

**RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE MINISTÈRE DE
L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE
UNIVERSITE FERHAT ABBES de SETIF 1**



Faculté des Sciences
Département d'Informatique

*Mémoire de Fin d'études
En vue d'obtention du diplôme de
LICENCE académique en informatique.*

Thème

**Développement d'une application Android pour la
messagerie instantanée.**

Présenté par :

- Benchikh Lina
- Khallaf Amani

Encadré par :

Mr. BOULKHIOUT Youssef

Promotion 2018/2019.

RÉSUMÉ

Ce travail est réalisé dans le cadre d'un mémoire de fin d'étude pour l'obtention du diplôme de licence académique en informatique, il consiste à créer une application Android pour la messagerie instantanée en utilisant des techniques et des outils comme l'Android Studio , JAVA et le langage UML pour réaliser l'étude conceptuelle de système. Cette application destinée aux Smartphones vise à satisfaire les utilisateurs de communiquer en privé avec d'autres personnes de leur choix de façon plus efficace et d'obtenir des résultats très intéressants.

Mots clés

UML, Application Android, Sockets.

ABSTRACT

This work is submitted in partial fulfillment of the requirements of an academic degree in computer science and to create a application for faculty of Sciences University of Ferhat Abbas setif 1, using techniques and tools like Android Studio, JAVA and the UML language. The aim of this application intended for smartphones is to gratify the users to communicate privately with other people of their choice more efficiently and to obtain very interesting results.

Key Words

UML, Application Android, Sockets.

DÉDICACES

Je dédie ce mémoire :

A mon père : Aucune dédicace ne saurait exprimer le respect que j'ai toujours eu pour vous. Rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être.

A ma mère : Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

A mes sœurs jumelles et mon frère

A mon binôme Lina et sa famille.

A mes amis et surtout Hachemi ... merci beaucoup pour votre aide, en témoignage de mon amour profond et de respect pour ses grands sacrifices

Merci pour toutes les valeurs de fraternité et pour tous les moments passés ensembles.

Et à toute ma famille et à tous ceux que j'aime.

J'espère que vous serez fier de moi !

DÉDICACES

Je dédie ce mémoire :

A mon père : Aucune dédicace ne saurait exprimer le respect que j'ai toujours eu pour vous. Rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être.

A ma mère : Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

A mes sœurs et mon frère

A mon binôme Amani et sa famille.

A mes amis et surtout Hachemi ... merci beaucoup pour votre aide, en témoignage de mon amour profond et de respect pour ses grands sacrifices

Merci pour toutes les valeurs de fraternité et pour tous les moments passés ensembles.

Et à toute ma famille et à tous ceux que j'aime.

J'espère que vous serez fier de moi !

REMERCIEMENTS

*Merci Allah de nous avoir donné la capacité
d'écrire et de réfléchir, la force d'y croire, la patience
d'aller jusqu'au bout du rêve et le bonheur de lever nos
mains vers le ciel et de dire "Elhamdeli Allah".*

*Tout d'abord, ce travail ne serait pas aussi riche sans l'aide et
l'encadrement de Mr. BOULKHIOU Youssef, on le remercie
pour la qualité de
son encadrement exceptionnel, pour sa patience sa disponibilité
durant notre préparation de ce
mémoire.*

*Nos remerciements vont également à toutes les
amis qui nous ont aidés et soutenue de près ou de loin.*

A tous Merci

Table des Matières

Introduction générale	1
Cahier de charge	3
1. Objectif	3

CHAPITRE I : CONCEPTION.

1. Introduction	6
2. Langage de modélisation unifié(UML)	6
2.1. Définition	6
2.2. Les différents types diagrammes d'UML	6
3. Les diagrammes utilisés dans notre projet	7
3.1. Diagramme de contexte	8
3.2. Diagramme des cas d'utilisations	8
3.2.1. Définition.....	8
3.2.2. Identification des acteurs	8
3.2.3. Diagramme de cas d'utilisation globale	9
3.3. Diagramme de Séquence	11
3.3.1. Définition	11
3.3.2. Diagramme de séquence « Inscription /Sign Up »	11
3.3.3. Diagramme de séquence « Log In »	12
3.3.4. Diagramme de séquence « Envoyer des messages »	13
3.3.5. Diagramme de séquence « Changer le mot de passe ».....	14
3.3.6. Diagramme de séquence « Log Off »	15
4. Diagramme de Classe	16
4.1. Définition	16
4.2. Notre diagramme de classe	17
5. Conclusion	17

CHAPITRE II : RÉALISATION.

1. Introduction	19
2. Environnement de travail	19
2.1. Environnement matériel.....	19
2.2. Environnement logiciel	19
2.2.1. Logiciel de développement de l'application.....	19
2.2.2. Logiciel pour éditer la base de données	21
2.2.3. Logiciel de modélisation	21
2.2.4. Logiciel pour éditer le cahier de charge.....	21
3. Présentation des Activités (des interfaces) de l'application.....	21
4. Une partie de code source	31
4.1.Code Java « Activity Forget Password ».....	31
4.2.Code XML « Activity Forget Password ».....	32
5. Des captures sur la base données	33
6. Référence	34
7. Conclusion	35

Liste des figures et diagrammes

<i>Figure 1 : Le symbole d'UML</i>	7
<i>Figure 2 : Les diagrammes UML</i>	8
<i>Figure 3 : diagramme de contexte</i>	8
<i>Figure 4 : diagramme de cas d'utilisation globale</i>	10
<i>Figure 5 : diagramme de séquence « Inscription »</i>	12
<i>Figure 6: diagramme de séquence « Login »</i>	13
<i>Figure 7 : diagramme de séquence « Envoyer des messages »</i>	14
<i>Figure 8 : diagramme de séquence « Changer le mot de passe »</i>	15
<i>Figure 9 : diagramme de séquence « LogOff »</i>	16
<i>Figure 10 : diagramme de classe</i>	17
<i>Figure 11 : Page d'accueil</i>	22
<i>Figure 12 : interface de log in</i>	22
<i>Figure 13 : interface de Sign up</i>	23
<i>Figure 14 : interface de main</i>	23
<i>Figure 15 : interface de help</i>	24
<i>Figure 16 : interface de paramètre</i>	25
<i>Figure 17 : Supprimer un message</i>	26
<i>Figure 18 : Changer mot de passe</i>	26
<i>Figure 19 : Oublier mot de passe</i>	26
<i>Figure 20 : Ajouter des amis</i>	27
<i>Figure 21 : Envoyer invitation</i>	27
<i>Figure 22 Liste des invitations</i> :	27
<i>Figure 23 :Invitations envoyées</i>	27

<i>Figure 24 : Interface de communication</i>	28
<i>Figure 25 : Interface de notification</i>	28
<i>Figure 26 : Supprimer un message</i>	28
<i>Figure 27 : Interface de contacts</i>	29
<i>Figure 28 : Interface de mon profile</i>	30
<i>Figure 29 : Diffuser un message</i>	30
<i>Figure 30 : Interface de la base de données</i>	33
<i>Figure 30 : Arbre de la base de données</i>	34

Liste des tableaux

Table 1 :Description de cas d'utilisation « inscription »	11
Table 2 :Description de cas d'utilisation « Log in »	13
Table 3 :Description de cas d'utilisation « envoyer des messages»	14
Table 4 :Description de cas d'utilisation « changer mot de paase »	15
Table 5 :Description de cas d'utilisation « Logoff »	16

Introduction Générale

Actuellement . . . le monde connaît une progression rapide dans l'utilisation des appareils téléphoniques portables (mobiles), grâce aux applications mobiles .Ces dernières sont capables de satisfaire les besoins actuels des utilisateurs, et facilitent l'accomplir des tâches et on offrant plusieurs services.

De nouveaux usages sont apparus tels que La messagerie instantanée est devenu très populaire aujourd'hui .C'est le moyen de communiquer en privé avec d'autres personnes de son choix.

L'utilisateur se connecte à un serveur qui contient les informations sur tous les utilisateurs inscrits. 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 et peuvent modifier le mot de passe.

L'objectif de notre travail est la conception et le développement d'une application Android pour «la messagerie instantanée ». Deux ou plusieurs utilisateurs de terminaux Android doivent avoir une interface permettant d'écrire des messages afin de les transmettre vers d'autres utilisateurs via le Wi-Fi.

Aussi, tous les messages échangés doivent être sauvegardés dans une base de données au niveau d'un serveur.

Notre projet est composé de cahier de charge et deux chapitres principaux structurés comme suit :

Chapitre 1 (Conception) : sera consacré à la conception de notre application, en utilisant le langage UML à travers des diagrammes.

Chapitre 2 (Réalisation) : on va présenter les outils utilisés pour la programmation, et les tables de la base de données relationnelles et expliquer le fonctionnement de notre application en utilisant quelques images des interfaces finies.

Nous terminons avec une conclusion générale...

Dans ce rapport nous présenterons les différentes phases que nous avons effectuées pour élaborer ce projet et donner une application évoluée qui suit les nouvelles technologies dans le développement de logiciel.

Cahier de Charge

1. Objectifs :

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

Un système de messagerie instantanée permet d'échanger des messages textuels ou des fichiers (images, vidéo, son...) en temps réel entre plusieurs utilisateurs connectés à un même réseau informatique. Contrairement au courrier électronique, ce moyen de communication permet de conduire un dialogue interactif.

La messagerie instantanée requiert l'emploi d'un logiciel client qui se connecte à un serveur de messagerie instantanée. Elle diffère du courrier électronique du fait que les conversations se déroulent instantanément (quasiment en temps réel, les contraintes temporelles n'étant pas fortes dans ces systèmes). La plupart des services modernes offrent un système de notification de présence, indiquant si les individus de la liste de contacts sont simultanément en ligne et leur disponibilité pour discuter.

Notre objectif est de réaliser une application mobile pour vous permettre d'envoyer et recevoir des messages et d'appeler vos amis et votre famille ...

Cette application permettra de réaliser les différentes opérations suivantes :

- ❖ **Inscription** : pour utiliser l'application, les utilisateurs peuvent s'inscrire directement via l'application Instant Chat en entrant le nom, et leur numéro de téléphone mobile ou leur adresse e-mail et le mot de passe.
- ❖ **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 :
 - **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.

- ❖ **Gestion de Présence** : les utilisateurs peuvent consulter les utilisateurs en ligne et afficher ces derniers.
- ❖ **Option de Partage** : la possibilité de diffuser un message, vers tous les utilisateurs connectés.
- ❖ **Changer Mot de Passe** : les utilisateurs peuvent changer ou modifier leurs mot de passe, si vous êtes oublié le mot de passe appuyer sur **Mot de passe oublié ?** saisissez l'adresse e-mail, et suivez les instructions qui s'affichent.
- ❖ **Ajouter Compte** : chaque utilisateur possède un compte ou plusieurs, et chaque compte possède un mot de passe pour se connecter au service.
- ❖ **Supprimer un Compte** : noter que vous pouvez aussi supprimer des comptes que vous avez ajoutés sur la base de données.
- ❖ **Discussion** : est un échange de messages entre deux interlocuteurs, à savoir vous et un de vos contacts.
- ❖ **Discussion en Groupe (Conférence)** : À la différence d'une discussion à plusieurs, dans une conférence, tous les participants voient ce que disent les autres. Chaque message tapez 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.
- ❖ **Autre fonctionnalités relatif à la base de données** :
 - Une fonctionnalité de sauvegarde automatique des messages dans une base de données.
 - Une fonctionnalité d'affichage des messages précédents (à partir de la base de données).
 - Une fonctionnalité de suppression de l'historique des messages (de la base de données).

Chapitre 01

Conception



1. Introduction :

Dans ce chapitre nous avons présenté la conception de notre projet, en utilisant les différents diagrammes d'UML. D'abord, on présente le diagramme de cas d'utilisation et de séquences pour spécifier les besoins fonctionnels de notre système. Après, nous avons réalisé le diagramme de classes pour la conception de la base de données. Nous avons utilisé StarUML comme outil de modélisation de nos trois diagrammes choisis dans notre conception.

