



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE IBN KHALDOUN – TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

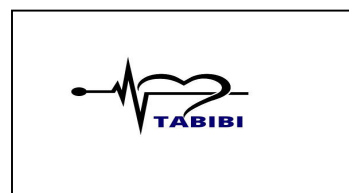
Pour l'obtention du diplôme de **Master**

Spécialité : Génie Logiciel

En vue de créer une startup



Par :



Zoukh Ahlem

Sur le thème

TABIBI : Application Mobile pour la gestion de La relation docteur patient

Soutenu publiquement le 13 /06 / 2024 à Tiaret devant le jury composé de :

Mr. AID Lahcene	MCA Université Tiaret	Président
Mr. HAMEURLAINE Messaoud	MCA Université de Tissemsilt	Encadrant
Mr. MOSTEFAOUI Kadda	MCA Université Tiaret	Examineur
Mr Mohamed Amine Bouzid	MCA Université Tiaret	Représentant de l'incubateur
Mr Announ Samia	Médecin Généraliste Tiaret	Représentant du partenaire économique

2023-2024

Remerciement

En premier lieu, nous remercions ALLAH, le tout puissant et miséricordieux, qui nous a donné la force le courage et la patience d'accomplir ce modeste travail.

Nos remerciements et nos profondes gratitudes vont à notre Encadreur Monsieur HAMEURLAINE Messaoud pour ses précieux conseils et pour tout le soutien et l'orientation. D'avoir bien voulu diriger notre travail, et d'avoir donné le meilleur de son savoir et de son aide, et surtout d'avoir fait preuve de beaucoup de patience, son aide durant toute la période du travail.

Nous tenons aussi à remercier les membres du jury pour leur précieux temps accordé à l'étude de notre mémoire.

Nous remercions nos enseignants pour leurs efforts, nos parents et nos proches pour l'amour et le soutien constant qu'ils nous ont témoigné tout au long de notre parcours. Merci à toutes et tous nos ami(e)s pour leurs encouragements.

Dédicace

Remerciez Dieu, assez et priez pour le bien-aimé Mustafa, sa famille et ceux qui
sont fidèles

Et pourtant.

Je voudrais remercier Dieu et le remercier de m'avoir aidé à compléter cette note
À ceux que Dieu a dotés de prestige et de révérence. A celui qui m'a appris à
donner sans attendre. À qui je porte fièrement son nom

Mon cher père

À mon ange dans la vie. Au sens de l'amour et au sens de la tendresse et de la
dévotion. À la grâce de la vie et au mystère de l'existence. Pour qui sa prière
était le secret de ma réussite et sa tendresse un baume chirurgical. Aux

amoureux les plus chers

Maman bien-aimée

Au phare de la science et des scientifiques. Au majestueux édifice (Université de
Tiaret)

A ceux que j'ai gagnés un cadeau du destin aux Compagnons du chemin dont
nous avons partagé les moments, chers amis, que Dieu les bénisse et leur
accorde

À tous ceux qui ont eu un impact sur ma vie. Et à tous ceux que mon cœur a
aimés et que ma plume a oubliés

Ahlem.

Résumé

Ce projet de fin d'étude présente la conception et la réalisation d'une application mobile pour le suivi des patients. Le système de santé Algérien se modernise de jour en jour se trouve dans la nécessité d'informatiser l'information médicale. Cette application destinée aux smartphones vise à satisfaire les besoins des patients. Afin d'atteindre cet objectif, il nous a été proposé de réaliser une application sous Android en proposant divers options et fonctionnalités par exemple on peut contacter un médecin, on a choisi de modéliser notre système avec le formalisme UML pour réaliser l'étude conceptuel du système, on a utilisé le Framework Flutter sous l'environnement de développement Android Studio, et FireBase l'implémentation de la base de données.

Mots clés : Flutter, Android Studio, VS code, Android, Dart, FireBase, Dossier médical

Abstract

This end-of-study project presents the design and realization of a mobile application for the follow-up of patients. The Algerian health system is modernizing day by day is in the need to computerize medical information. This application intended for smartphones aims to satisfy the needs of patients. In order to achieve this objective, we were proposed to realize an application under ANDROID by proposing various options and functionalities for example we can contact a doctor, we chose to model our system with the UML formalism to realize the conceptual study of the system, we used the Flutter Framework under the development environment Android Studio, and Firebase the implementation of the database.

Key words: Flutter, Android Studio, XCode, iOS, Dart, Firebase, Medical file

الملخص

هذا المشروع نهاية الدراسة يقدم تصميم وتنفيذ تطبيق محمول لمتابعة المرضى. يقوم النظام الصحي الجزائري بالتحديث يوماً بعد يوم ويحتاج إلى تحويل المعلومات الطبية إلى شكل إلكتروني. هذا التطبيق المخصص للهواتف الذكية يهدف إلى تلبية احتياجات المرضى. ولتحقيق هذا الهدف، تمت مقترحات لنا بتنفيذ تطبيق تحت ANDROID. من خلال اقتراح مجموعة من الخبرات والوظائف مثل النواصل مع الطبيب. قررنا نمذجة نظامنا باستخدام الفورمالية UML لتحقيق الدراسة المفهومية للنظام قمنا باستخدام android Studio ضمن بيئة تطوير Flutter ولتنفيذ قاعدة البيانات FireBase.

كلمات ANDROID، ملف طبي، Flutter، Android Studio، XCode، Dart، Firebase

مفتاحية:

TABLE DES MATIARES :

Remerciement

Dédicace

Introduction Général

Chapitre 01 Généralité sur Android et les applications mobiles.....1

Introduction :1

I.1. Les applications mobiles.....1

I.1.1. Historique1

I.1.2. Définition2

I.1.3. Avantages2

I.1.4. Inconvénients.....3

I.1.5. Les types d'application mobiles3

I.2. Système d'exploitation6

I.2.1. Définition6

I.2.2. Le différent système.....6

I.3. Système d'exploitation Android.....7

I.3.1. Historique9

I.3.2. Caractéristique10

I.3.3. Architecture d'Android.....10

I.3.4. Pourquoi l'Android11

I.3.4.1. Le cout.....11

I.3.4.2. Services Google11

I.3.4.3. Ouverture11

I.3.4.4. Multitâche11

I.4. Généralité sur UML.....12

I.4.1. La modélisation12

I.4.2. Historique UML.....12

I.4.3. Définition13

I.4.4. Diagramme d'UML13

I.4.5. Caractéristiques d'UML :.....14

I.4.6. Pourquoi le langage UML :.....14

I.5. CONCLUSION.....	15
Chapitre 02 analyses et conception.....	16
II.1. Introduction	17
II.2. Analyse	17
II.2.1. Objectif de l'application :.....	17
II.2.2. Présentation de l'application.....	18
II.2.3. Définition de problématique.....	18
II.2.4. Définition de solution	18
II.2.5. Définition des besoins	19
II.2.6. Identification des acteurs.....	20
II.2.7. Présentation des acteurs.....	20
II.3. Conception	22
II.3.1. Diagramme de cas d'utilisation.....	22
II.3.1.1. Définition de gramme de cas d'utilisation.....	22
II.3.1.2. Diagramme de cas d'utilisation en général.....	23
II.3.1.3. Description textuelle des cas d'utilisation.....	25
II.3.2. Digramme de séquence :.....	30
II.3.2.1. Définition de gramme de cas d'utilisation.....	30
II.3.2.2. Quelque diagramme de séquence.....	30
II.3.2.2.1. Diagramme de séquence « S'inscription ».....	30
II.3.2.2.2. Diagramme de séquence « Authentification ».....	31
II.3.2.2.3. Diagramme de séquence « Recherche une médecine ».....	32
II.3.2.2.4. Diagramme de séquence « prendre un Rendez-vous ».....	33
II.3.2.2.5. Diagramme de séquence « contacter une médecine ».....	34
II.3.2.2.6. Diagramme de séquence « Consulter les RDV ».....	35
II.3.3. Diagramme de classe	36
II.4. Conclusion	36
Chapitre 03 Réalisation.....	37
III.1. Introduction.....	37
III.2. Langage et environnement de programmation	37
III.2.1. Environnement de travail	38
III.2.2. Environnement logiciel :.....	39

III.3.	La charte graphique de TABIBI	44
III.4.	Présentation de quelque interface	45
III.4.1.	L'interface d'écran de démarrage de l'application mobile:.....	45
III.4.2.	Interface d'inscription	46
III.4.3.	Interface d'authentification :.....	47
III.4.4.	Interface d'accueil	48
III.4.5.	Interface Profile	49
III.4.6.	Interface Chercher Médecin.....	50
III.4.7.	Interface profil du médecin	51
III.4.8.	Interface réservation de rendez-vous	52
III.4.9.	Interface contacter un médecin	53
III.5.	Conclusion	54
	Conclusion générale.....	55

Liste des tableaux :

TABLEU 2-1 TABLEAU DES ACTEURS ET DES ACTIONS	22
TABLEU 2-2 -DESCRIPTION TEXTUELLE DU CAS D'UTILISATION : « S'INSCRIPTION »	25
TABLEU 2-3 DESCRIPTION DU CAS D'UTILISATION « S'AUTHENTIFICATION »	27
TABLEU 2-4 -DESCRIPTION TEXTUELLE DU DIAGRAMME : « RECHERCHER UN MEDECIN »	27
TABLEU 2-5 -DESCRIPTION TEXTUELLE DU DIAGRAMME : « PRENDRE UN RENDEZ-VOUS »	28
TABLEU 2-6 -DESCRIPTION TEXTUELLE DU DIAGRAMME : « CONTACTER UN MEDECIN»	29
TABLEU 2-7-DESCRIPTION TEXTUELLE DU DIAGRAMME : « CONSULTER LA BOITE DE RECEPTION»	30

Liste des figures :

Chapitre 1 : Généralités sur Android et Application mobile

FIGURE 1.1: TYPES DES APPLICATIONS MOBILES	4
FIGURE 1.2:L'ICONE REPRESENTANT LE SYSTEME IOS.	7
FIGURE 1.3: ARCHITECTURE WINDOWS PHONE	7
FIGURE1.4 :EXEMPLAIRE D'UN TELEPHONE UTILISANT BLACKBERRY OS.....	8
FIGURE1 :5L'ICONE REPRESENTANT LE SYSTEME ANDROID.....	9
FIGURE 1:6- L'ARCHITECTURE D'ANDROID.....	10
FIGURE 1:7HISTOIRE D'UML.	12
FIGURE1.8 L'ABREVIATION UML	13

Chapitre 2 : Analyse at conception

FIGURE2:1: DIAGRAMME DE CAS D'UTILISATION EN GENERALE.	23
FIGURE 2:2 DIAGRAMME DE SEQUENCE « INSCRIPTION ».....	31
FIGURE 2:3 DIAGRAMME DE SEQUENCE « AUTHENTIFICATION ».....	32
FIGURE 2:4 : DIAGRAMME DE SEQUENCE « RECHERCHE UN MEDECINE »	33
FIGURE 2:5 DIAGRAMME DE SEQUENCE « RECHERCHE UN RENDEZ-VOUS » ...	34
FIGURE 2:6 DIAGRAMME DE SEQUENCE « CONTACTER UN MEDECINE ».....	34
FIGURE 2:7 DIAGRAMME DE SEQUENCE « CONSULTER LES RDV »	35
FIGURE 2:8 DIAGRAMME DE CLASSE EN GENERALE.....	36

Chapitre 3: Réalisation

FIGURE3:1 LOGO D'ANDROID STUDIO.....	41
FIGURE 3 : 2 LOGO DE VISUAL STUDIO CODE.	41
FIGURE 3:3 LOGO DE FIREBASE	42
FIGURE 3: 4 LOGO DE STARUML.	42
FIGURE3:5 LOGO DE JDK	43
FIGURE 3:6 LOGO DE SDK	43
FIGURE 3 :7 PRESENTATION DU LOGO DE L'APPLICATION	44
FIGURE 3:8 PRESENTATION DE L'INTERFACE «DEMARRAGE».....	45
FIGURE3:9 PRESENTATION DE L'INTERFACE «INSCRIPTION».....	46
FIGURE3:10 PRESENTATION DE L'INTERFACE « AUTHENTIFICATION »	47

FIGURE3:11	PRESENTATION DE L'INTERFACE « ACCUEIL »	48
FIGURE3:12	PRESENTATION DE L'INTERFACE « PARAMETRES »	49
FIGURE3:13	: PRESENTATION DE L'INTERFACE « CHERCHER MEDECIN »	50
FIGURE3:14	PRESENTATION DE L'INTERFACE «PRENDRE UN RENDEZ-VOUS »	ERREUR ! SIGNET NON DEFINI.
FIGURE3:15	PRESENTATION DE L'INTERFACE « PROFILE MEDECIN »	ERREUR !
	SIGNET NON DEFINI	55

Introduction Générale

Introduction Générale :

Dans un contexte où les technologies mobiles prennent une place de plus en plus prépondérante dans nos vies quotidiennes, le secteur de la santé n'échappe pas à cette révolution numérique. Les applications mobiles médicales jouent un rôle crucial dans l'amélioration des services de santé, en offrant de nouveaux moyens d'interaction entre les professionnels de santé et les patients. C'est dans ce contexte en constante évolution que s'inscrit notre démarche de recherche et de développement d'une application mobile orientée vers la relation entre médecins et patients.

Le développement d'une application mobile médicale soulève des enjeux spécifiques, tant du point de vue de la technologie utilisée que des besoins des utilisateurs finaux. Ainsi, ce mémoire se propose d'explorer ces différentes dimensions à travers une approche méthodologique structurée et centrée sur l'expérience utilisateur.

L'objectif de notre travail est de faciliter la tâche au patient et au médecin en cette période de circulation active d'un virus, tout en permettant un gain de temps et de distance. Pour cela, nous allons développer une application mobile nommée "TABIBI".

Pour mener à bien notre travail, nous avons opté pour une approche en trois chapitres :

Le premier chapitre, intitulé "Généralités", est organisé en deux parties : la première aborde les concepts généraux concernant les applications mobiles, en commençant par leur définition, leur historique et les différents types d'applications, tout en examinant les avantages et les inconvénients qui leur sont

associés. Nous discutons également des différents systèmes d'exploitation tels qu'iOS, Windows Phone, BlackBerry OS, ainsi qu'une exploration approfondie du système Android. La deuxième partie traite des concepts généraux relatifs à UML.

Dans le deuxième chapitre, intitulé "Analyse et Conception", nous évoquons les objectifs à atteindre, posons la problématique, étudions l'existant et proposons une solution à développer dans le futur système. Nous identifions les acteurs qui interagissent avec le futur système à développer, ainsi que le diagramme de cas d'utilisation correspondant de manière générale. Ce cas d'utilisation est ensuite détaillé à l'aide des diagrammes de séquences, ce qui nous permet d'établir le diagramme de classe.

Le troisième chapitre est consacré à la phase de "Réalisation". Nous présentons l'environnement de développement de l'application mobile, mettons en évidence les langages et outils utilisés, et exposons les interfaces principales de l'application. Nous concluons notre travail par une conclusion générale illustrant la perspective d'amélioration de l'application.

