

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abou-Bakr Belkaid- TLEMCEM

Faculté de Technologie

Département de Télécommunications

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Télécommunications

Spécialité : Réseaux et Télécommunications

Par:

GHOMRI BENMEZIANE IMAD-DDINE

EL HADJ MIMOUN HICHEM

Thème

« Coach des sportifs » Application sport Android

Soutenu le : 04- 07 – 2021, devant le jury composé de :

H. ZERROUKI

Maître de Conférences à l'Université de Tlemcen

Président

R. MERZOUGUI

Professeur à l'Université de Tlemcen

Encadrant

M. HADJILA

Maître de Conférences à l'Université de Tlemcen

Examineur

Année Universitaire : 2020 / 2021

Dédicace

A nos mères,

« Vous nous avez donné la vie, la tendresse et le courage pour
réussir.

Tout ce que l'on peut vous offrir ne pourra exprimer l'amour et
la reconnaissance que nous vous portons.

En témoignage nous vous offrons ce modeste travail pour vous
remercier pour vos sacrifices et pour l'affection dont vous nous avez
toujours entourés ».

A nos pères,

« L'épaule solide, l'œil attentif, compréhensifs sont les personnes
les plus dignes de notre estime et de notre respect.

Aucune dédicace ne saurait exprimer nos sentiments, que dieu
les préserve et les leur prouve santé et longue vie ».

A nos frères,

A nos sœurs,

A nos familles,

A nos amis.

Remerciement

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

Nous tenons bien entendre à remercier particulièrement, notre cher encadrant, en l'occurrence Mr. **Rachid MERZOUGUI** Professeur à la Faculté de Technologie (Université de Tlemcen), qui par son encadrement ses précieux conseils, sa patience, sa générosité et enfin sa disponibilité ont fait notre œuvre a été largement facilité, nous ne saurions l'oublier.

Nous adressons nos remerciements pour les membres du jury :

Mr. **Mourad HADJILA** Maitre de conférences classe A à la Faculté de Technologie (Université de Tlemcen) et Mr. **Hadj ZERROUKI** Maitre de conférences classe B à la Faculté de Technologie (Université de Tlemcen) d'avoir accepté la participation à cette soutenance, qui feront l'honneur d'évaluer et d'examiner notre mémoire.

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

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

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

ملخص

الهدف من هذا المشروع هو تصميم تطبيق للهاتف الذكي موثوق وفعال وسهل، ومجهز بالوظائف اللازمة ويتناسب مع احتياجات الرياضيين.

لتنفيذ حلنا، استخدمنا عملية تطوير تعتمد على لغة النمذجة المعيارية UML في بناء مراحل وخدمات مختلفة للمنصة المقترحة. تم تنفيذ هذا التصميم في بيئة التطوير المتكاملة (Android Studio)، و java ك لغة برمجة، ونظام إدارة قاعدة بيانات Firebase.

في حين أنه قبل عشر سنوات، كان مثل هذا المشروع يتطلب الكثير من الموارد بالإضافة إلى بنية تحتية كبيرة، اليوم يمكن للهاتف الذكي البسيط أن يساهم بشكل فعال في العديد من المجالات.

الكلمات المفتاحية: مدرب ديكاتلون، أندرويد، حساب السرعات الحرارية، تطبيق رياضي للهاتف الذكي.

Résumé

Le but de ce projet est de réaliser une application mobile fiable, efficace et aisée, munie de fonctionnalités nécessaires et adéquates aux besoins des sportifs.

Pour mettre en œuvre notre solution, nous avons exploité un processus de développement basé sur la modélisation UML dans la construction de différentes étapes et services de la plateforme proposée. Cette conception est mise en œuvre sous l'environnement de développement Intégré (Android Studio), java comme langage de programmation et un système de gestion de base de données Firebase.

Alors qu'une dizaine d'années auparavant, un tel projet aurait nécessité de gros moyens ainsi qu'une grosse infrastructure, aujourd'hui un simple Smartphone peut contribuer efficacement dans de nombreux domaines.

Mots-clés : Coach décathlon, Android, Calcul des calories, Application mobile du sport.

Abstract

The goal of this project is to create a reliable, efficient and easy mobile application, equipped with the necessary functionalities and appropriate to the needs of athletes.

To implement our solution, we used a development process based on UML modeling in the construction of different stages and services of the proposed platform. This design is implemented under the Integrated development environment (Android Studio), java as programming language and a Firebase database management system.

Whereas ten years ago, such a project would have required a lot of resources as well as a big infrastructure, today a simple smartphone can effectively contribute in many areas.

Keywords : Decathlon coach, Android, Calorie calculation, Sports mobile application.

Table des Matières

DEDICACE.....	I
REMERCIEMENT	II
RESUME	III
TABLE DES MATIERES	IV
LISTE DES FIGURES.....	VII
LISTE DES TABLEAUX.....	IX
GLOSSAIRE	X
INTRODUCTION GENERALE	1
CHAPITRE I : SERVICES MOBILE	2
I.1 INTRODUCTION	3
I.2 TECHNOLOGIE MOBILE	3
I.2.1 <i>Terminaux Mobile</i>	3
I.2.1.1 Historique	3
I.2.1.2 Terminaux mobiles de Dernière Génération	3
I.2.1.3 Comparaison entre les différentes catégories des Smartphones	4
I.2.2 <i>Réseaux Mobiles</i>	5
I.2.2.1 2 ^{ème} génération	5
I.2.2.2 2,5 générations (GPRS)	5
I.2.2.3 2.75 générations (EDGE).....	5
I.2.2.4 3 ^{ème} génération	5
I.2.2.5 3G +.....	6
I.2.2.6 H+ (ou HSPA+).....	6
I.2.2.7 H+ double porteuse (ou DC-HSPA+)	6
I.2.2.8 4 ^{ème} génération	6
I.2.2.9 4 ^{ème} génération +.....	6
I.2.2.10 5 ^{ème} génération	6
I.2.3 <i>Développement Mobile</i>	7
I.2.3.1 OS mobile.....	7
I.2.3.2 Android (Google)	7
A. Architecture Android.....	8
B. Versions Android	10
C. Les Activités.....	12
D. Cycle De Vie D'une Activité	12
E. API	13
I.3 SERVICES MOBILE.....	14
I.3.1 <i>M-Commerce</i>	14
I.3.2 <i>Paiement mobile</i>	14
I.3.3 <i>M-Banking</i>	15
I.3.4 <i>M-Health</i>	15
I.3.5 <i>M-Marketing</i>	16
I.3.6 <i>Vidéo-conférence</i>	16
I.3.7 <i>Capteurs</i>	17
I.3.8 <i>Mobile Sportif</i>	17

I.4	CONCLUSION	18
II	CHAPITRE II : ANALYSE ET CONCEPTION	19
II.1	INTRODUCTION.....	20
II.2	PRESENTATION D'UML.....	20
II.2.1	<i>Définition</i>	20
II.2.2	<i>Présentation Générale des diagrammes UML</i>	20
II.3	ANALYSE.....	21
II.3.1	<i>Identifications des acteurs</i>	21
II.3.1.1	Définition d'un acteur	21
II.3.1.2	Les acteurs de notre système	21
II.3.2	<i>Diagramme de Contexte</i>	21
II.3.3	<i>Spécification des besoins</i> :	22
II.3.4	<i>Spécification des cas d'utilisation</i>	23
II.3.4.1	Définition	23
II.3.4.2	Notations de base du diagramme de cas d'utilisation.....	23
II.3.4.3	Relations entre cas d'utilisation.....	23
II.3.4.4	Diagramme de cas d'utilisation	24
II.3.4.5	Description textuelle des cas d'utilisation	25
II.4	CONCEPTION.....	27
II.4.1	<i>Diagramme de classes</i>	27
II.4.1.1	Définition	27
II.4.1.2	Caractéristiques d'une classe	28
II.4.1.3	Relations dans les diagrammes de classes.....	28
II.4.1.4	Diagramme	29
II.4.2	<i>Diagramme de Séquence</i>	31
II.4.2.1	Définition	31
II.4.2.2	Diagrammes.....	31
II.4.3	<i>Patrons de conception (Design Pattern)</i>	37
II.4.3.1	Patron Modèle-Vue-Controller « MVC »	37
II.4.3.2	Patron Client / Serveur à 3-tier	38
II.4.4	<i>Modèle Relationnel</i>	38
II.4.5	<i>Firestore</i>	39
II.4.5.1	Définition	39
II.4.5.2	Services Firestore.....	39
A.	Firestore Analytics.....	39
B.	Firestore Cloud Messaging (FCM)	39
C.	Firestore Authentification.....	39
D.	Base de données en temps réel (Realtime Database)	40
E.	Firestore Storage.....	40
F.	Firestore Test Lab.....	40
G.	Firestore Crash Reporting	40
H.	Firestore Notifications	40
II.5	CONCLUSION.....	40
III	CHAPITRE III : REALISATION	41
III.1	INTRODUCTION.....	42
III.2	PLATEFORME A REALISER	42
III.3	DESCRIPTION GENERALE DE NOTRE PROJET	42
III.3.1	<i>Établissement de la connexion</i>	42

<i>III.3.2 Traitement</i>	43
<i>III.3.3 Transmission des données</i>	43
III.3.3.1 Transmission entre deux terminaux mobiles.....	43
III.3.3.2 Mobile et serveur (base de données)	43
A. Connexion réseau sur un Serveur de la base de données	43
B. Courrier électronique.....	43
III.4 BILAN DE L'ANALYSE.....	44
<i>III.4.1 Support de Transmission</i>	44
<i>III.4.2 Communication via Internet</i>	44
<i>III.4.3 Outils de développement</i>	44
III.4.3.1 Environnement matériel.....	44
III.4.3.2 Environnement logiciel	45
<i>III.4.4 OS Mobile</i>	45
III.5 EXECUTION	45
<i>III.5.1 Application</i>	45
<i>III.5.2 Firebase</i>	56
III.6 CONCLUSION.....	57
CONCLUSION GENERALE ET PERSPECTIVES	58
REFERENCES	59

Liste des Figures

Figure I.1 : 5 générations de téléphonie mobile	5
Figure I.2 : Android linux kernel	8
Figure I.3 : Android Native C/C++ bibliothèques.....	8
Figure I.4 : Android Runtime	9
Figure I.5 : Java API framework	9
Figure I.6 : Android Applications	9
Figure I.7 : Cycle de vie d'une Activité	12
Figure I.8 : Applications M-Commerce.....	14
Figure I.9 : logo PayPal	14
Figure I.10 : BNA Bank	15
Figure I.11 : Logo Esiha	15
Figure I.12 : Logo App Alibaba.com	16
Figure I.13 : Applications vidéo meeting	16
Figure I.14 : Capteurs dans Smartphone	17
Figure I.15 : MonClubSportif	17
Figure II.1 : Diagramme de contexte.....	21
Figure II.2 : Diagramme de cas d'utilisation	24
Figure II.3 : Représentation UML d'une classe.	28
Figure II.4 : Diagramme de classe.....	29
Figure II.5 : Diagramme de séquence du cas d'utilisation « s'inscrire »	32
Figure II.6 : Diagramme de séquence du cas d'utilisation « s'authentifier »	33
Figure II.7 : Diagramme de séquence du cas d'utilisation « générer le profil »	34
Figure II.8 : Diagramme de séquence du cas d'utilisation « Paramètres de calculs »	35
Figure II.9 : Diagramme de séquence du cas d'utilisation « Visualisation des statistiques »	36
Figure II.10 : Patron MVC.....	37
Figure II.11 : Architecture « 3-tiers »	38
Figure III.1 : Plateforme à réaliser.....	43
Figure III.2 : logo de l'application.....	46
Figure III.3 : Interface page de garde	47
Figure III.4 : Interface Inscription.....	48

Figure III.5 : Email de vérification.....	48
Figure III.6 : Interface Authentification.....	49
Figure III.7 : Interface oublié mot de passe.....	50
Figure III.8 : Interface profil.....	51
Figure III.9 : Interface Informations d'utilisateur.....	51
Figure III.10 : Interface Barre de navigation.....	52
Figure III.11 : Interface d'Accueil.....	52
Figure III.12 : 1ère partie de l'activité calories	53
Figure III.13 : Liste des choix de niveaux d'activité physique.....	53
Figure III.14 : 2ème partie de l'activité calories.....	54
Figure III.15 : Liste des choix des sports.....	54
Figure III.16 : Boite de dialogue pour confirmer la réinitialisation de chronomètre.....	55
Figure III.17 : Représentation graphique des calories brulées pendant une semaine.....	55
Figure III.18 : 1ère partie de l'activité fréquence cardiaque.....	56
Figure III.19 : Représentation graphique de fréquence cardiaque pendant une semaine.....	56
Figure III.20: Firebase Realtime Database.....	57
Figure III.21 : Firebase Authentication.....	57

Liste des Tableaux

Tableau I.1 : Fonctions de chaque génération.....	6
Tableau I.2 : Versions Android	11
Tableau II.1 : Spécification des besoins.	22
Tableau II.2 : Relations entre cas d'utilisation.....	23
Tableau II.3 : Description de cas d'utilisation « S'inscrire »	25
Tableau II.4 : Description de cas d'utilisation « S'authentifier »	25
Tableau II.5 : Description de cas d'utilisation « Générer Profil »	26
Tableau II.6 : Description de cas d'utilisation « Calculs Paramètres »	26
Tableau II.7 : Description de cas d'utilisation « Visualisation des données »	27
Tableau II.8 : Relations entre classes	28
Tableau II.9 : Dictionnaire de données.....	31

GLOSSAIRE

A

API: Application Programming Interface

ADT: Android Development Tools

ASCII: American Standard Code for Information Interchange

AMOLED: Active-Matrix Organic Light-Emitting Diode

ART: Android Runtime

C

CPU: Central Processing Unit

D

DVM: Dalvik Virtual Machine

DC-HSPA: Dual Carrier- High Speed Packet Access

E

EDGE: Enhanced Data Rates for GSM Evolution

F

FCM: Firebase Cloud Messaging

G

GSM: Global System for Mobile Communications

GPRS: General Packet Radio Services

GPS: Global Positioning System

GCM: Google Clouds Messaging

H

HSPA: High Speed Packet Access

HTTP: HyperText Transfer Protocol

I

iOS: iPhone Operated System

IBM: International Business Machines

IDE: Integrated Development Environment

IP: Internet Protocol

J

JSON: JavaScript Object Notation

JVM: Java Virtual Machine

JIT: Just In Time

L

LTE: Long Term Evolution

M

MVC: Model View Controller

MMS: Multimédia Messaging Services

N

NFC: Near Field Communication

NoSQL: No Structured Query Language

O

OS: Operating System

P

POP: Post Office Protocol

R

RAM: Random Access Memory

S

SMS: Short Message Service

SDK: software development kit

SQL: Structured Query Language

SMTP: Simple Mail Transfer Protocol

T

TCP : Transport Control Protocol

V

VM: Virtual Machine

W

WAP: Wireless Application Protocol

WSP: Wireless Session Protocol

U

UMTS: Universal Mobile Telecommunications System

UML: Unified Modeling Language

Introduction Générale

Les technologies de l'information et de la communication ont été la révolution la plus importante et innovante qui a marqué ces dernières décennies. En effet, loin d'être un phénomène éphémère ces technologies nous ont apporté du confort dans notre vie quotidienne par leurs capacités à traiter l'information dans des délais raisonnables.

Cette révolution a permis l'émergence de la notion de la portabilité et de la mobilité qui permet un accès distant, instantané et un flux sans interruption d'informations. En effet, cela est symbolisé par l'apparition des différents appareils de haute technologie tels que les Smartphones et les tablettes qui sont dotées de plusieurs applications pratiques qu'elles soient gratuites ou payantes.

Dans ce contexte, notre projet s'inscrit dans un cadre général du développement et de la réalisation d'une application sportive sous Android qui aide les sportifs à pratiquer leurs activités dans des bonnes conditions, de façon fiables et efficaces. Il s'agit de surveiller leurs statistiques concernant le nombre des calories brûlées et fréquence cardiaque au cours d'une semaine enregistrés via une base de données distante.

Ce mémoire est organisé autour de trois chapitres :

- Le premier chapitre présente les services mobiles et toutes les technologies qui entourent le monde du mobile ;
- Le deuxième chapitre sera consacré à l'analyse et la conception de l'application proposée tout en exploitant la modélisation UML ;
- Le dernier chapitre décrit notre application mobile réalisée sous Android Studio tout en respectant un cahier des charges rédigé à l'avance.

Enfin, on terminera bien sûr par une conclusion générale et perspectives.

CHAPITRE I : SERVICES MOBILE

I.1 Introduction

Aujourd'hui en plus de transmettre du son et des SMS, les téléphones actuels sont capables de transmettre des MMS, de communiquer via les technologies Bluetooth, Wi-Fi..., et se connecter à Internet. Ce qui permet déjà d'envisager une multitude d'applications pour ces appareils qui n'étaient, à la base, que des outils de communication vocale.

Les réseaux de télécommunication mobiles entrent dans une nouvelle phase de leur évolution. Ils offrent actuellement, en plus de la téléphonie mobile, des systèmes de fourniture de services à grande échelle. Ainsi, une grande variété de ces services est proposée non seulement par des opérateurs réseau mais aussi par des fournisseurs ou développeurs de services. Ces services sont destinés non seulement au grand public mais également aux personnes dépendantes comme les personnes âgées, les handicapés, etc.

Ce chapitre a pour objectif de présenter le monde de la téléphonie mobile et ces services.

I.2 Technologie Mobile

I.2.1 Terminaux Mobile

I.2.1.1 Historique

Il est difficile d'imaginer le monde d'aujourd'hui sans téléphones portables ! Cependant, cet appareil n'est apparu dans nos vies qu'à la fin des années 1990, et depuis, il n'a cessé de se développer et de se transformer pour offrir de plus en plus de fonctions. Du premier téléphone d'aviation à la dernière génération de téléphones intelligents [1].

À proprement parler, le téléphone mobile n'est pas une invention, mais un objet né de la combinaison de plusieurs technologies (certaines technologies sont connues depuis la fin des années 1940). L'origine de cette innovation remonte à la technologie radio développée dans les années 1940. En effet, les téléphones portables utilisent des ondes électromagnétiques, tout comme les réseaux radio, téléviseurs et autres réseaux de communication dédiés à la police militaire ou au personnel médical. Dès 1947, une invention décisive a créé une unité hexagonale pour les téléphones Bell Labs, qui peut envoyer et recevoir des signaux dans trois directions.

L'invention du téléphone mobile est généralement attribuée au Dr Martin Cooper, directeur R&D de Motorola à l'époque. En avril 1973, il sera le premier à appeler [2].

I.2.1.2 Terminaux mobiles de Dernière Génération

Les Téléphones portables qui ne sont utilisés que pour la communication ou pour de multiples fonctions sont devenus un outil indissociable de la vie quotidienne humaine. Il existe plusieurs marques et modèles dans le marché. Ce terminal mobile est lancé en 1940 par Martin Cooper, directeur R&D de Motorola. Après de nombreuses générations de développement, sa taille, sa forme, sa capacité, sa fonction et d'autres aspects ont été de plus en plus améliorés [1].

Téléphone intelligent (également appelé téléphone intelligent au Québec, téléphone multifonction ou Ordiphone ou téléphone multifonction en France) est un type qui a généralement un écran tactile, un appareil photo numérique, une fonction d'assistant numérique personnel et certaines fonctions d'ordinateur portable. Téléphoner. La saisie des données se fait généralement à l'aide d'un écran tactile ou, plus rarement, d'un clavier ou d'un stylet. Selon le principe de l'ordinateur, en raison du système d'exploitation spécialement conçu pour les appareils mobiles, il peut exécuter divers logiciels / applications.

Ainsi, en plus des fonctions régulières du téléphone portable, il fournit également d'autres fonctions, telles que : agenda, télévision, calendrier, navigation Web, vérifier et envoyer des e-mails, emplacement géographique, répondre au téléphone / enregistreur, calculatrice, boussole, accéléromètre, gyroscope, messagerie vocale visuelle, la cartographie numérique, etc. La plupart des appareils tels que les cartes numériques bénéficient de la reconnaissance vocale et de la synthèse vocale.

Vous pouvez personnaliser votre smartphone en installant d'autres applications (telles que des jeux ou des utilitaires) via différents magasins d'applications en ligne pour chaque système d'exploitation. Une connexion Internet doit être établie via un réseau de téléphonie mobile ou un réseau Wi-Fi afin de pouvoir réaliser son potentiel [3].

I.2.1.3 Comparaison entre les différentes catégories des Smartphones

Samsung :

Cette marque a été créée à Séoul, en Corée du Sud en 1938. Ce n'est qu'en 1969 que la marque a pénétré le marché de l'électronique en vendant son premier téléviseur. Depuis lors, la société a vendu une variété de produits électroniques, des réfrigérateurs aux machines à laver, des fours à micro-ondes, des ordinateurs portables et, bien sûr, des téléphones.

Elle est caractérisée par un design en verre et métal, une recharge sans fil native, un paiement sans contact Samsung Pay, un déclenchement rapide de l'appareil photo, une surcouche TouchWiz épurée et un lecteur d'empreinte digitale [4].

IPhone :

Si techniquement, iPhone n'est pas le premier téléphone intelligent, alors c'est un produit qui popularise les téléphones intelligents. En proposant un écran multi-touch, une plateforme de téléchargement d'applications et un iPod intégré, l'iPhone va révolutionner le marché de la téléphonie mobile. En conséquence, la marque californienne est devenue un véritable pionnier dans ce domaine et a commencé à se développer rapidement pour devenir l'une des entreprises les plus puissantes au monde [4].

Oppo :

Présent sur le marché de la téléphonie mobile depuis 2008, c'est la série FIND sortie en 2011 qui marque un véritable tournant dans la position d'Oppo dans le secteur de la téléphonie mobile. Aujourd'hui, la marque chinoise occupe la quatrième place du classement mondial des fabricants de smartphones. Cependant, il reste peu connu en France malgré la qualité de ces appareils. La technologie Oppo est caractérisé par un écran AMOLED, double caméra, caméra frontale soignée, grande mémoire interne, bonne batterie, système d'exploitation ColorOS [5] [4].

Xiaomi :

Xiaomi, qui signifie « petit grain de riz » en chinois, est souvent appelée « pomme chinoise ». Fondée en 2010, la société a commencé par développer MIUI, le système d'exploitation désormais installé sur la majorité des smartphones de la marque. Les points forts de cette technologie sont : Grand écran AMOLED, belle caméra frontale, grande mémoire interne, une très bonne batterie, technologie NFC utilisée, processeur Snapdragon avec une grande RAM, Ultra performant, avec une interface qui rappelle iOS qui est un sous Android [6] [4].

