

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DES MATHÉMATIQUES ET

DE L'INFORMATIQUE



DEPARTEMENT D'INFORMATIQUE

MEMOIRE de fin d'étude

Présenté pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Systèmes d'Informations et Génie Logiciel

Par : SALEM Warda

MESSEGUEM Karima

SUJET

**Réalisation d'une application sous Android pour le suivi
des diabétiques.**

Soutenu publiquement le : / /2020 devant le jury composé de :

Nom et prénom Enseignant	Université de M'sila	Président
.....	Université de M'sila	Rapporteur
.....	Université de M'sila	Examineur
.....	Université de M'sila	Examineur

Promotion : 2019/2020.

Dédicaces

*Je dédie ce travail à : Ma Mère (**Malika Mihoubi**) qui a œuvré pour ma réussite de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour tout son assistance et sa présence dans ma vie reçois à travers ce travail modeste.*

*Mon père (**Lamri Salem**) qui est fière et trouve ici le résultat des années de sacrifices et les privations pour m'aider à avancer dans la vie puisse Dieu faire en sorte que ce travail son fruit merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.*

Tout ma famille, trois mes sœurs, deux mes frères et Surtout ma sœur du milieu, je lui donne une salutation spéciale, que Dieu ait pitié d'elle (Rabi yarhamha), ma sœur préférée et tous mes amis proches.

Warda SALEM

Remerciements

*Nous remercions Allah le tout puissant de nous avoir donné la santé, le courage et la
fois pour réaliser ce modeste travail avec volonté.*

*En premier lieu, notre encadreur **Dr. Fares Mezrag** son aide, ses conseils précieux il nous a
guidé dans notre travail et nous a aidé à trouver de solution pour avancer.*

*Je remercie aussi ma famille pour sa grande attention, son grand soutien et encouragement
tout au long de l'accomplissement de ce travail.*

*Aussi, je remercie tout particulièrement chaque personne qui de près ou de loin nous ont
apporté leur soutien dans la réalisation de ce projet.*

Enfin, j'exprime ma grande gratitude pour les membres du jury qui évaluera notre travail.

Warda SALEM

Remerciements

Je tiens à remercier toutes les personnes qui m'ont soutenu pour réaliser ce travail.

Je voudrais dans un premier temps remercier, mon encadrant MEZRAG Fares, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter ma réflexion.

Je remercie également toute l'équipe pédagogique et les enseignants du département d'Informatique.

Enfin, j'adresse mes remerciements aux membres du jury pour m'avoir honoré en acceptant d'évaluer ce travail.

Karima MESSEGUEM

Table des matières

Table des matières

Table des figures

Liste des tableaux

Introduction générale

1. Introduction	15
2. Les applications mobiles	15
2.1. Définition	15
2.2. Les types d'applications mobiles	15
2.2.1. Les applications web (connectées).....	15
➤ Présentation	15
➤ Fonctionnement	15
➤ Avantages	16
➤ Inconvénients	16
2.2.2. Les applications natives (non connectées).....	16
➤ Présentation	16
➤ Fonctionnement	16
➤ Avantages	17
➤ Inconvénients	17
2.2.3. Les applications hybrides	17
➤ Présentation	17
➤ Fonctionnement	17
➤ Avantages	18
➤ Inconvénients	15
3. Plateformes.....	18
3.1. Définition	18
3.2. Les principales plateformes	19
a. Symbian OS	19
b. iOS	19
c. Blackberry OS	19
d. Windows Phone	20
e. Ubuntu MID Editio.....	20
f. Bada OS.....	21

g. Palm OS.....	21
h. WebOS.....	22
i. Android	22
4. Présentation du système Android	22
4.1. Historique	22
4.2. Versions d'Android	23
4.3. Composants Android	26
4.4. Les outils de développement Android	26
• IDE « Android Developer Tools » et « Android Studio »	27
• La machine virtuelle Dalvik.....	28
• Le runtime Android (ART).....	28
4.5. Les éléments d'une application	28
4.6. Cycle de vie d'une application Android.	29
4.7. Pourquoi la programmation Android ?	30
5. Conclusion	31

Chapitre II : La télémédecine et le Diabète

1. Introduction	32
2. La Télémédecine	32
2.1. Définition	32
2.2. Les actes de la Télémédecine	32
2.2.1. La téléconsultation	32
2.2.2. La télé-expertise	33
2.2.3. La télésurveillance	33
2.2.4. La téléassistance	34
2.3. Usage De La Télémédecine	34
2.4. Intérêt de la télémédecine	34
3. Le Diabète	35
3.1. Définition	35
3.2. Les Types de Diabète	35
3.2.1. Le diabète de type 1	35
3.2.2. Le diabète de type 2	36
3.2.3. Diabète Gestationnel (14% des femmes enceintes)	36
3.3. Les complications du diabète	37

4. Relation entre le Diabète et la Télémédecine	38
• Principe.....	38
5. Diagnostic De Diabétique	38
6. Mesure de glycémie	39
• Valeur de glycémie.....	39
7. Conclusion	40

Chapitre III : Analyse et réalisation

1. Introduction	41
2. UML	41
2.1.Diagramme de cas d'utilisation	41
• Acteurs	41
• Specification.....	42
2.1.1. Cas d'utilisation « Authentification »	42
2.1.2. Cas d'utilisation « créer un compte »	42
2.1.3. Cas d'utilisation « Ajouter »	43
2.1.4. Cas d'utilisation « faire un test »	43
2.2.Diagramme de séquences	43
2.2.1. Diagramme de séquence « Authentification »	44
2.2.2. Diagramme de séquence « Recherche »	45
2.2.3. Diagramme de séquence « Test »	46
2.2.4. Diagramme de séquence « Ajout »	46
2.2.5. Diagramme de séquence « Suppression »	47
2.2.6. Diagramme de séquence « notification »	48
2.3.Diagramme d'activité	49
2.4.Diagramme de classe	51
3. Conclusion	51

Chapitre IV : Réalisation et Implémentation

1. Introduction	52
2. Programmation de développement	52
2.1.Le langage java	52
2.2.Le langage XML	52
2.3.Le langage HTML5	52
2.4.Atom (Editeur de texte)	53

3. Les bases de données	53
3.1. Les bases de données Firebase.....	53
3.1.1. Analytiques	53
3.1.2. Authentification	54
3.1.3. Cloud Messaging	54
3.1.4. Réaltine Data base	55
3.1.5. Crashlytics.....	55
3.1.6. Cloud Firestore	56
3.2. Architecture avec Cloud Firestore.....	55
3.3. Ajouter Firebase à l'application SugrDaib	56
3.4. Connecter Firebase avec Android	59
➤ Principe de fonctionnement	59
4. Présentation de quelques interfaces de notre application	60
5. Conclusion	68
Conclusion générale	69
Bibliographie	70
Résumé	72

LISTE DES FIGURES

Figure I.1. Les versions d'Android	23
Figure I.2. Les principaux composants du système d'exploitation Android	26
Figure I.3. Cycle de vie d'une application Android	29
Figure II.1. La téléconsultation	33
Figure II.2. La télé-expertise	33
Figure II.3. La télésurveillance	34
Figure II.4. La téléassistance	34
Figure II.5. Le diabète.	35
Figure II.6. Premiers symptômes du diabète de type 1	36
Figure II.7. Les symptômes du diabète de type 2	37
Figure II.8. Les complications du diabète	38
Figure III.1. Diagramme des cas d'utilisation	42
Figure III.2. Diagramme de séquence de l'authentification	45
Figure III. 3. Diagramme de séquence « Recherche »	45
Figure III.4. Diagramme de séquence « Test »	46
Figure III.5. Diagramme de séquence « Ajout »	47
Figure III.6. Diagramme de séquence « Suppression »	48
Figure III.7. Diagramme d'activité « authentification »	49
Figure III.8. Diagramme d'activité « ajouter un utilisateur».....	50
Figure III.9. Diagramme des classes	51
Figure IV.1. Firebase Authentication Database	54
Figure IV.2. Firebase Realtime Database.....	54
Figure IV.3. Firebase Cloud Firestore Database.....	55
Figure IV.4. L'architecture du serveur Playchat	56
Figure IV.5. Première étape de l'implémentation	56
Figure IV.6. deuxième étape de l'implémentation.....	57
Figure IV.7. Troisième étape de l'implémentation.....	57
Figure IV.8. Fichier build.gradle au niveau de l'application	57

Figure IV.9 Fichier build.gradle au niveau du projet.....	58
Figure IV.10. Dernière étape de l'implémentation.....	58
Figure IV.11. Console de Firebase.....	58
Figure IV.12. Connexion du Firebase avec Android.....	59
Figure IV.13. Connexion cloud firestore.....	60
Figure IV.14. Page d'accueil (administrateur)	60
Figure IV.15. Page d'accueil (médecin)	61
Figure IV.16. Page d'accueil (patient)	61
Figure IV.17. La fenêtre d'authentification (Administrateur)	61
Figure IV.18. Fenêtre d'authentification (patient)	62
Figure IV.19. Ajouter un patient	62
Figure IV.20. Vérification du numéro de téléphone	63
Figure IV.21. Fenêtres de diagnostic	64
Figure IV.22. Fenêtre de profil d'un patient	64
Figure IV.23. Fenêtres de mesure de Glycémie	65
Figure IV.24. Page d'évaluation de glycémie par graph	66
Figure IV.25. Page liste des médecins	66
Figure IV.26. Pages Alarme de notification	67
Figure IV.27. Envoi d'un message à un médecin	68

Liste des tableaux

Tableau I.1. Les versions d'Android	23
Tableau I.2. Tableau comparatif entre Eclipse et Android Studio	28
Tableau II.1. Les valeurs de glycémie	39
Tableau III.1. Authentification.....	42
Tableau III.2. Créer un compte	43
Tableau III.3. Ajouter	43
Tableau III.4. Faire un test	43

INTRODUCTION GÉNÉRALE

Introduction générale

L'informatique ne cesse d'envahir les différents domaines des activités humaines. En effet, cet outil permet entre autres l'automatisation des traitements, l'échange d'information, la conservation des données, l'exécution rapide des tâches. Il fournit aussi des systèmes capables de gérer et de résoudre tous les problèmes rencontrés dans les différentes entreprises.

Ainsi, jusqu'à présent, le Smartphone occupe une place de choix dans la vie de tous les jours à travers ses multiples services et plus particulièrement son gain de temps. On a tenu compte de cette innovation et de toutes ces opportunités pour créer notre application.

De plus, On a constaté que la croissance de la population dans les centres hospitaliers nécessite la mise en place d'une gestion ce qui nous amène à réfléchir sur un moyen de communication qui ouvre de nouvelles possibilités d'organisation du système de soin. De là est né le questionnement de ce travail, à savoir à quel point ce nouvel outil est utile dans le diagnostic des patients, notamment avec la pandémie du Covid-19 qui empêche les personnes souffrant des maladies chroniques, tel le diabète, de se déplacer pour faire leur diagnostic et leur soin.

L'objectif de notre travail est de concevoir et de réaliser un système informatique permettant à un utilisateur via son Smartphone de suivre son diabète au quotidien en notant et en enregistrant toutes les valeurs prises qui pourront être consultées par son médecin traitant et établir un diagnostic à distance.

Pour ce faire, nous avons divisé le travail en: partie théorique et partie développement de l'application.

La partie théorique est axée sur trois chapitres :

Le premier s'intitule « Applications mobile et Android», il a pour but de présenter les concepts liés aux applications mobiles et au système d'exploitation opté dans notre application.

Le deuxième chapitre s'intitule « La Télémédecine et le Diabète », il a pour objectif d'aborder la maladie du diabète pour laquelle sera basée notre application.

La deuxième partie sera consacrée tout d'abord au cadre méthodologique de notre système.

La réalisation fera l'objet d'un quatrième et dernier chapitre dans lequel nous définirons les outils de développement que nous avons utilisés et nous illustrerons également quelques interfaces de l'application mis en œuvre.

Enfin, nous conclurons ce travail en résumant les connaissances acquises durant la réalisation du projet ainsi que quelques perspectives quant à l'avenir de notre application.

CHAPITRE I : APPLICATIONS
MOBILES ET ANDROID

Chapitre I : Applications mobiles et Android

1. Introduction

Actuellement, les mobiles sont les appareils les plus utilisés pour communiquer ou naviguer sur Internet par rapport à un ordinateur. Cette utilisation importante de ces mobiles fait le développement des applications mobiles. Cela nous amène à définir cette notion avant qu'on se lance dans le développement d'une application mobile.

2. Les applications mobiles

2.1.Définition

Une application mobile ou application Smartphone est un logiciel d'application téléchargeable, conçu pour fonctionner sur un appareil mobile, tel qu'un Smartphone ou une tablette Pc. Elle vise à fournir des services similaires à ceux accessibles sur ordinateurs.

2.2.Les types d'applications mobiles

Selon leurs modes de fonctionnement, les applications mobiles se répartissent en trois catégories : les applications web (connectées), les applications natives (non connectée) et les applications hybrides.

2.3.Les applications web (connectées)

➤ Présentation

Une application web fonctionne en mode connecté qui impose la disponibilité permanente du réseau. Elle peut être uniquement lancée depuis un navigateur. Cette solution est également basée sur des langages web (HTML, XML, etc.). On peut alors les qualifier de site web pour appareil mobile. À chaque opération effectuée ou accès à des données, les informations sont lues sur Internet. Elles sont codées en Objective C sur iPhone, JAVA pour Android et Blackberry.

Une application mobile web est accessible par tous les Smartphones quel que soit leur marque et système d'exploitation.

➤ Fonctionnement

Le principe du fonctionnement des applications web est la lecture de données.

Une fois lancée, l'application se connecte au serveur web afin de télécharger un fichier, en général au format XML.

Les flux de ce fichier sont générés en temps réel par un script écrit en PHP, qui interprète la demande reçue et interroge la base de données en question. Dès que les données sont prêtes l'application les prend en charge, les interprète et les place dans l'interface.

Grace à cette technique de lecture de données, l'ensemble des applications (iPhone, Android, etc.) se connectent à la même source de données ce qui simplifie aussi la maintenance et les mises à jour.

Ce type d'application est recommandé pour des personnes qui utilisent des services connectés à un site web, tels que l'utilisation des réseaux sociaux, où généralement quand l'accès aux informations doit être rapide. [1]

➤ **Avantages**

- Le développement et l'administration sont très simples puisque l'application se situe sur un serveur central ;
- Nécessite de faibles ressources matérielles ;
- Possibilité de fournir des informations en temps réels ;
- Facilité de mise à jour du contenu des applications.

➤ **Inconvénients**

- La connexion réseau est obligatoire ;
- Interruption de service suite à une défaillance du réseau ;
- Le trafic réseau engendré est important ;
- Vulnérabilité du système, vue que les informations ne résident pas directement sur le client ;
- Echec de l'envoi de données volumineuses (grandes photos, vidéo) ;
- Plein de petits chargements qui peuvent être gênants au cours de l'utilisation.

2.3.1. Les applications natives (non connectées)

➤ **Présentation**

Ce sont des applications où "L'interface graphique utilise des composants Android, mais la localisation et le traitement des données est partagé entre le téléphone et le serveur." [2] Autrement dit, ce sont des applications où toutes leurs fonctions et données médias se trouvent dans le téléphone ou la tablette. Les deux éléments essentiels pour bénéficier des fonctionnalités d'une application non connectée consistent à avoir un téléphone avec l'ensemble des logiciels embarqués. Ce type d'applications est basé sur un langage propre à la plateforme ("langage natif") : codées en Objective C sur iPhone, JAVA pour Android et Blackberry.

➤ **Fonctionnement**

Le téléphone mobile lit tous les fichiers qu'il lui faut directement dans la mémoire de stockage du téléphone, ou dans une base de données embarquée.

Ce type d'application est destiné à remplir une fonction bien précise, sans pour autant avoir besoins d'évoluer dans le temps par leur contenu. [3]

➤ **Avantages**

- L'application est toujours disponible, quel que soit l'état du réseau ;
- L'interface utilisateur est plus conviviale ;
- L'application est toujours disponible ou que l'on soit ;
- L'accès aux données est très rapide, d'autant plus que les téléphones disposent ;
- D'une mémoire de type SSD, avec des débits de lecture assez élevés ;
- Lors du téléchargement de l'application, tout son contenu est immédiatement disponible.

➤ **Inconvénients**

- La multiplication des types d'appareils induit souvent des incompatibilités ;
- Les ressources consommées sont importantes.

2.3.2. Les applications hybrides

➤ **Présentation**

Une application hybride est une application issue de la combinaison des deux modes (connecté et déconnecté); c'est-à-dire elle combine des éléments HTML5 sous forme d'application web mobile et des éléments d'une application native. Elle permet d'utiliser les fonctionnalités natives des smartphones et d'être distribuée en tant qu'application sur les plateformes d'applications (App Store, Android Market, etc.).

Le principe d'une application hybride est de réduire les coûts et les délais de développement nécessaires par rapport au développement d'application native spécifique à un seul système d'exploitation.

Elles reposent à la fois sur :

-  Un serveur web/php alimenté par une base SQL ;
-  Un système de stockage Interne du téléphone ;
-  L'application en elle-même.

L'application en utilisation simple vient lire les données sur son disque de stockage, ou sa base de données interne, mais dispose d'une fonction qui permet de synchroniser son contenu.

➤ **Fonctionnement**

Une fois le téléphone est allumé, l'application vérifie tout d'abord sa connectivité puis envoie une requête de date de dernière mise à jour vers le serveur. Le serveur répond via un flux XML contenant une liste de données à récupérer, (URL(s) de fichiers médias, requêtes

SQL pour synchroniser avec la base de données embarquée, etc.). L'application télécharge toutes les informations de manière séquentielles (via HTTP ou FTP), ou exécute les requêtes SQL. Une fois toute la séquence terminée, l'interface est rechargée.

Ce type d'application se prête bien à l'usage professionnel, quand la fiabilité est importante, tout en gardant la possibilité d'échanger des informations rapidement.