2. Langage de modélisation unifié(UML) :

2.1.Définition :

UML est un langage de modélisation très complet, qui couvre de nombreux aspects du développement des logiciels, comme les exigences, l'architecture, les structures et les comportements. UML(Unified Modeling Language) est aujourd'hui le langage de modélisation d'applications informatiques le plus important du marché.



Ce langage est utilisé pour faire des diagrammes décrivant des

Programmes Informatiques en termes des éléments constitutifs. UML n'est pas limité à l'ingénierie du logiciel, mais est également utilisé dans l'ingénierie des systèmes. La représentation des structures organisationnelles peut être unifiée par la modélisation du langage UML. La mise en place de logiciels de conception intégrée pour un projet peut aussi être à l'acte (Diagrammes) d'un code initial du programme.

2.2.Les différents types diagrammes d'UML :

A partir de la version 2.0, UML a défini quatre nouveaux diagrammes ainsi qu'une nouvelle structuration de sa collection de treize diagrammes : structure, comportement et interaction comme suit :

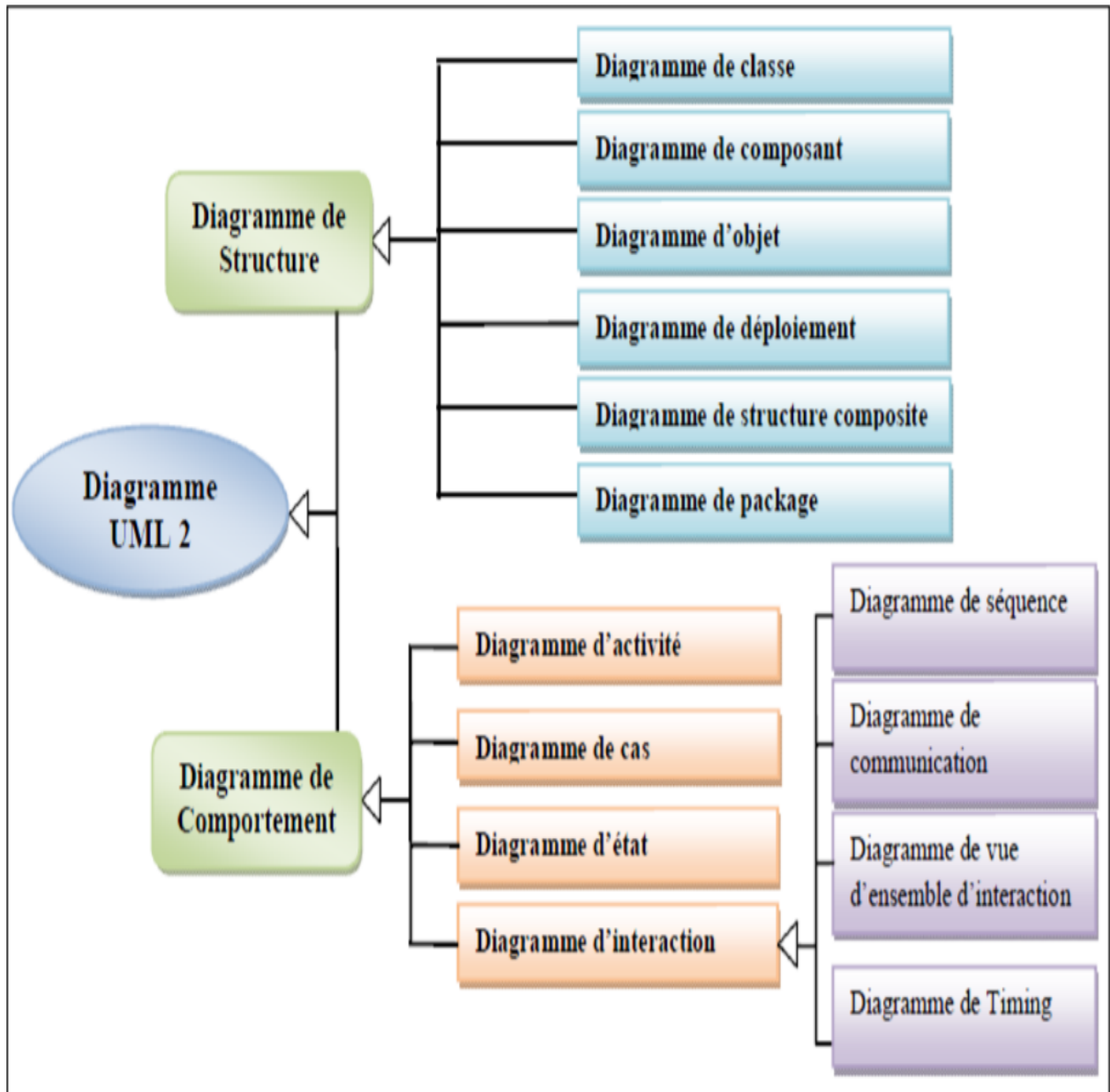


Figure 2: Les diagrammes UML.

3. Les diagrammes utilisés dans notre projet :

3.1. Diagramme de contexte :

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

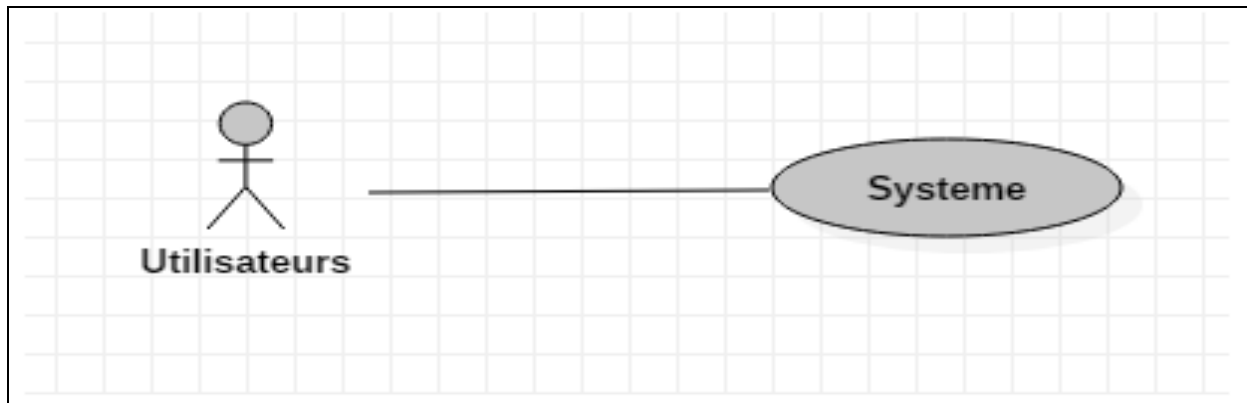


Figure 3 : Diagramme de Contexte

3.2. Diagramme des cas d'utilisations :

3.2.1. Définition :

Diagramme de cas d'utilisation est le premier diagramme du modèle UML. Il permet de représenter les interactions fonctionnelles entre les acteurs et le système étudié. Il est particulièrement important pour l'organisation et l'identification des grandes fonctionnalités du système.

On peut avoir différents types de relations entre les cas d'utilisations :

- ✚ Généralisation : le cas d'utilisation fils est une spécialisation du cas père.
- ✚ Inclusion : une instance du cas source comprend le comportement décrit par le cas Cible.
- ✚ Extension : le cas source ajoute son comportement au cas destination ; l'extension est soumise à la vérification d'une condition (point d'extension).

3.2.2. Identification des acteurs :

Un acteur est Une entité quelconque ayant un comportement qui inclut d'autres systèmes faisant appel au système modélisé ou auquel le système modélisé fait appel. Plus généralement, il s'agit d'une catégorie de personnes qui peut agir comme utilisateur du système et interagir avec ce dernier (Actor).

Dans notre système, nous pouvons identifier *l'utilisateur* comme un acteur.

3.2.3. Diagramme de cas d'utilisation globale :

L'application *Let's Talk*, permet d'échanger des messages en temps réel entre plusieurs utilisateurs connecté via WI-FI.

L'utilisateur, peut accéder à l'application ,s'il possède un compte , il peut effectuer un Log in (saisir son adresse e-mail et le mot de passe), et avoir ainsi des service qu'il peut effectuer comme modifier le mot de passe d'un compte , ajouter plusieurs compte ,supprimer un compte ,ajouter des amis(envoyer et recevoir les invitations) ,comme il peut envoyer des messages en indiquant la destination parmi sa liste d'amis, et aussi peut recevoir des message et effectuer un Log off, sinon il s'inscrit .

Dans ce cas le système affiche un formulaire d'inscription qui contient un nom et adresse e-mail, mot de passe et la confirmation de mot de passe, à condition que l'adresse e-mail n'existe pas dans la base de données

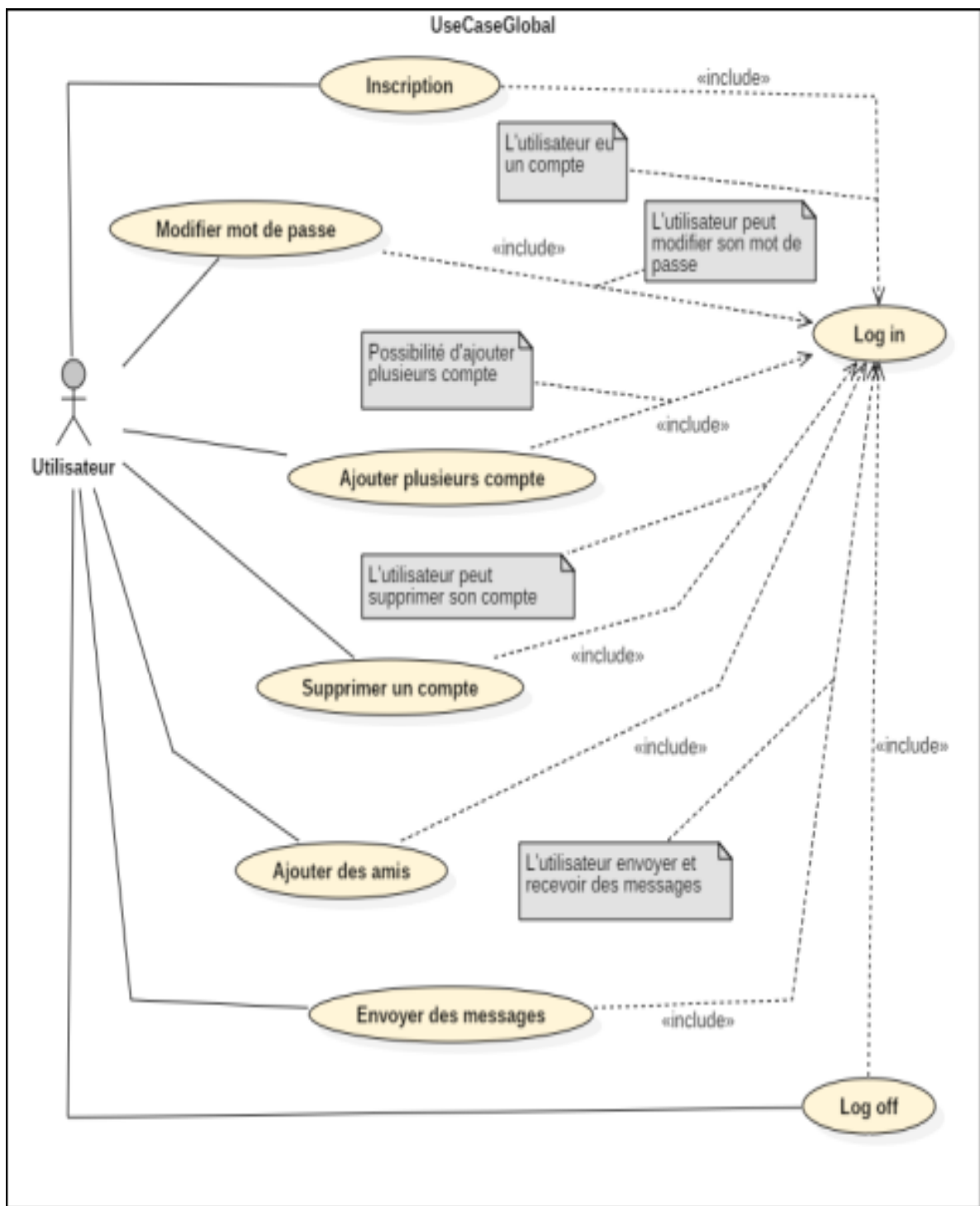


Figure 4 : Diagramme de cas d'utilisation globale.