I.2.2 Réseaux Mobiles

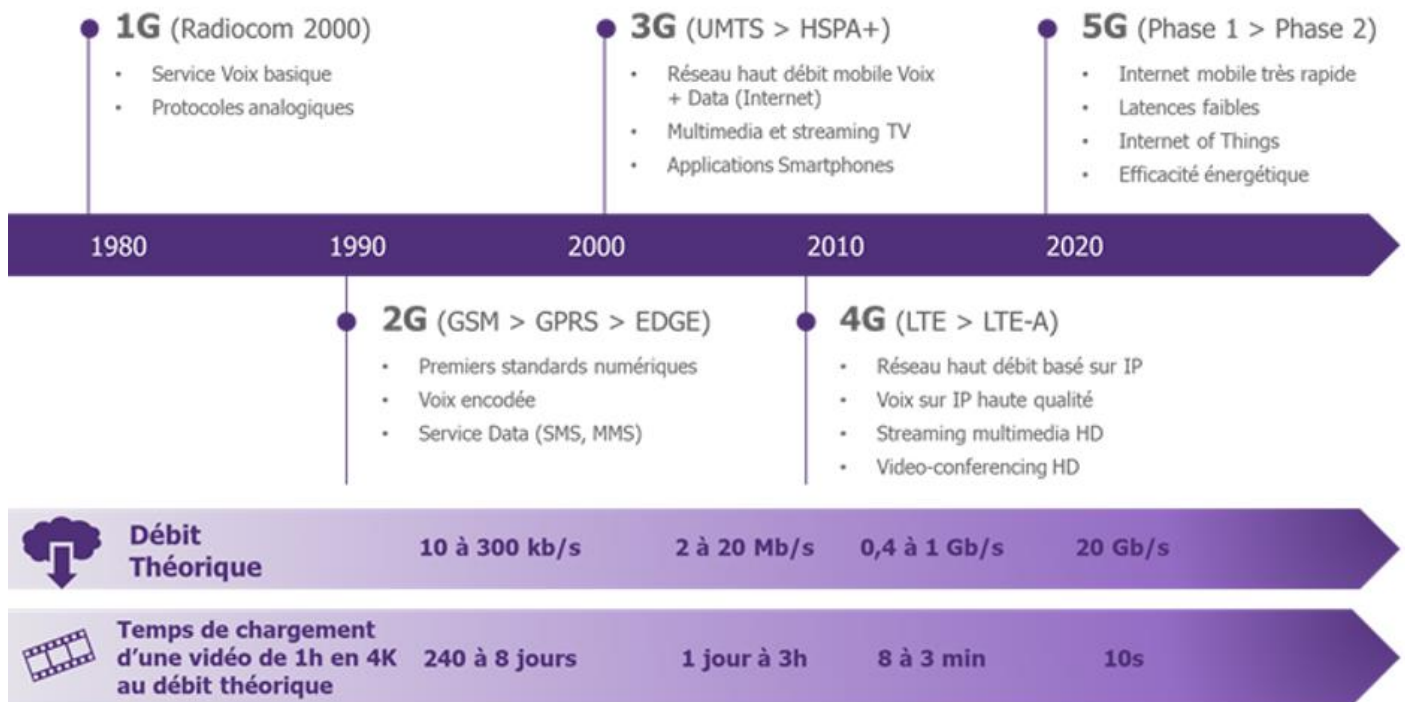


Figure I.1 : 5 générations de téléphonie mobile [7]

Les Réseaux mobiles sont construits selon une architecture cellulaire qui permet d'utiliser plusieurs fois les mêmes fréquences dans un territoire. Depuis 30 ans, plusieurs générations de technologies se sont succédées permettant de passer d'un service de communication téléphonique vocale et de transmission de SMS, aux services data, internet mobile, très haut débit et objets connectés : GSM, 3G, 4G / LTE (Evolution à long terme) et 5G [8].

I.2.2.1 2^{ème} génération

Ce réseau, également appelé GSM pour Global System for Mobile Communications, permet uniquement la communication vocale. Cette norme numérique a été créée en 1982, donc il y a près de quarante ans [9].

I.2.2.2 2,5 générations (GPRS)

Signifiant *General Packet Radio Service*, ce réseau est également appelé 2.5G car il est situé entre la génération 2G et la 3G. Il est possible, avec celui-ci, d'échanger des données (SMS, données, appels...) avec une vitesse théorique maximale de 171,2 Kb / s par seconde [9].

I.2.2.3 2.75 générations (EDGE)

Plus communément appelé **EDGE** pour *Enhanced Data rates for GSM Evolution*, la vitesse ici peut atteindre 384 Kb / s par seconde. Cette norme est appelée « pré-3G » [9].

I.2.2.4 3^{ème} génération

Utilisant le standard **UMTS** (*Universal Mobile Telecommunications System*) peut fournir un accès Internet haut débit de 144 kbit/s à 2 Mbit/s avec un débit moyen de plusieurs centaines de Kbit/s. Cela vous permet notamment de télécharger des données, des applications ou des jeux plus rapidement, envoyez la vidéo, regardez des vidéos en streaming sur YouTube, Dailymotion, etc. organisez une vidéoconférence, accédez à la télévision mobile et bénéficiez du GPS.

Les réseaux 3G fonctionnent sur les bandes de fréquences 900 MHz et 2100 MHz [9].

Veillez noter que la 3G a connu les trois développements continus suivants :

I.2.2.5 3G +

Est également connue sous le nom de "H" du **HSPA** (*High Speed Packet Access*), et le trafic est compris entre 300 kbit/s et 14,4 Mbit/s, avec une moyenne de 3,6 Mbit/s. Bref, la 3G+ est 7 fois plus rapide que la 3G [9].

I.2.2.6 H+ (ou HSPA+)

La vitesse moyenne est de 5 Mbit/s et la vitesse théorique maximale est de 21 Mbit/s [9].

I.2.2.7 H+ double porteuse (ou DC-HSPA+)

La vitesse moyenne est de 10 Mbit/s et la vitesse maximale est de 42 Mbit/s. Bref, la 3G et son développement ont permis de démocratiser complètement l'utilisation de l'Internet mobile [9].

I.2.2.8 4^{ème} génération

Offre aux utilisateurs la possibilité d'accéder à Internet à très haut débit. Entre autres, il permet un Transfert rapide des fichiers volumineux (photos, musique, vidéos, etc.), de regarder la vidéo haute définition, de passer des appels vidéo de meilleure qualité. La 4G est basée sur la norme **LTE** (*Long Term Evolution*), et sa vitesse théorique atteint 150 Mbit/s [9].

I.2.2.9 4^{ème} génération +

A été lancé en 2014, et sa vitesse de développement a même dépassé celle des réseaux 4G, creusant encore l'écart de vitesse de connexion avec les réseaux 3G. Le réseau 4G + offre une vitesse moyenne de 200 à 300 Mbit/s. Dans les meilleures conditions, l'utilisation du réseau 4G + peut même atteindre une vitesse de 1 Gb/s. Le réseau 4G + est une véritable révolution dans le domaine des réseaux mobiles, il se connecte à plusieurs bandes de fréquences en même temps, vous permettant ainsi de vous connecter à Internet en continu [9]. En 2019, presque toutes les grandes villes seront desservies via le réseau 4G +. Avant l'arrivée des réseaux 5G en 2020, tous les opérateurs fourniront ces derniers.

I.2.2.10 5^{ème} génération

5G est le réseau mobile de cinquième génération. Il s'agit d'une nouvelle norme mondiale sans fil suivant les réseaux 1G, 2G, 3G et 4G. La 5G réalise un nouveau type de réseau conçu pour connecter presque tout (y compris les machines, les objets et les équipements) entre eux. La technologie 5G est basée sur 8 exigences techniques :

- ✓ Débit de données jusqu'à 10 Gbit / s -> 10 à 100 fois celui des réseaux 4G et 4,5G
- ✓ Délai de 1 milliseconde.
- ✓ Bande passante par unité de surface a été multipliée par 1000.
- ✓ Jusqu'à 100 fois plus d'appareils connectés par unité de surface (par rapport à la 4G LTE).
- ✓ Disponibilité est de 99,999%.
- ✓ 100% de couverture.
- ✓ Réduisez 90% de la consommation d'énergie du réseau.
- ✓ Les appareils IoT à faible consommation ont une autonomie de batterie allant jusqu'à 10 ans [10].

Sigle	Vitesse	Fonction
2G/E	Jusqu'à 0,34 Mbit/s	Appels, SMS, internet limité
3G	Jusqu'à 2 Mbit/s	Regarder des vidéos, télécharger des applications, GPS, appels vidéo
3G+/H	Jusqu'à 14 Mbit/s	Version plus rapide de la 3G
4G LTE	Jusqu'à 150 Mbit/s	Photos ou vidéos lourdes, vidéos avec une meilleure qualité d'image
4G+ LTEA	Jusqu'à 300 Mbit/s	Version plus rapide de la 4G
5G	Jusqu'à 2,94 Gbit/s	10 fois plus rapide que la 4G, connexion entre objets électroniques

Tableau I.1 : Fonctions de chaque génération

I.2.3 Développement Mobile

I.2.3.1 OS mobile

Le Système d'exploitation mobile (OS) peut être défini comme un logiciel permettant le fonctionnement des téléphones intelligents ou mobiles de base. En tant que tels, ils permettent aux utilisateurs de passer des appels, de naviguer sur le téléphone entre toutes les pièces, de télécharger des applications et même de configurer et de personnaliser leurs smartphones [11].

Il existe environ 20 systèmes d'exploitation sur le marché, dont certains sont directement développés par les fabricants de téléphones mobiles pour eux-mêmes, d'autres forment un groupe d'entreprises pour partager les coûts et réaliser des synergies, tandis que d'autres veulent simplement se concentrer sur les opérations de développement de système.

Plus généralement, on peut citer l'existence de quatre systèmes d'exploitation sur le marché des systèmes d'exploitation. Il est recommandé de les rechercher en fonction de leur importance : Android (*Google*), iOS (*Apple Inc.*), Windows Phone (*Microsoft*) et Blackberry OS (*Rim*) [11].

Dans le cadre de ce projet de fin d'étude nous proposons une application mobile développée sous Android, c'est pour cette raison que nous allons présenter ce système dans les sections suivantes.

I.2.3.2 Android (Google)

Android a été créé par une société, nommée android Inc. à Palo Alto en Californie, aux États-Unis en 2003 dont le fondateur était Andy Rubin. L'intention initiale de la société était de développer un système d'exploitation avancé pour les appareils photo numériques, mais elle s'est ensuite rendu compte que le marché pour de tels appareils n'était pas assez important et a donc détourné son attention vers la production d'un système d'exploitation pour téléphone intelligent. Android a ensuite été acquis par Google en 2005 et a dévoilé sa distribution en 2007 avec la formation de Open Handset Alliance (OHA) dirigée par Google. Le système d'exploitation mobile Android est basé sur Linux Kernel et est développé par Google. Le système d'exploitation Android possède sa propre machine virtuelle

appelée DVM, qui est utilisée pour exécuter l'application Android. L'une des raisons du succès du système d'exploitation de Google est l'amélioration constante de ses nombreuses versions, chaque nouvelle version offrant des fonctionnalités plus avancées et un accès plus rapide à Internet. Une autre raison de la popularité d'Android est sa forte collaboration avec les fabricants d'appareils mobiles [12].

A. Architecture Android

Linux Kernel : Les services de base (notamment les pilotes matériels, la gestion des processus et de la mémoire, la sécurité, le réseau et la gestion de l'alimentation) sont gérés par un linux Kernel (la version spécifique du noyau (Kernel) dépend de la version de la plate-forme Android et de la plate-forme matérielle) [13].

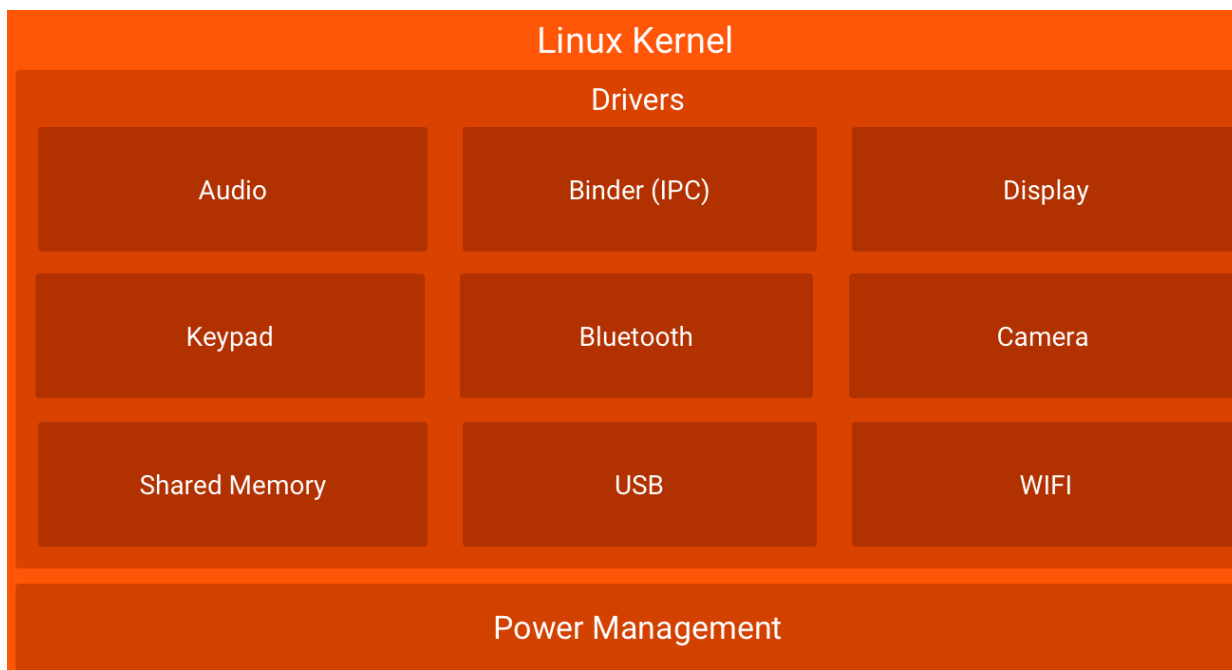


Figure I.2 : *Android linux kernel* [14]

Android Bibliothèques : Android comprend un ensemble de bibliothèques C et C++ utilisé par différents composants du système Android. Les développeurs utilisent ces bibliothèques par le biais du framework d'application Android. Parfois, cette couche est appelée "Native Layer" car le code est écrit en C et C++. Et optimisé pour le matériel, par opposition aux applications et au framework Android, où il est écrit en Java [15].

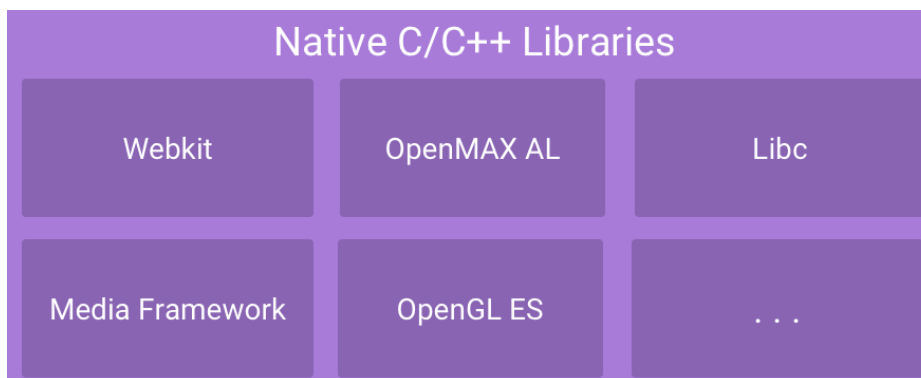


Figure I.3 : *Android Native C/C++ bibliothèques* [14]

Android Runtime : L'Android Runtime (ART) est la nouvelle JVM qui a été introduite à titre expérimental dans Android 4.4, et qui est devenue plus tard, avec Android 5.0, le moteur d'exécution officiel des applications Android. Avant Android 5.0, les applications Android s'exécutaient au-dessus de la machine virtuelle (VM) Dalvik.

Par rapport à l'approche juste-in-time (JIT) de la VM Dalvik, ART repose sur la compilation ahead-of-time (AOT) pour traduire le bytecode en code machine pendant l'installation de l'application. Cela permet au code de l'application d'être ensuite exécuté directement par l'environnement runtime de l'appareil [16].



Figure I.4 : Android Runtime [14]

Android Framework : Android Framework se compose de l'environnement de conteneur et de l'infrastructure pour installer et exécuter les applications de manière sécurisée et fiable. Il comprend les différents services système dont les applications peuvent bénéficier et fournit la mise en œuvre des namespaces : android.*, java.* et dalvik.* [17].

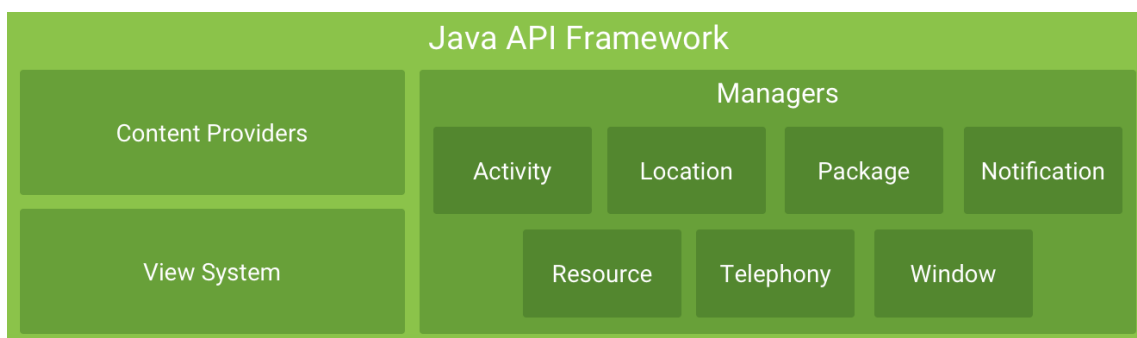


Figure I.5 : Java API framework [14]

Application : Par défaut, Android est livré avec un riche ensemble d'applications, dont le navigateur, le programme SMS, le calendrier, le client de messagerie, les cartes, le gestionnaire de contacts, un lecteur audio, etc. Ces applications sont écrites en langage de programmation Java.

Google Play (la place de marché pour Android) fournit des alternatives à ces applications, si l'utilisateur le souhaite. Android ne fait pas de différence entre les applications écrites par les utilisateurs ou fournies par le système d'exploitation, par exemple, l'application de navigation. Un utilisateur peut télécharger Firefox, Opera ou d'autres navigateurs, et Android les traitera de la même manière que le navigateur intégré. Les utilisateurs peuvent remplacer les applications par défaut par les applications de leur choix [15].

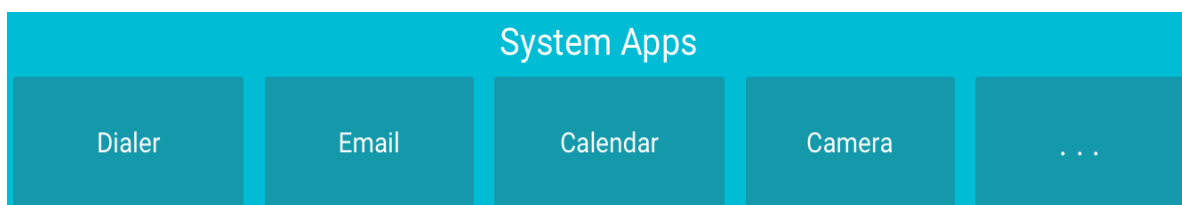


Figure I.6 : Android Applications [14]

B. Versions Android

Versions	Évolution
Android 1.0 G1 (2008)	Au moment de sa sortie, Android 1.0 fournit 35 applications via l'Android Market. Son Google Maps utilise le GPS et le Wi-Fi du téléphone et dispose d'un navigateur Android intégré.
Android 1.5 Cupcake (2009)	Cupcake est la première mise à jour majeure d'Android, introduisant des widgets pour l'écran d'accueil, le clavier virtuel, l'enregistrement vidéo pour la caméra et les options de copier / coller pour le navigateur Web.
Android 2.0 Eclair (2009)	Grâce à Android Eclair, les utilisateurs de Microsoft peuvent profiter de la prise en charge d'Exchange, de plusieurs comptes Google, des moteurs de recherche dans les messages texte, les gestes multi-touch et les améliorations de l'appareil photo grâce au flash et au zoom numérique.
Android 2.2 Froyo (2010)	Froyo apporte Flash Player 10.1, qui permet aux téléphones intelligents de lire du contenu vidéo et audio en streaming. Le flash de l'appareil photo peut être utilisé pour la vidéo, étend la compatibilité Bluetooth et vous pouvez transformer votre téléphone en point d'accès Wi-Fi.
Android 2.3 Gingerbread (2011)	Introduction de la technologie NFC (Near Field Communication), qui permet au terminal de se connecter à d'autres appareils compatibles à proximité. Les appels vidéo peuvent désormais être passés via la caméra frontale et le système d'exploitation a hérité du gestionnaire de téléchargement.
Android 3.0 Honeycomb (2011)	Cette mise à jour est la première mise à jour qui s'applique uniquement aux tablettes Android. Il introduit la prise en charge des graphiques 3D, du chat vidéo avec Google Talk, du partage réseau via Bluetooth et du mode plein écran dans la bibliothèque de photos.
Android 4.0 Ice Cream Sandwich (2011)	Ice Cream Sandwich coordonne les versions smartphone et tablette du système d'exploitation en ajoutant des processus, une reconnaissance faciale pour déverrouiller les téléphones mobiles et des SMS préprogrammés pour réduire l'effet en direct des appels et des enregistrements vidéo.
Android 4.1 Jelly Bean (2012)	Grâce au "Butter Project", Jelly Bean s'est amélioré en termes de performances et de fluidité. Les notifications sont plus riches, le navigateur Chrome est utilisé par défaut, les widgets peuvent être redimensionnés, et Google Now est pré-installé.

Android 4.4 KitKat (2013)	KitKat ajoute des emoji au clavier Google, utilise Google Cloud Print pour l'impression à distance et utilise la RAM plus rarement pour fonctionner sur les smartphones d'entrée de gamme.
Android 5.0 Lollipop (2014)	Grâce à l'introduction de la « conception matérielle » et de son interface plate, l'esthétique s'est considérablement développée. Les notifications sont désormais affichées sous forme de bannières d'écran de verrouillage et d'alertes contextuelles.
Android 6.0 Marshmallow (2015)	Introduction d'un mode ze qui peut économiser la batterie, prise en charge native des lecteurs d'empreintes digitales, USB-C et Ultra HD pour les applications.
Android 7.0 Nougat (2016)	En utilisant Nougat, vous pouvez enfin fermer toutes les applications dans l'aperçu en un seul geste. Cette mise à jour modifie le teint de la peau des emoji, enrichit le menu des paramètres rapides et le système d'exploitation peut être utilisé avec la plate-forme de réalité virtuelle Google Daydream.
Android 8.0 Oreo (2017)	Android Oreo a amélioré les capacités multitâches, amélioré le copier-coller, la sécurité et la gestion de la batterie.
Android 9.0 Pie (2018)	La mise à jour Android 9.0 rend les smartphones Android plus rapides et plus écoénergétiques. Pour atteindre cet objectif, Android utilise l'intelligence artificielle pour proposer des applications et des raccourcis qui répondent aux besoins des utilisateurs, en chargeant des aperçus des informations qu'ils recherchent, sans avoir à ouvrir une application ou une page Web.
Android 10 Q (2019)	Lors de l'utilisation d'Android 10, Google a abandonné le nom du dessert pour sa mise à jour système. Désormais, il faut se contenter du graphisme "simple" de la dixième édition. Cette mise à jour a été publiée en septembre 2019 et se concentre sur les fonctionnalités de confidentialité et de sécurité.
Android 11 (2020)	L'édition 2020 se concentrera toujours sur les fonctionnalités de confidentialité et de sécurité. Le contrôle des autorisations est plus poussé et un nouveau système de capture d'écran est apparu, tout comme le système de bulles de notification. Nous avons rassemblé toutes les informations importantes sur Android 11 dans cet article.
Android 12 (2021)	Début 2021, il y a encore relativement peu d'informations sur Android 12. Cependant, le géant américain devrait apporter des changements visuels majeurs en ajoutant de nouveaux widgets. Les fonctionnalités conçues pour protéger la confidentialité des utilisateurs deviendront également le cœur des nouvelles fonctionnalités.