CHAPITRE 01 :
Généralités sur Android et Applications
mobiles

Introduction :

Aujourd'hui, les Smartphones et les autres terminaux mobiles sont devenus indispensables pour faciliter notre vie quotidienne. La croissance de l'utilisation de ces appareils mobiles et la progression continue de ces derniers nécessitent aussi une évolution au niveau des systèmes d'exploitation mobile et des applications mobiles.

Ce chapitre présente, les applications mobiles, les systèmes d'exploitation mobile et surtout le système d'exploitation Android, et Généralité sur UML.

I.1. Les applications mobiles

I.1.1. Historique :

Les applications mobiles sont apparues dans les années 1990 elles sont liées aux développements d'Internet et des télécommunications des réseaux sans fils et des technologies agents et à l'apparition et la démocratisation des terminaux mobiles : smartphones, tablettes tactiles... Le nombre d'applications mobiles disponible augmente rapidement à partir de 2007 date du lancement de l'iPhone. Les deux principaux services de plateformes de téléchargement l'App Store et le Google Play sont lancés en 2008[1]

Les smartphones ont transformé la manière dont nous interagissons avec le monde numérique, rendant les informations et services accessibles à portée de main. Les tablettes tactiles, offrant un écran plus grand, ont élargi les possibilités d'utilisation pour des tâches variées, allant de la lecture numérique à la productivité professionnelle.

Ainsi, les innovations technologiques des années 1990 ont jeté les bases d'un écosystème d'applications mobiles en constante évolution, améliorant notre quotidien de manière significative. [1]

I.1.2.Définition :

Une application mobile est un programme téléchargeable de façon gratuite ou payante et exécutable à partir du système d'exploitation d'un dispositif mobile (PDA, Smartphone, tablette...), Elle permet d'accéder à des services pratiques des informations multimédias ou de différentes sources de divertissement. Les applications mobiles développent des usages dans tous les domaines : réseaux sociaux, médical, actualité, pharmaceutique, jeux, agenda, opérations bancaires, GPS, boîte mail, visualisation de chaînes de télévision, écoute de la radio ou musique, météo, faire du shopping Elles sont pour la plupart distribuées depuis des plateformes de téléchargement telles que l'App Store (plateforme d'Apple), le Google Play (plateforme de Google / Androïde), ou encore le Windows Phone Store (plateforme de Microsoft). [2]

Chez les mobinautes, l'application mobile est similaire à un site internet pointu en raison de sa connexion à internet, de plus l'interface du site et de l'application mobile s'avèrent identiques sauf que l'application demeure fondamentalement défini comme un logiciel. En ce sens, les applications mobiles se regroupent en plusieurs séries suivant des critères basiques. [3]

I.1.3.Avantages

Une application mobile revêt de nombreux avantages comparés à un site web mobile, en voici 5 principaux :

- Un confort d'usage et une expérience utilisateur inégalée.
- L'accès direct aux contenus de l'application mobile via l'icône présent sur le Dashboard du téléphone ou de la tablette (mode d'accès sans URL).
- Un fonctionnement en mode déconnecté.

- Elle permet d'utiliser et d'intégrer toutes les fonctionnalités téléphone (accéléromètre, gyroscope, GPS, caméra...), ce qui n'est pas forcément le cas des Web Apps.
- L'implémentation de fonctionnalités natives comme par exemple les notifications « PUSH ».

I.1.4. Inconvénients

Le principal inconvénient d'une application mobile réside dans l'obligation de se conformer aux règles des différentes plateformes mobiles. Cela inclut l'approbation nécessaire des App Stores pour diffuser l'application ou ses mises à jour, ainsi que les conditions tarifaires imposées. De plus, une application mobile peut ne pas être compatible avec tous les systèmes d'exploitation mobiles.

Le coût de développement est également un obstacle majeur, car il est généralement plus élevé pour une application mobile déployée sur plusieurs plateformes comparé à un site mobile ou une Web App. Cela nécessite souvent un développement spécifique pour chaque technologie, entraînant des coûts supplémentaires pour assurer la disponibilité sur tous les modèles de dispositifs.

Enfin, pour que l'utilisateur accède à la dernière version de l'application, il doit la mettre à jour depuis le store, contrairement aux sites mobiles et aux Web Apps qui se mettent à jour automatiquement. [4]

I.1.5. Les types d'application 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.



Figure1.1: Types des applications mobiles

1. Les applications Web (connectées) :

L'application Web est un logiciel applicatif hébergé sur un serveur et accessible à travers un navigateur web. Cependant contrairement à un logiciel traditionnel les utilisateurs n'ont pas la nécessité d'installer le logiciel. La connexion avec l'application se fait via un navigateur internet comme si c'était un site web.

Les avancées technologiques ont permis l'apparition des progressive Web App (PWA). Une Progressive Web App est une Web App ayant intégré des fonctionnalités grâce à des « Service Worker » qui était jusqu'à présent réservé aux applications natives.

Les avantages d'une PWA sont divers. D'une part elle permet d'avoir des frais de développement moins élevés qu'une application native ou hybride car les langages de programmation sont plus courants (HTML, JAVA, PHP, REACT, Angular ...). Son développement repose sur la segmentation et l'association des briques d'une application au travers d'API cette conception permet de prendre le meilleur de chaque technologie utilisée dans l'architecture métier afin d'obtenir un parfait compromis entre une application Native/mobile et un site web.

D'autre part son accessibilité depuis un navigateur lui donne également l'avantage d'être cross-plateforme c'est-à-dire accessible depuis n'importe quel device : Desktop, tablette et mobile. Les ressentis avancés permettent aux PWA d'utiliser des fonctionnalités natives comme l'utilisation Offline, GPS à la demande, géolocalisation, notifications push et la gestion du cache, appareil photo, le micro, la boussole.

2. Les applications natives (non connectées) :

Une application native est une application mobile qui fait l'objet de développements distinct pour chacun des systèmes d'exploitation utilisés (iOS, Android, etc.), et ainsi elle fonctionne de façon optimale avec tous les types d'appareils. Elle a un développement plus

complexe que les autres car les langages de programmation sont différents selon les systèmes d'exploitation. Par exemple, les applications pour l'Apple Store sont développées soit en Objective-C ou en Swift mais les applications pour Google Play sont développées en Java et Kotlin. Les applications Windows sont développées en C#. [5]

La différence est significative car ces divers langages feront appel à ce que l'on appelle les couches de bases de l'appareil comme l'appareil photo ou encore l'accéléromètre.

3. Les applications hybrides :

Entre l'application native et l'application web on trouve l'application « cross-platform » Hybride. Comparées aux applications natives les applications hybrides sont moins chères et plus rapides à développer.

Une application Hybride a une conception plus simple et rapide que les applications natives car seulement une version du code est déployée pour les différents systèmes d'exploitation (Apple, Android, Windows). Lors du développement de l'application un Software Development Kit (SDK) est utilisé pour avoir accès à une librairie d'ensemble de briques préconstruites adaptées aux différents systèmes d'exploitation et permettant la transposition en code binaire (langage natif).

Les développements en « cross-platform » ont l'avantage de réduire la maintenance et de faciliter l'ajout de fonctionnalités car il n'y a qu'une seule version du code.

Dans les applications « cross-platform » on trouve les applications « hybrides » et les applications « natives générées ».

Une application « natives générées » est une application développée à partir un langage exclusion cross-platform » type C#, Ruby, JavaScript, React Native les SDK sont plus utilisé sont Xamarin et Titanium.

Une application « hybride » est une application un contenu web est alors « encapsulé » avec une interface mobile on y retrouve à la différence de la « native générée » du code HTML. Pour ce faire, des SDK telles qu'Apache Cordova (le plus courant), PhoneGap, Rho Mobile sont utilisés [6]

I.2. Système d'Exploitation Mobile :

I.2.1.Définition :

Un système d'exploitation mobile (Mobile Operating System) est un logiciel conçu pour gérer et contrôler le fonctionnement des applications sur un appareil. Il décide quelles applications peuvent fonctionner sur l'appareil et peut être défini comme un ensemble de programmes responsables de la gestion des opérations, de la mémoire et du WAP, ainsi que de la coordination et de la synchronisation des applications, et de la communication avec le matériel. [7].

I.2.2.Les différents systèmes :

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

- **IPhone (IOS) :**

Est le système d'exploitation mobile développé par Apple pour l'iPhone. Reconnu pour sa fluidité et son inactivité ; C'est le système le plus abouti à ce jour.

IOS se caractérise par une interface peu chargée, qui permet l'exécution de tâches rapidement pour les utilisateurs réguliers, mais qui laisse peu de place à la personnalisation. IOS est dérivé de Mac OS X et donc d'Unix.



Figure 1: 2:L'icône représentant le système iOS.

- **Windows Phone : Une Alternative sur le Marché des Smartphones**

De Microsoft, initialement lancé en 2003 - comme le successeur de Windows mobile - pour ses smartphones et tablettes, puis il a été utilisé par des biens connus marque comme dans des très puissants dispositifs tels que Samsung, LG et HTC.

Windows Phone a créé son propre application store nommé Windows Phone Marketplace

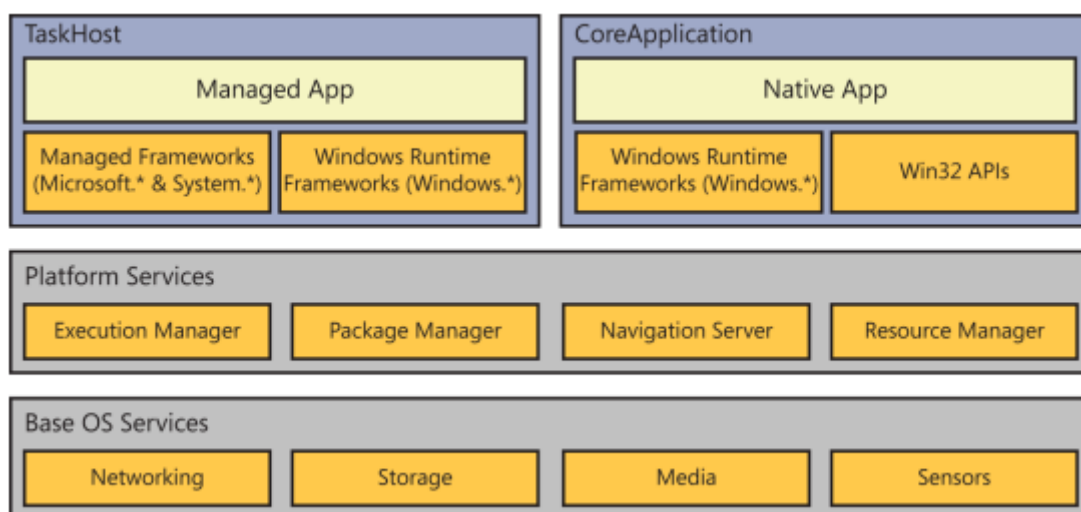


Figure 1: 3 Architecture Windows Phone

- **BlackBerry OS :**

Le système développé par le fabricant canadien RIM (Research In Motion) a été pionnier dans l'introduction de la notification instantanée d'e-mails en mode push. Il améliore également l'expérience mobile en comprimant les pages web et les pièces jointes des e-mails. Ainsi, la fonctionnalité e-mail est devenue le principal atout des BlackBerry, ce qui les a rendus très populaires auprès des cadres et des dirigeants. [8]



Figure 1:4 Exemple d'un téléphone utilisant BlackBerry OS.

- **Android :**

Basé sur le noyau Linux, Android est un système d'exploitation développé par Google et l'Open Handset Alliance. Il est aujourd'hui présent sur une large gamme d'appareils, allant des smartphones aux tablettes en passant par les montres digitales et les autoradios. Android est gratuit pour les fabricants d'appareils qui souhaitent l'adopter et partiellement open source. Il intègre plusieurs services de Google, offrant ainsi un accès rapide à des services Internet tels que Gmail, YouTube, Google Talk, Google Calendre et Google Maps. [9]



Figure1: 5L'icône représentant le système Android.

I.3. Système d'Exploitation Android

I.3.1.Historique :

En juillet 2005, Google fait l'acquisition d'une petite startup nommée Android Inc., spécialisée dans le développement d'applications pour téléphones mobiles. Suite à cette acquisition, des spéculations abondent quant à l'entrée de Google sur le marché des mobiles, mais aucune information concrète n'émerge concernant ses plans ou ses produits.

Après cette acquisition par Google, une équipe dirigée par Andy Rubin, l'un des fondateurs d'Android Inc., se lance dans le développement d'un système d'exploitation open source pour les appareils mobiles, basé sur Linux. Pendant cette période, de nombreuses rumeurs circulent sur les intentions de Google, y compris le développement d'applications de recherche pour les mobiles et la conception d'un nouveau téléphone portable.

Le 5 novembre 2007, l'Open Handset Alliance (OHA) est officiellement établie dans le but de définir des normes open source pour les appareils mobiles. Le premier standard de l'OHA est révélé, il s'agit d'Android, une plateforme open source pour les appareils mobiles basée sur le noyau Linux 2.6 [10].

I.3.2. Caractéristique :

Android est une pile applicative pour les appareils mobiles qui comprend un système d'exploitation, middleware et des applications clés. Le SDK Android fournit les outils et les API nécessaires pour commencer à développer des applications sur la plate-forme Android en utilisant le langage de programmation Java

I.3.3. Architecture d'Android :

Le diagramme suivant illustre les composants principaux du système d'exploitation Android.:

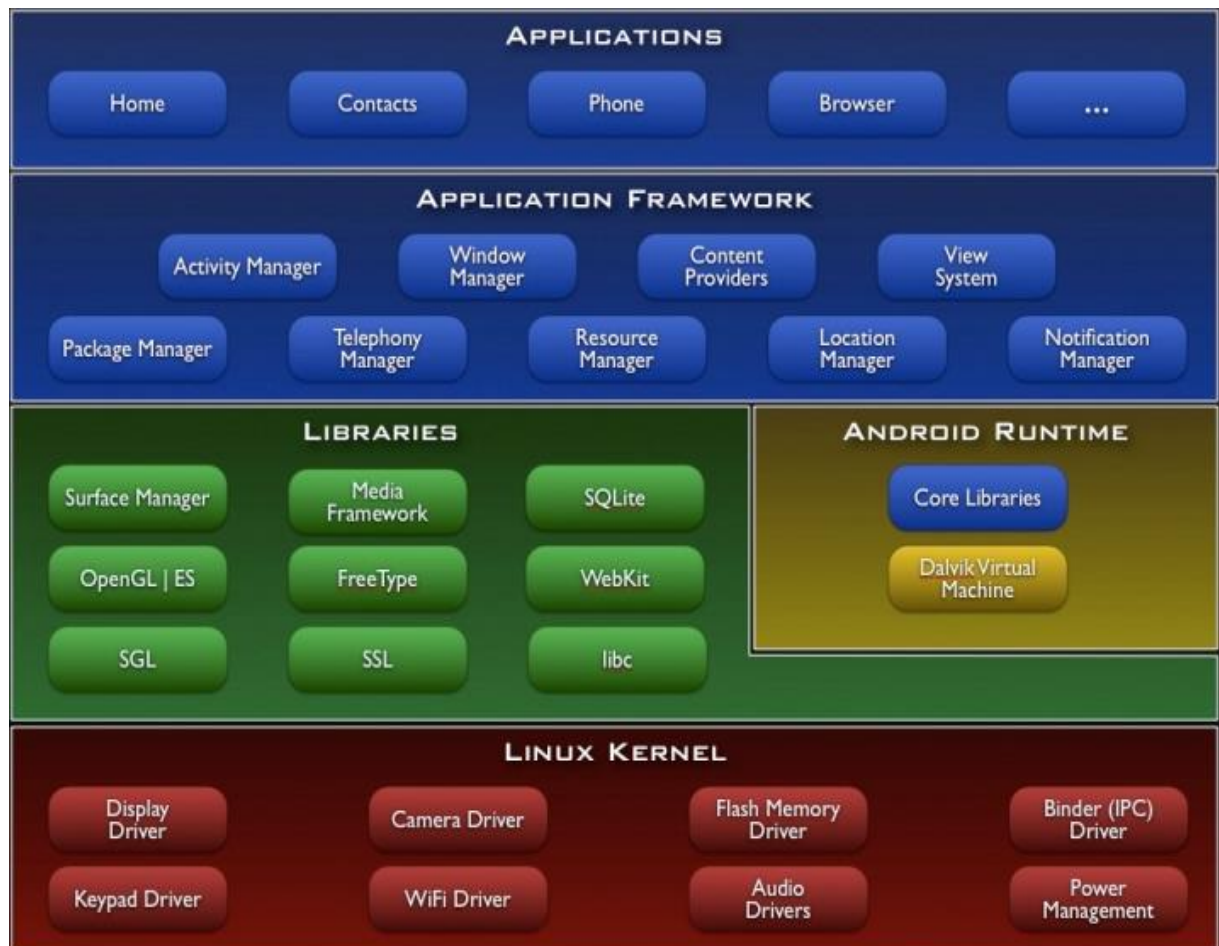


Figure 1: 6- L'architecture d'Android

I.3.4. Pourquoi l'Android :