3.3.Diagramme de Séquence :

3.3.1. Définition :

Représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs. Un diagramme de séquence représente en général un seul ‘scénario’ de cas d’utilisation ou flux d’évènements.

✚ **Scenario** : Représente une succession particulière d'enchaînements, s'exécutant du début à la fin du cas d'utilisation, un enchaînement étant l'unité de description de séquences d'actions.

✚ **Ligne de vie** : Représente l’ensemble des opérations exécutées par un projet

✚ **Message** : Un message et une transmission d’information unidirectionnelle entre deux objets, l’objet émetteur et l’objet récepteurs. Dans un diagramme de séquences deux types de message peuvent être distingués :

- Message synchrone.
- Message asynchrone.

3.3.2. Diagramme de séquence « Inscription /Sign Up » :

Titre	Inscription.
Pré condition	Aucun
Séquence	<ul style="list-style-type: none">➤ Demande le formulaire d’inscription➤ Le système affiche le formulaire (le nom, e-mail, mot de passe et la confirmation de mot de passe)➤ L’utilisateur saisit les informations.➤ L’utilisateur valide les informations.➤ Le système vérifie la confirmation des informations fournis.
Post condition	L’utilisateur s’inscrire.
Chemin Alternatif	Dans le cas où les informations sont incorrectes ou incomplètes, le système réaffiche le formulaire d’inscription et attend que l’utilisateur ressaisisse ses informations.

Tableau 1 : Description de cas d’utilisation « Inscription »

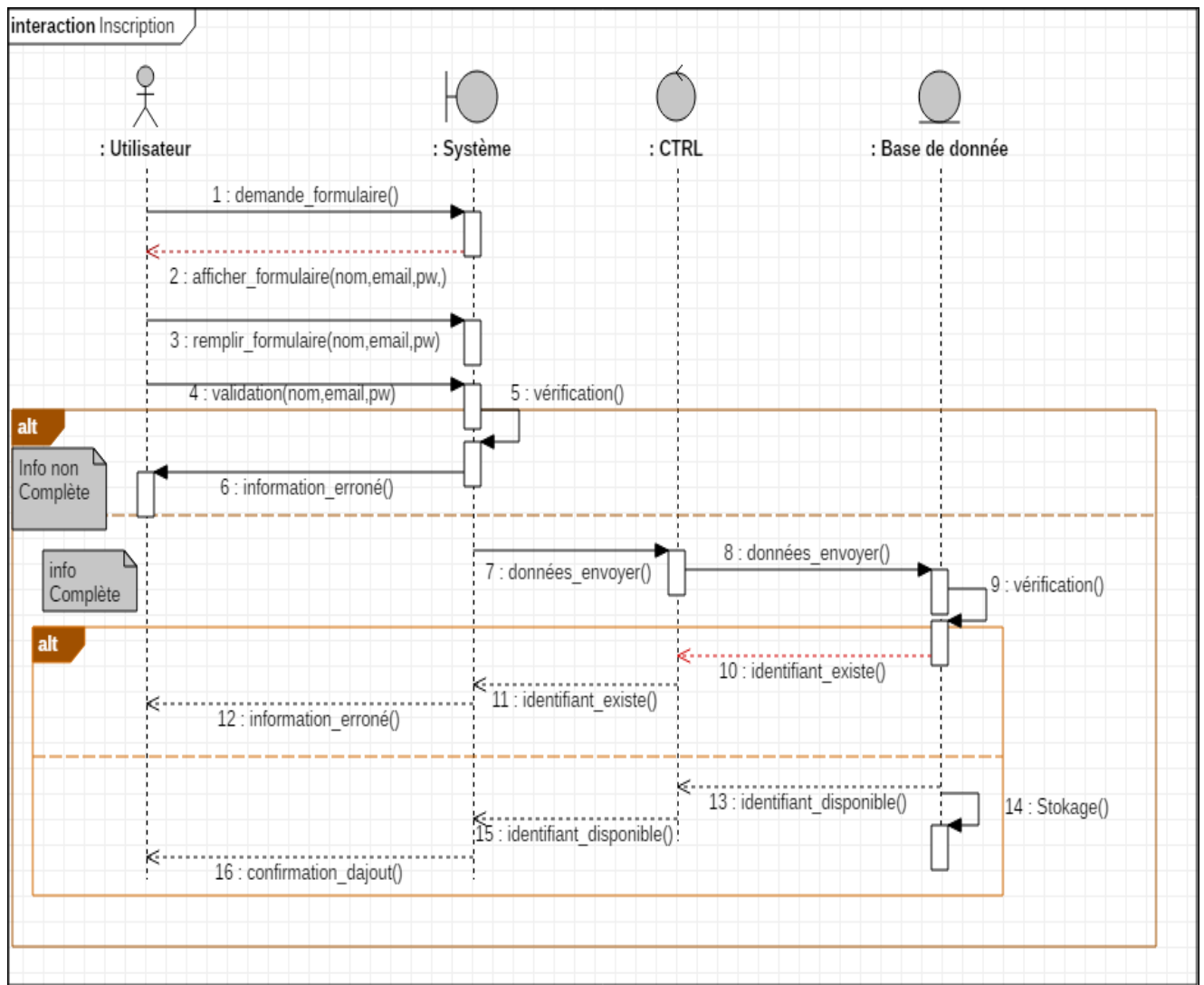


Figure 5 : Diagramme de séquence « Inscription ».

3.3.3. Diagramme de séquence « Log In » :

Titre	Log In
Pré condition	L'utilisateur inscrit.
Séquence	<ul style="list-style-type: none"> ➤ Demande de connexion. ➤ Le système affiche le formulaire de log in (demande l'adresse e-mail, et le mot de passe) ➤ L'utilisateur saisit les informations. ➤ L'utilisateur valide les informations. ➤ Le système vérifier la confirmation des informations fournis.

	➤ Le système donne l'accès à la page suivante.
Post Condition	L'utilisateur accéder au système.
Chemin Alternatif	Dans le cas où les informations sont incorrectes ou incomplètes, le système réaffiche le formulaire d'inscription et attend que l'utilisateur ressaisisse ses informations.

Tableau 2 : Description de cas d'utilisation « Log In »

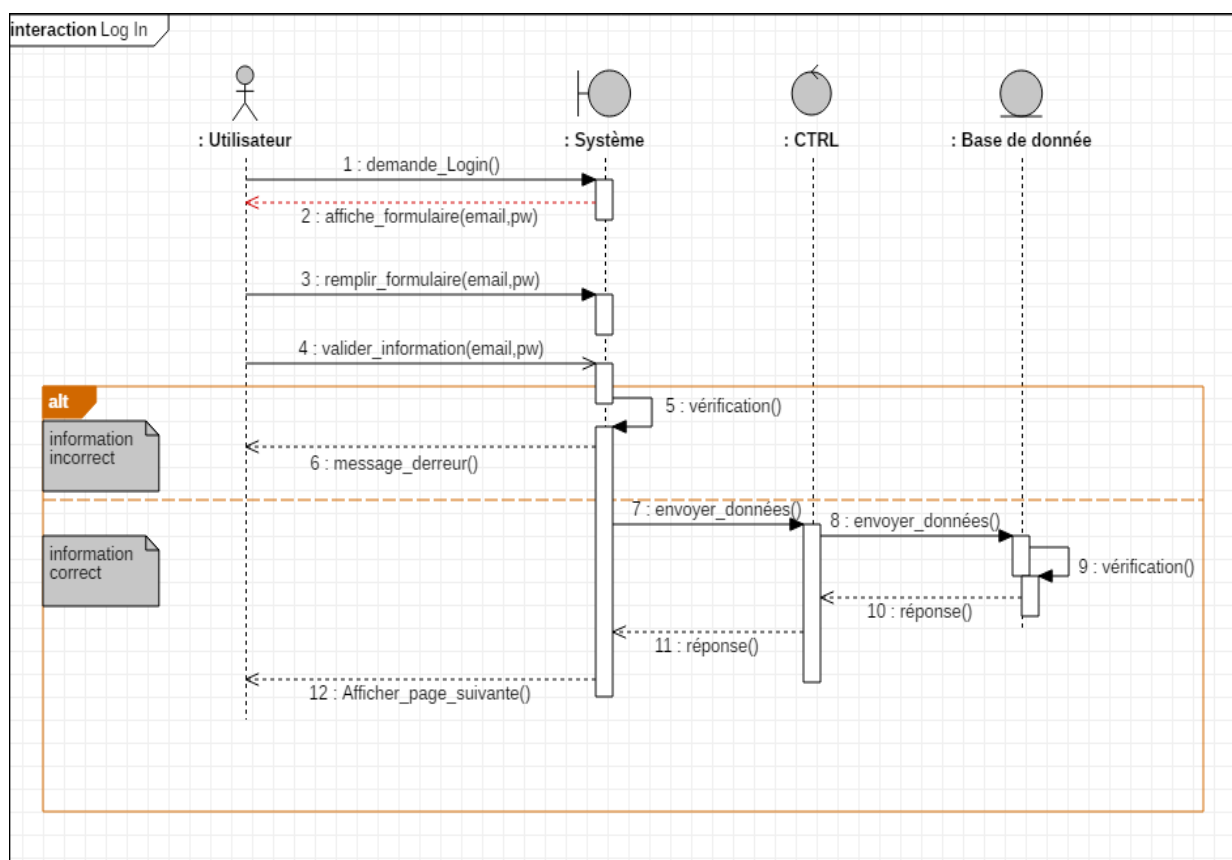


Figure 6: Diagramme de séquence « Log In ».

3.3.4. Diagramme de séquence « Envoyer des messages »

Titre	Envoyer message.
Pré-Condition	<ul style="list-style-type: none"> ➤ Log in ➤ Le récepteur doit être dans la liste des amis

Séquence	<ul style="list-style-type: none"> ➤ l'utilisateur choisir l'ami dans la liste ➤ le système s'affiche l'interface des messages ➤ l'utilisateur saisie son message, après il clique sur envoyer.
Post Condition	Aucun.
Chemin Alternatif	Aucun.