Tableau I.2 : Versions Android [18]

C. Les Activités

Google décrit une activité comme un composant d'application qui fournit un écran avec lequel un utilisateur d'interagir avec lui. Une activité peut servir à afficher des informations, à appeler quelqu'un ou même à jouer à un jeu. Une application Android typique se compose de plusieurs activités et permet à l'utilisateur de naviguer entre elles [19].

D. Cycle De Vie D'une Activité

Par défaut, chaque application Android s'exécute dans un processus distinct. Android gère les ressources disponibles sur l'appareil et peut fermer les applications pour libérer des ressources si nécessaire (à l'exclusion des applications en cours d'exécution).

Le choix de l'application à fermer dépend en grande partie de l'état du processus dans lequel elle se trouve. Si Android doit choisir entre deux applications dans le même état, il choisira l'application qui est dans cet état depuis le plus longtemps.

Cycle de vie d'une activité est très complexe et sa compréhension est cruciale pour le développement Android. La figure suivante résume le cycle de vie d'une activité.

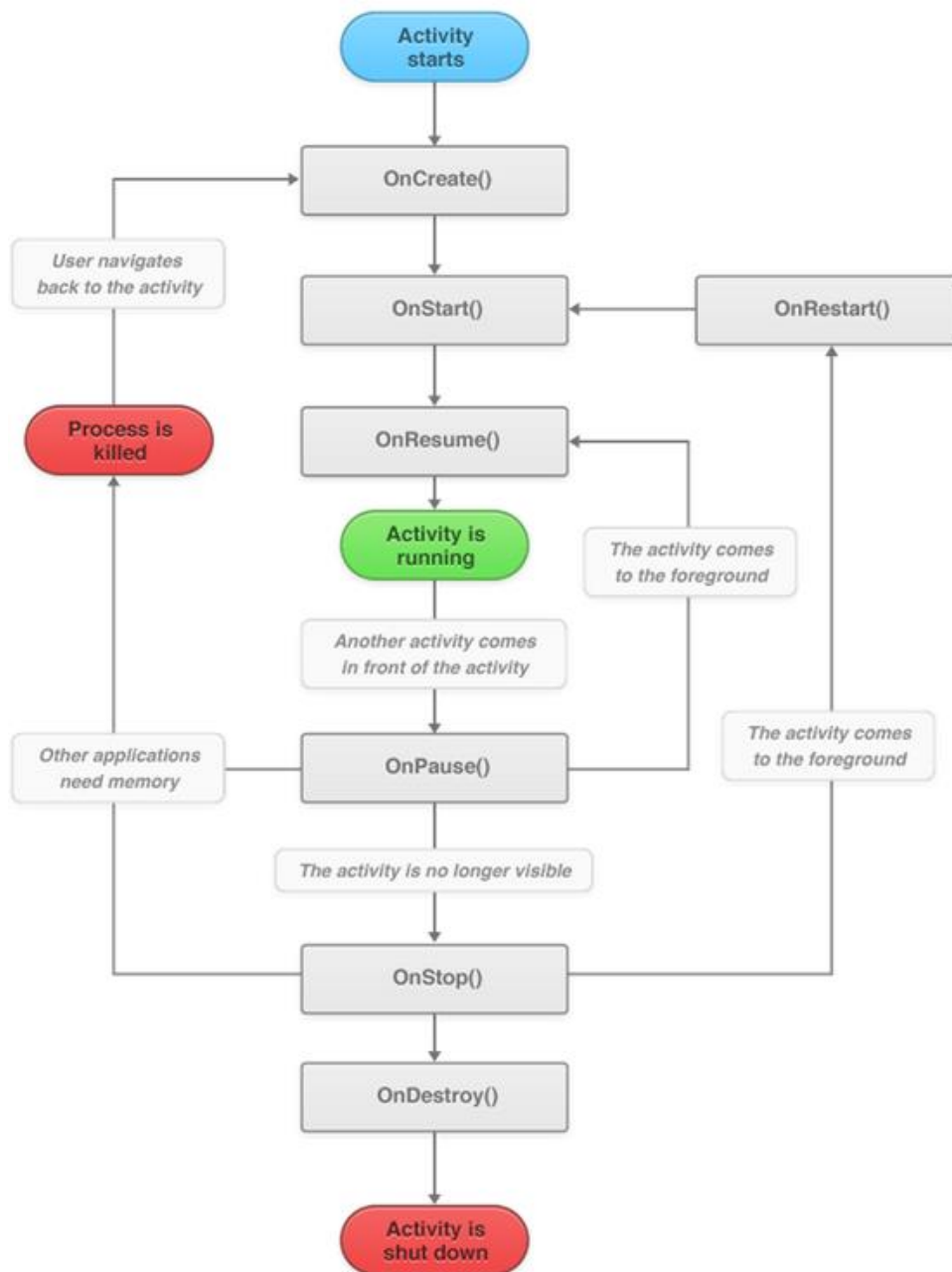


Figure I.7 : Cycle de vie d'une Activité [20]

Voici une explication rapide des méthodes fournies par Android, pour gérer les phases du cycle de vie [21]:

- **onCreate** : Cette méthode est exécutée lorsque l'activité est créée.

Ici, nous préparons tout pour l'application, y compris l'interface utilisateur (comme l'appel de la méthode setContentView), les graphiques et le son.

- **onStart** : Cette méthode est exécutée lorsque l'application est en phase de démarrage.

- **onResume** : Cette méthode s'exécute après **onStart** mais peut également être saisie, peut-être plus logiquement, si notre activité est reprise après avoir été précédemment mise en pause. Nous pouvons recharger des données utilisateur précédemment enregistrées (comme une note importante) depuis lorsque l'application a été interrompue, peut-être par un appel téléphonique ou par l'utilisateur.

- **onPause** : Vous commencez probablement à comprendre ces méthodes. Cela se produit lorsque notre application se met en pause. Ici, nous pouvons sauvegarder des données non sauvegardées (telles que la note) qui pourraient être rechargées dans **onResume**. Les activités passent toujours en état de pause lorsqu'un autre élément d'interface utilisateur est affiché au-dessus de l'activité en cours (par exemple, une boîte de dialogue contextuelle), ou lorsque l'activité est sur le point d'être arrêtée (par exemple, lorsque l'utilisateur navigue vers une autre activité).

- **onStop** : Ceci concerne la phase d'arrêt. C'est ici que nous pouvons défaire tout ce que nous avons fait dans **onCreate**, comme libérer les ressources du système ou l'écriture d'informations dans une base de données. Si nous arrivons ici, nous allons probablement être détruits un jour ou l'autre.

- **onDestroy** : C'est le moment où notre activité est finalement détruite. Il n'y a pas de retour en arrière à cette phase. C'est notre dernière chance de démanteler notre application de manière ordonnée.

E. API

API signifie Application Programming Interface (interface de programmation d'applications). Une API peut fournir un crochet pour les collègues partenaires ou des développeurs tiers d'accéder à des données et à des services pour créer rapidement des applications. Les API de Twitter et de Facebook en sont des exemples célèbres.

Il existe des API ouvertes à tous les développeurs, des API ouvertes uniquement aux partenaires, et les API utilisées en interne pour améliorer le fonctionnement de l'entreprise et faciliter la collaboration entre les équipes.

API est donc essentiellement un contrat. Une fois qu'un tel contrat est en place, les développeurs sont incités à utiliser l'API car ils savent qu'ils peuvent s'y fier. Le contrat renforce la confiance, ce qui augmente l'utilisation. Le contrat rend également la connexion entre le fournisseur et le consommateur beaucoup plus efficace puisque les interfaces sont documentées, cohérentes, et prévisibles [22].

1. Principe de fonctionnement de l'API

L'un des concepts derrière cette technologie numérique est la divulgation des données. Qu'il s'agisse d'un article ou d'une page, le contenu du site Web est accessible via le point de terminaison. En effet, en accédant à une certaine URL, l'utilisateur peut récupérer des données du site placées au format de données JSON. L'API permet également aux développeurs de créer des applications plus rapidement. Ils leur évitent d'avoir à « réinventer » ou « re-développer » et de passer du temps à créer des fonctionnalités existantes. Cela peut améliorer les performances et offre de grandes possibilités de développement.

2. Types des APIs

Il existe quatre API principales qui sont souvent utilisées [23] :

- Les API ouvertes, qui sont accessibles au public et peuvent être utilisées par tous.
- Les API partenaires qui sont conçues sur mesure par les entreprises pour offrir un accès aux partenaires commerciaux afin qu'ils puissent réserver ou acheter des articles spécifiques, tels que des billets ou des bons d'achat.

- Les API privées (ou internes) ne sont pas destinées à la consommation publique mais utilisées en interne.
- APIs composites : Elles combinent diverses API de données et de services. Ses principales capacités accélèrent le cours de la mise en œuvre, ainsi que le fonctionnement des visualisateurs à l'intérieur des frontières du web.

I.3 Services Mobile

Les Solutions mobiles évoluent rapidement. D'abord offertes sur des applications natives d'Android, elles se sont transformées en services numériques indépendants des appareils, soutenus par des technologies de pointe. La gamme complète de capacités mobiles aide nos clients à suivre ce rythme afin de pouvoir offrir une expérience utilisateur optimale. On peut citer comme solutions ou services :

I.3.1 M-Commerce

Le terme "m-commerce" a été défini par un certain nombre de chercheurs, avec de subtiles variations dans ces définitions au fil des ans. Durlacher (2000) a défini le m-commerce comme "toute transaction ayant une valeur monétaire et effectuée via un réseau de télécommunications mobiles". Le m-commerce est considéré par certains comme un sous-ensemble du commerce électronique, et est défini comme " une extension naturelle du commerce électronique qui permet aux utilisateurs d'interagir avec d'autres utilisateurs ou entreprises en mode sans fil, à tout moment et en tout lieu " (Coursaris, 2003) [24].



Figure I.8 : Applications M-Commerce

I.3.2 Paiement mobile

Paiement mobile est effectué avec un instrument de paiement mobile tel qu'une carte de crédit mobile ou un portefeuille mobile. En plus les instruments de paiement purement mobiles, la plupart électroniques et de nombreux instruments de paiement physiques ont été mobilisés. En outre, les paiements mobiles, comme tous les autres paiements, se répartissent en deux grandes catégories : les paiements pour les achats quotidiens, et les paiements de factures (paiements crédités). Pour les achats, les paiements mobiles complètent ou concurrencent l'argent liquide, les chèques, les cartes de crédit et les cartes de débit. Pour les factures, les paiements mobiles donnent généralement accès à des instruments de paiement en compte, tels que les transferts d'argent, les paiements bancaires par Internet, les affectations de débit direct ou l'acceptation des factures électroniques [25].



Figure I.9 : logo PayPal [26]

I.3.3 M-Banking

La banque mobile a évolué lorsque les institutions financières (notamment les banques) ont commencé à proposer leurs opérations en ligne. Cette nouvelle approche présente de nombreux avantages pour les prestataires de services, comme le fait de permettre à la banque de s'affranchir des restrictions liées à des zones géographiques spécifiques, afin d'opérer sur de nouveaux marchés, d'élargir les options de livraison, d'améliorer les performances, d'accroître la fidélité des clients et de réduire le coût des opérations. De plus, les clients bénéficient d'un service personnalisé et d'une réduction du temps d'attente pour un service plus pratique [27] [28] [29] [30].

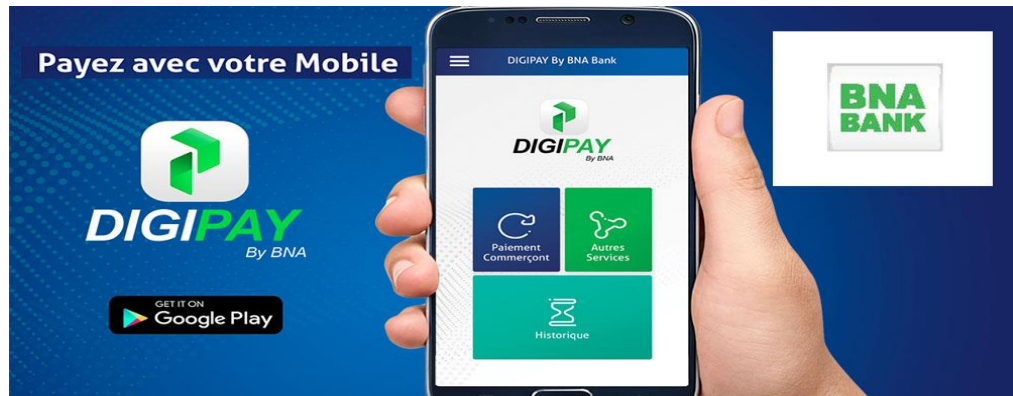


Figure I.10 : BNA Bank [31]

I.3.4 M-Health

L'utilisation créative des nouvelles technologies mobiles d'information et de détection en matière de santé (mHealth) a le potentiel de réduire le coût des soins de santé et d'améliorer la recherche et les résultats en matière de santé. Ces technologies peuvent permettre une surveillance continue de la santé, tant au niveau individuel qu'au niveau de la population, encourager les comportements sains pour prévenir ou réduire les problèmes de santé, soutenir l'autogestion des maladies chroniques par les patients, améliorer les connaissances des prestataires, réduire le nombre de visites médicales et fournir des interventions personnalisées, localisées et à la demande, d'une manière inimaginable auparavant [32].



Figure I.11 : Logo Esiha [33]

I.3.5 M-Marketing

Marketing mobile combine toutes les techniques de marketing utilisées sur les appareils mobiles (en particulier les smartphones). De manière générale, le marketing mobile peut également être défini comme l'ensemble des mesures de marketing prises vers des destinations personnelles en déplacement. Il convient de souligner que les tablettes électroniques sont également généralement considérées comme des terminaux d'accès mobiles et que la plupart de l'utilisation des tablettes électroniques ne se fait pas en déplacement, mais à domicile.



Figure I.12: Logo App Alibaba.com [34]

Principales technologies et applications du marketing mobile sont :

- Utiliser des sites Web et des applications mobiles optimisés pour les appareils mobiles
- Publicité sur les sites Web et les applications mobiles
- Fonction de localisation du point de vente -Lien commercial affiché sur le smartphone [35].

I.3.6 Vidéo-conférence

Logiciel de visioconférence, parfois appelé vidéo-conférence, conférence Web ou « Visio » en abrégé, est un outil informatique qui peut établir une connexion en temps réel entre deux ou plusieurs parties distantes via Internet pour simuler une réunion en face à face.



Figure I.13 : Applications vidéo meeting [36]

Visioconférence rassemble des personnes généralement incapables de se faire face. Dans sa forme la plus simple, la "vidéo" transmet des images fixes et du texte en plus d'une voix de mauvaise qualité. Mais la plupart des outils actuels permettent la transmission de flux vidéo et de voix haute définition [37].

I.3.7 Capteurs

La Plupart des appareils modernes ne sont pas seulement des outils de communication ou de navigation sur Internet. Ils ont des capacités sensorielles, qui sont obtenues par leurs capteurs. Ces capteurs nous fournissent des informations brutes très précises, qui peuvent être interprétées pour comprendre les transitions d'état subies par le terminal. Par exemple, il existe des accéléromètres, des gyroscopes, des capteurs de champ magnétique, etc. Tous ces capteurs nous permettent d'explorer de nouvelles voies et d'offrir aux utilisateurs de nouvelles possibilités. Par conséquent, nous verrons dans ce chapitre comment surveiller ces capteurs et comment les gérer. Par conséquent, nous verrons les informations fournies par les capteurs et comment déduire ce que l'utilisateur a fait à l'appareil



Figure I.14 : Capteurs dans Smartphone [38]

Capteurs peuvent être divisés en trois catégories : capteurs de mouvement, capteurs de position, capteurs environnementaux [39].

- Actuellement, le mobile est introduit dans pratiquement tous les domaines y compris le domaine sportif.

I.3.8 Mobile Sportif

Vous souhaitez prendre soin de votre corps, mais vous manquez de temps et / ou d'argent ? Grâce à la technologie numérique, les applications mobiles peuvent désormais être téléchargées sur les smartphones sur Android et iOS afin que vous puissiez vous entraîner à tout moment. Transformez votre téléphone en coach personnalisé et bénéficiez du suivi progressif de votre condition physique.

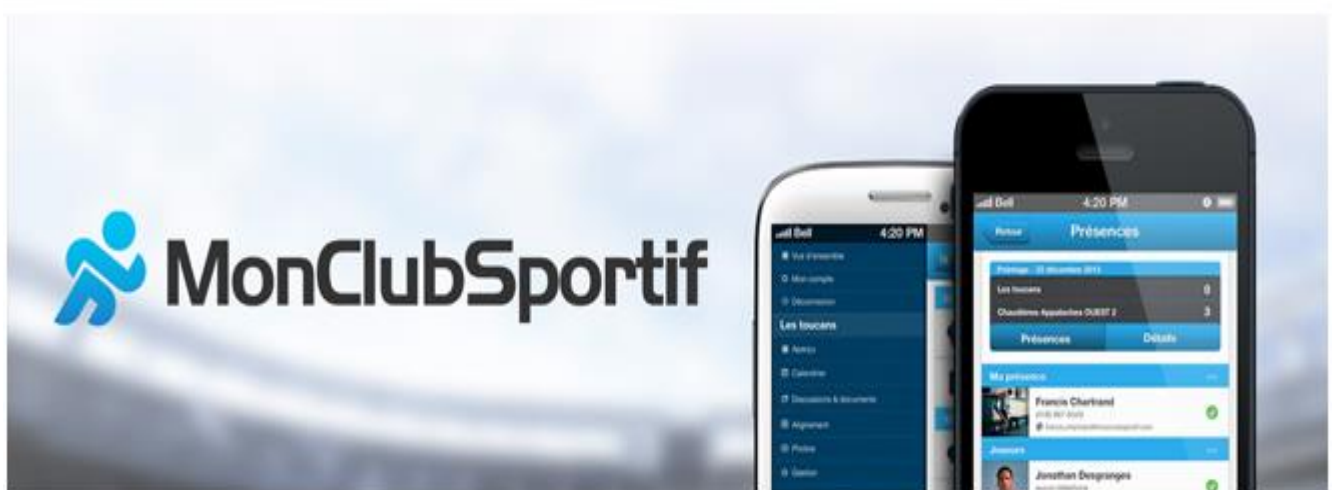


Figure I.15 : MonClubSportif [40]

Applications sportives peuvent être définies comme des applications téléchargeables sur les smartphones, permettant aux utilisateurs de mesurer leurs activités sportives ou de se fixer des objectifs et c'est ce qu'on cherche à développer via ce projet. Ils peuvent être utilisés seuls, ou vous pouvez transporter votre smartphone avec vous pendant l'exercice, ou vous pouvez utiliser des bracelets connectés tels que les bracelets Fitbit qui peuvent enregistrer les données souhaitées [41].

I.4 Conclusion

Dans ce chapitre, nous avons présenté tous qu'il entoure la technologie mobile et ses services dont le but de situer les objectifs visés par notre projet qui est le domaine sportif.

Le chapitre suivant sera consacré à la conception et la modélisation de notre application.

CHAPITRE II : ANALYSE ET CONCEPTION

II.1 Introduction

L'utilisation de la modélisation est la pratique de base du développement de logiciels. Son rôle est d'identifier les problèmes, trouver des solutions, formuler des solutions et enfin se préparer à l'action.

Modèle est une représentation abstraite du système afin qu'il puisse être étudié et documenté. C'est le principal outil de communication entre les différentes parties prenantes du projet.

De plus, alors que le système devient de plus en plus complexe, sa compréhension et son contrôle global ont dépassé la capacité d'une seule personne. L'établissement d'un modèle abstrait permet de résoudre ce problème.

Ce chapitre présentera spécifiquement la conception de notre application. Nous introduisons tout d'abord le langage UML et ses différents schémas. Ensuite, nous présenterons les types de diagrammes pour modéliser réellement le projet.

II.2 Présentation d'UML

II.2.1 Définition

UML fournit les moyens de visualiser, de documenter et de modéliser des applications logicielles avant de les coder. Il est indépendant de tout langage de programmation qui sera utilisé pour coder le logiciel d'application final. UML représente une collection des meilleures pratiques d'ingénierie qui ont fait leurs preuves dans modélisation de grands systèmes complexes [42] [43] [44] [45]. De nombreuses tentatives réussies ont été envisagées pour étendre l'application d'UML à des domaines autres que l'informatique [44].

II.2.2 Présentation Générale des diagrammes UML

Version UML 2.0 est construite autour de treize (13) diagrammes différents, elle comporte donc quatre nouveaux diagrammes par rapport à la version précédente. Chaque schéma est destiné à représenter un système logiciel selon une vision spécifique. UML modélise le système selon deux représentations : L'un implique la structure du système dans un état « statique », et l'autre implique sa dynamique. En fonctionnement. Ces deux représentations peuvent être utilisées comme références pour modéliser les éléments suivants : Composants du système et leurs interactions fonctionnelles.

II.3 Analyse

II.3.1 Identifications des acteurs

II.3.1.1 Définition d'un acteur

Acteur est un rôle qu'un utilisateur joue par rapport au système.

En principe, cependant, un acteur peut non seulement être [46]:

- Le créateur d'une action (par exemple, l'initiateur d'un cas d'application)
- Mais aussi le destinataire d'un résultat attendu.

II.3.1.2 Les acteurs de notre système

Les acteurs qui interviennent dans notre système sont : **Sportif** et **Coach**.

Sportif : personne sportive

Coach : entraîneur professionnel de sportif

II.3.2 Diagramme de Contexte

Diagramme de contexte est le niveau le plus élevé d'un diagramme de flux de données. Il met en évidence le flux d'informations entre le système et les composants externes.