Dans notre projet, nous avons choisi d'utiliser la plate-forme Android pour la réalisation, Ce choix n'était pas au hasard, mais il y a des critères nous prenez-les en considération, citons :

I.3.4.1. Le coût :

De côté de développement, tout outil nécessaire pour réaliser un projet avec Android mis à disposition gratuitement par Google. Et d'autre côté, il permet aux entreprises de choisir parmi un vaste éventail d'appareils, tel qu'appareils haute de gamme comme le Samsung S10, et de l'autre plus économique tel que Huawei P20

I.3.4.2. Services Google :

Android est développé par Google, Ce qui signifie que Android fonctionne d'une façon très efficace tous services de le dernier, et facilement laisse aux développeurs de gérer.

I.3.4.3. Ouverture :

Cela très orienté vers les développeurs, toutes version Android respectent le principe d'open-source cette caractéristique permet aux développeurs d'accès et de manipuler les APIs et ses différemment composants selon les exigences du projet et les sources financières.

I.3.4.4. Multitâche :

Une fonctionnalité très appréciée par les utilisateurs finaux. Android offre aux ses utilisateurs la capacité d'effectuer plusieurs tâches, ouvrir plusieurs applications à la fois, et de gérer facilement grâce à son excellente interface. [11]

I.4. Généralité sur UML

I.4.1. La modélisation :

La méthodologie est une approche organisée de façon rationnelle afin d'atteindre un résultat et de progresser efficacement dans le projet en réalisant simultanément des tâches de conception et de mise en œuvre

I.4.2. Histoire d'UML :

UML, fruit de la convergence de trois méthodes d'analyse orientées objet (la méthode OOD, la méthode OMT et la méthode OOSE), a vu le jour grâce à l'initiative de Rational Software. Les principaux auteurs de ces méthodes ont convenu de créer un langage de modélisation unifié, qui est devenu une norme après son adoption par l'Object Management Group (OMG) en 1997. Initialement conçu comme un ensemble de diagrammes destinés aux développeurs adoptant une approche orientée objet, UML a évolué vers sa version UML2 en 2004. Cette évolution a permis son utilisation pour décrire des systèmes d'information, notamment dans le cadre de l'élaboration des spécifications. [12]



Figure 1: Histoire d'UML.

I.4.3. Définition :

UML, acronyme pour « Unified Modeling Language », est un langage visuel de modélisation, à la fois graphique et textuel. Il est employé pour la compréhension, la spécification, la documentation et la conception de systèmes, ainsi que pour l'esquisse d'architectures logicielles. Il permet également de communiquer les différentes composantes d'un système d'information. Bien que son origine soit liée au développement logiciel, ce langage peut être adapté à d'autres disciplines nécessitant la description d'un système. [13]

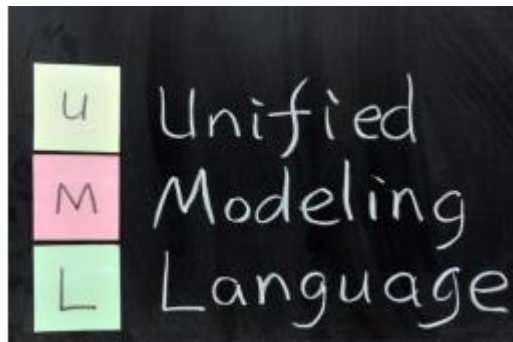


Figure 1: 8 L'abréviation UML

I.4.4. Diagrammes d'UML :

Les diagrammes d'UML représentent les traitements et les données modélisés sous forme de graphe et chacun d'eux représente un concept particulier d'un système logiciel.

- **Diagrammes Statistiques :** Représentent l'aspect statistique d'un logiciel
 - ✓ Diagramme de classe.
 - ✓ Diagramme de package.
 - ✓ Diagramme d'objet.
 - ✓ Diagramme de composants.
 - ✓ Diagramme de déploiement.

- **Diagrammes dynamique** : Représentent l'aspect dynamique du logiciel, ainsi que le comportement et les interactions.
 - ✓ Diagramme d'activité.
 - ✓ Diagramme de séquence.
 - ✓ Diagramme d'état de transition.
 - ✓ Diagramme de communication.
 - ✓ Diagramme de temps.
- **Diagrammes fonctionnels** : Représentent le fonctionnement d'un système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit.
 - ✓ Diagramme de cas d'utilisation.
 - ✓ Diagramme de collaboration.

I.4.5. Caractéristiques d'UML :

Parmi les caractéristiques attrayantes en UML nous pouvons citer :

- ✓ UML est une notation, pas une méthode.
- ✓ UML a été adopté par toutes les méthodes objet.
- ✓ UML est dans le domaine public, c'est une Norme.

I.4.6. Pourquoi le langage UML :

Il vaut mieux modéliser un système avant de le réaliser car la modélisation permet de mieux comprendre le fonctionnement du système. Plusieurs raisons conduisent à préconiser l'utilisation d'UML pour définir un système d'information car UML nous permet de :

- Obtenir une modélisation de très haut niveau indépendante des langages et des environnements.
- Faire collaborer des participants de tout horizon autour d'un même document de synthèse.

- Faire des simulations avant de construire un système.
- Exprimer dans un seul modèle tous les aspects statiques, dynamiques, juridiques, spécifications, etc.
- Documenter un projet.
- Générer automatiquement la partie logicielle d'un système [14]

I.5. Conclusion :

Dans ce chapitre, nous avons présenté en grosso modo des généralités sur les applications mobiles et les avantages et les inconvénients de ces différents types, nous avons aussi défini et comparé entre les systèmes d'exploitation mobile, décrivant aussi le système Android, son architecture, ensuite nous avons présenté une idée générale sur la technologie de UML. Le chapitre suivant sera consacré à la phase de l'analyse et la conception du Système d'informatique.

CHAPITR 2 :
ANALYSE ET
CONCEPTION

2.1. Introduction :

L'analyse et la conception du Système d'informatique est une étape très importante dans le cycle de vie d'une application, dans ce chapitre on va faire une phase d'analyse, suivi d'une étape de conception.

Dans la phase d'analyse, on cherche d'abord à bien comprendre et à décrire de façon précise les besoins d'utilisateur. L'objectif de l'application ; présentation de l'application, c'est ce qu'on appelle « l'analyse de besoins ». Après validation de notre compréhension du besoin, on passe pour présenter le processus simplifié qui est la méthode qu'on va suivre pour le développement de notre application. En fin, on applique cette méthode pour faire la conception en utilisant les diagrammes UML.

2.2. Analyse :

2.2.1. Objectif de l'application :

Nous envisageons de développer une application visant à améliorer la relation entre les professionnels de la santé et les patients, dans un domaine particulièrement sensible où les applications mobiles sont devenues un moyen essentiel d'accès aux services clients. Cette application comprendra plusieurs fonctionnalités clés, telles que la facilité de recherche de médecins spécialisés sans quitter le domicile, la possibilité de prendre rendez-vous à distance en quelques clics et de communiquer directement avec les médecins. Tout en conservant un historique complet des activités et des maladies du patient. De plus, l'application proposera un suivi médical par des spécialistes qui fourniront des conseils adaptés à l'état de santé du patient, contribuant ainsi à améliorer la santé publique et le bien-être des patients.

2.2.2. Présentation de l'application :

Notre mission dans le cadre de ce projet est de créer une application mobile révolutionnaire dédiée à la gestion efficace de la relation médecin-patient. Grâce à une interface intuitive, elle simplifie la prise de rendez-vous et encourage une communication instantanée pour des soins de santé personnalisés.

Notre application aura comme principales fonctionnalités :

- Gestion de Rendez-vous.
- Gestion de chat direct.

2.2.3. Définition de problématique :

Le domaine médical en Algérie est confronté à divers défis, notamment en ce qui concerne la communication entre les patients et les médecins :

- La difficulté à trouver un médecin spécialisé répondant à leurs besoins.
- Les obstacles rencontrés pour obtenir rapidement et facilement des rendez-vous médicaux.
- Les limitations de la communication entre les médecins et les patients.
- Le manque de suivi des traitements médicaux.
- L'absence d'historique des consultations, ce qui pose souvent problème lors de nouvelles consultations médicales.

2.2.4. Définition de la solution :

Notre projet vise à proposer une solution aux problèmes mentionnés précédemment. Pour ce faire, nous envisageons de développer une application mobile baptisée « TABIBI », offrant plusieurs fonctionnalités :

- Trouver le médecin approprié et spécialisé depuis chez soi, simplement en utilisant son téléphone.
- Prendre des rendez-vous en ligne, ce qui permet d'économiser du temps et de l'effort, sans avoir à contacter le cabinet médical ou à se déplacer.
- Faciliter la communication sécurisée entre les médecins et les patients.

2.2.5. Définition des besoins :

Cette étape constitue un élément crucial en amont de tout processus de développement. Son objectif principal est de garantir la conception d'un logiciel approprié, en décrivant de manière générale les fonctionnalités du système et en définissant les objectifs à atteindre en répondant à la question : quelles sont les fonctions du système ?

Dans cette section du chapitre, nous examinons les besoins des utilisateurs, à la fois fonctionnels et non fonctionnels, dans le cadre de notre projet, afin de développer une application de qualité répondant aux exigences suivantes :

2.2.5.1. Besoins fonctionnelle :

- ❖ Enregistrement et connexion pour les professionnels de la santé et les patients.
- ❖ Amélioration de l'efficacité de la recherche de professionnels médicaux.
- ❖ Planification de rendez-vous en ligne.
- ❖ Minimisation du temps d'attente pour les consultations.
- ❖ Simplification des échanges entre les acteurs.
- ❖ Mise en place d'une messagerie sécurisée avec les médecins.
- ❖ Fluidité des téléconsultations.

2.2.5.2. Besoin non fonctionnelle :

- ✓ Fiabilité : l'application doit fonctionner de façon cohérente sans erreurs et doit être satisfaisante

- ✓ Facilité d'utilisation : l'application doit être adaptée à l'utilisateur sans qu'il ne fournisse aucun effort (utilisation claire et facile).
- ✓ Sécurité :
- ✓ Compatibilité et portabilité :
- ✓ Performance.

2.2.6. Identification des Acteurs :

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

2.2.7. Présentation des Acteurs :

- **Administrateur** : le seul acteur ayant le droit d'accès, la mise à jour de l'application, gérer les médecins et les patients
- **Patient** : c'est un utilisateur ayant déjà créé un compte sur notre application en tant que patient, ils peuvent chercher une médecine pour une consultation ou pour avoir une idée sur les prix.
- **Médecin** : c'est un utilisateur ayant déjà créé un compte sur notre application en tant que médecin.

Dans le tableau ci-dessous on présentera nos trois acteurs principaux cadre que nous avons distingué

Acteur	Action
--------	--------

Administrateur	A1 Authentification
	A2 Gestion Médecin <ul style="list-style-type: none"> ➤ Confirmer un Médecin ➤ Supprimer un Médecin
	A3 Gestion patient <ul style="list-style-type: none"> ➤ Confirmer un Patient ➤ Supprimer un patient
	A4 Gestion de spécialité <ul style="list-style-type: none"> ➤ Ajouter une spécialité ➤ Modifier une spécialité
Patient	A5 S'authentifier
	A5 modifier profile
	A6 rechercher un médecin
	A7 prendre RDV
	A8 Consulter le profile médecin
	A9 Contacter un médecin
	A10 Consulter mes RDV
Médecin	A11 S'authentifier
	A12 Modifier profile
	A13 Contacter un Patient

	A14 Gestion des RDV <ul style="list-style-type: none"> ❖ Consulter les RDV. ❖ Confirmer les RDV. ❖ Refuser une RDV.
	A15 Consulter mes patients

Tableau 2 :1 tableau des acteurs et des actions

2.3. Conception :

Pour décrire la conception de notre application, nous commençons par les diagrammes de cas d'utilisation, puis les diagrammes de séquence et à la fin le diagramme de classe

2.3.1. Diagramme de cas d'utilisation :

2.3.1.1. Définition de diagramme de cas d'utilisation :

Un cas d'utilisation est utilisé pour définir le comportement d'un système ou la sémantique de toute autre entité sans révéler sa structure interne. Chaque cas d'utilisation spécifie une séquence d'actions que l'entité réalise, en interagissant avec les acteurs de l'entité.

2.3.1.2. Diagramme de cas d'utilisation en générale :

Nous présentons le diagramme de cas d'utilisation suivant pour la compréhension du fonctionnement général du système.