Tableau 3 : Description de cas d'utilisation « Envoyer des messages »

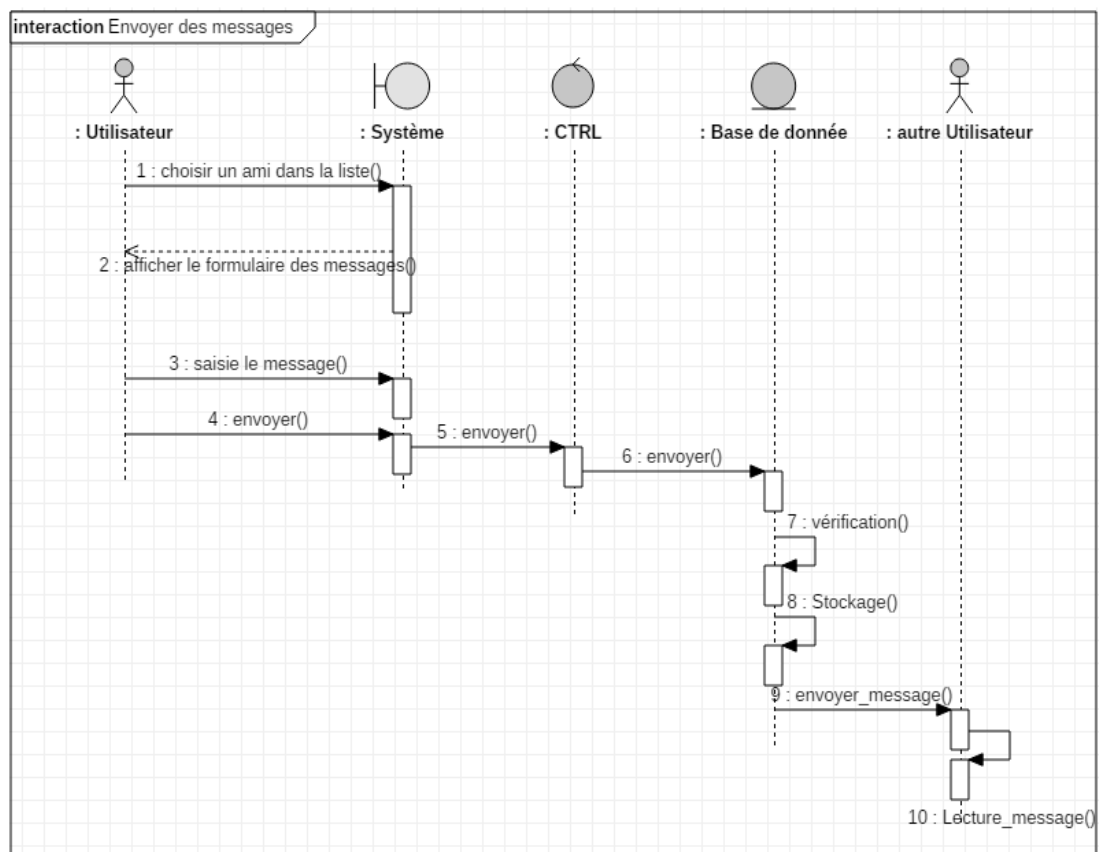


Figure 7 : Diagramme de séquence « Envoyer des messages »

3.3.5. Diagramme de séquence « Changer le mot de passe »

Titre	Changer le mot de passe
Pré condition	Aucun
Séquence	<ul style="list-style-type: none"> ➤ L'utilisateur entre son ancien mot de passe avec le nouveau ➤ Le système vérifier l'ancien mot de passe

	➤ Le système effectue le nouveau mot de passe au compte de ce utilisateur
Post Condition	Aucun
Chemin Alternatif	Aucun

Tableau 4 : Description de cas d'utilisation « Changer le mot de passe »

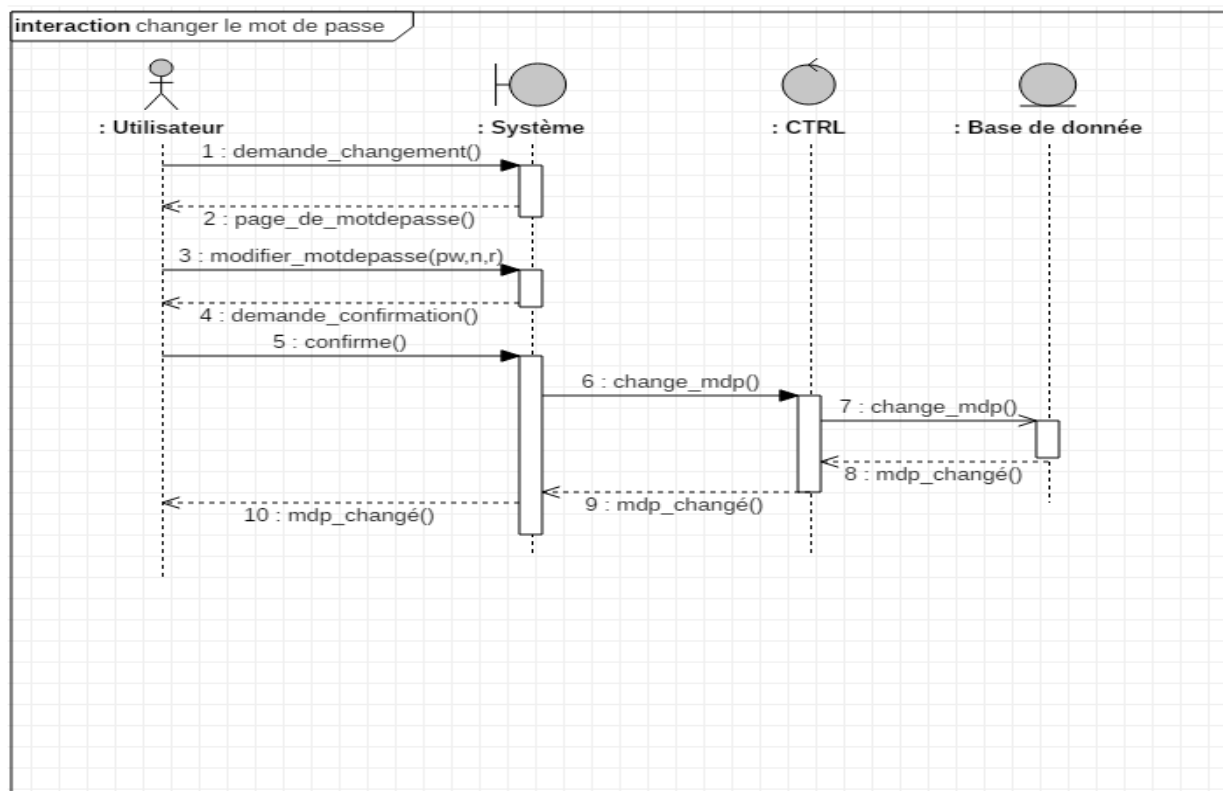


Figure 8 : Diagramme de séquence « Changer le mot de passe ».

3.3.6. Diagramme de séquence « Log Off»

Titre	Log Off
Pré condition	Aucun.

Séquence	<ul style="list-style-type: none"> ➤ Accès à l'application. ➤ Accès au menu de l'application. ➤ Clique sur le bouton « Déconnexion ». ➤ Le système valide la confirmation. ➤ L'application renvoi automatiquement la page d'accueil.
Post Condition	L'utilisateur déconnecté
Chemin Alternatif	Aucun.

Tableau 5 : Description de cas d'utilisation «Log Off»

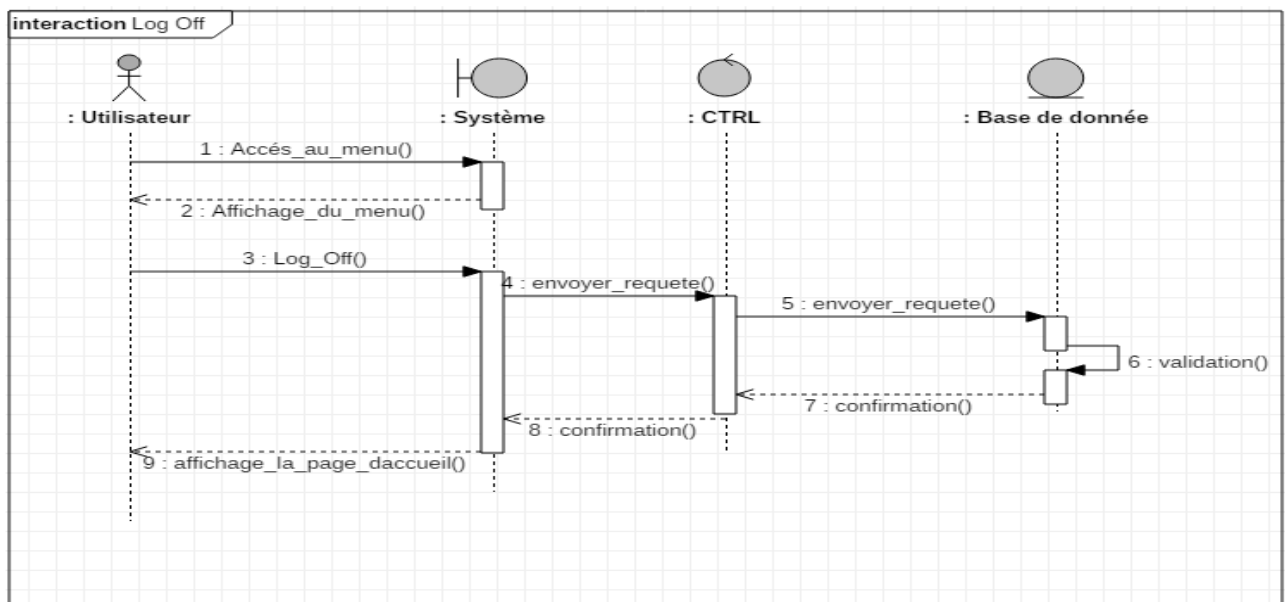


Figure 9 : Diagramme de séquence « Log Off ».

4. Diagramme de classe :

4.1.Définition :

Un diagramme de classe exprime de manière générale la structure statique du système en termes de classes et de relations entre classes. Un diagramme de classes regroupe généralement les éléments de modélisation suivants : Associations, Agrégations, Classes, Compositions, Généralisations, Interfaces, Interface-Réalisations, Réalisations.

4.2. Notre Diagramme de Classe :

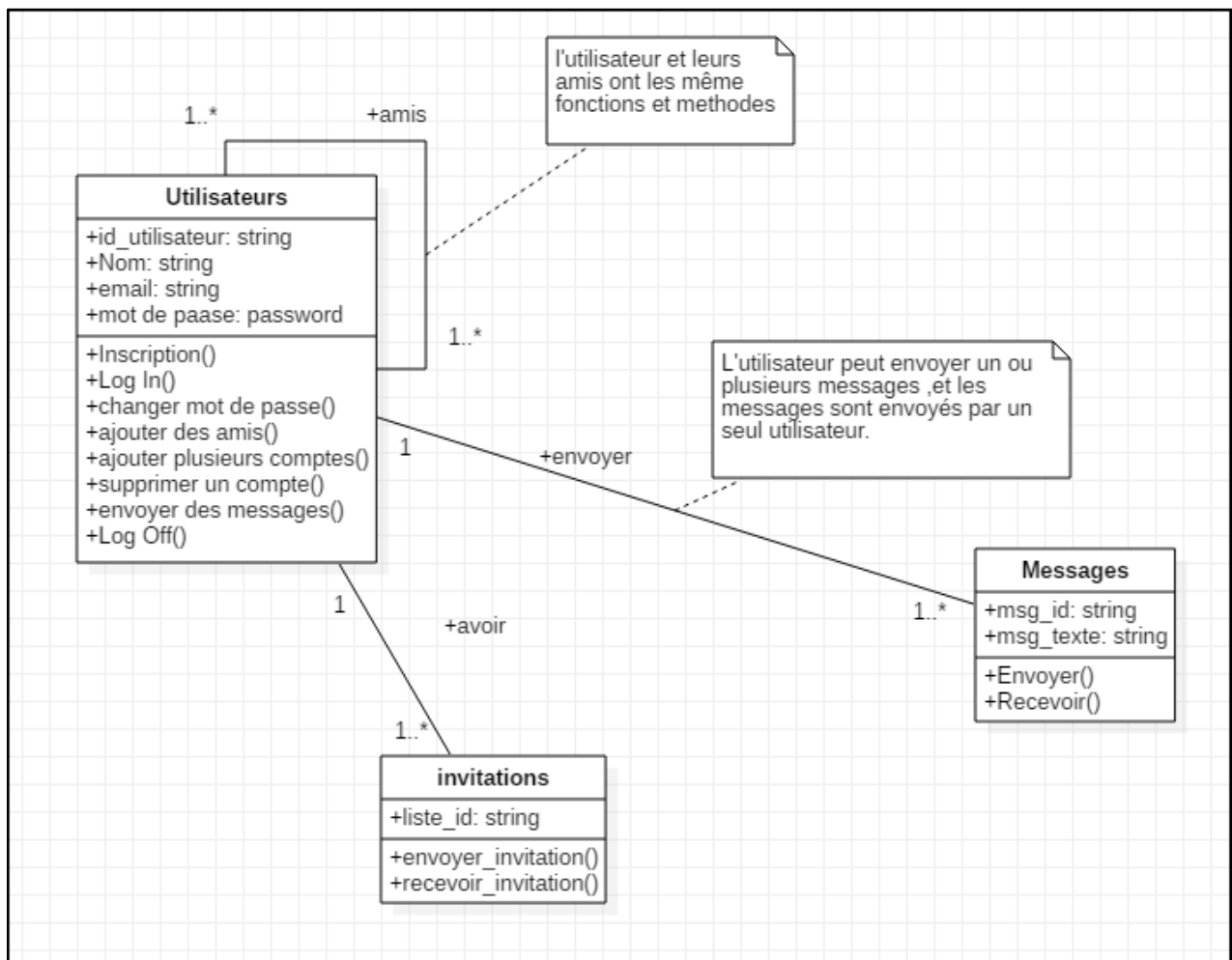


Figure 10 : Diagramme de classe.

