

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

Université A/Mira de Béjaïa

Faculté des Sciences Exactes
Département d'informatique



Thème

*Conception et réalisation d'une application
de messagerie instantanée sous Android*

MÉMOIRE DE FIN D'ETUDE

En vue de l'obtention du diplôme de Master

Option : Administration et sécurité des réseaux

Soutenu devant le jury composé de :

Présidente : Mme BOULAHROUZ

Examineur : Mr LARBI

Examinatrice : Melle ALBANE

Réalisé par :

Mr Lahcene KEDJAR

Mr Mokhtar OUMAKHLOUF

Encadrés par :

Mr AISSANI Sofiane

Dédicaces

À mes très chers parents

Avec un énorme plaisir, un coeur ouvert et une immense joie, que je dédie mon travail à mes très chère, respectueux et magnifiques parents qui m'ont soutenus tout au long de ma vie, dont leurs mérites, leurs sacrifices, leurs qualités humaines m'ont permis de vivre ce jour : les mots me manquent pour exprimer toute la reconnaissance, la fierté et le profond amour que je vous porte pour les sacrifices qu'ils ont consenti pour ma réussite, qu'ils trouvent ici le témoignage de mon attachement ma reconnaissance, gratitude et respect, que Dieu leur préservent bonne santé et longue vie. Tous mes sentiments de reconnaissance pour vous.

À mon frère «Yanis» et mes très chères soeurs «Fatima» et «Mounia»

J'espère atteint le seuil de vos espérances. Que ce travail soit l'expression de ma profonde affection. Je vous remercie pour le soutien moral et l'encouragement que vous m'avez accordés. Je vous souhaite tout le bonheur que vous méritez. En leur souhaitant un brillant avenir.

À mon binôme Lahcene.

À toute ma famille et à tous ceux que j'aime et qui m'aiment.

À mes ami(e)s

À tous ceux qui ont su m'apporter aide et soutien aux moments propices au long de mes études. Je dédie ce travail, reconnaissant et remerciant chaleureusement.

M.OUMAKHLOUF

Dédicaces

A mes parents

Je vous dois ce que je suis aujourd'hui grâce à votre amour, à votre patience et vos innombrables sacrifices. Que ce travail, soit pour vous une petite compensation et reconnaissance envers ce que vous avez fait d'incroyable pour moi. Que DIEU, le tout puissant, vous préserve et vous procure santé et longue vie afin que je puisse à mon tour vous combler.

A mes très chères soeurs **Hanane** et **Nadjet** et **Imane**, je vous remercie pour le soutien moral et l'encouragement que vous m'avez accordés. Je vous souhaite tout le bonheur que vous méritez. En leur souhaitant un brillant avenir.

A mon binôme **Mokhtar**.

A mes chères amis **Billal**, **Fares**, **Redouan**, **Yanis**., et à tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste travail.

L.KEDJAR

Remerciements

Ce mémoire n'aurait pas pu être confectionné si DIEU le tout puissant nous a avait pas doté d'une santé physique et morale à chaque instant ; c'est pourquoi, nous le remercions à l'infini pour ce don inestimable dont il nous a gratifié.

Nous tenons bien entendu à remercier particulièrement, notre cher encadreur, en l'occurrence **Mr : AISSANI Sofiane**, de l'Université de BEJAIA qui, par son encadrement ses précieux conseils, sa patience, sa générosité et enfin sa disponibilité ont fait que notre oeuvre a été largement facilité : nous ne saurions l'oublier.

Nos remerciements s'adressent également à tous les enseignants qui ont participé à notre formation tout le long de notre cursus.

Nous ne pouvons pas oublier de remercier tous les membres de nos familles pour leurs soutiens et leurs encouragements et particulièrement nos très chers parents.

Merci à tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce travail.

Table des matières

Table des figures	iv
Liste des tableaux	v
Liste des abréviations	vi
Introduction générale	1
1 Applications mobiles	2
1.1 Introduction	2
1.2 Application mobiles	2
1.2.1 Définition	2
1.2.2 Objectifs	2
1.2.3 Supports et outils de la mobilité	3
1.2.4 Types d'application mobiles	4
1.3 Systèmes d'exploitation mobiles	5
1.3.1 Caractéristique des systèmes d'exploitation mobiles	5
1.3.2 Types systèmes d'exploitation mobiles	5
1.4 Présentation d'Android	6
1.4.1 Le choix d'Android	6
1.4.2 Les avantages de la plateforme Android	7
1.4.3 Cycle de vie d'une activité sous Android	8
1.5 Conclusion	9
2 Etude de l'existant	10
2.1 Introduction	10
2.2 Définitions	10
2.2.1 Messagerie Instantanée (MI)	10
2.2.2 Chat «IRC : Internet Relay Chat»	11
2.2.3 Groupe de discussion	11
2.2.4 Forum	11
2.3 Fonctionnement de la MI	11
2.4 Fonctionnalités de la messagerie instantanée (MI)	11
2.4.1 Connexion et disponibilité	12
2.4.2 Gestion d'une liste de contacts	12
2.4.3 Discussion	12
2.4.4 Conférence	13
2.5 Analyse de l'existant	13

2.5.1	Présentation de l'application WhatsApp pour Android	13
2.5.2	Présentation de l'application Facebook Messenger pour Android . .	14
2.5.3	Présentation de l'application Skype-Android	15
2.5.4	Analyse concurrentielle	16
2.6	Conclusion	17
3	Conception	18
3.1	Introduction	18
3.2	Spécification des besoins	18
3.2.1	Les besoins attendus de l'application	18
3.2.2	Présentation de l'approche UP	19
3.3	Identification des acteurs	22
3.4	Diagramme de contexte du système à réaliser	22
3.5	Les Diagrammes des cas d'utilisation	23
3.5.1	Le diagramme global des cas d'utilisations	23
3.5.2	Les différents cas d'utilisation	24
3.5.3	Le cas d'utilisation «Authentification»	25
3.5.4	Le cas d'utilisation «Ajouter un ami »	26
3.5.5	Le cas d'utilisation «Inscription / Enregistrement»	27
3.5.6	Le cas d'utilisation «Déconnexion »	28
3.5.7	Le cas d'utilisation «Envoyer des messages»	29
3.6	Diagrammes de séquence	30
3.6.1	Définition d'un diagramme de séquence	30
3.6.2	Diagramme de séquence du cas d'utilisation «Inscription»	31
3.6.3	Diagramme de séquence du cas d'utilisation «Authentification » . .	32
3.6.4	Diagramme de séquence du cas d'utilisation «Ajout des amis» . . .	33
3.6.5	Diagramme de séquence du cas d'utilisation «Envoyer des messages».	34
3.6.6	Diagramme de séquence du cas d'utilisation «Déconnexion»	35
3.7	Diagramme de classe	36
3.7.1	Définition	36
3.7.2	Diagramme de classe de l'application	37
3.7.3	Règle de dérivation du modèle relationnel à partir d'un modèle de classes	37
3.8	Conclusion	39
4	Réalisation	40
4.1	Introduction	40
4.2	Outils de développement	40
4.2.1	Eclipse	40
4.2.2	Langage Java	41
4.2.3	Langage XML	41
4.2.4	JSON	42
4.2.5	SDK	42
4.2.6	Android développement tools (ADT)	43
4.2.7	Android virtual device (AVD)	44
4.2.8	Wamp server (Mysql serveur)	45
4.2.9	Langage PHP	46
4.2.10	Notepad++	46
4.3	Présentation des interfaces de l'application	47
4.3.1	Interface d'accueil	47

4.3.2	Interface d'inscription	48
4.3.3	Interface de la Liste d'amis	49
4.3.4	Menu «Ajouter un nouveau ami / Déconnexion»	50
4.3.5	Interface d'ajout des amis	51
4.3.6	Barre des notifications	52
4.3.7	Interface de confirmation des invitations	53
4.3.8	Interface de la Liste des amis en ligne et hors ligne	54
4.3.9	Interface de discussion	55
4.4	Conclusion	55
	Conclusion et Perspectives	56

Table des figures

1.1	<i>Cycle de vie d'une activité sous Android. [20]</i>	8
2.1	<i>La page d'accueil de l'application WhatsApp.</i>	14
2.2	<i>Page d'accueil de l'application Facebook Messenger.</i>	15
2.3	<i>Page d'accueil de l'application Skype-Android.</i>	16
3.1	<i>Le processus 2Up. [9]</i>	20
3.2	<i>les diagrammes UML. [7]</i>	21
3.3	<i>Diagramme de contexte du système à réaliser.</i>	22
3.4	<i>Diagramme global des cas d'utilisations associés aux acteurs.</i>	23
3.5	<i>Diagramme de séquence du cas d'utilisation «Inscription».</i>	31
3.6	<i>Diagramme de séquence du cas d'utilisation «Authentification».</i>	32
3.7	<i>Diagramme de séquence du cas d'utilisation «Ajout des amis».</i>	33
3.8	<i>Diagramme de séquence du cas d'utilisation «envoyer des messages».</i>	34
3.9	<i>Diagramme de séquence du cas d'utilisation «Déconnexion».</i>	35
3.10	<i>Diagramme de classe.</i>	37
3.11	<i>Passage d'une classe a une relation.</i>	37
3.12	<i>Passage un a plusieurs.</i>	38
4.1	<i>Environnement d'Eclipse (ADT).</i>	41
4.2	<i>Android SDK Manager.</i>	43
4.3	<i>La Barre d'outils d'Eclipse.</i>	44
4.4	<i>Émulateur Android.</i>	45
4.5	<i>Interface d'identification.</i>	47
4.6	<i>Interface d'inscription.</i>	48
4.7	<i>Interface de la Liste d'amis.</i>	49
4.8	<i>Menu «Ajouter un nouveau ami / Déconnexion».</i>	50
4.9	<i>Interface d'ajout des amis.</i>	51
4.10	<i>Barre des notifications.</i>	52
4.11	<i>Interface de confirmation des invitations.</i>	53
4.12	<i>Interface de la Liste des amis en ligne et hors ligne.</i>	54
4.13	<i>Interface de discussions.</i>	55

Liste des tableaux

3.1	<i>Le formalisme de description des cas d'utilisation.</i>	24
3.2	<i>Description du cas d'utilisation «Authentification».</i>	25
3.3	<i>Diagramme du cas d'utilisation «Authentification».</i>	25
3.4	<i>Description du cas d'utilisation «Ajouter un ami».</i>	26
3.5	<i>Diagramme du cas d'utilisation «Ajouter un ami».</i>	26
3.6	<i>Description du cas d'utilisation «Inscription / Enregistrement».</i>	27
3.7	<i>Diagramme du cas d'utilisation «Inscription».</i>	27
3.8	<i>Description du cas d'utilisation «Déconnexion».</i>	28
3.9	<i>Diagramme du cas d'utilisation «Déconnexion».</i>	28
3.10	<i>Description du cas d'utilisation «Envoyer des messages».</i>	29
3.11	<i>Diagramme du cas d'utilisation «Envoyer des messages».</i>	29

Liste des abréviations

AIM	Aol Instant Messenger.
AMR	Adaptive Multi Rate.
API	Application Programming Interface.
AVD	Android Virtual Device.
A2DP	Advanced Audio Distribution Profile
BSD	Berkeley Software Distribution.
CSS	Cascading Style Sheets.
GPS	Global Positioning System.
HTML	Hyper Text Markup Language.
IBM	International Business Machines.
IDE	Integrated Development Environment
IOS	Iphone Operating System.
IP	Internet Protocol.
IRC	Internet Relay Chat.
JNI	Java Native Interface.
JSON	Java Script Object Notation.
JSP	Java Server Pages.
MIT	Massachusetts Institute of Technology.
OS	Operating System.
PDA	Personnel Digital Assistant.
PHP	Hypertext Preprocessor.
SDK	Software Development Kit
SGBD	Système de Gestion de Base de Données.
SGML	Standard Generalized Markup Language.
SMS	Short Message Service.
SQL	Structured Query Language.
SWT	Standard Widget Toolkit.
UIQ	User Interface Quartz.
UML	Unified Modeling Language .
UP	Unified Process.
USB	Universal Serial Bus.
VM	Virtual Machine.
XML	eXtended Markup Language.

Introduction générale

Aujourd'hui, le monde connaît une avancée considérable dans l'utilisation des appareils téléphoniques portables (mobiles) grâce aux applications mobiles, ces dernières sont capables de satisfaire les besoins actuels des utilisateurs avec de nombreuses fonctionnalités et en offrant plusieurs services.

De nouveaux usages sont apparus tels que les jeux, la lecture audio, etc sur un plan plus pratique, le téléphone mobile dispose des variétés d'utilisations et l'appareil est devenu plus utile que jamais. L'essor du mobile a connu une évolution considérable. Les réseaux de télécommunication mobiles sont également en expansion : le nombre d'utilisateurs de mobile est en constante progression et la couverture territoriale est largement répandue.

Notre projet s'inscrit dans un cadre général du développement et de la réalisation d'une application de chat « messagerie instantanée » qui fonctionne sous Android qui permet à plusieurs utilisateurs d'échanger des messages textes et les sauvegarder dans une base de données.

Notre mémoire est structuré en quatre chapitres, le premier chapitre porte sur les applications mobiles et leurs objectifs, ensuite, on va citer les différents supports et outils de la mobilité, les types d'application mobiles et les différents systèmes d'exploitation et une présentation détaillée du système Android.

Le deuxième chapitre sera dédié à l'étude de l'existant, on parlera de la messagerie instantanée, son fonctionnement, ensuite, on va citer les différents protocoles de communications ainsi leurs avantages et leurs défauts et les fonctionnalités de la messagerie, et enfin nous présenterons l'analyse d'exemples d'applications de messagerie sous ANDROID

Le troisième chapitre sera consacré à la conception qui est une étude préalable de notre projet, elle est une étape primordiale et un passage obligatoire, car il s'agit de décrire les besoins de notre système, on modélisant avec le langage UML.

La réalisation de notre application fera l'objet du chapitre quatre, ou nous allons présenter les outils ainsi que les langages de programmations utilisés, on va expliquer le fonctionnement de notre application en utilisant quelques captures d'écran.

Nous concluons notre mémoire par exposer l'ensemble des connaissances acquises au cours de la réalisation de notre projet, et nous exposerons quelques perspectives .

Applications mobiles

1.1 Introduction

Aujourd'hui, les applications mobiles ont pris une place importante dans notre vie quotidienne, et prennent de plus en plus d'espace dans l'utilisation de nos terminaux mobiles, elles sont conçues pour des plateformes mobiles et utilisées pour des services de l'information, medias sociaux, jeux etc.

Dans ce chapitre nous abordons la notion de l'application mobile et ces objectifs, ses types, les différents systèmes d'exploitation mobiles, enfin nous allons faire une présentation détaillée sur le système Android.

1.2 Application mobiles

Les applications mobiles sont apparues dans les années 1990, elles sont liées aux développements d'Internet et des télécommunications, des réseaux sans fils et des technologies agents, et à l'apparition et la démocratisation des terminaux mobiles : smartphones, tablettes ...etc.

1.2.1 Définition

Une application mobile est un logiciel applicatif développé pour un appareil électronique mobile, tel qu'un assistant personnel, un téléphone portable, un smartphone, un baladeur numérique, une tablette tactile, ou encore certains ordinateurs fonctionnant avec le système d'exploitation Windows Phone.