➤ **Avantages**

- Un bon mélange entre la disponibilité et la flexibilité du contenu ;
- Application légère à télécharger ;
- Seules les nouvelles informations sont téléchargées (ou celles mises à jours évidemment), les autres sont stockées dans le téléphone.

➤ **Inconvénients**

- Les données peuvent ne plus être valables si l'on oublie de faire la mise à jour ;
- Le premier lancement implique le téléchargement de tout le contenu. Même en Wifi, cela peut prendre du temps dans certain cas.

3. Plateformes

3.1.Définition

Une plateforme en informatique est une base de travail à partir de laquelle on peut lire, écrire, développer et utiliser un ensemble de logiciels. Elle est composée de:

1. Matériel (exemple : microprocesseur) ;
2. Système d'exploitation mobile : Un système permettant, entre autres la gestion de la connectivité sans fil et l'interfaçage avec d'autres applications externes ;
3. Des outils logiciels :
 - ✓ de développement :
 - Bibliothèque logicielle ;
 - Editeur de texte, compilateur ;
 - Environnement de développement intégré.
 - ✓ de gestion de projet :
 - Gestionnaire de bugs.
 - ✓ de gestion de base de données (SGBD) :
 - MySQL ;
 - Oracle.
 - ✓ de serveur web :
 - Apache.

3.2.Les principales plateformes

Les appareils mobiles sont des appareils à ressources limitées ou faibles. C'est pourquoi, une adaptation du système d'exploitation à leurs limites est nécessaire. Pour se faire, plusieurs plateformes issues de différentes compagnies sont destinées au développement des systèmes d'exploitation et d'applications mobiles. Parmi elles on peut citer : Symbian OS de Nokia, iOS de Appel's, BlackBerry OS de RIM's, Windows Phone de Microsoft, ubuntu MID Edition d'Ubuntu, Bada de Samsung, Palm, webOS et Android de Google.

a. Symbian OS

Symbian OS est le système d'exploitation mobile développé par la société Symbian, pour le système de messagerie électronique. Symbian est soutenu par plusieurs technologies comme BlackBerry via BlackBerry Connect 1 et Microsoft via Exchange ActiveSync, etc. La position du Symbian dans le marché des systèmes d'exploitation pour les Smartphones est garantie du fait que Nokia, l'une des plus grandes firmes de téléphones dans le monde soit son principal actionnaire.[4]

Cette réduction dans le marché des systèmes d'exploitation a poussé la compagnie Nokia en janvier 2008 à envisager des plans afin de passer Symbian en une plateforme open source. Selon le calendrier de lancement du Symbian, le code source est lancé en juin 2010. Il est publié sous la licence " la Licence publique Eclipse 1.0. ".

Symbian est extrêmement puissant, il offre un OS de soutien au niveau de la plupart des fonctionnalités que l'on peut trouver dans Palm, Windows Mobile et BlackBerry. C'est une plateforme flexible permettant aux développeurs d'ajouter facilement leurs technologies et leurs infrastructures à la plateforme Symbian. De plus, il est soutenu par de grandes compagnies de l'industrie mobile telles que Nokia, Sony Ericsson, Motorola, etc. [4]

b. iOS

iOS est le système d'exploitation mobile développé par Apple pour l'iPhone, l'iPodtouch, et l'iPad. C'est l'un des systèmes d'exploitation mobiles les plus avancés au monde.

La plateforme iOS est dotée d'une interface utilisateur conviviale, élégante et intuitive : il suffit d'avoir l'iPhone, l'iPad et l'iPod touch en main pour aussitôt les maîtriser.

Les mises à jour iOS sont gratuites et faciles à télécharger du fait qu'elles sont notifiées automatiquement par l'appareil dès leurs apparitions. [5]

c. Blackberry OS

BlackBerry OS est un système d'exploitation multitâche propriétaire conçu pour les téléphones mobiles de la gamme BlackBerry, par la société canadienne Research In Motion (RIM). [6].

BlackBerry OS fut créé à l'origine pour le marché des hommes d'affaires. Il est doté donc de différentes applications pour le business tels que : les messages électroniques, les messages texte (SMS), les messages MMS, le BlackBerry Messenger, le web navigateur, le mémo, etc. Ce système est surtout connu pour son support natif des courriels à travers le protocole Mobile Information Device Profile (MIDP). En effet, ce système permet un service de messagerie électronique pour les entreprises en utilisant BlackBerry Enterprise Server. Il soutient aussi la technologie de la pièce jointe de différents types tels que les fichiers avec les extensions .zip, .html, .doc, .dot, .ppt, .pdf, etc. Dotés d'une molette, d'un clavier QWERTY et d'une interface utilisateur, les BlackBerry sont d'excellents appareils pour faciliter l'utilisation du système de courriel. [6]

Maintenant, BlackBerry devient de plus en plus populaire dans le monde et offre de multiples fonctionnalités pour le grand public comme le multimédia (l'appareil photo, le camera, etc.).

d. Windows Phone

Windows phone, dit aussi Windows Mobile est développé par la firme Microsoft. C'est une plateforme propriétaire et non open source, créée à l'origine pour être une version mobile de windows avec une interface utilisateur conforme à la version actuelle de windows. Windows Mobile supporte beaucoup de types d'audio, vidéo et apporte même la possibilité de télécharger, de jouer de la musique et de regarder le TV en ligne.

La messagerie électronique du Windows Mobile n'est pas aussi performante que celle du Blackberry. En effet, la messagerie électronique est moins sécurisée dans le windows mobile. Windows mobile utilise la même collection d'APIs que Windows ce qui facilite le développement d'applications pour les développeurs du Microsoft et l'exécution pour les applications de Microsoft, ce qui n'est pas avantageux pour des applications et des développeurs autres que ceux du Microsoft.

Dans ces nouvelles versions (exemple : version 6), Windows Mobile a amélioré la capacité de connexion et de compatibilité avec Microsoft Office. Il a intégré plusieurs applications de communication comme Windows Live, hotMail, Messenger, etc.

Maintenant, il y a plusieurs appareils qui marchent sous Windows Mobile dans le monde comme quelques modèles du Motorola, Samsung, T-Mobile, etc. [7]

e. Ubuntu MID Edition

Ubuntu MID3 Edition est un système en code open source, développé par la communauté Ubuntu et parrainé par les sociétés Canonical 4 et Intel. Les MIDs (Mobile Internet Device) peuvent être des terminaux mobiles multimédia dotés ou non de fonctions de téléphonie, créée sous l'idée de "mini - ordinateur portable".

L'usage principal des MIDs est la navigation Internet en mobilité. Comme Android, grâce au noyau Linux, Ubuntu MID Edition est sûr.

Il est très souple, personnalisable et simple d'utilisation. Il supporte aussi des variétés d'application de Web 2.0 comme le navigateur Web, la messagerie électronique, la caméra, le VoIP, la Messagerie instantanée, le GPS, le blog, la TV, les jeux électroniques, etc. Pour la technologie de communication, il soutient le Wi-Fi, Bluetooth, GPS et WiMAX.

Maintenant, Ubuntu MID Edition est utilisé dans quelques modèles des appareils utilisant une plate-forme Intel Atom, Nokia N800 web tablette, etc.

Au final, Ubuntu MID Edition inclue les mêmes services d'Internet qu'on trouve sur ordinateurs, mais utilisée généralement pour les MIDs et non pas pour les téléphones portables.

f. Bada OS

Bada est le système d'exploitation pour Smartphones, propriétaire de Samsung. Il est développé au début de l'année 2010 [8], après avoir été dévoilé au Mobile World Congress (le congrès mondial du mobile) de Barcelone dans la même année.

Bada peut fonctionner d'une manière variée selon l'appareil sur lequel il est utilisé. Les périphériques utilisant Bada comportent toujours trois touches : la touche centrale, la touche Répondre et la touche Raccrocher.

La première sert à afficher la liste des applications présentes dans l'appareil et/ou à quitter une application, en la laissant en arrière-plan.

La deuxième sert à répondre uniquement aux appels. Il y a aussi une touche volume, présente à gauche. Quant à La troisième, elle sert à raccrocher lors d'un appel, à afficher les widgets présents sur l'appareil ou à fermer l'application, bien que certaines applications natives, comme Musique, restent actives.

g. Palm OS

Palm OS est le système d'exploitation développé par la société Palm. Il est facile à utiliser et simple à apprendre. Il permet de minimiser les étapes pour naviguer entre les écrans et choisir les applications. Par exemple, pour lancer un programme, il suffit d'appuyer sur son icône et dès qu'on passe à un autre, l'application en cours prenne fin automatiquement.

Palm OS a vu son évolution passée des versions mono-tâches (5.0 et ultérieur) en version multitâche (6.0). Le fondement de cette nouvelle version se base sur les versions mono-tâches précédentes et offre plusieurs améliorations comme la communication et le multimédia.[7]

h. WebOS

webOS est un système d'exploitation entièrement open source basé sur un noyau Linux, développé comme une dernière version de Palm. Il est très différent des anciennes versions, il se base sur les applications utilisant HTML, JavaScript et CSS.

Une application webOS est lancée soit par un utilisateur ou par une autre application. Elle supporte le multitâches, mais peut être conçue pour fonctionner uniquement en arrière-plan. Cette application d'arrière-plan interagit avec l'utilisateur principalement en affichant des alertes et des notifications.

En 2013, Samsung annonça l'arrêt du développement de Bada pour se concentrer sur le développement d'une nouvelle plateforme baptisée Tizen.

i. Android

Android est un système d'exploitation développé par Google pour téléphones portables de nouvelle génération, PDA et autres terminaux mobiles (tablette). Il se base sur un noyau Linux 2.6. Cette plateforme met à disposition un kit de développement (SDK) basé sur le langage Java. Android a été conçu pour intégrer au mieux les applications existantes de Google, pour permettre un accès rapide aux services d'internet comme le service de courrier Gmail, YouTube, Google Talk, l'agenda Google Calendar et encore la cartographie Google Maps.

4. Présentation du système Android

4.1.Historique

Android a été conçu par une startup éponyme spécialisée dans le développement d'applications mobiles en 2003 puis a été racheté par google en 2005 et annoncé officiellement le 15 novembre 2007. [9]

Afin de promouvoir ce nouveau système d'exploitation ouvert, Google a su fédérer autour de lui un consortium d'une trentaine d'entreprises : l'Open Handset Alliance (OHA). Le but de cette alliance est de mettre en place des normes standards dans le domaine de la téléphonie mobile. Ce qui permet aux développeurs d'applications Android d'obtenir la permission d'intégrer, d'agrandir et de remplacer des composants existants.

En 2015, Android est le système d'exploitation mobile le plus utilisé dans le monde, devant iOS d'Apple, avec plus de 80 % de parts de marché dans les smartphones pour l'ensemble de ses versions et adaptations. [10]

4.2. Versions d'Android

Android a connu plusieurs versions, et chaque nouvelle version sert à corriger des bugs et ajouter de nouvelles fonctionnalités, ayant chacune d'elle droit à un nom de code sur la thématique des desserts suivant une logique alphabétique.



Figure I.1. Les versions d'Android

Dans le tableau suivant, il y a le nom de code de chaque version ainsi que sa date de sortie avec les caractéristiques majeures ajoutées à chaque une d'elles. [11]

Nom	Version	Date de sortie	Majeurs caractéristiques
Apple pie	1.0	11 Novembre 2007	- Téléchargement et mises à jour des applications via Android Market. - Support du Wi-Fi et du Bluetooth.
Bananas split	1.1	22 Octobre 2008	"Afficher" & "Cacher" le pavé numérique, inclus dans le menu d'appel - Support pour sauvegarder les fichiers attachés aux MMS
Cupcake	1.5		- Support pour les Widgets, qui permettent d'accéder rapidement à certaines informations de l'application à laquelle ils sont rattachés. - Ajout d'une option de rotation automatique

Donut	1.6	15 Septembre 2009	• Possibilité pour les développeurs d'intégrer leurs contenus dans les résultats de recherche. • Amélioration de la rapidité dans la recherche et les applications utilisant la caméra.
Eclair	2.0	26 Octobre 2009	• Performance matériel optimisée. • Support de plus de taille d'écran et résolutions.
Froyo	2.2	20 Mai 2010	• Amélioration de la vitesse d'exécution. • Partage de connexion USB.
Gingerbread	2.3	06 Décembre 2010	5. Support du NFC et de nouveaux capteurs (comme le gyroscope et le baromètre). • Amélioration de la gestion de l'alimentation et du contrôle des applications.
Honeycomb	3.0	22 Février 2011	• Meilleur support des tablettes • Bureau tridimensionnel avec widgets améliorés.
Ice Cream Sandwich	4.0	19 Octobre 2011	• Amélioration du texte d'entrée et du correcteur orthographique. • Ajout du WI-FI direct.
	4.1	09 Juillet 2012	• Amélioration de la vitesse d'exécution et de la gestion d'appareil photo. • Android Beam (Transfert de données rapide : NFC + Bluetooth).
Jelly Bean	4.2	13 Novembre 2012	• Widgets en mode verrouillé. • Multi-utilisateurs pour les tablettes
	4.3	24 Juillet 2013	• Ajout du support pour la résolution 4K. • Support de Bluetooth basse consommation (BLE) et du profil Bluetooth (AVRCP) 1.3
KitKat	4.4	31 Octobre 2013	• Nouvelle interface translucide. • Amélioration du système de notification et des performances.
Lollipop	5.0	03 Novembre 2014	• Nouvelle interface de design «Materialdesign»

Marshmallow			Amélioration de la rapidité et de la gestion de la batterie.
	5.1	09 Mars 2015	- Protection par blocage en cas de perte ou vol - Amélioration de la stabilité et des performances
	6.0	Fin mai 2015	- Réorganisation du gestionnaire des permissions. - Barre d'action rapide pour tablette. - Intégration Option informations.
Nougat	7.0	10 mars 2016	- Double partition système (A/B) pour pouvoir faire les mises à jour en arrière-plan. - Mode nuit. - Ajout d'un bouton Tout effacer dans le multi-tâches.
Oreo	8.x.x	21 août 2017	- l'apparition d'un mode Picture-in-picture (PiP) - une nouvelle stratégie d'optimisation de la batterie.
Pie	9.x	07 mars 2018	une gestion intelligente de la batterie et de la luminosité.
Android 10		03 septembre 2019.	- Mises à jour du système via le Google Play Store. - Nouveaux contrôles pour les autorisations de localisation des utilisateurs. - Bulles de discussion (comme Facebook Messenger). - Nouvelles autorisations pour accéder aux photos, aux fichiers vidéo et audio.

Tableau I.1. Les versions d'Android

4.3. Composants d'Android

Plusieurs couches comme un système Unix :

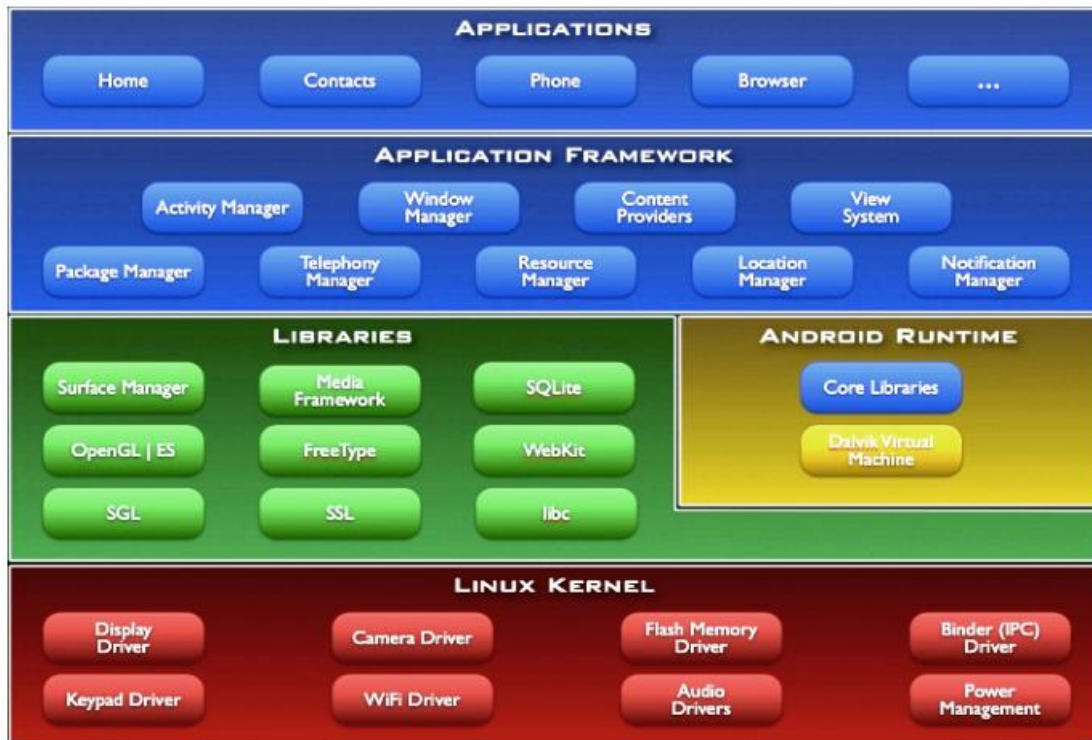


Figure I.2. Les principaux composants du système d'exploitation Android.

Ainsi, les principaux composants de ce système sont :

4.3.1. Le noyau Linux

Android repose sur un noyau Linux (version 2.6) qui gère les services du système, comme la sécurité, la gestion de la mémoire et des processus, la pile réseau et les pilotes. Il agit également comme une couche d'abstraction entre le matériel et la pile logicielle.

4.3.2. Les bibliothèques

En interne, Android inclut un ensemble de bibliothèques C et C++ utilisées par de nombreux composants de la plateforme Android.

4.3.3. Moteur d'exécution Android

Chaque application Android s'exécute dans son propre processus, avec sa propre instance de machine virtuelle Dalvik.