5. Conclusion :

Dans ce chapitre, nous avons essayé de développer la partie conception d'une application « la messagerie instantanée » à l'aide d'UML. Et on a proposé quelques diagramme d'UML pour associe au cette application et des descriptions pour mieux comprendre la fonctionnalité de cette applications.

Chapitre 02

Réalisation



1. Introduction :

Dans ce chapitre on va représenter l'environnement matériel et logiciel que nous avons utilisé pour réaliser notre application, nous expliquerons éventuellement l'environnement et les outils de développement de notre application ,ainsi que la manipulations effectuées sur la base de données). Et on a décrit la partie pratique et graphique de notre projet, on donnant ensuite quelque capture d'écrans sur les interfaces dans leur état final aussi une partie de code source

2. Environnement de travail :

Il y'a deux environnements précisés sont : Environnement logiciel et matériel.

2.1.Environnement matériel :

Notre application« *la messagerie instantanée* »est réalisée par les ordinateurs suivant :

- Ordinateur 01 :
 - la marque : *hp*
 - Processeur : intel®core ™ [i5-6200U CPU @2.30 GHz](#) 2.30GHz
 - Mémoire : 8 Go RAM.
 - Système d'exploitation : Windows 8.
 - smart phone : Samsung SM-J700H Android 6.0.1, API 23
- Ordinateur 02 :
 - la marque : *Toshiba*
 - Processeur : intel®core ™ [i3-5005U CPU @2.00 GHz](#) 2.00GHz
 - Mémoire : 4 Go RAM.
 - Système d'exploitation : Windows 10.
 - smart phone : Samsung SM-J700H Android 6.0.1, API 23

2.2.Environnement logiciel :

Pour la réalisation de notre application, nous avons au recours différents logiciel pour les manipulations sur la base de données, la modélisation des diagrammes, et les logiciels utilisés pour éditer notre projet.

2.2.1. Logiciel de développement de l'application :

Android studio (3.4.1) : est un environnement de développement pour développer des applications mobiles Android. Il est basé sur IntelliJ IDEA et utilise le moteur de production Gradle. Il peut être téléchargé sous les systèmes d'exploitation Windows, macOS et

Linux.Android Studio permet principalement d'éditer les fichiers Java/Kotlin et les fichiers de configuration XML d'une application Android. Il propose entre autres des outils pour gérer le développement d'applications multilingues et permet de visualiser rapidement la mise en page des écrans sur des écrans de résolutions variées simultanément. Il intègre par ailleurs un émulateur permettant de faire tourner un système Android virtuel sur un ordinateur.

JAVA :

Java est une technique informatique développée initialement par Sun Microsystems puis acquise par Oracle suite au rachat de l'entreprise. Défini à l'origine comme un langage de programmation, Java a évolué pour devenir un ensemble cohérent d'éléments techniques et non techniques

Java est un des termes les plus connus du monde de l'informatique et de l'Internet. Sa notoriété est telle que Sun, avant d'être acheté par Oracle Corporation, a décidé de l'utiliser pour son symbole boursier au Nasdaq, symbole qui était SUNW à l'origine, et est devenu JAVA. est utilisé dans une grande variété de plates-formes depuis les systèmes embarqués et les téléphones mobiles, les ordinateurs individuels, les serveurs, les applications d'entreprise, les superordinateurs, etc.

XML :

L'*Extensible Markup Language*, généralement appelé **XML**, « langage de balisage extensible » en français, est un métalangage informatique de balisage générique qui est un sous-ensemble du Standard Generalized Markup Language (SGML). Sa syntaxe est dite « extensible » car elle permet de définir différents langages avec chacun leur vocabulaire et leur grammaire, comme XHTML, XSLT, RSS, SVG... Elle est reconnaissable par son usage des chevrons (<, >) encadrant les noms des balises. L'objectif initial de XML est de faciliter l'échange automatisé de contenus complexes (arbres, texte enrichi, etc.) entre systèmes d'informations hétérogènes (interopérabilité). Avec ses outils et langages associés, une application XML respecte généralement certains principes :

- la structure d'un document XML est définie et validable par un schéma ;
- un document XML est entièrement transformable dans un autre document XML.

2.2.2. Logiciel pour éditer la base de données :

Firebase : est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, Ios, JavaScript, Node.js, Java, Unity, PHP, C++ ...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel. Lancé en 2011 sous le nom d'Envolv, par Andrew Lee et par James Templin, le service est racheté par Google en octobre 2014. Il appartient aujourd'hui à la maison mère de Google : Alphabet.

Toute l'implémentation et la gestion serveur de Firebase est à la charge exclusive de la société Alphabet. Les applications qui utilisent Firebase intègrent une bibliothèque qui permet les diverses interactions possibles.

2.2.3. Logiciel de modélisation :

Star UML : est un logiciel de modélisation UML, cédé comme open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL. StarUML gère la plupart des diagrammes spécifiés qui donnent chacun une vision différente de notre projet.

2.2.4. Logiciel pour éditer le cahier de charge :

Microsoft Office Word 2010 : est une application sous Windows utilisée pour éditer les traitements de texte et ses différents formats.

3. Présentation des Activités (des interfaces) de l'application :

Dans cette partie on va représenter les différentes interfaces de notre application et on va faire une petite explication de chaque étape.

Au premier lancement de l'application , la première interface que s'affiche c'est le logo de l'application .

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

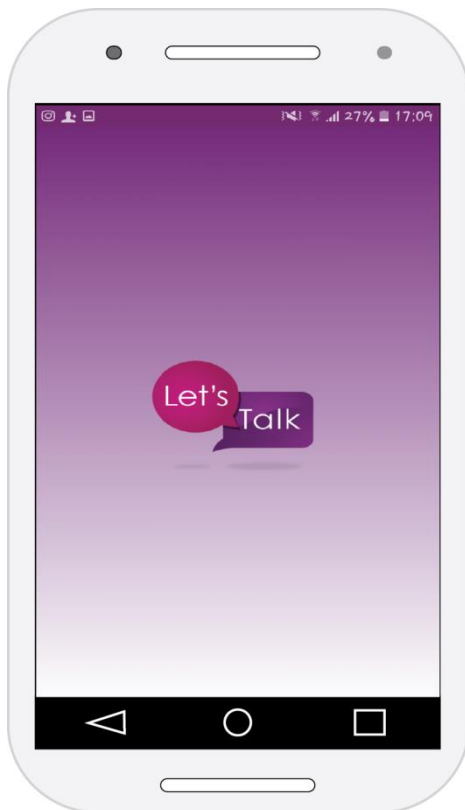


Figure 11: Page d'accueil

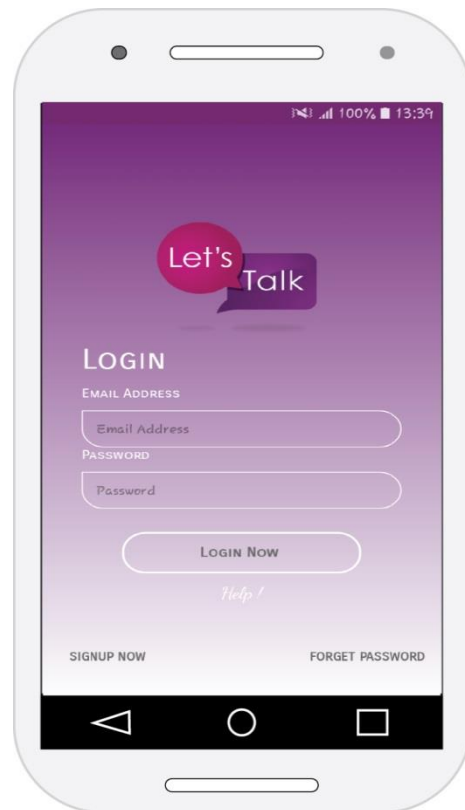


Figure 12:Log in

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

Sur cette interface, l'utilisateur doit saisir un nom (username), un email et un mot de passe et confirme le mot de passe puis leur numéro de téléphone .. puis cliquer sur le bouton inscrire (sign up).

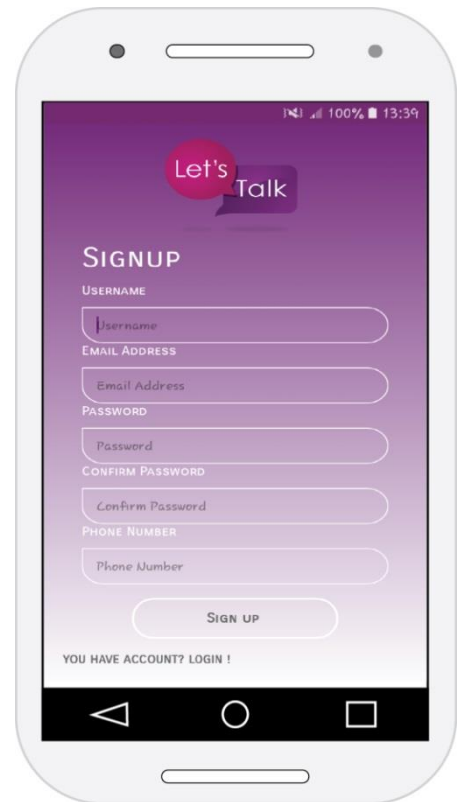


Figure 13: Sign up

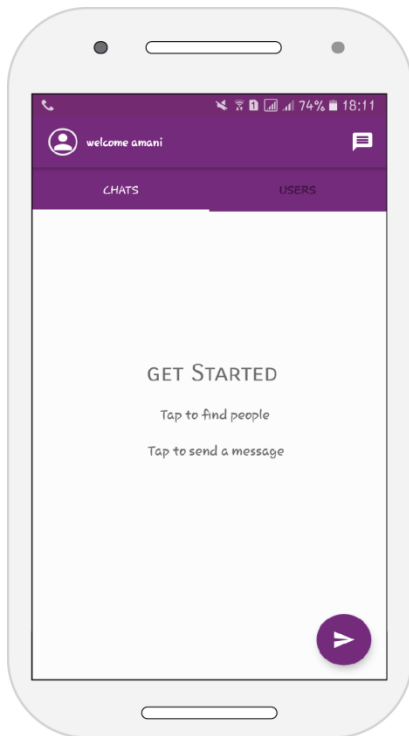


Figure 14 : Interface de main

La première fois que l'utilisateur se connecte à son compte, cette interface s'affiche et lui indique de trouver ses amis et leur envoyer des messages.