1.2.2 Objectifs

Elles visaient d'abord la productivité et à faciliter la récupération d'informations telles que courrier électronique, calendrier électronique, contacts, marché boursier et informa-

tions météorologiques. la demande du public et la disponibilité d'outils de développement ont conduit à une expansion rapide dans d'autres domaines, comme : les jeux mobiles, les automatismes industriels, le GPS et les services basés sur la localisation, les opérations bancaires, les suivis des commandes, l'achat de billets, des applications médicales mobiles, la réalité virtuelle.[1]

1.2.3 Supports et outils de la mobilité

Les supports du multimédia mobile peuvent être regroupés en 4 grandes catégories :

1.2.3.1 Terminaux mobiles

De part sa diversité, l'univers de terminaux est sans doute le révélateur de la dynamique du monde mobile. Jusqu'à une époque récente, on connaissait comme seuls objets communicants les téléphones et les ordinateurs. Mais de plus en plus, la miniaturisation des puces, des mémoires, des antennes et des batteries permet aujourd'hui d'ajouter des capacités d'acquisition, de traitement, de stockage, de communication et même d'action à un nombre croissant d'objets rendus intelligents . Nous nous limitons ici à un inventaire sommaire des outils de mobilité les plus populaires qui permettent à nombre d'utilisateurs d'avoir une connexion de type multimédia mobile.[1]

- **PC portable et Tablet PC (e-books)** : Le Pocket PC de Microsoft est l'exemple type de l'ordinateur de poche. Doté d'un système d'exploitation, Un Pocket PC rappelle un PC avec son menu Démarrer et la suite Pocket Office : Pocket Internet Explorer, Outlook, Pocket Word et Pocket Excel. Le Tablet PC, lui est un ultra-portable avec toutefois des caractéristiques spéciales : son écran est tactile, il reconnaît votre écriture grâce à un stylet, et la transforme en texte. L'unique charnière centrale de son écran permet à ce dernier de pivoter et se rabattre sur le clavier.[1]

1.2.3.2 Assistants numériques (PDA)

Le PDA (Personnel Digital Assistant) ou assistant numérique personnel est un ordinateur de poche faisant office d'assistant personnel. Un assistant numérique personnel - ou PDA - est à l'origine un agenda électronique destiné à la prise de rendez-vous, à la planification des tâches et au transport de données personnelles. Il s'est, depuis, beaucoup enrichi en fonctionnalités communicantes (push mail, carte 3G, connexions Bluetooth, etc.). [1]

1.2.3.3 Smartphone

Les Smartphones sont des téléphones dits intelligents qui assurent en priorité les fonctions de communication et de navigation sur internet. Ces terminaux proposés à la fois par les constructeurs informatiques et les équipementiers télécoms associent les fonctions des assistants personnels et celles des téléphones mobiles. Un utilisateur de Smartphone consomme deux fois plus de voix, et dix fois plus de data, c'est-à-dire de données (mail, accès Internet). [1]

1.2.3.4 Terminaux converges

Il y a une convergence des PDA, Pocket PC, smartphone et autres produits, car tous les professionnels sont demandeurs d'un produit simple rassemblant toutes les applications mobiles avec Bluetooth, Wi-Fi, etc. Ces terminaux offrent des services voix ou données et sont capables de synchroniser des informations personnelles et/ou des courriels avec des serveurs, des postes de travail ou des ordinateurs portables. Ils sont dotés d'un système d'exploitation évolué comme Palm OS, Windows Mobile 5.0, ainsi que la plateforme Symbian.[1]

1.2.4 Types d'application mobiles

Actuellement, il existe 3 principaux types d'applications mobiles :

1.2.4.1 Les Applications natives (Natives App)

Celles-ci correspondent à des logiciels créés uniquement pour une plateforme mobile. Le développement de ces logiciels se fait au travers du SDK ou software développement kit de la plateforme mobile choisie. Le nom de ces applications vient du fait qu'elles sont développées avec l'utilisation de langages natifs comme par exemple le langage JAVA ou le langage Objective-C. Les natives App sont téléchargées à partir d'une plateforme de téléchargement qui est souvent un App Store. [18]

1.2.4.2 Les applications web (Web App)

Ces applications correspondent à des sites web qui sont conçus spécialement pour un affichage sur mobile optimisé. Pour accéder à ces sites web, on utilise le navigateur internet disponible sur le mobile. Ces applications mobiles sont développées principalement à partir de technologies web comme HTML5 ou encore CSS3. Grâce au support HTML5 il est possible d'accéder à environ 80% des fonctions présentes sur le mobile. Par exemple cela permet d'accéder à la géolocalisation, à l'accéléromètre, gérer la fonction multitouche ou

encore permettre la synchronisation offline lorsque le mobile perd et retrouve sa connexion. [18]

1.2.4.3 Les applications de type hybride ou Hybrid App

Ces applications sont considérées comme un mix à la fois entre les applications natives et les applications web. En effet, elles sont compatibles avec toutes les plateformes mobiles. Ces applications sont principalement développées à l'aide d'HTML5 aujourd'hui qui est très performant mais utilisent aussi d'autres langages web comme CSS et JavaScript. [18]

1.3 Systèmes d'exploitation mobiles

Un système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface.[19]

1.3.1 Caractéristique des systèmes d'exploitation mobiles

Un système d'exploitation mobile regroupe un ensemble des fonctionnalités dont : [19]

- La gestion de la mémoire.
- La gestion des microprocesseurs et l'ordonnancement.
- La gestion de système de fichiers.
- La gestion des Entrées/Sorties.
- La gestion de sécurité.
- La gestion de fonctionnalités multimédia.

1.3.2 Types systèmes d'exploitation mobiles

Les principaux systèmes d'exploitations mobile sont : Android, Bada, BlackBerry OS, iOS, OpenMoko, Palm OS, HP webOS, Symbian OS, Windows CE, Windows Mobile, Windows Phone 7. [19]

- **Android** : est un système d'exploitation open source utilisant le noyau Linux, pour smartphones, PDA et terminaux mobiles conçu par Android, une startup rachetée par Google, et annoncé officiellement le 5 novembre 2007. D'autres types d'appareils possédant ce système d'exploitation existent, par exemple des téléviseurs et des tablettes.

- **Bada** : est le système d'exploitation pour smartphone haut de gamme et milieu de gamme de Samsung basé sur le système d'exploitation propriétaire SHP OS. Tous les téléphones sous Bada ont un nom commençant par Samsung Wave.
- **BlackBerry OS** : est un système d'exploitation qui fonctionne sur le smartphone BlackBerry. Il permet aux développeurs de mettre en place des applications en utilisant les APIs BlackBerry, mais toute application doit être signée numériquement par le compte RIM du développeur.
- **Symbian OS** : est un système d'exploitation le plus utilisé pour des smartphones et PDA, qui a été conçu par Symbian Ltd. Symbian OS fournit les fonctionnalités essentielles du système d'exploitation, notamment le coeur du système, ainsi que les API communes et une interface utilisateur de référence. Il a été adopté par différents fabricants de téléphones portables de 2G et 3G (Nokia, Sony Ericsson, Motorola, Samsung, etc.). Les principales interfaces utilisateur sont S60 série (pour clavier numérique), UIQ (pour écran tactile), S80 série (pour clavier alphanumérique).

1.4 Présentation d'Android

Il existe plusieurs types de système d'exploitation pour le téléphone mobile, chaque système a ses avantages et ses limites. Il est donc très difficile de choisir la plateforme répondant bien à l'objectif d'une société ou d'une personne.

Dans cette partie nous allons présenter la raison du choix d'Android, nous aurons aussi à présenter ce système et les outils pour créer une application dans la pratique.

1.4.1 Le choix d'Android

Android est un système d'exploitation open Source pour smartphones, PDA et autres terminaux mobiles, conçu par Android, une start-up rachetée par Google en juillet 2005. Il est donc gratuit et librement modifiable, ce qui explique d'ailleurs le nombre de mobiles qui l'utilise et ce indépendamment de leur fabricant. Cet aspect lui permet également d'être présent sur d'autres types d'appareils possédant ce système d'exploitation tels que les téléviseurs et les tablettes. Ceci peut d'ailleurs constituer une opportunité de se détacher du seul cadre du téléphone mobile. [2]

Le noyau Linux lui fournit une grande mémoire, la gestion de processus, le modèle de sécurité, le soutien des bibliothèques partagées...etc. Le SDK de l'Android offre complètement

les APIs, avec un accès facile pour développer l'application.[2]

- Une API, ou interface de programmation en français, est un ensemble de règles à suivre pour pouvoir dialoguer avec d'autres applications. Dans le cas de Google API, il permet en particulier de communiquer avec Google Maps.

Android est un système d'exploitation puissant et moderne, qui se caractérise par la simplicité et la flexibilité ; cela signifie que le système est développé avec un simple langage java, et il s'adapte à beaucoup de structures différentes.[2]

1.4.2 Les avantages de la plateforme Android

- + Pas de License à obtenir, pas de dépense pour la distribution et le développement.
- + Développer des applications location-based en utilisant le GPS.
- + Utiliser des cartes géographiques avec Google Maps.
- + Recevoir et émettre des SMS, envoyer et recevoir des données sur le réseau mobile.
- + Enregistrer et lire image, son et vidéo.
- + Des outils de stockage de données partagés (SQLite en version Sandbox).
- + Une accélération matérielle pour la 2D et la 3D.
- + Des services et des applications qui peuvent tourner en tâche de fond : qui peuvent réagir au cours d'une action, à votre position dans la ville, à l'heure qu'il est, suivant l'identité de l'appelant.
- + Une plateforme de développement qui favorise la réutilisation de composants logiciels et le remplacement des applications fournies.[3]

1.4.3 Cycle de vie d'une activité sous Android

Voici un diagramme qui explique le cycle de vie d'une activité sous Android :[20]

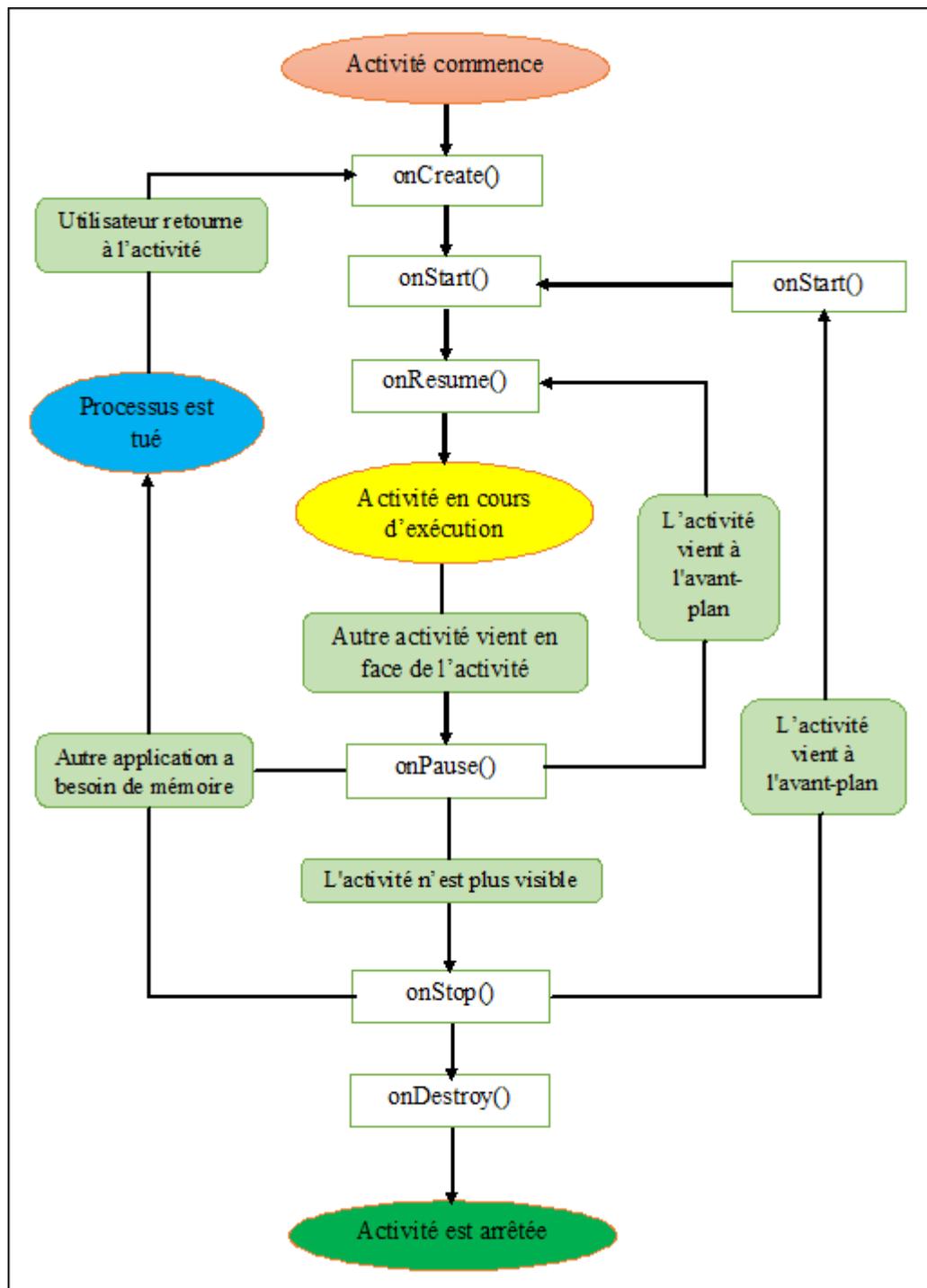


FIGURE 1.1 – Cycle de vie d'une activité sous Android. [20]

OnCreate

Cette méthode est appelée à la création de l'activité (Activity). Elle sert à initialiser votre activité ainsi que toutes les données nécessaires à cette dernière. Quand la méthode OnCreate est appelée, on lui passe un Bundle en argument. Ce Bundle contient l'état de sauvegarde enregistré lors de la dernière exécution de votre activité.

OnStart

Cette méthode est appelée dans le cas où votre application est en arrière-plan et qu'elle repasse en avant-plan. Si votre activité ne peut pas aller en avant-plan quelle que soit la raison, l'activité sera transférée à OnStop.

OnResume

Cette méthode est appelée après OnStart au moment où l'application repasse en avant-plan ou en arrière-plan à cause d'une autre application.

OnPause

Appelée juste avant qu'une autre activité passe en OnResume. À ce stade, notre activité n'a plus accès à l'écran, vous devez arrêter de faire toute action en rapport avec l'interaction utilisateur. Vous pouvez par contre continuer à exécuter des algorithmes nécessaires mais qui ne consomment pas trop de CPU.

OnStop

Appelée quand l'activité n'est plus visible quelle que soit la raison.

OnDestroy

Appelée quand l'application est totalement fermée (Processus terminé).

1.5 Conclusion

Dans ce chapitre, nous avons donné un petit aperçu sur les applications mobiles, ainsi que ses différents objectifs. Et puis on a présenté les différents supports et outils de la mobilité et on a énuméré les différents types d'application mobile, puis on a défini les systèmes d'exploitation mobiles ainsi leurs caractéristiques et leurs types. À la fin on a fait une Présentation d'Android dont sa description, ses avantages, ses fonctionnalités et son Architecture.

Etude de l'existant

2.1 Introduction

Il existe plusieurs types de messagerie instantanée dont chacune d'elle présente ses propres avantages par rapport aux autres selon le domaine d'utilisation.

Nous présenterons dans ce chapitre, l'étude de l'existant qui doit être élaborer avant d'entamer la spécification des besoins et la conception de notre application, et pour cela nous allons donner une présentation de la messagerie instantanée, son fonctionnement et ses fonctionnalités, enfin nous présenteons l'analyse d'exemples d'applications de messagerie sous Android.

2.2 Définitions

2.2.1 Messagerie Instantanée (MI)

La messagerie instantanée, le dialogue en ligne (désignée par chat, en anglais et en français tchat « bavardage »), permet l'échange instantané de messages textuels et de fichiers entre plusieurs personnes par l'intermédiaire d'ordinateurs connectés au même réseau informatique, et plus communément celui d'Internet. Contrairement au courrier électronique, ce moyen de communication permet de conduire un dialogue interactif. [14]

C'est une méthode de communication bidirectionnelle qui est à mi-chemin entre l'e-mail (pour son caractère écrit) et le téléphone (pour ses échanges en temps réel) en se créant son propre réseau de correspondants. Ce contact pourrait s'apparenter à une partie de « ping-pong écrit ». A la différence du chat, il s'agit de dialoguer avec des correspondants faisant partie d'une liste privée dans laquelle sont intégrées uniquement les personnes souhaitées. [14]

2.2.2 Chat «IRC : Internet Relay Chat»

C'est un moyen de communiquer en direct avec des groupes de personnes via l'Internet. Les clients se connectent à des serveurs qui sont eux-mêmes reliés entre eux, formant un réseau IRC. Tous les clients sont donc susceptibles de communiquer entre eux en temps réel à travers le réseau. [14]

2.2.3 Groupe de discussion

Technologie permettant d'échanger des idées et des informations sur des sujets généralistes ou précis par mails interposés, les groupes de discussion utilisent le réseau Usenet (un réseau comparable à Internet, mais dédié à l'échange de messages des groupes de discussion) Comme les messages sont reçus directement sur les ordinateurs des internautes, il faut disposer d'un logiciel client approprié (appelé aussi lecteur de news), Outlook Express sait parfaitement bien gérer les groupes de discussion. [14]

2.2.4 Forum

On communique en passant par un serveur sur Internet qui centralise les messages, on n'utilise donc pas son logiciel de messagerie, ni le réseau Usenet. [14]

2.3 Fonctionnement de la MI

La messagerie instantanée est donc un moyen de communiquer en privé avec d'autres personnes de son choix. Le client se connecte à un serveur qui contient les informations sur tous les utilisateurs inscrits, connectés ou non. Chaque personne possède un pseudonyme qui n'est pas forcément unique, et un identifiant unique dans la base de données du serveur. Un mot de passe est aussi indispensable pour se connecter au service. Deux personnes peuvent communiquer en direct si elles sont simultanément connectées au serveur. Sinon, elles ont la possibilité de consulter leurs messages dans leur boîte aux lettres au moment où elles se connectent. [14]

2.4 Fonctionnalités de la messagerie instantanée (MI)

Les fonctionnalités offertes par les services de messagerie instantanée sont décrites ci-après.

2.4.1 Connexion et disponibilité

Le serveur d'identification est interconnecté avec une base de données dans laquelle les utilisateurs seront déclarés et les différents statuts possibles sont : [4]

- **Déconnecté** : Le programme est déconnecté du serveur si bien que vous ne pouvez pas savoir si vos contacts sont présents, eux vous voient comme déconnecté.

- **Connecté (et disponible)** : Vous êtes donc connecté et disponible pour envoyer et recevoir des messages de vos contacts.

- **Absent** : En choisissant ce statut, vous indiquez à tous vos contacts que vous êtes absent.

- **Non disponible** : Signifie que vous n'êtes pas disponible pour discuter et que vous préféreriez donc ne pas être contacté.