Figure II.1 : *Diagramme de contexte*

II.3.3 Spécification des besoins :

Le tableau (II.1) résume les besoins fonctionnels de notre application :

Acteur	Besoin
Sportif	<ul style="list-style-type: none">B1 : S'inscrireB2 : S'authentifierB3 : Entrer ses informationsB4 : Enregistrer ses informationsB5 : Afficher ses informationsB6 : Voir son besoin énergétiqueB7 : Sélectionner le sport en questionB8 : Démarrer le chronomètreB9 : Mettre le chronomètre en pauseB10 : Réinitialiser le chronomètreB11 : Consulter les Calories brûléesB12 : Consulter nombre de battement de cœur MoyenB13 : Envoyer les statistiques au coachB14 : Visualiser les statistiques graphiquement
Coach	<ul style="list-style-type: none">B1 : Consulter les statistiques

Tableau II.1: *Spécification des besoins.*

II.3.4 Spécification des cas d'utilisation

II.3.4.1 Définition

Cas d'utilisation est un cas d'utilisation d'un système par certains utilisateurs pour réaliser un processus. De manière un peu plus précise, un cas d'utilisation décrit la séquence d'événements de certains types d'utilisateurs, appelés acteurs, utilisant une partie de la fonctionnalité du système pour réaliser un processus.

Par conséquent, pour un utilisateur, un cas d'utilisation est une façon d'utiliser le système. Un cas d'utilisation est décrit en termes de séquence d'interactions entre certains acteurs et le système par laquelle le système fournit un service aux acteurs. Chaque cas d'utilisation capture ensuite une partie des exigences fonctionnelles pour certains utilisateurs. Tous les cas d'utilisation décrivent ensemble les exigences fonctionnelles globales du système [47].

II.3.4.2 Notations de base du diagramme de cas d'utilisation

Système : Système est utilisé pour définir la portée du cas d'utilisation.

Cas d'utilisation : Fonction du système

Acteur : Une personne interagit avec le cas d'utilisation

Lien de communication : Participation d'un acteur à un cas d'utilisation est indiquée en reliant un acteur à un cas d'utilisation par un lien solide.

II.3.4.3 Relations entre cas d'utilisation

Les cas d'utilisation partagent différents types de relations. La définition de la relation entre deux cas d'utilisation est la décision des analystes logiciels du diagramme de cas d'utilisation. Une relation entre deux cas d'utilisation consiste essentiellement à modéliser la dépendance entre les deux cas d'utilisation. La réutilisation d'un cas d'utilisation existant en utilisant différents types de relations réduit l'effort global nécessaire au développement d'un système.

Les relations entre les cas d'utilisation sont les suivantes :




Relation	Symbole	Signification
Includes		Un cas d'utilisation contient un comportement qui est commun à plus d'un autre cas d'utilisation. La flèche pointe vers le cas d'utilisation commun.
Extends		Un cas d'utilisation différent traite les exceptions du cas d'utilisation de base. La flèche pointe du cas d'utilisation étendu vers le cas d'utilisation de base.
Généralisation		Un "cas d'utilisation" UML est plus générale qu'un autre "cas". La flèche pointe vers le "cas" générale.

Tableau II.2: Relations entre cas d'utilisation [48]

II.3.4.4 Diagramme de cas d'utilisation

Le diagramme de la **Figure II.2** résume tous les cas d'utilisation associés à nos acteurs.

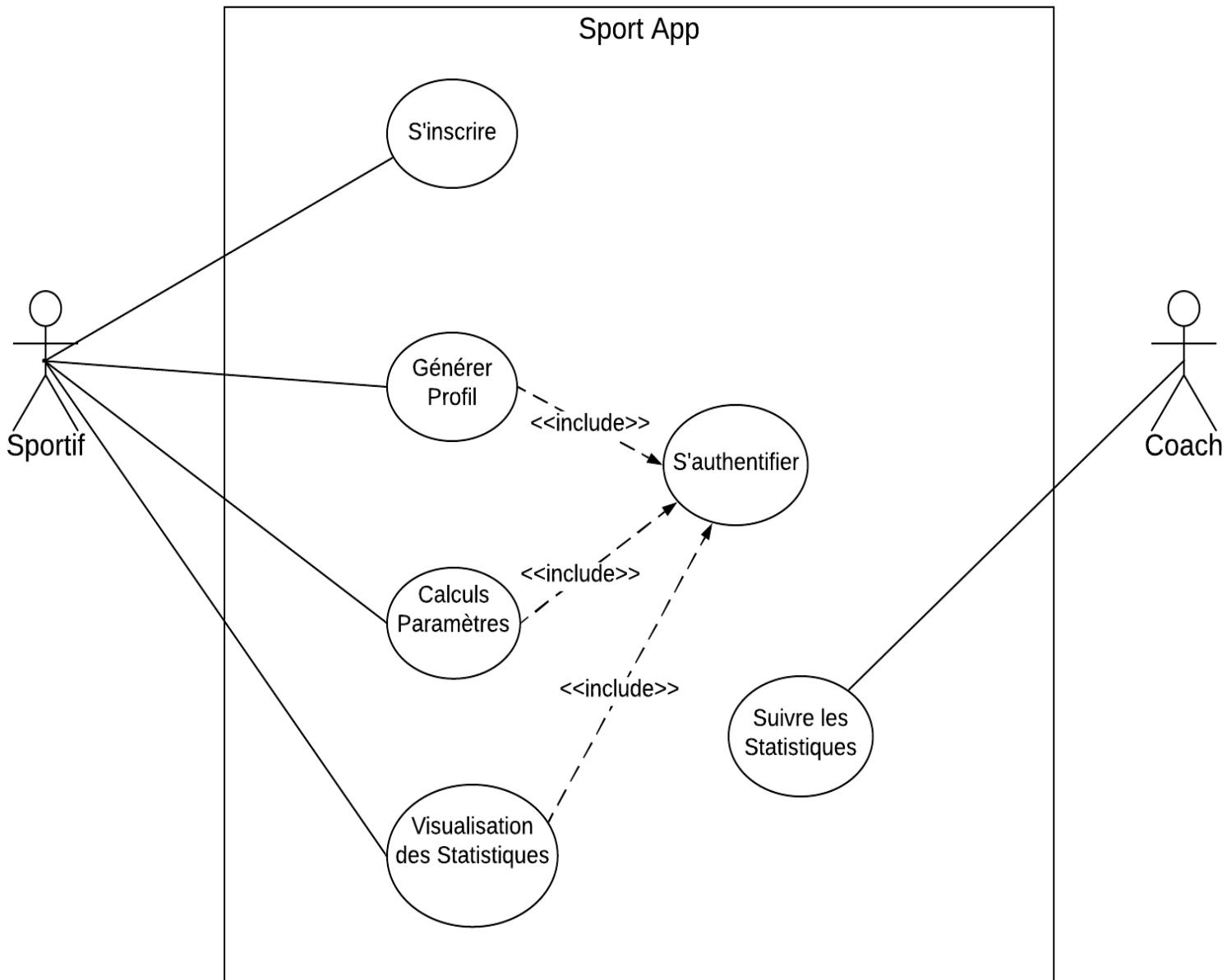


Figure II.2 : *Diagramme de cas d'utilisation*

II.3.4.5 Description textuelle des cas d'utilisation

Description de cas d'utilisation « S'inscrire »

Cas d'utilisation	S'inscrire
Acteur	Sportif
Objectif	Créer un compte
Scénario nominal	<ul style="list-style-type: none">- L'utilisateur demande l'inscription- Le système affiche le formulaire d'inscription- L'utilisateur remplit le formulaire et valide- Le système vérifie la validité des informations [A1]- Le système crée un compte pour l'utilisateur.
Alternative [A1]	Si un champ d'information n'est pas valide ou l'utilisateur existe déjà, le système affiche un message d'erreur.

Tableau II.3 : Description de cas d'utilisation « S'inscrire »

Description de cas d'utilisation « S'authentifier »

Cas d'utilisation	S'authentifier
Acteur	Sportif
Objectif	Vérifie que l'utilisateur a le droit d'accéder à l'application
Scénario nominal	<ul style="list-style-type: none">- L'utilisateur demande l'accès à l'application- Le système affiche l'interface Authentification- L'utilisateur introduit son identifiant et le mot de passe- Le système vérifie les informations et l'existence de l'utilisateur. [A1]- Le système donne l'accès à l'interface Profil.
Alternative [A1]	Si un champ d'information n'est pas valide ou l'utilisateur n'existe pas, le système affiche un message d'erreur.

Tableau II.4 : Description de cas d'utilisation « S'authentifier »

Description de cas d'utilisation « Générer Profil »

Cas d'utilisation	Générer Profil
Acteur	Sportif
Objectif	Entre Les informations / mise à jour des informations
Précondition	Authentification
Scénario nominal	<ul style="list-style-type: none"> - Le système affiche l'interface profil. - L'utilisateur entre ses informations avec la validation [A1]. - Le système enregistre les informations.
Alternative [A1]	Si un champ d'information est vide, le système affiche un message d'erreur.

Tableau II.5 : Description de cas d'utilisation « Générer Profil »

Description de cas d'utilisation « Calculs Paramètres »

Cas d'utilisation	Calculs Paramètres
Acteur	Sportif
Objectif	Calculez les besoins énergétiques + calories brûlées + fréquence cardiaque moyenne
Précondition	Authentification
Scénario nominal	<ul style="list-style-type: none"> - Le système affiche l'interface correspondante - L'utilisateur sélectionne son niveau d'activités physique. - Le système affiche son besoin énergétique. - L'utilisateur sélectionne le sport qu'il va pratiquer. - L'utilisateur contrôle le chronomètre pendant son activité. - Le système calculer les calories brûlées et la fréquence cardiaque moyenne.

Tableau II.6 : Description de cas d'utilisation « Calculs Paramètres »

Description de cas d'utilisation « Visualisation des données »

Cas d'utilisation	Visualisation des données
Acteur	Sportif
Objectif	Afficher les statistiques sous forme d'un graphe
Précondition	Authentification
Scénario nominal	- L'utilisateur connecte à la base de données via le réseau [A1] . -Le système affiche les statistiques sous forme courbe.
Alternative [A1]	S'il n'y a pas de connexion internet, le système doit afficher un message d'erreur et afficher le graphe avec les statistiques de dernière connexion au réseau.

Tableau II.7 : Description de cas d'utilisation « Visualisation des données »

II.4 Conception

II.4.1 Diagramme de classes

II.4.1.1 Définition

Les Diagrammes de classes sont les diagrammes les plus courants dans la modélisation des systèmes orientés objet. Ce diagramme explore la conception détaillée du système. Le diagramme de classes est conçu à l'aide du diagramme de cas d'utilisation. Nous pouvons identifier tous les "Noms" dans les cas d'utilisation comme des classes et les "verbes" comme des méthodes des classes. Les diagrammes de classes sont les diagrammes UML les plus populaires utilisés par la communauté orientée objet. Ils décrivent les objets d'un système et leurs relations. Le diagramme de classes se compose d'attributs et de fonctions. Un seul diagramme de classes décrit un aspect spécifique du système et la collection de diagrammes de classes représente le système entier. Fondamentalement, le diagramme de classes représente la vue statique d'un système. Les diagrammes de classes sont les seuls diagrammes UML qui peuvent être mappés directement avec les langages orientés objet. Ils sont donc largement utilisés par la communauté des développeurs [49].

II.4.1.2 Caractéristiques d'une classe

Dans UML, une classe est représentée sur le diagramme de classes par un rectangle avec trois sections horizontales, comme montre la **Figure II.3**.

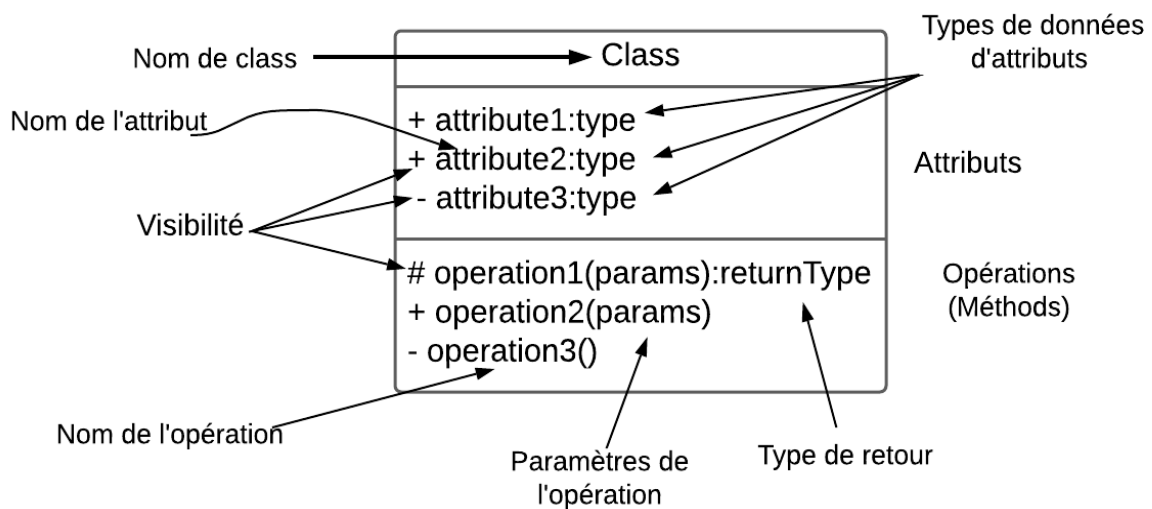


Figure II.3 : Représentation UML d'une classe.

a - Nom : La première ligne d'une forme de classe.

b - Attributs : La deuxième ligne d'une forme de classe. Chaque attribut de la classe est affiché sur une ligne distincte.

c - Opérations : La troisième ligne d'une classe. Également appelées méthodes, les opérations sont affichées sous forme de liste, chaque opération figurant sur une ligne distincte.

d - Visibilité : Utilisez des marqueurs de visibilité pour indiquer qui peut accéder aux informations contenues dans une classe.

Visibilité privée (private) indiquée par le signe -, cache les informations à tout ce qui est extérieur à la classe.

Visibilité publique (public) indiquée par le signe +, nous permet d'utiliser les informations marquées à tout ce qui est extérieur à la classe.

Visibilité protégée (protected) indiquée par le signe #, permet aux classes enfants d'accéder aux informations qu'elles ont héritées d'une classe parent.

II.4.1.3 Relations dans les diagrammes de classes

Type de relation	Notation
Association	
Inheritance	
Realisation /implementation	
Dependency	
Aggrégation	
Composition	

Tableau II.8 : Relations entre classes

II.4.1.4 Diagramme

Diagramme de classes présente les classes utilisées par le système ainsi que leurs liens, le diagramme de classes de notre système est représenté sur la **Figure II.4**.

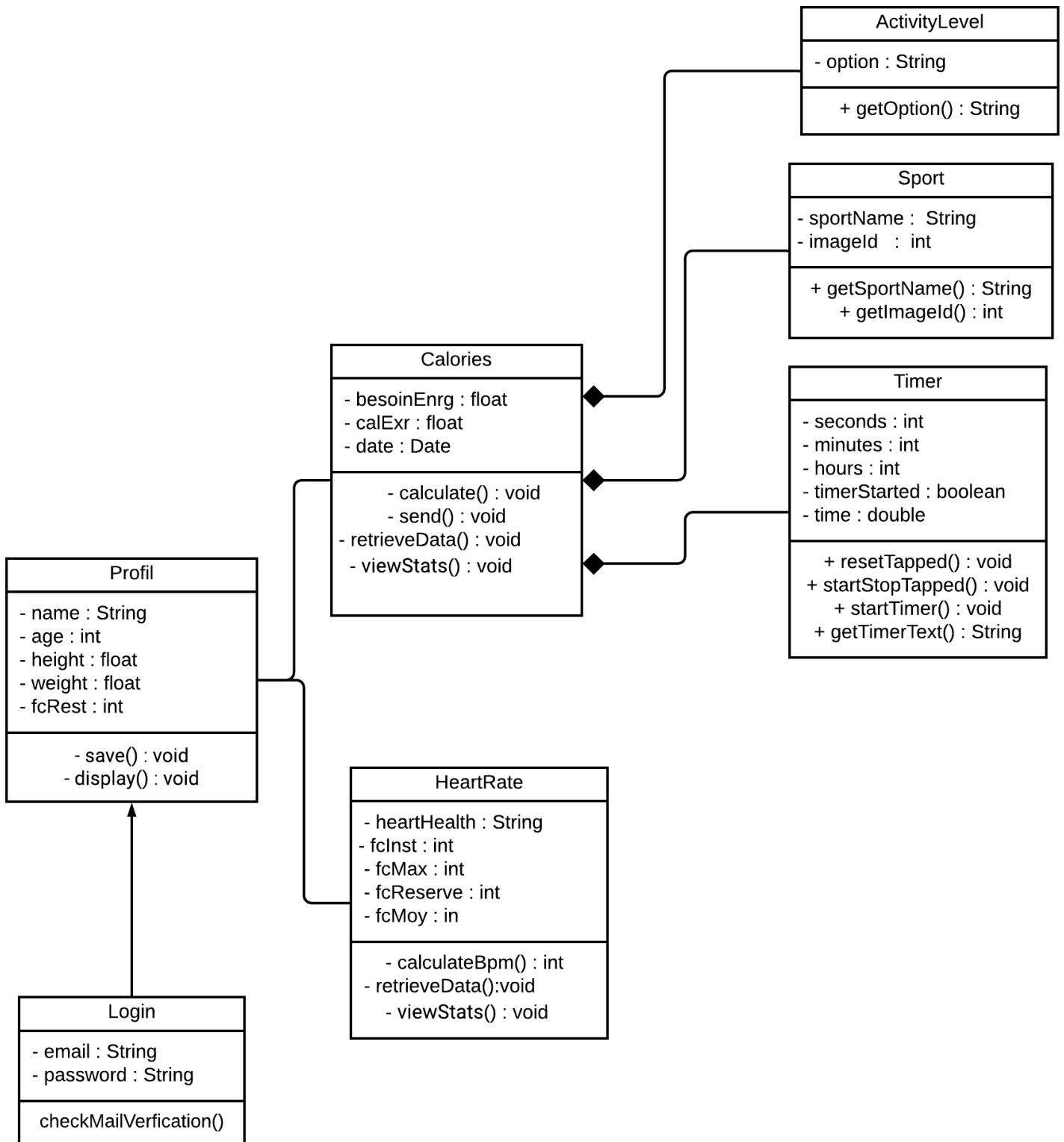


Figure II.4 : *Diagramme de classes*

Class	Méthode	Attribut	Code	Type de données
Login	checkMailVerification()	E-mail	Email	Chaine de caractères
		Mot de passe	Password	Chaine de caractères
Profil	save() display()	Nom d'utilisateur	name	Chaine de caractères
		Age d'utilisateur	age	Entier
		Taille d'utilisateur	height	Flottant
		Poids d'utilisateur	weight	Flottant
		Fréquence cardiaque de repos d'utilisateur	fcRest	Entier
Calories	calculate() send() retrieveData() viewStats()	Besoin énergétique d'utilisateur	BesoinEnrg	Flottant
		Calories brûlées pendant une séance de sport	CalExr	Flottant
		La date et l'heure de l'envoi	Date	Date (dd/mm/yyyy HH:MM:SS)
HeartRate	calculateBpm() retrieveData() viewStats()	Santé cardiaque	heartHealth	Chaine de caractères
		Fréquence cardiaque instantanée	fcInst	Entier
		Fréquence cardiaque Max	fcMax	Entier
		Fréquence cardiaque de reserve	fcReserve	Entier

ActivityLevel	getOption()	Option de Niveau d'activités physique	Option	Chaine de caractères
Sport	getSportName()	Nom de sport	sportName	Chaine de caractères
	getImageId()	Id de l'image de sport	imageId	Entier
Timer	resetTapped()	Nombre de seconds	seconds	Entier
	stopStartTapped()	Nombre de minutes	minutes	Entier
	startTimer()	Nombre d'heurs	hours	Entier
	getTimerText()	Le minuteur est en marche	timerStarted	Booléenne
		Temp écoulé	time	Flottant

Tableau II.9 : *Dictionnaire de données.*

II.4.2 Diagramme de Séquence

II.4.2.1 Définition

Les Diagrammes de séquence sont des artefacts UML essentiels pour la modélisation des aspects comportementaux d'un système [50] [51]. Ces diagrammes sont particulièrement bien adaptés aux logiciels orientés objet, où ils représentent le flux de contrôle pendant la phase de développement de l'objet. Diagramme de séquence montre un ensemble d'objets en interaction et la séquence de messages échangés entre eux. Le diagramme peut également contenir informations supplémentaires sur le flux de contrôle pendant l'interaction, telles que des conditions comme les conditions (par exemple, "si la condition c alors envoyer le message m sinon envoyer le message n") et l'itération (par exemple, "envoyer le message m plusieurs fois") ou le comportement dépendant de l'état [51].

II.4.2.2 Diagrammes

- ❖ Diagramme de séquence du cas « S'inscrire »
- ❖ Diagramme de séquence du cas « S'authentifier »
- ❖ Diagramme de séquence du cas « Générer Profil »
- ❖ Diagramme de séquence du cas « Calculs Paramètres »
- ❖ Diagramme de séquence du cas « Visualisation des données »

La **Figure II.5** Présente le diagramme de séquence de cas d'utilisation « S'inscrire », qui montre le flux détaillé d'inscription d'un utilisateur.

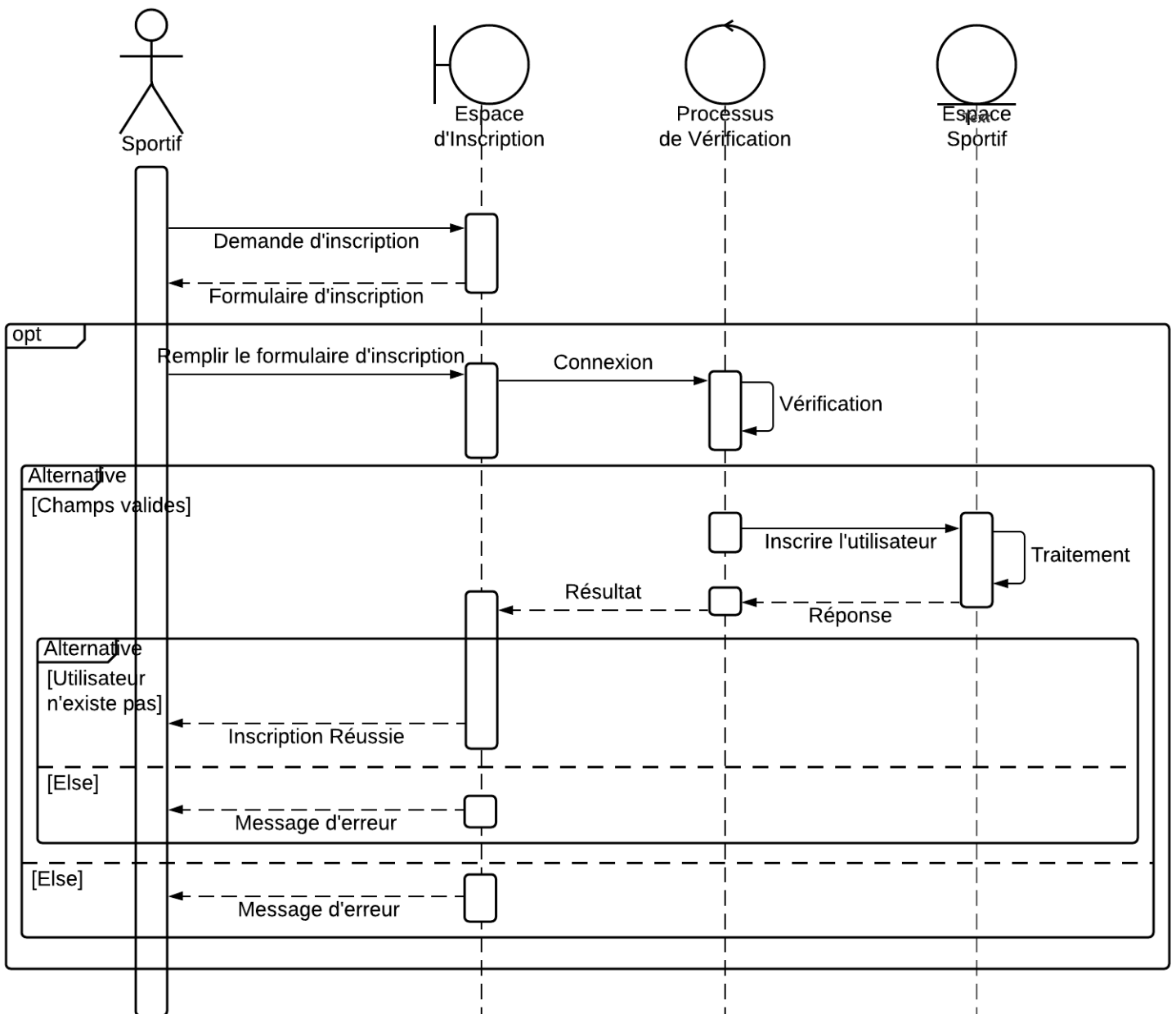


Figure II.5 : Diagramme de séquence du cas d'utilisation « s'inscrire »

La **Figure II.6** montre les étapes de cas d'utilisation « S'authentifier ».