L'utilisateur peut accéder à une petite page Help pour lui montrer comment utiliser l'application :

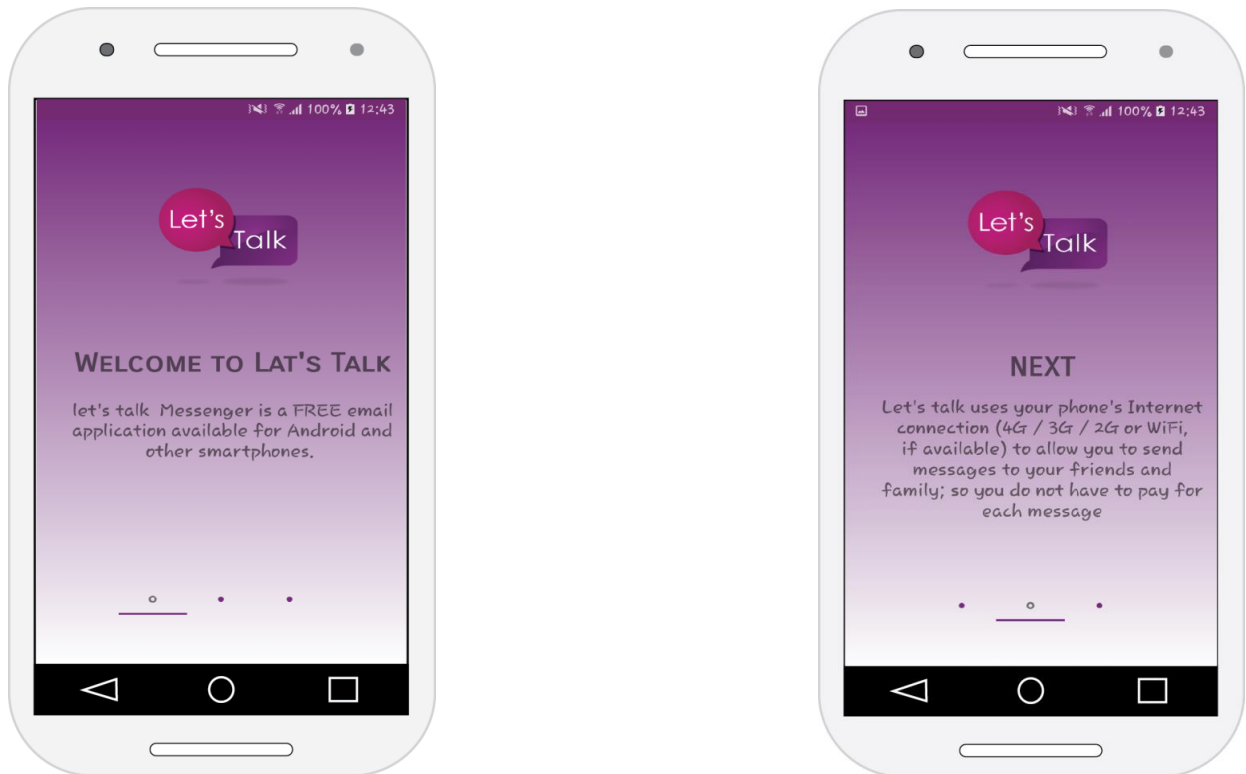
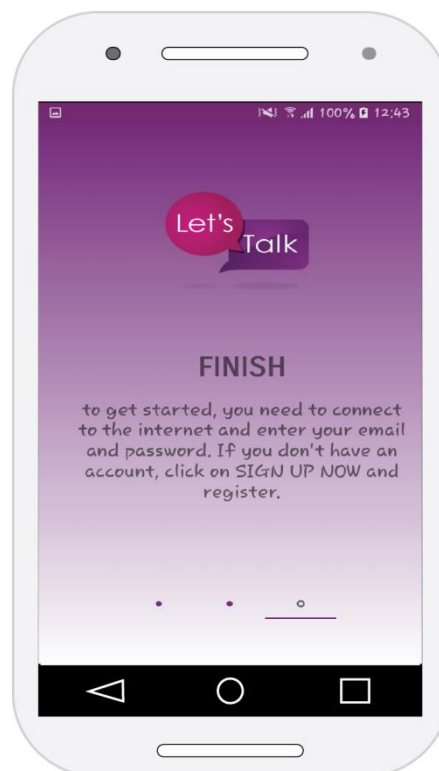


Figure 15: interface de Help



Passons maintenant aux paramètres(Settings) ,cette partie est concernant le coté paramètres de l'utilisateur ..

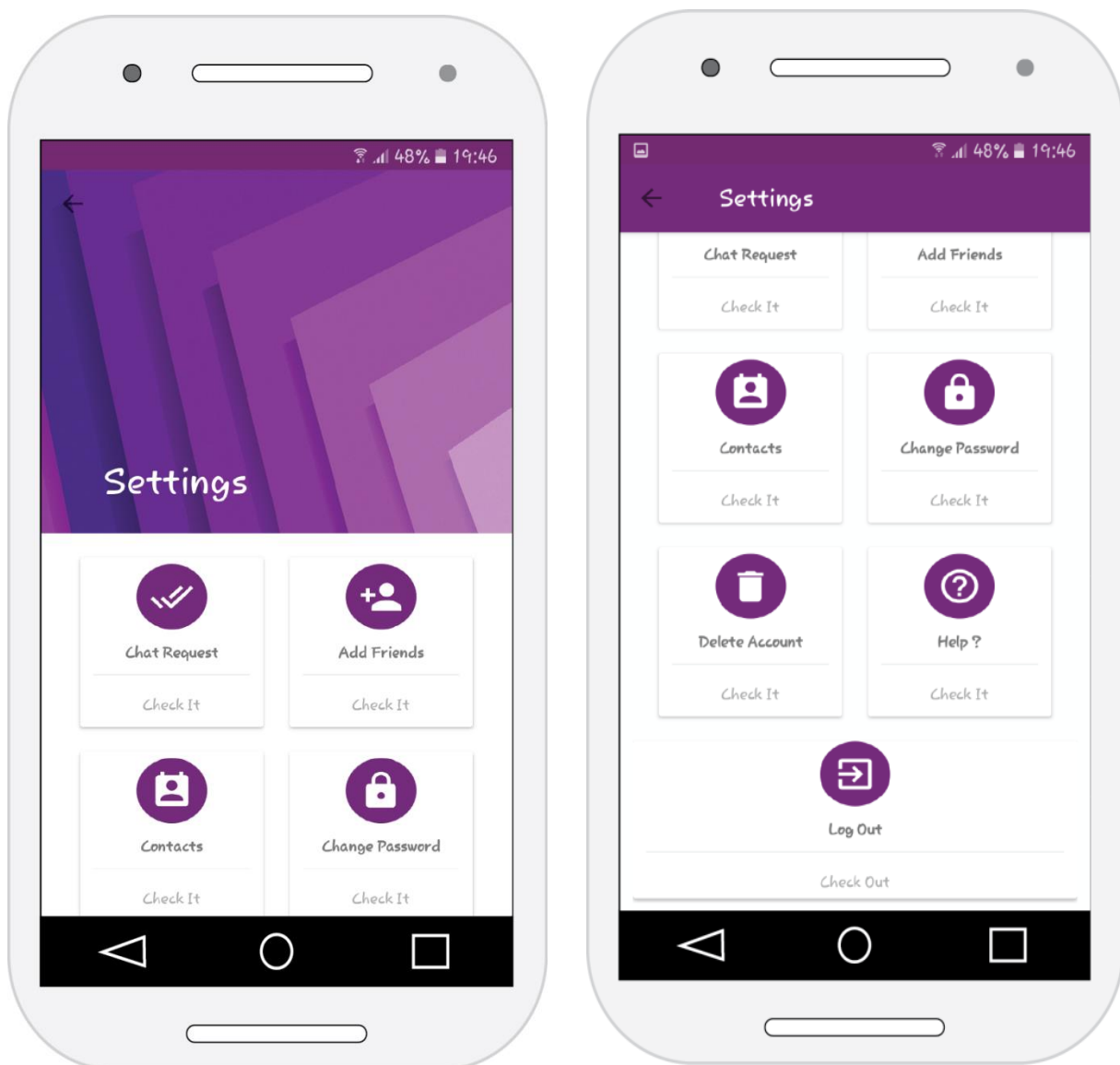


Figure 16: Interface de Paramètre

Dans cette interface de (Delete Account), Lorsque vous aurez supprimé votre compte, vous ne pourrez plus récupérer le contenu ni les informations que vous avez partagés .

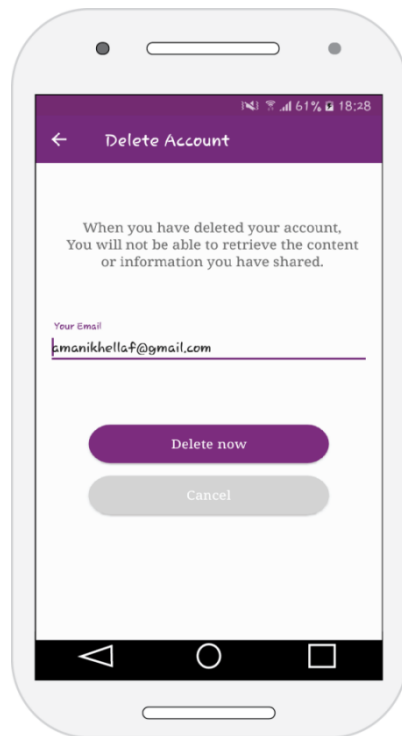


Figure 17: Supprimer un compte

Si l'utilisateur oublié son mot de passe il peut le trouver par l'adresse email .comme aussi il peut changer son mot de passe .. il saisir l'ancien mot de passe avec le nouveau et confirmer le nouveau mot de passe puis clique sur le bouton confirme. Le mot de passe va changer tout simplement.

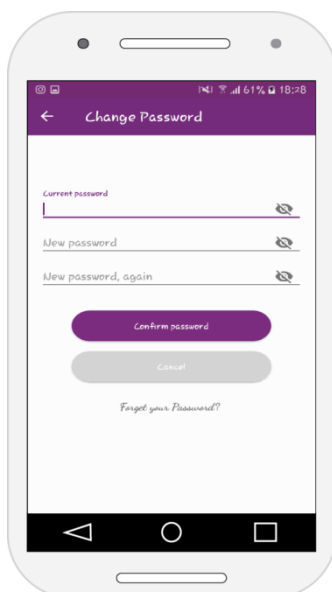


Figure 18 : Changer mot de passe

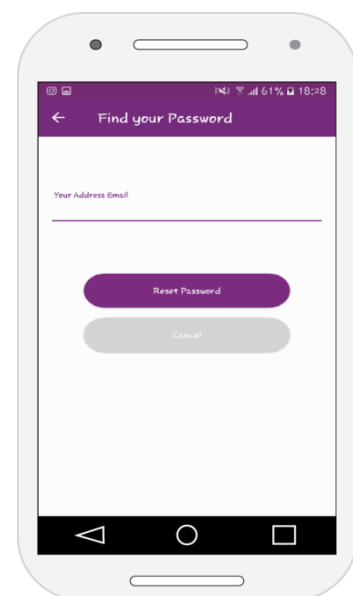


Figure 19 : Oublier mot de passe

Sur Cette interface, l'utilisateur aura la possibilité d'ajouter des amis.

