

Mémoire de fin d'étude

Pour l'obtention du diplôme de master En informatique.

Options : Systèmes d'informations

Thème :

Conception et Réalisation d'une Application Android : Handyman

Présenté par :

Brahimi Fatma Zohra
Moussi Roqiya

Devant le jury :

Mr. Atig Yahia	Encadreur
Mr. Sidaoui Boutkhil	Président
Mr. Yahiaoui Yasser	Examineur

Année universitaire : 2021 – 2022

Remerciement

-Nous remercions premièrement le Dieu pour nous donner la puissance pour accomplir ce travail.

-A notre encadreur professionnel Mr. Atiq Yahia, pour tout le temps que vous accorder-nous et pour tous les conseils que vous nous a prodigués.

-A nos parents pour leurs soins et leurs instructions si précieux et leur grande préoccupation pour nous.

- Je tiens à exprimer mes remerciements aux membres de jury, qui ont accepté d'évaluer notre travail de thèse.

- A vous nos frères et sœurs ainsi qu'à nos ami (e)s, et à tous ceux qui nous avons soutenu et nos soutiennent encore. Du fond du cœur Merci, et que Dieu vous protège.

Dédicace

À Nos très chers parents qui nous ont fourni au quotidien un soutien et une confiance,
Que dieu vous protège et vous garde pour nous.

À nos chers frères et nos précieuses sœurs, les mots ne peuvent résumer notre reconnaissance
et notre amour à votre égard.

À tous les membres de nos famille (la famille Brahimi et la famille Moussi)

À nos adorables amies, pour votre fidélité et votre soutien

À tous nos amis avec lesquels nous avons partagé nos moments de joie et de bonheur.

À tout nos enseignants pour votre soutien, votre enseignement et vos conseils tout au long
de notre parcours éducatif et professionnel.

TABLE DES MATIÈRE

Résumé	6
Chapitre 1: Introduction	7
Chapitre 2 :Préliminaires	9
2.1. Introduction	9
2.2. Les applications mobiles	9
2.2.1.Définition.....	9
2.2.2.Type de des applications mobiles.....	9
2.2.3.Composants des applications Android	10
2.2.4.Caractéristiques des applications.....	11
2.2.5.Inconvénients et Avantages des Applications Mobiles	11
2.3. Systèmes d'exploitation mobile	12
2.3.1.Définition.....	12
2.3.2.Principaux systèmes d'exploitation.....	12
2.3.3.Découverte de la plateforme Android	13
2.4. Conclusion.....	17
Chapitre 3: Etat de l'art	18
3.1. Introduction	18
3.2. Etude de l'existant.....	18
3.3. Applications Algériennes	18
3.3.1.Tashil	18
3.3.2.Fixit.....	18
3.3.3.Youchoz.....	19
3.4. Analyse de l'existant (Marché local)	19
3.5. Comparaison entre les applications Algériennes(Tashil et Fixit et Youchoz)	19
3.6. Applications internationales.....	20
3.6.1.Thumbtack.....	20
3.6.2.Taskrabbt	20
3.8. Critiques d'applications locales/internationales.....	20
3.9. Conclusion.....	20
Chapitre 4: Conception	21
4.1. Introduction	21
4.2. Modélisation des besoins avec UML	21
4.2.1.Définition de UML.....	21
4.2.2.Les diagrammes de UML	21
4.2.3.Diagramme des casd'utilisation	23

4.2.4.Diagramme de classe de l'application « HANDYMAN »	25
4.2.5.Diagrammes de séquences.....	27
4.3. Conclusion.....	32
Chapitre 5:Implémentation	33
5.1. Introduction	33
5.2. Environnements de travail	33
5.2.1.Les outils utilisés	33
5.2.2.Les langages utilisés	34
5.3. Description de l'application	35
5.3.1.Logo de l'application	35
5.3.2.Interface de Politique de confidentialité.....	36
5.3.3.Espace d'administrateur	36
5.3.4.Espace Client	40
5.3.5.Espace Bricoleur.....	46
5.4. Conclusion.....	48
Chapitre 6 : Conclusion et perspectives	49
Bibliographie	50

TABLE DES FIGURES

Figure 2.1 Les applications mobile.....	9
Figure 2.2 Architecture du logiciel Android.....	13
Figure 2.3 Cycle de vie d'une activité d'Android.....	16
Figure 4.1 Diagramme de cas d'utilisation.....	24
Figure 4.2 Diagramme de classe d'application.....	26
Figure 4.3 Diagramme de séquence "Authentification".	27
Figure 4.4 Diagramme de séquence "Gérer compte (ajouter)".	28
Figure 4.5 Diagramme de séquence "Gérer compte (modifier)".	29
Figure 4.6 Diagramme de séquence "Gérer compte (supprimer)".	29
Figure 4.7 Diagramme de séquence "prendre rendez-vous".	30
Figure 4.8 Diagramme de séquence "confirmer bricoleur".	31
Figure 4.9 Diagramme de séquence "recherche d'un bricoleur".	32
Figure 5.1 Logo de l'application.	35
Figure 5.2 Interface de Politique de confidentialité de l'application.....	36
Figure 5.3 L'interface d'inscription d'administrateur	36
Figure 5.4 L'interface d'Authentification d'administrateur	36
Figure 5.5 L'interface d'accueil d'administrateur.	37
Figure 5.6 Interface de Liste des clients	37
Figure 5.7 Liste cliquable sur un client.....	37
Figure 5.8 Fiche d'informations d'un client.	37
Figure 5.9 Interface de liste des catégories.	38
Figure 5.10 Interface d'ajouter une catégorie.	38
Figure 5.11 Interface de liste des bricoleurs non confirmés.	39

Figure 5.12 Interface de liste des bricoleurs.	39
Figure 5.13 Interface de liste des bricoleurs confirmés	39
Figure 5.14 Fiche d'information.	39
Figure 5.15 Interface de téléchargement le diplôme.....	39
Figure 5.16 Interface de recherche par l'administrateur.....	40
Figure 5.17 Interface de recherche par Nom de bricoleur.	40
Figure 5.18 Interface de choix.	40
Figure 5.19 Interface d'inscription de client.....	41
Figure 5.20 Interface d'Authentification de client.	41
Figure 5.21 Interface des service bricolage.	42
Figure 5.22 Interface de catégorie "Electric".....	42
Figure 5.23 Interface de Google maps personnaliser.....	43
Figure 5.24 Interface de profil de client.....	43
Figure 5.25 Interface de recherche par client.....	44
Figure 5.26 Interface de profil de bricoleur.	44
Figure 5.27 Interface de l'Appelle téléphonique	44
Figure 5.28 Interface de message d'attente d'approbation de rendez-vous.	45
Figure 5.29 Interface de rendez-vous (jour et heure) Envoyé par le bricoleur.	45
Figure 5.30 Interface d'inscription de bricoleur.	46
Figure 5.31 Interface de message d'attente de confirmation.....	46
Figure 5.32 Interface d'Authentification de bricoleur après la confirmation d'admin.	46
Figure 5.33 Interface de bricoleur.....	46
Figure 5.34 Interface de liste de demandes des rendez-vous.....	47
Figure 5.35 Interface de fixer une date et l'heure de rendez-vous.	47

LISTE DES TABLEAUX

Tableau 2.1 comparaison entre Version noyau Android et Version noyau linux	14
Tableau 3.2 Comparaison entre les applications (Tashil et Fixit et Youchoz).....	19

Résumé

Le but de ce travail est de réaliser une application mobile fiable pour améliorer et faciliter l'accès aux services de bricolage à domicile.

Nous avons analysé les solutions existantes déjà dans le marché national algérien et international. Nous avons aussi comparé ces solutions entre eux et contre l'international. Pour la conception, UML est utilisé comme langage de modélisation.

Pour l'implémentation, nous avons utilisé l'environnement de développement intégré Android Studio qui nous a permis d'élaborer des interfaces ergonomiques et faciles à utiliser et PHP comme langages de programmation, un système de gestion de base de données Xampp. A la fin de ce travail un ensemble de questions ouvertes et de perspectives sont discutés pour les recherches et les améliorations futures.

Mots clés : Application Mobile, Android, Bricoleurs,

Abstract

The purpose of this work is to achieve a reliable mobile application to improve and facilitate access to home services.

We have analyzed the existing solutions already in the Algerian national and international market. We also compared these solutions against each other and against the international. For design, UML is used as the modeling language.

For the implementation, we used the environment integrated development Android Studio which allowed us to develop ergonomic and easy to use interfaces and PHP as programming languages, an Xampp database management system. At the end of this work a set of open questions and perspectives are discussed for future research and improvement.

Keywords: Mobile Application, Android, Handyman,

ملخص

يهدف هذا العمل إلى تحقيق تطبيق جوال موثوق به لتحسين وتسهيل الوصول إلى الخدمات المنزلية. لقد قمنا بتحليل الحلول الموجودة بالفعل في السوق الجزائري الوطني والدولي. قمنا أيضًا بمقارنة هذه الحلول ببعضها البعض ومع الحلول الدولية. بالنسبة للتصميم ، يتم استخدام UML كلغة نمذجة. من أجل التنفيذ ، استخدمنا بيئة التطوير المتكاملة Android Studio والتي سمحت لنا بتطوير واجهات مريحة وسهلة الاستخدام و PHP كلغات برمجة ، نظام إدارة قواعد بيانات Xampp. في نهاية هذا العمل تتم مناقشة مجموعة من الأسئلة ووجهات النظر المفتوحة للبحث والتحسين في المستقبل. الكلمات الرئيسية: تطبيق الهاتف المحمول ، اندرويد، العمال المهرة.

Chapitre 1 : Introduction

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 dotés de plusieurs applications pratiques.

L'utilisation des Smartphones a connu un grand essor ces dernières années, l'utilisation du téléphone est devenue l'une des choses nécessaires dont on ne peut pas se passer, grâce à la disponibilité d'un ensemble d'applications qui aident l'utilisateur dans divers domaines. Ces applications fournissent un ensemble de services dont il a besoin.

Les applications mobiles ont envahi notre quotidien. Consulter vos mails, jouer, parler à un ami, faire du shopping, consulter les actualités ou la météo, retrouver un trajet ou un hôtel, consulter vos comptes, vous pouvez le faire en un clic via votre téléphone portable.

Entreprises ou particuliers, nous avons tous (ou presque), aujourd'hui recours à ces applications qui font partie intégrante de notre vie et sans lesquelles, pour beaucoup, la vie serait presque plus compliquée, Facilité, rapidité, simplicité : c'est l'univers des applications mobiles.

Toutefois, ces applications sont peu nombreuses et leur développement est tardif en Algérie par rapport à d'autres pays. Parmi ces services, nous pouvons citer la recherche d'un bricoleur pour les services à domicile, ce service en Algérie est actuellement une tâche difficile, surtout suite à un voyage ou un déménagement. A vrai dire, il existe localement peu d'applications qui peuvent aider leurs utilisateurs à obtenir facilement des services à domicile.

Les services de bricolage sont des travaux dont les gens ne peuvent pas se passer. Cependant il est difficile de localiser ces services lorsque ces gens déménagent ou voyagent dans de nouvelles régions, car les fournisseurs de services demandés sont situés dans différentes régions et se différencient selon les coûts, la qualité et le type de service qu'ils fournissent.

Le but de ce travail est de proposer une solution à ce problème à travers une application Android qui aide les utilisateurs à la recherche d'un bricoleur et obtenir facilement des informations mais aussi les coordonnées des fournisseurs des services à domicile.

Dans notre application, nous avons pu connecter facilement le bricoleur avec le client, et les combiner dans une seule application. Le client peut rechercher le bricoleur et prendre rendez-vous avec lui ou le contacter, et le bricoleur peut ajouter son travail dans l'application et communiquer avec un client. Ainsi que la confirmation du bricoleur avant de publier son travail via l'administrateur de l'application pour une application plus fiable que les autres applications..

Chapitre 1 : Introduction

La suite de ce mémoire est structurée comme suit :

Dans le Chapitre II nous présentons les connaissances préliminaires nécessaires pour le développement de notre application. Ce chapitre introduit aussi le monde des applications mobile et leurs systèmes d'exploitation ainsi que le cycle de vie d'activité d'Android.

Le Chapitre III est une analyse des travaux similaires sur le marché local mais aussi sur l'international.

La partie conceptuelle est présentée dans le Chapitre IV. Cette conception nous permet par la suite d'achever une implémentation guidée de notre proposition.

Le chapitre V détaille l'implémentation et l'expérimentation de notre application

En fin, le chapitre VI présente une conclusion et quelques perspectives sont discutées pour clôturer ce mémoire.