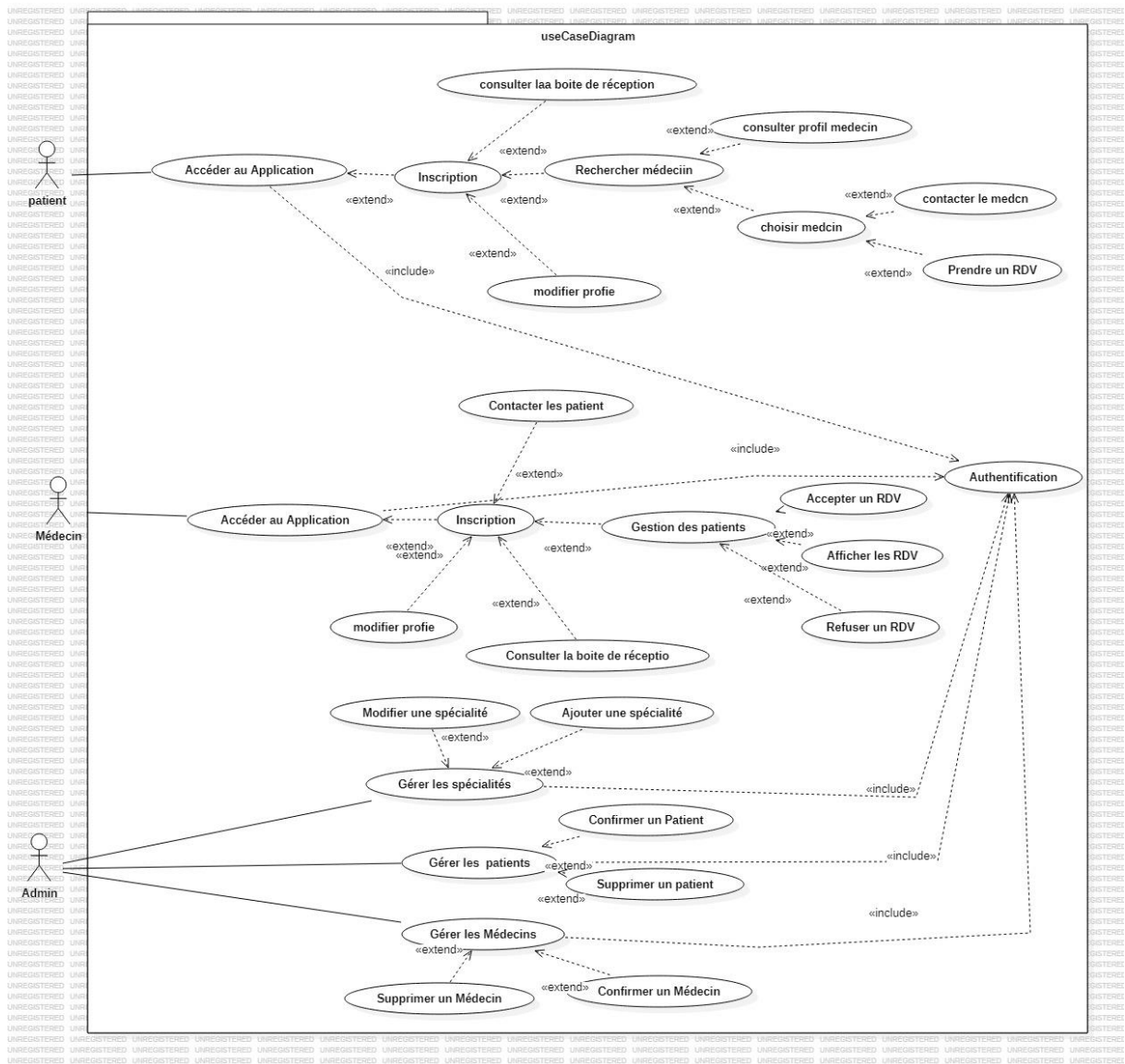


Figure2: 1: diagramme de cas d'utilisation en générale.

2.3.1.3. Description textuelle des cas d'utilisation

Bien que de nombreux diagrammes d'UML permettent de décrire un cas, il est recommandé de rédiger une description textuelle car c'est une forme souple qui convient dans bien des situations. Une description textuelle couramment utilisée se compose de deux parties :

- ✓ La première partie permet d'identifier le cas. Elle doit contenir :
 - Le nom du cas.
 - Un résumé de son objectif.
 - Les acteurs impliqués (principaux et secondaires).

- ✓ La deuxième partie contient la description du fonctionnement du cas sous la forme d'une séquence de messages échangés entre les acteurs et le système. Elle contient une séquence nominale qui correspond au fonctionnement nominal du cas.

Cette séquence se développe en trois points :

- Les préconditions : Elles indiquent dans quel état est le système avant que se déroule la séquence.
- L'enchaînement des messages.
- Les post-conditions : Elles indiquent dans quel état se trouve le système après le déroulement de la séquence nominale.

Comme on peut y avoir :

- Une séquence alternative : diverge de la séquence nominale (c'est un embranchement dans une séquence nominale).
- Une séquence d'exception : intervient quand une erreur se produit (l'enchaînement nominal s'interrompt, sans retour à la séquence nominale).

Nous décrivons les cas suivants :

- S'inscrire
- S'authentifier.
- Rechercher un médecin.
- Prend un rendez-vous.
- Contacter un médecin.

- Consulter la boîte de réception

- **Description textuelle du cas d'utilisation : « S'inscription »**

Description textuelle du cas d'utilisation : « S'inscription »

Identification :

Nom du cas : « s'inscription»

Résumé : Permettre à un utilisateur de s'inscrire

Acteur principal : utilisateur.

Pré-condition

- L'utilisateur doit renseigner le formulaire d'inscription.

Enchaînement nominale

1. L'utilisateur remplir le formulaire d'inscription,
2. Le système vérifie la validation des champs saisiie par l'utilisateur ;
3. Le système crée un compt pour l'utilisateur
4. Le système envoie un message pour confirmer au l'utilisateur que l'inscription est réussie

Enchaînement alternatif

- Les information saisies par l'utilisateur ne sont pas valide.

L'Enchainemen démarre au poit 3 du scénario nominal :

3. Le système indique au l'utilisateur que les informations saisies sont incorrecte ;
4. Le système demande au l'utilisteur de vérifier les informations saisies.

Post-condition :

- L'utilisateur est inscrit dans la base de données
- Un message de confirmation de l'inscription est envoyé au l'utilisateur.

Tableu 2-2 -Description textuelle du cas d'utilisation : « S'inscription »

- **Description textuelle du diagramme : « S' Authentification »**

Description textuelle du cas d'utilisation : « S' Authentification »
<p>Identification :</p> <p>Nom du cas : « S' Authentification »</p> <p>Résumé : Permettre à un patient ou médecin de s' Authentifier afin d'accéder à son compt personnel</p> <p>Acteurs principaux : patient et médecin.</p>
<p>Pré-condition</p> <ul style="list-style-type: none"> - L'utilisateur doit renseigner le formulaire d'authentification . <p>Enchaînement nominale</p> <ol style="list-style-type: none"> 1. Patient/médecin remplir le formulaire d'authentification, 2. Le système vérifie la validation des champs saisiés par le patient et médecin ; 3. Le système traite les données envoyées par l'utilisateur 4. Le système enregistrer les données envoyées par l'utilisateur dans la base de données 5. Le système afficher l'espace de patient/médecine <p>Enchaînement alternatif</p> <ul style="list-style-type: none"> • Les informations saisiés par le patient/médecin ne sont pas valides. <p>L'Enchaînement démarre au point 3 du scénario nominal :</p> <ol style="list-style-type: none"> 3. Le système indique au patient/médecin que les informations saisiés sont incorrectes ; 4. Le système demande au patient/médecin de vérifier les informations saisiés. <p>Post-condition :</p> <ul style="list-style-type: none"> • L'authentification du patient/médecin est réussie

- Le patient/médecin pourra se déconnecter de son compte..

Tableau 2-3 description du cas d'utilisation « S'Authentification »

- Description textuelle du diagramme : « Rechercher un médecin »

Description textuelle du cas d'utilisation : « Rechercher un médecin »
<p>Identification :</p> <p>Nom du cas : « Rechercher un médecin »</p> <p>Résumé : Permettre à un patient de faire la recherche des médecins.</p> <p>Acteur principal : patient.</p>
<p>Pré-condition</p> <ul style="list-style-type: none"> - L'utilisateur doit renseigner le formulaire de recherche . <p>Enchaînement nominale</p> <ol style="list-style-type: none"> 1. Patient remplit le formulaire de recherche, 2. Le système vérifie la validation des champs saisis par le patient . 3. Le système traite les données envoyées par l'utilisateur. 4. Le système répond à la demande du client. <p>Enchaînement alternatif</p> <ul style="list-style-type: none"> • Les informations saisies par le patient/médecin ne sont pas valides. <p>L'Enchaînement démarre au point 3 du scénario nominal :</p> <ol style="list-style-type: none"> 4. Le système indique au patient/médecin que les informations saisies sont incorrectes ; 5. Le système demande au patient/médecin de vérifier les informations saisies. <p>Post-condition :</p> <ul style="list-style-type: none"> • Trouver le médecin ou la liste des médecins recherchés.

Tableau 2 :4 -Description textuelle du diagramme : « Rechercher un médecin »

- Description textuelle du diagramme : « Prendre un rendez-vous »

Description textuelle du cas d'utilisation : « Prendre un rendez-vous »
<p>Identification :</p> <p>Nom du cas : « Prendre un rendez-vous »</p> <p>Résumé : Permettre à un patient de prendre un rendez-vous.</p> <p>Acteur principal : patient.</p>
<p>Pré-condition</p> <ul style="list-style-type: none"> - L'utilisateur doit s'authentifier. - Le patient doit faire la recherche d'un médecin et consulter le calendrier des rendez-vous du médecin. <p>Enchaînement nominale</p> <ol style="list-style-type: none"> 1. Patient prend un rendez-vous, 2. Le système envoie les informations nécessaires à la base de données. 3. La base de données enregistre la demande de patient. <p>Post-condition :</p> <ul style="list-style-type: none"> • Prise de rendez-vous réussie.

Tableau 2 :5 -Description textuelle du diagramme : « Prendre un rendez-vous »

- Description textuelle du diagramme : « Contacter un médecin »

Description textuelle du cas d'utilisation : « Contacter un médecin »
<p>Identification :</p> <p>Nom du cas : « Contacter un médecin »</p> <p>Résumé : Permettre à un patient de Contacter un médecin.</p> <p>Acteur principal : patient.</p>
<p>Pré-condition</p> <ul style="list-style-type: none"> - L'utilisateur doit s'authentifier. - Le patient doit faire la recherche d'un médecin - Le patient doit Contacter un médecin <p>Enchaînement nominale</p> <ol style="list-style-type: none"> 1. Patient saisie un message pour le médecin, 2. Le système vérifie la validation des coordonnées du médecin. 3. Le système enregistre le message . <p>Post-condition :</p> <ul style="list-style-type: none"> • Message envoyé.

Tableau 2 :6 -Description textuelle du diagramme : « Contacter un médecin »

- Description textuelle du diagramme : « Consulter la boîte de réception »

Description textuelle du cas d'utilisation : « Consulter la boîte de réception »
<p>Identification :</p> <p>Nom du cas : « Consulter la boîte de réception »</p> <p>Résumé : Permettre à un patient de Consulter la boîte de réception.</p> <p>Acteur principal : patient.</p>
<p>Pré-condition</p> <ul style="list-style-type: none"> - L'utilisateur doit s'authentifier. - Le patient doit accéder à sa messagerie. <p>Enchaînement nominale</p> <ol style="list-style-type: none"> 1. Patient consulte sa boîte de réception, 2. Le système renvoie les messages envoyés par les patients. 3. Le patient fait la consultation. <p>Post-condition :</p> <ul style="list-style-type: none"> • Consultation réussie.

Tableau 2 :7-Description textuelle du diagramme : « Consulter la boîte de réception »

2.3.2. Diagramme de séquence :

2.3.2.1. Définition de diagramme de séquence :

Un diagramme de séquence est un diagramme UML qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction [16]

2.3.2.2. Quelques diagramme de séquence :

Nous allons présenter ci-après, les diagrammes de séquence de quelques cas d'utilisation :

2.3.2.2.1. Diagramme de séquence « S’inscription »

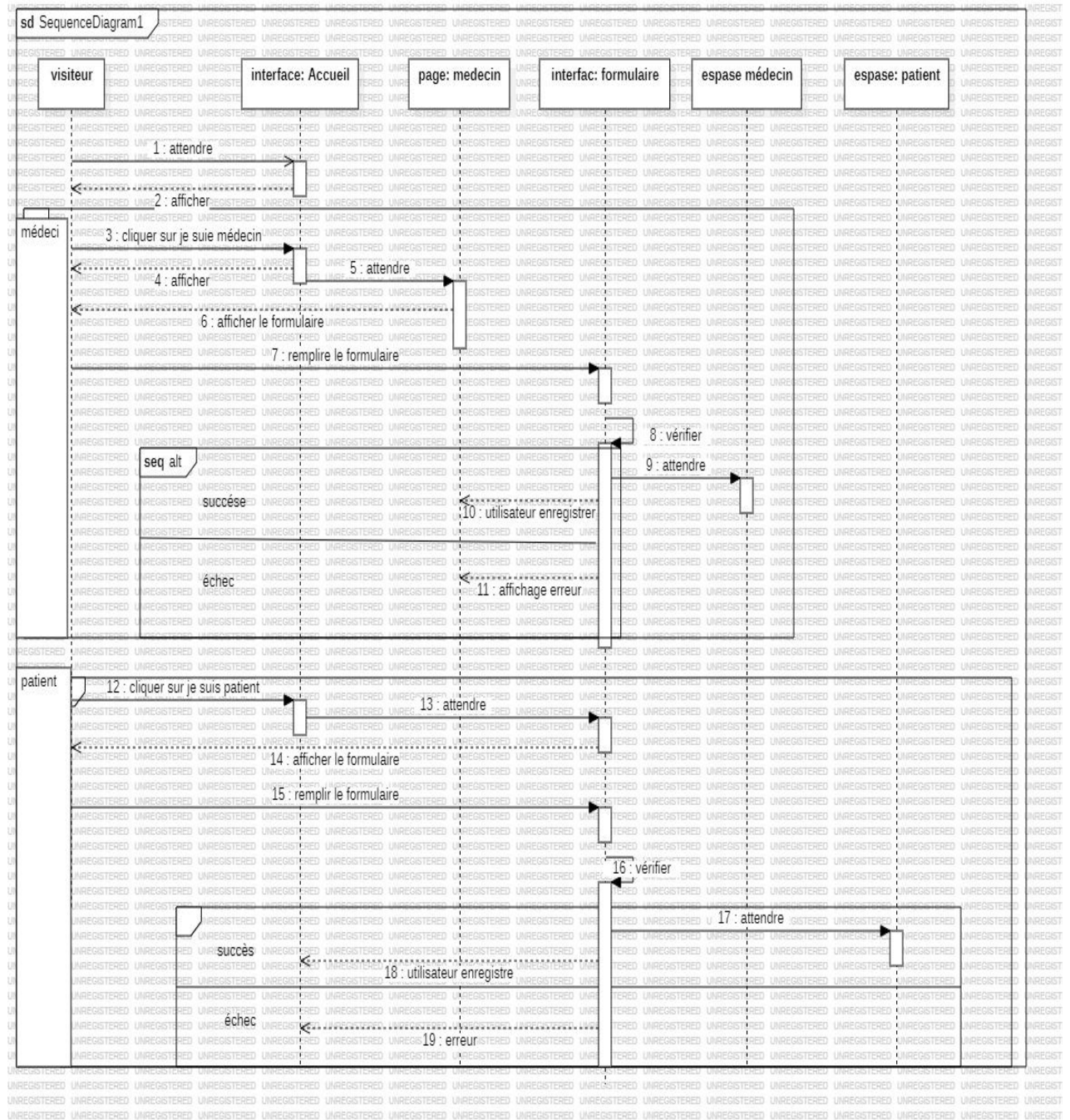


Figure 2: 2 Diagramme de séquence « Inscription »