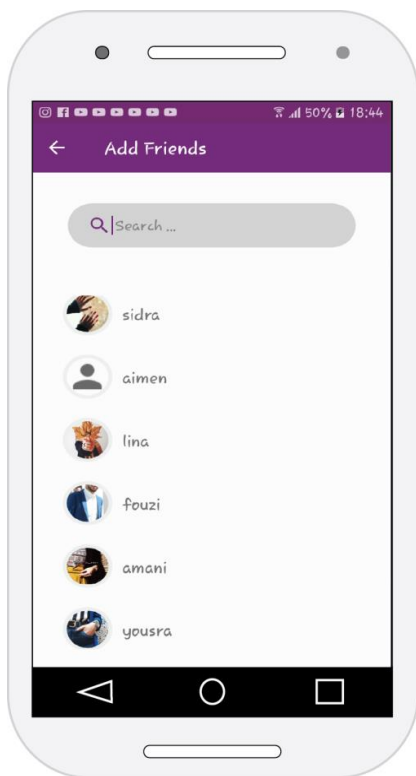


Figure 10 : Ajouter des amis

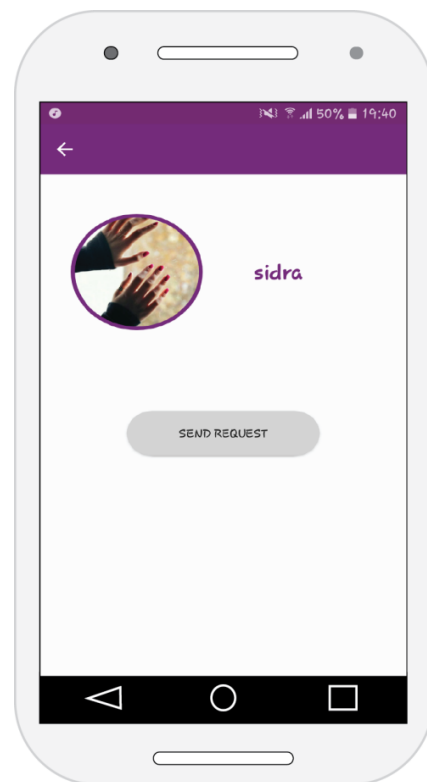


Figure 22 : Envoyer invitation

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

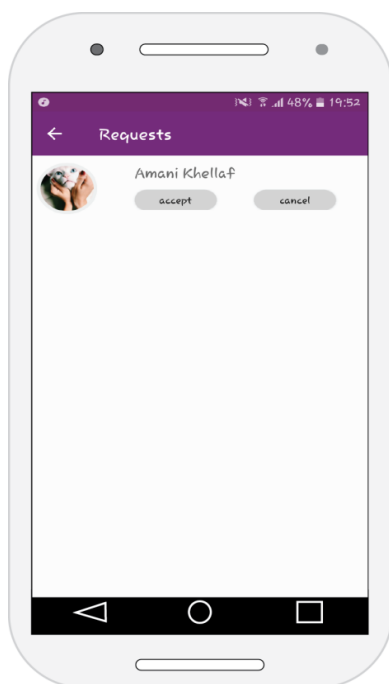


Figure 22 : Liste des invitations

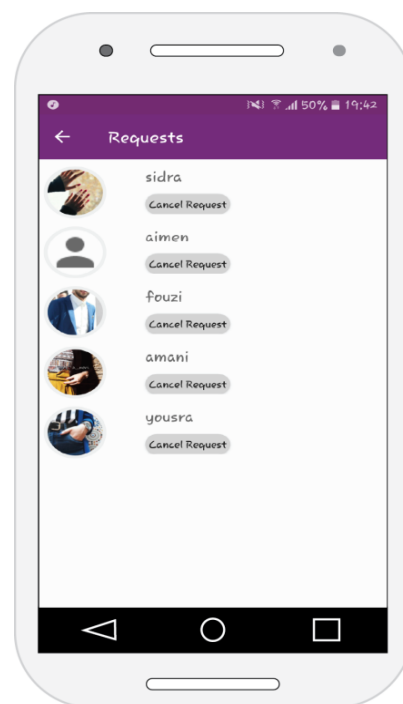


Figure 23: Les invitations envoyées

Sur cette interface, l'utilisateur pourra consulter ses messages échangés avec un correspondant, ainsi que relancer la discussion. et l'utilisateur aura un accès sur l'ensemble de ses notifications « réception de messages ».

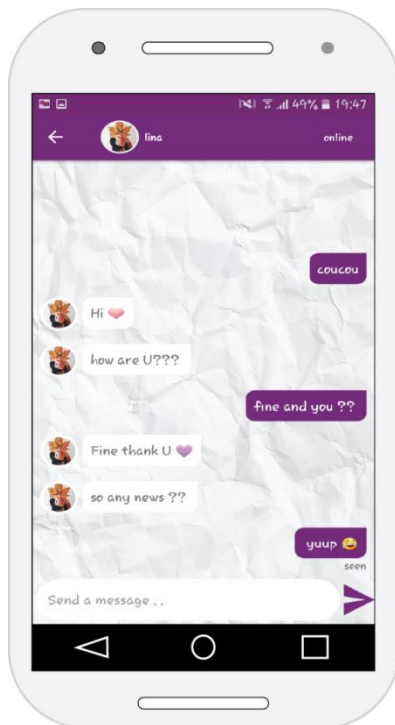


Figure24: Interface de communication

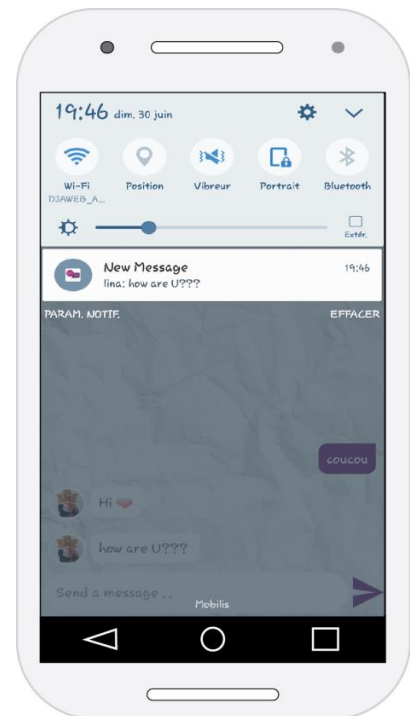


Figure 25: Interface de notification



Figure 26:Supprimer un message

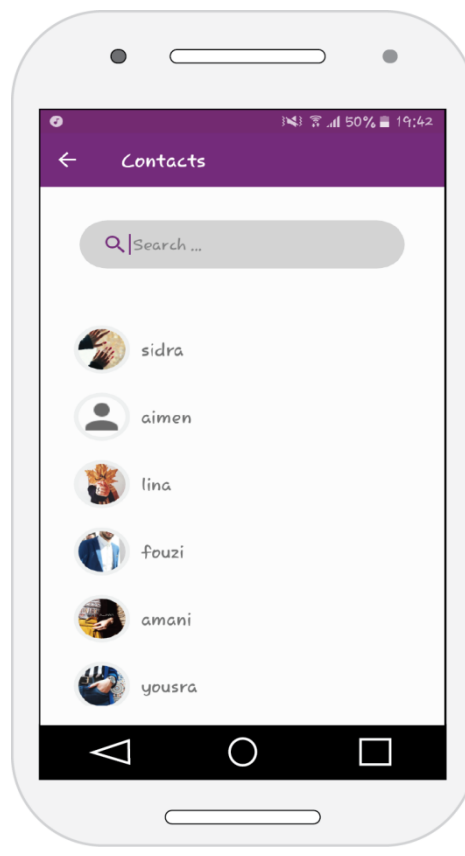
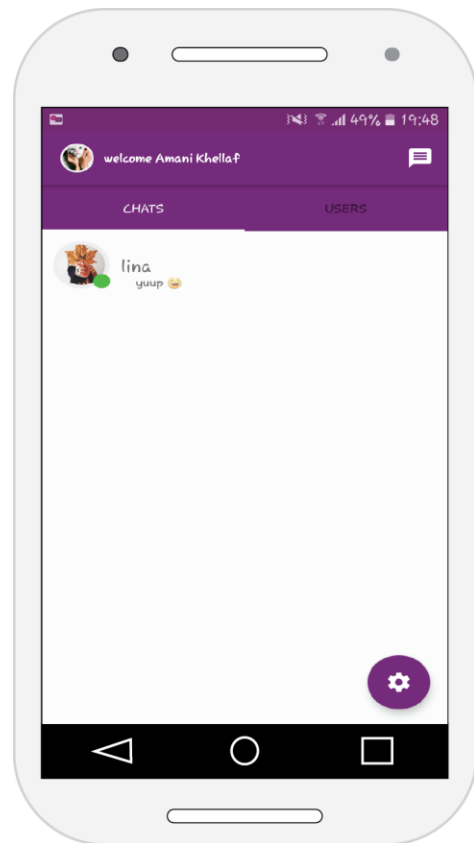
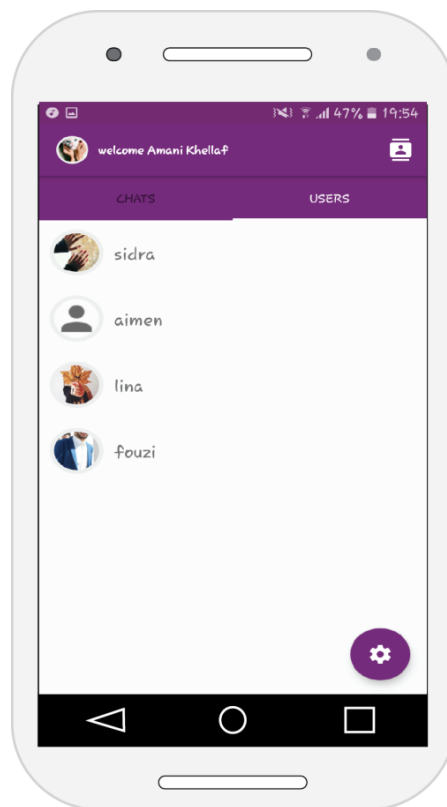
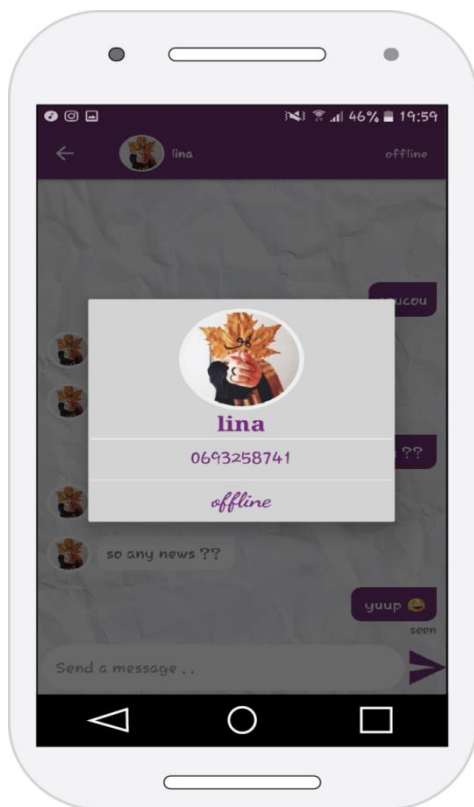


Figure 27:
Interface des contacts

Sur cette interface (Figure 26). Nous trouvons les informations personnelles de l'utilisateur l'adresse email et leur numéro de téléphone .. comme il peut mettre une photo de profile pour son compte ,et il peut la changer .

Et on a deux champs dans cette interface .. le premier champs il contient la liste des contacts (Contacts) et l'autre champs les paramètres (Settings).