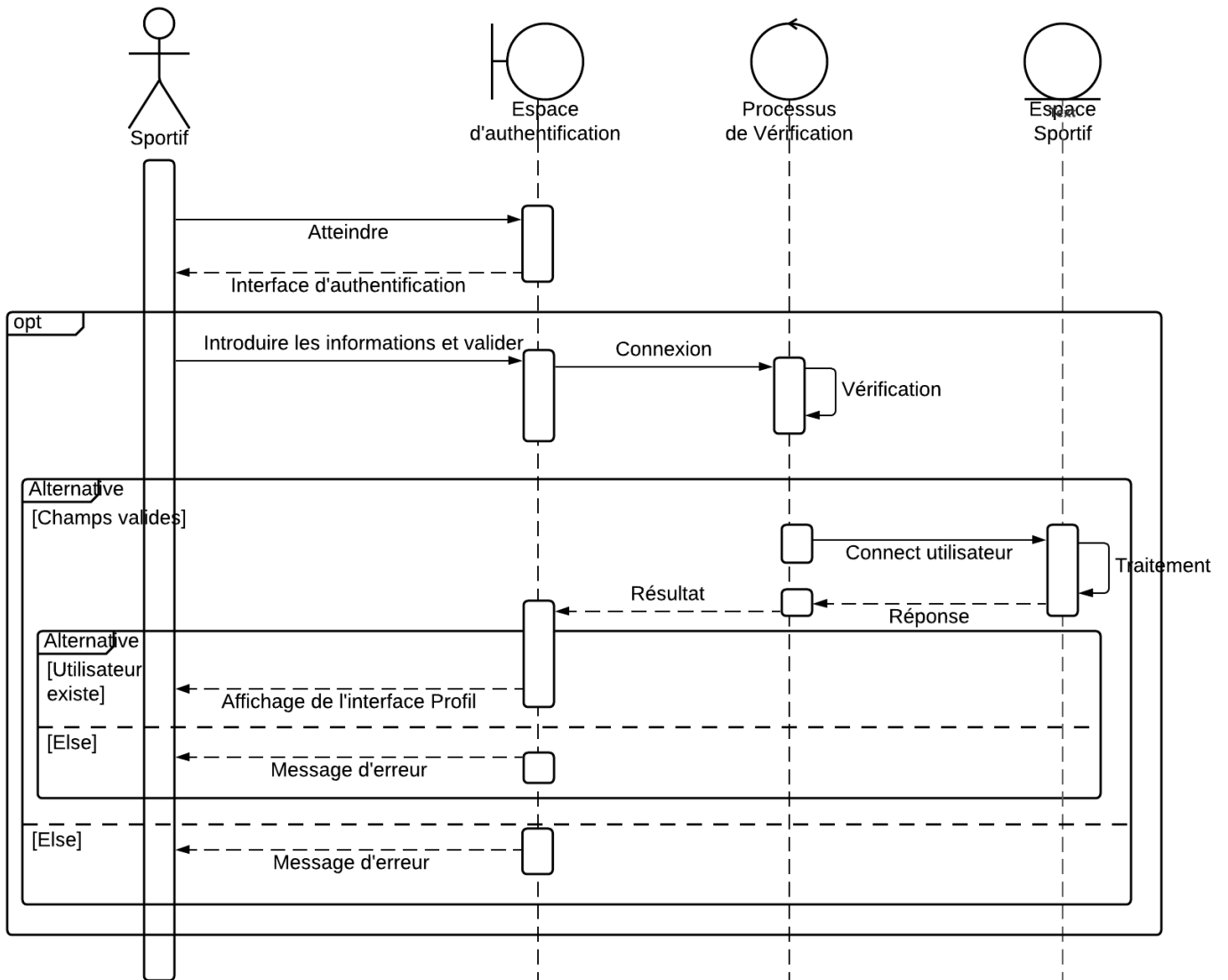


Figure II.6 : Diagramme de séquence du cas d'utilisation « s'authentifier »

La génération de profil à propos de diagramme de cas d'utilisation est montrée sur la **Figure II.7**

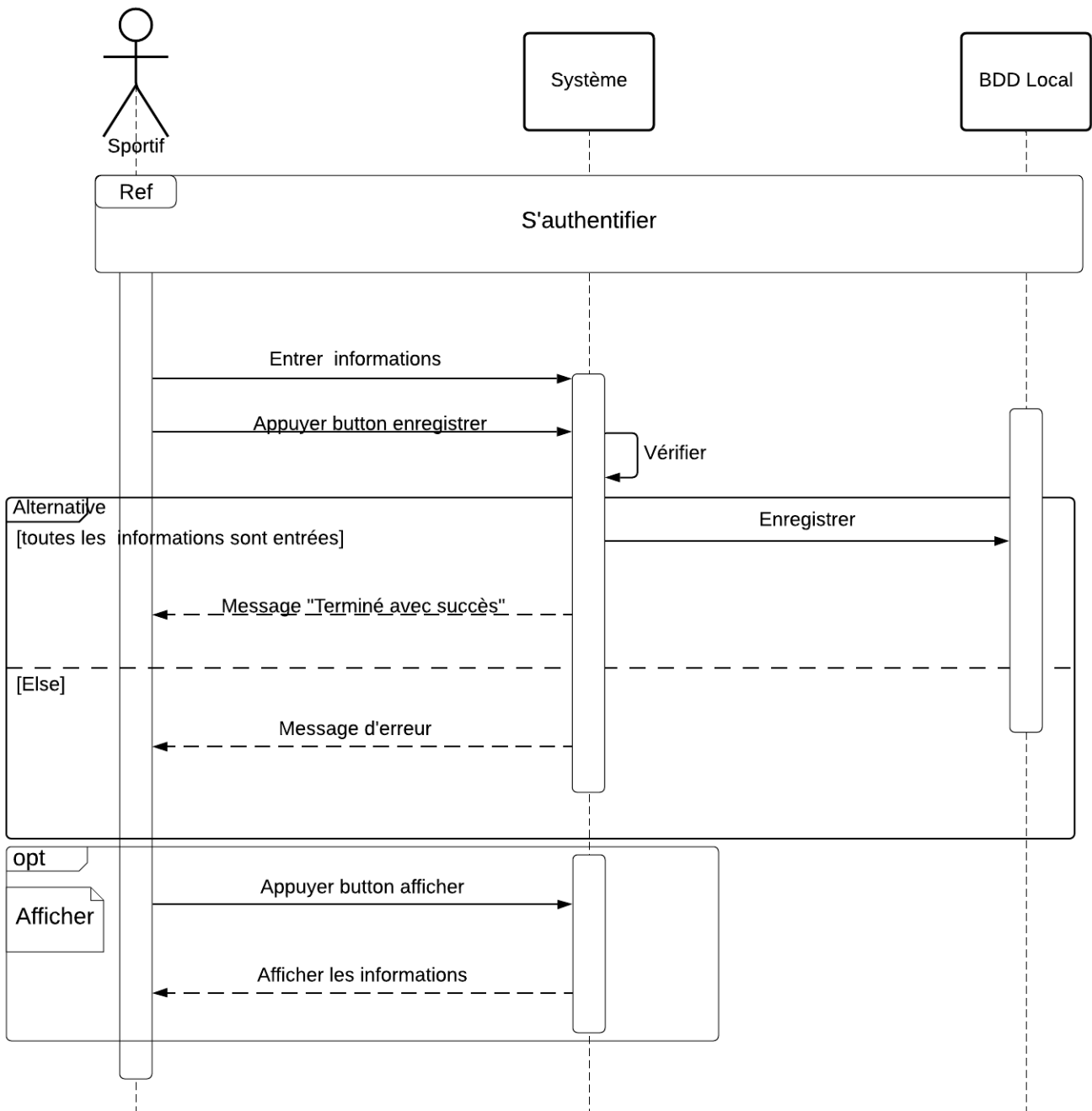


Figure II.7 : *Diagramme de séquence du cas d'utilisation « générer le profil »*

La **Figure II.8** représente le scénario détaillé de cas d'utilisation : « paramètres de calculs ».

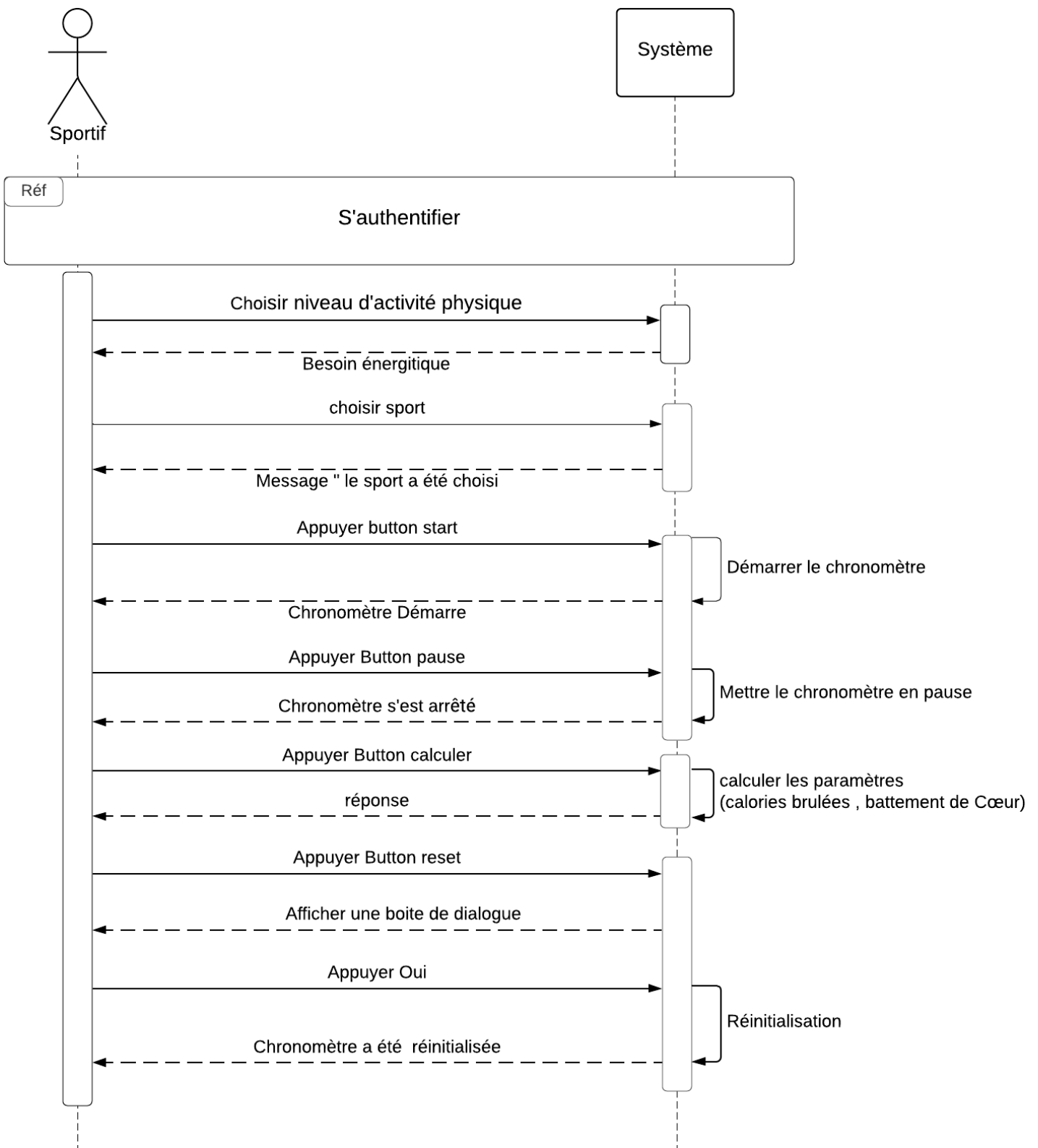


Figure II.8 : *Diagramme de séquence du cas d'utilisation « Paramètres de calculs »*

Pour que le sportif voir ses statistiques, il faut des étapes à suivre, la **Figure II.9** montre ces étapes.

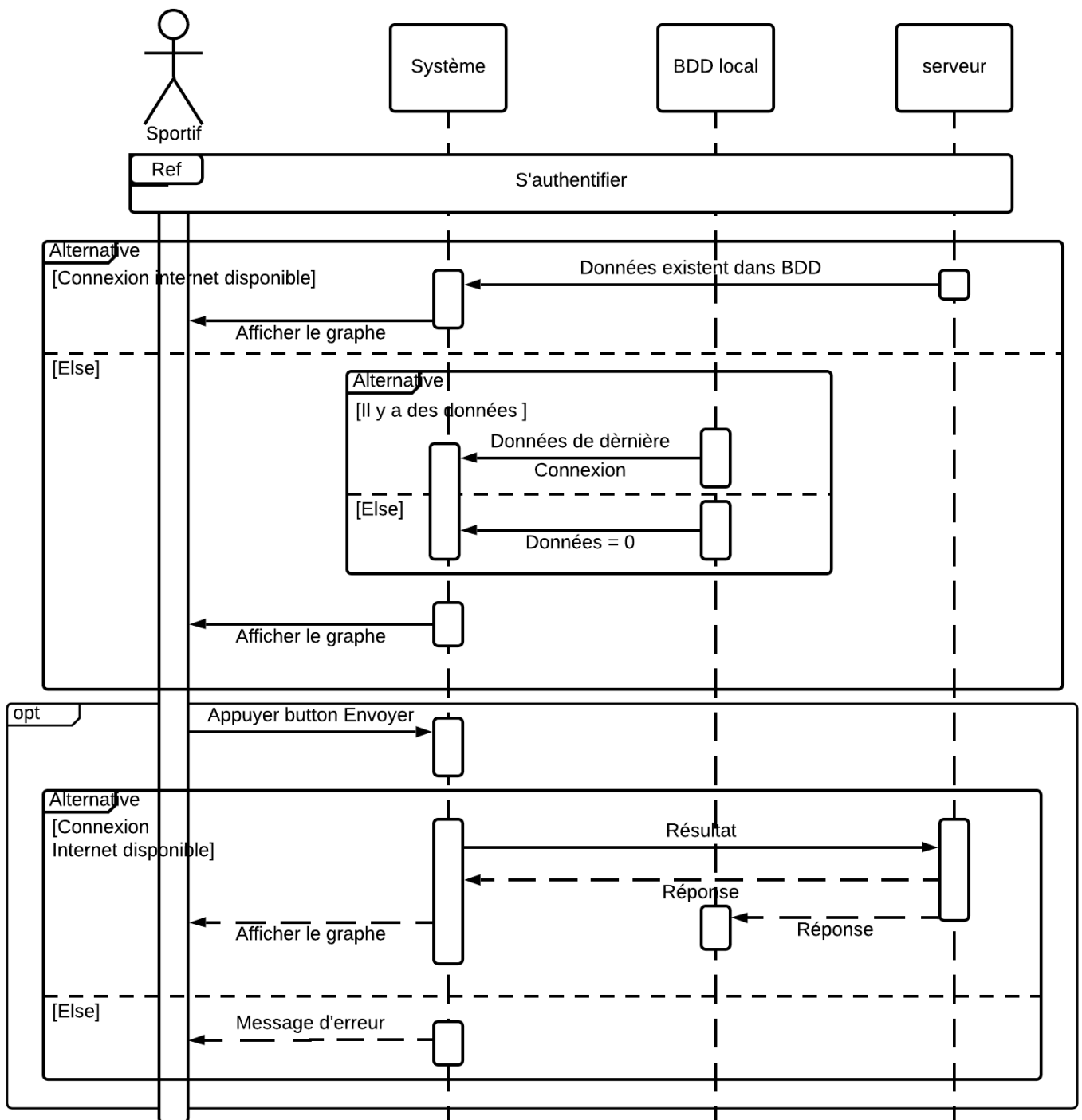


Figure II.9 : Diagramme de séquence du cas d'utilisation « Visualisation des statistiques »

II.4.3 Patrons de conception (Design Pattern)

Patron de conception est un bon pratique dans le domaine informatique mais aussi dans le développement logiciel, il synthétise et analyse les problèmes de la conception de logiciel et apporte les solutions, on peut avoir plusieurs modules de patrons qui interagissent entre eux afin de donner les solutions.

Les patrons utilisés dans notre application :

- ❖ Patron Modèle-Vue-Controller MVC
- ❖ Patron Client/Serveur à 3-tier

II.4.3.1 Patron Modèle-Vue-Controller « MVC »

Modèle d'architecture MVC est une façon de présenter les informations ou les données à l'utilisateur et la façon dont l'utilisateur interagit/traite avec la vue des données.

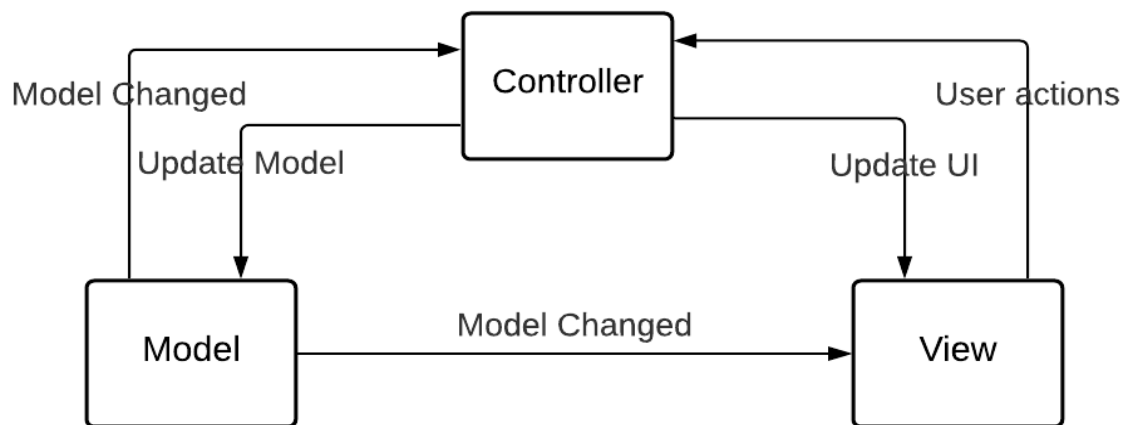


Figure II.10 : *Patron MVC*

Les 3 couches de ce modèle de conception :

Modèle : Modèle est responsable de la gestion des données de l'application.

Vue : Il s'agit de la présentation de données dans un format particulier, déclenchée par la décision d'un contrôleur de présenter les données.

Contrôleur : Contrôleur est chargé de répondre aux entrées de l'utilisateur et d'effectuer des interactions sur les objets du modèle de données.

II.4.3.2 Patron Client / Serveur à 3-tier

Architecture de système client-serveur à 3-tier : Cette architecture implique le client, le serveur de base de données et le serveur d'applications. L'architecture à 3 niveaux peut être étendue à N niveaux, ce qui implique davantage de serveurs d'applications. Dans cette architecture, le client contient uniquement la logique de présentation, ce qui nécessite moins de ressources et moins de codage pour le client [52].

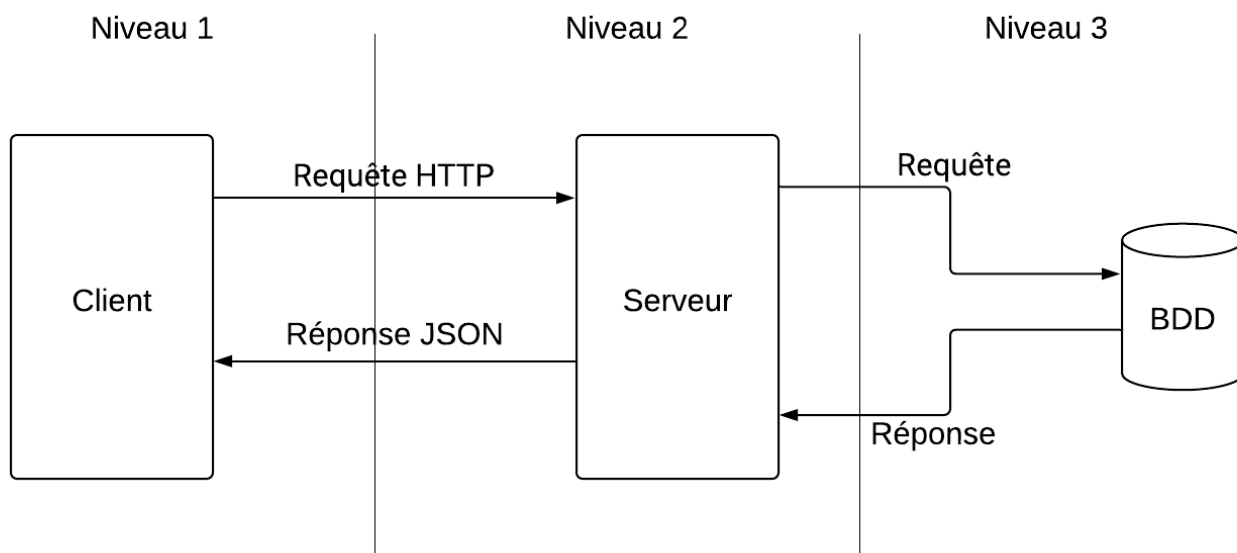


Figure II.11 : Architecture « 3-tiers »

II.4.4 Modèle Relationnel

Le modèle relationnel a été défini par E.F. Codd dans IBM SanJosé en 1970.