Chapitre 2 : Préliminaires

2.1.Introduction

Les progrès récents survenus dans les technologies mobiles ont permis la réalisation de multiples appareils mobiles prenant de plus en plus. Les projets d'applications mobiles sont devenus un moyen essentiel de création de nouveaux services. Dans ce chapitre, nous présenterons un ensemble de définitions sur les applications mobiles (Section 1.2) et les systèmes d'exploitation d'Android (Section 1.3).

2.2.Les applications mobiles

2.2.1. Définition

Une application mobile est une application logicielle développée spécifiquement pour une utilisation sur de petits appareils informatiques sans fil, tels que les Smartphones et les tablettes, plutôt que sur des ordinateurs de bureau ou des ordinateurs portables.

Les applications mobiles sont conçues pour tenir compte des exigences et des contraintes des appareils et pour tirer parti de toutes les fonctionnalités spécialisées dont ils disposent. De nombreuses sociétés proposent désormais la création d'applications mobiles pour vous aider à développer votre marque. Les applications mobiles sont parfois classées selon le fait qu'elles soient basées sur le Web ou natives. Une troisième catégorie, les applications hybrides, combine des éléments d'applications natives et Web. [1]

2.2.2. Type de des applications mobiles

Il existe trois familles d'applications mobiles : les applications natives, les applications Web, les applications hybrides [2], la figure 2.1 illustre ces trois familles :

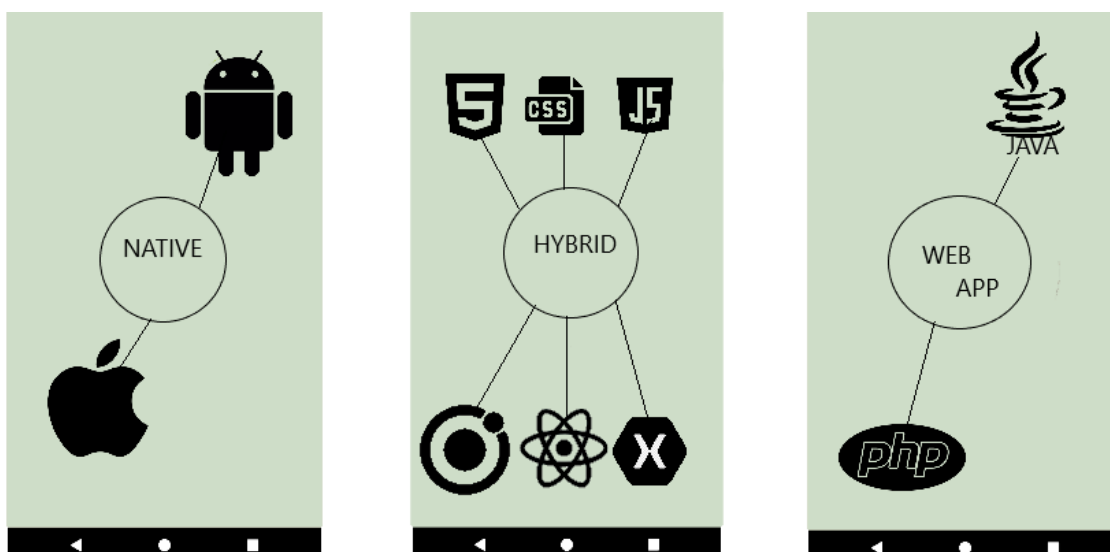


Figure 2.1 Les applications mobile.

Chapitre 2 : Préliminaires

- **Les applications Natives**

Les applications mobiles Natives sont développées spécifiquement pour un seul système d'exploitation : iOS ou Android. Ces applications requièrent exclusivement le langage « natif », plus généralement le langage JAVA pour Android ou le langage Objective-C/Swift pour iOS.

- **applications Web**

Les Web App Responsive correspondent à des sites Web conçus spécialement pour un affichage optimisé sur mobile et tablette. Ces applications mobiles sont développées principalement à partir de technologies Web comme le HTML5, Javascript ou encore CSS3.

- **Les applications Hybrid**

Les applications de type hybride sont développées à partir des mêmes technologies que les Web App pour s'adapter sur toutes les plateformes mobiles et pour les deux systèmes d'exploitation iOS & Android. Ces applications sont principalement développées à l'aide de HTML5 mais aussi d'autres langages Web comme le CSS et le JavaScript.

2.2.3. Composants des applications Android

Une application Android est composée des éléments basiques suivants [3]:

- **Activities** (Activités en Français) :

Une activité est la composante principale pour une application Android. Elle représente l'implémentation et les interactions de vos interfaces.

- **Services** :

Un service, à la différence d'une activité, ne possède pas de vue mais permet l'exécution d'un algorithme sur un temps indéfini. Il ne s'arrêtera que lorsque la tâche est finie ou que son exécution est arrêtée.

- **Broadcast and IntentReceivers** :

Un Broadcast Receiver comme son nom l'indique permet d'écouter ce qui se passe sur le système ou sur votre application et déclencher une action que vous aurez prédéfinie. C'est souvent par ce mécanisme que les services sont lancés.

- **Content providers** :

Les "content providers" servent à accéder à des données depuis votre application. Vous pouvez accéder aux contacts stockés dans le téléphone à l'agenda, aux photos ainsi que d'autres données depuis votre application grâce aux content providers.

- **Les Intents** :

Les composantes Android communiquent via des messages système que l'on appelle intent. Ils sont émis par le terminal pour prévenir les différentes applications du déclenchement d'événements.

Chapitre 2 : Préliminaires

2.2.4. Caractéristiques des applications

1. Système d'évaluation :

L'utilisateur peut évaluer application dans tous les magasins d'applications et peut rédiger une critique rapide de ce qu'il aime et n'aime pas à propos de l'application. En lisant ces critiques, pouvez mettre à jour l'application avec les dernières fonctionnalités et la corriger de toute erreur. C'est l'une des fonctionnalités les plus importantes d'une application réussie.

2. La possibilité de personnalisation :

L'une des caractéristiques de l'application à succès que les utilisateurs apprécient beaucoup est la possibilité de personnalisation, comme le changement de couleurs et de polices. Offrir la possibilité de personnaliser avec l'option de réglage pour convenir à un large éventail de public. La chose la plus importante à propos de la personnalisation est les options de confidentialité qui intéressent de nombreux utilisateurs.

3. Convivialité: (facilité d'utilisation) :

La convivialité consiste simplement à développer l'application avec les fonctionnalités et les caractéristiques appropriées pour le bon utilisateur cible.

4. Simplicité :

Les principales fonctionnalités de l'application doivent être identifiées et fournies à l'utilisateur de manière simple et ne pas compliquer l'application avec des fonctionnalités pouvant alourdir l'utilisateur.

5. Navigation hors ligne :

Si application a besoin d'une connexion Internet pour être utilisée, pouvez profiter de la fourniture de ces services ou contenus hors ligne, ce qui augmentera l'interaction et l'engagement de l'utilisateur avec l'application.

6. Performances et rapidité :

La performance de l'application est un facteur majeur dans l'admiration de l'utilisateur pour l'application, c'est-à-dire qu'elle remplit toutes ses fonctions. En plus de la rapidité de réponse, lorsque l'utilisateur appuie sur un bouton, il est censé répondre en moins d'une seconde.

7. Facilité de navigation et de recherche :

Faciliter la navigation des utilisateurs et proposer des options de recherche dans l'application.

2.2.5. Inconvénients et Avantages des Applications Mobiles

Les applications mobiles présentent de nombreux avantages et inconvénients sont: [4]

Inconvénients

- La soumission aux normes et règles édictées par les sociétés des plateformes mobiles à savoir Apple, Google, Windows et autres.

Avantages

- Une parfaite ergonomie est assurée pour les applications mobiles en comparaison aux sites mobiles cela encourage les utilisateurs à demeurer fidèles aux applis. En effet, le développement d'application mobile tient compte la taille du Smartphone, le temps de chargement et autres paramètres.

Chapitre 2 : Préliminaires

- Les applications mobiles favorisent l'intégration des options de téléphone et ainsi, l'expérience utilisateur devient plus développée.

2.3. Systèmes d'exploitation mobile

2.3.1. Définition

Un système d'exploitation, abrégé parfois SE, (ou Operating System en anglais, que l'on rencontre souvent sous l'abréviation OS) représente l'ensemble des programmes qui pilote les différents composants (disque dur, écran, processeur, mémoire etc...) de l'appareil informatique et lui permet donc de fonctionner. Il permet donc de faire l'interface entre l'utilisateur et le matériel informatique. Il est d'ailleurs chargé en premier lors du démarrage de l'appareil. Le système d'exploitation permet de faire fonctionner les différents périphériques (carte son, carte graphique, souris, clavier etc...). L'utilisation de pilotes (drivers en anglais) permettent au système d'exploitation de communiquer et de donner des instructions aux périphériques qui ne pourraient fonctionner sans eux.[5]

2.3.2. Principaux systèmes d'exploitation

1. Android¹

Android est le système d'exploitation le plus répandu. Développé par Google, il n'est pas rattaché à une marque de Smartphones (contrairement à IOS). Vous avez donc un large choix de marques qui proposent ce système : Samsung, Acer, HTC, Alcatel, LG, Huawei, Sony... Les fabricants intègrent Android à leurs Smartphones et y apportent quelques modifications.

2. IOS²

Après Android, IOS est le deuxième système d'exploitation le plus répandu. Il équipe uniquement les « iPhone », c'est-à-dire les Smartphones de la marque Apple. Le premier iPhone fut commercialisé en 2007. Depuis, plusieurs modèles sont sortis, généralement accompagnés d'une nouvelle version du système d'exploitation. Par exemple, fin 2014, la sortie de l'iPhone 6 a coïncidé avec l'arrivée de l'IOS 8.

3. Windows Mobile³

Windows 10 Mobile (anciennement Windows Phone) est un système d'exploitation développé par Microsoft que l'on retrouve sur plusieurs marques comme Nokia, HTC et Acer...

Ce système d'exploitation est plus récent que ses concurrents (Android et IOS), ceci explique par exemple que son store contient moins d'applications à l'heure actuelle.

4. BlackBerry OS⁴

BlackBerry OS est probablement le moins connu des systèmes d'exploitation présentés ici. On ne le trouve que sur les Smartphones de la marque BlackBerry.

Chaque version de BlackBerry s'appelle « OS » + le numéro de la version. En 2017, c'est la version OS 10.3.2 qui est généralement installée sur les nouveaux appareils.

¹<https://www.android.com>

²<https://www.apple.com/fr/ios/ios-15>

³<https://www.microsoft.com/fr-fr/download/windowsphone.aspx>

⁴<https://www.blackberry.com/us/en>

Chapitre 2 : Préliminaires

Le store de BlackBerry (BlackBerry App World) compte beaucoup moins d'applications que les leaders du marché (App Store et Google Play Store).

2.3.3. Découverte de la plateforme Android

1. Définition

Lancé en juin 2007, Android est un système d'exploitation créé par Google, et fondé sur le noyau Linux. Il équipe aujourd'hui une grande majorité de smartphones et tablettes dans le monde. Son principal concurrent est l'iOS d'Apple.

L'une des spécificités d'Android par rapport à ses concurrents est qu'il peut être modifié par les différents constructeurs mobiles. Android est représenté par un petit robot vert nommé Bugdroid.[6]

2. Présentation du système d'exploitation Android

Le système d'exploitation ou Operating System (OS) est basé sur une architecture à quatre niveaux, décrite par la figure 2.2

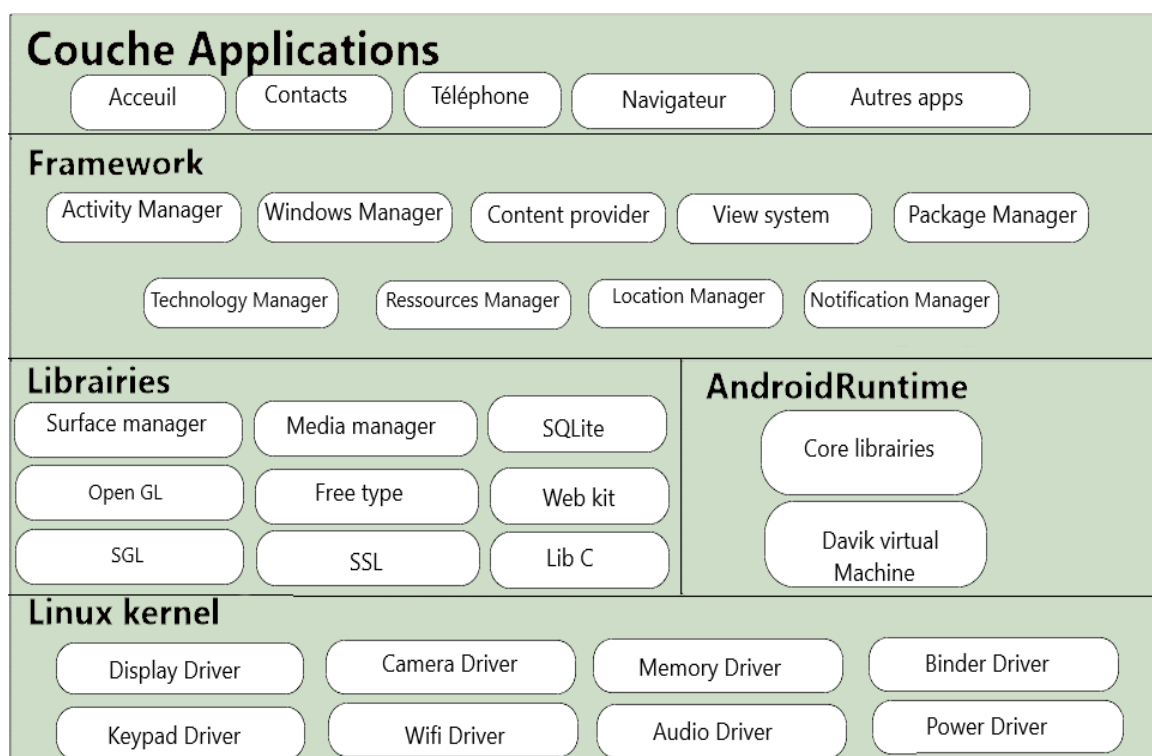


Figure 2.2 Architecture du logiciel Android.