- **Non visible** : Vous êtes occupé et demandez donc à ne pas être dérangé.

2.4.2 Gestion d'une liste de contacts

Les programmes de messagerie vous permettent de gérer une liste de contact. A la différence des chats sur Internet, lorsque vous vous connectez sur un serveur, vous ne récupérez pas automatiquement la liste des autres personnes connectées. Heureusement d'ailleurs car n'oubliez pas que certaines messageries publiques regroupent plusieurs millions d'utilisateurs simultanés.

C'est donc à chacun de se constituer sa propre liste de contacts. Pour cela, il suffit de rechercher dans l'annuaire du serveur les personnes que l'on désire avoir comme contact de façon à communiquer avec elles. [4]

2.4.3 Discussion

Une discussion est un échange de messages entre deux interlocuteurs, à savoir vous et un de vos contacts. Ces messages sont de taille très variable et ils sont habituellement limités par les systèmes à quelques centaines de caractères. Mais rien ne empêche l'utilisateur de découper ce que vous avez à dire sur plusieurs messages. [4]

2.4.4 Conférence

À la différence d'une discussion à plusieurs, dans une conférence, tous les participants voient ce que disent les autres. Chaque message tapé et envoyé une seule fois par utilisateur vers le serveur qui se charge de le copier et de l'envoyer à tous les participants de la conférence. De la même façon, les réponses sont dupliquées et envoyées à tous. [4]

2.5 Analyse de l'existant

Dans cette section, nous présentons les applications WhatsApp Messenger et facebook Messenger et skype Android. Ensuite nous procédons à une analyse concurrentielle.

2.5.1 Présentation de l'application WhatsApp pour Android

WhatsApp Messenger est une application de messagerie disponible pour Android et autres Smartphones. WhatsApp utilise la connexion d'internet de votre téléphone (4G/3G/2G/EDGE ou WiFi, si disponible) pour envoyer le message à ou appeler vos amis et votre famille. Passez du SMS à WhatsApp pour envoyer et recevoir des messages, des appels, des photos, des vidéo, et des Messages Vocaux, utilisé par 600 millions de personnes.

L'entreprise WhatsApp fondée en 2009 par Jan Koum et Brian Acton, deux anciens ingénieurs de Yahoo, a pour but de remplacer le SMS. Ses bureaux sont basés à Mountain View en Californie. [21]



FIGURE 2.1 – La page d'accueil de l'application WhatsApp.

2.5.2 Présentation de l'application Facebook Messenger pour Android

C'est une application développée essentiellement par facebook pour permettre aux membres du réseau social d'accéder et consultés leurs comptes et de dialoguer avec leurs amis sans être forcément connectés au site web.

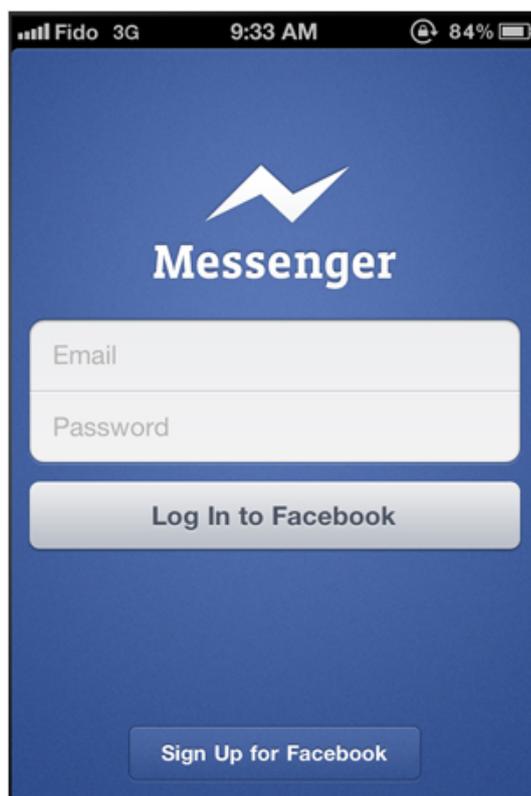


FIGURE 2.2 – Page d'accueil de l'application Facebook Messenger.

2.5.3 Présentation de l'application Skype-Android

C'est une application qui fait bénéficier aux utilisateurs d'Android d'une communication permanente avec leurs différents contacts. Facile à manipuler, elle permet d'envoyer des messages instantanés en toute liberté.

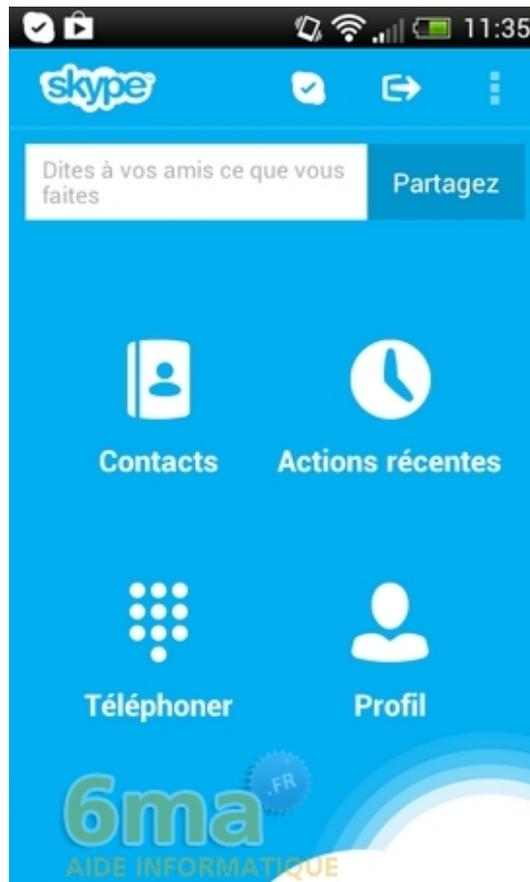


FIGURE 2.3 – Page d'accueil de l'application Skype-Android.

2.5.4 Analyse concurrentielle

Toutes ces applications sont destinées aux utilisateurs disposants d'appareils téléphoniques intelligents, elles offrent plusieurs services : envoi et réception gratuite de messages instantanés, échange de photos, vidéos, et messages vocaux dont chacune à ces propres avantages par rapport aux autres.

Whatsapp :

- Elle se synchronise avec la liste de contact et identifie les personnes disposant du logiciel dans leurs applications mobile.
- Contrairement à d'autres applications de messagerie instantanée qui permettent aux utilisateurs de communiquer exclusivement entre des smartphone fonctionnant sous le même système d'exploitation, WhatsApp corrige ce problème.
- Pas besoin de créer un nom d'utilisateur et mot de passe Messenger car WhatsApp fonctionne avec le numéro de téléphone.
- Elle Offre aussi la possibilité de partager la position actuelle avec les contacts.

Facebook Messenger :

- Enregistrer des vidéos et prendre des selfies ou d'autres photos directement dans l'application, et envoyer-les d'un seul geste.
- Les bulles de discussion permettent aux utilisateurs de continuer leurs conversations tout en utilisant d'autres applications.
- Animation et enrichissement des conversations avec des autocollants.

skype-Android :

- Visiophonie et appels audio sous 3G ou WiFi.
- Appels économiques vers des lignes fixes et mobiles, et SMS à des tarifs avantageux.
- Envoi de fichiers de toutes tailles.

2.6 Conclusion

Dans ce chapitre, on a présenté quelques définitions sur la messagerie instantanée (MI), son évolution et son fonctionnement, enfin on a énumérée ces différentes fonctionnalités offertes qui seront utiles dans le prochain chapitre. Enfin on a fait une présentation de quelques applications de messagerie instantanée sous Android qui existe, suit à une analyse concurrentielle dont on a cité les avantages de chacune par rapport aux autres.

Conception

3.1 Introduction

La phase de conception est la première étape dans la réalisation d'un projet, elle doit d'écrire de manière non ambiguë le fonctionnement futur du système, afin d'en faciliter la réalisation. Pour cela, différentes méthodes existent permettant de formaliser les étapes préliminaires du développement.

Dans ce chapitre, nous présentons les objectifs de notre application, ce qui nous amène à identifier les possibilités du système et les besoins des utilisateurs que nous essayons de projeter dans des diagrammes de cas d'utilisations globaux et détaillés.

3.2 Spécification des besoins

3.2.1 Les besoins attendus de l'application

L'application envisagée doit satisfaire les besoins fonctionnels qui seront exécutés par le système et les besoins non fonctionnels qui perfectionnent la qualité logicielle du système.

3.2.1.1 Les besoins fonctionnels

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter, il ne devient opérationnel que s'il les satisfait. Cette application doit couvrir principalement les besoins fonctionnels suivants :

- Enregistrement des utilisateurs.
- Authentification des utilisateurs.
- Envoyer et recevoir des messages textes.
- Exécute un service de fond afin d'obtenir des messages, même lorsque l'application est fermée.

- Utiliser une zone de notification lorsqu'un nouveau message ou une nouvelle invitation est reçue.
- Gestion d'amis (Ajout des amis par nom d'utilisateur, affichage de la liste des amis avec leurs statut(en ligne/hors ligne),accépter ou refuser des invitations.
- Se déconnecter (quitter l'application).

3.2.1.2 Les besoins non fonctionnels

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système. Les principaux besoins non fonctionnels de notre application se résument comme suit :

- Le code doit être clair pour permettre des futures évolutions ou améliorations.
- L'ergonomie : l'application offre une interface conviviale et facile à utiliser.
- La sécurité : l'application doit respecter la confidentialité des données.
- Garantir l'intégrité et la cohérence des données à chaque mise à jour et à chaque insertion.

3.2.2 Présentation de l'approche UP

3.2.3.1 Processus et architecture

Un processus définit une séquence d'étapes, en partie ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. [8]

Le processus unifié (Unified Process)

Un processus unifié est un processus de développement logiciel construit sur UML ; il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques. [8]

L'objectif du Processus unifié est de guider les développeurs vers l'implémentation et le déploiement efficaces de systèmes répondant aux besoins des clients. [9]

Le processus 2TUP

2TUP signifie « 2 Track Unified Process ». C'est un processus UP qui répond aux caractéristiques que nous venons de citer. Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes. «2 Track » signifient littéralement que le processus suit deux chemins. Il s'agit

des chemins « fonctionnels » et « d'architecture technique », qui correspondent aux deux axes des changements imposés au système informatique.[9]

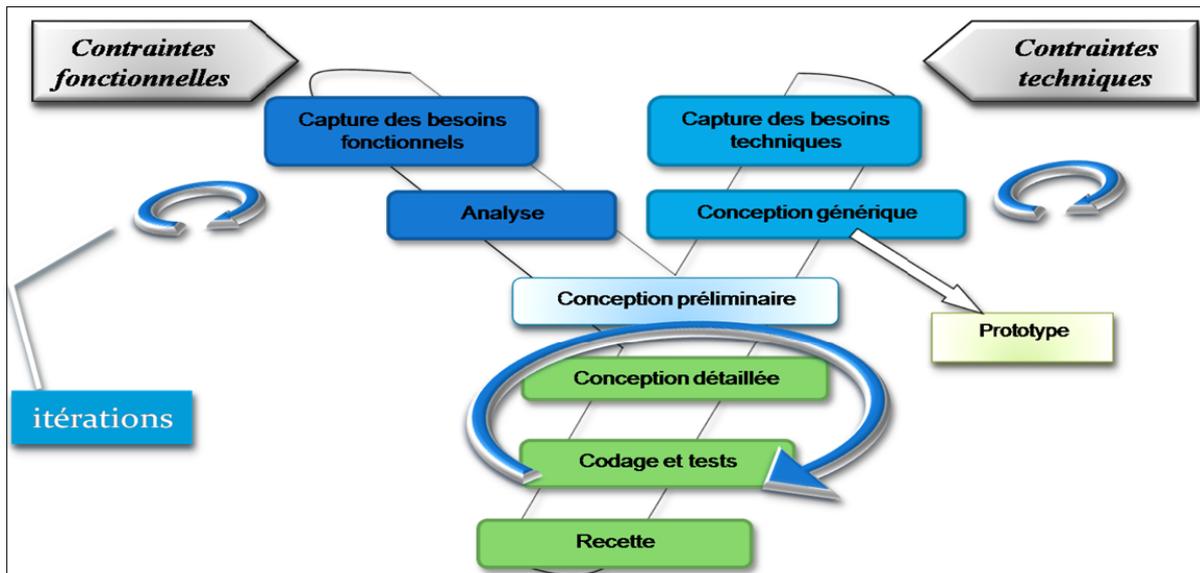


FIGURE 3.1 – *Le processus 2Up.* [9]

3.2.3.2 Le langage UML

UML (Unified Modeling Language), se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML modélise l'ensemble des données et des traitements en élaborant des différents diagrammes. [5]

En clair, il ne faut pas designer UML en tant que méthode (Il y manque la démarche) mais plutôt comme une boîte d'outils qui sert à améliorer les méthodes de travail.

3.2.3.3 L'intérêt d'UML

UML est un langage semi-formel et normalisé : [6]

- Gain de précision.
- Gagne de stabilité.
- Encourage l'utilisation d'outils.

UML est un support de communication performant :

- Il cadre l'analyse
- Il facilite la compréhension de représentations abstraites complexes.
- Son caractère polyvalent et sa souplesse en font un langage universel.

3.2.3.4 Définition d'un modèle

Un modèle est une vue subjective mais pertinente de la réalité. Un modèle définit une frontière entre la réalité et la perspective de l'observateur. Ce n'est pas "la réalité", mais une vue très subjective de la réalité. Bien qu'un modèle ne représente pas une réalité absolue, un modèle reflète des aspects importants de la réalité, il en donne donc une vue juste et pertinente. [6]

3.2.3.5 Les différents types diagrammes d'UML

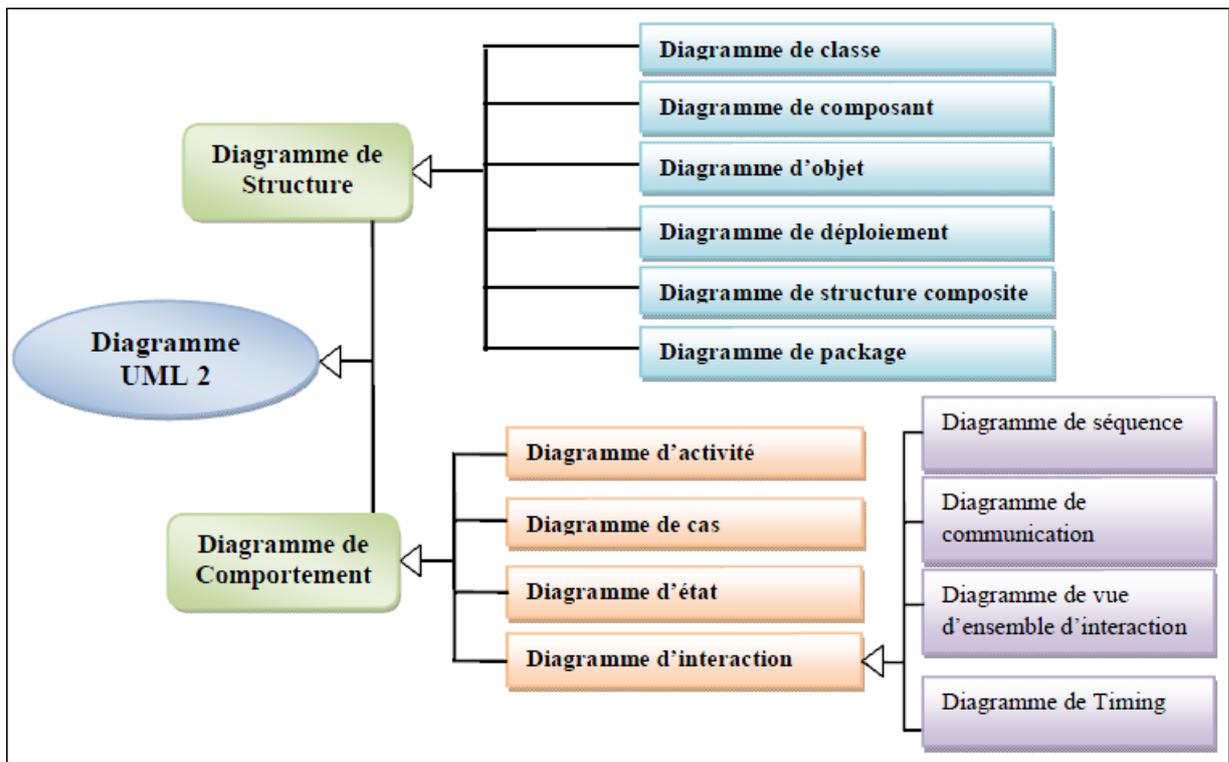


FIGURE 3.2 – les diagrammes UML. [7]

3.3 Identification des acteurs

Un acteur représente un rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.[10]

Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données.[11]

Dans notre cas, nous avons un acteur qui est :

Utilisateur : a un accès au système via un contrôle d'accès (login et mot de passe).

Les opérations qu'il peut effectuer sont :

- Envoyer des invitations par nom d'utilisateur.
- Afficher la liste des amis avec leurs statues (en ligne / hors ligne).
- Envoyer et recevoir des messages text.
- Accepter ou refuser des invitations.
- Se déconnecter.