4.3.4. Le framework

Android offre aux développeurs la possibilité de créer des applications extrêmement riches.

4.4. Les outils de développement Android

Le développement des applications sous Android nécessite les outils suivants:

•Le SDK Android

Le Kit de développement logiciel Android (Android SDK) contient les outils nécessaires pour créer, compiler et déployer les applications Android. La plupart de ces outils sont en ligne de commande. Les outils sont:

- les bibliothèques Java pour créer des logiciels
- les outils de mise en boîte des logiciels
- AVD : Android Virtual Device, un émulateur de tablettes pour tester les applications

Il y a aussi SDK Manager ou le gestionnaire du SDK, une application qui permet de choisir les composants à installer et mettre à jour.

•Le débogueur « ADB »

Le SDK Android contient un débogueur appelé « Android Debug Bridge » ou aussi « ADB », qui permet de connecter un appareil Android virtuel ou réel, dans le but de gérer le périphérique ou de déboguer une application. Il permet aussi d'interagir avec l'émulateur pour effectuer des opérations de configuration, d'installation.[12]

•Les IDE « Android Developer Tools » et « Android Studio »

Google propose deux environnements de développement intégrés (IDE) pour développer de nouvelles applications.

a) Eclipse

Les outils de développement Android sont basés sur l'IDE Eclipse.

ADT « Android Developer Tools » est un ensemble de composants (plug-ins), qui étendent l'IDE Eclipse avec des capacités de développement Android.

b) Android Studio

Google propose également cet IDE appelé pour la création d'applications Android quel que soit le terminal sous-jacent (smartphone, tablette, montre, TV...) et qui est basé sur l'IDE IntelliJ.

Le tableau suivant compare les deux IDE

	Eclipse	Android Studio
Facilité d'installation	Moyen	Simple
Langue	Nombreuse	Anglais
Performance	Peut être lourde	Rapide
Génération de variante et de multiple APK	Non	Oui
Editeur d'interface graphique	Oui	Oui
Signature d'APK et gestion de Keystore	Oui	Oui

Support NDK	Oui	A venir
-------------	-----	---------

Tableau I.2. Tableau comparatif entre Eclipse et Android Studio.

•La machine virtuelle Dalvik

Dalvik est une machine virtuelle incorporée dans le système d'exploitation Android. Cette machine permet l'exécution simultanée de plusieurs applications sur un appareil de faible capacité (peu d'espace mémoire et peu de puissance de calcul).

•Le runtime Android (ART)

Les dernières versions d'Android introduisent une nouvelle machine, le runtime Android. Le moteur qui permet l'exécution des applications pour Android.

4.5. Les éléments d'une application sous Android: [12]

Une application Android peut être composée des éléments suivants:

- **des activités** (android.app.Activity): il s'agit d'une partie de l'application présentant une vue à l'utilisateur.
- **des services** (android.app.Service): il s'agit d'une activité tâche de fond sans vue associée. Il permet l'exécution d'un algorithme sur un temps indéfini. Il ne s'arrêtera que lorsque la tâche est finie ou que son exécution est arrêtée.
- **des fournisseurs de contenus** (android.content.ContentProvider): permettent le partage et l'accès aux informations au sein ou entre applications, ainsi que l'accès aux contacts stockés dans le téléphone à l'agenda, aux photos ainsi que d'autres données depuis votre application grâce aux content providers.
- **des widgets** (android.appwidget.*): une vue accrochée au Bureau d'Android.
- **des Intents** (android.content.Intent): permettent d'envoyer un message pour un composant externe sans le nommer explicitement.
- **des récepteurs d'Intents** (android.content.BroadcastReceiver): Un Broadcast Receiver comme son nom l'indique permet d'écouter ce qui se passe sur le système ou sur l'application et déclencher une action prédéfinie. C'est souvent par ce mécanisme que les services sont lancés.

- **des notifications** (android.app.Notifications): permettent de notifier l'utilisateur de la survenue d'événements.

4.6. Cycle de vie d'une application Android

Le cycle de vie d'une application est considéré les différents états d'une activité.

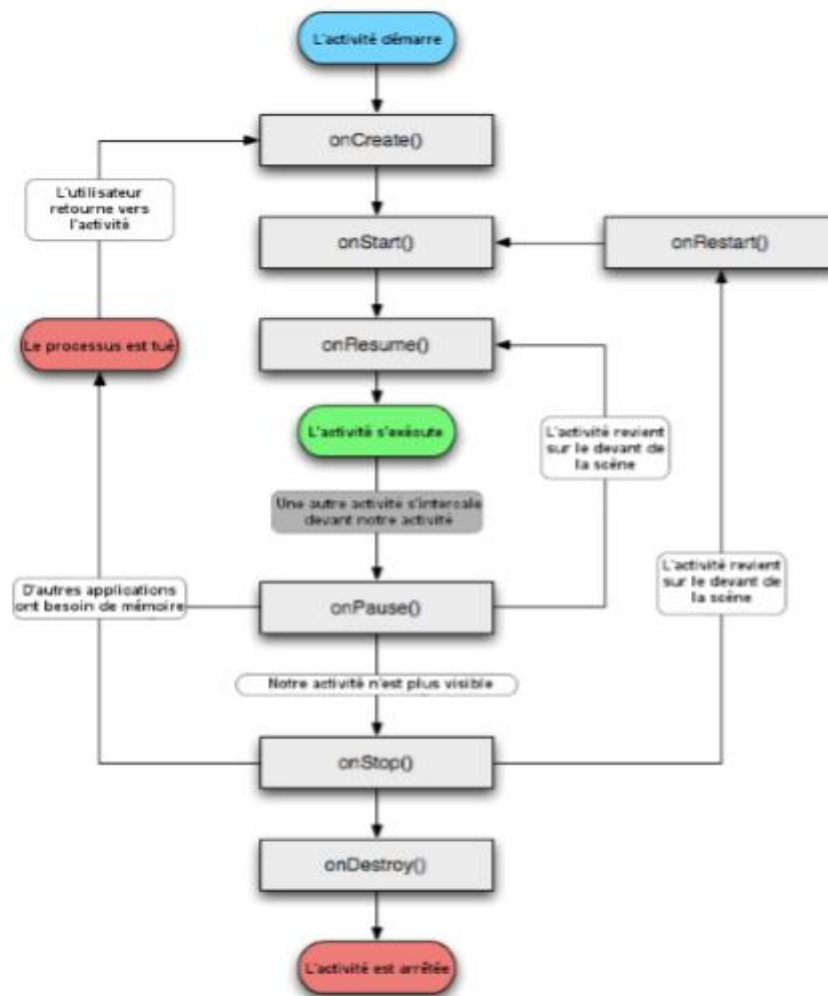


Figure I.3. Cycle de vie d'une application Android. [13]

4.6.1. onCreate ()

Elle sert à initialiser l'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 l'activité.

4.6.2. onStart ()

Signifie le début d'exécution de l'activité.

4.6.3. onResume ()

Si on avoir un état de pause, onResume () est appelée pour commencer l'activité à partir de l'état que cette activité mise en pause.

4.6.4. onPause ()

Cette méthode est appelée au passage d'une autre activité en premier plan. L'intérêt d'un tel appel est de sauvegarder l'état de l'activité et les différents traitements effectués par l'utilisateur.

4.6.5. onStop ()

Cette méthode est appelée quand l'activité n'est plus visible quel que soit la raison. Dans cette méthode vous devez arrêter tous les traitements et services exécutés par votre application.

4.6.6. onDestroy ()

Cette méthode est appelée quand votre application est totalement fermée (Processus terminé). Les données non sauvegardées sont perdues.

4.7. Pourquoi la programmation Android ?

Android est le roi incontestable des applications mobiles. Aujourd'hui, il y a plus de 700000 applications sur la boutique Google Play [14]. A ce nombre, il faut encore ajouter celles qui n'y ont pas été soumises. Ce nombre est dû des caractéristiques que Android offre:

- Le langage de programmation Java a été choisi par Google pour créer des applications Android.
- De plus en plus d'appareils mobiles fonctionnent sous Android: smartphones, tablettes, *wearablesdevices* comme montres intelligentes ou encore *Smart TV*.
- Android offre le kit de développement (SDK) qui inclut un débogueur et des bibliothèques logicielles.

- Avec Android, les changements sont fréquents. Chaque année, Google lance une nouvelle version de son système opérationnel.

Cependant, la principale différence d'Android face aux autres systèmes est qu'il soit open source. Cette particularité le rend donc gratuit, et personnalisable par les constructeurs et les opérateurs de téléphonie mobile. Donc, Android se montre à la tête des systèmes d'exploitation vue la flexibilité de sa plateforme et offre la possibilité aux développeurs de créer des applications extrêmement riches et innovantes.

5. Conclusion

Dans ce chapitre, nous avons introduit les concepts de base relatifs aux applications mobiles et présenté les principaux systèmes mobiles en mettant l'accent sur Android, ce qui va nous servir de base pour les chapitres à venir.

**CHAPITRE II : LA
TÉLÉMÉDECINE ET LE
DIABÈTE**

Chapitre II : La Télémédecine et le Diabète

1. Introduction

Dans ce chapitre, nous allons se focaliser sur l'exploitation de la télémédecine pour le suivi du diabète. Nous allons définir par la suite le diabète et ses différents types et symptômes, puis nous étudierons la relation entre la télémédecine et le diabète, enfin nous terminerons et le diagnostic avec les différentes manières de mesurer la glycémie.

2. La Télémédecine

2.1. Définition

La télémédecine est l'une des composantes de la "télésanté" s'appuyant sur les nouvelles technologies, cette pratique médicale met en rapport soit le patient et un ou plusieurs professionnels de santé, parmi lesquels un professionnel médical ; soit plusieurs professionnels de santé, parmi lesquels au moins un professionnel médical. Cette définition globale du code de la santé publique énumère également de possibles indications. Elle permet d'établir un diagnostic ; d'assurer, pour un patient à risque, un suivi dans le cadre de la prévention ou un suivi post thérapeutique ; de requérir un avis spécialisé ; de préparer une décision thérapeutique ; de prescrire des produits, de prescrire ou de réaliser des prestations ou des actes ou d'effectuer une surveillance de l'état des patients.

La télémédecine concerne la prestation d'un acte médical, à la différence de la télésanté, de l'e-santé ou de la santé, qui recouvre également des services plus larges qui vont de l'information à l'éducation, en passant par la recherche ou la formation continue.

2.2. Les actes de la Télémédecine

Les applications qui couvrent le champ de la télémédecine sont :

- ✓ La téléconsultation.
- ✓ La télé-expertise.
- ✓ La télésurveillance.
- ✓ La téléassistance.

2.2.1. La téléconsultation

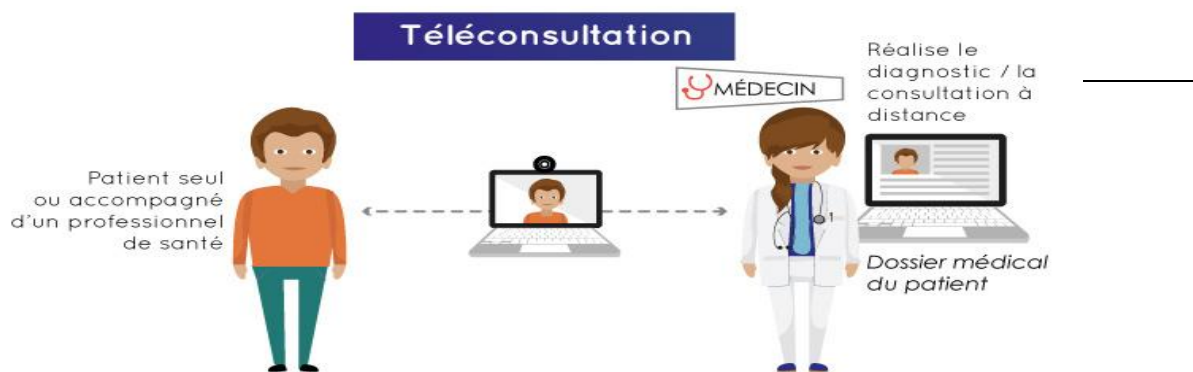


Figure II.1. La téléconsultation.

Un médecin donne une consultation à distance à un patient, lequel peut être assisté d'un professionnel de santé. Le patient et/ou le professionnel à ses côtés fournit les informations, le médecin à distance pose le diagnostic.

2.2.2. La télé-expertise

Pour permettre à un professionnel médical de solliciter à distance l'avis d'un ou de plusieurs professionnels médicaux en raison de leurs compétences particulières, sur la base des informations médicales liées à la prise en charge d'un patient. Contrairement à la téléconsultation, le patient n'est pas forcément acteur dans le processus de télé-expertise. Le dialogue ne peut s'établir qu'entre médecins qui discutent d'un cas nécessitant un diagnostic particulier.



Figure II.2. La télé-expertise.

2.2.3. La télésurveillance

La télésurveillance permet médical d'interpréter à distance des données recueillies sur le lieu de vie du patient d'un indicateur clinique radiologique se situant en établissement de santé, en structure médico-sociale ou à leur domicile.



Figure2.3. La télésurveillance.

2.2.4. La téléassistance

La téléassistance médicale est donc une solution pour répondre à des enjeux simples, certaines régions sont dépourvues de moyens médicaux, cette technique est un moyen pour tout un chacun de bénéficier des meilleurs conseils et interventions, quel que soit le lieu dans lequel la personne malade se situe. Tous les individus sont donc mis sur un pied d'égalité vis à vis de la santé et de la médecine.



Figure II.4. La téléassistance.

2.3. Intérêt de la télémédecine

La télémédecine est d'abord apparue à l'hôpital, sous forme de "télé-expertise" dans des spécialités de pointe [15]. Sa principale application était la concertation entre médecins de différentes disciplines en liaison avec le médecin traitant.

Aujourd'hui, cette démarche novatrice se développe et ouvre de nouvelles perspectives dans l'organisation des soins.

Elle présente en effet plusieurs intérêts :

- Elle permet de développer les soins à domicile, pour améliorer le suivi des patients et prévenir les complications.
- Elle permet de limiter les déplacements (notamment pour les patients âgés ou handicapés).

- Elle facilite l'accès aux soins dans les zones d'accès difficile.
- Elle raccourcit les délais d'attente.
- Elle facilite la concertation entre médecins généralistes et spécialistes.

3. Le Diabète

3.1. Définition :

Le diabète est une maladie chronique qui survient lorsque le pancréas ne produit pas assez d'insuline ou lorsque l'organisme n'est pas capable d'utiliser efficacement l'insuline qu'il produit. Cela se traduit par un taux de sucre dans le sang (glycémie) élevé : on parle d'hyperglycémie.

L'OMS prévoit qu'en 2030, le diabète soit la septième cause de décès dans le monde [16].

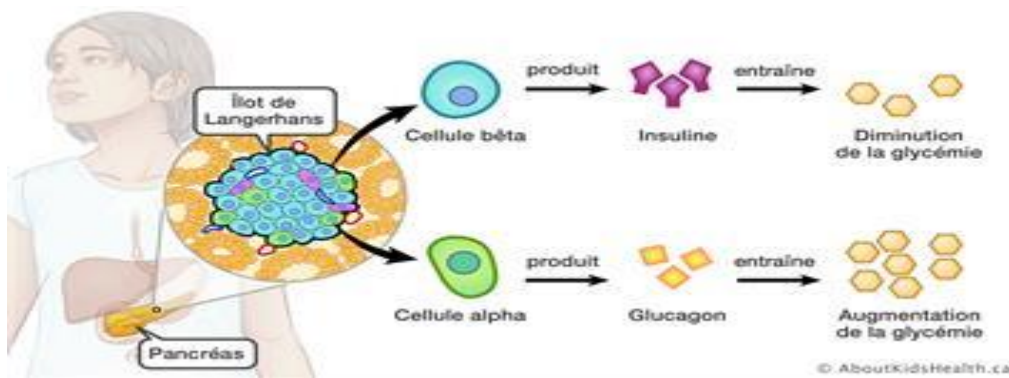


Figure II.5. Le diabète.

3.2. Les Types de Diabète

Il existe deux principaux types de diabète : le type 1 et le type 2. Parfois, le diabète se développe aussi pendant la grossesse :

3.2.1. Le diabète de type 1

Le diabète de type 1 est un diabète touchant plutôt les personnes jeunes. La cause exacte reste inconnue mais une pathologie auto-immune détruit les cellules bêta du pancréas qui sécrètent normalement l'insuline, aussi que les environnements et certains virus, certains pensent une maladie génétique conséquence l'attaque les certains cellules du pancréas. Le risque grimpe à 70 %. Dans le diabète de type 1, le risque se situe entre 4 et 8%, plus précisément 8 % si le père est diabétique, 4 % si c'est la mère (mais 30 % si les deux parents

le sont). L'unique traitement actuellement est l'apport d'insuline : soit sous forme d'injections, soit avec une pompe à insuline appareil portable [17].

Le pancréas ne produit plus du tout ou pas assez d'insuline ce qui provoque les symptômes classiques d'hyperglycémie :

- soif, polyurie et polydipsie
- perte de poids involontaire
- fatigue
- vision floue
- douleurs abdominales



Figure II.6. Premiers symptômes du diabète de type 1.

3.2.2. Le diabète de type 2

Le diabète de type 2 apparaît surtout chez des personnes en surpoids après 40 ans. Il est dû à une résistance du corps à l'insuline, cette hormone ne parvenant plus à faire rentrer le sucre dans les cellules. Aussi peut être génétique, l'obésité et le manque d'activité. Lorsque l'un des deux parents est diabétique de type 2, le risque de transmission à la descendance est de l'ordre de 40 % et si les deux parents sont atteints. Le diabète de type 2 étant une maladie évolutive, après l'augmentation progressive des antidiabétiques, des injections d'insuline seront proposées en complément au patient lorsque la carence en insuline sera trop importante.