• LE NOYAU

Comme vous pouvez le constater grâce au schéma ci-dessus le système d'exploitation Android est segmenté en plusieurs parties. Le noyau de celui-ci est basé sur un noyau Linux. Pour des raisons évidentes, que le noyau est open source et donc éditable par tout le monde. Ce qui a permis à Google de ne pas partir d'une feuille blanche.[7]

1. Le noyau permet avant tout la communication entre le système et les applications et le matériel. C'est pour cela qu'on retrouve dans cette couche les pilotes composants comme l'écran, la carte Wi-Fi, etc..[7]

Chapitre 2 : Préliminaires

2. Il sert également à contrôler les Wakelocks, qui sont des tâches de fond permettant la synchronisation de certaines applications. Tel que les e-mails, mais aussi de tâche comme votre alarme matinale.[7]

Le tableau 2.1 présenter une comparaison entre version noyau Android et version noyau Linux.

Version Android	Nom Android	Version noyauLinux
1.5	Cupcake	2.6.27
1.6	Donut	2.6.29
2.0/2.1	Eclair	2.6.29
2.2	Froyo	2.6.32
2.3	Gingerbread	2.6.35
3	Honeycomb	2.6.36
4.0	Ice Cream Sandwich	3.0.1
4.1	Jelly Bean	3.0.31
4.2	Jelly Bean	3.4.0
4.3	Jelly Bean	3.4.39
4.4	Kit Kat	3.10
5	Lollipop	3.16.1
6	Marshmallow	3.18.10
7.0/7.1	Nougat	4.4.1
8	Oreo	4.10

Tableau2.1 comparaison entre Version noyau Android et Version noyau linux.

Chapitre 2 : Préliminaires

• LES LIBRAIRIES ET ANDROID RUNTIME

• Librairies

Les librairies logicielles sont des morceaux de code compilés. Ce qui permet d'alléger le code des applications qui n'auront plus qu'à aller chercher les morceaux de code les intéressants. Par exemple imaginons une bibliothèque appareil photo qui comporte la fonction de zoomer avec les doigts. Chaque application utilisant cette fonctionnalité peut aller la chercher dans cette bibliothèque, au lieu de l'écrire à chaque fois. [7]

• Android Runtime

Android Runtime est arrivé sous Android 2.2, avec pour but de remplacer les machines virtuelles dalvik. Elles permettent l'exécution en simultanée de plusieurs applications sur des appareils peu performant. ART compile les applications en code sources avant son exécution, ce qui permet une meilleure optimisation. Mais cette migration est en pause car un procès car d'après Oracle Google enfreint certains de leurs brevets.[7]

• APPLICATION FRAMEWORK

Dans la couche application Framework se trouve les CorePlatform Services (services cœurs de la plateforme). Sur Android un service est une application sans interaction avec l'utilisateur et qui fonctionne en arrière-plan. Les Framework fournissent des API (Application Programming Interface qui signifie Interface de Programmation Applicative en français). Qui permettent aux développeurs des applications plus riches en fonctionnalité.[7]

• APPLICATIONS

Dans la dernière couche, se trouvent les applications aussi bien celles fourni avec votre appareil que celles disponibles sur les différents magasins.[7]

3. Gestion des événements du cycle de vie d'une activité

Le système, pour des raisons de priorisation d'activités (coup de téléphone), peut tuer une activité quand il a besoin de ressources. Pour cette raison, aucune activité ne peut penser pouvoir vivre jusqu'au bout de son traitement, elle doit être considérée comme un humain. Elle peut avoir un accident, être à l'hôpital et/ou mourir.[8]

Une activité possède quatre états que sont [8]:

- « **Active** » : l'activité est lancée par l'utilisateur, elle s'exécute au premier plan ;
- « **En Pause** » : l'activité est lancée par l'utilisateur, elle s'exécute et est visible, mais elle n'est plus au premier plan. Une notification ou une autre activité lui a volé le focus et une partie du premier plan ;
- « **Stoppée** » : l'activité a été lancée par l'utilisateur, mais n'est plus au premier plan et est invisible. L'activité ne peut interagir avec l'utilisateur qu'avec une notification ;
- « **Morte** » : l'activité n'est pas lancée.

Chapitre 2 : Préliminaires

La figure 2.3 indique le cycle de vie d'une activité et les méthodes appelées lors des changements d'état:

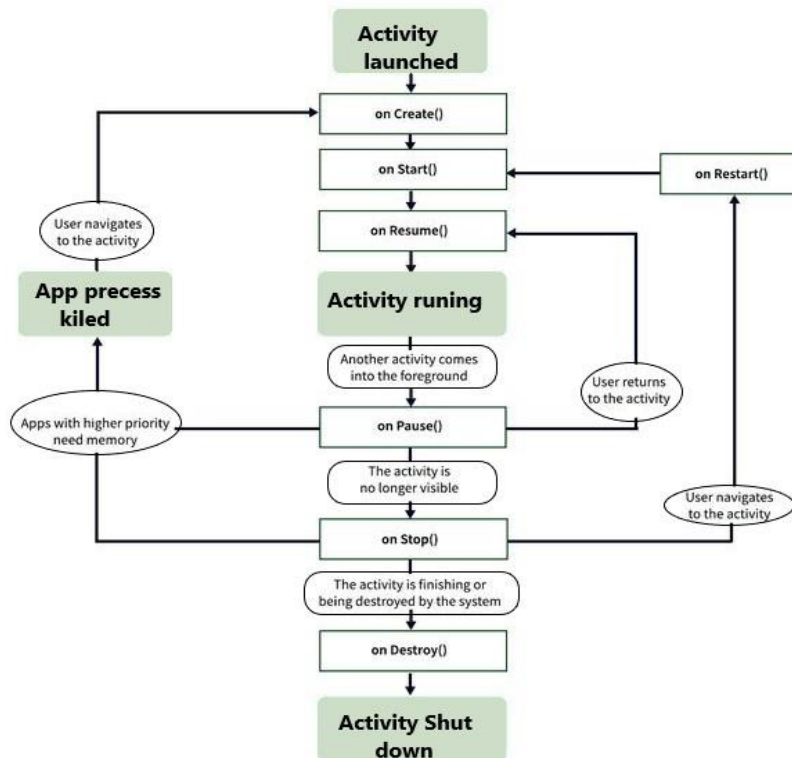


Figure 2.3 Cycle de vie d'une activité d'Android.

Ainsi, les différentes méthodes sont présentées ci-dessous.

La méthode **onCreate** est appelée :

- au premier lancement de l'activité ;
- si l'activité est ressuscitée, le bundle passé en paramètre sera celui sauvegardé par `onSaveInstanceState()` ;
- si l'état du terminal change et que l'activité est associée à cet état (passage du mode portrait au mode paysage).

Elle configure les IHM et tous les traitements d'initialisation qui ne sont effectués qu'une seule fois au lancement de l'activité.

La méthode **onDestroy** est appelée lors de la mort de l'activité (soit naturelle, soit par le système). Parfois, l'urgence du système détruira l'activité sans même appeler **onDestroy**. Cette méthode doit libérer les ressources allouées dans la méthode **onCreate**. La seule méthode qui est appelée avec certitude avant la destruction de l'activité est **onPause()**.

Les méthodes **onStart**, **onRestart** et **onStop** n'ont pas grand intérêt.

Chapitre 2 : Préliminaires

Les méthodes **onPause** et **onResume** sont celles dans lesquelles l'activité doit sauvegarder ses états et les restituer.

La méthode **onResume** est appelée immédiatement avant que l'activité ne passe au premier plan. De ce fait, c'est le bon endroit pour reconstruire les données affichées par l'IHM et mettre celle-ci à jour, quitte à lancer un thread de reconstruction des données.

Inversement, la méthode **onPause** est la seule par laquelle il est certain que l'application passera avant de se faire détruire. Il convient donc de sauvegarder l'état de l'application dans cette méthode.

La sauvegarde et la restauration s'effectuent au travers des méthodes **onSaveInstance** et **onRestoreInstance**. L'objet Bundle est passé en paramètre de l'une et renvoyé par l'autre. L'objet Bundle est une carte (map) intelligente qui stocke des paires typées (Key, Object) et possède l'ensemble des méthodes pour enregistrer et récupérer ces paires (**putBoolean(String key, boolean value)** et **boolean getBoolean(String Key)** par exemple et ce pour la plupart des types connus).

Toute la subtilité du mécanisme réside dans le choix fait par le développeur des données à sauvegarder. Il ne doit pas vouloir y mettre la terre entière et être précis dans ce qu'il choisit de stocker et de restaurer.

2.4. Conclusion

Dans ce chapitre nous avons présenté une généralité des applications mobile et leurs systèmes d'exploitation. Nous avons aussi détaillé le système d'exploitation Android ainsi que le cycle de vie d'activité du système Android.

Dans le chapitre suivant nous discutons une analyse des travaux similaires à notre application sur le marché local mais aussi sur l'international.

Chapitre 3 : Etat de l'art

3.1.Introduction

Le chapitre présent a pour objectif de présenter les travaux similaires qui tentent à répondre aux besoins des demandeurs de service de bricolage. Nous analysons au début les travaux qui touchent le marché algérien dans la Section 3.3, tandis que la Section 3.6 étudie les travaux visant le marché international. La Section 3.8 est une comparaison entre les travaux locaux et internationaux. En fin, la Section 3.9 conclue le chapitre.

3.2.Etude de l'existant

L'objectif de l'étude est pour comprendre le projet actuel qu'il existe dans le marché, mais sont peu nombreuses et leur développement tardif en Algérie par rapport à d'autres pays.

L'humains est besoin ces applications dans son quotidien, exemple :

En cas d'urgence comme des fuites de robinet, une panne de voiture entre autres, il devient difficile pour quelqu'un d'accéder immédiatement à ces services. Dans les cas où l'urgence est grave, comme une panne électrique d'un équipement électronique, cela peut entraîner des calamités comme un incendie qui ont de profondes conséquences.

Alors nous avons faire cette étude pour traiter les lacunes qui existe dans ces applications et essayerons de faire une bonne solution dans notre application.

3.3.Applications Algériennes

3.3.1. Tashil

Tashil⁵est l'une des applications algériennes qui proposent des services d'offre et de demande en Algérie, et elle met à disposition de nombreux professionnels, artisans et divers modèles de services qui se sont inscrits dans l'application pour afficher leurs services, et de ce point de vue, il est facile pour le utilisateur à la recherche d'un service particulier, de rechercher son propriétaire en entrant simplement dans l'application et une vue différente des visualiseurs de services sur la carte, et dès que vous cliquez sur l'un d'eux, ses informations complètes apparaissent, notamment son nom, son adresse, numéro de téléphone et le type de service qu'il fournit.

3.3.2. Fixit

Fixit⁶est l'une des applications Algérienne de mise en relation entre particuliers et bricoleurs et/ou artisans dans différents domaines tels que la plomberie, l'électricité, réparation électroménagers et parabole, etc.

Outre la possibilité de commander un bricoleur dans l'immédiat, l'application permet de le réserver également à l'avance pour une date ultérieure à la convenance de l'utilisateur.