3.4 Diagramme de contexte du système à réaliser

La figure ci-dessous montre l'interaction entre un acteur qui est l'utilisateur et le système que nous allons mettre en place :

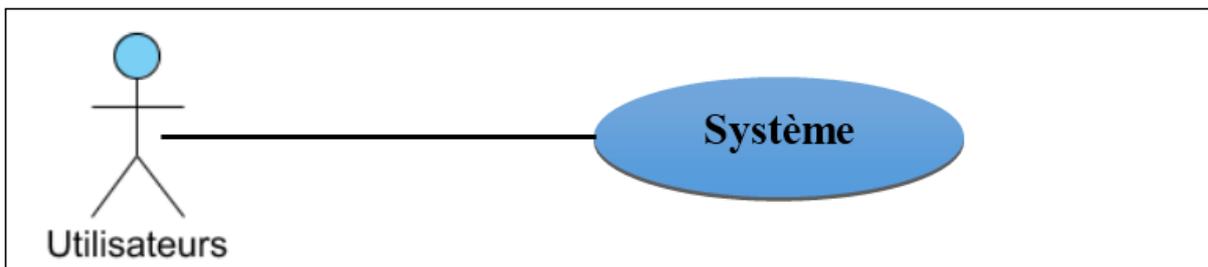


FIGURE 3.3 – *Diagramme de contexte du système à réaliser.*

3.5 Les Diagrammes des cas d'utilisation

Le diagramme de cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en oeuvre. [10]

3.5.1 Le diagramme global des cas d'utilisations

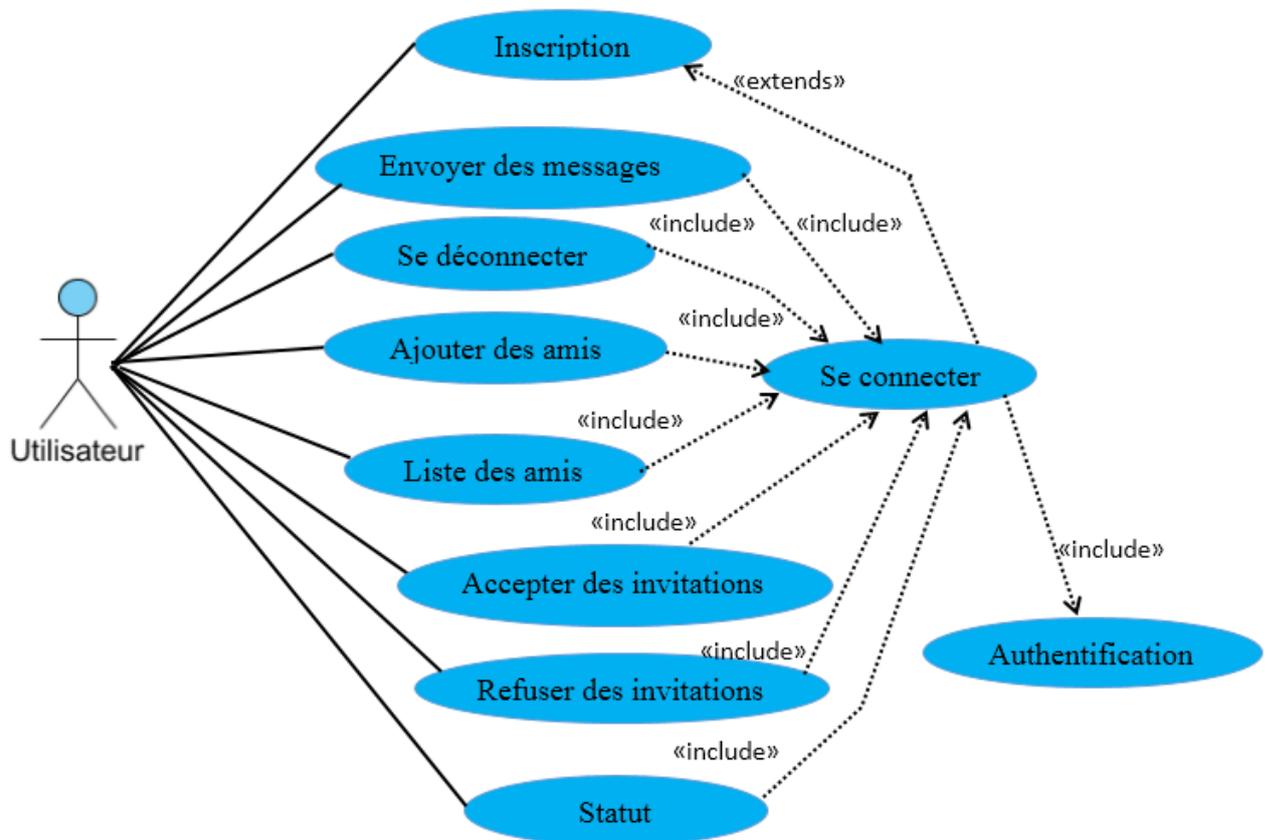


FIGURE 3.4 – Diagramme global des cas d'utilisations associés aux acteurs.

3.5.2 Les différents cas d'utilisation

L'étude de cas d'utilisation a pour objectif de déterminer ce que chaque utilisateur attend du système. La détermination du besoin est basée sur la représentation de l'interaction entre l'acteur et le système.

À chaque cas d'utilisation doit être associée une description textuelle des interactions entre l'acteur et le système et les actions que le système doit réaliser en vue de produire les résultats attendus par les acteurs. Pour exprimer les cas d'utilisations de notre système, nous avons choisi le formalisme suivant :

Numéro du cas d'utilisation	Nom du cas d'utilisation.
Résumé	But du cas d'utilisation.
Acteur	Acteurs participants au cas d'utilisation.
Précondition	Condition qui doit être remplie avant le début du cas d'utilisation.
Scénario nominal	Séquence d'actions normales associées au cas d'utilisation.
Alternative	Séquence d'actions alternatives pouvant conduire également à un succès.
Exception	Séquences d'actions conduisant à un échec.

TABLE 3.1 – *Le formalisme de description des cas d'utilisation.*

3.5.3 Le cas d'utilisation «Authentification»

Cas d'utilisation N° 1	Authentification
Résumé	Vérification de l'identité des utilisateurs (Login et mot de passe).
Acteurs	Utilisateur.
Précondition	L'utilisateur doit avoir un compte.
Scénario nominal	<p>[début]</p> <ul style="list-style-type: none"> · Demande de connexion ; · Le système affiche le formulaire d'authentification (demande le nom d'utilisateur et le mot de passe ; · L'utilisateur saisit son login et son mot de passe puis valide ; · Le système vérifie la conformité des informations fournies A1 ; · Le système donne l'accès à l'interface correspondante ; <p>[fin]</p>
Alternative A1	Dans le cas où les informations fournies sont incomplètes ou incorrectes le système réaffiche le formulaire d'authentification et attend que l'utilisateur ressaisisse ses informations ;

TABLE 3.2 – Description du cas d'utilisation «Authentification».

Remarque

A1 (alternative) est une étiquette utilisée dans le cas où les informations saisies sont incomplètes ou incorrectes pour que le système réaffiche le formulaire d'authentification

Diagramme du cas d'utilisation «Authentification»

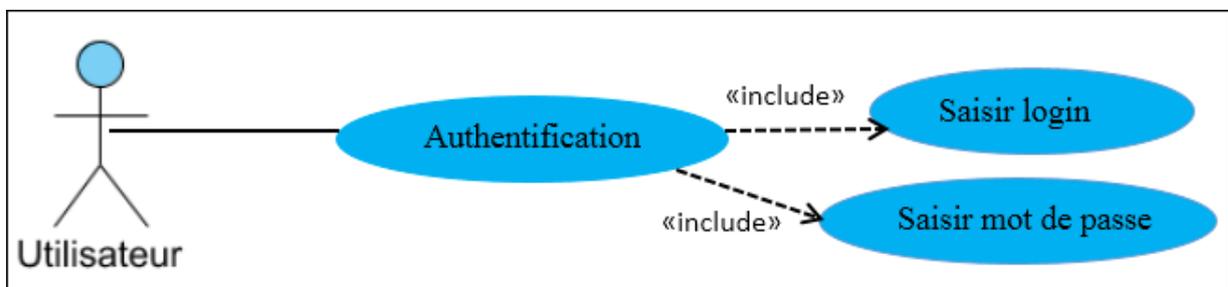


TABLE 3.3 – Diagramme du cas d'utilisation «Authentification».

3.5.4 Le cas d'utilisation «Ajouter un ami »

Cas d'utilisation N° 2	Ajout un ami
Résumé	L'utilisateur a le privilège d'ajouter des nouveaux amis.
Acteurs	Utilisateur.
Précondition	Authentification.
Scénario nominal	<p>[début]</p> <ul style="list-style-type: none"> · Authentification ; · L'utilisateur demande l'interface d'ajout d'un ami ; · Le système affiche l'interface ; · L'utilisateur saisie le pseudo de l'ami a inviter ; · Confirmer l'action ; <p>[fin]</p>

TABLE 3.4 – Description du cas d'utilisation «Ajouter un ami».

Diagramme du cas d'utilisation «Ajouter un ami»

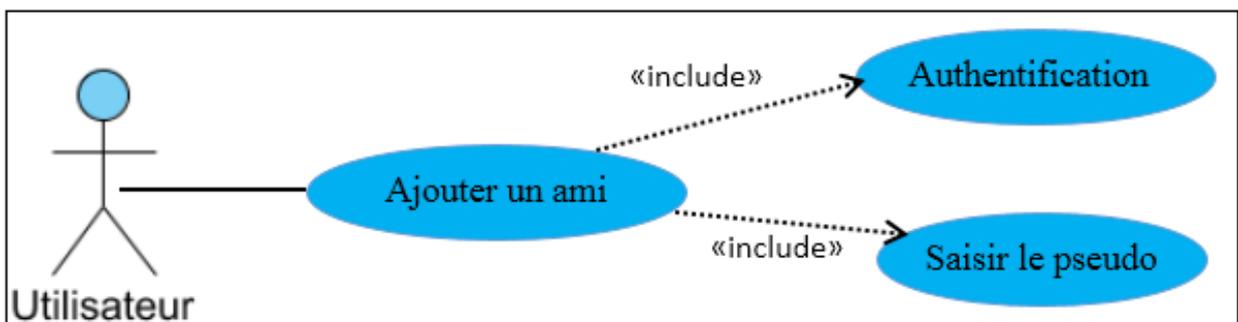


TABLE 3.5 – Diagramme du cas d'utilisation «Ajouter un ami».

3.5.5 Le cas d'utilisation «Inscription / Enregistrement»

Cas d'utilisation N° 3	Inscription / Enregistrement
Résumé	Permet à l'utilisateur de s'inscrire.
Acteurs	Utilisateur.
Précondition	Aucune.
Scénario nominal	[début] <ul style="list-style-type: none"> · Demande de formulaire d'inscription ; · Le système affiche le formulaire d'inscription (demande le nom d'utilisateur (pseudonyme), email, mot de passe ; · L'utilisateur saisit ses informations puis valide ; · Le système vérifie la conformité des informations fournies A1 ; [fin]
Alternative A1	Dans le cas où les informations fournies sont incomplètes ou incorrectes le système réaffiche le formulaire d'inscription et attend que l'utilisateur ressaisisse ses informations ;

TABLE 3.6 – Description du cas d'utilisation «Inscription / Enregistrement».

Diagramme du cas d'utilisation «Inscription»

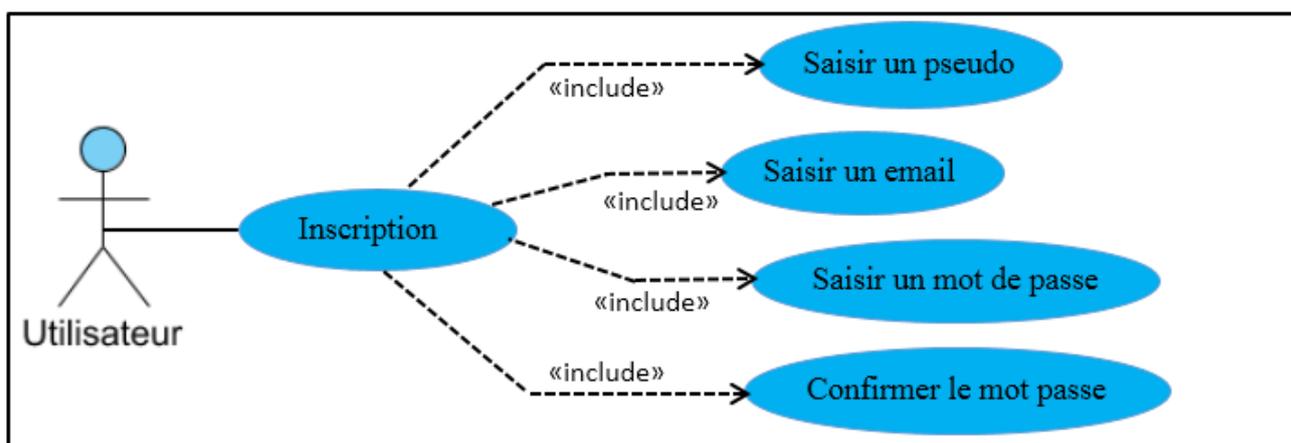


TABLE 3.7 – Diagramme du cas d'utilisation «Inscription».

3.5.6 Le cas d'utilisation «Déconnexion»

Cas d'utilisation N° 4	Déconnexion.
Résumé	Permet à l'utilisateur de quitter son compte.
Acteurs	Utilisateur.
Précondition	Authentification.
Scénario nominal	[début] <ul style="list-style-type: none"> · Accès à l'application. · Accès au menu de l'application. · Cliquer sur le bouton « Déconnexion ». · L'application renvoi automatiquement la page d'accueil. [fin]

TABLE 3.8 – Description du cas d'utilisation «Déconnexion».

Diagramme du cas d'utilisation «Déconnexion»

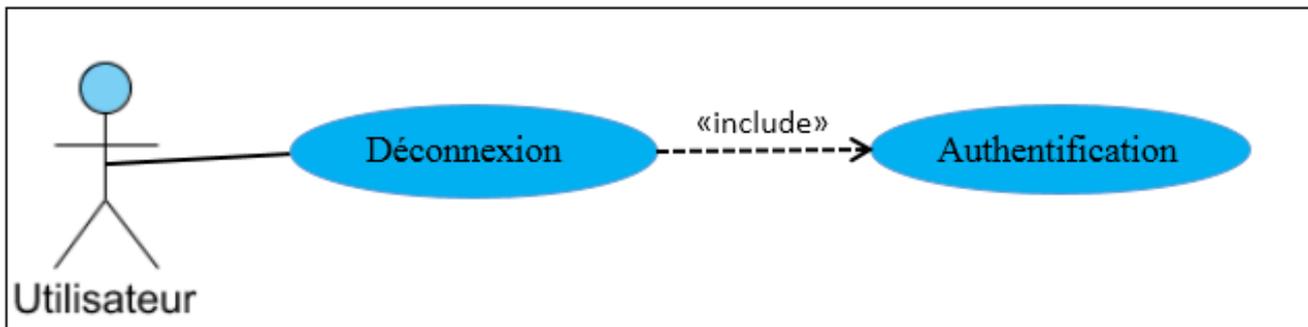


TABLE 3.9 – Diagramme du cas d'utilisation «Déconnexion».

3.5.7 Le cas d'utilisation «Envoyer des messages»

Cas d'utilisation N° 5	Envoyer des messages
Résumé	Permet à l'utilisateur d'envoyer des messages à ses amis qui s'affichent dans une liste.
Acteurs	Utilisateur.
Précondition	Authentification, le récepteur doit être dans la liste des amis.
Scénario nominal	<p>[début]</p> <ul style="list-style-type: none"> · L'utilisateur choisit l'ami dans la liste ; · Le système affiche le formulaire des messages ; · L'utilisateur saisit son message puis il clique sur le bouton envoyer ; <p>[fin]</p>

TABLE 3.10 – Description du cas d'utilisation «Envoyer des messages».

Diagramme du cas d'utilisation «Envoyer des messages»

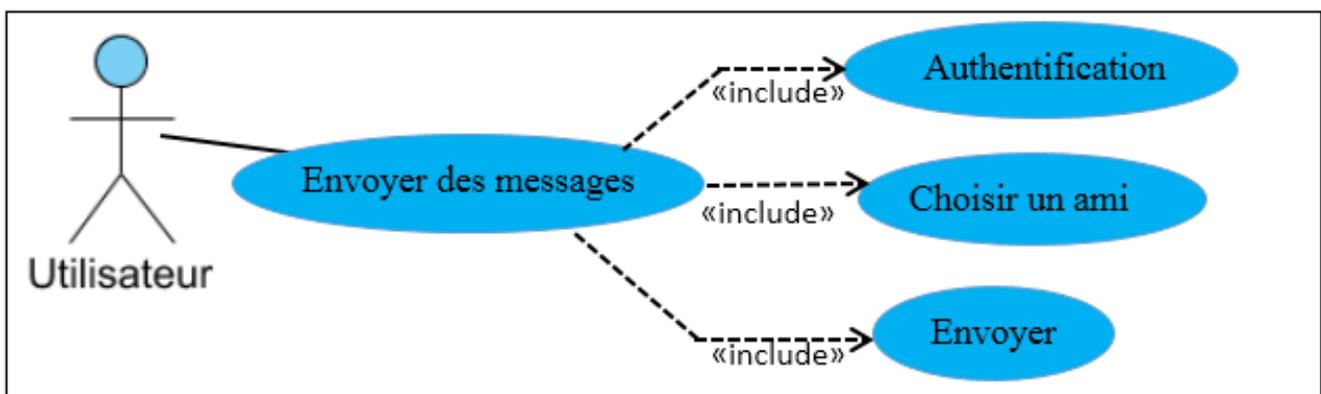


TABLE 3.11 – Diagramme du cas d'utilisation «Envoyer des messages».