Aspects de base du modèle relationnel :

- Algèbre permettant la manipulation des tables et des relations
- Définir le processus de conception des collections relationnelles

On dit que le modèle est relationnel, car il peut parcourir des structures de données qui utilisent des chemins non prédéfinis formés dynamiquement par des requêtes

Le concept de modèle relationnel est dérivé de la théorie des ensembles

Ce type de modèle est associé aux concepts suivants [53] :

- **Domaine**
- **Tables relationnelles**
- **Les attributs**
- **Tuple (ou tuple)**

Un domaine est un ensemble de valeurs significatives pour l'utilisateur

Table relationnelle : un sous-ensemble du produit cartésien de la liste de domaines (pas nécessairement unique).

Table des relations commence généralement par Nom.

Définir une table (ou une relation) comme une table de données.

Les colonnes sont **des attributs**, les lignes sont **des tuples**.

-Un ou plusieurs **attributs** peuvent identifier de manière unique chaque tuple de la table : clé primaire.

-Si la clé primaire est composée d'un seul attribut et composée d'autres attributs, alors la clé primaire est très simple.

-La clé est un ensemble minimal d'attributs, Décidez de tous les autres.

-Il peut y avoir plusieurs clés identiques Relation ; nous en choisissons généralement une comme clé de base.

-Etablir un lien entre 2 tables relationnelles en copiant la clé primaire d'une table vers une autre table

-Cette clé dupliquée est appelée clé externe (ou externe).

II.4.5 Firebase

II.4.5.1 Définition

Firebase est considéré comme une plateforme d'application web. Elle aide les développeurs à créer des applications de haute qualité. Elle stocke les données au format JavaScript Object Notation (JSON) qui n'utilise pas de requête pour insérer, mettre à jour, supprimer ou ajouter des données. Il s'agit du backend d'un système qui est utilisé comme une base de données pour stocker des données [54].

II.4.5.2 Services Firebase

Firebase dispose de huit technologies conçues pour améliorer votre expérience de développement d'applications, et elles sont les suivantes :

A. Firebase Analytics

Il fournit un aperçu de l'utilisation des applications. Il s'agit d'une solution payante de mesure des applications qui fournit également un engagement de l'utilisateur. Cette fonctionnalité unique permet au développeur d'applications de comprendre comment les utilisateurs utilisent l'application. Le SDK a la possibilité de capturer des événements et des propriétés de manière autonome et permet également d'obtenir des données personnalisées [54].

B. Firebase Cloud Messaging (FCM)

Anciennement connu sous le nom de Google Clouds Messaging (GCM), FCM est un service payant qui est une solution multiplateforme pour les messages et les notifications pour Android, les applications Web et IOS [54].

C. Firebase Authentication

Lorsque votre application doit connaître l'identité de l'utilisateur afin de lui fournir des données distinctives, une forme de connexion sécurisée est nécessaire. La mise en place et la maintenance d'une infrastructure de connexion est une proposition difficile et coûteuse. Du point de vue de l'utilisateur, le fait de donner des informations d'identification et des données personnelles à une application peut également nuire à l'expérience de l'utilisateur et l'amener à refuser de l'utiliser. C'est pourquoi les applications qui permettent aux utilisateurs de se connecter avec des informations d'identification connues, comme celles fournies par Google, Facebook et Twitter, sont de plus en plus populaires. En gardant à l'esprit ces problèmes et ces tendances, Firebase Authentication a été conçu pour vous offrir une API simple qui vous permettra d'utiliser l'ouverture de session à partir de fournisseurs fédérés, un simple schéma email/mot de passe ou l'intégration de tout back-end d'authentification existant que vous possédez. Elle s'intègre aux services Firebase tels que la base de données Realtime afin que vous puissiez contrôler qui accède à quelles données [55].

D. Base de données en temps réel (Realtime Database)

Il s'agit d'une base de données NoSQL hébergée dans le Cloud. Elle permet la synchronisation entre les appareils connectés et elle est disponible en l'absence de connectivité réseau via un cache local. Il s'agit d'une base de données pilotée par les événements qui fonctionne très différemment des bases de données SQL traditionnelles. Il n'y a pas de code côté serveur et de niveaux d'accès à la base de données, tout le codage est effectué dans le client. Chaque fois que des données changent dans la base de données, des événements sont déclenchés dans le code client. Et mettre à jour l'état de votre interface utilisateur en réponse. Il contient un langage de règles basé sur des expressions, appelé Firebase Realtime Database Security Rules, qui définit la façon dont les données sont structurées et quels utilisateurs ont des droits sur ces données. Sa conception ultime est d'être réactive, vous permettant de construire une expérience en temps réel qui peut servir les utilisateurs à grande échelle. En tant que tel, il est un peu différent de celui d'une base de données traditionnelle [55].

E. Stockage Firebase

Il facilite le transfert facile et sécurisé des fichiers indépendamment de la qualité du réseau pour les applications Firebase. Il est soutenu par Google Cloud Storage qui est un service de stockage d'objets rentable. Le développeur peut l'utiliser pour stocker des images, de l'audio, de la vidéo ou tout autre contenu généré par l'utilisateur [54].

F. Firebase Test Lab

Ce service fournit une infrastructure basée sur le cloud pour tester les applications Android. En une seule opération, les développeurs peuvent lancer le test de leurs applications sur une grande variété d'appareils et de configurations d'appareils. Les différents résultats des tests, comme les captures d'écran, les vidéos et les journaux, sont disponibles dans la console Firebase. Même si un développeur n'a pas écrit de code de test pour son application, Test Lab peut exercer l'application automatiquement, à la recherche de plantages [54].

G. Firebase Crash Reporting

Les rapports détaillés des erreurs sont créés dans l'application. Les erreurs sont regroupées en groupes de traces de pile similaires et triées en fonction de leur gravité. Les autres caractéristiques sont : le développeur peut enregistrer des événements personnalisés pour aider à capturer les étapes menant à un crash [54].

H. Notifications Firebase

Cette application permet aux développeurs d'applications mobiles d'envoyer des notifications ciblées aux utilisateurs et les services sont disponibles gratuitement [54].

II.5 Conclusion

Dans ce chapitre, nous avons détaillé la conception de notre système proposé selon une approche orientée objet pour garantir la fiabilité et l'efficacité en phase de mise en œuvre. Pour cela nous avons présenté les différents diagrammes UML utilisés, à savoir : diagramme de classes, diagramme de cas d'utilisation et diagrammes de séquence représentée par le paradigme visuel.

Dans le prochain chapitre nous présenterons la mise en œuvre réelle et la réalisation de l'application en précisant les outils de développement exploités et les différentes étapes d'exécution de notre plateforme.

CHAPITRE III : REALISATION

III.1 Introduction

La réalisation représente l'étape qui suit immédiatement la phase de conception, c'est l'aboutissement final de notre projet et par conséquent l'achèvement de l'application.

Dans ce chapitre nous dressons le contexte particulier de l'étude à long terme des différentes technologies et protocoles utilisés dans le monde de la téléphonie mobile. Nous commençons par la présentation de la plateforme à réaliser, par la suite, nous décrivons la description générale de notre projet, suivi du bilan et l'exécution de l'application réalisée.

III.2 Plateforme à réaliser

Notre plateforme proposée est schématisée sur la **Figure III.1** :

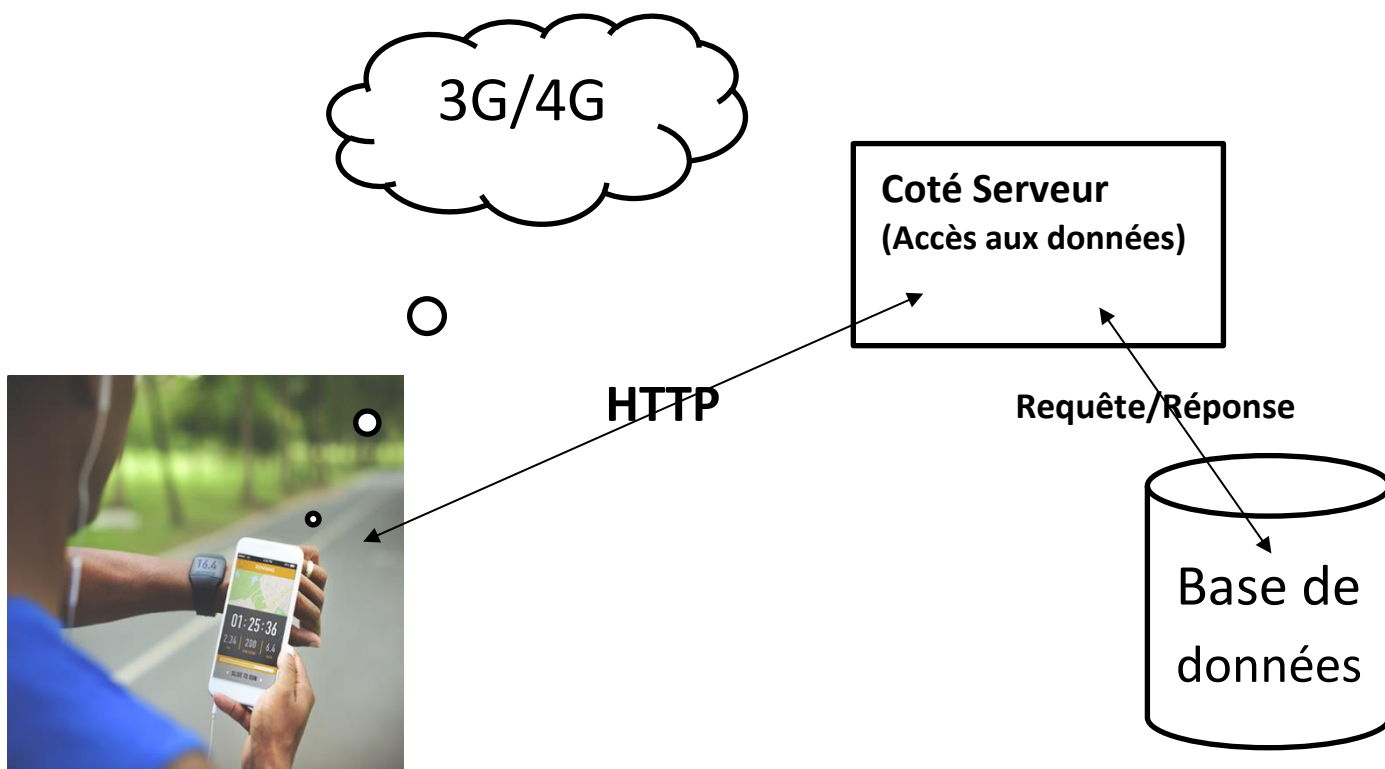


Figure III.1 : Plateforme à réaliser

Le sportif a besoin de passer par les étapes suivantes :

- ❖ Lance l'application dans son mobile
- ❖ Inscription
- ❖ Authentification
- ❖ Profil
- ❖ Traitement : Calcul calories, Fréquence cardiaque

III.3 Description générale de notre projet

Notre projet suit les trois étapes suivantes :

III.3.1 Établissement de la connexion

Les principaux atouts d'un terminal sans fil sont sa connectivité et son accessibilité, qui permettent de rester connecté avec le monde entier à tout instant et en tout lieu couvert par un réseau de télécommunication.

La première partie donc qui devra être réalisée concerne l'interconnexion à distance des bases de stockages des données et des dispositifs auprès du Coach. Tous ces équipements doivent se mettre d'accord sur le même tunnel (port, adresse...) pour qu'ils puissent échanger toutes sortes de données. La connectivité est assurée par le protocole de communication http via les méthodes GET et POST.

III.3.2 Traitement

Cette partie ne nécessite pas d'étude particulière, si ce n'est la recherche des bibliothèques de l'Android Studio qui permettent une telle manipulation de traitement.

III.3.3 Transmission des données

La troisième étape du projet consiste à trouver les meilleurs moyens pour le transfert des données sportifs et les résultats au Coach. Ce transfert entre des équipements hétérogènes (terminal mobile, serveur...) fait appel à un grand nombre de technologies, qu'il va falloir étudier et tester de manière à évaluer la technologie la plus adaptée à ce projet. L'assistant mobile de notre plateforme est dans notre cas un Smartphone. Il exploite deux modes de transmission :

III.3.3.1 Transmission entre deux terminaux mobiles

Il n'existe pas énormément de possibilités. En effet, il est possible d'envoyer des SMS, des MMS et des e-mails. Les plus répandus sont les e-mails et les MMS. De plus, ces deux possibilités se révèlent plus sophistiquées que les autres. Elles permettent d'envoyer toute sorte de documents électroniques (texte, image, son...).

III.3.3.2 Mobile et serveur (base de données)

Ceci implique l'existence d'une connexion http entre l'outil sportif qui est dans ce cas le terminal mobile et le serveur de la base de données via une passerelle WAP pour la transmission des données.

Ce transfert est basé sur une communication WSP/http [56]. Comme son nom l'indique Wireless Session Protocol, couche session permet de définir les paramètres de connexion pour effectuer des transactions. Elle permet ainsi à la couche application de bénéficier de deux types de sessions différentes :

- Session orientée connexion où la couche session va interagir avec la couche transaction.
- Session orientée non-connexion où la couche session va directement agir au niveau de la couche transport pour l'envoi de datagrammes bruts.

WSP est dans son ensemble l'équivalent du protocole http (dans sa version 1.1). On retrouve d'ailleurs un bon nombre d'implémentations identiques au http dans WSP.

Là aussi, il existe deux possibilités. On peut transférer les informations nécessaires par une connexion réseau sur un serveur (base de données) ou par e-mail.

A. Connexion réseau sur un Serveur de la base de données

Cette partie ne nécessite pas d'étude particulière, si ce n'est la recherche des bibliothèques et les paquetages nécessaires qui permettent une telle manipulation d'une base de données à distance (MySQL par exemple).

B. Courrier électronique

C'est la possibilité de transférer les mesures et les résultats obtenus sur Internet par e-mail. Ce service, qui fonctionne un peu comme le service de courrier réel, est basé sur l'utilisation de deux protocoles principaux : SMTP et POP.

POP étant utilisé pour consulter le courrier reçu, il ne sera pas abordé dans cette partie. Par contre le protocole SMTP qui est à la base de tout envoi de courrier électronique nécessite un développement.

B.1 SMTP :

Le transfert du courrier par le protocole SMTP se fait d'un serveur à un autre en connexion point à point. Il s'agit d'un protocole fonctionnant en mode connecté, encapsulé dans une trame TCP/IP.

Le courrier est remis directement au serveur de courrier du destinataire. SMTP fonctionne grâce à des commandes textuelles envoyées au serveur SMTP (par défaut sur le port 25). Chacune des commandes envoyées par le client (validée par la chaîne de caractères ASCII, équivalent à un appui sur la touche entrée) est suivie d'une réponse du serveur SMTP composée d'un numéro et d'un message descriptif [57].

III.4 Bilan de l'Analyse

L'analyse effectuée a permis de mieux comprendre le fonctionnement des principaux standards, technologies et protocoles qui peuvent être ciblés dans la conception et le développement de notre application. Mais, cette analyse a surtout révélé certains choix effectués selon nos besoins, nos moyens et notre contexte.

Après avoir étudié les différentes technologies, standards et protocoles disponibles dans le monde des communications sans fil et les environnements mobiles, un bilan d'analyse a été défini pour chaque étape de ce PFE suivant un cahier de charge adapté.

III.4.1 Support de Transmission

En termes d'exploitation des données envoyés par un Smartphone qui les traite par des algorithmes spécifiques et transmet ensuite les résultats du sportif, la solution la plus adaptée consiste en de simples téléphones portables (*Middleware*) liés par un système de transmission étendu 3G ou 4G.

L'exploitation du canal 3G ou 4G nous a permis de transmettre les résultats quotidiens du sportif au coach. Ce sont des technologies disponibles et rapides.

III.4.2 Communication via Internet

Le choix du protocole http est dicté par les caractéristiques suivantes :

- ❖ http est indépendant du réseau.
- ❖ Le port du protocole http est plus facilement ouvrable sur les pare-feu.

Remarque : La version https de ce protocole peut être exploitée pour échanger et communiquer des informations confidentielles.

III.4.3 Outils de développement

III.4.3.1 Environnement matériel

Pour développer notre projet nous avons besoin d'un ordinateur qui aura au minimum les caractéristiques suivantes :

- ❖ Système d'exploitation : Microsoft® Windows® 8/10 64 bits.
- ❖ CPU : Architecture CPU x86_64 ; Intel Core de 2^{ème} génération ou plus récent, ou CPU AMD avec prise en charge d'un hyperviseur Windows.
- ❖ RAM : 8 Go ou plus.
- ❖ Stockage : 8Go disponible minimum.
- ❖ Résolution d'écran minimal : 1280 x 800.

Et pour lancer notre application on a besoin d'un terminal mobile qui aura au minimum :

- ❖ Android Version : Android 6.0 (Marshmallow).
- ❖ API Level : 23.

III.4.3.2 Environnement logiciel

Dans notre projet nous avons choisi de travailler avec l'IDE Android Studio en raisons :

- ❖ Android Studio apporte un confort d'utilisation non négligeable face à son concurrent direct, ADT Eclipse.
- ❖ Android Studio est IDE officiel pour le développement d'applications Android.
- ❖ En plus du puissant éditeur de code et des outils de développement d'IntelliJ, Android Studio offre encore plus de fonctionnalités qui améliorent notre productivité lors de la création d'applications Android.

Des outils de *Java* ont été implémentés sous l'environnement Android Studio IDE.

Donc, le choix du *Java* est justifié par les problèmes suivants associés au codage en *Kotlin* :

- ❖ **Management de la mémoire** : avec *Java*, La mémoire est gérée automatiquement par JVM. Alors qu'en *Kotlin*, La consommation de mémoire est élevée.
- ❖ **Sécurité** : *Java* est plus sécurisé.
- ❖ **Vitesse de compilation** : La vitesse de compilation de *Java* est plus rapide que *Kotlin*.

C'est donc pour ces raisons que *Java* a été choisie pour la réalisation des travaux de ce projet. Mais il faut souligner que les deux langages se valent plus ou moins, et qu'il aurait probablement été possible de travailler sur les deux plateformes et d'obtenir le même résultat.

Les applications mobiles ont généralement besoin de services back-end pour fonctionner avec l'interface utilisateur front-end. Cela signifie qu'il doit y avoir du code et des serveurs back-end pour travailler avec les applications mobiles. Firebase est un service en Cloud qui permet d'alimenter le back-end des applications. Firebase peut prendre en charge le stockage des données et l'authentification des utilisateurs. Après avoir intégré les applications mobiles avec Firebase, nous n'avons plus besoin d'écrire du code back-end ou de gérer l'infrastructure [58].

Donc, le choix de Firebase est justifié par les caractéristiques suivantes :

- ❖ Firebase nous fait gagner du temps.
- ❖ Notre code de base reste plus petit.
- ❖ Nos bugs sont moins nombreux.

III.4.4 OS Mobile

L'utilisation du système d'exploitation Android est généralement le plus répandu pour les Smartphones. Il occupe une plus grande part de marché (environ 77 %) par rapport à Apple (environ 20 %). En plus l'Android est open source (gratuit) et il a plus de fonctionnalités avec une plus grande marge de manœuvre. Un autre avantage pour Android permet un stockage amovible tel que des cartes mémoire, les utilisateurs de cet OS peuvent facilement ajouter de mémoire supplémentaire pour répondre à leurs besoins.

III.5 Exécution

III.5.1 Application

- Pour lancer l'application cliquer sur le logo (**Figure III.2**)



Figure III.2 : logo de l'application

Abou-Bekr Belkaid University
TLEMCEM
Faculty of Technology
Department of Telecommunications



Theme :
Coach Des Sportifs



Developed by :

GHOMRI BENMEZIANE IMAD-DDINE

EL HADJ MIMOUN HICHAM

Framed by :

Pr. MERZOUGUI RACHID

START

Figure III.3 : *Interface page de garde*

- Cliquer sur « Start » pour entrer dans l'application.

Interface d'inscriptions

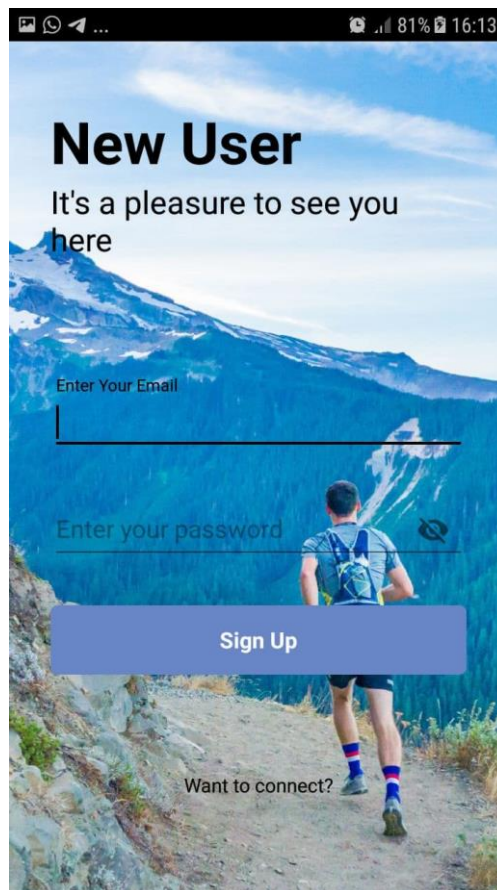


Figure III.4 : *Interface Inscription*

- Pour un nouveau utilisateur il faut entrer l'e-mail et créer le mot de passe, ensuite une étape de vérification du compte depuis l'adresse email saisie.

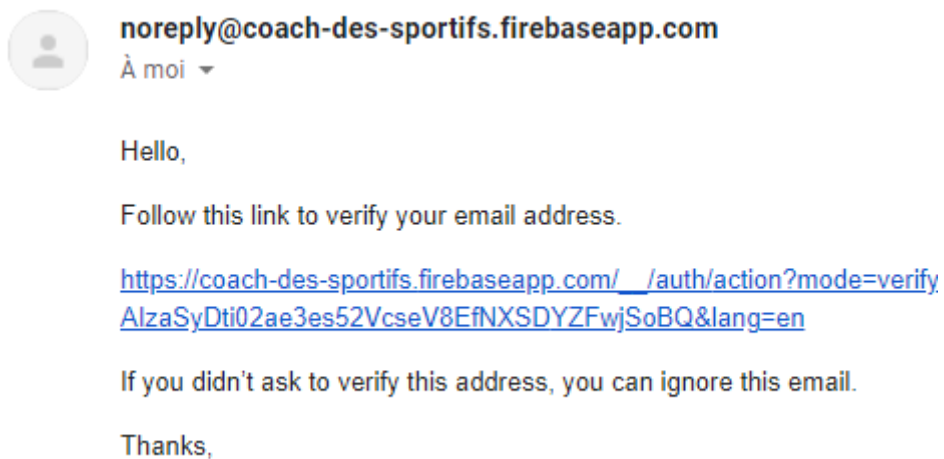


Figure III.5 : *Email de vérification*

Interface d'authentification

Pour que l'utilisateur puisse utiliser l'application, il doit d'abord s'authentifier en introduisant son identifiant et son mot de passe.