⁵https://play.google.com/store/apps/details?id=com.cododo.tashil&hl=en_US&gl=US

⁶https://play.google.com/store/apps/details?id=com.fixit.customer&hl=en_US&gl=US

Chapitre 3 : Etat de l'art

3.3.3. Youchoz

C'est une application de mise en relation entre les particuliers et les artisans de différents services (plomberie, électricité, maçonnerie, peinture...). Composée de deux applications :

Youchoz⁷: pour aide les clients à trouver un artisan.

Youchoz Artisan⁸: pour aide les artisans à trouver de nouveaux projets.

3.4. Analyse de l'existant (Marché local)

Ces applications dont le but principal est de connecter l'utilisateur et le travailleur, mais ce sont des applications qui ne sont pas très populaires dans la société algérienne à cause du manque et de la difficulté de les utiliser et elles contiennent des problèmes au niveau du temps de service et de la localisation.

3.5. Comparaison entre les applications Algériennes (Tashil et Fixit et Youchoz)

Le tableau 3.1 présenter une comparaison entre les applications Algériennes.

	TASHIL	FIXIT	Youchoz	Youchoz Artisan
Taille	12MO	8.9MO	32MO	31MO
Nombre de téléchargements	Plus 1000	Plus 5000	Plus 1000	Plus 1000
Disponible pour les Plates-formes	Android	Android, IOS	Android, IOS	Android, IOS
Les catégories de service disponibles	plomberie, électricité, services médicales, blanchisserie, jardinage, service à domicile, service automobile,	plomberie, chauffage, froid et climatisation, réparation électroménagers, petit travaux de bricolage, électricité, installation parabole et Jardinage	plomberie, électricité, maçonnerie, peinture, chauffage, climatisation, carrelage, électricité, réseau, ferronnerie, Jardinage Cuisine, électroménagers, nettoyage, menuiserie...	

Tableau 3.1 Comparaison entre les applications algériennes (Tashil et Fixit et Youchoz).

⁷<https://play.google.com/store/apps/details?id=com.youchoz.particulier>

⁸<https://play.google.com/store/apps/details?id=com.youchoz.professionnel>

Chapitre 3 : Etat de l'art

3.6. Applications internationales

3.6.1. Thumbtack (application de service) :

Thumbtack⁹ est une société privée de services de bricolage à la demande basée en Californie. Ils offrent près de 1000 services différents parmi lesquels choisir. Thumbtack fournit des services tels que tuteur à domicile, professionnel du maquillage, dresseur de chiens, fleuriste, nettoyeur et bien d'autres. Vous pouvez consulter les avis et les notes avant de choisir le professionnel.

3.6.2. Taskrabbit

Taskrabbit¹⁰ est l'application américaine de bricoleur à la demande qui permet aux clients de réserver des services d'entretien ménager tels que le nettoyage, le déménagement, l'assemblage de meubles, etc., ainsi que de commander des produits d'épicerie. Actuellement, ils fournissent des services dans 47 villes des États-Unis, 4 villes du Royaume-Uni et une ville du Canada.

3.7. Analyse de l'existant (Marché international)

Arriver jusqu'ici, nous pouvons constater que les applications nationales et internationales sont similaires en terme d'objectives. Par contre, la comparaison en terme de performances donne aux application étrangères une avancée en matière d'efficacité, vitesse, facilité d'utilisation...

3.8. Critiques d'applications locales/internationales

Tout d'abord, nous constatons que les applications algériennes représentent des plateformes bien structurées. Cette structuration a permis de produire des applications de hautes performances. Par contre, les applications de l'étranger sont bien mieux en termes de :
L'incitation du consommateur à les utiliser : Grâce à des publicités alléchantes.
Moyens de paiement faciles : Utilisation de cartes bancaires internationales comme (Visacard, Mastercard...)
Des programmes de fidélisation de la clientèle;
La fourniture d'autres services complémentaires : fournit des services importantes mais sont pas exister dans les autres.

3.9. Conclusion

Dans ce chapitre, nous avons analysé les travaux similaires à notre proposition selon deux classes: les travaux locaux et internationales, et Nous avons aussi amené une comparaison entre les travaux algériennes selon un ensemble des caractéristiques. Enfin nous avons signalé les clés de succès des travaux internationaux par rapport à travaux locaux.

⁹ <https://play.google.com/store/apps/details?id=com.thumbtack.pro&hl=fr&gl=US>

¹⁰ <https://play.google.com/store/apps/details?id=com.taskrabbit.droid.consumer&hl=fr&gl=US>

Chapitre 4 : Conception

4.1.Introduction

Le présent de chapitre nous permet d'identifier toutes les fonctionnalités de notre future application, Ceci se fera par l'identification des acteurs et la définition de tous les besoins qui seront par la suite modélisés par un diagramme de cas d'utilisation(Section 4.3) pour chaque entité, suivi de leurs diagramme de classes(Section 4.4)et enfin les diagrammes de séquence(Section 4.5).

4.2.Modélisation des besoins avec UML

4.2.1. Définition de UML

UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue.[9]

4.2.2. Les diagrammes de UML

UML dans sa version 2 propose treize diagrammes qui peuvent être utilisés dans la description d'un système. Ces diagrammes sont regroupés dans deux grands ensembles [10]:

A. Les diagrammes structurels

Ces diagrammes, au nombre de six, ont vocation à représenter l'aspect statique d'un système (classes, objets, composants...).

- **Diagramme de classe** : Ce diagramme représente la description statique du système en intégrant dans chaque classe la partie dédiée aux données et celle consacrée aux traitements. C'est le diagramme pivot de l'ensemble de la modélisation d'un système.
- **Diagramme d'objet** : Le diagramme d'objet permet la représentation d'instances des classes et des liens entre instances.
- **Diagramme de composant** : Ce diagramme représente les différents constituants du logiciel au niveau de l'implémentation d'un système.
- **Diagramme de déploiement** : Ce diagramme décrit l'architecture technique d'un système avec une vue centrée sur la répartition des composants dans la configuration d'exploitation.
- **Diagramme de paquetage** : Ce diagramme donne une vue d'ensemble du système structuré en paquetage. Chaque paquetage représente un ensemble homogène d'éléments du système (classes, composants...).
- **Diagramme de structure composite** : Ce diagramme permet de décrire la structure interne d'un ensemble complexe composé par exemple de classes ou d'objets et de composants techniques. Ce diagramme met aussi l'accent sur les liens entre les sous-ensembles qui collaborent.

Chapitre 4 : Conception

B. Les diagrammes de comportement

Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs. Sept diagrammes sont proposés par UML :

- **Diagramme des cas d'utilisation** :Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un des diagrammes les plus structurants dans l'analyse d'un système.
- **Diagramme d'état-transition (machine d'état)** :Ce diagramme montre les différents états des objets en réaction aux événements.
- **Diagramme d'activités** : Ce diagramme donne une vision des enchaînements des activités propres à une opération ou à un cas d'utilisation. Il permet aussi de représenter les flots de contrôle et les flots de données.
- **Diagramme de séquence** :Ce diagramme permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets.
- **Diagramme de communication (anciennement appelé collaboration)** : Ce diagramme est une autre représentation des scénarios des cas d'utilisation qui met plus l'accent sur les objets et les messages échangés.
- **Diagramme global d'interaction** : Ce diagramme fournit une vue générale des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d'activités.
- **Diagramme de temps** : Ce diagramme permet de représenter les états et les interactions d'objets dans un contexte où le temps a une forte influence sur le comportement du système à gérer.

➤ Nous avons utilisé les diagrammes suivants pour modéliser notre système :

- Diagramme de cas d'utilisation illustré dans la Figure 4.1.
- Diagramme de classe illustré dans la Figure 4.2.
- Diagramme de séquence illustré dans la Figure 4.3, la Figure 4.4 ,la Figure 4.5,la Figure 4.6,la Figure 4.7,la Figure 4.8 et la Figure 4.9.

Chapitre 4 : Conception

4.2.3. Diagramme des cas d'utilisation

- **Identification des auteurs :**

Client : désigne la personne qui utilise notre application afin de pouvoir voir les bricoleurs disponibles dans application.

Bricoleur : désigne la personne responsable de fournir service de bricoleur.

Administrateur : désigne la personne responsable de la maintenance de l'application et de la gestion des comptes utilisateurs. Elle assure le bon fonctionnement du serveur de données et de sa sécurité.

- **Identification des cas d'utilisation:**

Client: crée compte ,Recherche d'un service, Évaluer un service , Prendre un rendez-vous.

Bricoleur : crée profil et compte, s'inscrire dans une catégorie , confirmer un rendez-vous.

Administrateur : Crée compte administrateur, confirmer sur service bricoleur, statistiques des clients.

Chapitre 4 : Conception

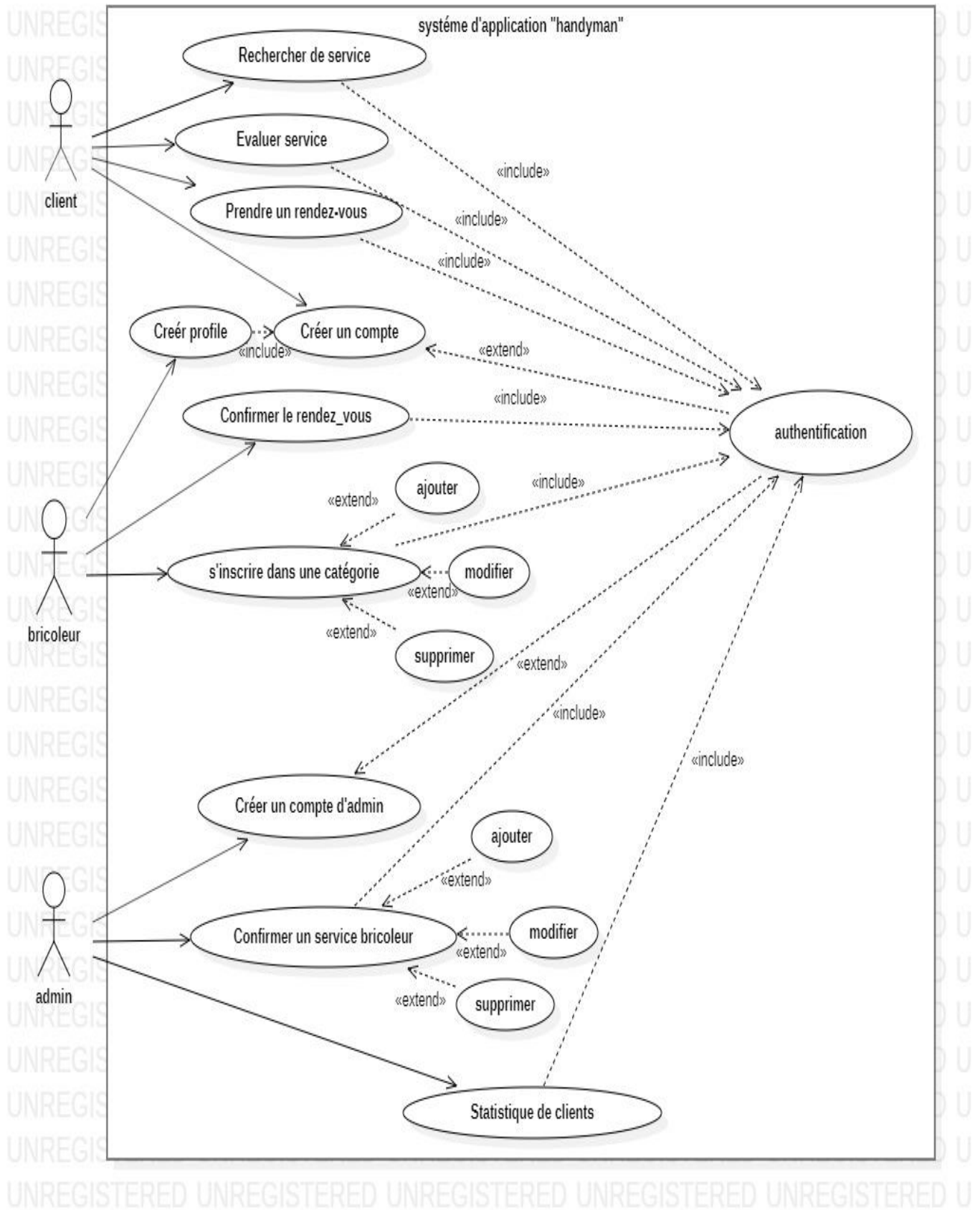


Figure 4.1 Diagramme de cas d'utilisation.