Parfois certains signes sont présents tels que :

- des infections fréquentes et une cicatrisation lente
- un syndrome des ovaires polykystiques
- un AcanthosisNigricans (taches cutanées épaisses, d'aspect bronzé, grises, brunes ou noires).

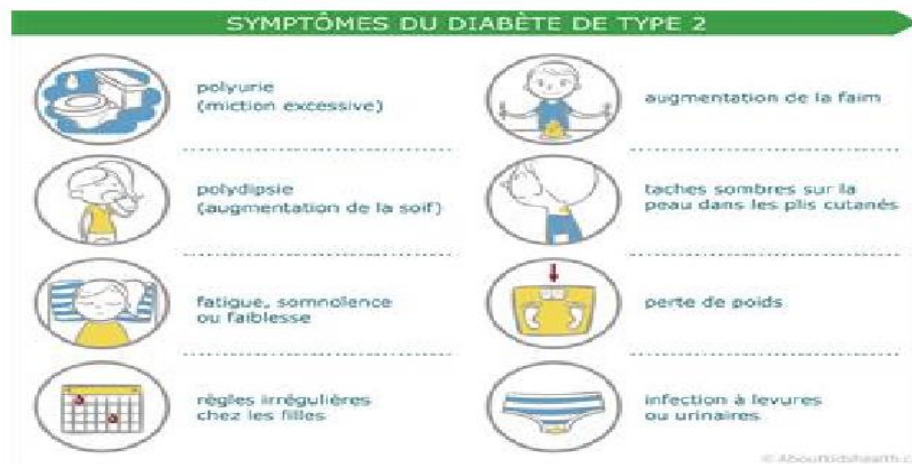


Figure II.7. Les symptômes du diabète de type 2.

3.2.3. Diabète Gestationnel (14% des femmes enceintes)

Le diabète gestationnel est un diabète survenant uniquement pendant la grossesse. Les femmes qui en souffrent sont par la suite à plus haut risque de diabète de type 2. À cause d'une sécrétion insuffisante d'insuline, l'importance du dépistage chez la femme enceinte. Certains facteurs de risques sont associés à son apparition, tels que [17]:

- ethnie non-caucasienne
- obésité
- âge > 30 ans
- femme ayant déjà accouché d'un nouveau-né de plus de 4kg.

3.3. Les complications du diabète

Le diabète peut entraîner des complications touchant de nombreuses parties du corps comme le cerveau, les yeux, le cœur, les reins et les nerfs. Les complications liées au diabète peuvent être à long terme (chroniques) ou à court terme (aiguës). Les complications à long

terme se produisent lorsque la glycémie n'est pas bien contrôlée et reste élevée pendant une période prolongée. Les complications à long terme liées au diabète incluent : [18]

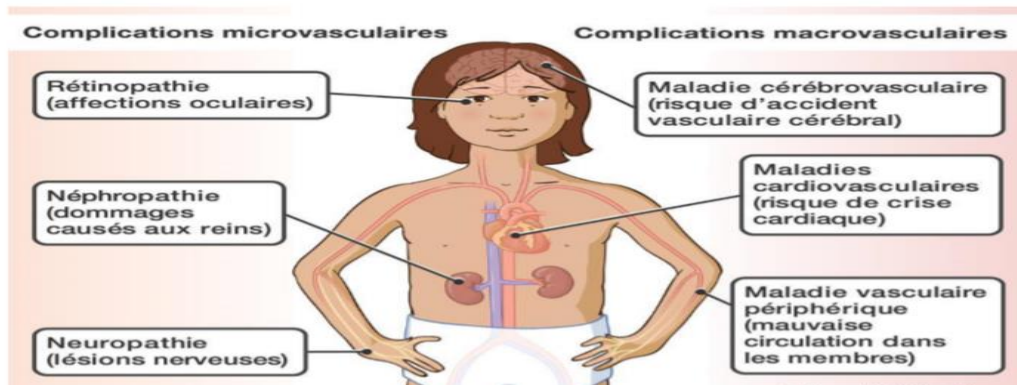


Figure II.8. Les complications du diabète

4. Relation entre le Diabète et la Télémédecine

La télémédecine repose sur l'exploitation des technologies de l'information et de la communication (TIC) comme déjà mentionné auparavant. L'objectif est de suivre et de traiter les diabétiques à distance, à travers de multiples communications électroniques échangées avec leur médecin.

- **Principe**

Son fonctionnement est simple: les échanges entre le patient et son médecin ont lieu à travers divers supports technologiques, comme Internet, à l'aide d'e-mails ou de messages sur des forums dédiés, par téléphone portable via des SMS, ou par le biais de vidéo consultations.

C'est l'occasion pour le professionnel de santé de s'entretenir avec son malade, pour lui délivrer différents conseils pratiques, d'ordre nutritionnel ou relatif à la pratique d'une activité physique, par exemple. Mais tout cela ne peut se faire, dans le cas du diabète, que si le patient transmet correctement le recueil de ses valeurs glycémiques. Il doit donc être impliqué, pour contrôler régulièrement son taux de sucre sanguin et fournir ces chiffres à son médecin. Un suivi quotidien qui n'est pas réalisable sans la télémédecine, c'est là sans doute l'un de ses points forts.

5. Diagnostic de Diabétique

Plusieurs méthodes peuvent être utilisées pour diagnostiquer le diabète. Voici quelques tests sanguins courants [19]. Le médecin peut n'avoir besoin que d'un seul test pour diagnostiquer le diabète.

- **Test de glucose dans le plasma à jeun** : Le diabète sucré est défini par une glycémie à jeun (au moins 8 h de jeûne) $\geq 1,26$ g/l (7 mmol/l) vérifiée à 2 reprises (grade B). Ce critère de diagnostic n'est pas un seuil d'intervention pharmacologique ;
- **Test d'hémoglobine A1C (HbA1C)** : Mesuré à l'aide d'un test normalisé et validé, en l'absence de facteurs compromettant la fiabilité du taux d'HbA1c et non en cas de diabète de type 1 soupçonné.
- **Test Glycémie post prandiale** : Pour ce qui est la façon dont le corps réagit à un pic de glucose, dite postprandiale, la mesure a lieu une heure et demie ou deux heures après le début du repas.

6. Mesure de glycémie

Il y a 3 cas pour déterminer valeur de glycémie :

- **Hypoglycémie** : en état à jeun (g/L)[<0.7].
- **Glycémie normale** : en état à jeun (g/L)[0.7-1], en état à post –prandiale[<=1.4].
- **hyperglycémie**: en état à jeune (g/L)[1-1.25].
- **Diabète** : en état à jeun [≥ 1.26], en état à post –prandiale [>1.8].

✓ Valeur de glycémie

Catégorie	Valeur à jeun (g/l)		Post-prandiale (g/l)
Hypoglycémie	Valeur minimale	Valeur maximale	Valeur 2 heures après la consommation du glucose
Glycémie normal	<0.7	-	-
Hyperglycémie modérée	1	1.25	-
Diabète	>1.26	-	>1.8

Table 2.1 : Les valeurs de glycémie.

7. Conclusion

Dans ce chapitre, nous avons mis l'accent sur la télémédecine, puis nous avons abordé la maladie du diabète pour laquelle sera appliquée notre solution de télésurveillance.

CHAPITRE III : ANALYSE ET CONCEPTION

Chapitre III: Analyse et Conception

1. Introduction

La réalisation d'un système nécessite la modélisation qui permet de prévoir et d'étudier les informations relatives à ce système. Pour se faire, on a opté pour le langage UML qui permet de représenter des concepts graphiques et de modéliser les applications. Cette modélisation UML montre les différents acteurs du système ainsi que les rôles qu'ils peuvent tenir. Nous avons utilisé StarUML pour créer les différents diagrammes UML.

2. UML

UML (Unified Modeling Language) permet de représenter et de communiquer les différents aspects d'un système d'information, On va modéliser les acteurs et les différents besoins de notre système par un diagramme de cas d'utilisation, suivi de leurs diagrammes de séquences, puis un diagramme d'activité et enfin un diagramme de classes.

2.1. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation donne une vue du système dans son environnement extérieur ; ainsi qu'il définit la relation entre l'utilisateur et les éléments que le système met en œuvre.

Acteurs

Il y'a trois acteurs qui peuvent utiliser notre application :

- L'administrateur est un utilisateur qui a tous les droits de gestions et de contrôles.
- Le médecin et le patient qui gèrent l'application dans la limite de leurs comptes.

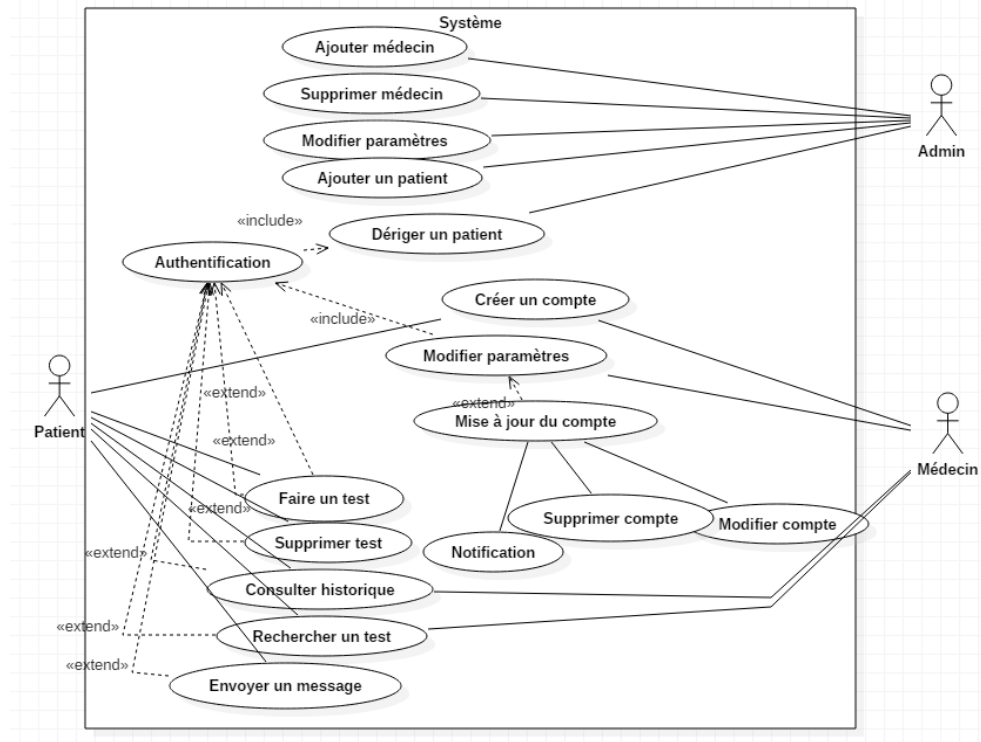


Figure III.1. Diagramme des cas d'utilisation.

✓ Spécification des exigences :

On va décrire certains cas d'utilisation identifiés dans le diagramme précédent.

2.1.1. Cas d'utilisation « Authentification »

Cas d'utilisation	Authentification
Acteur	Médecin, patient
Description	Authentifier l'utilisateur à partir de son email et mot de passe
Pré-condition	L'appareil mobile doit être connecté à Internet
Scénario	Saisir l'email et mot de passe, puis valider
Post-condition	L'authentification avec succès
Scénario d'exception	Authentification échouée, message d'erreur doit s'afficher à l'utilisateur

Tableau III.1. Authentification

2.1.2. Cas d'utilisation « Créer un compte »

Cas d'utilisation	Créer un compte
Acteur	Médecin, patient
Description	L'utilisateur crée un compte dans l'application
Pré-condition	L'appareil mobile doit être connecté à l'internet

Scénario	remplir les différents champs, puis valider
Post-condition	L'inscription avec succès
Scénario d'exception	Inscription échouée, message d'erreur doit s'afficher à l'utilisateur

Tableau III.2. Créer un compte.**2.1.3. Cas d'utilisation « Ajouter »**

Cas d'utilisation	Ajouter
Acteur	Administrateur
Description	L'administrateur ajoute un médecin ou un patient
Scénario	Valider les champs remplis par un médecin ou un patient
Post-condition	L'ajout avec succès
Scénario d'exception	L'ajout échoué, message d'erreur doit s'afficher

Tableau III.3. Ajouter**2.1.4. Cas d'utilisation « faire un test »**

Cas d'utilisation	Faire un test
Acteur	Patient
Description	Le patient établit un test en enregistrant des valeurs
Pré-condition	Le patient doit être authentifié.
Scénario	Saisir les valeurs puis valider
Post-condition	test avec succès
Scénario d'exception	test échoué, message d'erreur doit s'afficher à l'utilisateur

Tableau III.4. Faire un test**2.2. Diagramme de séquences**

La description des cas d'utilisation se fait par l'identification des scénarios possibles entre les différents acteurs du système. Dans notre cas il s'agit d'un administrateur, médecin et patient.

Chaque scénario peut être représenté par un diagramme de séquences.

Dans notre application, on distingue les cas suivants :

2.2.1. Diagramme de séquence « Authentification »

L'utilisation de l'application nécessite l'authentification auprès du système, cette opération consiste à accéder au compte et à activer l'application. De ce fait, l'utilisateur doit saisir son identifiant et son mot de passe associé.

En revanche, le système fait appel à une vérification de remplissage et d'authentification en comparant les champs remplis avec ceux de la base de données.

Si les paramètres sont corrects, cela signifie l'existence du compte, il l'on résulte un accès direct vers la page d'accueil.

Si les paramètres ne figurent pas dans la base de données, la page d'authentification sera réaffichée pour une réintroduction des paramètres.

Après trois tentatives d'accès au compte, un message sera affiché indiquant au visiteur que son mot de passe est oublié, en lui donnant la possibilité de le récupérer.

Sinon un message d'erreur de manque de remplissage sera affiché.

Sinon un message d'erreur du manque de remplissage sera affiché. L'administrateur à son tour doit s'authentifier (en raison de sécurité) afin d'avoir les privilèges qui lui ont été attribués : manipulation de la base de données (ajout, suppression, etc.). Le principe d'authentification étant le même que celui du patient.

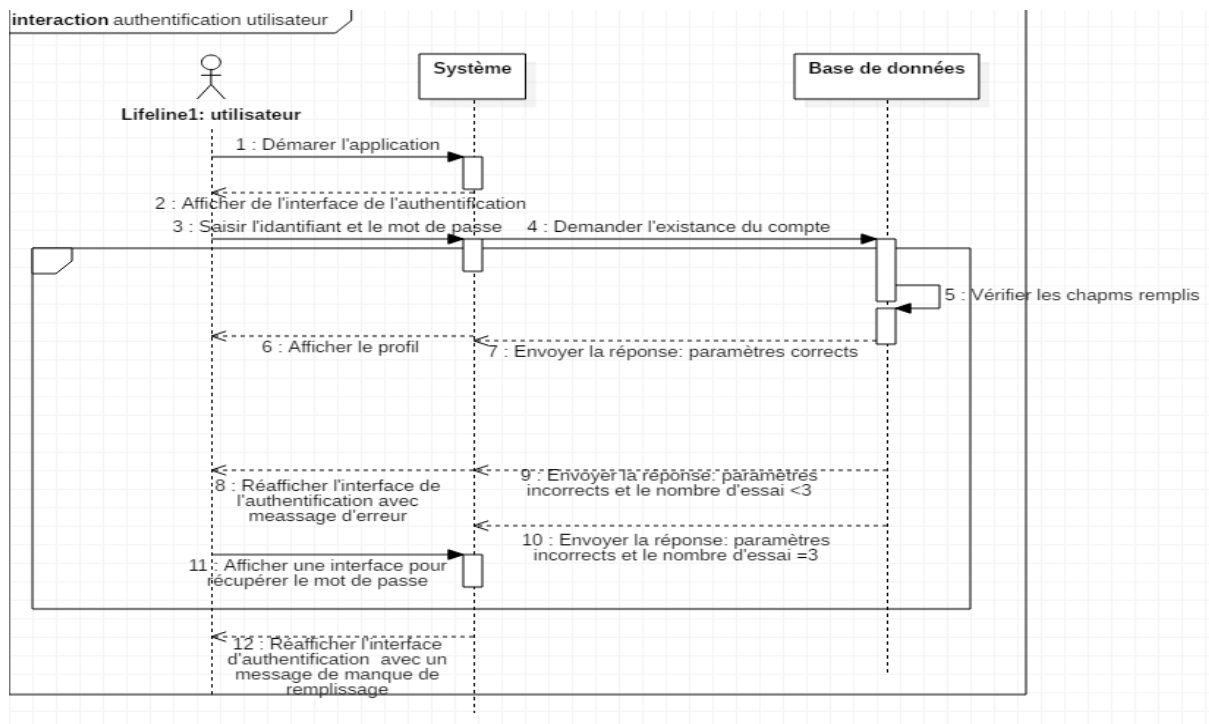


Figure III.2. Diagramme de séquence « Authentification »

2.2.2. Diagramme de séquence « Recherche »

Ce scénario consiste à faire une recherche dans la base de données selon des critères de recherche.