2.3.2.2.2. Diagramme de séquence « Authentification »

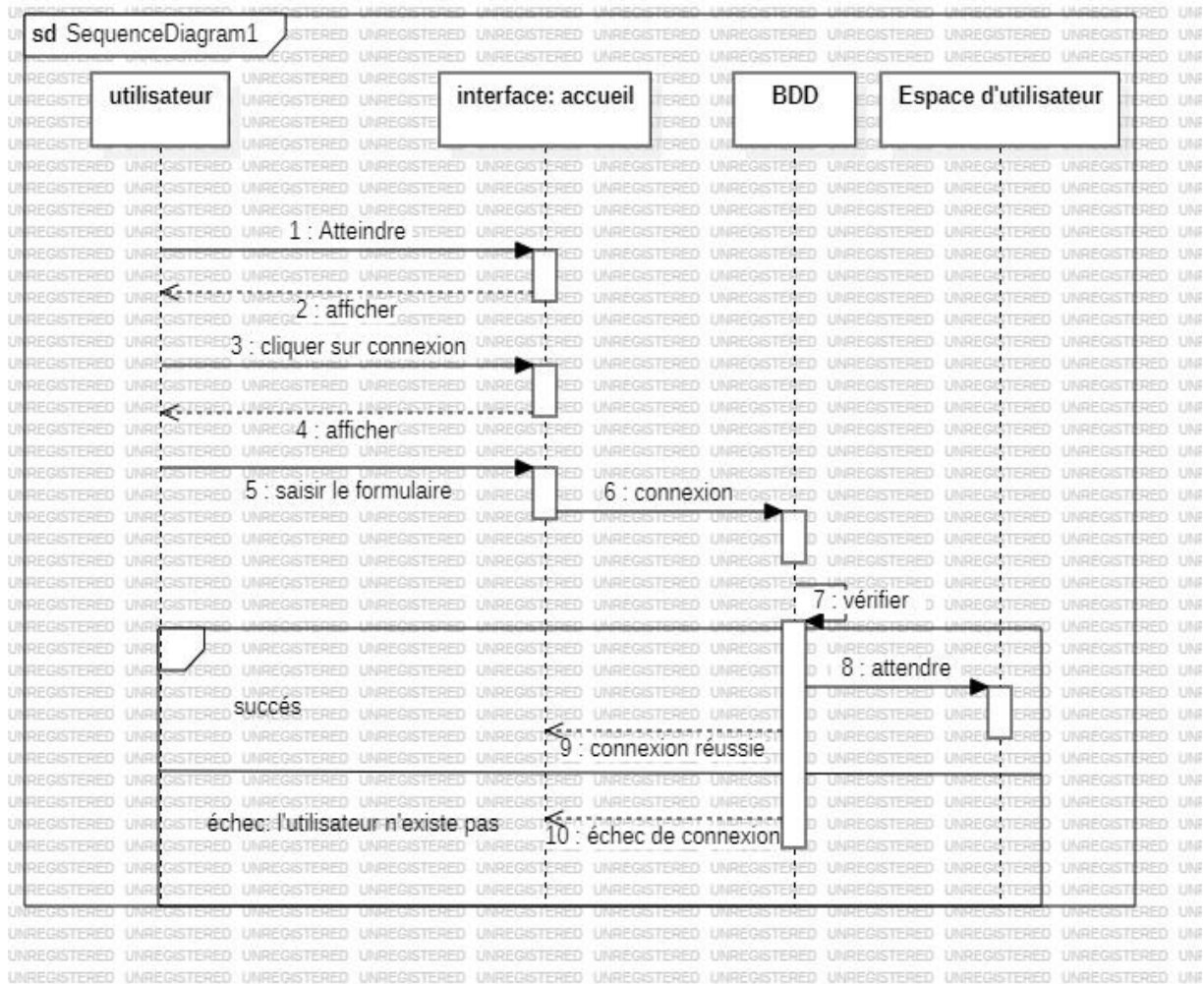


Figure 2: 3 Diagramme de séquence « Authentification »

2.3.2.2.3. Diagramme de séquence « Recherche un médecin »

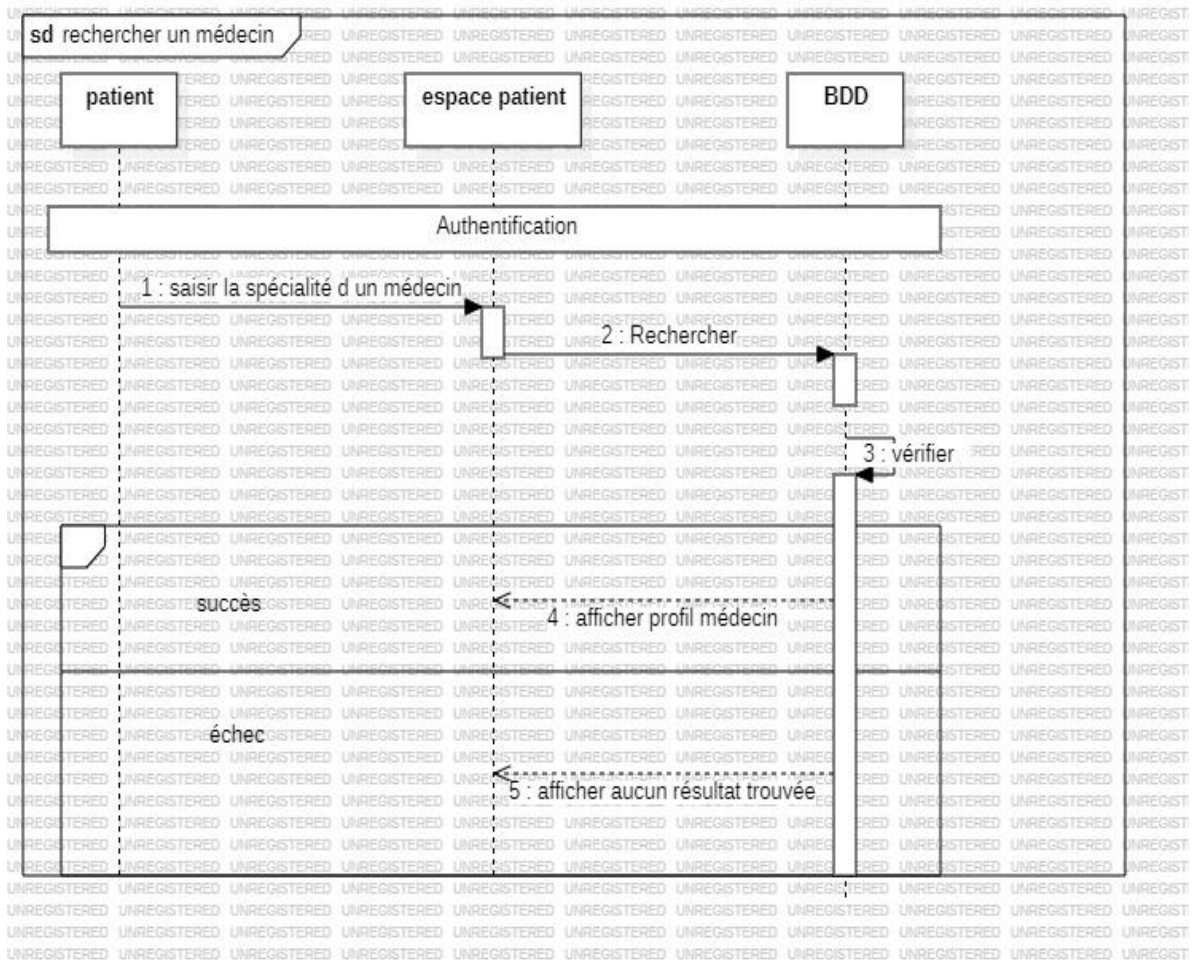


Figure 2: 4 : Diagramme de séquence « Recherche un médecin »

2.3.2.2.4. Diagramme de séquence « prendre un Rendez-vous »

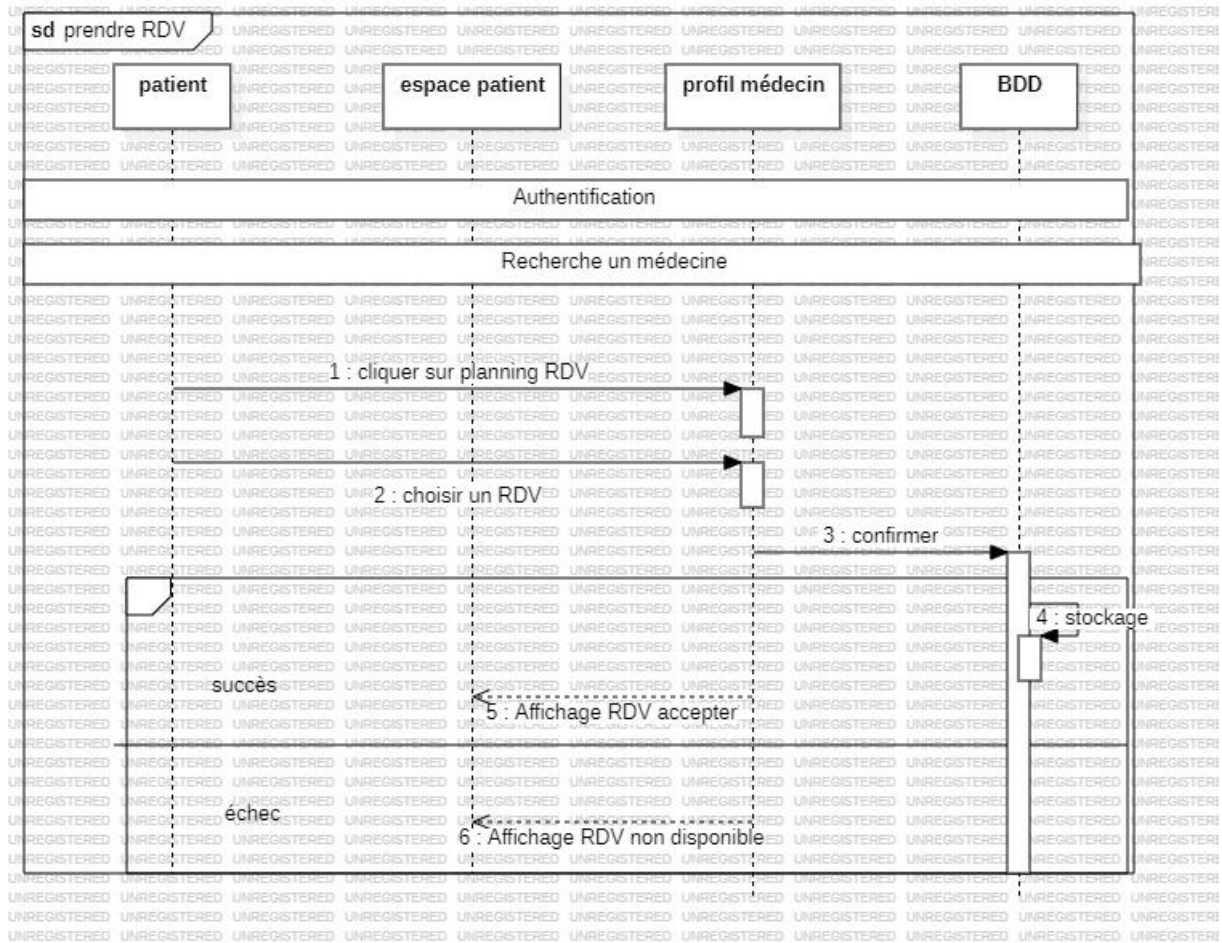


Figure 2: 5 Diagramme de séquence « Recherche un Rendez-vous »

2.3.2.2.5. Diagramme de séquence « contacter un médecin »

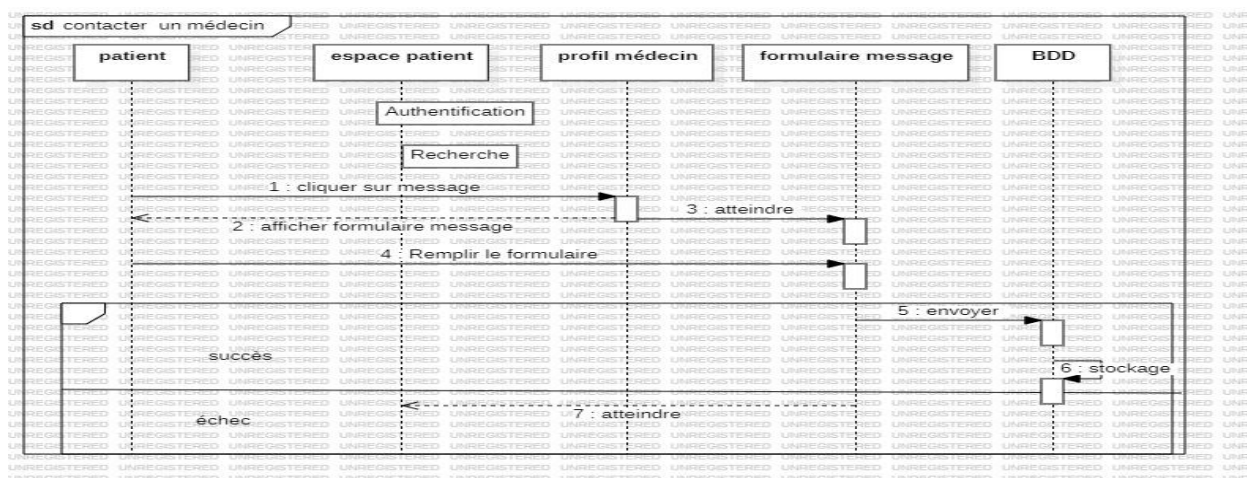


Figure 2: 6 Diagramme de séquence « contacter un médecin »