Chapitre 4 : Conception

4.2.4. Diagramme de classe de l'application « HANDYMAN »

Le diagramme de classe de notre application est représenté par la figure 4.2

- **Modèle logique des données :**

- Bricoleur**(id_bricoleur,id_catégorie)

- Client** (id_client)

- Administrateur**(nom_admin, mot_de_passe)

- Compte**(id_compte)

- Catégorie**(id_catégorie,nom_catégorie,liste_bricoleur)

- RDV**(id_rdv,jour,heur)

- Localisation**(id_local,latitude,longtude)

- Confirmer**(#administrateur,#bricoleur)

- Gérer**(#administrateur,#compte)

- RDV**(#bricoleur,#client,jour,heur)

- Localiser** (#compte,#localisation)

Chapitre 4 : Conception

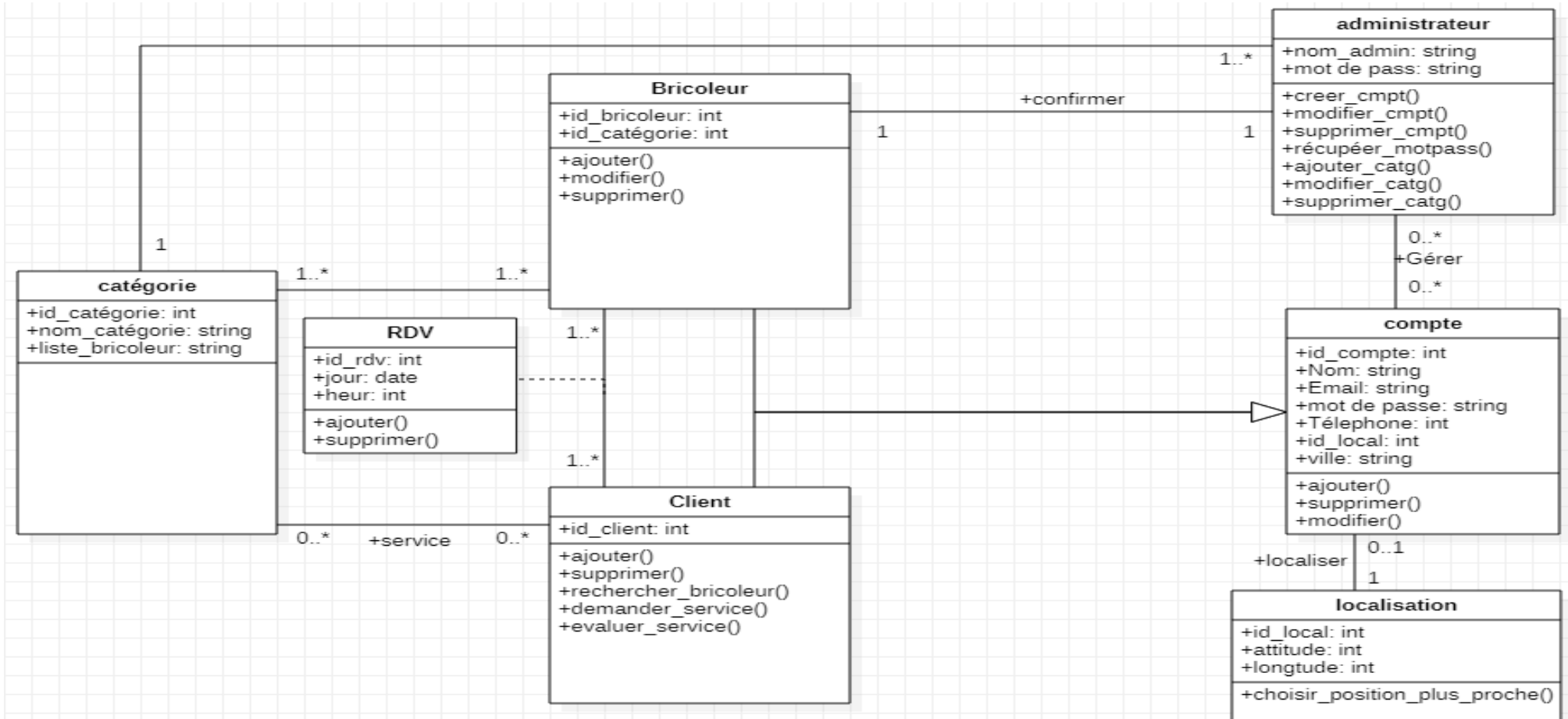


Figure 4.2 Diagramme de classe d'application.

Chapitre 4 : Conception

4.2.5. Diagrammes de séquences

- **Diagrammes de séquence des cas d'utilisation "Authentification"**

Certaines actions sur l'application nécessitent l'authentification de l'utilisateur. La Figure 4.3 présente séquence d'interactions entre l'utilisateur et le système pour authentification. L'utilisateur demande l'affichage du formulaire authentification. Ensuite, notre système l'affichera de son côté et après l'utilisateur saisit le mot de passe et c'est au système de vérifier sa validité qui affichera par la suite la page d'accueil, sinon il y a erreur du mot de passe.

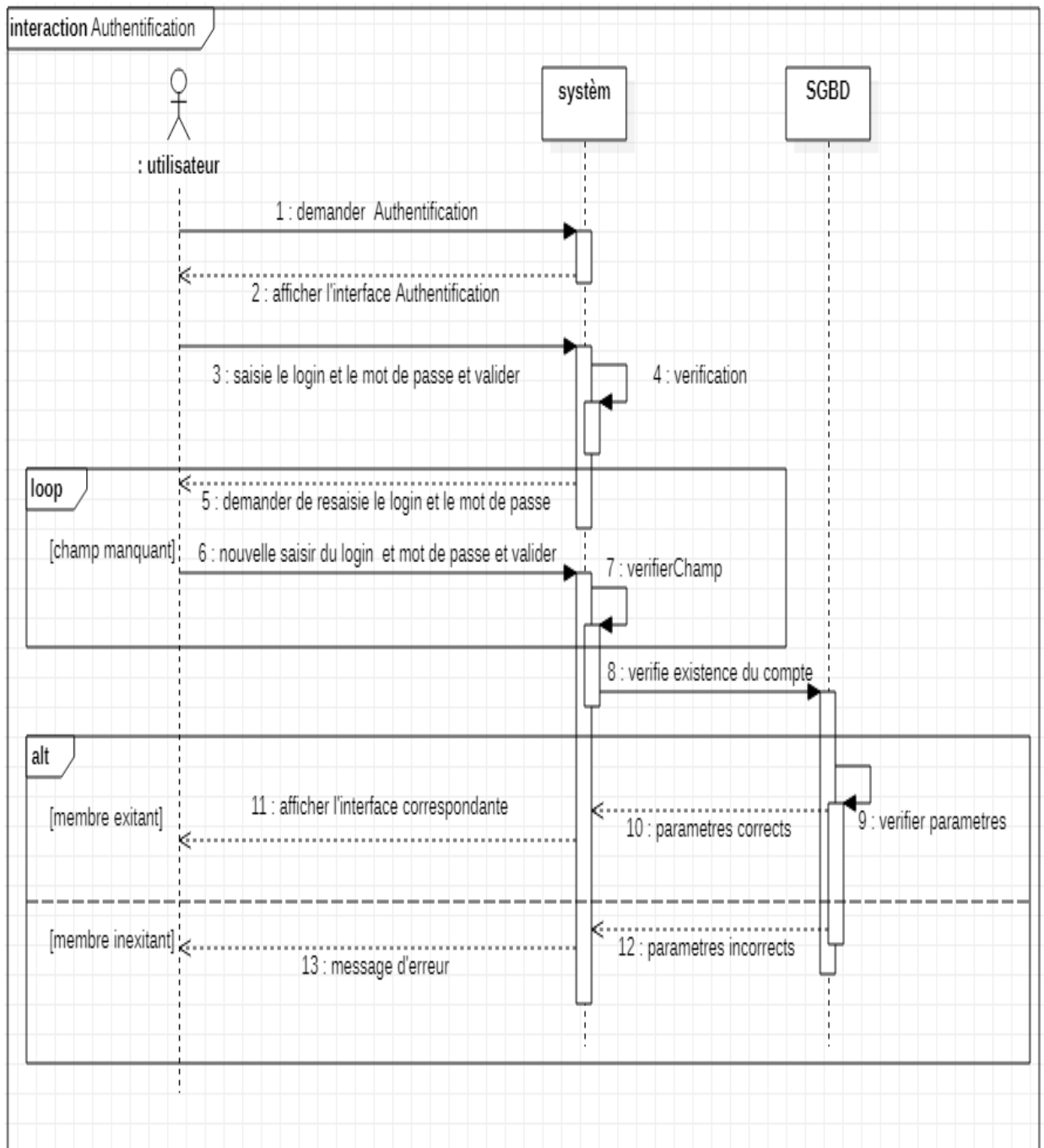


Figure 4.3 Diagramme de séquence "Authentification".

Chapitre 4 : Conception

- Diagrammes de séquence des cas d'utilisation "gérer compte"

L'inscription est une étape indispensable. En effet, chaque utilisateur doit avoir son propre compte pour utiliser l'application. Si l'utilisateur est un bricoleur alors il doit remplir un formulaire d'inscription contenant les champs : nom, prénom, date de naissance, etc. Il peut être modifié et supprimé leur compte.

Le scénario pour gérer compte pour utilisateur se déroule selon les diagrammes présentés à la Figure 4.4 pour ajouter compte et la Figure 4.5 pour modifier compte, la Figure 4.6 pour supprimer compte,

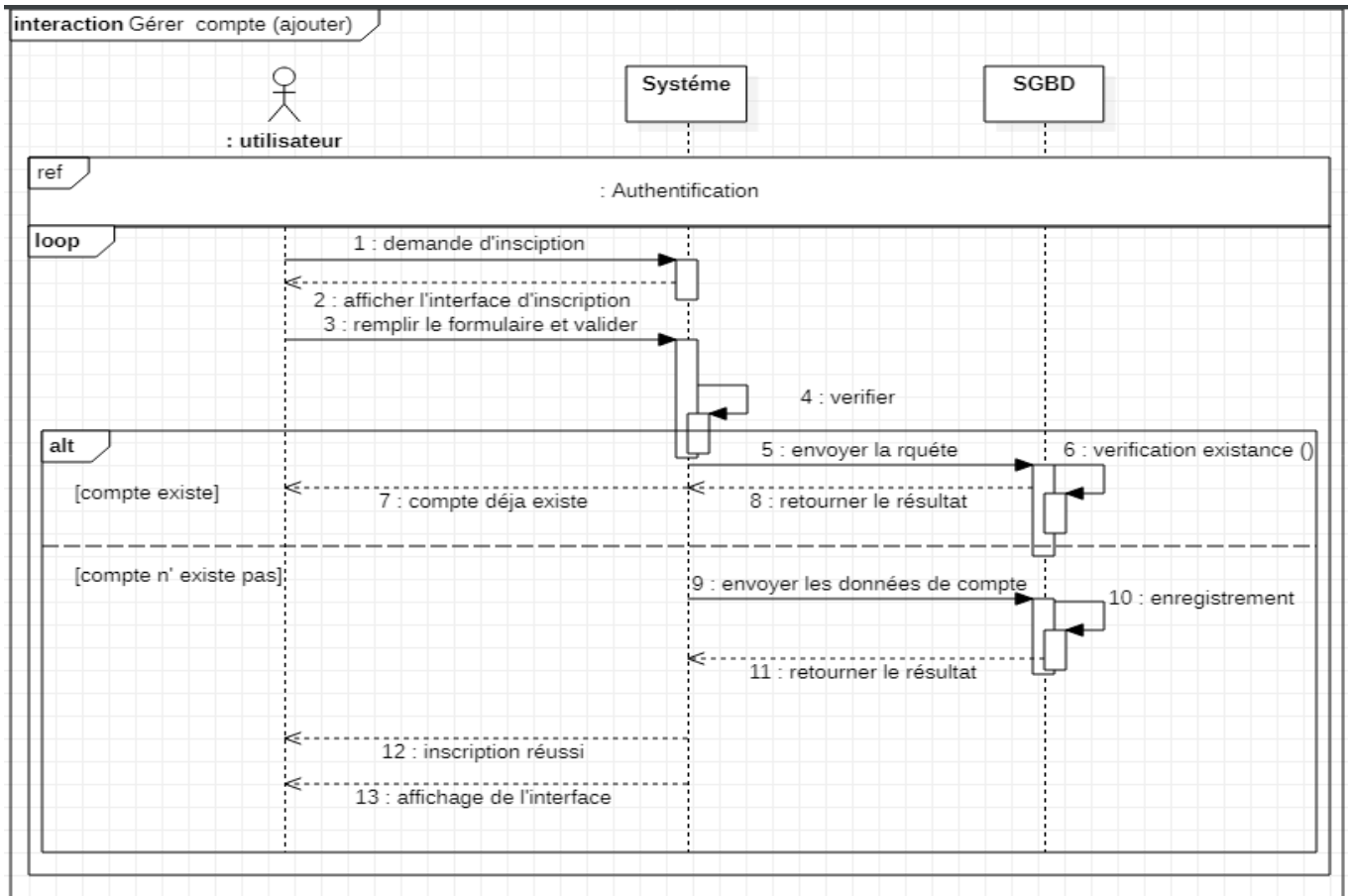


Figure 4.4 Diagramme de séquence "Gérer compte (ajouter)".