3.6 Diagrammes de séquence

3.6.1 Définition d'un diagramme de séquence

Un diagramme de séquences est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont.

Les diagrammes de séquences sont organisés en fonction du temps qui s'écoule au fur et à mesure que nous parcourons la page.

Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence.[12]

3.6.2 Diagramme de séquence du cas d'utilisation «Inscription»

Le diagramme de séquence suivant illustre les interactions nécessaires pour l'inscription de l'utilisateur.

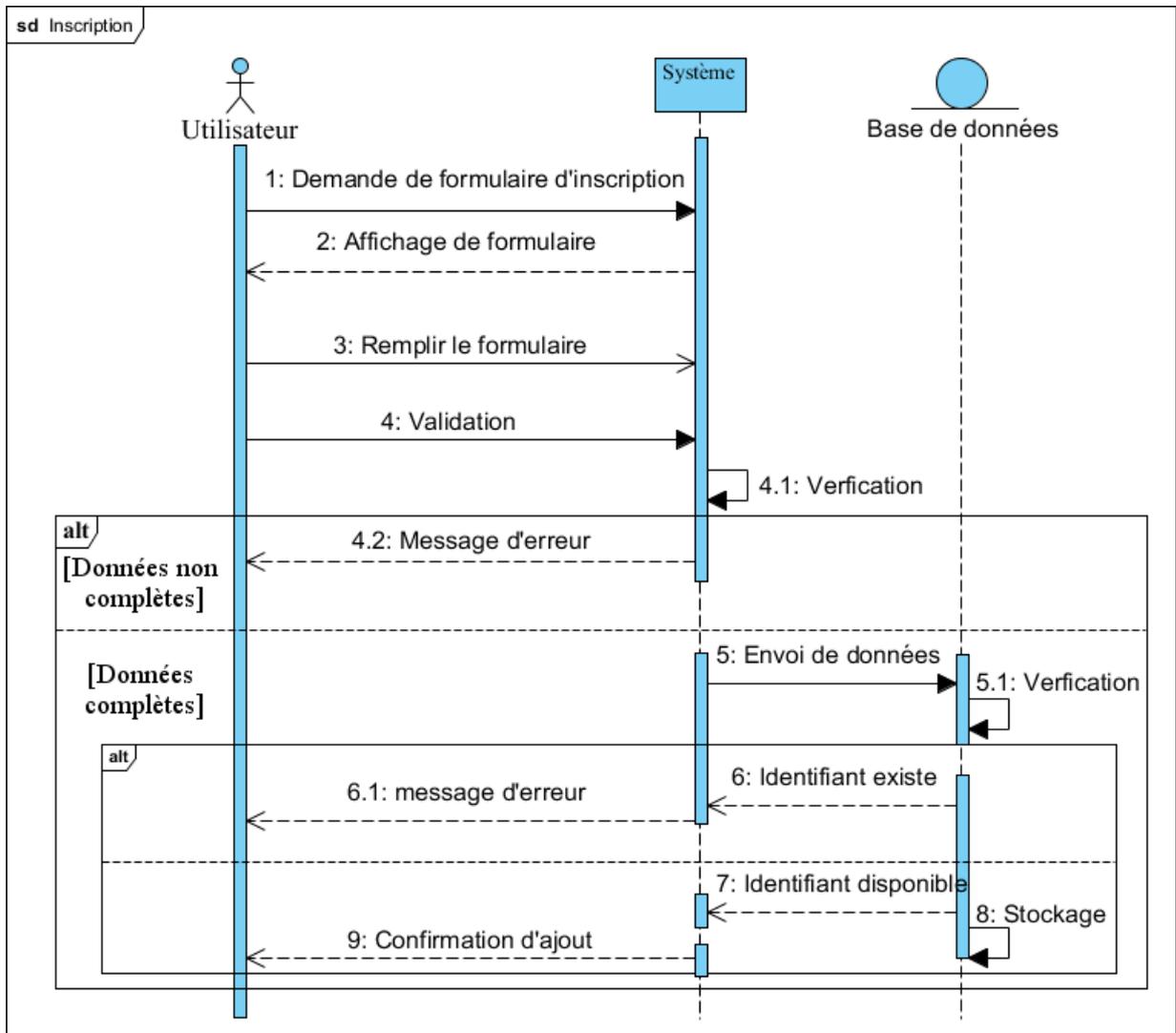


FIGURE 3.5 – Diagramme de séquence du cas d'utilisation «Inscription».

3.6.3 Diagramme de séquence du cas d'utilisation «Authentification»

Le diagramme de séquence suivant illustre les interactions nécessaire pour l'authentification de l'utilisateur.

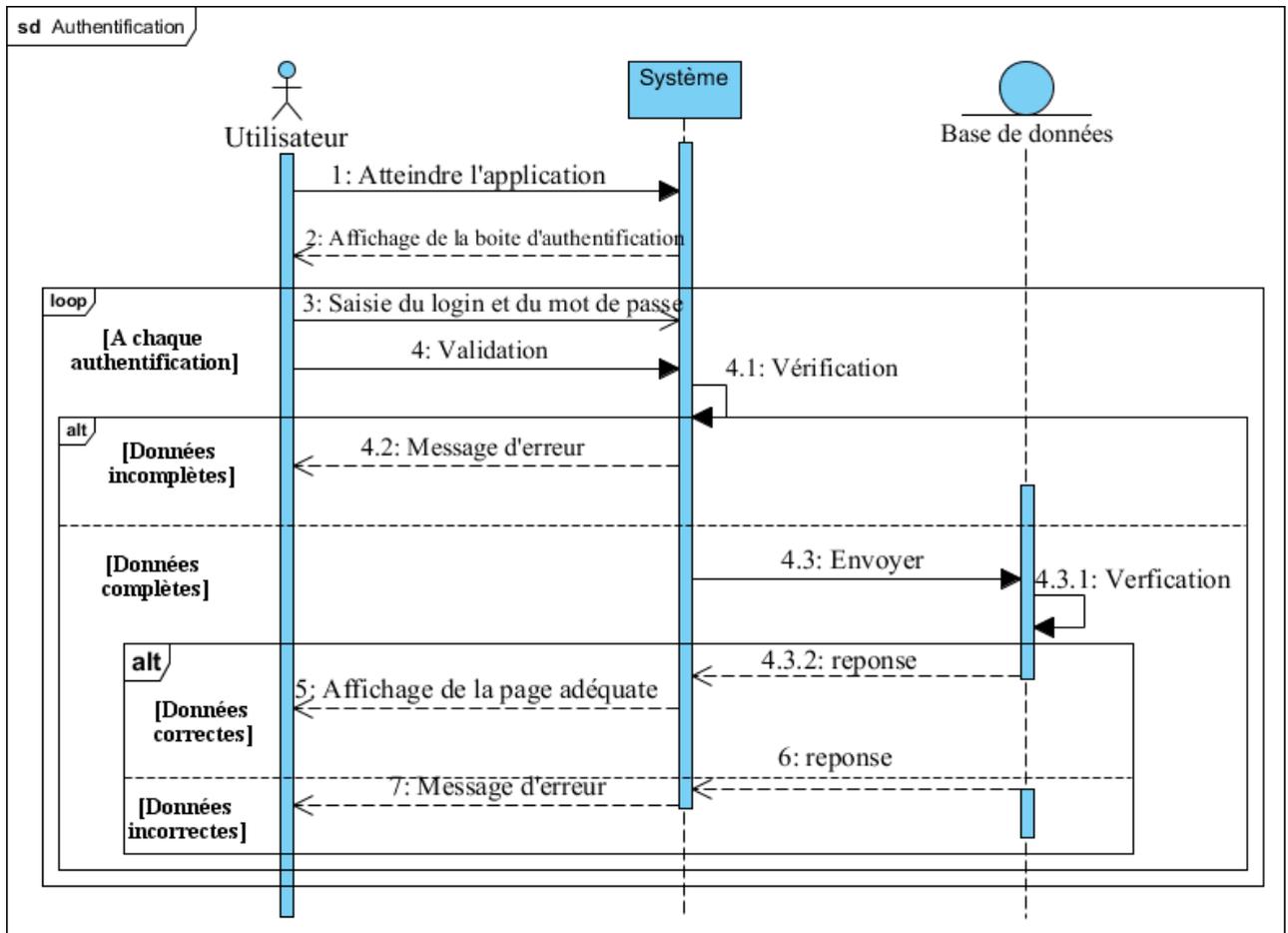


FIGURE 3.6 – Diagramme de séquence du cas d'utilisation «Authentification».

3.6.4 Diagramme de séquence du cas d'utilisation «Ajout des amis»

Le diagramme de séquence suivant illustre les interactions nécessaires pour ajouter des amis.

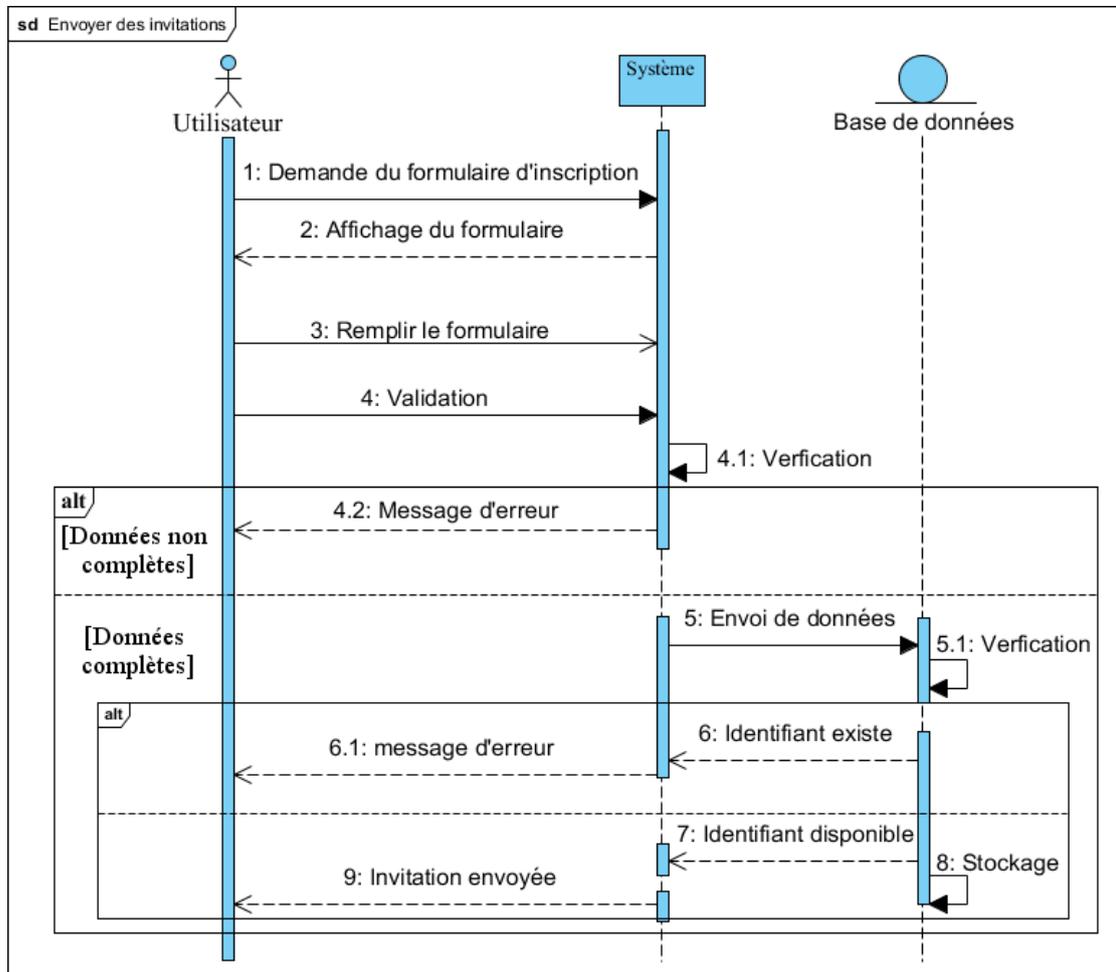


FIGURE 3.7 – Diagramme de séquence du cas d'utilisation «Ajout des amis».

3.6.5 Diagramme de séquence du cas d'utilisation «Envoyer des messages».

Le diagramme de séquence suivant illustre les interactions nécessaire pour l'envoi des messages.

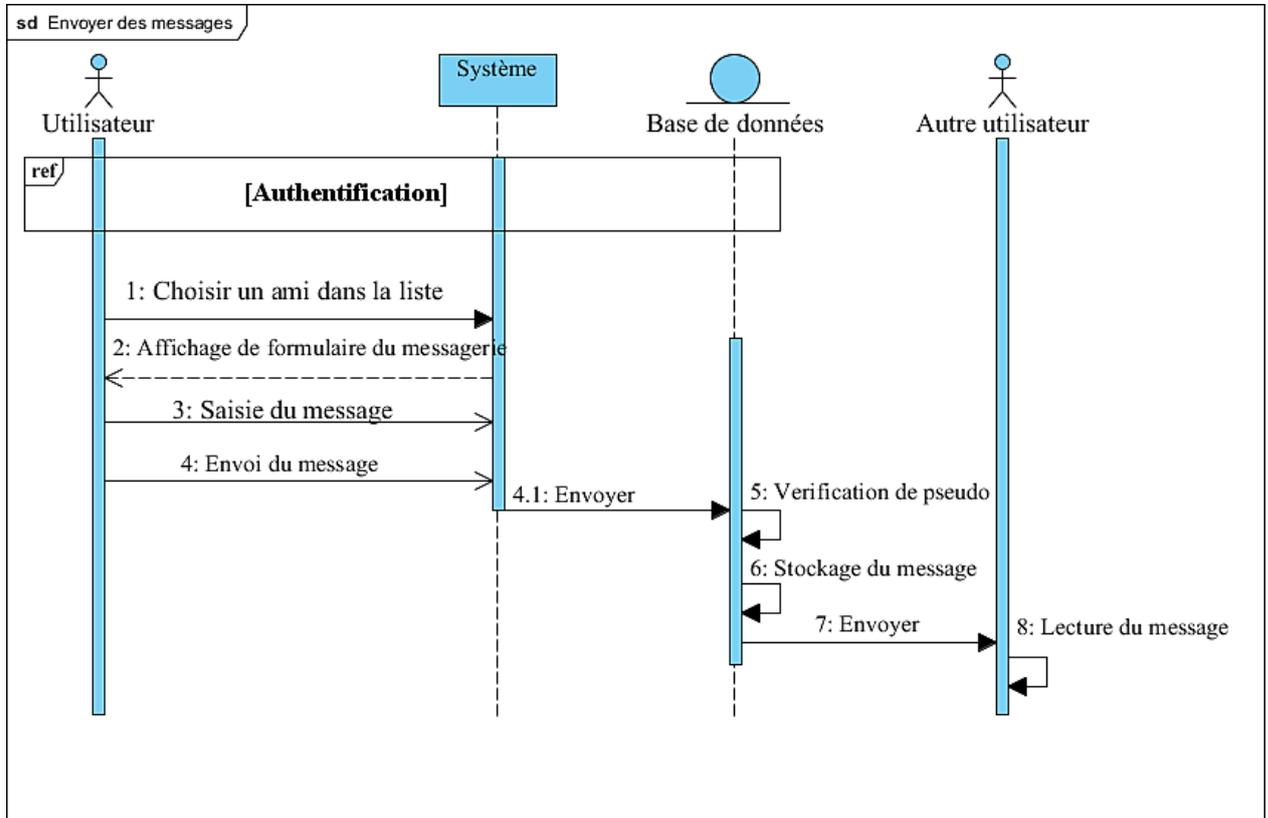


FIGURE 3.8 – Diagramme de séquence du cas d'utilisation «envoyer des messages».

3.6.6 Diagramme de séquence du cas d'utilisation «Déconnexion»

Le diagramme de séquence suivant illustre les interactions nécessaire pour la déconnexion.