2.3.2.2.6. Diagramme de séquence « Consulter les RDV »

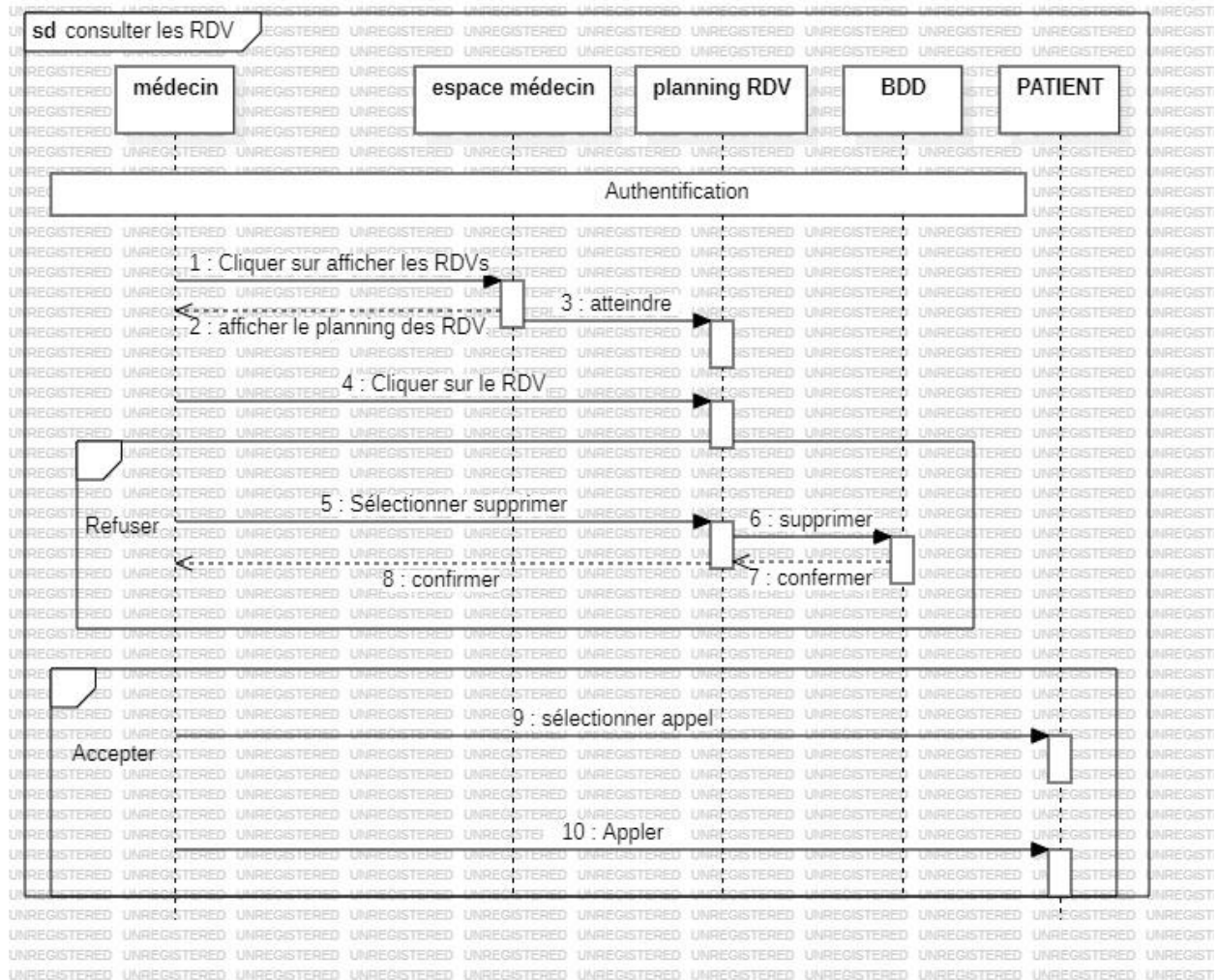


Figure 2: 7 Diagramme de séquence « consulter les RDV »

2.3.3. Diagramme de classe :

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Il s'agit donc d'une vue purement statique car on ne tient pas compte du facteur temporel dans le comportement du système. Le diagramme de classes modélise les concepts du domaine d'application ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application. Il permet aussi

de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier [16]

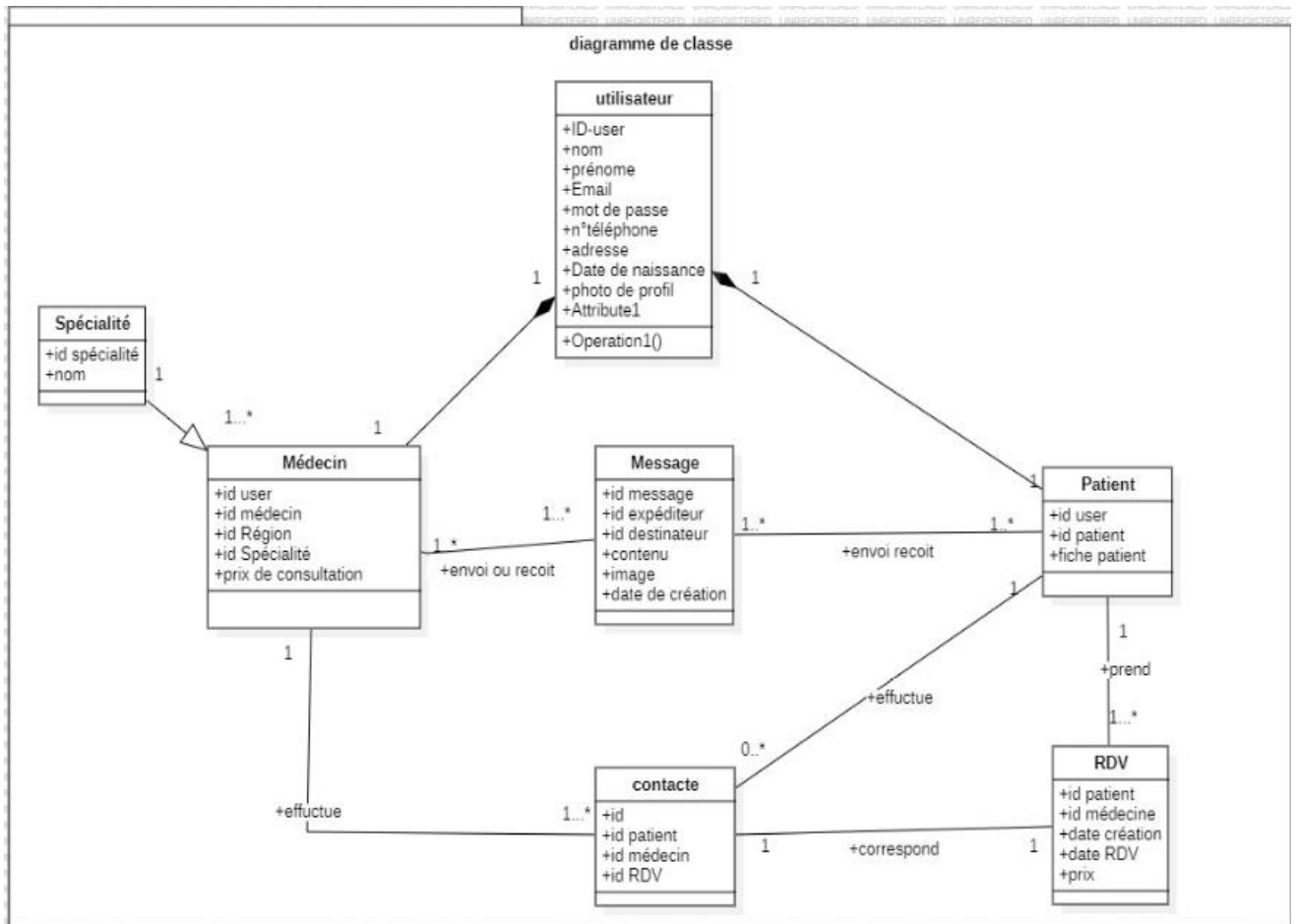


Figure 2: 8 diagramme de classe en générale.

2.4. Conclusion :

Durant ce chapitre nous avons présenté l'étape analyse et conception des besoins et planifier le fonctionnement de notre future Application, dont nous avons décrit d'une façon détaillé la modélisation en se basant sur les diagrammes du langage UML à savoir le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classe pour faciliter la réalisation concrète de la plateforme qui fera l'objet du chapitre suivant.

Chapitre 03 :

Réalisation

3.1. Introduction :

La réalisation représente l'étape qui suit immédiatement la phase de conception. C'est l'aboutissement final de notre projet et c'est la raison d'être du projet lui-même. Dans ce chapitre nous allons présenter dans un premier lieu l'environnement matériel et logiciel de développement, par la suite, nous décrirons la phase d'implémentation en nous basant sur quelques interfaces.

3.2. Langage et Environnement de programmation:

3.2.1. Environnement de travail :

Pour la réalisation de notre projet, nous avons à utiliser un ordinateur ACER caractérisé par :

- Système d'exploitation : Windows 10.
- Processeur : Intel R Corê(TM) i3 4300U 2.40 GHz.
- Mémoire vive : 4 Go.
- Disque Dur : 250 Go SSD

3.2.2. Environnement logiciel :

a) Les langages :

o Flutter :

Flutter est un kit de développement de logiciel (SDK) d'interface utilisateur open-source créé par Google il est utilisé pour développer des applications multiplateformes pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code. Dont la première version a été publiée

sous forme de projet open source à la fin de l'année 2018. Flutter met à disposition une grande variété de bibliothèques d'éléments d'IU standard pour Android et iOS. Et permet aussi de recharger la page automatiquement sans faire l'exécution du code ce rechargement simplifie le débogue.

Les applications développées avec Flutter prennent l'aspect d'applications typiques des systèmes correspondants et se comportent également de manière similaire sans que le programmeur c'est-à-dire vous n'ait besoin de prêter attention à ces caractéristiques.

Les principales composantes de Flutter sont :

- La plate-forme Dart les applications flutter sont écrit en Dart et utilisent de nombreuses fonctionnalités plus avancées du langage les versions des applications Flutter sont compilées avec ahead-of-time (AOT) compilation à la fois sur Android et iOS rendant possible les hautes performances de Flutter sur les appareils mobile et fonctionne dans la machine virtuelle Dart et dispose d'un moteur d'exécution « Just-in-time »
- Le moteur de Flutter le moteur de Flutter écrit principalement en C++ fournit un support de rendu de bas niveau en utilisant la bibliothèque graphique Sika de Google, il s'interface avec des SDK spécifiques à la plate-forme fournis par Android et iOS
- La bibliothèque fondation la bibliothèque fondation écrite en Dart fournit des classes et des fonctions de base qui sont utilisées pour construire des applications comme API pour communiquer avec le moteur
- Les Widget spécifiques à la conception le Framework Flutter contient deux ensembles de widget qui sont conformes à des langages de conception spécifiques, les widgets Matériel Design implémente le langage de conception [17]

○ **Dart :**

C'est un langage de programmation orienté objet basé sur la classe avec un récupérateur de mémoire optimisé pour les applications sur plusieurs plateformes. Il est développé par Google et se concentre sur la réalisation d'applications mobiles, web, bureau...

La programmation dans Dart doit être une alternative attrayante à JavaScript dans des navigateurs Web modernes à cause des faiblesses de JavaScript que les développeurs ne peuvent pas être corrigées par son développement.

Le langage Dart est développé et lancé depuis 2010. La syntaxe de son programme ressemble au langage de programme C.

Google a créé de nombreux langages de programmation Dart depuis l'émergence de Flutter. Ce langage retrouve des couleurs et a même connu une version 2.0. Il est très performant pour gérer l'allocation et éliminer des objets en mémoire qui ne sont plus utilisés « garbage collector ». Il offre deux modes de fonctionnement. AOT (Ahead Of Time) permet de générer une application native pour chaque plateforme et le code sera optimisé directement pour l'architecture qu'il fonctionnera, et JIT (Just In Time) offre la fonctionnalité de Hot Reload lors de développement. [18]

b) Les outils :

○ **Android studio :**

Android Studio est un environnement de développement intégré (IDE) pour le développement sur la plateforme Android. Il a été annoncé en mai 2013.

Android est disponible librement sous la licence Apache 2.0. Basé sur le logiciel IDEA de JetBrains 'IntelliJ', Android Studio est conçu spécifiquement pour le

développement Android. Il est disponible en téléchargement sur les systèmes d'exploitation ; Windows, Mac OS et Linux

Android Studio permet principalement d'éditer les fichiers Java et les fichiers de configuration d'une application Android. Il propose aussi des outils pour gérer le développement d'applications multilingues et permet de visualiser la mise en page des différents types et tailles d'écrans avec des résolutions variées simultanément [19]



Figure3: 1 Logo d'Android Studio.

○ **Visuel studio :**

Visual Studio Code est un éditeur de code source léger mais puissant qui s'exécute sur votre bureau et est disponible pour Windows, MacOS et Linux. Il est livré avec une prise en charge intégrée de JavaScript, Type Script et Node.js et dispose d'un riche écosystème d'extensions pour d'autres langages et environnements d'exécution (tels que C++, C#, Java, Python, PHP, Go, .NET). [20]



Figure 3 :2 Logo de Visual Studio Code.

- **FireBase :**

FireBase est un ensemble d'outils pour l'hébergement et le développement d'applications mobiles et web, qui permet l'envoi de notifications et de publicités, la remontée des erreurs et des clics effectués dans l'application. [21]



Figure 3: 3 Logo de FireBase

- **Star UML :**

StarUML est un logiciel de modélisation UML, commercialisé en open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL. L'objectif de la reprise de ce projet était de remplacer des solutions commerciales comme IBM Rational Rose ou Borland Together. StarUML prend en charge la plupart des diagrammes spécifiés dans la norme UML 2.0. [22]



Figure 3: 4 Logo de StarUML.

○ **Java Développement Kit (JDK) :**

« Le Java Développement Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé, transformé en byte code destiné à la machine virtuelle Java. » [23]



Figure3: 5 Logo de JDK

○ **Android Software Développement Kit (SDK) :**

Le SDK est un ensemble d'outils que met à disposition Google afin de nous permettre de développer des applications pour Android. Il est disponible pour Windows, MacOS X et linux et inclut des outils ainsi qu'un émulateur Android pour exécuter des applications. [24]



Figure 3: 6 Logo de SDK

3.3. La charte graphique de TABIBI :

La charte graphique, dont la dénomination correcte est cahier des normes graphiques, est un document de travail qui contient l'ensemble des règles fondamentales d'utilisation des signes graphiques qui constituent l'identité graphique d'une organisation, d'un projet, d'une entreprise.

Dans ce qui suit, nous présenterons la charte graphique de notre application

1) Le choix de couleurs :

Etant donné que TABIBI sera utilisée dans le domaine médical, ce qui fait qu'on a opté pour une couleur qui est adéquate au thème.

Un vert qui invite au calme et au repos. Il est symbole de croissance, de santé, de fraîcheur et de naturelle loge de l'application.

2) Le Logo de l'application :

Le logo de notre application, a été choisi d'une manière à ce qu'il soit soft, qu'il mette en avant le nom attribué à nos applications.



Figure 3 : 7 Présentation du Logo de l'application

3.4. Présentation de quelques interfaces de l'application :

Dans la présente partie, nous décrivons les diverses interfaces de notre application, ces interfaces sont les suivantes :

3.4.1. L'interface d'écran de démarrage de l'application mobile:

C'est la première interface qui s'affiche au patient après que ce dernier a décidé de télécharger et d'installer l'application dans son smartphone.

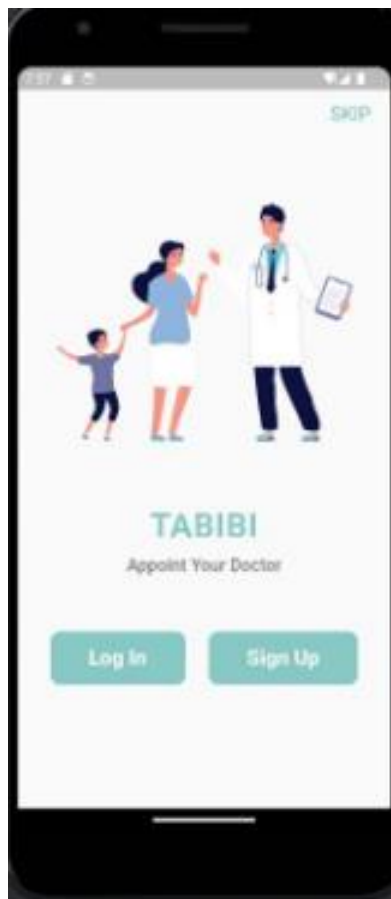


Figure 3: 8 Présentation de l'interface «Démarrage»

3.4.2. Interface d'inscription :

Cette interface (**figure 3.9**) contient une suite des champs de saisie que le patient devrait remplir avec ses informations personnelles pour effectuer son inscription.



Figure3: 9 Présentation de l'interface «Inscription»

3.4.3. Interface d'authentification :

Une interface (**figure 3.10**) qui permet au patient de se connecter à son compte. Pour cela, ce dernier doit saisir son email et son mot de passe pour pouvoir y accéder à l'interface d'accueil.