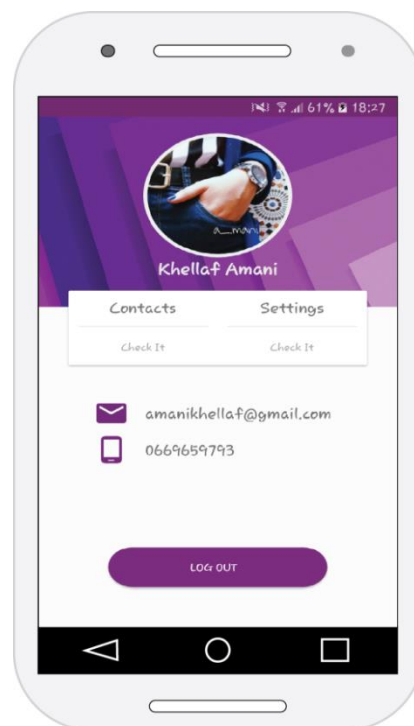


Figure 28: Mon profile

L'utilisateur peut envoyer un messages a plusieurs utilisateurs

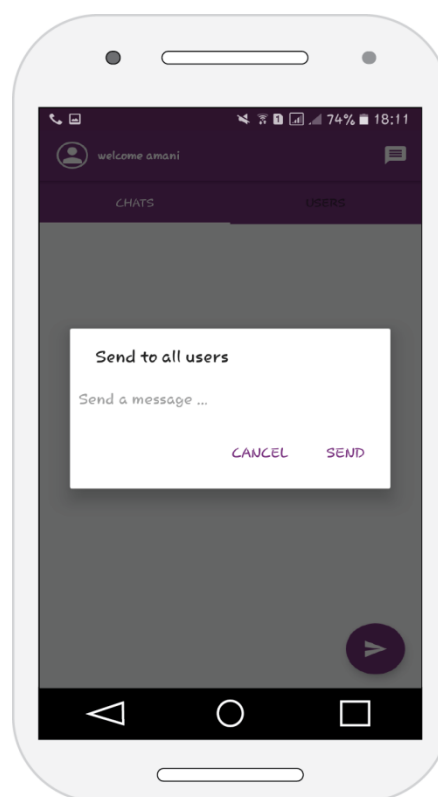
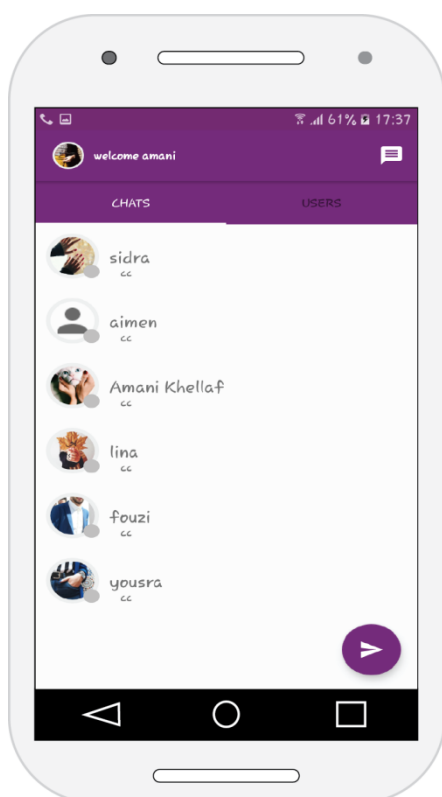


Figure 29: diffuser le message

4. Une partie de code source « Activity Forget Password »

Code JAVA :

```
package com.example.letstalk;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

public class ResetPassword extends AppCompatActivity {
    EditText send_email;
    Button btn_reset;
    Button btn_cancel;
    FirebaseAuth firebaseAuth;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_reset_password);
        // tool bar
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        setTitle("Find your Password");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayShowHomeEnabled(true);

        send_email = findViewById(R.id.send_email);
        btn_reset = findViewById(R.id.btn_reset);
        firebaseAuth=FirebaseAuth.getInstance();
        btn_cancel=findViewById(R.id.btn_cancel);

        btn_cancel.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new
Intent(getApplicationContext(),LogIn.class));
                finish();
            }
        });

        btn_reset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email =send_email.getText().toString();
                if(email.equals("")){
```



```

        Toast.makeText(ResetPassword.this, "All fileds are
required!", Toast.LENGTH_SHORT).show();
    }else {

firebaseAuth.sendPasswordResetEmail(email).addOnCompleteListener(new
OnCompleteListener<Void>() {
    @Override
    public void onComplete(@NonNull Task<Void> task) {
        if (task.isSuccessful()) {
            Toast.makeText(ResetPassword.this, "Please
check your email", Toast.LENGTH_SHORT).show();
            startActivity(new
Intent(ResetPassword.this, LogIn.class));
        }else {
            String error =
task.getException().getMessage();

Toast.makeText(ResetPassword.this, error, Toast.LENGTH_SHORT).show();
        }
    }
});
    }
}

});
}

// fonction pour reteur de tool bar
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if (item.getItemId()==android.R.id.home){
        finish();
    }
    return super.onOptionsItemSelected(item);
}
}
}

```

Code XML :

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".ResetPassword">
    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#732A79"
        android:theme="@style/Base.ThemeOverlay.AppCompat.Dark.ActionBar"
        app:popupTheme="@style/MenuStyle">
    </android.support.v7.widget.Toolbar>
    <android.support.design.widget.TextInputLayout
        android:id="@+id/current"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

        android:layout_marginTop="80dp">
        <android.support.design.widget.TextInputEditText
            android:hint=" Your Address Email"
            android:layout width="335dp"
            android:inputType="textEmailAddress"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:id="@+id/send_email"/>
    </android.support.design.widget.TextInputLayout >

    <Button
        android:id="@+id/btn_reset"
        android:text="Reset Password"
        android:textColor="#fcfcfc"
        android:textAllCaps="false"
        android:layout_width="250dp"
        android:layout_height="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/button_round"
        android:layout_marginTop="70dp" />

    <Button
        android:id="@+id/btn_cancel"
        android:text="Cancel"
        android:textColor="#fcfcfc"
        android:textAllCaps="false"
        android:layout_width="250dp"
        android:layout_height="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/button_round1"
        android:layout_marginTop="15dp" />

</LinearLayout>

```

5. Des captures sur la base de données « FireBase » :

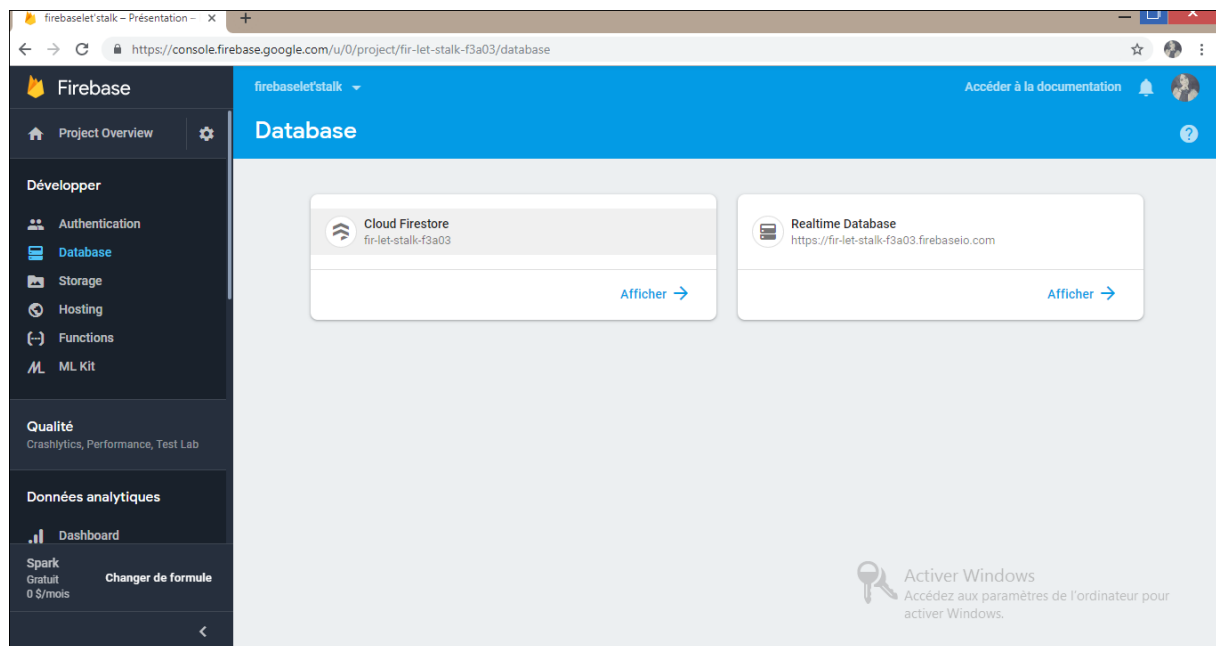


Figure 30: interface de Base de données

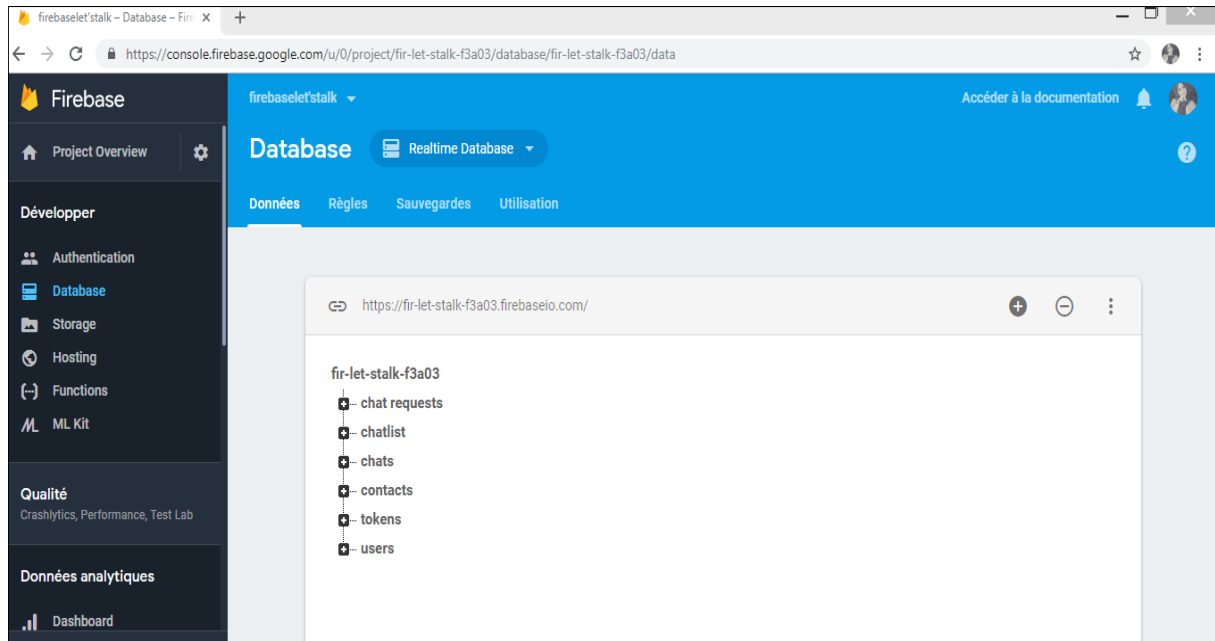


Figure 31:Arbre de la base de données

6. Référence :

<https://firebase.google.com/docs/android/setup?authuser=0>

<https://github.com/>

<https://stackoverflow.com/>

<https://medium.com/@bendaniel10/mini-project-android-edittext-validation-9bd8ab04576a>

<https://www.rapidtables.com/>

7. Conclusion Générale :

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 (FireBase), et présenter les différentes interfaces de notre application.

Malgré toutes les difficultés que nous avons rencontrées durant la réalisation de ce mémoire surtout au niveau de la dernière version 3.4.1, 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 œuvre d'une application mobile, également nous avons appris à manipuler toute les d'outils: Android studio ,FireBase... et quelques langages de programmation telles que JAVA,XML...

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

- Log in with facebook .
- Changer le langage de l'application.
- Création des groupes de discussions.
- Bloquer un Des amis.
- Envoyer des images.

Enfin, nous souhaitons que ce modeste travail satisfera les utilisateurs et apportera un maximum d'aide et pourra être bénéfique et utile pour tous les citoyens.

Nous espérons que notre travail soit généralisé pour inclure toutes les wilayas d'Algérie.