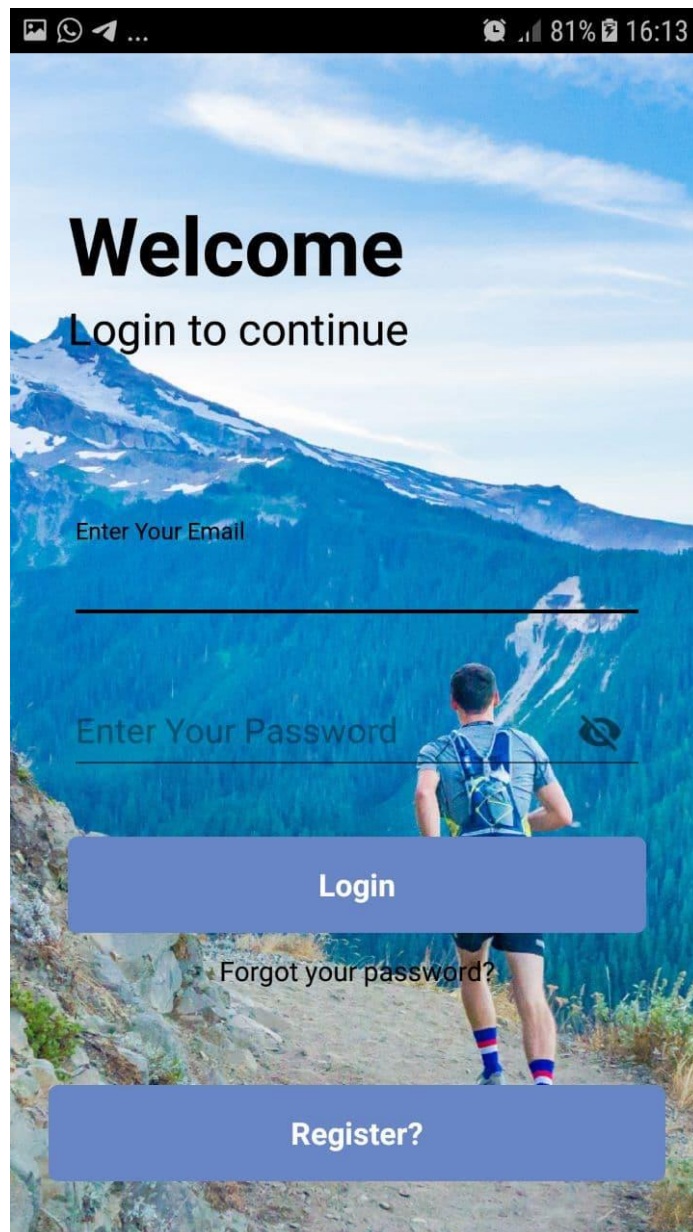


Figure III.6 : *Interface Authentification*

- Cliquer sur le bouton « Login » pour connecter.
- Si vous avez oublié votre mot de passe cliquer sur « Forgot your password ? ».
- Si vous voulez inscrire à l'application cliquer sur « Register ? ».

Interface d'oublié le mot de passe :

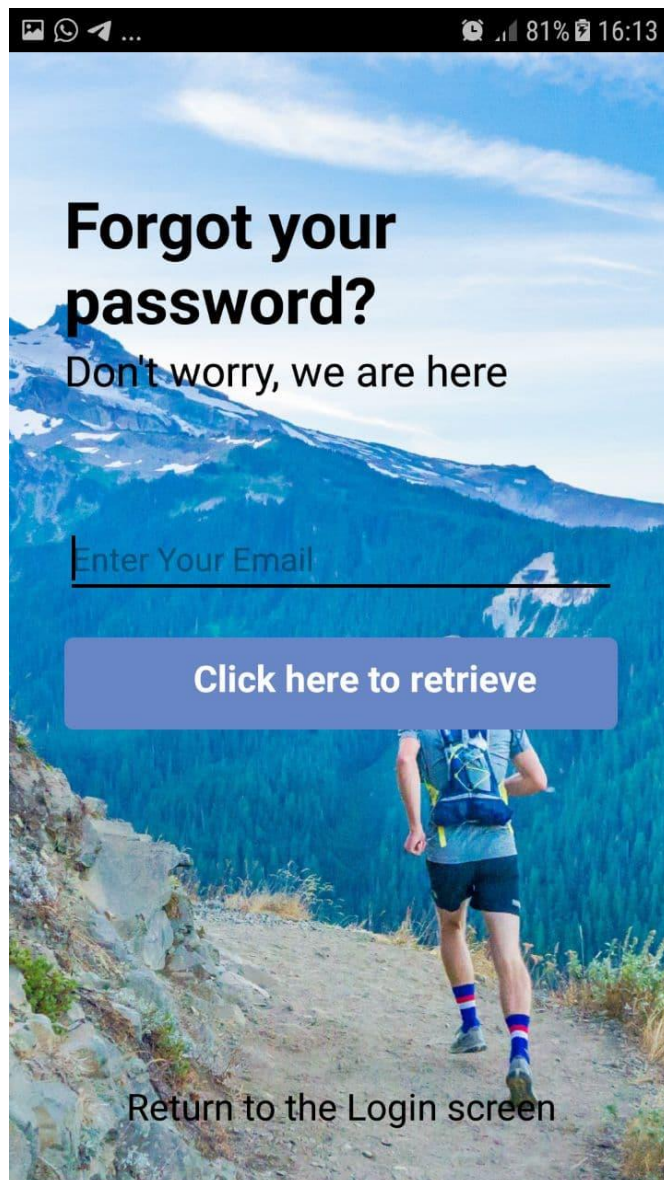
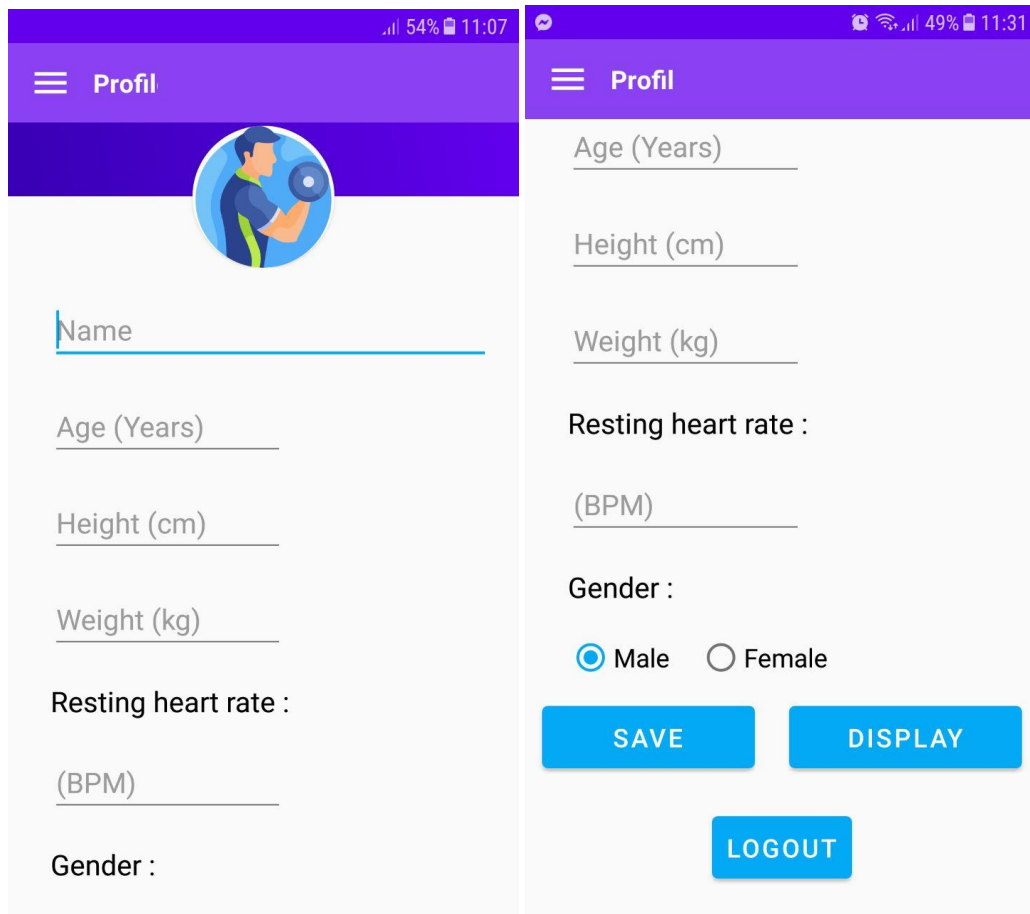


Figure III.7 : *Interface oublié mot de passe*

- Pour récupérer votre mot de passe entrer votre e-mail puis cliquer sur le bouton « click here to retrieve ».
- Si non cliquer sur « Return to the Login screen » pour retourner à l'interface de connexion.


Interface Profil

Après l'authentification l'utilisateur accède à cette interface (**Figure III.8**), où l'utilisateur peut entrer ses informations qui peuvent être affichées.

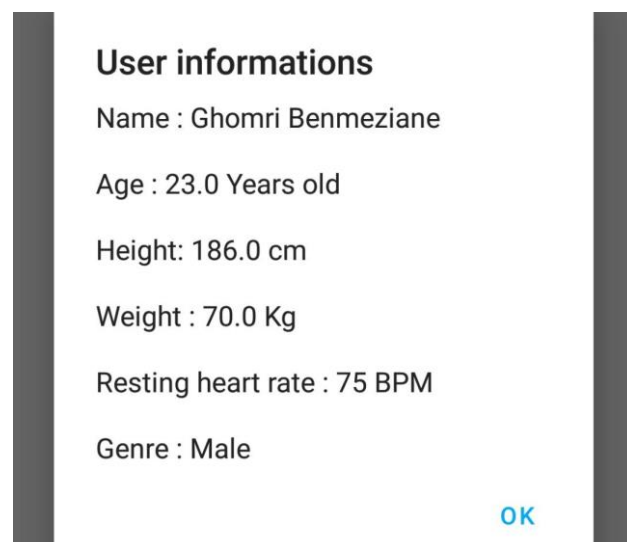


The image displays two screenshots of a mobile application's 'Profil' (Profile) screen. The left screenshot shows the input form with fields for Name, Age (Years), Height (cm), Weight (kg), Resting heart rate (BPM), and Gender (Male/Female). The right screenshot shows the same form with the 'SAVE', 'DISPLAY', and 'LOGOUT' buttons visible.

Figure III.8 : *Interface profil*

- Entrer vos informations (nom, âge, taille, poids, fréquence cardiaque de repos, genre).
- Pour enregistrer vos informations cliquer sur « save ».
- Pour afficher vos informations cliquer sur « display ».
- Pour entrer à la barre de navigation cliquer sur 
- Pour déconnecter cliquer sur « Logout ».

Interface Informations d'utilisateur



The image shows a dialog box titled 'User informations' displaying the user's profile data: Name: Ghomri Benmeziane, Age: 23.0 Years old, Height: 186.0 cm, Weight: 70.0 Kg, Resting heart rate: 75 BPM, Genre: Male. An 'OK' button is visible at the bottom right.

Figure III.9 : *Interface Informations d'utilisateur*

Interface Barre de navigation

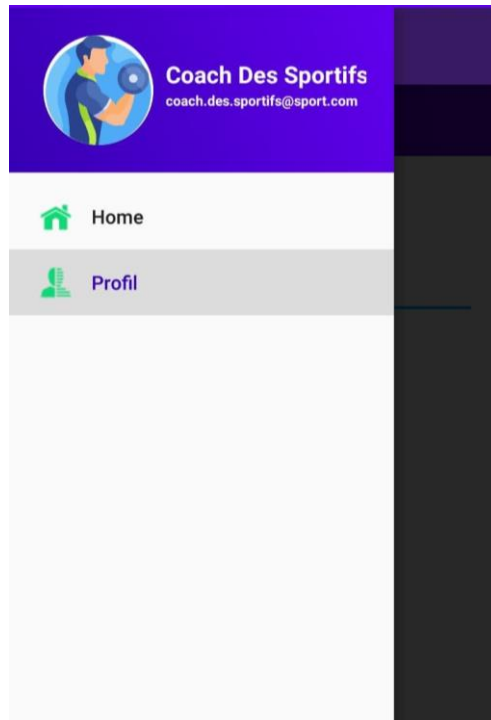


Figure III.10 : *Interface Barre de navigation*

- Pour accéder à l'accueil cliquer sur « Home ».
- Pour accéder au profil cliquer sur « Profil ».

Interface d'accueil

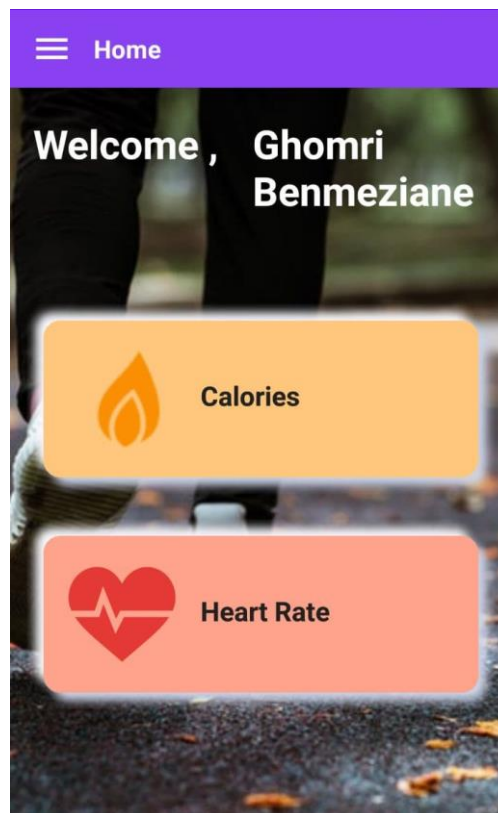


Figure III.11 : *Interface d'Accueil*

- Pour accéder à l'activité calories cliquer sur « Calories ».
- Pour accéder à l'activité fréquence cardiaque cliquer sur « Heart Rate ».

Interfaces Calories

- Sélectionner le niveau de l'activité physique pour voir votre besoin énergétique.

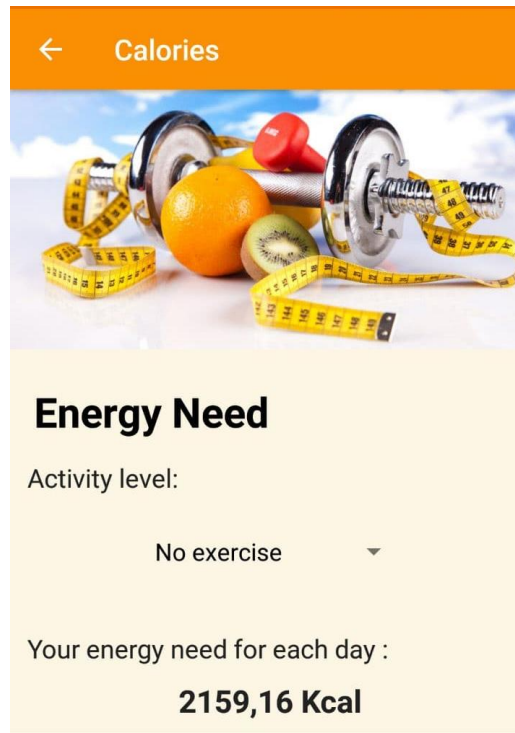


Figure III.12 : 1^{ère} partie de l'activité calories

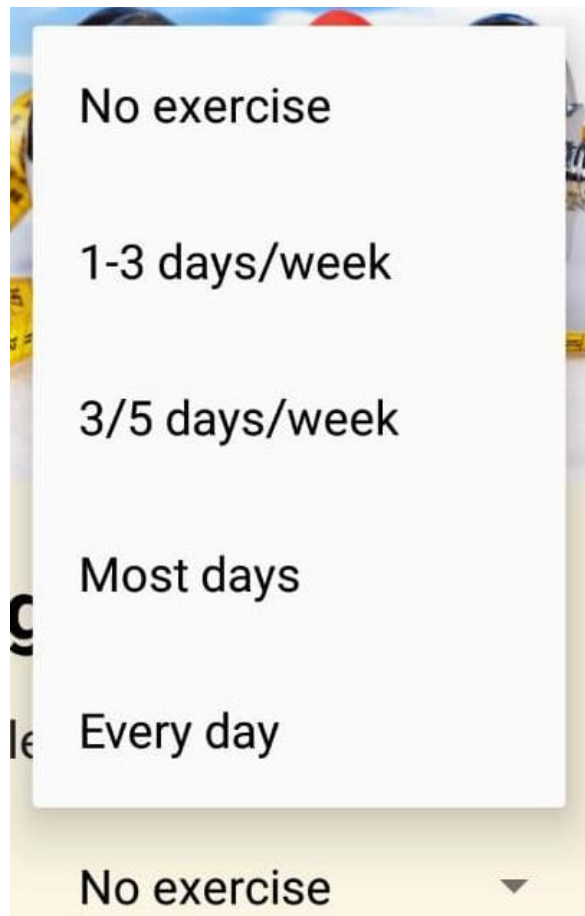


Figure III.13 : Liste des choix de niveaux d'activité physique

- Sur cette interface :
Choisi le type du sport que vous voulez le pratiquer, contrôle le chronomètre, calcule les calories brûlées et envoie les résultats au coach.

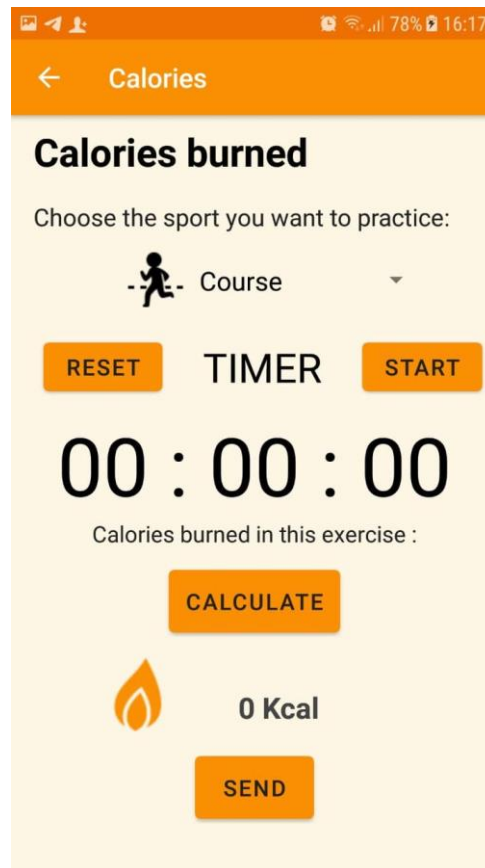


Figure III.14 : 2^{ème} partie de l'activité calories

- Choisissez le sport que vous allez pratiquer.
- Pour lancer le chronomètre cliquer sur le bouton « start », et cliquer sur « pause » pour le mettre en pause, et cliquer sur « Reset » pour le réinitialiser.
- Pour calculer les calories brûlées pendant votre activité sportive cliquer sur « calculate ».
- Pour envoyer vos résultats (calories brûlées, fréquence cardiaque moyenne, le sport, la durée, la date) cliquer sur « send ».

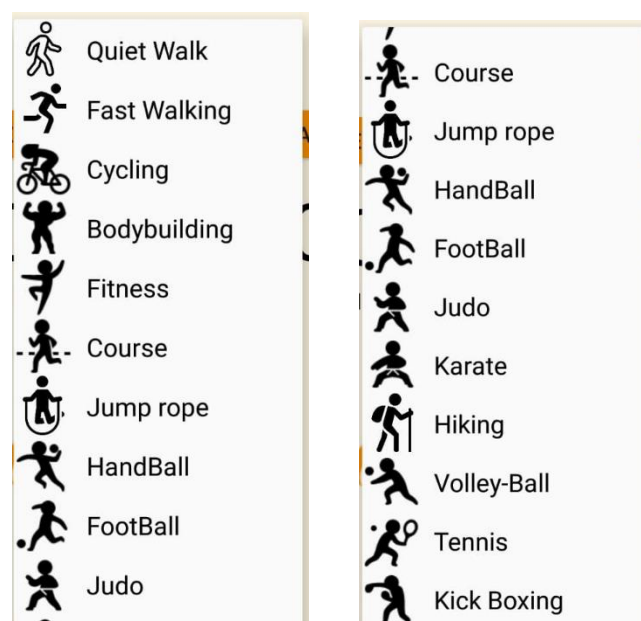


Figure III.15 : Liste des choix des sports

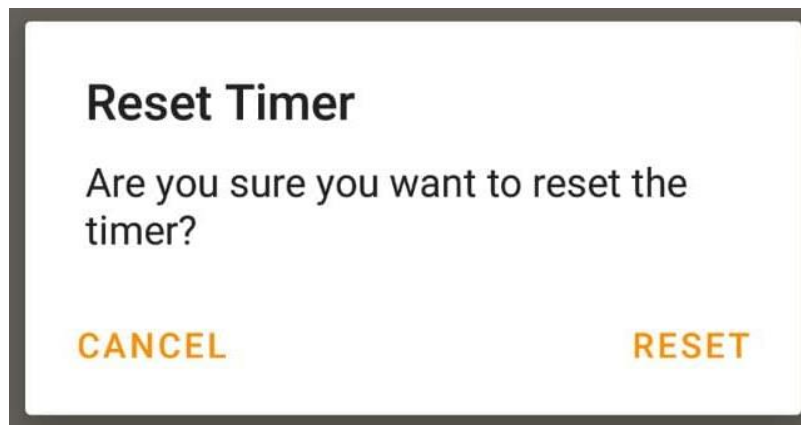


Figure III.16 : Boite de dialogue pour confirmer la réinitialisation de chronomètre

- Enfin, vous aurez un graphe qui présente vos calories brulées pendant une semaine.