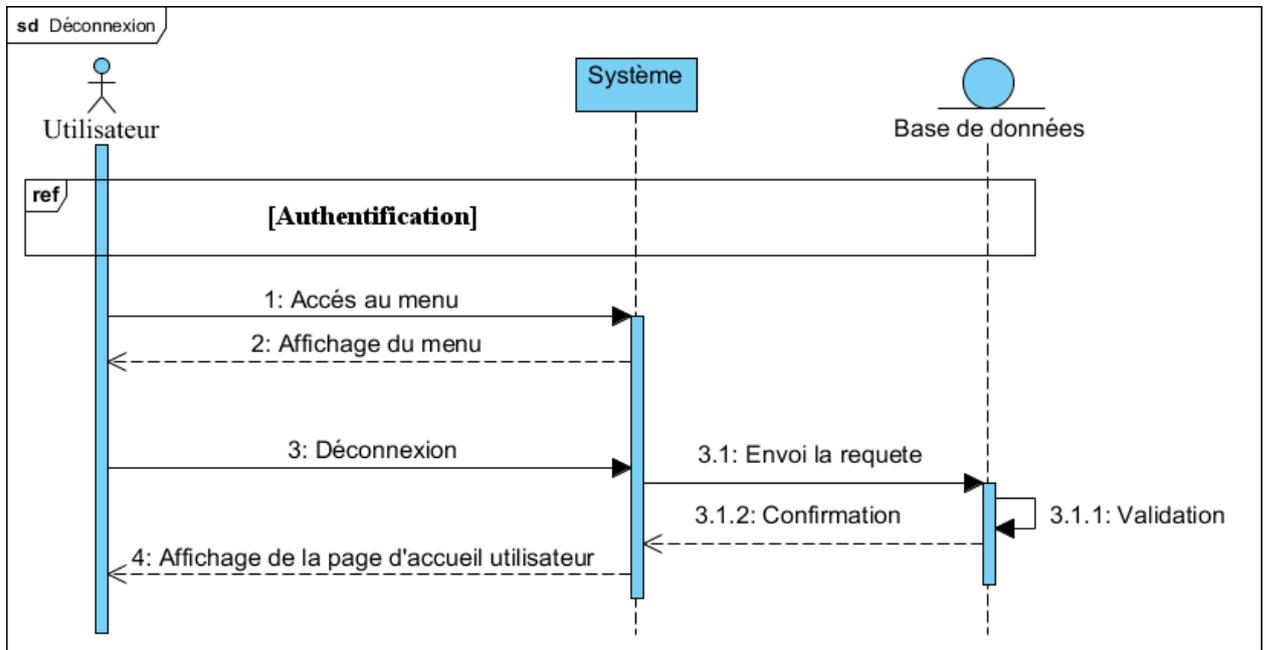


FIGURE 3.9 – Diagramme de séquence du cas d'utilisation «Déconnexion».

3.7 Diagramme de classe

3.7.1 Définition

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Il montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation. Il est important de noter qu'un même objet peut très bien intervenir dans la réalisation de plusieurs cas d'utilisation. Les cas d'utilisation ne réalisent donc pas une partition des classes du diagramme de classes. Un diagramme de classes n'est donc pas adapté (sauf cas particulier) pour détailler, décomposer, ou illustrer la réalisation d'un cas d'utilisation particulier.

Il s'agit d'une vue statique, car on ne tient pas compte du facteur temporel dans le comportement du système. Le diagramme de classes modélise les concepts du domaine d'application ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application. Chaque langage de Programmation orienté objet donne un moyen spécifique d'implémenter le paradigme objet (pointeurs ou pas, héritage multiple ou pas, etc.), mais le diagramme de classes permet de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier.

Les principaux éléments de cette vue statique sont les classes et leurs relations : association, généralisation et plusieurs types de dépendances, telles que la réalisation et l'utilisation. [22]

3.7.2 Diagramme de classe de l'application

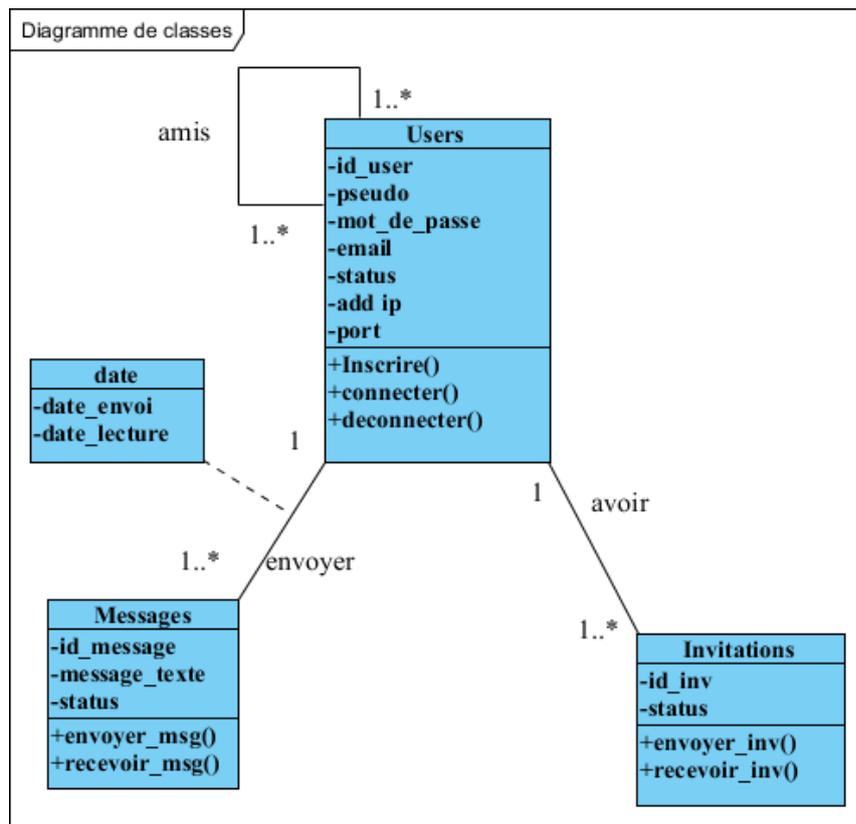


FIGURE 3.10 – Diagramme de classe.

3.7.3 Règle de dérivation du modèle relationnel à partir d'un modèle de classes

- Règle 1 : Transformation des classe Chaque entité devient une relation.[13]

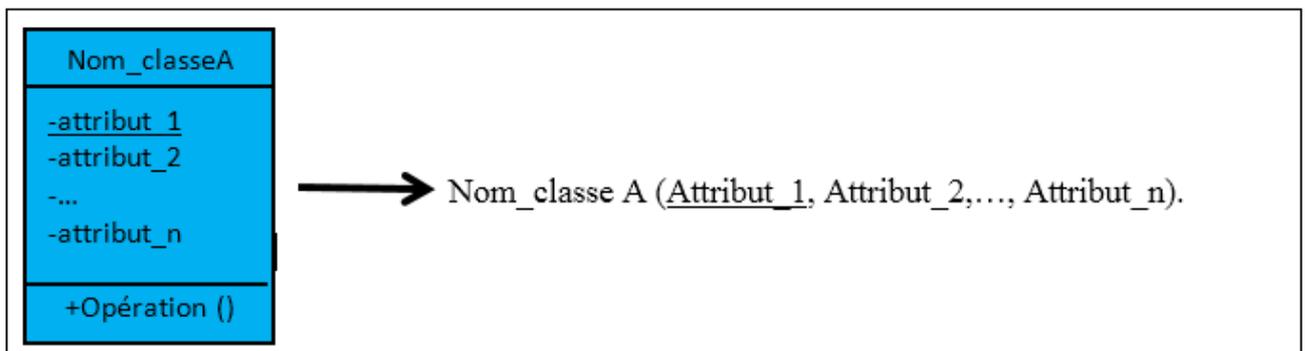


FIGURE 3.11 – Passage d'une classe a une relation.

• **Règle 2 : Association un-a-plusieurs** Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.[13]

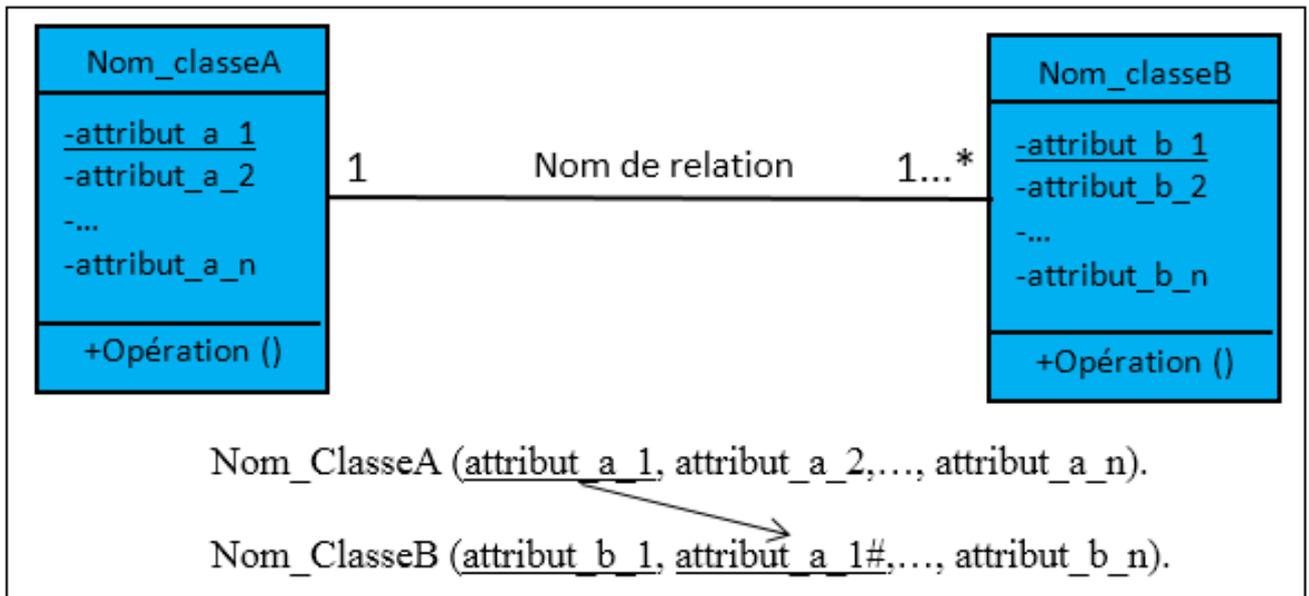


FIGURE 3.12 – Passage un a plusieurs.

• **Règle 3 : Association un-a-un** Il faut ajouter un attribut clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas de UML, il faut ajouter un attribut clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité (classe) connectée à l'association.[13]

– L'application des règles de passage énumérées précédemment, nous permet d'avoir le modèle relationnel de la base de données de notre application à mettre en œuvre

• **Users (id_user, pseudo, mot_de_passe, email, status, add_ip, port)**; status = En ligne / Hors ligne.

• **Amis (id_user1#, id_user2#, status)**; status = En ligne / Hors ligne.

• **Messages (id_message, id_user1#, id_user2#, message_texte, status)**; Status = lu / non lu

- **Invitations** (**id_inv**, **id_user1#**, **id_user2#**, **status**); status (Accepter / Refuser); Id_user1= émetteur, id_user2 = récepteur

3.8 Conclusion

Ce chapitre nous a permis de présenter le langage de modalisation UML et le processus de développement unifié (Unified Processus). Ensuite on a déterminé le cadre du projet de réalisation de l'application et de définir les besoins, en identifiant toutes les entités internes qui vont interagir avec le système (acteurs). Nous avons représenté l'ensemble de séquence d'actions, en identifiant les cas d'utilisation de tous les acteurs du système. En suite, nous avons ressensé la description graphique des cas d'utilisation, en réalisant des diagrammes de séquence décrivant les scénarios nominaux des cas d'utilisations essentiels.

En fin, nous avons réalisé la conception appropriée à notre application selon les concepts de base de diagramme de classe ainsi que les règles de modélisation. Ensuite, nous avons recensé les règles de passage de diagramme de classe vers le modèle relationnel qui nous permet d'avoir le schéma de la base de données de l'application à réaliser. Cela fait la base pour la phase de réalisation telle qu'on va garantir la fiabilité et l'efficacité.

Réalisation

4.1 Introduction

A ce stade du processus, les cas d'utilisation sont terminés. Le problème a été analysé en profondeur ; nous avons défini une conception mieux appropriée aux besoins de l'application.

Ce chapitre est consacré à la réalisation et la mise en oeuvre de notre application de messagerie instantanée, nous allons présenter les outils de développement adoptés, soit l'environnement utilisé qui est Eclipse, ainsi que les langages de programmation (JAVA, XML, PHP), et nous allons présenter le système de gestion de base de données wamp serveur (mysql serveur), ainsi le langage de manipulation de bases de données SQL , et enfin nous allons montrer les interfaces principaux et fenêtres de l'application.

4.2 Outils de développement

4.2.1 Eclipse

Eclipse IDE est un environnement de développement intégré libre (le terme Eclipse désigne également le projet correspondant, lancé par IBM) extensible, universel, permettant potentiellement de créer des projets de développement. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions. [23]

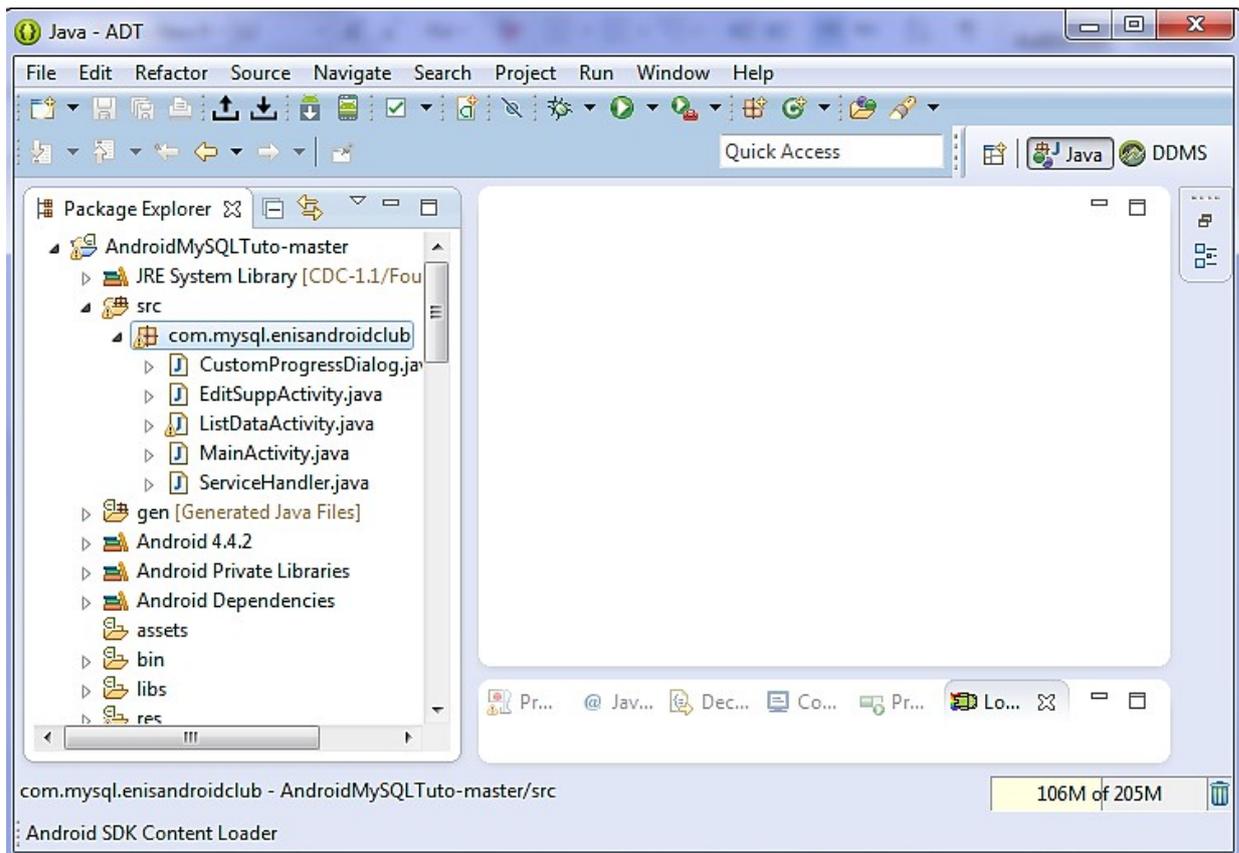


FIGURE 4.1 – Environnement d'Eclipse (ADT).