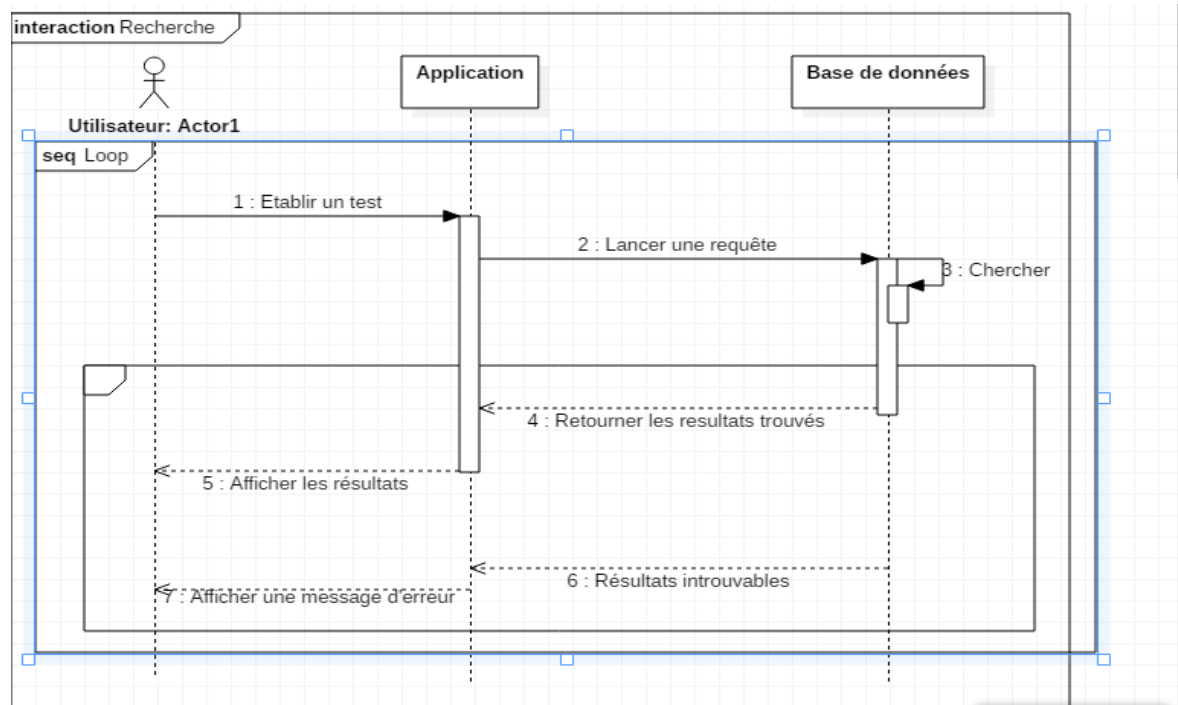


Figure III. 3. Diagramme de séquence « Recherche ».

2.2.3. Diagramme de séquence « Test »

Le test consiste à saisir la valeur et choisir l'état de (à jeun ou après un repas).

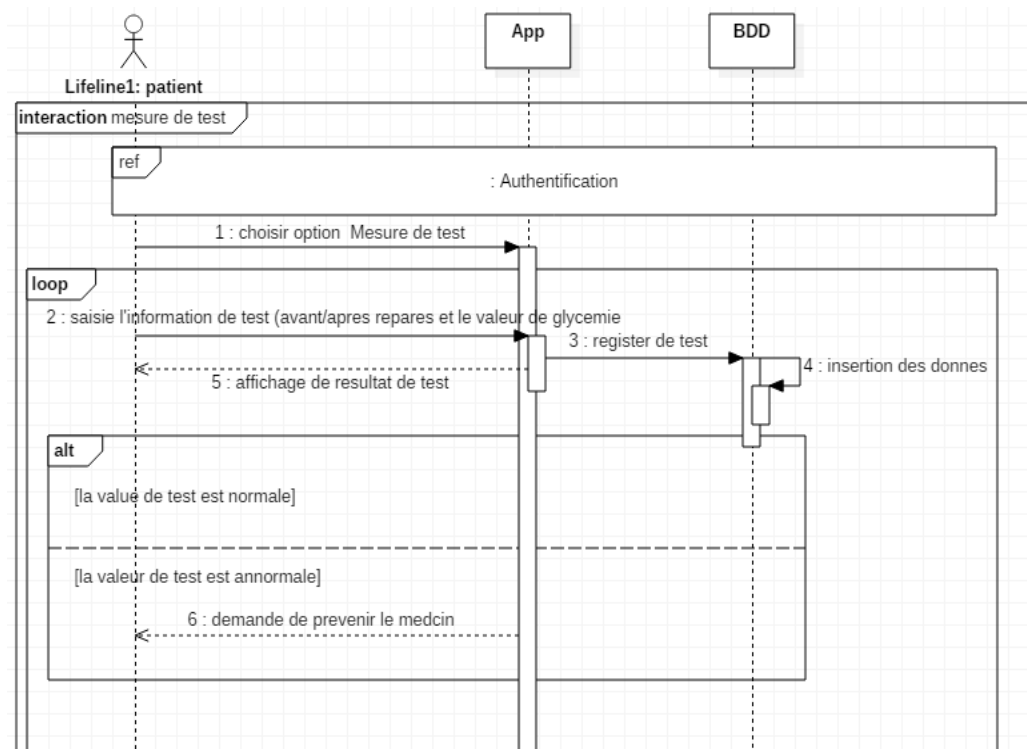


Figure III.4. Diagramme de séquence « Test »

2.2.4. Diagramme de séquence « Ajout »

Pour ajouter un médecin ou un patient dans la base de données, la procédure est simple. L'administrateur n'a qu'à remplir tous les attributs puis cliquer sur ajouter. Par exemple si l'on veut ajouter un patient à la base de données, l'administrateur accède à l'interface de connexion puis s'authentifie.

Quand l'interface d'administration lui sera affichée, il choisit l'onglet "patient" puis il clique sur le bouton ajouter. À ce moment-là une interface d'ajout d'un patient s'affiche contenant le formulaire que l'administrateur doit remplir avant de presser le bouton valider. Le système vérifie que tous les champs sont remplis puis envoie une requête d'ajout à la base de données et un message de confirmation de l'ajout sera affiché. Dans le cas où il y a un manque de remplissage un message d'erreur est prévu.

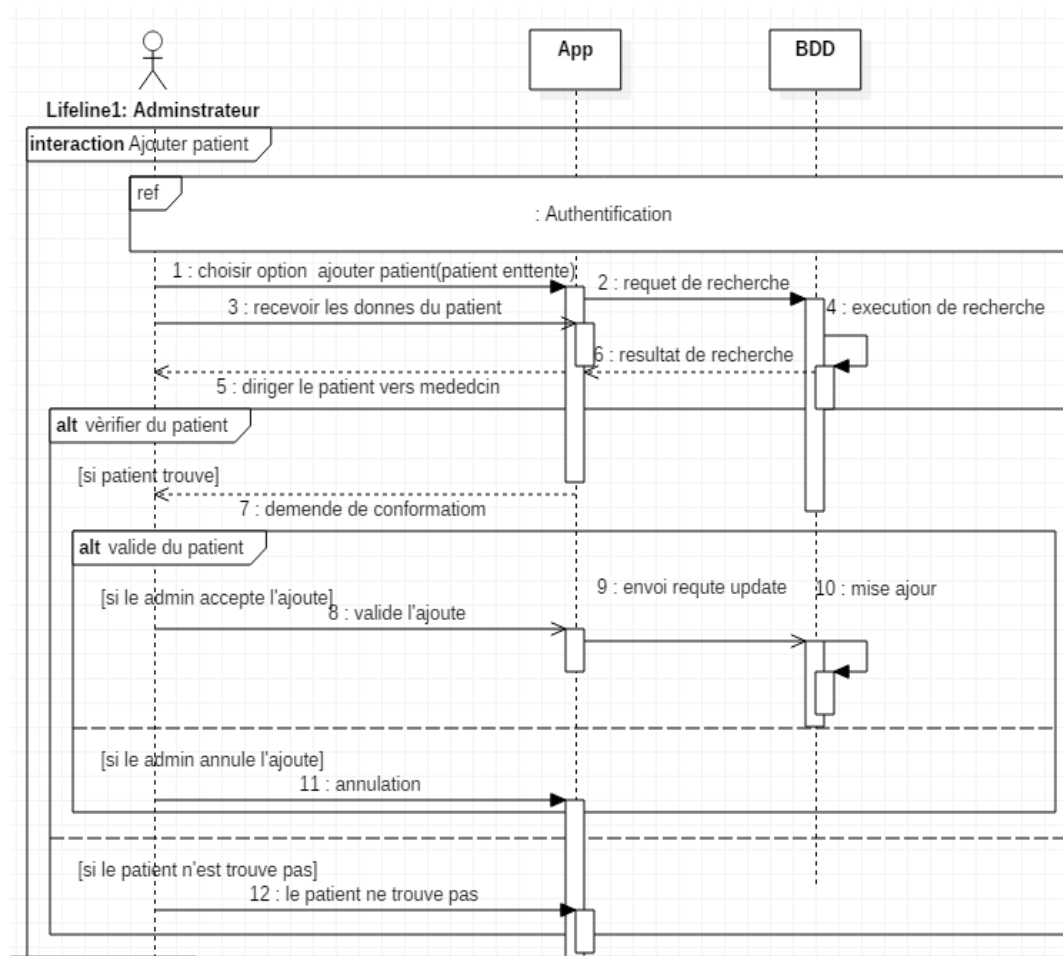


Figure III.5. Diagramme de séquence « Ajout »

2.2.5. Diagramme de séquence « Suppression »

Ce scénario consiste à supprimer un test dans la base de données.

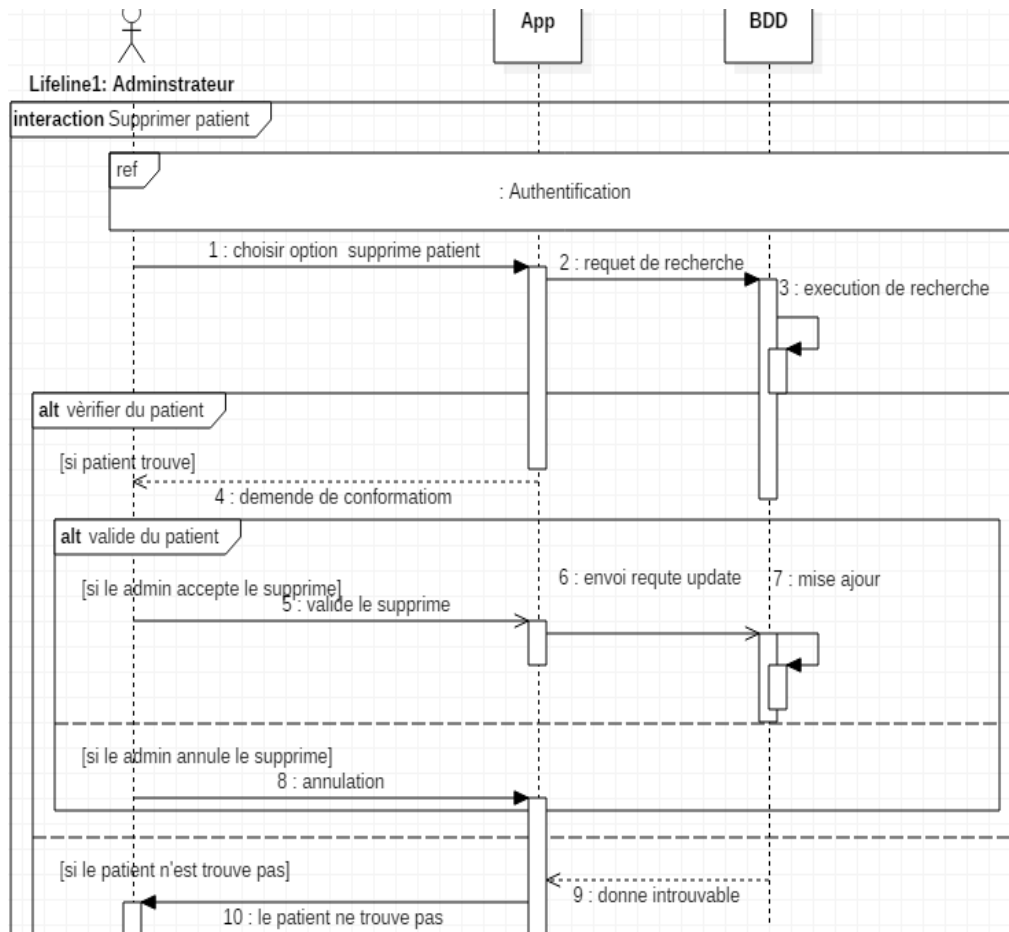


Figure III.6. Diagramme de séquence « Suppression »

2.2.6. Diagramme de séquence « notification »

Ce scénario consiste à rappeler l'utilisateur de ses tests quotidiens à des heures qu'il aura choisis.

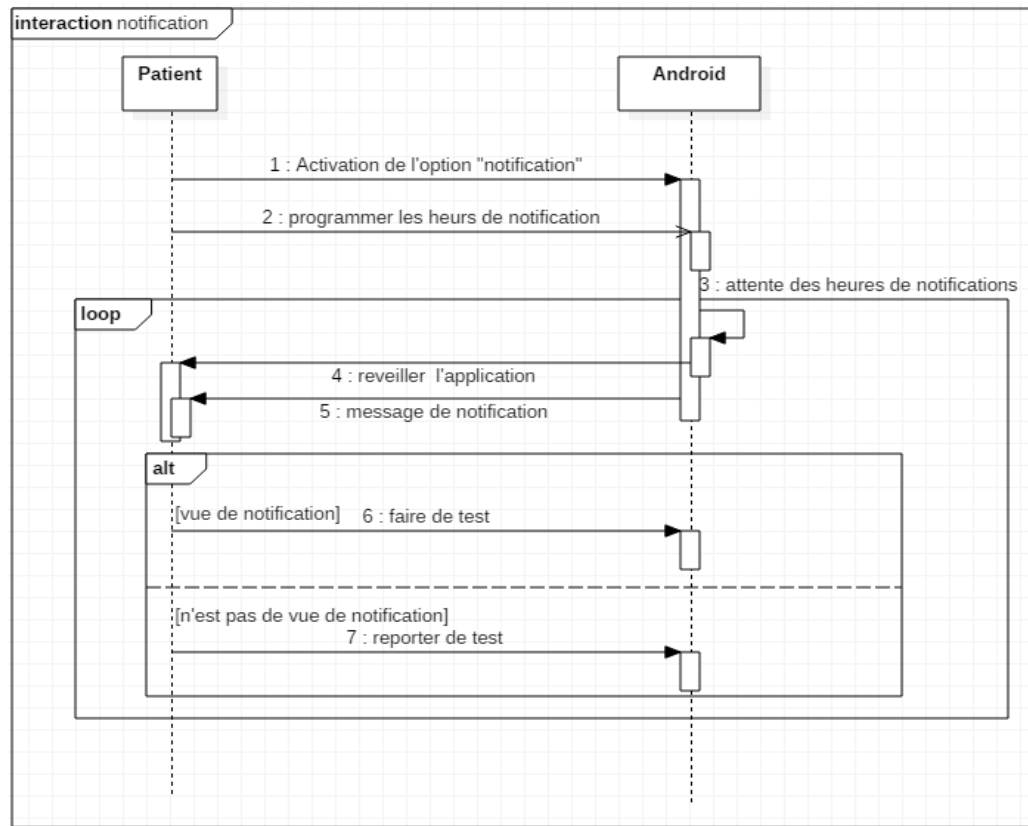


Figure III.7. Diagramme de séquence « Notification »

2.3. Diagramme d'activité

Le diagramme d'activité donne une vision des enchaînements des activités relatives à une opération où un cas d'utilisation.

2.3.1. Diagramme d'activité « authentification »

Après le démarrage de l'application, le système affichera la page d'authentification. Une fois le login et le mot de passe saisis, le système vérifie la validité de ces derniers au niveau de la base de données pour afficher la page d'accueil, en cas d'erreur le système réaffiche la page d'authentification. Ci-dessous est présenté le diagramme d'activité du cas d'utilisation « S'authentifier ».

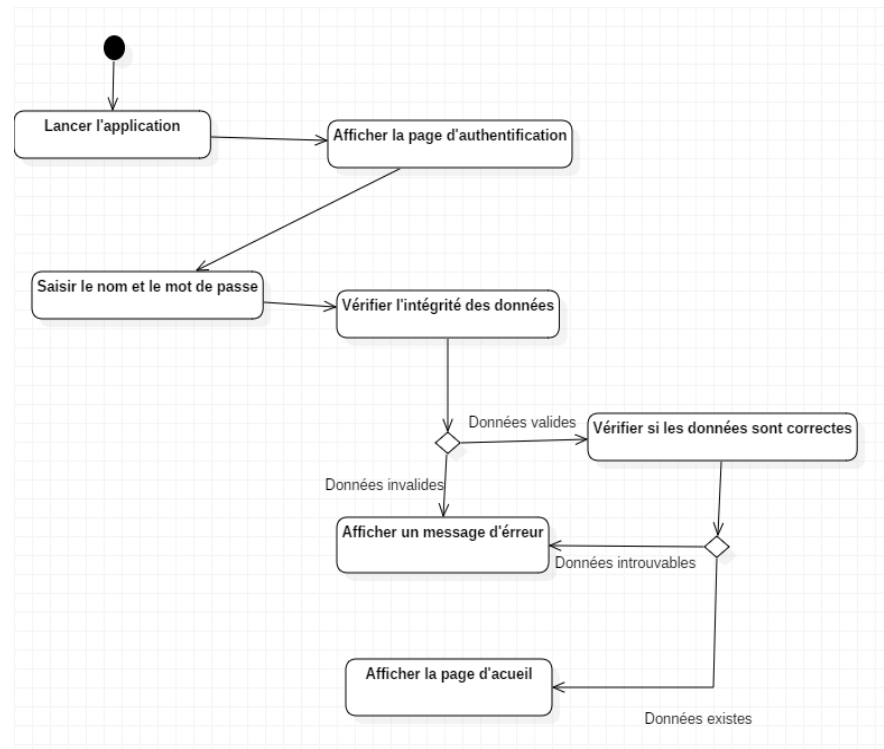


Figure III.7. Diagramme d'activité « authentification »

2.3.2. Diagramme d'activité « ajouter un utilisateur »

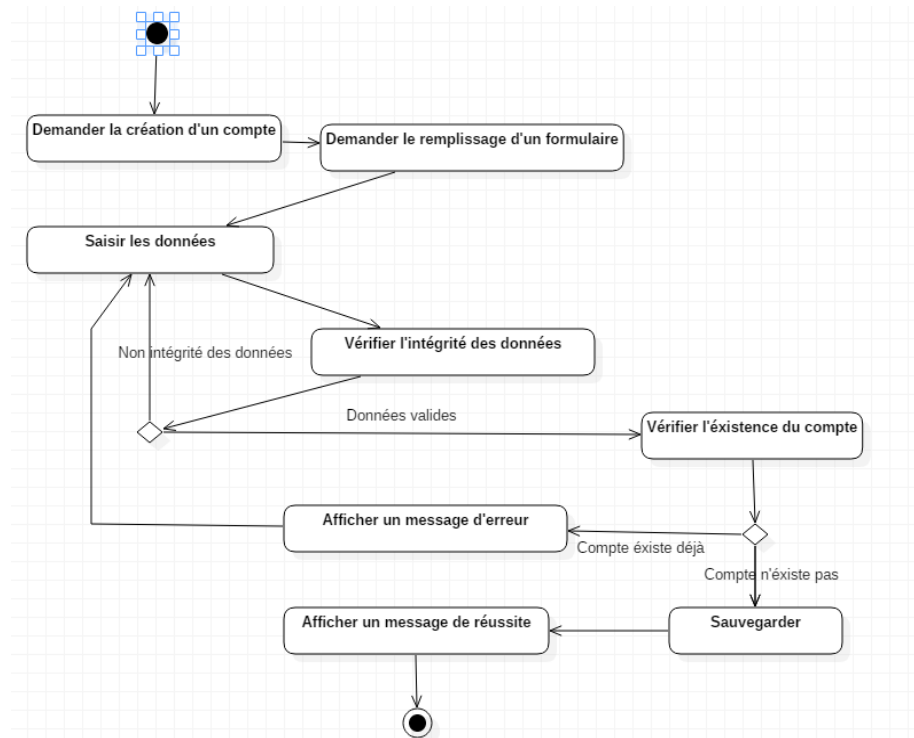


Figure III.8. Diagramme d'activité « ajouter un utilisateur ».