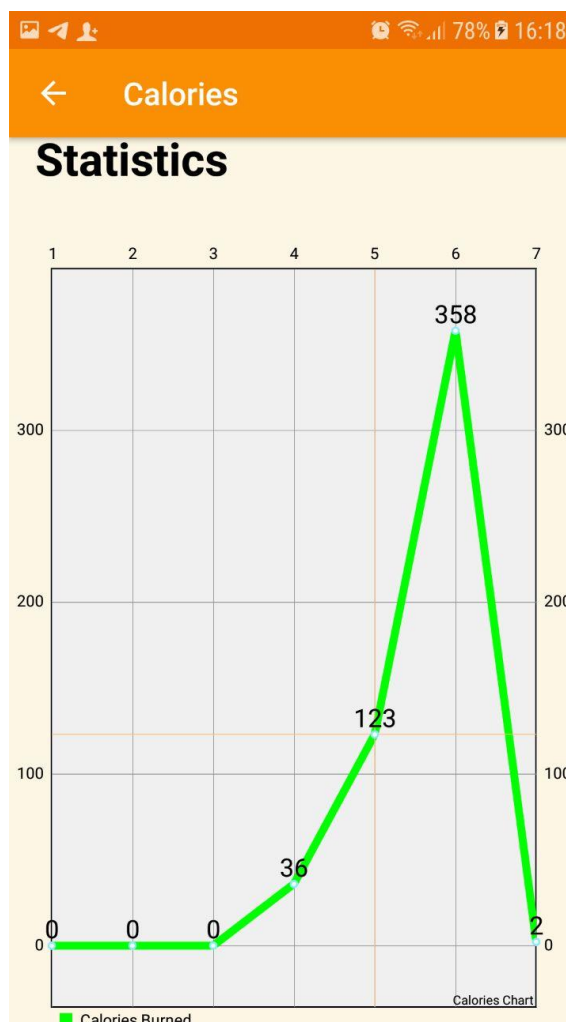


Figure III.17 : Représentation graphique des calories brulées pendant une semaine

Interfaces Fréquence cardiaque

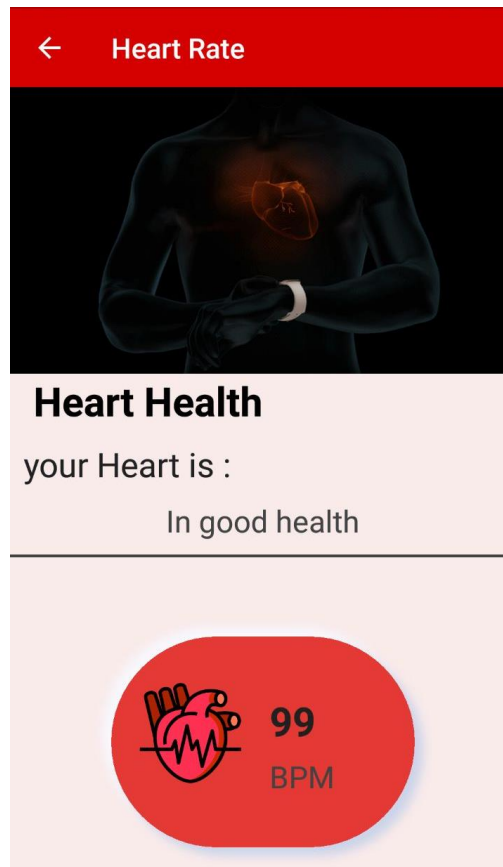


Figure III.18 : 1^{ère} partie de l'activité fréquence cardiaque

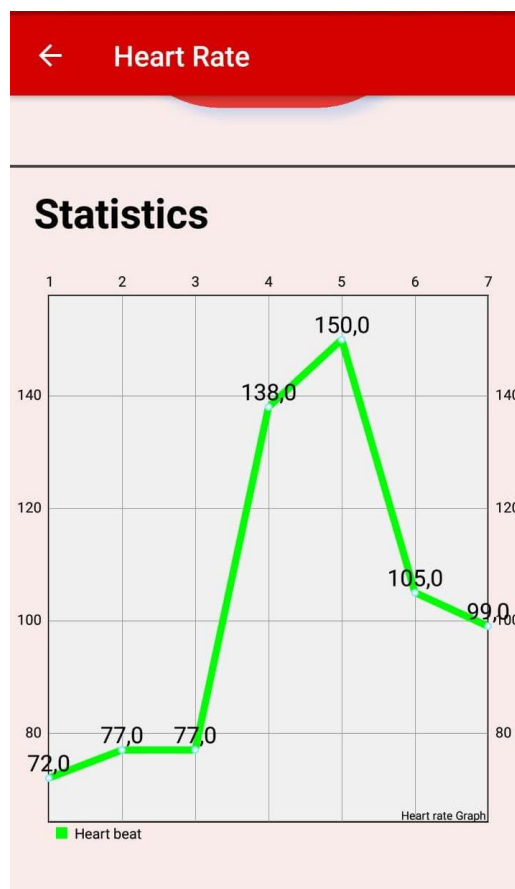


Figure III.19 : Représentation graphique de fréquence cardiaque pendant une semaine

- Voir la santé cardiaque.
- Voir votre fréquence cardiaque moyenne de jour.
- Enfin, vous aurez un graphe qui affiche votre fréquence cardiaque pendant une semaine.

III.5.2 Firebase

Dans notre projet, Nous avons utilisé deux services de Firebase :

❖ **Realtime Database :**

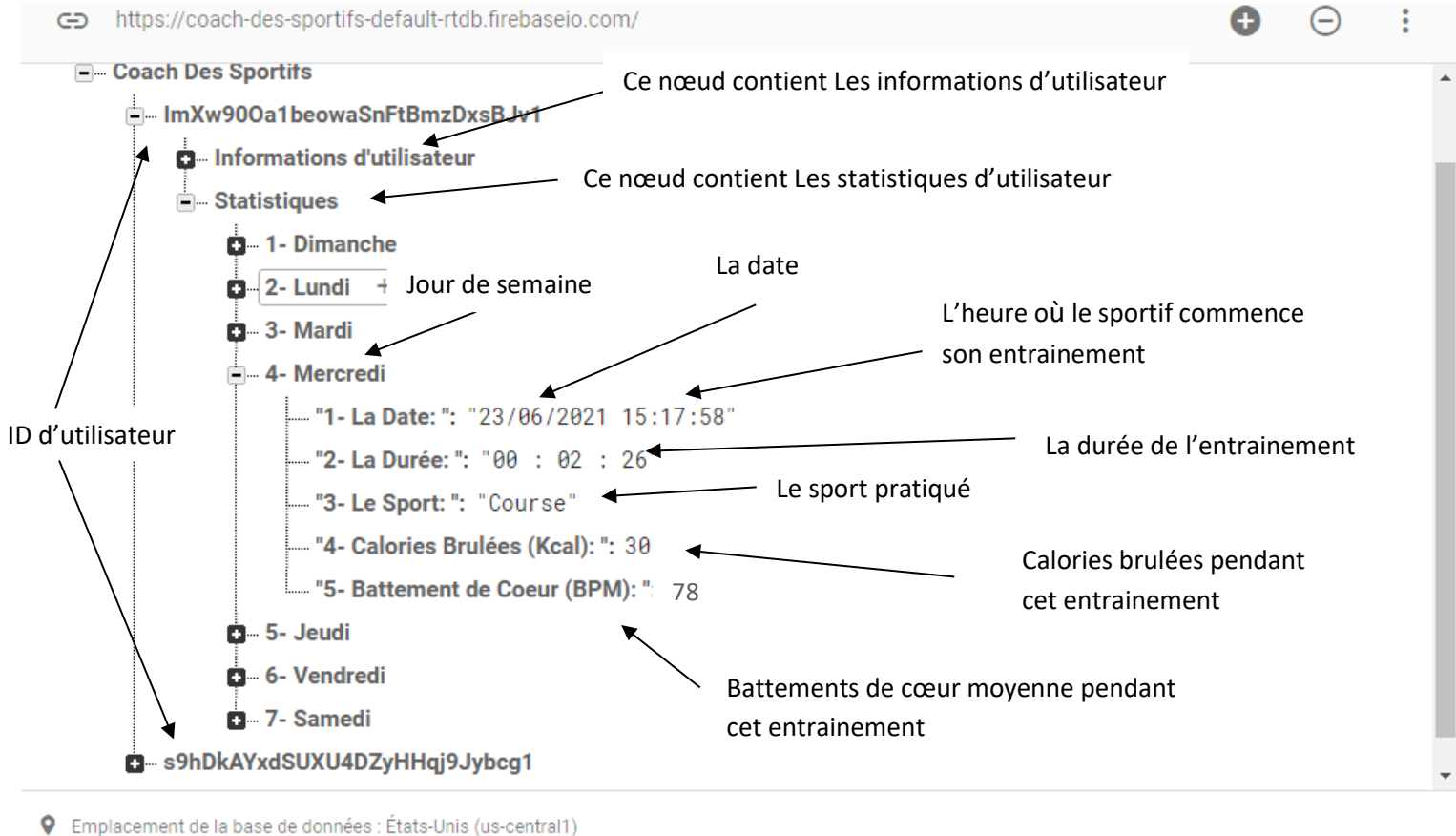


Figure III.20 : Firebase Realtime Database

❖ **Firebase Authentication :**

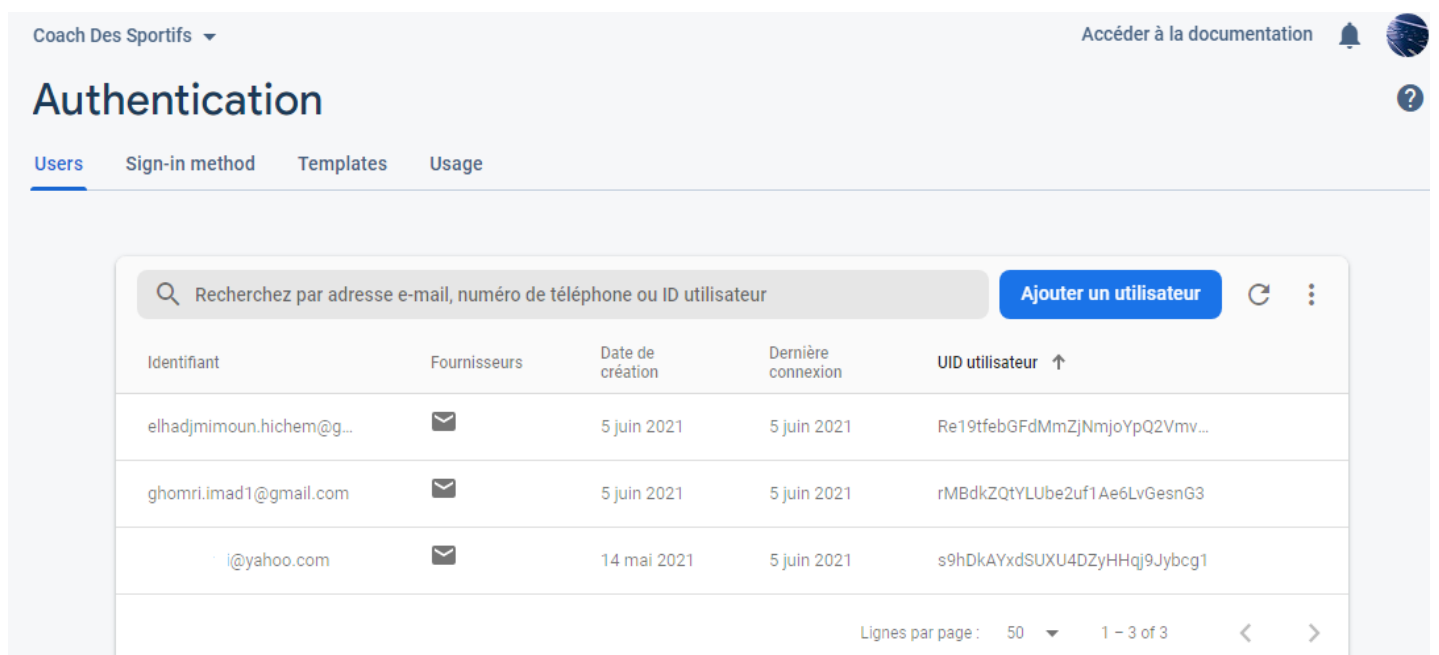


Figure III.21 : Firebase Authentication

III.6 Conclusion

Nous avons présenté dans ce chapitre, un panorama des standards, technologies et protocoles liés à notre plateforme mobile avec un bilan de l'analyse justifiant les choix selon nos besoins et le contexte ciblé. Enfin, on terminera le chapitre par exposer les différentes étapes d'exécution de l'application réalisée.

Conclusion Générale et Perspectives

L'objectif de ce projet est de réaliser une application sportive pour aider et contrôler les gens que ce soit homme et femme à pratiquer les activités sportives de façon fiables et efficaces. Cette application permet aux sportifs de rester en forme et en performances. Auparavant un tel projet aurait nécessité un grand budget ainsi qu'une grosse infrastructure aujourd'hui un simple Smartphone peut suffire.

Pour cela, nous avons conçu et implémenté cette application mobile sous Java de l'outil Android Studio avec la modélisation UML de l'architecture et les différentes étapes d'exécution.

Nous avons commencé par l'identification des besoins qu'on a ensuite modélisé sous, forme de diagrammes de cas d'utilisation, diagrammes de séquences, diagramme de classe et terminer par la réalisation de notre application. Lors de ce travail, Nous avons pu mettre en pratique nos connaissances théoriques acquises durant notre formation.

Toutes les étapes effectuées pendant la réalisation de ce projet nous a permis d'enrichir notre expérience notamment dans les différents outils et langages dédiés à la programmation sur mobile.

Bien que notre application ne soit pas encore finie, nous avons l'attention de la compléter et l'améliorer en termes des options en plus auprès du Coach, de l'implémentation des capteurs pour mesurer la vitesse instantanée et la fréquence cardiaque.

Références

- [1] «Premier téléphone portable : l'histoire de son invention !», 09 août 2019. [En ligne]. Available: <https://assurance.carrefour.fr/assurance-des-appareils-nomades/mobile/innovation/tout-savoir-sur-linvention-du-telephone-portable>. [Accès le 4 Avril 2021].
- [2] elmekki, «Le téléphone portable : histoire et évolutions», 14 septembre 2016. [En ligne]. Available: <https://www.gralon.net/articles/photo-et-video/telephonie-et-portables/article-le-telephone-portable---histoire-et-evolutions-1190.htm>. [Accès le 4 Avril 2021].
- [3] «Smartphone», [En ligne]. Available: https://fr.wikipedia.org/wiki/Smartphone#Années_2020_:_émergence_des_smartphones_pliants.
- [4] G. Manceau, «Meilleur smartphone Xiaomi - Comparatif 2021», 07 décembre 2020. [En ligne]. Available: <https://www.clubic.com/smartphone/android/article-873925-1-comparatif-meilleur-smartphone-xiaomi.html>.
- [5] «Smartphone de la marque Oppo : Avis et modèles», mars 2021. [En ligne]. Available: <https://www.iphonerevolt.org/smartphones-oppo/>.
- [6] «Meilleur smartphone par marque - Comparatif 2021», mars 2021. [En ligne]. Available: <https://www.clubic.com/mobilite-et-telephonie/article-879192-1-comparatif-marque-meilleur-smartphone.html>.
- [7] «De la 2G à 4G», [En ligne]. Available: <https://www.digitalcorner-wavestone.com/2020/01/de-la-2g-a-la-4g/>.
- [8] B. Fradelle, «Réseaux cellulaires mobiles», mars 2021. [En ligne]. Available: <https://www.tactis.fr/reseaux-cellulaires-mobiles/>.
- [9] A. R. Mishra, *Advanced Cellular Network Planning and Optimisation: 2G/2.5G/3G...Evolution to 4G*, Wiley; 1st edition, 2006.
- [10] «Qu'est-ce que la 5G? Caractéristiques et utilisations», [En ligne]. Available: <https://www.thalesgroup.com/fr/europe/france/dis/mobile/inspiration/5g>.
- [11] «Les systèmes d'exploitation des smartphones», [En ligne]. Available: <https://www.monpetitmobile.com/choisir-mobile/systemes-exploitation-smartphones>.
- [12] M. Narmatha et S. Venkata KrishnaKumar, «Study on Android Operating System And Its Versions», *International Journal of Scientific Engineering and Applied Science (IJSEAS)*, Février 2016.
- [13] Reto Meier et Ian Lake, *Professional Android*, Wrox; 4th edition, 2018.
- [14] «Platform Architecture», [En ligne]. Available: <https://developer.android.com/guide/platform> . [Accès le 29 Mai 2021].
- [15] Anmol Misra et Abhishek Dubey, *Android Security: Attacks and Defenses*, Auerbach Publications; 1st edition, 2013.
- [16] O. Cinar, *Android Quick APIs Reference*, Apress; 1st edition, 2015.

- [17] Rudi Cilibrasi et Nathan Blair, *Wearable Android: Android Wear and Google FIT App Development*, Wiley; 1st edition, Aout 2015.
- [18] Marc Zaffagni avec CNET.com, «Android : retour sur toutes les versions de l'OS Google,» 19 Février 2021. [En ligne]. Available: <https://www.cnetfrance.fr/news/android-retour-sur-toutes-les-versions-de-l-os-google-39872199.htm>.
- [19] R. Rafols, *Learning Android Application Development*, Packt Publishing , 2016.
- [20] olivier, 30 mai 2013. [En ligne]. Available: <https://www.e-naxos.com/Blog/post/Cross-plateforme-Android-Part-3-Activite-et-cycle-de-vie.aspx>.
- [21] John Horton, *Android Programming for Beginners*, Packt Publishing, 2015.
- [22] Daniel Jacoboson, Greg Brail et Dan Woods, «what is an API,» chez *APIs : A Strategy Guide*, O'Reilly Media, 1st edition, 2012, p. 4.
- [23] Camille Siegle et Arun Dorairajan, «What is an API,» [En ligne]. Available: <https://apifriends.com/api-management/what-is-an-api/>. [Accès le 25 Avril 2021].
- [24] P. Duhan, *M-Commerce: Experiencing the Phygital Retail*, Apple Academic Press; 1st edition, 2019.
- [25] T. Dahlberg, N. Mallat, J. Ondrus et A. Zmijewska, «Past, present and future of mobile payments research,» *Electronic Commerce Research and Applications*, 2007.
- [26] [En ligne]. Available: <https://www.j2store.org/extensions/paypal-express-checkout.html>.
- [27] Liao, Z., & Cheung, M. T., «Measuring consumer satisfaction in internet banking: a core framework,» *Communications of the ACM*, pp. 47-51, 2008.
- [28] Cheng, D., Liu, G., & Qian, C, «On Determinants of User Acceptance of Internet Banking: A Theoretical Framework and Empirical Study.In *Advanced Management of Information for Globalized Enterprises,*» *IEEE Symposium*, pp. 1-5, Septembre 2008.
- [29] Cheung, C. M., & Lee, M. K, «Consumer satisfaction with internet shopping: a research framework and propositions for future research.,» *The 7th international conference on Electronic commerce*, pp. 327-334, aout 2005.
- [30] Yee, B. Y., &Faziharudean, T. M., «Factors affecting customer loyalty of using Internet banking in Malaysia,» *Journal of Electronic Banking Systems*, p. 21, 2010.
- [31] 11 juin 2020. [En ligne]. Available: <http://www.tn24.tn/ar/article/paiement-mobile-la-bna-lance-son-application-digipay-by-bna-270458>.
- [32] Krishna S, Boren SA, Bales EA, «Healthcare via cell phones: A systematic review,» *Telemed J E Health*, vol. 15, p. 231–240, 2009.
- [33] hachi-tech, 10 mai 2021. [En ligne]. Available: <https://play.google.com/store/apps/details?id=com.hachitech.dz.esiha&hl=en&gl=US>.

- [34] 20 juin 2018. [En ligne]. Available: [https://www.bigstockphoto.com/fr/image-246296210/stock-photo-sankt-petersburg%2C-russia%2C-june-20%2C-2018%3A-alibaba-application-icon-on-apple-iphone-x-smartphone-scree/](https://www.bigstockphoto.com/fr/image-246296210/stock-photo-sankt-petersburg%2C-russia%2C-june-20%2C-2018%3A-alibaba-application-icon-on-apple-iphone-x-smartphone-scre/).
- [35] B.Bathelot, «Marketing mobile - Définitions Marketing,» 28 novembre 2016. [En ligne]. Available: <https://www.definitions-marketing.com/definition/marketing-mobile/>.
- [36] a. rawat, 30 avril 2020. [En ligne]. Available: <https://inc42.com/buzz/zoom-bluejeans-meet-their-match-as-google-makes-video-conferencing-free/>.
- [37] P. Ducellier, «Que signifie Visioconférence? - Definition IT de Whatis.fr,» décembre 2020. [En ligne]. Available: <https://www.lemagit.fr/definition/Visioconference>.
- [38] [En ligne]. Available: <http://webstar-electro.com/actualite/comprendre-les-capteurs-de-smartphones-et-leur-utilite-21819.htm>.
- [39] F. Espiau, «Les capteurs - Créez des applications pour Android,» 26 février 2020. [En ligne]. Available: <https://openclassrooms.com/fr/courses/2023346-creez-des-applications-pour-android/2028962-les-capteurs>.
- [40] t. desroches, 17 avril 2018. [En ligne]. Available: <https://monclubsportif.com/kb/appli-gestion-equipe-sport-monclubsportif-iphone-android-en-bref/>.
- [41] «Le marché des applications pour le sport - France,» 02 mars 2021. [En ligne]. Available: <https://www.businesscoot.com/fr/etude/le-marche-des-applications-pour-le-sport-france>.
- [42] Arief L.B et Speirs N.A, «Automatic generation of distributed system simulations from UML in Modelling and Simulation, A tool for next millennium,» *13-th European Simulation Multiconference*, vol. 2, pp. 507-509, 1-4 June 1999.
- [43] Kobryn Cris, «UML 2001 A Standardization Odyssey,» *Communications of the ACM*, vol. 42, n° 110, pp. 28-37, October 1999.
- [44] Mrozek Z, «UML as integration tool for design of the mechatronic system,» *Second Workshop on Robot Motion and Control*, pp. 189-194, Oct 18-20;2001.
- [45] M. Pristley, *Practical object-oriented design Using UML*, McGraw-Hill Education; 2nd edition, 2003.
- [46] «What is an Actor in UML?,» [En ligne]. Available: <https://t2informatik.de/en/smartpedia/actor/>. [Accès le 4 Mai 2021].
- [47] Zhiming Liu, *Object-Oriented Software Development Using UML*, UNU/IIST (United Nations University International Institute for Software Technology), Mars 2001.
- [48] «Use Case Relationships,» [En ligne]. Available: <https://www.w3computing.com/systemsanalysis/use-case-relationships/>. [Accès le 5 Mai 2021].
- [49] Arti Mishra et Shikha Arya, «Role of UML in Object Oriented Design,» *INTERNATIONAL JOURNAL FOR RESEARCH & DEVELOPMENT IN TECHNOLOGY*, vol. 6, pp. 49-52, Octobre 2016.

- [50] Grady Booch, Games Rumbgh et Ivar Jacobson, «Sequence Diagrams,» chez *The Unified Modeling Language*, Addison-Wesley Professional; 2nd edition, 1999, pp. 75-80.
- [51] Falk Fraikin et Thomas Leonhardt , «SeDiTeC – Testing Based on Sequence Diagrams,» KBSE, 2002.
- [52] H. S. Oluwatosin, «Client-Server Model,» *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 16, pp. 67-71, Février 2014.
- [53] z. labri. [En ligne]. Available: <https://www.labri.fr/perso/zemmari/m1dfac/c3.pdf>.
- [54] Chunnu Khawas et Pritam Shah, «Application of Firebase in Android App Development-A Study,» *International Journal of Computer Applications*, vol. 179, n° %146, June 2018.
- [55] Laurence Moroney, *The Definitive Guide to Firebase*, Apress®, 2017.
- [56] A. TANENBAUM, Réseaux, 3eme Édition DUNOD, 1996.
- [57] G. PUJOLLE, *Les Réseaux*, Paris, France: Éditions EYROLLES, 2000.
- [58] F. Cheng, «Firebase,» chez *Build Mobile Apps with Ionic 2 and Firebase: Hybrid Mobile App Development*, Sandringham, Auckland, New Zealand, Apress®, 2017, p. 4.