Figure3: 10 Présentation de l'interface « Authentification »

3.4.4. Interface d'accueil :

L'interface d'accueil joue un rôle d'une porte qui mène vers les autres interfaces, à l'aide des deux barres de menu vertical et horizontal qui se situe à gauche et en bas de l'écran.

Le patient navigue facilement vers les autres interfaces de plus, il peut chercher un médecin dans la barre de recherche selon la position de ce dernier.



Figure3: 11 Présentation de l'interface « Accueil »

3.4.5. Interface Profile :

Elle permet au patient de modifier les informations personnelles de son compte sur l'application et les valider

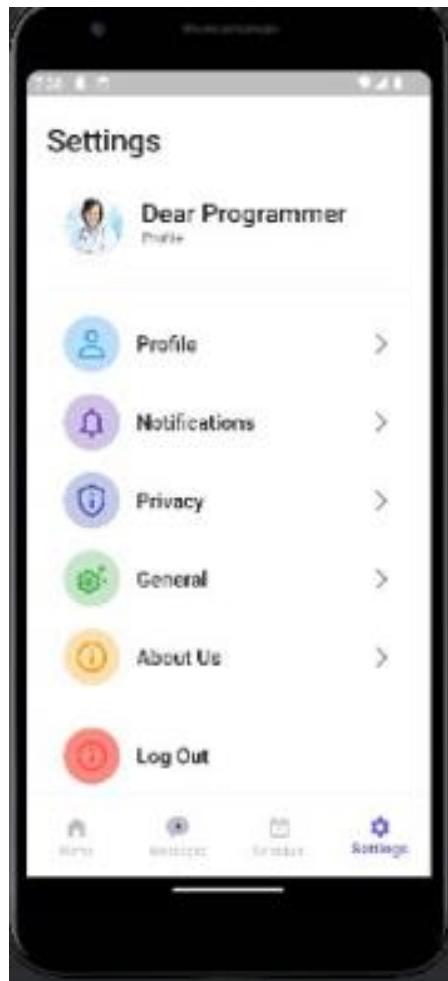


Figure3: 12 Présentation de l'interface « Profil »

3.4.6. Interface Chercher Médecin

Cette interface (**figure 3.13**) permet au patient de chercher un médecin. Ce dernier saisie un nom ou une spécialité médicale dans la barre de recherche. Ensuite le système lui affiche une liste des médecins d'où le patient peut sélectionner un seul et consulte son profil

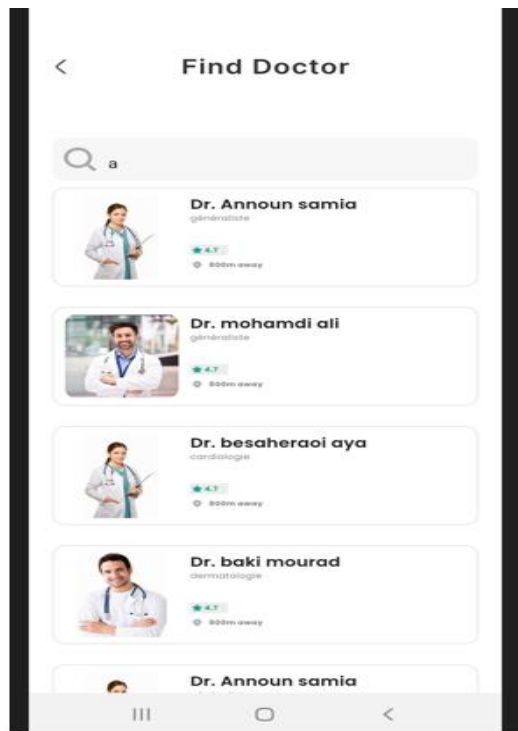


Figure3. 13 : Présentation de l'interface « Chercher médecin »

3.4.7. Interface prendre un rendez-vous :

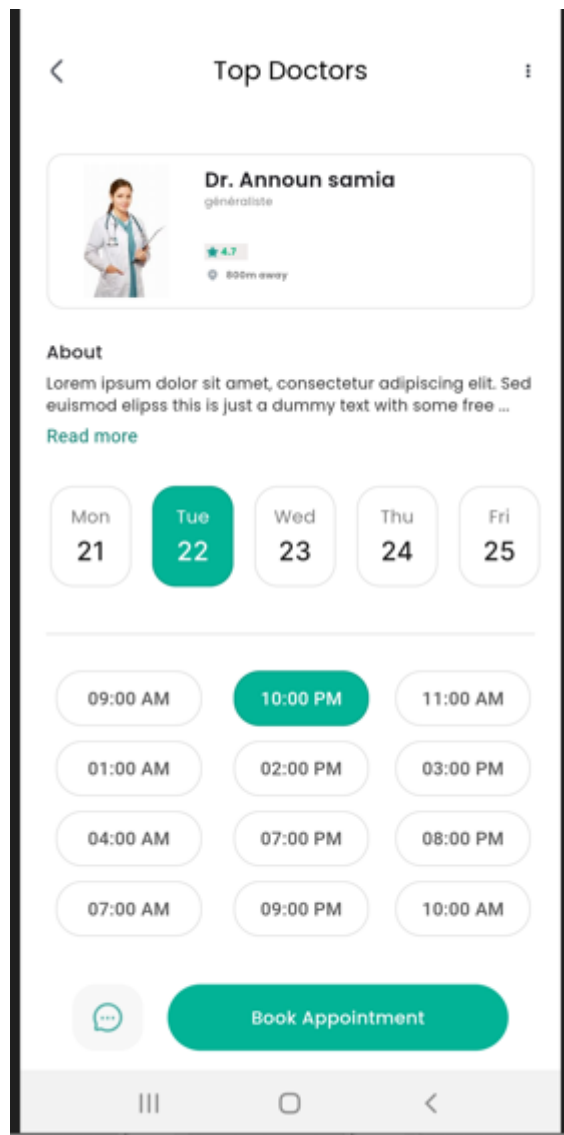


Figure 3 : 14Présentation de l'interface «Prendre un rendez-vous »

3.4.8. Interface profil du médecin :

Une interface qui affiche les informations détaillées du médecin sélectionné et ensuite cliquer sur prendre un rendez-vous.

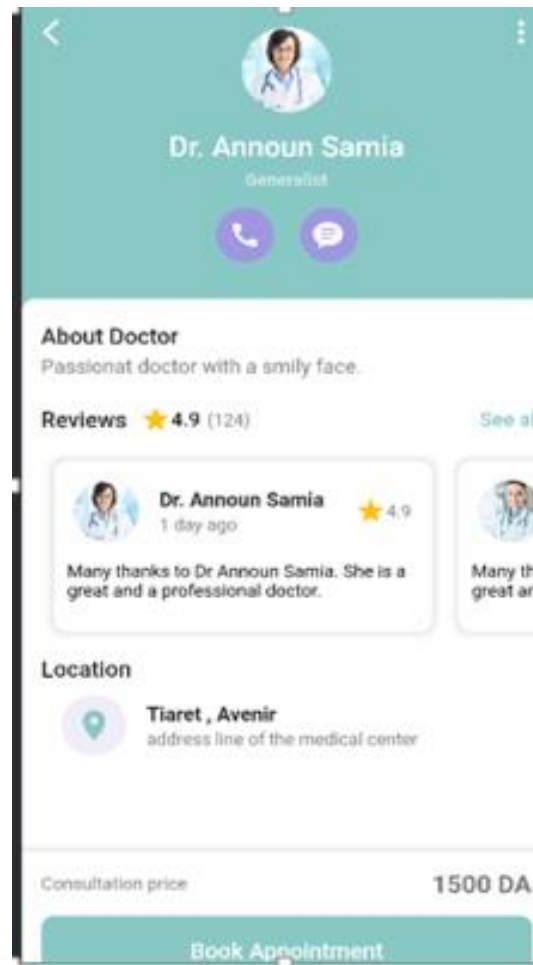


Figure 3 :15Présentation de l'interface « Profile médecin »

3.5. Conclusion

La phase de réalisation est l'étape la plus importante dans le cycle de vie d'une application. Dans ce dernier chapitre, nous avons présenté l'environnement de développement, les langages ainsi que les principaux outils utilisés qui nous ont permis de réaliser nos applications mobile.

Nous avons également présenté nos applications à travers une arborescence des vues de ces dernières ainsi qu'à travers les interfaces conçues pour leur utilisation.

Conclusion Générale

Conclusion Générale :

En conclusion, ce mémoire de recherche a mis en évidence l'importance de développer une application mobile médicale pour améliorer la communication et la collaboration entre les professionnels de santé et les patients. À travers une approche méthodologique rigoureuse et une analyse approfondie des besoins des utilisateurs, nous avons conçu une application mobile efficace et intuitive, offrant une expérience utilisateur optimale.

Ce travail de recherche a souligné l'impact positif que peut avoir une telle application sur la qualité des soins et la gestion des données médicales, en facilitant l'interaction entre les acteurs de la santé et en favorisant une prise en charge plus personnalisée des patients. En mettant l'accent sur l'innovation technologique et la centralité de l'expérience utilisateur, ce mémoire s'inscrit dans une démarche visant à améliorer les services de santé et à répondre aux défis actuels du secteur de la santé numérique.

En encourageant la collaboration entre les professionnels de santé, les patients et les développeurs d'applications, ce mémoire cherche à promouvoir une approche holistique de la santé numérique, axée sur la co-construction de solutions adaptées aux besoins réels des utilisateurs. En définitive, il s'agit de démontrer comment les technologies mobiles peuvent être mises au service de la santé pour contribuer à une meilleure prise en charge des patients et à une amélioration globale des services de santé.

Ainsi, ce mémoire aspire à ouvrir de nouvelles perspectives et à inspirer de futurs travaux de recherche et projets innovants dans le domaine de la santé numérique, en mettant en lumière le potentiel des applications mobiles pour transformer positivement le secteur de la santé et renforcer la relation entre les professionnels de santé et les patients.

Bibliographie

[1]: Web: https://fr.wikipedia.org/wiki/Application_mobile

[2]: [Application mobile-Définitions Marketing» L'encyclopédie illustrée du marketing](#) (definitions-marketing.com)

[3]: <https://jobphoning.com/dictionnaire/application-mobile#>

[4]:

[5]: <https://mbamci.com/comprendre-le-developpement-dapplication/>

[6] : LEE S. Barney. Developing Hybrid Applications for the iPhone : Using HTML, CSS, and Javascript to Build Dynamic Apps for the iPhone. Addison-Wesley Professional, 2009. 195 p.

[7]. ERHAT, Yasser. LAAOUAR, Djaafar. Réalisation d'une application mobile de gestion de la vie estudiantine pour les Smartphones Android. Centre Universitaire : 'Abdelhafid Boussof', 2013.

[8] : <http://www.techno-science.net/?onglet=news&news=11005>.

[9] : https://www.editions-ellipses.fr/PDF/9782340025547_extrait.pdf

[10]:<http://www.phonandroid.com/toute-l-histoire-et-la-chronologie-d-android-dossier.html>

En octobre 2008, la première version bêta d'Android est publiée, sans nom officiel.

[11] : <https://fr.wikipedia.org/wiki/Android>

[12] : PASCAL, Roques, UML Modéliser une application web, Eyrolles, 2008, 246 p

[13] : A. V. ANDRE PASCAL, Développement de logiciels avec UML 2 et OCL. Ellipses, 2013.

[14]NNICK, Lassus. Méthodologie des systèmes d'information – UML Cour, CNAM ANGOULÊME, 2000.

[15] : B. Y.-. B. Oussama, Conception et réalisation d'un système d'apprentissage dans les réseaux sociaux, Tlemcen, 2011

[18] « <http://www.zdnet.fr/actualites/android-studio-une-version-10-pour-l-ide-de-google39811025.html> » consulté le 10/03/2024

[19] « Josko Marsic Gordan Struklee. Implementating schc in smart meters. <https://code.visualstudio.com/>, (Consulter le 15/03/2024).

[20] firebase <https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0>

[21] <https://inf1410.teluq.ca/files/2014/01/INF1410-PresentationStarUML.pdf>.

[22] « <http://www.commentcamarche.net/contents/559-java-introduction> » consulté le 11/03/2024

[23] « <https://www.techopedia.com/definition/4220/android-sdk> » consulté le 11/03/2024.



Business Model Canva

1. Fiche technique du projet

Nom et prénom	Zoukh Ahlem
Intitulé du projet	Médicale
Statut juridique	EURL
Numéro de téléphone	06-73-41-33-42
Adresse email	Ahlemzoukh2020@gmail.com
Ville d'activité	Tissemsilt

2. Le projet

TABIBI est une application mobile visant à faciliter la relation entre le patient et le médecin. vise à fournir un moyen pratique et efficace de communication entre le patient et le médecin. Le patient peut, à travers l'application, prendre des rendez-vous avec le médecin, recevoir des consultations médicales par téléphone ou par chat, obtenir des conseils de santé et suivre régulièrement son état de santé

Cette application permet aux patients d'accéder rapidement et en toute confiance aux soins médicaux sans avoir à attendre dans de longues files d'attente ou à visiter fréquemment les cabinets médicaux. De plus, les médecins peuvent, grâce à l'application, suivre leurs patients de manière plus efficace et leur fournir les soins nécessaires en temps opportun

Cette application peut être utile aux patients souffrant de maladies chroniques, nécessitant un suivi médical régulier. De plus, elle peut contribuer à améliorer la



qualité des soins de santé de manière générale et à réduire la charge de travail sur les établissements de santé.

3. La valeur proposée

1) Le problème à résoudre :

- Longs délais d'attente : les patients peuvent devoir attendre des semaines, voire des mois, pour obtenir un rendez-vous avec un spécialiste, ce qui entraîne des retards dans l'obtention du traitement médical nécessaire.
- Difficulté à prendre rendez-vous : certains patients peuvent avoir des difficultés à prendre rendez-vous en raison d'un manque de connaissances sur le système de santé ou de difficultés à naviguer dans le processus de planification.
- Accès limité aux spécialistes : certains patients peuvent vivre dans des zones où il y a une pénurie de spécialistes, ce qui rend difficile pour eux d'obtenir les soins dont ils ont besoin.
- Heures d'ouverture limitées : certains hôpitaux peuvent avoir des heures d'ouverture limitées, ce qui rend difficile pour les patients de prendre rendez-vous en dehors des heures normales de bureau.
- Surpeuplement : certains hôpitaux peuvent être surbookés, ce qui entraîne
- Longs délais d'attente, retards dans l'obtention d'un traitement et patients refusés.
- Urgences : certains hôpitaux sont submergés d'urgences et peuvent ne pas être en mesure de prendre rendez-vous.

2) Les autres projets qui ont ciblé le même problème et qui ont été mis en œuvre :

- Beesiha.

3) Objectifs du projet ou résultats attendus :



- Fournissez un moyen simple de planifier un rendez-vous.
- Donne accès à de nombreux spécialistes.
- Fournit des options de rendez-vous 24 heures sur 24.
- Prise de rendez-vous facile.
- Fournit une ambulance en cas d'urgence.