2.4. Diagramme de classe

Un diagramme de classe se définit comme étant un ensemble de classes contenant des attributs et des opérations, reliées les unes aux autres par des relations et ceci en ayant des conditions de participation (cardinalités).

Le diagramme de classe retenu à la fin de conception est présenté dans la figure suivante :

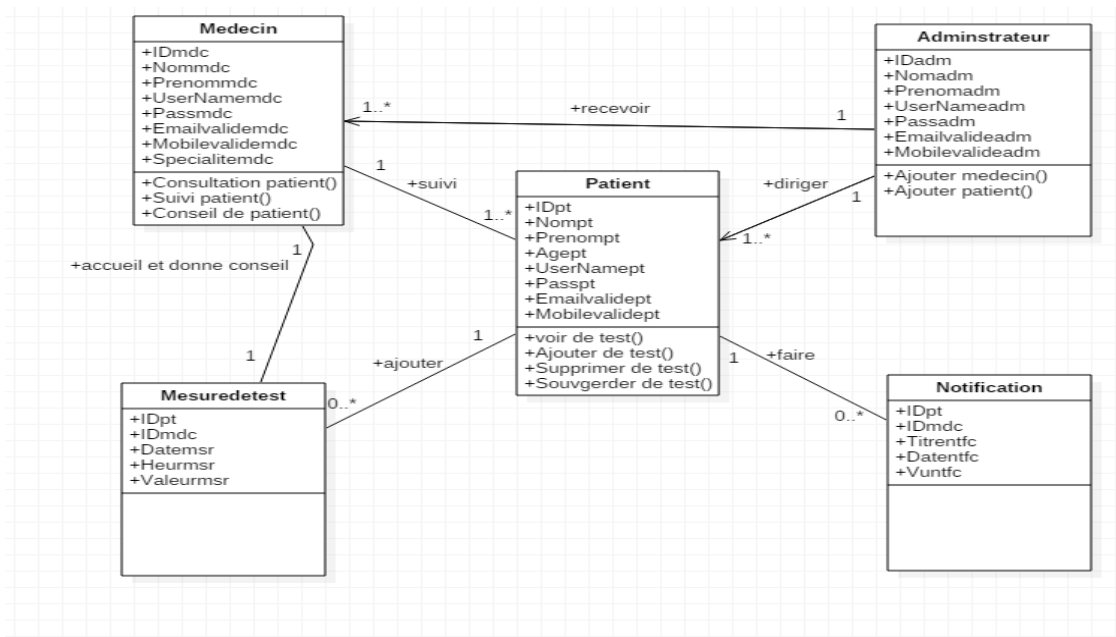


Figure III.9. Diagramme de classes.

3. Conclusion

Dans ce chapitre, nous avons analysé le fonctionnement de notre système grâce à des diagrammes UML. Nous avons décrit d'abord les cas d'utilisation en recensant de manière textuelle toutes les interactions entre les acteurs et le système suivi par un diagramme des cas d'utilisation. Par la suite, Nous avons complété cette description textuelle par des diagrammes des séquences. Et à la fin, définissant les relations entre les entités, nous sommes parvenus à concevoir le diagramme de classes. Ce qui nous permet de commencer le développement de l'application, ainsi que la base de données optée.

CHAPITRE IV : RÉALISATION ET IMPLÉMENTATION

Chapitre IV : Réalisation et Implémentation

1. Introduction

Après avoir réalisé la conception appropriée à notre application, nous allons dans ce chapitre décrire le processus de réalisation de notre application mobile SUGAR DIAB. Ceci en spécifiant la programmation, l'implémentation de la base de données et un aperçu sur les interfaces de notre application.

2. Programmation de développement

Pour développer des applications sous Android ou un site web Atome, un ensemble d'outils est obligatoire.

2.1. Le langage java

Le langage Java est un langage programmation lors de sa création les années 90 par Sun Microsystems appartient de langage programmation orienté objet. Java a l'avantage d'être modulaire, Cela signifie que les entreprises comptent sur eux pour créer leurs logiciels.

2.2. Le langage XML

Le XML (Extensible Markup Language) est un langage de balises. Est une recommandation du W3C, Le XML se veut compréhensible par tous car il permet de créer notre propre vocabulaire grâce à un ensemble de règles et de balises personnalisables. Donc faciliter les échanges de données entre les machines et les logiciels, en quelque sorte un langage HTML amélioré permettant de définir de nouvelles balises.

2.3. Le langage HTML5

Le HTML5 est une version du célèbre format HTML utilisé pour concevoir les sites Internet. Celui-ci se résume à un langage de balisage lancé en octobre 2014, cette version apporte de nouveaux éléments. Le HTML5 est un format de langage développé par le W3C.

HTML5 regroupe plusieurs technologies destinées notamment au développement d'applications : HTML5, CSS3 et JavaScript. Aussi de DHTML : Dynamics HTML, en référence à ces technologies qui rendent les pages web aptes à se modifier au fil de la consultation, directement sur le navigateur web.

2.4. Atom (Editeur de texte)

Atom est un éditeur de texte portable créé par GitHub et développé au-dessus de la plate-forme d'application de bureau Electron. Cela signifie qu'il peut être utilisé en tant qu'éditeur de texte pour la programmation de base jusqu'à un IDE complet. Il est également extrêmement personnalisable, il fournit des milliers de paquets conçus par la communauté (mise en évidence de la syntaxe, amélioration de la définition d'UX, etc.) et des thèmes adaptés aux besoins de chacun. Il est également disponible sous Windows, MacOS et Linux.

3. Les bases de données

Elle permet de stocker une collection d'informations organisées afin d'être facilement consultables, gérables et mises à jour. Il existe plusieurs solutions pour le stockage des données les plus importantes utilisées par Android : Firebase

3.1. Les bases de données Firebase

Notre application appartient à la famille des réseaux sociaux, il nous est apparu que la meilleure solution pour la base de données est la mettre dans les Cloud. Pour cela, nous avons utilisé la base de données Firebase.

3.2. Définition

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application ([Android](#), [iOS](#), [Javascript](#), [Node.js](#), [Java](#), [Unity](#), [PHP](#), [C++](#) ...).[20]

Firebase est un backend très utile pour le développement d'applications mobiles. Il offre de nombreuses fonctionnalités telles que l'authentification et la sécurité, la base de données en temps réel et le stockage de fichiers, les analyses, les notifications et bien d'autres.

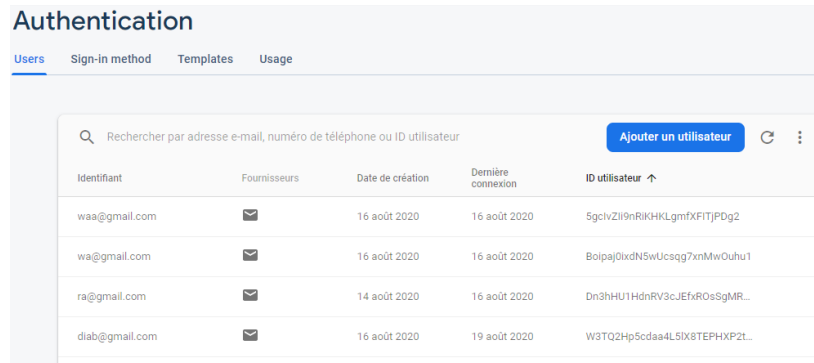
Il fournit iOS, Web, NodeJS, C ++, Java Server et le SDK pour Android qui est l'objet de notre étude.

La base de données Firebase est une base de données NoSQL qui stocke ses données sous la forme d'objets JSON hiérarchiques. Il n'y a pas de tables ou d'enregistrements de forme quelconque, comme une base de données SQL aurait normalement, juste des nœuds qui constituent une structure clé-valeur.

Firebase propose un certain nombre de services, notamment :

3.2.1. Analytiques – ils offrent un service gratuit et illimité. Ils fournissent des données sur le comportement des utilisateurs dans les applications iOS et Android.

3.2.2. Authentification - L'authentification Firebase permet aux développeurs de créer facilement des systèmes d'authentification sécurisés.



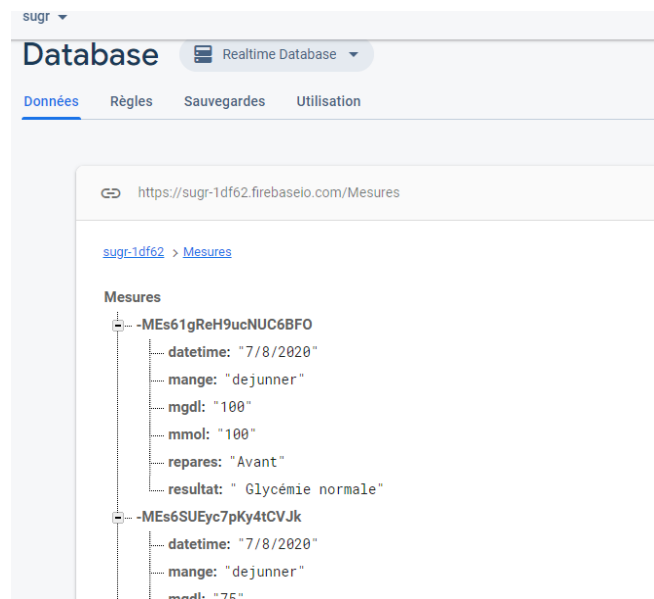
The screenshot shows the 'Users' tab in the Firebase Authentication console. It features a search bar at the top with the placeholder text 'Rechercher par adresse e-mail, numéro de téléphone ou ID utilisateur'. To the right of the search bar is a blue button labeled 'Ajouter un utilisateur'. Below the search bar is a table with the following columns: 'Identifiant', 'Fournisseurs', 'Date de création', 'Dernière connexion', and 'ID utilisateur ↑'. The table contains four rows of user data.

Identifiant	Fournisseurs	Date de création	Dernière connexion	ID utilisateur ↑
waa@gmail.com	✉	16 août 2020	16 août 2020	5gclvZli9nRikHKLgmfxFITjPDg2
wa@gmail.com	✉	16 août 2020	16 août 2020	Boipaj0ixdN5wUcsqg7xmMwOuhu1
ra@gmail.com	✉	14 août 2020	16 août 2020	Dn3hHU1HdnRV3cJEfxROsSgMR...
diab@gmail.com	✉	16 août 2020	19 août 2020	W3TQ2Hp5cdaa4L5IX8TEPHXP2t...

Figure IV.1.FirebaseAuthentificationDatabase

3.2.3. Cloud Messaging - un outil de messagerie multiplateforme qui permet de recevoir et de diffuser de manière fiable des messages sur iOS, Android et le Web sans frais.

3.2.4. Réaltine Data base - une base de données NoSQLi qui permet le stockage des données et la synchronisation entre les utilisateurs en temps réel.



The screenshot shows the 'Database' tab in the Firebase Realtime Database console. It displays a JSON structure for a database named 'Mesures'. The URL bar shows 'https://sugr-1df62.firebaseio.com/Mesures'. The JSON structure is as follows:

```

{
  "Mesures": {
    "-MEs61gReH9ucNUC6BFO": {
      "datetime": "7/8/2020",
      "mange": "dejunner",
      "mgdl": "100",
      "mmol": "100",
      "repares": "Avant",
      "resultat": "Glycémie normale"
    },
    "-MEs6SUEyc7pKy4tCVJk": {
      "datetime": "7/8/2020",
      "mange": "dejunner",
      "mgdl": "75"
    }
  }
}

```

Figure IV.2Firebase Realtime Database

3.2.5. Crashlytics - FirebaseCrashlytics est un reporteur de crash en temps réel qui aide les développeurs à suivre, prioriser et résoudre les problèmes de stabilité.

3.2.6. Cloud Firestore est une base de données de développement mobile, Web et serveur flexible et évolutive de Firebase et de Google Cloud Platform. Comme la base de données en temps réel de Firebase, elle maintient les données synchronisées entre les applications clientes et pour le mobile et le Web afin de créer des applications réactives.

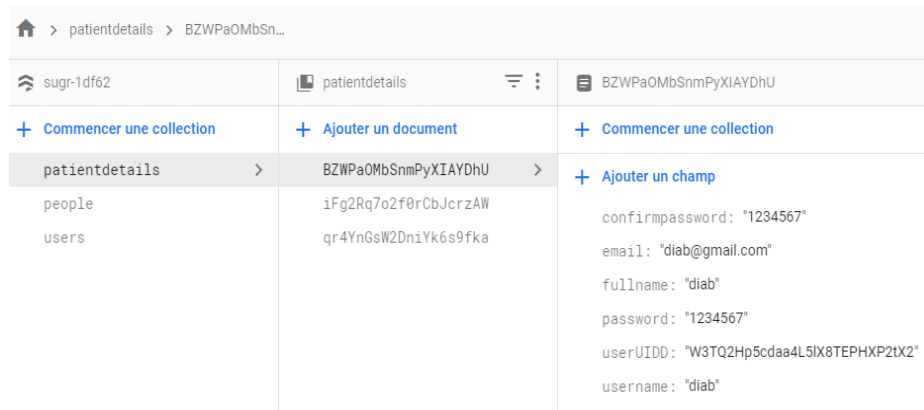


Figure IV.3 Firebase Cloud Firestore Database

3.3. Objectif

L'objectif premier de Firebase est de libérer de la complexité de création et de la maintenance d'une architecture serveur, tout en garantissant une simplicité dans l'utilisation. Pour cela, Firebase a été décomposée en plusieurs produits extrêmement riches et adaptés au monde du mobile. Firebase a été conçu principalement pour les développeurs mobile afin de leur permettre de créer plus facilement des architectures serveur performantes.

3.4. Architecture avec Cloud Firestore

Un ensemble de servlets Java qui s'exécute dans l'environnement flexible App Engine s'enregistrent en tant que programmes d'écoute avec Firebase. Les servlets répondent aux nouveaux journaux d'événements utilisateur et traitent les données des journaux. Ils utilisent des transactions pour garantir qu'un seul servlet gère chaque journal d'événements utilisateur.

Le schéma ci-dessous illustre l'architecture du serveur Playchat.

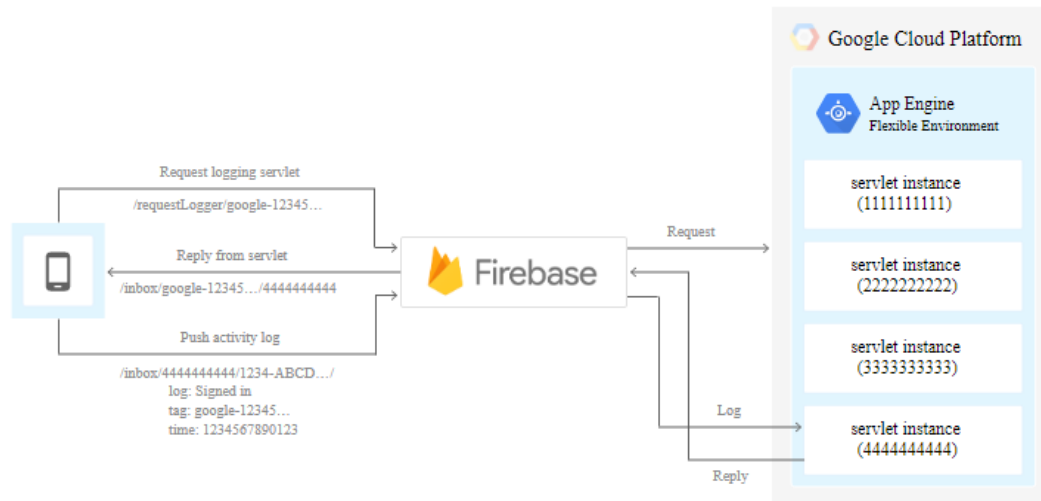


Figure IV.4. L’architecture du serveur Playchat.

3.5.Ajouter Firebase à l’application SugarDiab

Ouvrir <https://firebase.google.com/console/> Et connecter à l'aide des informations d'identification du compte gmail.

Étape 1 : Enregistrer l’application dans la plateforme de Firebase en remplissant les champs nécessaires.

Figure IV. 5 Première étape de l’implémentation.