4.2.2 Langage Java

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp). [24]

4.2.3 Langage XML

Le langage XML (eXtended Markup Language) est un format général de documents orienté texte. Il s'est imposé comme un standard incontournable de l'informatique. Il est aussi bien utilisé pour le stockage de documents que pour la transmission de données entre applications. Sa simplicité, sa flexibilité et ses possibilités d'extension ont permis de l'adapter à de multiples domaines allant des données géographiques au dessin vectoriel en passant par les échanges commerciaux. De nombreuses technologies se sont développées

autour de XML et enrichissent ainsi son environnement.[15]

Le langage XML dérive de SGML (Standard Generalized Markup Language) et de HTML (HyperText Markup Language). Comme ces derniers, il s'agit d'un langage orienté texte et formé de balises qui permettent d'organiser les données de manière structurée. [15]

4.2.4 JSON

JSON (JavaScript Object Notation Notation Objet issue de JavaScript) est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou générable par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript. JSON est un format texte complètement indépendant de tout langage, mais les conventions qu'il utilise seront familières à tout programmeur habitué aux langages descendant du C, comme par exemple : C lui-même, C++, C, Java, JavaScript, Perl, Python et bien d'autres. Ces propriétés font de JSON un langage d'échange de données idéal. [25]

JSON se base sur deux structures : [25]

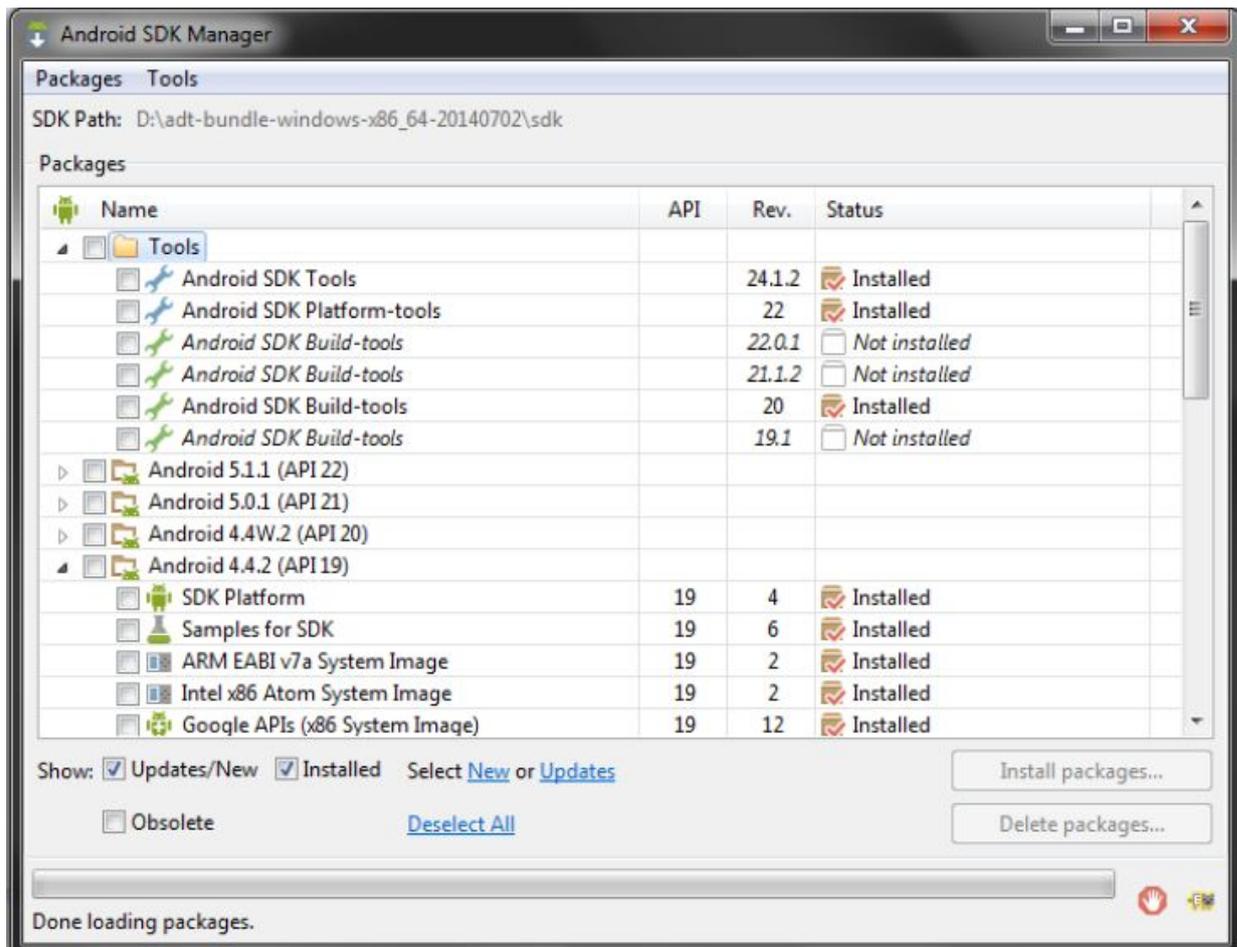
- Une collection de couples nom/valeur. Divers langages la réifient par un objet, un enregistrement, une structure, un dictionnaire, une liste typée ou un tableau associatif.
- Une liste de valeurs ordonnées. La plupart des langages la réifient par un tableau, un vecteur, une liste ou une suite.

4.2.5 SDK

SDK est l'acronyme anglais pour Software Development Kit, il s'agit généralement d'un ensemble d'outils d'aide à la programmation proposé aux développeurs par l'éditeur d'un environnement de programmation spécifique ou d'un système d'exploitation.

Un SDK comprend généralement des fonctionnalités d'aide à la programmation et des outils de débogage ou debogage.

Les SDK sont relativement couramment utilisés dans le domaine du marketing digital. Ils sont par exemple utilisés pour programmer des créations spécifiques en rich media, des applications Facebook ou des applications mobiles. Les développeurs d'applications mobiles utilisent par exemple l'IOS SDK et l'Android SDK. [26]

FIGURE 4.2 – *Android SDK Manager.*

4.2.6 Android développement tools (ADT)

Android Development Tools (ADT) est un plugin pour l'IDE Eclipse qui étend les capacités d'Eclipse qui nous permettent de configurer rapidement de nouveaux projets Android, créer une interface utilisateur de l'application, ajouter des paquets basés sur l'API Framework Android, déboguer nos applications utilisant les outils de SDK Android, et même exporter signé (ou non signé) des fichiers .apk afin de distribuer notre application. [27]

4.2.7 Android virtual device (AVD)

L'Android Virtual Device, aussi appelé AVD, est un émulateur de terminal sous Android, c'est-à-dire que c'est un logiciel qui se fait passer pour un appareil sous Android sur ordinateur. C'est la raison pour laquelle on a pas besoin d'un périphérique sous Android pour développer et tester la plupart des applications ! En effet, une application qui affiche un calendrier par exemple peut très bien se tester dans un émulateur, mais une application qui exploite le GPS doit être éprouvée sur le terrain pour que l'on soit certain de son comportement. [16]



Pour créer ou bien lancer un émulateur.

FIGURE 4.3 – La Barre d'outils d'Eclipse.

Voici notre emulateur :

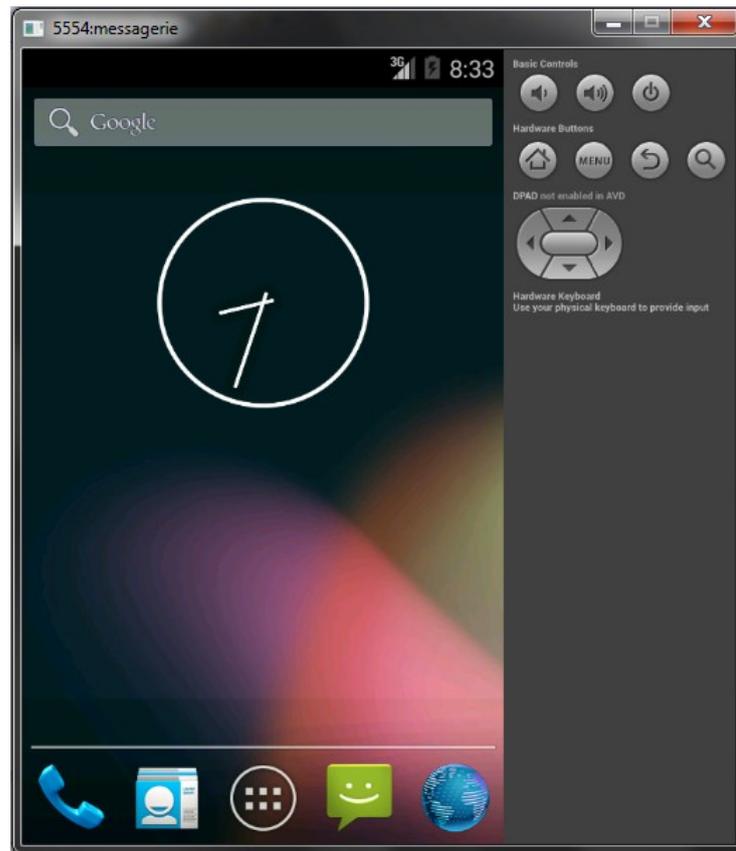


FIGURE 4.4 – *Émulateur Android.*

4.2.8 Wamp server (Mysql serveur)

Pour implémenter notre base des données, nous avons utilisé l'environnement de création de base de données PHPMyAdmin et le système de gestion de base de données MySQL.

Présentation

WampServer est une plateforme de développement Web sous Windows. Il permet de développer des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement les bases de données.

Langage de Manipulation de Bases de Données MySQL

Est un système de gestion de base de données qui gère les fichiers constituant une base, consiste en un ensemble de programme chargé de gérer une ou plusieurs base de données et qui fonctionne selon une architecture client/serveur, MySQL propose une représentation extrêmement simple de l'information sous forme de table. MYSQL est aussi libre et

gratuit. Si nous avons choisi ce SGBD c'est plus pour des raisons de performance et de fonctionnalité ouverte, MySQL est beaucoup moins complexe à installer et à administrer que d'autres systèmes[17]

4.2.9 Langage PHP

PHP est un langage de programmation informatique essentiellement utilisé pour produire à la volée des pages web dynamiques. Dans sa version 5 lancée en juillet 2004, PHP s'est imposé comme le langage de référence sur le web en raison de sa simplicité, de sa gratuité et de son origine de logiciel libre.

PHP est considéré par certains comme une plateforme de développement en raison de l'étendue et de la richesse de sa bibliothèque. [28]

4.2.10 Notepad++

Notepad++ est un éditeur de texte très léger, très puissant et libre (licence GPL). Il est parfait pour programmer avec des langages ne nécessitant pas d'environnement de développement (html, css, javascript, php) ou en ayant un peu pratique (python, processing), ou pour du traitement de données. Il prend en charge par défaut une cinquantaine de langages différents.[29]

4.3 Présentation des interfaces de l'application

Dans ce qui suit, nous allons présenter les interfaces de notre application.

4.3.1 Interface d'accueil

Cette interface offre un aperçu de la page d'accueil de l'application. Elle offre aux utilisateurs un service d'inscription et d'identification.



FIGURE 4.5 – Interface d'identification.

4.3.2 Interface d'inscription

Cette figure offre un aperçu de l'interface de création d'un nouveau compte.

Sur cette interface, l'utilisateur doit saisir un pseudo (identifiant), un email et un mot de passe, puis cliquer sur le bouton inscrire.



FIGURE 4.6 – Interface d'inscription.

4.3.3 Interface de la Liste d'amis

La première fois que l'utilisateur se connecte à son compte, cette interface s'affiche et lui indique que sa liste d'amis est vide.



FIGURE 4.7 – Interface de la Liste d'amis.

4.3.4 Menu «Ajouter un nouveau ami / Déconnexion»

Sur Cette interface, l'utilisateur aura la possibilité d'ajouter des amis, ou encore de se déconnecter de son compte.



FIGURE 4.8 – Menu «Ajouter un nouveau ami / Déconnexion».

4.3.5 Interface d'ajout des amis

L'interface ci-dessous met en disposition de l'utilisateur la possibilité d'ajouter un ami en inscrivant l'identifiant de ce dernier.



FIGURE 4.9 – Interface d'ajout des amis.

4.3.6 Barre des notifications

Sur cette interface, l'utilisateur aura un accès sur l'ensemble de ses notifications : réception de messages et nouvelles invitations.

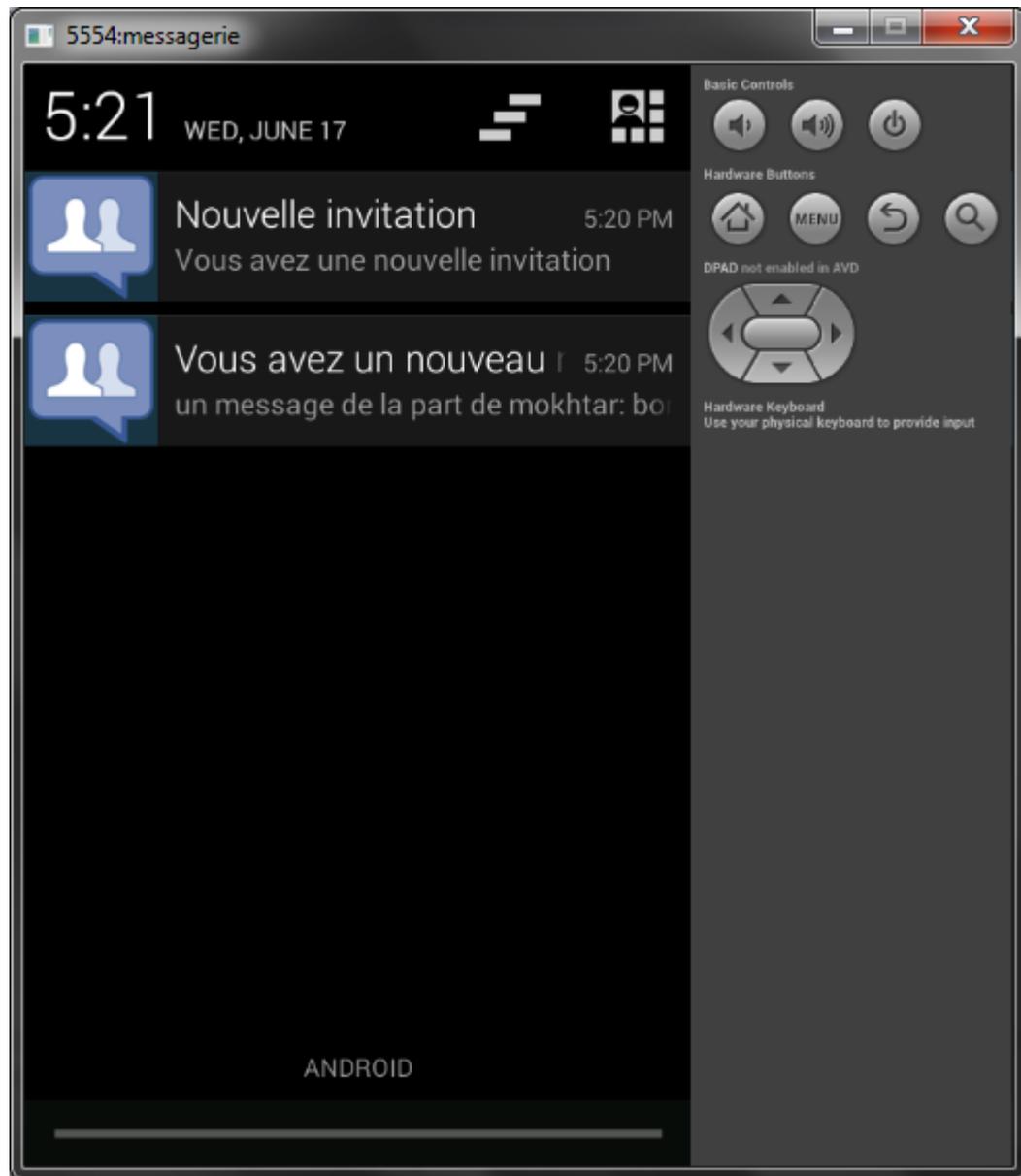


FIGURE 4.10 – Barre des notifications.

4.3.7 Interface de confirmation des invitations

L'utilisateur aura aussi la possibilité d'accepter ou de refuser une invitation.