4. Valeur proposée selon les critères suivants

Valeur innovante ou nouvelle :	Compte personnelles : Chaque patient a son propre compte pour faciliter l'accès à ses informations
Valeur au prix :	La valeur-prix est obtenue en fournissant une application efficace et abordable du système d'information sur la santé. Une solution de santé est offerte qui répond aux besoins financiers d'une entreprise de santé et peut établir un équilibre entre le rendement et le coût.
Valeur conceptuelle :	Interface utilisateur facile à utiliser et attrayante, expérience utilisateur pratique et adaptée aux différents utilisateurs
Valeur avec un rendement élevé :	Un système d'information sur la santé robuste et efficace qui répond aux besoins de l'établissement en matière de santé. L'application est rapide, stable, fiable et capable de traiter un



	grand volume de données de manière efficace et précise.
Valeur pour tous les services :	Fournir un service intégré et complet à l'établissement de santé. Cela comprend la formation continue et le soutien technique, les mises à jour et le développement continu de l'application, et la prestation de services de maintenance continue, de gestion des données et de sécurité.

4) Customer Segments

Géographique الجغرافية	Démographique (B2C)	Démographique (B2B)	Psycho graphique العوامل النفسية والشخصية	Comportementa l السلوكيات
Continent القارة "افريقيا"	Age العمر 'كل الفئات العمرية'	Secteur القطاع 'قطاع الصحة'	Classe sociale طبقة الاجتماعية 'كل الطبقات الاجتماعية'	Usage استخدام 'البحث عن الطبيب وحجز المواعيد عن بعد وإمكانية اجراء محادثات مع الطبيب المختص'



Pays الدولة "الجزائر"	Sexe الجنس "كل الاجناس"	Nombre d'employés عدد العمال في القطاع	Niveau de vie المستوى المعيشي ////	Loyauté ///الوفاء
Région الجهة "الغربية للجزائر"	Revenus annuel متوسط الدخل	Maturité de l'entreprise نضج المؤسسة	Valeurs القيم	Intérêt اهتمام
Département الولاية "تيسمسيلت"	Etat matrimonial الحالة الاجتماعية	Situation financière الحالة المالية للمؤسسة	Personnalité الشخصية	Passion الهواية وشغف
Ville الدائرة او البلدية ثنية الحد	Niveau d'étude المستوى الدراسي "جامعي"	Détention/ actionariat الملكية/المساهمة	Convictions المعتقدات	Sensibilité حساسيات
Quartier الحي ///	Profession المهنة	Valorisation/ capitalisation boursière التقييم / القيمة السوقية	Présence digitale et sur les réseaux sociaux استعمال التكنولوجيا في التواصل	Habitude de consommation عادة الاستهلاك



<p>Climat</p> <p>المناخ</p> <p>"معتدل"</p>	<p>Culture</p> <p>الثقافة</p>	<p>Business model</p> <p>نموذج الأعمال</p>	<p>Centres d'intérêts</p> <p>مراكز الاهتمام</p>	<p>Mode de paiement</p> <p>طرق الدفع</p> <p>"باستخدام بريدي موب"</p>
	<p>Religion</p> <p>الدين</p> <p>"الإسلام"</p>	<p>Secteur servi</p> <p>القطاع الذي يخدمه</p> <p>"قطاع الصحة"</p>		<p>Connaissance</p> <p>المعرفة</p>
	<p>Langue</p> <p>اللغة</p>	<p>Technologie utilisée</p> <p>التكنولوجيا المستعملة</p> <p>"تطوير تطبيقات الهاتف"</p>		<p>Nature de la demande</p> <p>طبيعة الطلب</p> <p>"اشترك شهري"</p>
		<p>Format du produit ou packaging</p> <p>شكل المنتج أو التعبئة والتغليف</p> <p>تطبيق هاتف</p>		<p>Fréquence d'achat</p> <p>عدد مرات الطلب على السلعة</p> <p>"كل شهر"</p>



<p>Ventes directes :</p>	<p>L'équipe de vente interne peut être utilisée pour communiquer directement avec les établissements médicaux, les hôpitaux et les cliniques pour commercialiser et fournir des applications mobiles et des services informatiques de santé</p>
<p>Grossistes :</p>	<p>Peut coopérer avec les grossistes de l'industrie de la santé pour commercialiser et distribuer le produit aux établissements médicaux et aux cliniques.</p>
<p>Distributeurs :</p>	<p>Les distributeurs locaux ou mondiaux peuvent être utilisés pour distribuer et connecter l'application mobile et les technologies de l'information sur la santé aux utilisateurs finaux</p>
<p>Distribution au détail :</p>	<p>Les magasins médicaux, les centres médicaux et les magasins en ligne peuvent être situés pour distribuer et vendre l'application mobile et les services associés</p>

La relation avec client

- Assurer un support technique : cela implique d'offrir une assistance technique et une formation aux clients afin de les aider à utiliser l'application et les technologies de l'information en santé de manière efficace, en comprenant pleinement les caractéristiques et les fonctionnalités.



- Service client de qualité : fournir un service client exceptionnel en répondant de façon rapide et efficace aux demandes et aux problèmes, tout en satisfaisant les besoins des clients de manière courtoise.
- Communication régulière : maintenir une communication continue avec les clients à travers des bulletins d'information, des e-mails, des événements spéciaux, ainsi que leur participation à des expositions et des événements professionnels.
- En utilisant ces moyens pour gérer et entretenir des relations clients efficaces, il est possible d'améliorer les interactions et la communication avec les clients, de répondre à leurs besoins, de garantir leur satisfaction et d'établir des relations solides et durables.
-

Les Partenaires clé

Les partenaires clés d'une application de TABIBI vous dans le domaine médical peuvent inclure :

- Les professionnels de santé : les médecins, les infirmières, les praticiens et autres professionnels de santé qui utilisent l'application pour organiser leurs consultations et gérer leurs rendez-vous avec les patients.
- Les établissements de santé : les hôpitaux, les cliniques, les cabinets médicaux et autres établissements de santé qui utilisent l'application pour gérer leurs rendez-vous et leur planning.
- Les patients : les utilisateurs finaux de l'application, qui prennent des rendez-vous, consultent leur planning et communiquent avec leur équipe médicale à travers la plateforme.

En travaillant en étroite collaboration avec ces partenaires clés, une application de gestion des rendez-vous médicaux peut offrir une expérience optimale aux patients, améliorer l'efficacité des établissements de santé et renforcer la collaboration entre les différents acteurs du secteur médical.





طبيعة الشراكة	معلومات حول الشركاء	الشركاء
	Établissements médicaux et hôpitaux	الشريك الأول
الشريك ليس منافس وإنما هي عملية ترتيب متبادل المنفعة يكون لنا معه مصلحة مشتركة في تطوير منتجات جديدة استراتيجية مصممة لتقليل المخاطر، والتي قد ترتبط بإحضار منتج جديد إلى السوق علاقتنا مع الشريك هي علاقات بين المشتري والمورد	Médecins, infirmières et équipes médicales	الشريك الثاني
الشريك ليس منافس وإنما هي عملية ترتيب متبادل المنفعة يكون لنا معه مصلحة مشتركة في تطوير منتجات جديدة استراتيجية مصممة لتقليل المخاطر، والتي قد ترتبط بإحضار منتج جديد إلى السوق علاقتنا مع الشريك هي علاقات بين المشتري والمورد	Patients et patients potentiels + Établissements de santé publics et privés	الشريك الثالث

structure Costs تكاليف



10000DA	تكاليف التعريف بالمنتج أو المؤسسة Frais d'établissement
100000DA	تكاليف (التكوين-برامج الاعلام الالي المختصة) Logiciels, formations
180000DA	Droit au bail الحق في الإيجار
10000DA	Caution ou dépôt de garantie وديعة أو وديعة تأمين
150000DA	Matériel الآلات-المركبات-الأجهزة
250000DA	Matériel de bureau تجهيزات المكتب

المجموع = 610 000DA

نفقاتك أو التكاليف الثابتة الخاصة بمشروعك



60000DA	Assurances التأمينات
320000DA	Téléphone, internet الهاتف والانترنت
100000DA	Eau, électricité, gaz فواتير الماء - الكهرباء - الغاز
500000DA	Nettoyage des locaux تنظيف المباني
120000DA	Budget publicité et communication ميزانية الإعلان والاتصالات

650 000 DA = المجموع

Revenue Stream مصادر الإيرادات

20%	Apport personnel ou familial المساهمة الشخصية أو العائلية
80%	Prêt n°1 (nom de la banque) اسم البنك قرض رقم
//	Autre financement (libellé) تمويل آخر

100% المجموع =



رقم الأعمال

بيع المنتج في السنة الأولى **Votre chiffre d'affaires de la première année**

متوسط أيام العمل في الشهر	بيع المنتج في السنة الأولى
05	1Mois 0 الشهر
05	2Mois 0 الشهر
05	3Mois 0 الشهر
05	4Mois 0 الشهر
10	5Mois 0 الشهر
10	6Mois 0 الشهر
10	7Mois 10 000DA الشهر
15	8Mois 10 000DA الشهر
15	9Mois 15 000DA الشهر
20	10Mois 20 000DA الشهر
20	11Mois 20 000DA الشهر
20	12Mois 25 000DA الشهر



المجموع = 10000000DA

2% النسبة المئوية للزيادة في حجم الأعمال بين كل شهر لسنة الأولى

Votre chiffre d'affaires de la deuxième année بيع المنتج في السنة الثانية

متوسط أيام العمل في الشهر	بيع المنتج في السنة الثانية
25	1Mois الشهر 25 000DA
25	2Mois الشهر 25 000DA
30	3Mois الشهر 30 000DA
32	4Mois الشهر 32 000DA
35	5Mois الشهر 35 000DA
60	6Mois الشهر 60 000DA
90	7Mois الشهر 90 000DA
100	8Mois الشهر 100 000DA
140	9Mois الشهر 140 000DA
160	10Mois الشهر 160 000DA
210	11Mois الشهر 210 000DA
280	12Mois الشهر 280 000DA



935 000DA = لمجموع

5% النسبة المئوية للزيادة في حجم الأعمال بين كل شهر لسنة الثانية

بيع المنتج في السنة الثالثة Votre chiffre d'affaires de la troisième année

متوسط أيام العمل في الشهر	بيع المنتج في السنة الثالثة
330	1Mois الشهر 330 000DA
390	2Mois الشهر 390 000DA
450	3Mois الشهر 450 000DA
500	4Mois الشهر 500 000DA
600	5Mois الشهر 600 000DA
720	6Mois الشهر 720 000DA
850	7Mois الشهر 850 000DA
860	8Mois الشهر 860 000DA
900	9Mois الشهر 900 000DA
920	10Mois الشهر 920 000DA
980	11Mois الشهر 980 000DA
1000	12Mois الشهر 1 000 000DA



المجموع = 5 743 900DA

7% النسبة المئوية للزيادة في حجم الأعمال بين كل شهر لسنة الثالثة

تطور حجم رقم الأعمال في السنة

- 30% و2 السنة 1 النسبة المئوية للزيادة في حجم الأعمال بين السنة
- 40% و3 السنة 2 النسبة المئوية للزيادة في حجم الأعمال بين السنة

حاجتك لرأس المال العامل

يوم 30	متوسط مدة الاعتمادات الممنوحة للعملاء بالأيام Durée moyenne des crédits accordés aux clients en jours
يوم 30	متوسط مدة ديون الموردين بالأيام Durée moyenne des dettes fournisseurs en jours

رواتب الموظفين ومسؤولين الشركة

3000000DA	رواتب الموظفين Salaires employés
720000DA	صافي أجور المسؤولين Rémunération nette dirigeant

Designed for:

Designed by:

Date:

Version:

Business Model Canvas

Key Partners

Who are our Key Partners? Who are our key suppliers? Which Key Resources are we acquiring from partners? Which Key Activities do partners perform?

MOTIVATIONS FOR PARTNERSHIPS: Optimization of economy, Reduction of risk and uncertainty, Access to particular resources and activities

Key Activities

What Key Activities do our Value Propositions require? Our Distribution Channels? Customer Relationships? Revenue streams?

CATEGORIES:

Production, Problem Solving, Platform/Network

Key Resources

What Key Resources do our Value Propositions require? Our Distribution Channels? Customer Relationships Revenue Streams?

TYPES OF RESOURCES: Physical, Intellectual (brand patents, copyrights, data), Human, Financial

Value Propositions

What value do we deliver to the customer? Which one of our customer's problems are we helping solve? What bundles of products and services are we offering to each Customer Segment? Which customer needs are we satisfying?

CHARACTERISTICS: Newness, Performance, Customization, "Getting the Job Done", Design, Brand/Status, Price, Cost Reduction, Risk Reduction, Accessibility, Convenience/Usability

Customer Relationships

What type of relationship does each of our Customer Segments expect us to establish and maintain? Which ones have we established? How are they integrated with the rest of our business? How costly are they?

Channels

Through which Channels do our Customer Segments want to be reached? How are we reaching them? How are our Channels integrated? Which ones are best? Which ones are most cost-efficient? How are we integrating them with customer routines?

Customer Segments

For whom are we creating value? Who are our most important customers? Is our customer base a Mass Market, Niche Market, Segmented, Diversified or Sided Platform?

Cost Structure

What are the most important costs inherent in our business model? Which Key Resources are most expensive? Which Key Activities are most expensive?

IS YOUR BUSINESS MODEL DRIVEN BY COSTS?

Driven (leanest cost structure, low price value proposition, maximum automation, extensive outsourcing), Value

Revenue Streams

For what value are our customers really willing to pay? For what do they currently pay? How are they currently paying? How would they prefer to pay? How much does each Revenue Stream contribute to overall revenues?

TYPES: Asset sale, Usage fee, Subscription, Advertising, Commission, Rental/Leasing, License, Royalty, Investment

Subscription Fees, Lending/Renting/Leasing, Licensing, Brokerage fees, Advertising

(focused on value creation, premium value proposition).

(salaries, rents, utilities), Variable costs, Economies of scale, Economies of scope

SAMPLE CHARACTERISTICS: F

FIXED PRICING: List Price, Product feature dependent, Customer segment dependent, Volume dependent

DYNAMIC PRICING: Negotiation (bargaining), Yield Management, Real-time-Market

Business Model Canvas

Designed for:

Designed by:

Date:

Version:

Key Partners <ul style="list-style-type: none"> ➤ Praticiens de santé ➤ Business angles et autres financeurs ➤ Réseau des cofondateurs ➤ Etablissements de santé 	Key Activities <ul style="list-style-type: none"> ➤ Développement de l'application ➤ Prospection commerciale 	Value Propositions <ul style="list-style-type: none"> ➤ Application de prise de rdv médical rapide et pratique ➤ Téléconsultation plus récemment 	Customer Relationships <ul style="list-style-type: none"> ➤ Relation client numérique / automatisée ➤ Evaluation de l'application. 	Customer Segments <p>2 segments :</p> <ul style="list-style-type: none"> ➤ Patients ➤ Professionnels et établissements de santé
Cost Structure <ul style="list-style-type: none"> ➤ Coût salarial (développeurs, commerciaux, etc.) ➤ Coût de l'application (développement, maintenance, etc.) ➤ Coût de propriété intellectuelle (domaine, marque, etc.) 	Revenue Streams <ul style="list-style-type: none"> ➤ Abonnement mensuel pour les professionnels de santé 			
Key Resources	Channels <ul style="list-style-type: none"> ➤ Canal direct (sans intermédiaire) : internet, sms /mails, application 			