Étape 2: Télécharger le fichier contenant les services de Google, et le placer dans le dossier sous lenom « app » située dans le package de projet

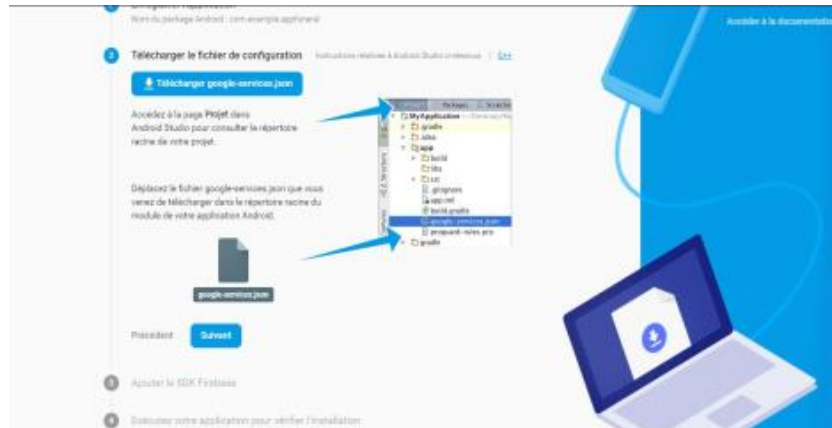


Figure IV. 6 deuxième étape de l'implémentation.

Étape 3 : Ajouter le SDK de Firebase au niveau de projet et de m'application, puis appuyer sur « synchroniser » dans la barre qui apparaît sans l'IDE.



Figure IV. 7 Troisième étape de l'implémentation

```
dependencies {  
  
    implementation fileTree(include: ['*.jar'], dir: 'libs')  
  
    implementation 'com.google.firebase:firebase-firestore:21.4.3'  
    implementation 'com.google.firebase:firebase-database:19.2.0'  
    implementation 'com.google.firebase:firebase-auth:17.0.0'  
    implementation 'com.google.firebase:firebase-core:16.0.4'  
}
```

Figure IV. 8 Fichier build.gradle au niveau de l'application.

```
dependencies {  
    classpath 'com.android.tools.build:gradle:3.5.1'  
    classpath 'com.google.gms:google-services:4.3.3'  
}  
  
allprojects {  
    repositories {  
        google()  
        jcenter()  
        maven {  
            url "https://maven.google.com"  
        }  
    }  
}
```

Figure IV. 9 Fichier build.gradle au niveau du projet.

Étape 4 : Exécuter l'application pour vérifier l'installation de SDK

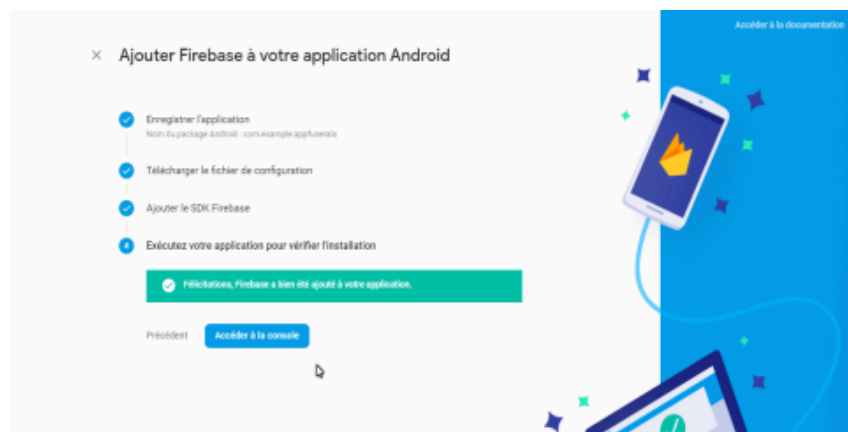


Figure IV.10 Dernière étape de l'implémentation

Enfin, accéder à la console de Firebase

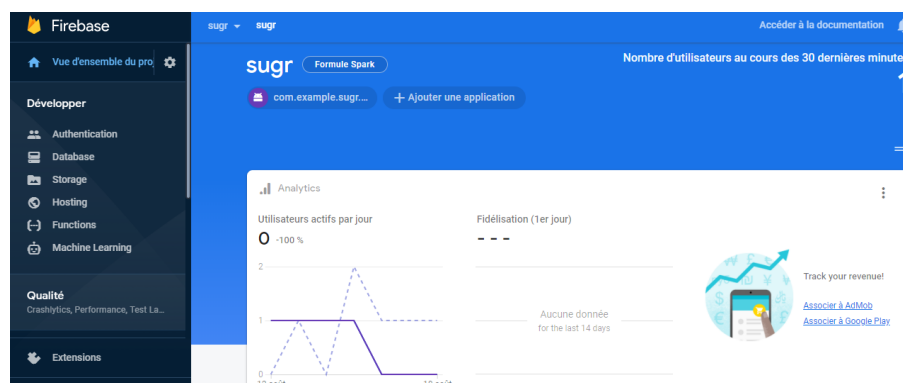


Figure IV. 11 Console de Firebase.

3.6.Connecter Firebase avec Android

En suivant ces étapes :

Cliquer sur Outils>Firebase pour ouvrir la fenêtre Assistant.

Cliquer pour développer l'une des fonctionnalités répertoriées (par exemple,Authentification), puis cliquer sur le didacticiel Premiers pas pour se connecter à Firebase et ajouter le code nécessaire à l'application.

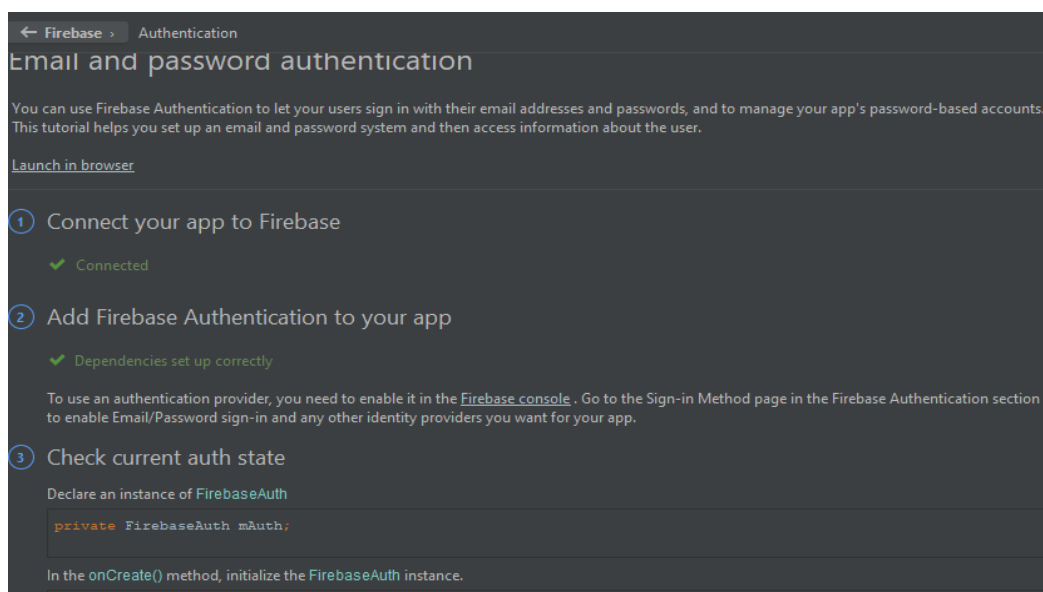


Figure IV.12. Connexion du Firebase avec Android.

➤ Principe de fonctionnement

L'application va envoyer une demande servlet (connexion cloud firestore)

```

firebaseAuth.createUserWithEmailAndPassword(textEmail , textPassword)
    .addOnCompleteListener((task) -> {
        if (task.isSuccessful()) {
            Toast.makeText( context: patient.this , {text: "Registration Complete " , Toast.LENGTH_SHORT).show();
            Map < String, Object > user = new HashMap <>();
            user.put("fullname" , textFullName);
            user.put("username" , textusername);
            user.put("email" , textEmail);
            user.put("password" , textPassword);
            user.put("confirmpassword" , textConfirmpassword);
            user.put("userUIDD",firebaseAuth.getCurrentUser().getUId());
            fbstore.collection( collectionPath: "patientdetails")
                .add(user)
                .addOnSuccessListener(new OnSuccessListener < DocumentReference >() {
                    @Override
                    public
                    void onSuccess(DocumentReference documentReference) {
                        startActivity(new Intent(getApplicationContext() , login.class));
                    }
                });
        } else { Toast.makeText( context: patient.this , {text: "authentication failed" , Toast.LENGTH_SHORT).show();
    }}.addOnFailureListener((e) -> {
        Log.d( tag: "TAG" , msg: "Firebase Exception " + e.getMessage());
    });
}

```

Figure IV.13. Connexion cloud firestore

4. Présentation de quelques interfaces de l'application

4.2.Accueil

4.2.1. Administrateur

C'est une page qui contrôle l'ajout et la suppression du patient et voit toutes les informations personnelles et comment le patient est dirigé vers le médecin par l'administrateur.

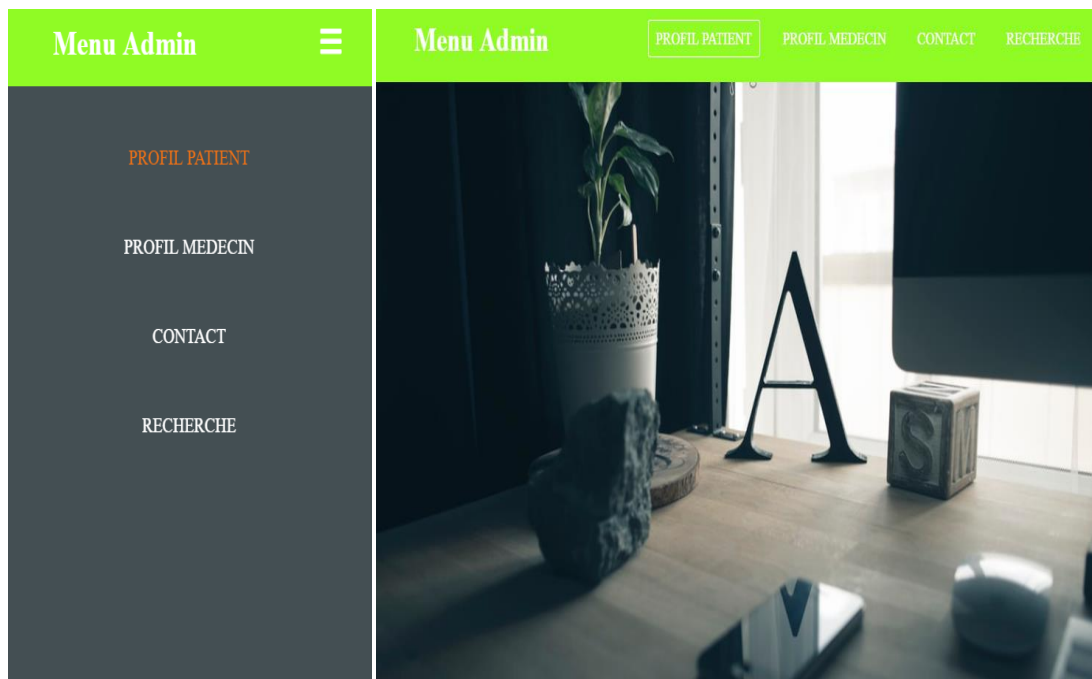


Figure IV.14. Page d'accueil (administrateur).

4.2.2. Médecin

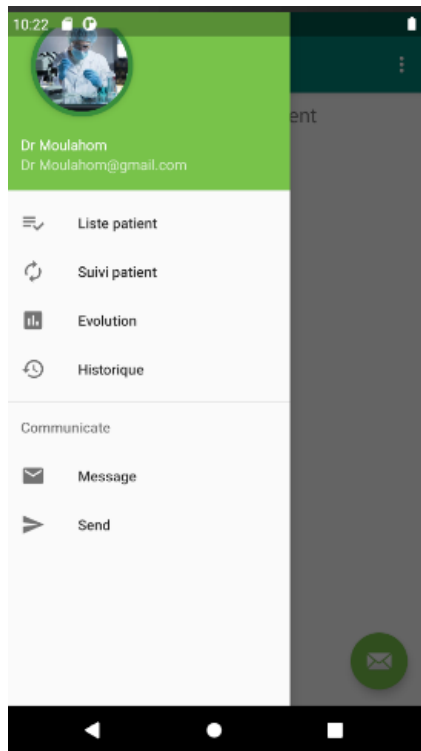


Figure IV.15. Page d'accueil (médecin)

4.2.3. Patient

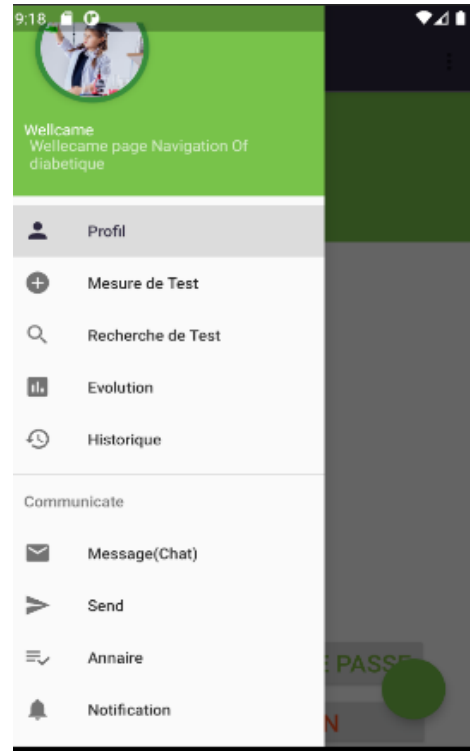


Figure IV.16. Page d'accueil (patient)

4.3. Authentification

4.3.1. Administrateur

C'est la page qui permet à l'administrateur de s'authentifier.

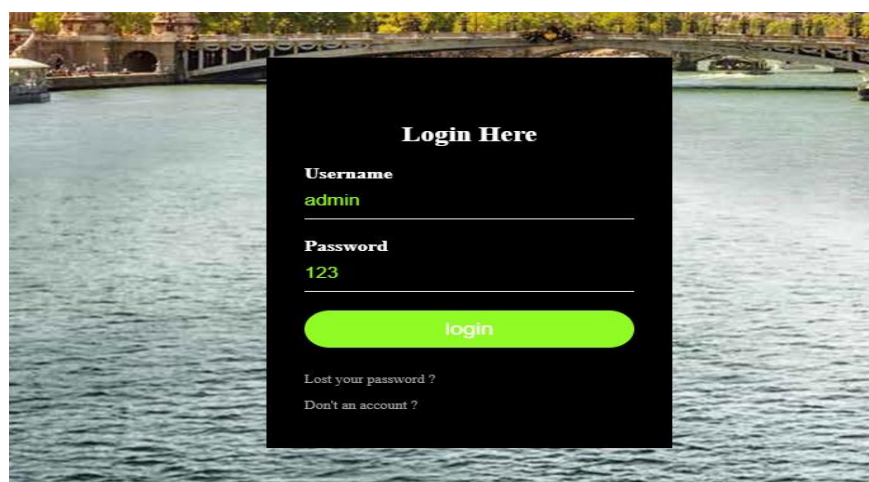


Figure IV.17. La fenêtre d'authentification (Administrateur).

4.3.2. Patient/Médecin

Pour les patients et les médecins, qui sont déjà inscrits dans l'application.

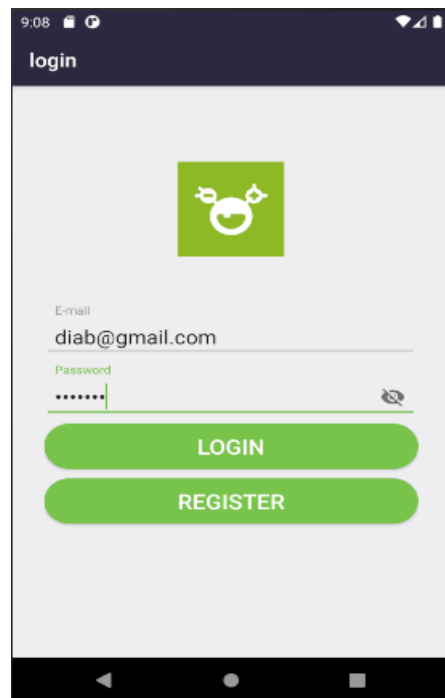


Figure IV.18. Fenêtre d'authentification (patient).

4.4. Ajout d'un patient

Pour créer un nouveau compte d'un patient.

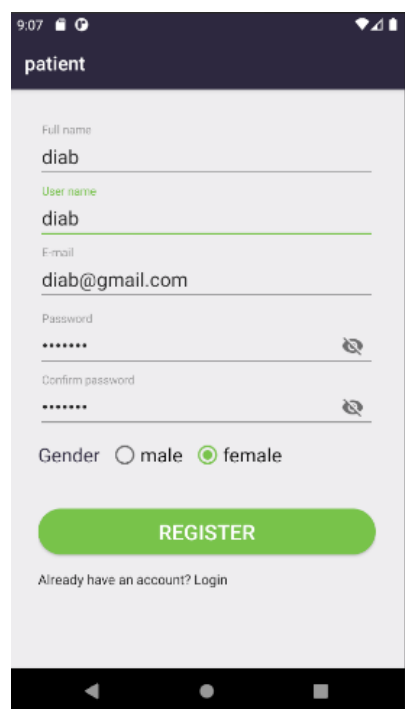


Figure IV.19. Ajouter un patient.

4.4.1. Vérifier le numéro de téléphone

C'est pour sécuriser le compte, il s'agit de la saisie du numéro de téléphone puis l'envoi du code de confirmation.