Chapitre 4 : Conception

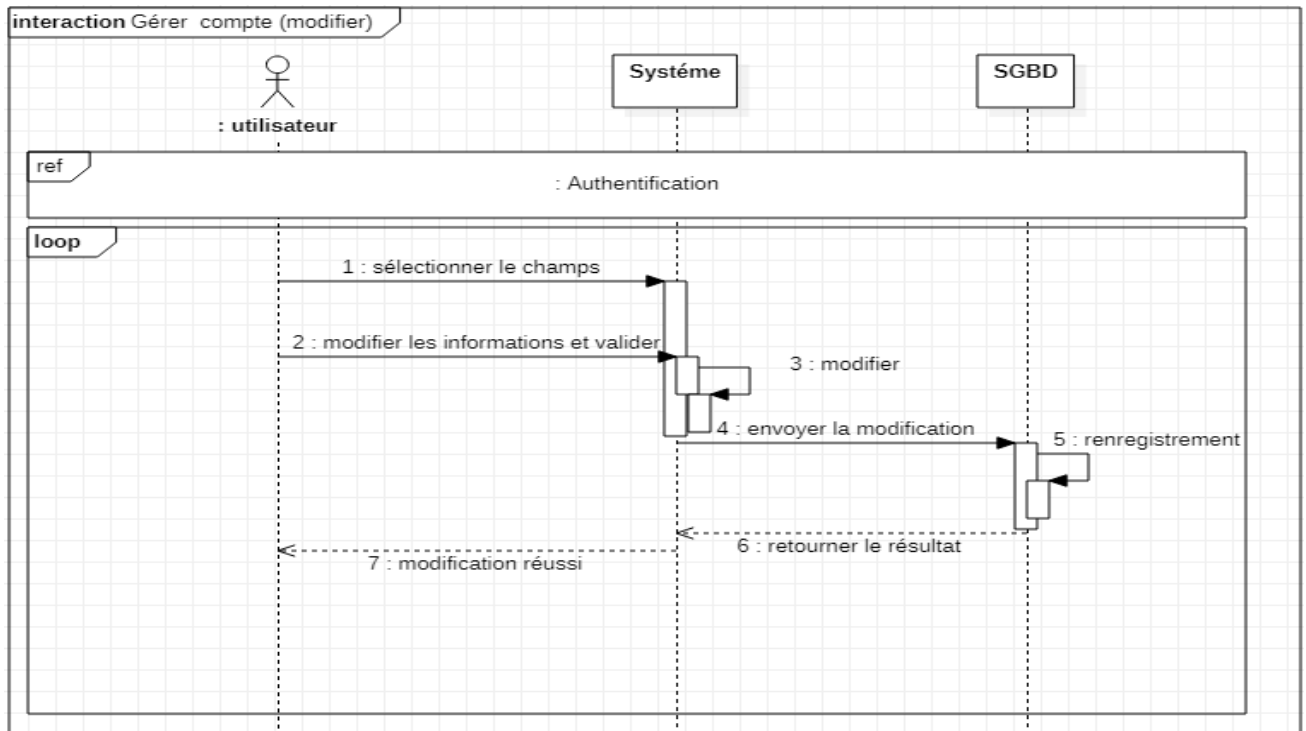


Figure 4.5 Diagramme de séquence "Gérer compte (modifier)".

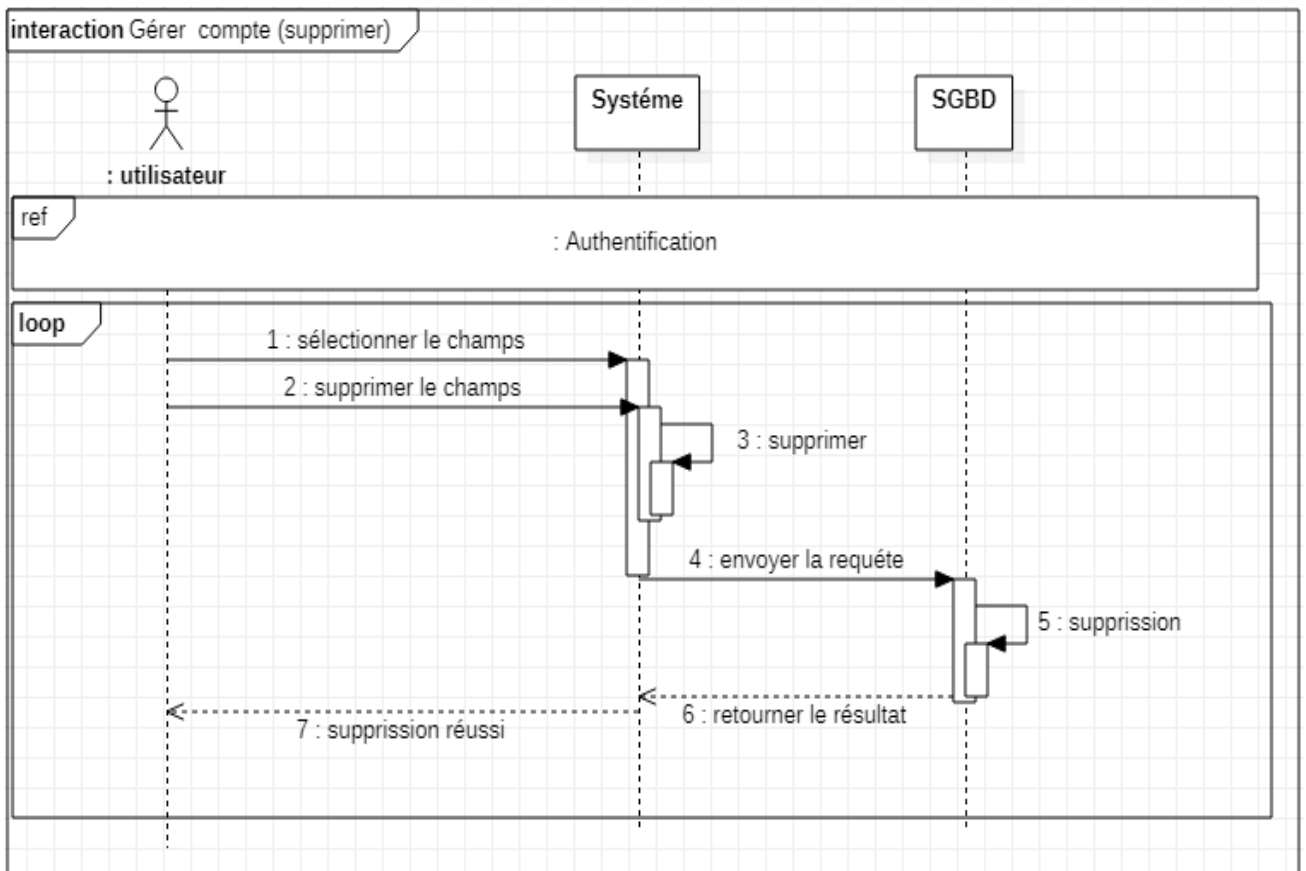


Figure 4.6 Diagramme de séquence "Gérer compte (supprimer)".

Chapitre 4 : Conception

- Diagrammes de séquence des cas d'utilisation "prendre rendez-vous"

Client demande au bricoleur de prendre rendez-vous, si le bricoleur est disponible il accepte et confirme le rendez-vous, sinon il refuse le rendez-vous.

Le scénario pour prendre rendez-vous entre le client et le bricoleur se déroule selon le diagramme présenté à la Figure 4.7

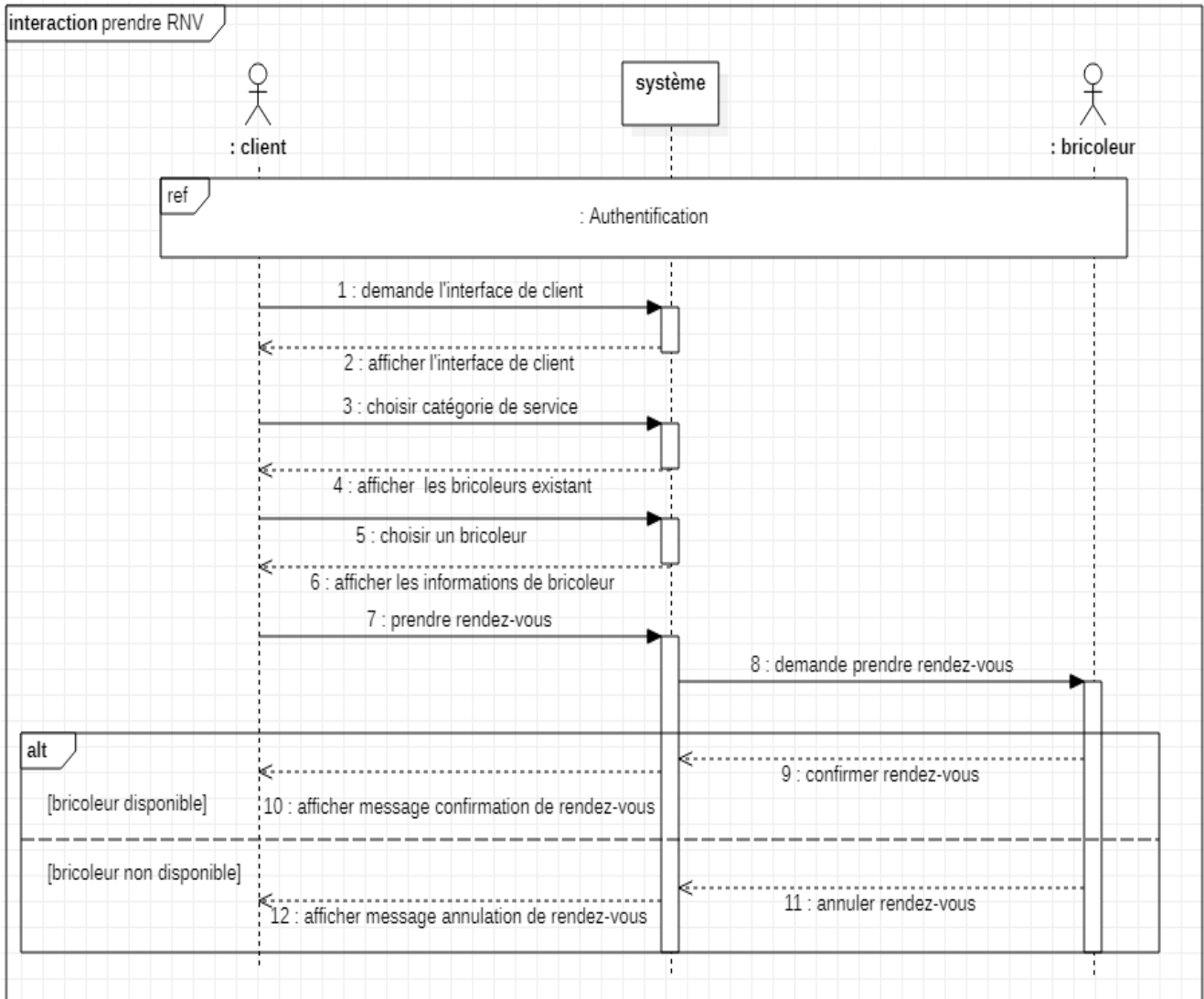


Figure 4.7 Diagramme de séquence "prendre rendez-vous".

Chapitre 4 : Conception

- **Diagrammes de séquence des cas d'utilisation "confirmer bricoleur"**

le bricoleur ne peut pas accéder à son interface avant que l'administrateur va confirmé son diplôme si il est valable ou non, se déroule selon le diagramme présentée à la Figure 4.8.

