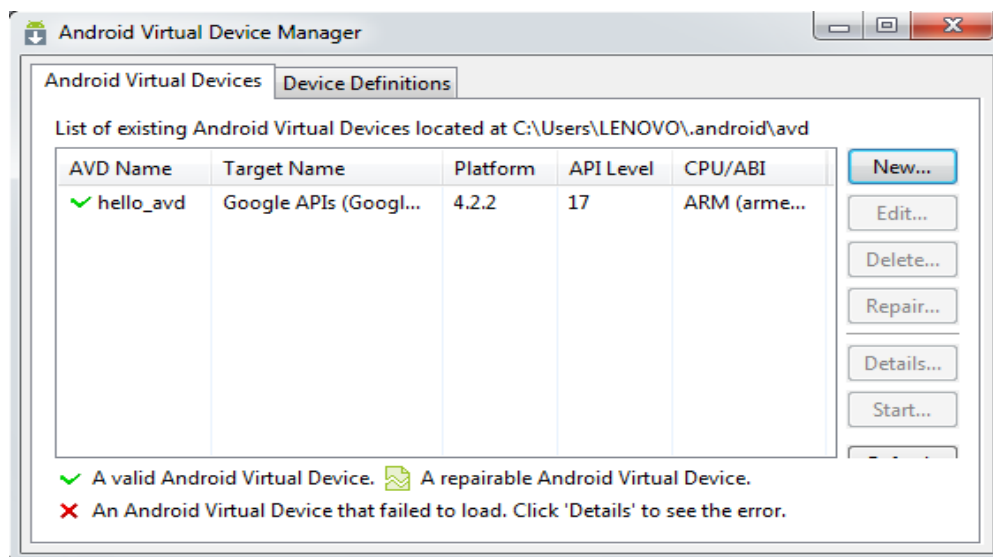


Figure 20. Liste des AVD créés

1.5. Bibliothèques externes

Quelques bibliothèques Android et Java utilisées :

- **android.os** : elle offre des services de base du SE, le passage de message, et de la communication interprocessus sur le mobile.
- **android.view** : elle offre les classes qui gèrent la disposition de l'écran et l'interaction avec l'utilisateur de mobile.
- **android.widget** : elle contient widget des éléments d'interface à utiliser sur l'écran de votre demande.

- **org.apache.http** : elle offre le standard de java qui possède les interfaces et classes de base des composants HTTP.
- **org.json** : elle permet de manipuler des objets json qui servent au transfert des données entre le mobile et le serveur.
- **android.app** : elle offre des classes de haut niveau encapsulant le modèle d'application générale Android.
- **android.content** : elle contient des classes pour l'accès et la publication de données sur un mobile.
- **android.database** : elle contient des classes qui permettent d'explorer les données renvoyées par un fournisseur de contenu.

1.6. Fichier source d'un projet

On a deux types de fichier source:

1.6.1. 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.

1.6.2. 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.

1.7. 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 :

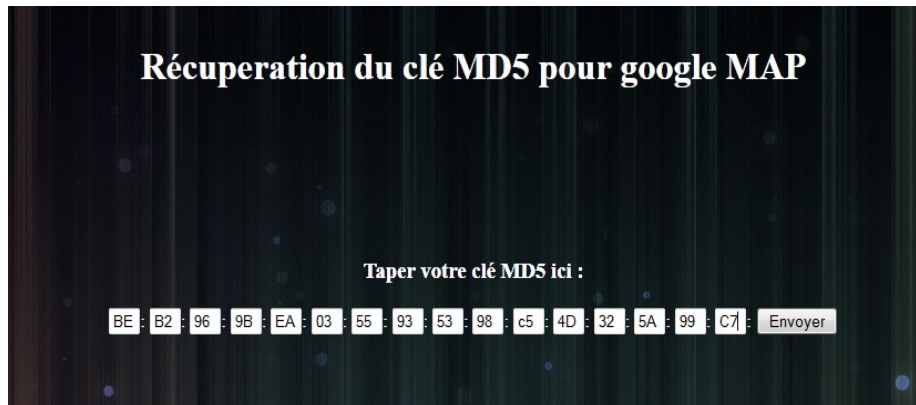


Figure 21. Récupération de la clé MD5 pour Google MAP

➤ Key Api

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

Et voici notre clé :

```
<com.google.android.maps.MapView
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:apiKey="0oP4U3z7NwRFHUw8sHenZ51RkxbU1Pu7d3eNsQg"
/>
```

Figure 22. Obtention de la clé API

2. Persistance de donnée

2.1. 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.

- Visiteur (numV, nom, Prénom, sexe, lieuNaissance, dateNaissance),
- Lieu (numL, latitude, longitude, adresse, titre, photo, #numFavoris),
- Favoris (#numV, #numL, numFavoris, DateFavoris, dateCreation),
- Promotion (numP, description),
- OffrirPromotion (#numP, #numL, dateDebut, dateFin),
- TypeLieu (numT, désignation, nom, adresse, tel).

2.2. Modèle physique du système d'information

Afin d'implanter le schéma relationnel obtenu, nous utilisons un système de gestion de base de donnée (SGBD) qui est un ensemble de logiciels permettant de manipuler des bases de données, c'est-à-dire de les créer, de les organiser et de les utiliser. Ce système permet de présenter le modèle physique du système d'information.