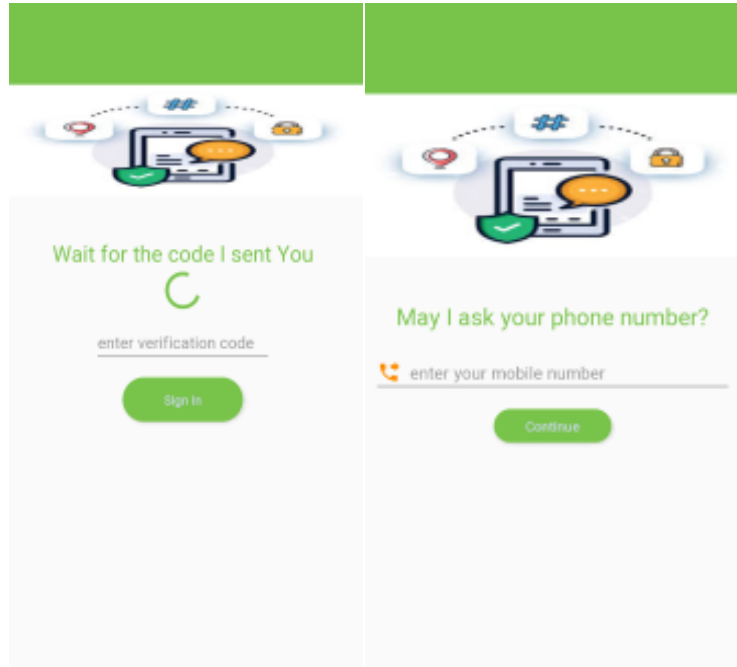


Figure IV.20. Vérification du numéro de téléphone.

4.5.Diagnostic

Il s'agit d'un ensemble de questions à poser au patient pour déterminer l'état de développement de la glycémie.

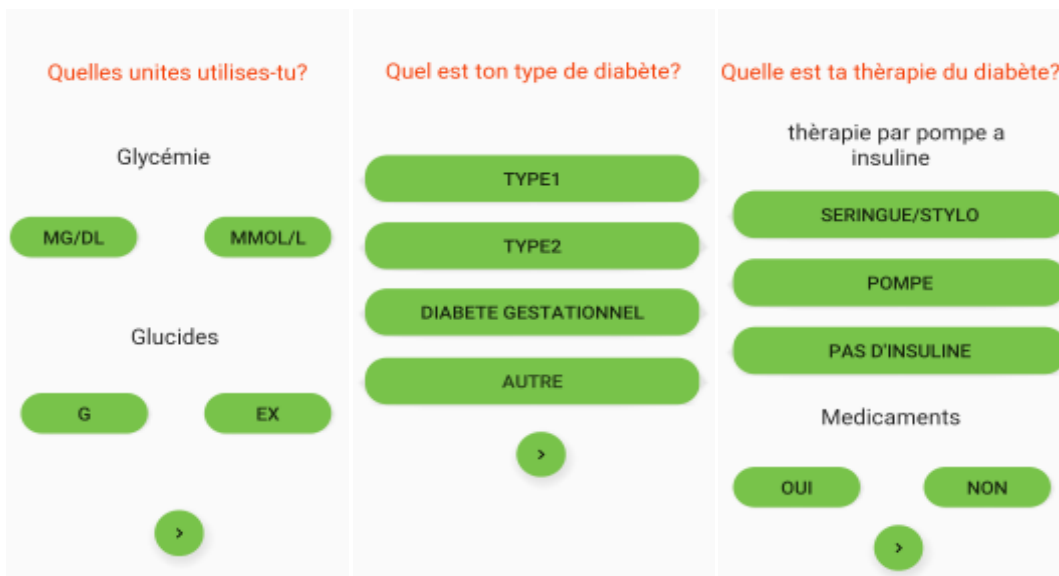


Figure IV.21. Fenêtres de diagnostic.

4.6.Profil du patient

Contenant les informations du patient et la déconnexion.

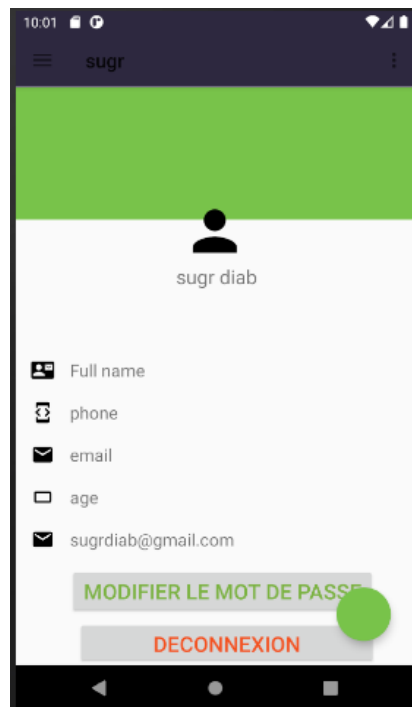


Figure IV.22. Fenêtre de profil d'un patient.

4.7.Mesure de test

Sur la fenêtre d'accueil du patient, en cliquant sur le bouton « Mesure de test », une fenêtre « test » apparaîtra, le patient saisit sa valeur de glycémie, il choisit son état : avant/après (petit déjeuner, déjeuner ou dîner) puis, il clique sur « Save ». Deux cas sont possibles : le cas d'une valeur normale va être affichée en vert et le cas d'une valeur anormale (Hyperglycémie en rouge /Hypoglycémie en jaune).

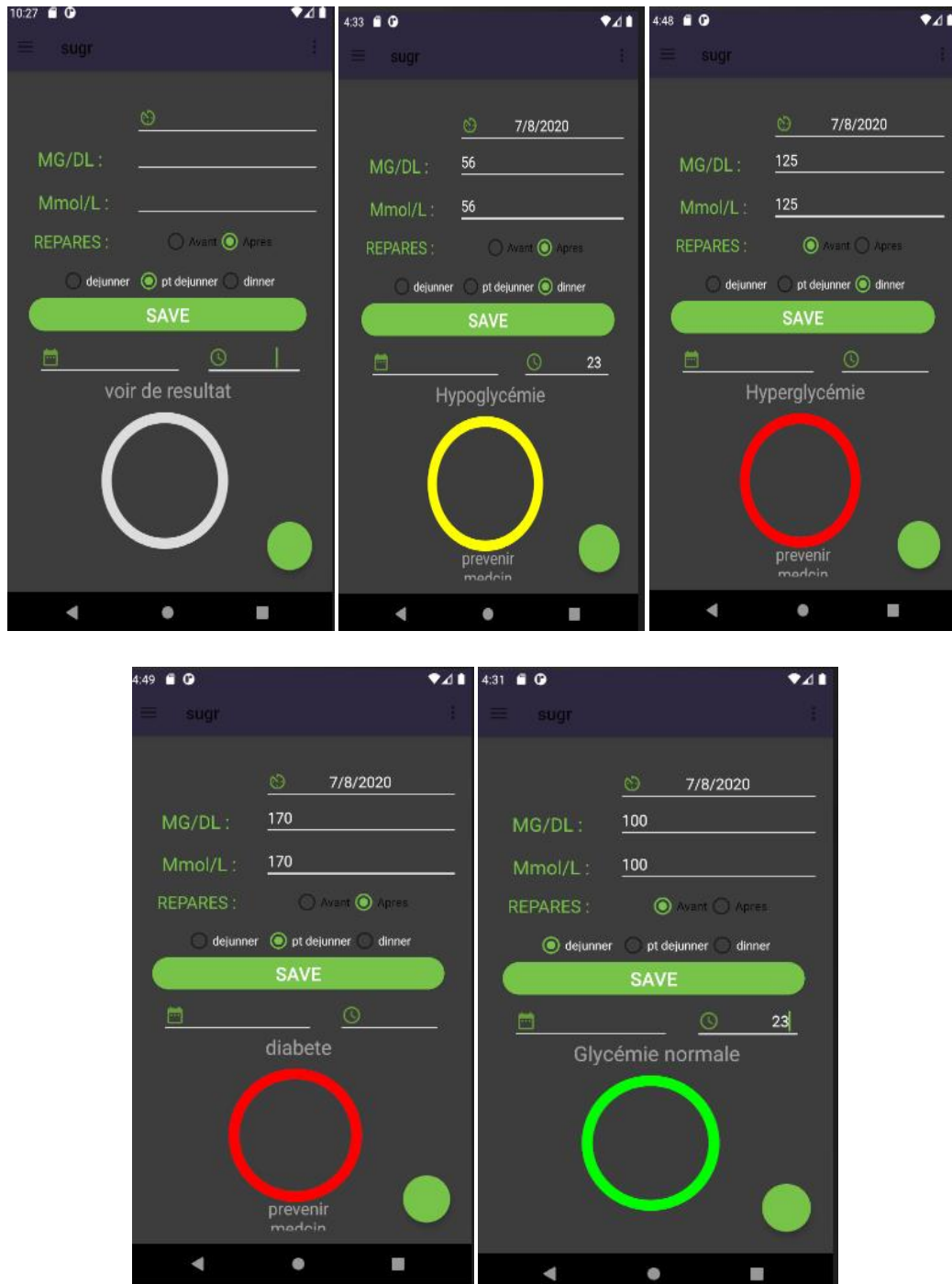


Figure IV.23. Fenêtres de mesure de Glycémie.

4.8.Evaluation de la Glycémie

Après avoir saisi les valeurs de glycémie en cliquant sur le bouton «Insert », la valeur sera affichée sur le graphique par heure et par jour.



Figure IV.24. Page d'évaluation de glycémie par graph.

4.9. Liste des Medcins

Le patient sélectionne le médecin qui le contacte en urgence et lui demande des conseils.

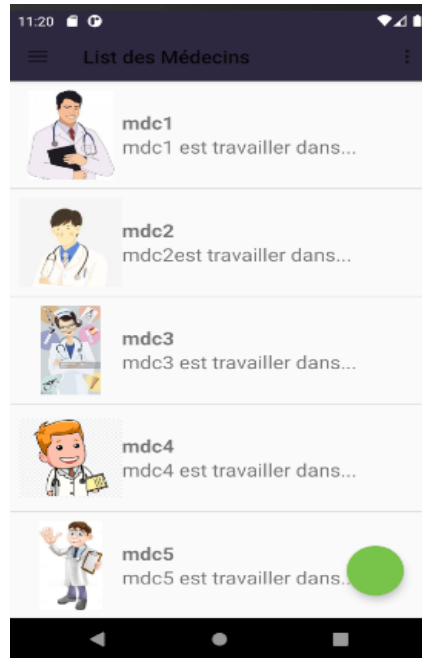


Figure IV.25. Page liste des médecins.

4.10. Notification

Activer la notification consiste à choisir le bouton « Open Time picker » puis régler l'heure et ouvrir l'alarme, lorsque le message de notification est affiché « c'est l'heure de la mesure ».

Désactiver la notification consiste à choisir le bouton « Cancel Alarm ».



Figure IV.26. Pages Alarme de notification.

4.11. Envoyer un message à un médecin

Pour envoyer un message à un médecin, il faut d'abord indiquer son e-mail puis le message à envoyer.

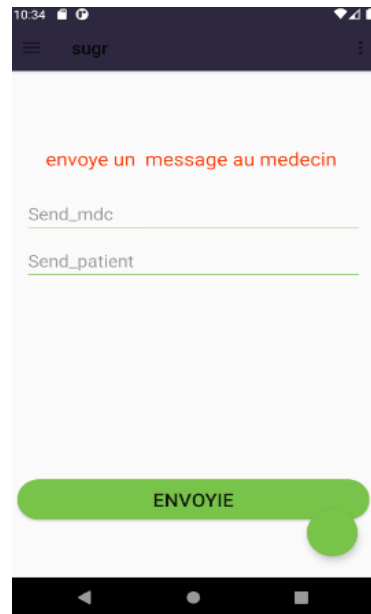


Figure IV.27. Envoi d'un message à un médecin.

5. Conclusion

Dans ce dernier chapitre, nous avons présenté l'environnement de développement de notre application et les différents outils et bibliothèques utilisés. Par la suite, nous avons présenté les différentes interfaces de notre application.

CONCLUSION GÉNÉRALE

Conclusion générale

Le diabète représente un sérieux problème de santé. Des grands efforts ont été mis en dispositif pour lutter contre cette maladie et multiples approches ont contribué à la prise en charge et le bon suivi des patients. Cela nous a amené à réfléchir sur un moyen permettant le bon suivi des patients atteints de cette maladie.

Pour se faire, nous avons présenté les différentes étapes de réalisation d'une application mobile sous Android pour le suivi des diabétiques. Nous avons commencé d'abord, par des généralités sur la programmation mobile sous Android. Nous avons également abordé la maladie du diabète. Ensuite, nous avons présenté des diagrammes UML permettant de bien comprendre la structure statique et dynamique du système. Enfin, nous avons présenté la base de données utilisée.

Toutefois, notre objectif principal est de concevoir un système dédié aux patients diabétiques et les personnes qui sont autour d'eux pour résoudre leurs problèmes et faciliter leurs tâches.

À l'issue de ce projet, notre application restera toujours ouverte à des perspectives d'amélioration. Nous envisageons d'ajouter une autre langue pour une très grande utilisation. Ainsi, nous envisageons ainsi d'ajouter de nouvelles fonctionnalités selon les recommandations des utilisateurs.

BIBLIOGRAPHIE

Bibliographie

- [1] Application mobile web ou natif_ekitopeople.htm.InBOUDJADI.F& CHEKLAT.L, "Conception et réalisation d'une application mobile sensible au contexte pour un musée, Juillet 2013.
- [2] Jean-FrancoisLalande , "Développement sous Android ", May 2015 .
- [3][http ://www.iphonedevoloppeur.fr/2011/04/05/fonctionnementduneapplicationiphone/](http://www.iphonedevoloppeur.fr/2011/04/05/fonctionnementduneapplicationiphone/)
- [4] N.TIEN THINH. Système d'exploitation pour les mobiles, Hanoi, Juillet 2009.
- [5] [//fr.wikipedia.org/wiki/IOS\(Apple\)](http://fr.wikipedia.org/wiki/IOS(Apple)), consulté le 01/01/2020.
- [6] Hugues DELALIN, Departement Service et Réseaux de Communication IUT de Lens - Université d'Artois, Année 2005/2006.
- [7]N.TIEN THINH. Système d'exploitation pour les mobiles, Hanoi, Juillet 2009.
- [8] <https://fr.wikipedia.org/wiki/Bada>, consulté le 01/01/2020.
- [9] Gauthier Picard ,Programmation sous Android, 2012.
- [10] Wikipédia, <https://fr.wikipedia.org/wiki/Android>, consulté le 22/02/2020.
- [11] B.D, BY, Conception et réalisation d'une application mobile sous Android « Cas d'étude : Le suivi des diabétiques », Université Abderrahmane Mira, Bejaia, 2016.
- [12]Jean-Francois Lalande, Développement sous Android, May 2015.
- [13]«Developer Guides,» [En ligne]: <https://developer.android.com/guide>, In A.S, Application Android pour les funérailles de la ville de M'Sila, M'sila, 2019.
- [14] <https://fr.yeeply.com/blog/apprendre-developpement-android/>, consulté le 01/01/2020.
- [15]<https://www.gralon.net/articles/materiel-et-consommables/materiel-medical/article-latelemedecine---definition-et-avantages-4828.htm>, consulter le 14/6/2019 à 22 h 27.

- [16] Organisation Mondiale de la santé, Rapport Mondial sur la santé, <https://apps.who.int/iris/bitstream/handle/10665/254648/9789242565256-fre.pdf;jsessionid=99B6CFA6A0B65477E2123161B81DD7CB?sequence=1>.
- [17] Dubois-Laforgue, Danièle et Timsit, José. Diabète de type 1 et environnement. s.l. : M/S 2000, 2000, In L.K, « Réseau social pour suivi des patients diabétiques », université de M'sila, 2019.
- [18] <https://www.gammeonetouch.fr/a-propos-du-diabete/commencez-votreparcours/complications-du-diabete>, consulté le 12/08/2020.
- [19] American Diabetes Association. (ADA) Standards of Medical Care in Diabetes–2018. Diabetes Care 2018 ; vol 41, Suppl. 1. Version en ligne, In L.K, « Réseau social pour suivi des patients diabétiques », université de M'sila, 2019.
- [20] Wikipedia, <https://fr.wikipedia.org/wiki/Firebase>, consulté le 25/06/2020.

ملخص

تم إنجاز مذكرة التخرج هذه في إطار مشروع التخرج للحصول على شهادة ماستر في الإعلام الآلي، يتعلق بتطوير تطبيق اندرويد لأجل المتابعة المستمرة لمرضى السكري في مركز إستشفائي. لأجل ذلك قدمنا مفاهيم نظرية تتعلق بعالم الأندرويد وتطبيقات الهاتف المحمول و اخترنا منهجية UML لشرح الخصائص الوظيفية الأساسية لتطبيقنا.

ومن أجل إنجاز المشروع، استعملنا برنامج Android Studio وفيما يخص قاعدة البيانات اعتمدنا Firebase التي سهلت عملية التطوير.

كلمات مفتاحية: *Android، مرضى السكري، UML، Firebase.*

Abstract

This memorial has been realized as part of graduation project to get master degree in computer science.

This project aims to develop an Android application to follow up people with diabetes in a hospital. For this, we presented theoretical concepts which are related to Android and mobile applications world and we opted for UML approach to describe the main features of our system.

For the realization, we used integrated developing environment Android Studio, then we implemented Firebase services which made developing easier for us.

Keywords: *Android, People with diabetes, UML, Firebase.*

Résumé

Ce mémoire a été réalisé dans le cadre du projet de fin d'études pour l'obtention du diplôme de master en informatique. Il consiste à développer une application mobile sous Android pour le suivi des diabétiques dans un centre hospitalier.

Pour ce faire nous avons présenté les concepts théoriques liés au monde d'Android et Applications mobile et nous avons opté pour l'approche UML pour décrire les principales fonctionnalités de notre système. Pour la réalisation, nous avons utilisé l'environnement de développement intègre Android Studio. Ainsi, nous avons implémenté les services de Firebase qui nous a facilité le développement

Mots clés : *Android, diabétiques, UML, Firebase.*