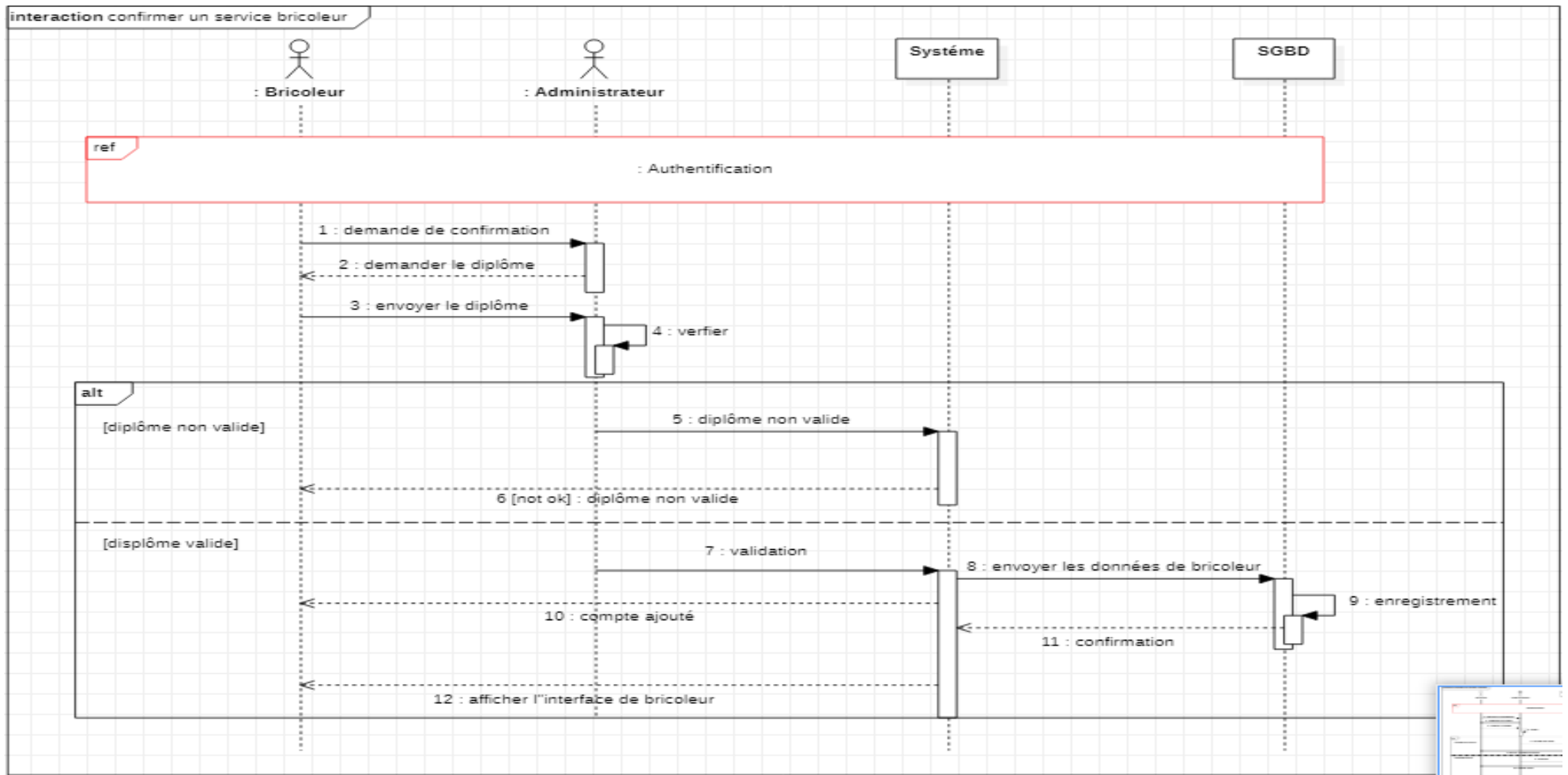


Figure 4.8 Diagramme de séquence "confirmer bricoleur".

Chapitre 4 : Conception

- **Diagrammes de séquence des cas d'utilisation "la recherche d'un bricoleur"**

Le scénario pour la recherche d'un bricoleur a partir client se déroule selon la chronologie présentée à la Figure 4.9.

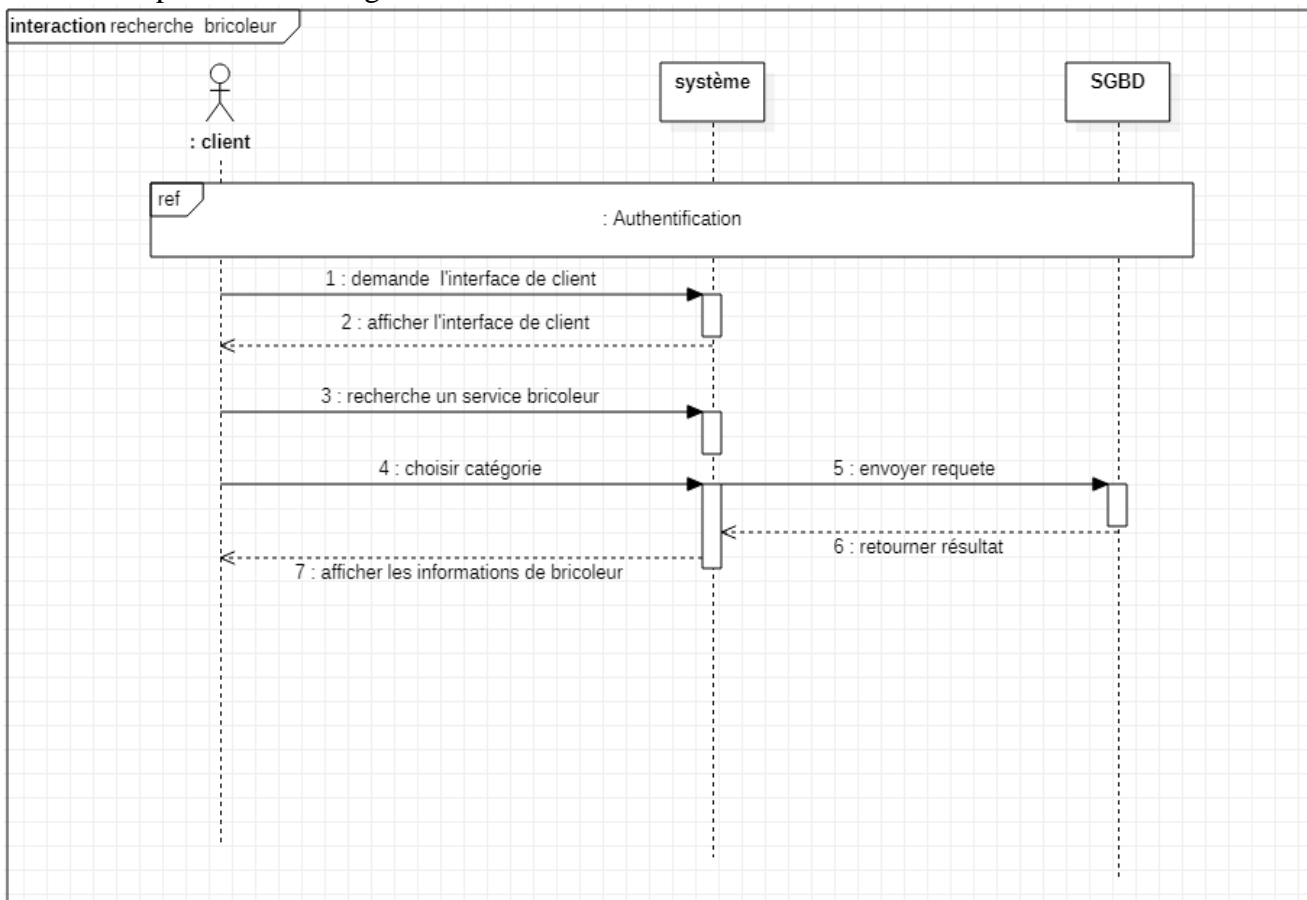


Figure 4.9 Diagramme de séquence "recherche d'un bricoleur".

4.3. Conclusion

Dans ce chapitre, nous avons décrit en détail les cas d'utilisation en recensant de manière détaillée toutes les interactions entre les acteurs et le système, Nous avons présenté les différents diagrammes UML utilisés à savoir : le diagramme de classes, le diagramme de cas d'utilisation et les diagrammes de séquences représentés avec un logiciel nommé Star UML ¹¹;
La concrétisation de notre application est le sujet du chapitre suivant.

¹¹ <https://staruml.io/>

Chapitre 5 : Implémentation

5.1.Introduction

Dans ce chapitre, nous présentons les différents outils et les langages utilisés pour l'implémentation de notre travail(Section 5.2), accompagnés de quelques interfaces illustrant les fonctionnalités de l'application développée (Section 5.3).

5.2.Environnements de travail

5.2.1.Les outils utilisés

1. Android studio

Android Studio est l'environnement de développement intégré (IDE) officiel pour le développement d'applications Android. Pour prendre en charge le développement d'applications au sein du système d'exploitation Android, Android Studio utilise un système de génération basé sur Gradle, un émulateur, des modèles de code et une intégration Github. le langage de programmation du studio Android est : Java et Kotlin. [11]

2. XAMPP

XAMPP est un ensemble de logiciels libres. Le nom est un acronyme venant des initiales de tous les composants de cette suite. Ce dernier réunit donc le serveur Web Apache, la base de données relationnelle et système d'exploitation MySQL ou MariaDB ainsi que les langages scripts Perl et PHP. L'initiale X représente tous les systèmes d'exploitation possibles, à savoir Linux, Windows et Mac OS X. [12]

3. Le serveur Apache

Apache est un logiciel de serveur web gratuit et open-source qui alimente environ 46% des sites web à travers le monde. Le nom officiel est Serveur Apache HTTP et il est maintenu et développé par Apache Software Foundation. C'est l'un des serveurs web les plus anciens et les plus fiables avec une première version sortie il y a plus de 20 ans, en 1995.Apache est un logiciel multiplateforme, il fonctionne donc à la fois sur les serveurs Unix et Windows. [13]

4. PhpMyAdmin

PhpMyAdmin est un outil logiciel libre écrit en PHP, destiné à gérer l'administration de MySQL sur le Web. PhpMyAdmin supporte un large éventail d'opérations sur MySQL et Maria DB. Les opérations fréquemment utilisées (gestion des bases de données, des tables, des colonnes, des relations, des index, des utilisateurs, des autorisations, etc.) peuvent être effectuées via l'interface utilisateur, tout en ayant la possibilité d'exécuter directement n'importe quelle instruction SQL. [14]

Chapitre 5 : Implémentation

5.2.2. Les langages utilisés

1. JAVA

Java est un langage objet permettant le développement d'applications complètes s'appuyant sur les structures de données classiques (tableaux, fichiers) et utilisant abondamment l'allocation dynamique de mémoire pour créer des objets en mémoire.

Le langage Java permet également la définition d'interfaces graphiques (GUI : Graphical User Interface) facilitant le développement d'applications interactives et permettant à l'utilisateur de "piloter" son programme dans un ordre non imposé par le logiciel.

Un programme Java est portable au sens où il peut s'exécuter sur des ordinateurs fonctionnant avec différents systèmes d'exploitation. [15]

2. XML(eXtensible Markup Language)

XML (pour eXtensible Markup Language) est un langage (ou « méta-langage », langage pour définir d'autres langages) qui permet de structurer de l'information. Il utilise, comme HTML (et la comparaison s'arrête là), des balises pour « marquer » les différentes parties d'un contenu, ce qui permet aux applications qui utilisent celui-ci de s'y retrouver. Il s'agit d'un texte simple et non structuré dont aucune application automatisée ne peut extraire avec pertinence les informations. [16]

3. PHP(Hypertext Preprocessor)

PHP est un langage de programmation, très proche syntaxiquement du langage C, destiné à être intégré dans des pages HTML. Contrairement à d'autres langages, PHP est principalement dédié à la production de pages HTML générées dynamiquement. Le langage PHP a été créé par RasmusLerdorf en 1994. [16]

4. JSON (JavaScript Object Notation)

JavaScript Object Notation, ou simplement JSON, est apparu au début des années 2010. Au départ, JSON est un format d'échange de données entre le front-end web (typiquement un navigateur) et le back-end. Ce format était pratique, standard et léger. JSON permet de structurer les données, les valeurs que l'on souhaite échanger entre les différentes applications ou couches applicatives. [17]

5. SQL

SQL (Structured Query Language) est un langage de « programmation » standardisé qui est utilisé pour gérer des bases de données relationnelles et effectuer diverses opérations sur les données qu'elles contiennent. Créé à l'origine dans les années 1970, SQL est régulièrement utilisé non seulement par les administrateurs de bases de données, mais aussi par les développeurs qui écrivent des scripts d'intégration de données et les analystes de données qui cherchent à mettre en place et à exécuter des requêtes analytiques.[18]

Chapitre 5 : Implémentation

5.3. Description de l'application

Nous allons présenter les principales interfaces graphiques de notre application « HEIMWERKER ZUHAUSE ». Elle comprend trois espaces :

- Espace de bricoleur.
- Espace de client.
- Espace d'administrateur.

5.3.1. Logo de l'application

Cette figure représente logo de notre application « HEIMWERKER ZUHAUSE ».



Figure 5.1 Logo de l'application.

Chapitre 5 : Implémentation

5.3.2. Interface de Politique de confidentialité

Cette interface apparaît au client et au bricoleur et est un document qui stipule comment collecte, traite les données .