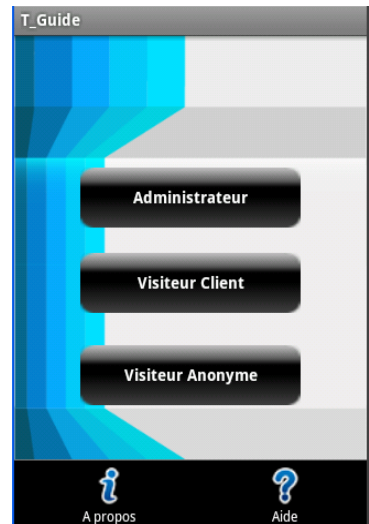
Le SGBD choisi est le MySQL sous sa version 5.

3. Résultats obtenus

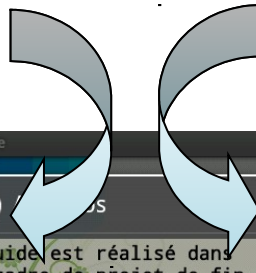
Notre application est développée de façon incrémentale et itérative. Comme première itération, nous avons commencé par la réalisation de cas d'utilisation "consultation des endroits" puis le cas d'utilisation "s'inscrire" suivie des autres. Dans cette partie, on va présenter quelques résultats obtenus.



Interface de démarrage



Interface d'accueil



Rubrique A propos

Rubrique Aide

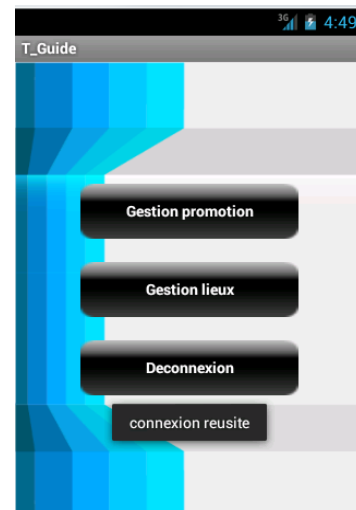


Figure 23. Interface d'accueil de l'application

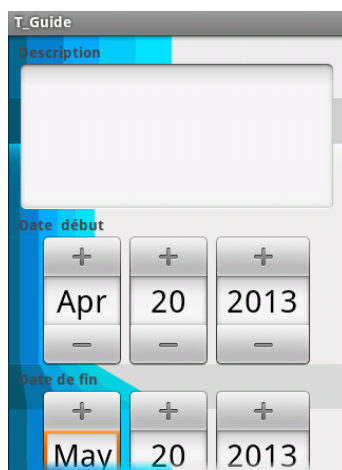
Lors du démarrage de l'application, un "splash screen" se lance, puis une interface permet soit d'accéder en tant qu'administrateur ou visiteur client ou visiteur anonyme, ou bien d'avoir les propos l'application ou bien d'accéder à l'aide. Sachant que l'interface change selon le type de l'utilisateur. Ces interfaces sont illustrées par la figure 23.



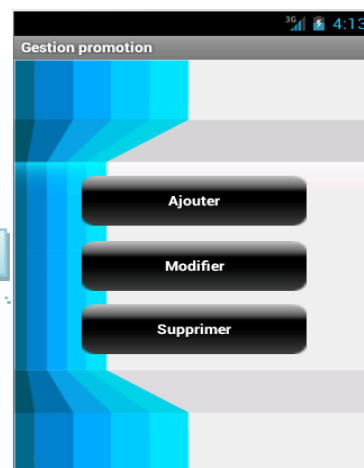
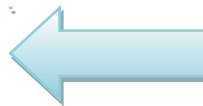
Interface d'authentification



Interface de Gestion



Interface d'ajout une promotion



Interface Gestion promotion



Figure 24. L'application coté administrateur

Pour cette interface, l'utilisateur administrateur doit fournir les informations d'authentification (nom d'utilisateur, mot de passe).

En cas du succès, l'interface de la liste de fonctionnalité d'un administrateur s'affiche.

En cliquant sur le bouton « Gestion promotion », l'interface « Gestion promotion » s'affiche. Suite à un choix d'ajouter une promotion, l'interface adéquate s'affiche.



Interface d'inscription

Interface de validation d'inscription

Interface de Choix



Interface Google Map

**Figure 25.** *Partie de l'application coté touriste*

La figure 25 n'illustre pas toutes les interfaces de l'application. En effet, elles représentent les principales activités de l'application comme la consultation des endroits, gestion des favoris, etc.

L'application possède une partie spécifique à l'administrateur. Il contient donc une activité pour la gestion des lieux, une autre pour la gestion des promotions, etc.

L'API Android offre une panoplie de composant permettant de faciliter le développement d'une application tel que les listActivity, Adaptor, etc. Aussi, il offre la possibilité d'avoir des contextes menu en plus de menu qui s'affiche à partir du bouton menu disponible dans les téléphones.

Conclusion

Notre projet a permis de concevoir et de réaliser une application mobile servant de guide touristique pour trouver les endroits à la demande, tel que les restaurants, les cafétérias, les parkings, etc.

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 d'utilisation.

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 la technologie J2EE et des logiciels libres tels que l'IDE Eclipse, le SGBD

MySQL et le serveur d'application Wamp et d'apprécier la qualité des services offerts par ce type de logiciels.

Bibliographie

- [1]. www.eclipse.org
- [2]. www.wampserver.com
- [3]. www.tunisiait.com
- [4]. www.latribune-online.com
- [5]. www.lemag.pro.bouyguestelecom.fr
- [6]. www.lemag.pro.bouyguestelecom.fr
- [7]. Fr.wikipedia.org
- [8]. Fr.wikipedia.org
- [9]. Fr.wikipedia.org/wiki/Android
- [10]. Damien Guignard, Julien Chable, Emmanuel Robles : Livre Programmation
Android
- [11]. Fr.wikipedia.org
- [12]. www.linternaute.com