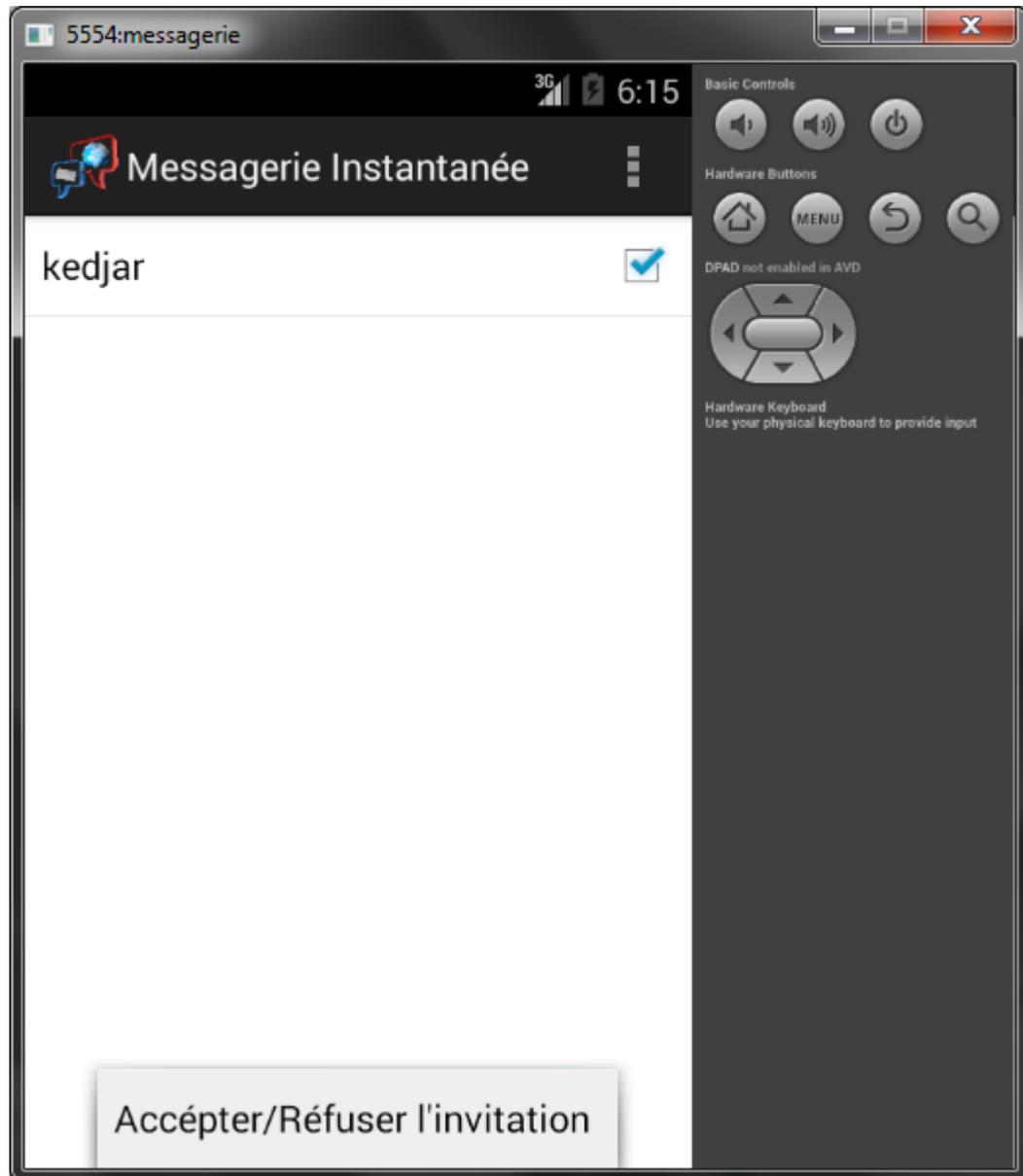


FIGURE 4.11 – *Interface de confirmation des invitations.*

4.3.8 Interface de la Liste des amis en ligne et hors ligne

L'interface ci-dessous est un exemple sur la vue de la liste d'amis (en ligne et hors ligne).

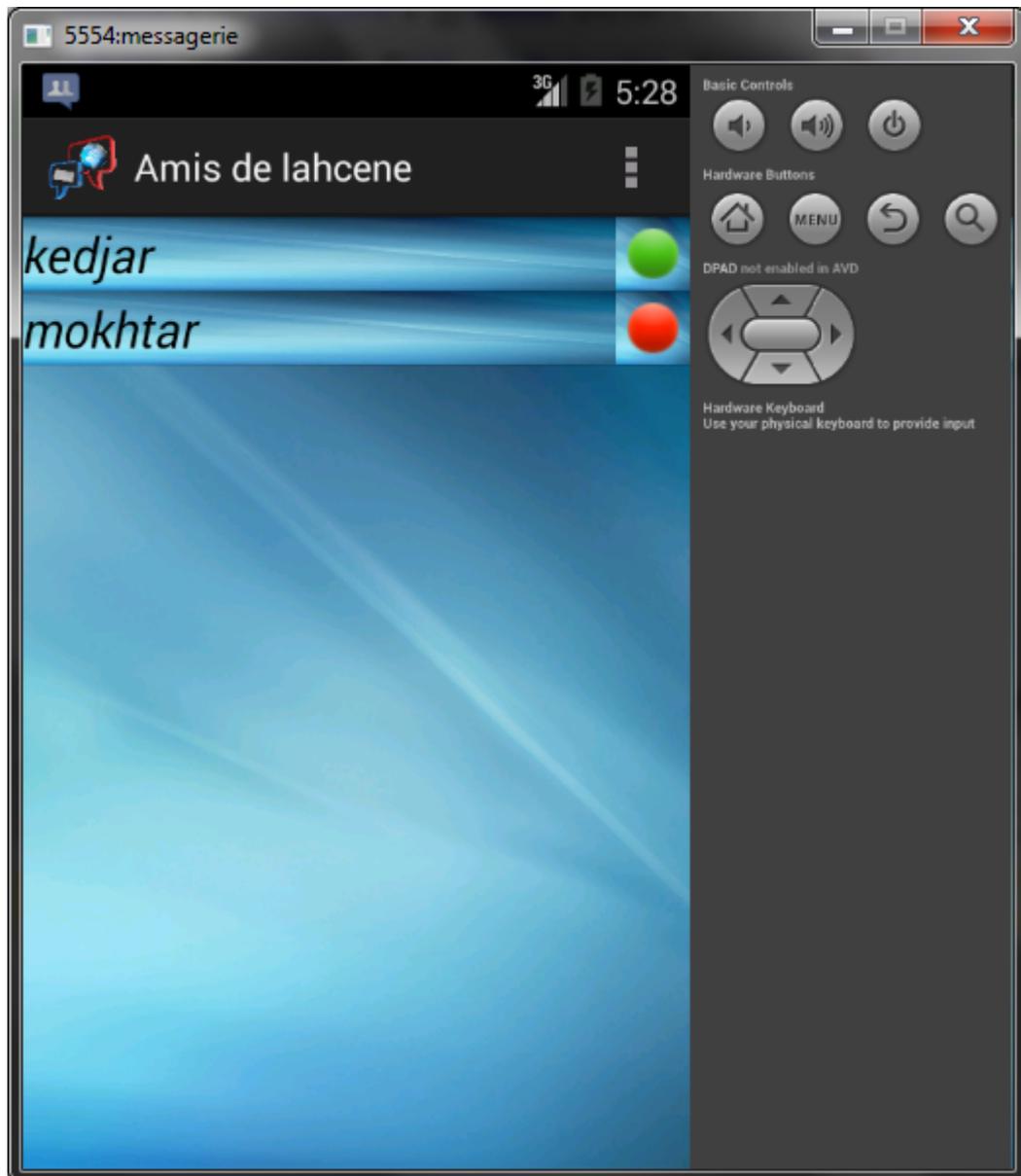


FIGURE 4.12 – Interface de la Liste des amis en ligne et hors ligne.

4.3.9 Interface de discussion

Sur cette interface, l'utilisateur pourra consulter ses messages échangés avec un correspondant, ainsi que relancer la discussion.

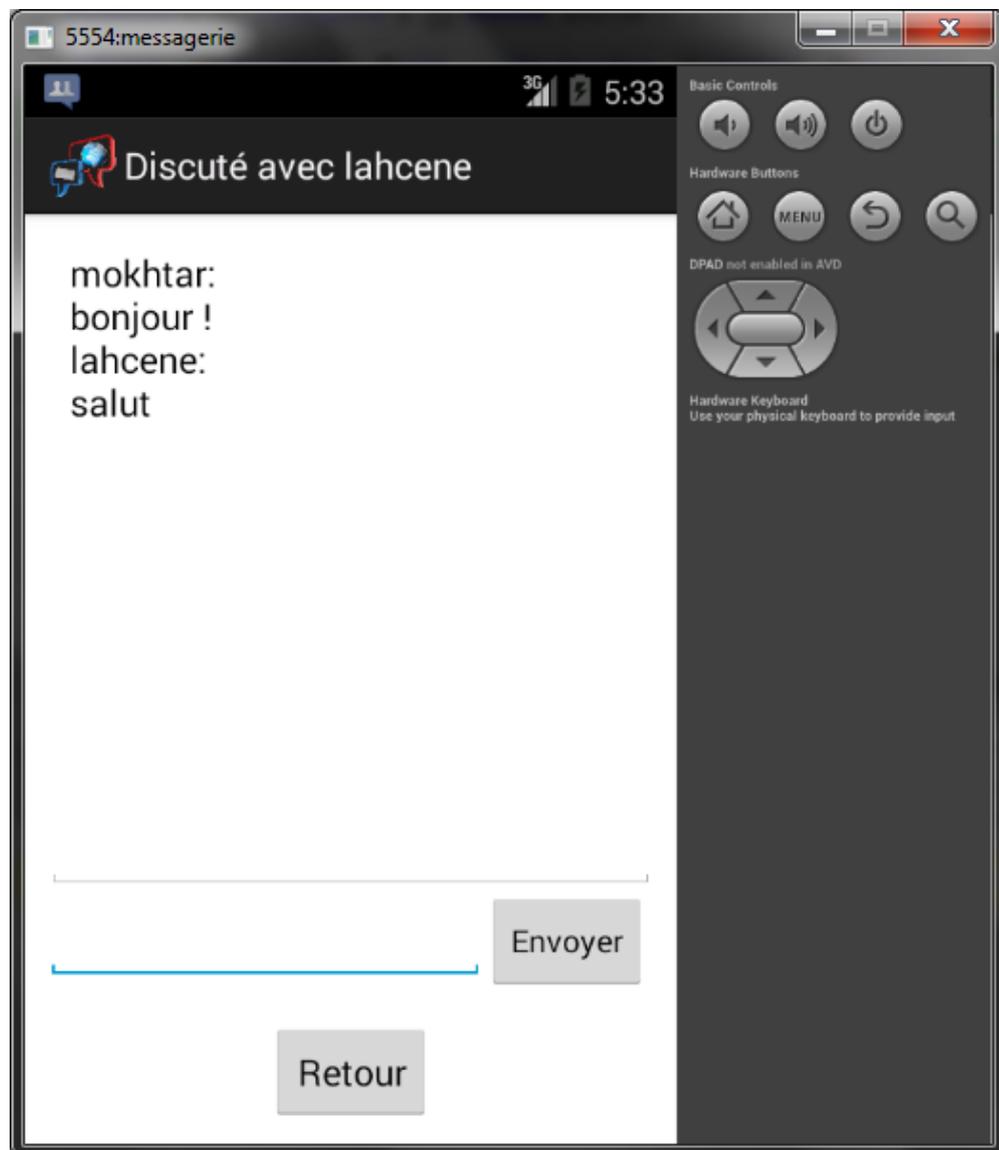


FIGURE 4.13 – Interface de discussions.

4.4 Conclusion

La phase réalisation est l'étape la plus importante dans le cycle de vie d'une application. Dans ce chapitre, nous avons décrit brièvement le processus de réalisation de notre application en spécifiant l'environnement de développement associé à notre système. En effet, nous avons achevé l'implémentation tout en respectant la conception élaborée. En d'autres termes, nous détenons la version finale de l'application, installée dans notre environnement de développement.

Conclusion générale et Perspectives

Nous sommes parvenus, par le biais de ce projet, à réaliser une application qui fonctionne sous Android qui permet aux utilisateurs d'échanger des messages textes et les sauvegarder dans une base de données.

Au cours de la phase d'analyse nous avons structuré et défini les besoins du système. Il s'agit de formuler, d'affiner et d'analyser la pluparts des cas d'utilisations par les diagrammes UML.

La phase de conception suit immédiatement la phase d'analyse, il s'agit alors d'étendre la représentation effectuée au niveau de l'analyse en y intégrant les aspects techniques les plus proches des préoccupations des besoins techniques. L'élément principal à livrer au terme de cette phase est le diagramme de classe ainsi que le schéma relationnel.

Enfin, nous avons entamé la réalisation en utilisant les outils d'implémentation appropriés l'intégration du contenu, la gestion de la base de données (MySQL), et présenter les différentes interfaces de notre application.

Ce travail nous a permis d'apprendre énormément de choses concernant le développement sous Android, et ce mémoire présente notre première expérience pour la mise en oeuvre d'une application mobile, également nous avons appris à manipuler toute une panoplie d'outils :Eclipse supportant l'IDM, Wamp server, et quelques langages de programmation telles que JAVA,XML,PHP.

Ce fut une occasion pour nous de compléter de manière transversale nos compétences en informatique, d'élargir et d'approfondir nos connaissances et les apprécieraux diverses réalités du terrain. Cependant des perspectives d'améliorations de notre application restent envisageables pour être enrichie par des fonctionnalités avancées telles que :

- La suppression des amis.

- La modification des identifiants.
- Création des groupes de discussions.
- Partage des fichiers.
- Renforcer la sécurité en limitant le nombre de tentatives d'authentifications .

Bibliographie

- [1] *TAAKAM R. Le multimédia mobile, master 2 système d'information multimédia et internet 2010.*

- [2] *COSTENOBEL G. et BAROIS P. Projet text to speech, Université Lille 1 Sciences et technologies, 2010/2011.*

- [3] *BENNEFOI F. Terminaux mobiles communications (TMC), Programmation Android, Avril 2011.*

- [4] *Agence Régional de Santé(ARS), messagerie instantanée, manuel d'utilisation, ARS-IM- MANUTIL, Manuel Utilisateur.*

- [5] *ROQUES P. les cahiers du programmeur UML2, 4eme Edition EYROLLES, 2008.*

- [6] *STEFFE J. Cours UML, ENITA de Bordeaux, mars 2005. Et DIGALLO F. Méthodologie des systèmes d'information-UML, CNAM (Conservatoire National des Arts et Métiers) 2000/2001.*

- [7] *AYADI H. et ASLI F. Conception et réalisation d'un système de messagerie électronique, département d'informatique. Licence en informatique, 2010/2012.*

- [8] *ROQUES P et VALLEE F. UML en action « de l'analyse des besoins a la conception en JAVA », 2eme Edition EYROLLES, 2003.*

- [9] *JOCOBSON I. et BOOCH G. et RUMBAUGH J. Le processus unifié de développement (the unified software development process), Edition EYROLLES, 2000.*
 - [10] *AUDIBERT L. UML 2.0, département informatique, IUT (Institut Universitaire de Technologie de Villetaneuse).*
 - [11] *ROQUES P. UML2 par la pratique 5eme Edition EYROLLES, 2006.*
 - [12] *RAISSI A. mémoire de MASTER, conception et développement d'un site web de e-commerce pour le compte de LSAT NOKIA, 2012/2013, Université virtuelle de TUNIS.2011*
 - [13] *MOGHRANI A. et AZZOUG Z. conception et réalisation d'une application de suivi des patients dans un établissement hospitalier, Université Abderrahmane Mira-Béjaïa-licence informatique 2009.*
 - [14] *CLAUDE A. et LEGRAND M. Membres du GREG et enseignantes en Terminale ACA et BTS Assistant de Gestion PME-PMI, creg.ac-versailles.fr, 2004.*
 - [15] *CARTON O. L'essentiel de XML, cours de M2 Pro à l'Université Paris Diderot.*
 - [16] *ESPIAU F. Créez des applications pour Android, Année 2013.*
 - [17] *ROEGEL D. Oracle : sql, IUT Nancy 2 , roegel@loria.fr, Année 1998/1999.*
-

Webliographiques

- [18] *[http://www.mangotreestudios.fr/blog/les différents types d'applications mobile](http://www.mangotreestudios.fr/blog/les-diff%C3%A9rents-types-d-applications-mobile).*

- [19] *<http://www.slideshare.net> : généralités et contexte des systèmes d'exploitation mobile, 2012.*

- [20] *<http://www.devellopez.com/Mobiles/Android>.*

- [21] *<http://play.google.com/store/apps/details?id=com.whatsapp>=fr.*

- [22] *<http://www.laurent-audibert.devellopez.com/cours-uml>.*

- [23] *<http://www.techno-science.net>.*

- [24] *<http://www.futura-sciences.com>.*

- [25] *<http://www.json.org/jsonfr.html>.*

- [26] *<http://www.definitions-webmarketing.com/Definition-SDK>.*

- [27] *<http://developer.android.com/tools/sdk/eclipse-adt.html>.*

- [28] *<http://www.jobintree.com/dictionnaire/definition-php-309.html>.*

- [29] *<http://nliautaud.fr/wiki/articles/notepad>.*

Résumé :

La disponibilité croissante des mobiles présente un avantage pour les usagers comme pour les développeurs des applications. Le développement des applications mobiles se positionne actuellement comme une activité majeure du génie logiciel.

Le secret des appareils mobiles repose sur les systèmes d'exploitation qu'ils utilisent, ils fournissent des caractéristiques favorables à un développement réussi des différentes applications sur mobiles.

Le but de mémoire est la conception et la réalisation d'une application de chat «messagerie instantanée» qui fonctionne sous le système Android. Pour la développer, nous avons utilisé plusieurs technologies parmi lesquelles, nous citons : La plateforme Android en utilisant son outil de développement SDK Android, le format léger d'échange de données «JSON», le langage UML pour réaliser l'étude conceptuelle du système, le PHP pour la gestion de base de données et le MySQL comme serveur de base de données.

Mots clés : Android, Application mobile, Messagerie instantanée, UML.

Abstract :

The increasing availability of mobile is an advantage for users as for developers of applications, the development of mobile applications is currently positioned as a major activity of software engineering.

The secret of mobile devices is based on the operating systems they use, they provide favorable characteristics for successful development of different applications on mobile.

The aim of this dissertation is the design and implementation of an application of "instant messaging" that runs under the Android system. To develop it we used several technologies : The Android platform using the Android SDK development tool,JSON, UML to achieve the conceptual design of the system, PHP to manage the data base and MySQL as the database server.

Key words : Android, Mobile application, Instant messaging, UML,