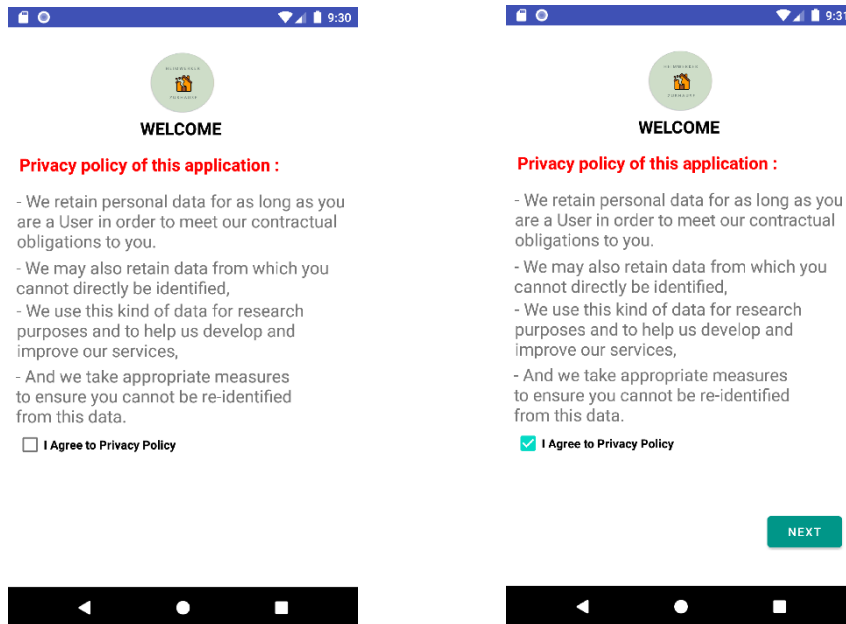


Figure 5.2 Interface de Politique de confidentialité de l'application.

5.3.3. Espace d'administrateur

➤ Interface l'inscription et d'Authentification

La Figure5.3 représente l'interface de l'inscription de l'administrateur, et la Figure 5.4 représente l'interface d'Authentification (saisir leurs mot de passe et leurs email afin d'accéder à leurs interface).

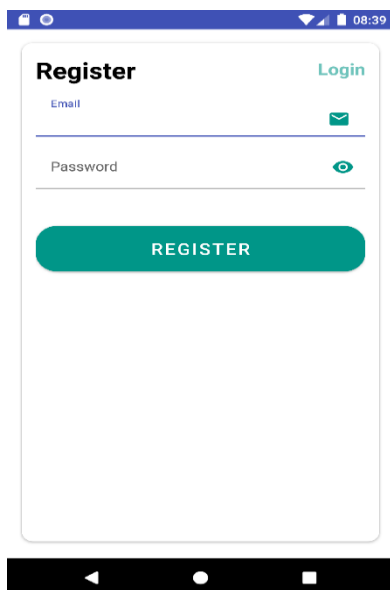


Figure5.3 L'interface d'inscription d'administrateur

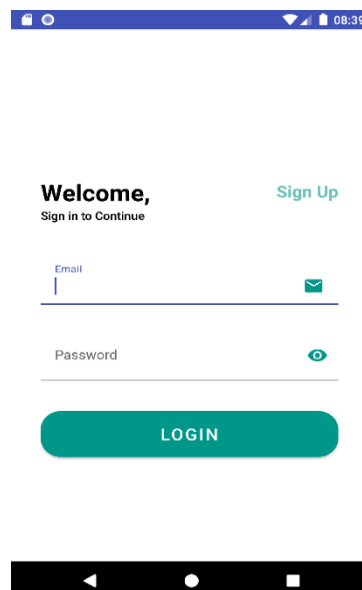


Figure5.4 L'interface d'Authentification d'administrateur

Chapitre 5 : Implémentation

➤ Interface Accueil

C'est la première interface affichée lors du lancement de l'application de l'administrateur. (Figure 5.5).

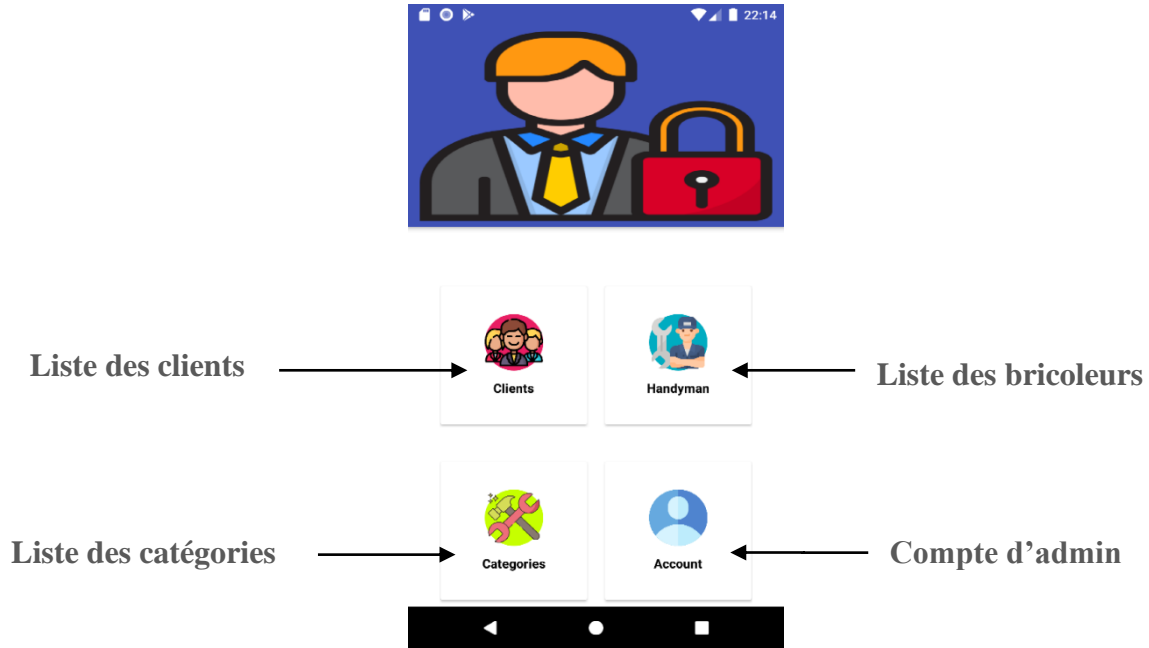


Figure 5.5 L'interface d'accueil d'administrateur.

➤ Interface de liste des clients

La figure 5.6 représente la liste des clients qui utilisent notre application, et l'admin peut voir les informations de certains clients dans la liste dans la figure 5.7 et la figure 5.8.

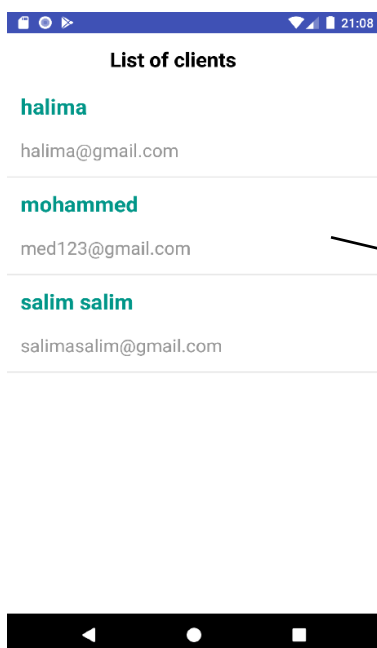


Figure 5.6 Interface de la liste des clients.

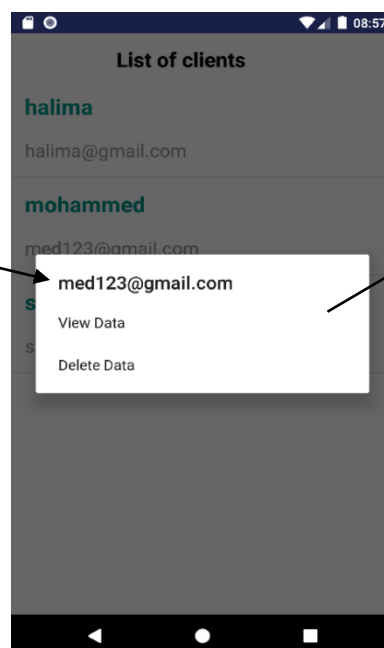


Figure 5.7 Liste cliquable sur un client.

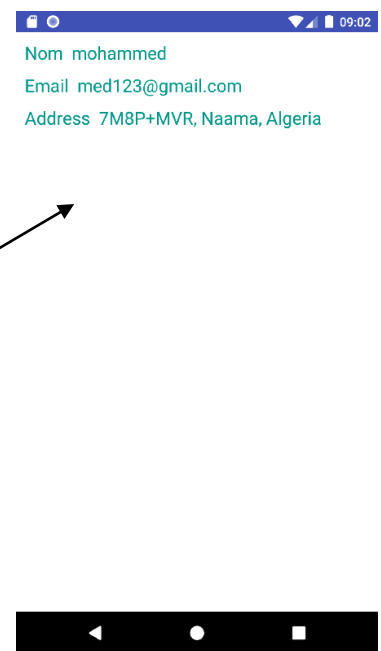


Figure 5.8 Fiche d'informations d'un client.

Chapitre 5 : Implémentation

➤ Interface de liste des catégories

La figure 5.9 représente la liste des catégories de bricoleur qui existent dans l'application et d'autre catégorie peuvent être ajoutées par l'administrateur(Figure 5.10).

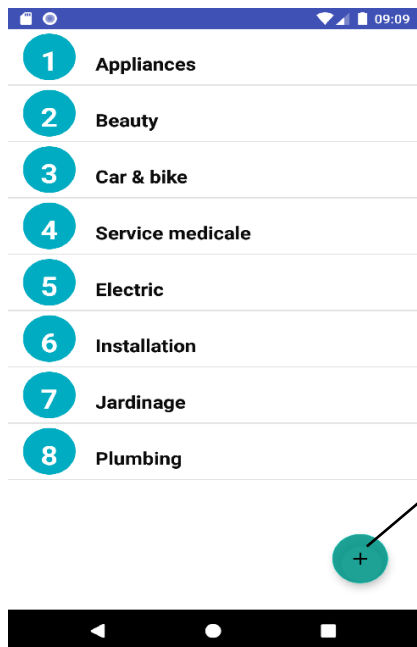


Figure 5.9 Interface de liste des catégories.

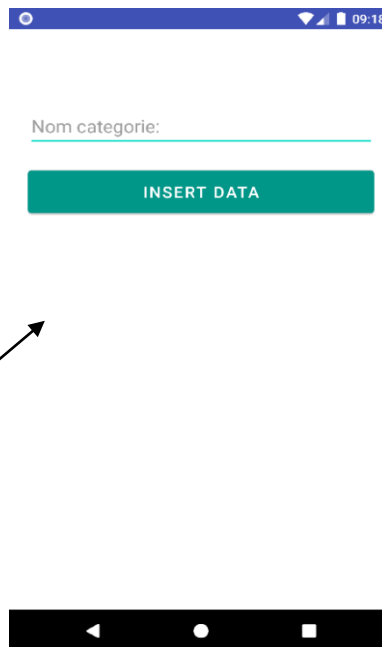


Figure 5.10 Interface d'ajouter une catégorie.

➤ Interface de liste des bricoleurs

Cette interface représente la liste des bricoleurs, elle contient deux listes (Figure 5.12): liste des bricoleurs qui confirmés par l'administrateur(Figure 5.13).Et liste des bricoleurs pas encore confirmé (Figure 5.11),leurs informations sont vérifiées (par le diplôme de bricoleurs) l'administrateur soit confirmés soit rejetés(Figure 5.14).

Chapitre 5 : Implémentation

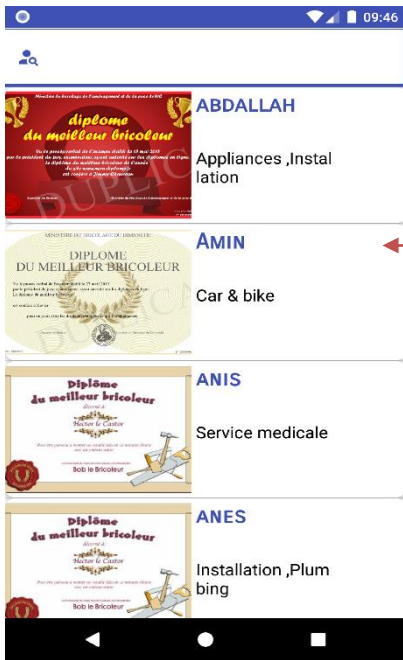


Figure 5.11 Interface de liste des bricoleurs non confirmés.

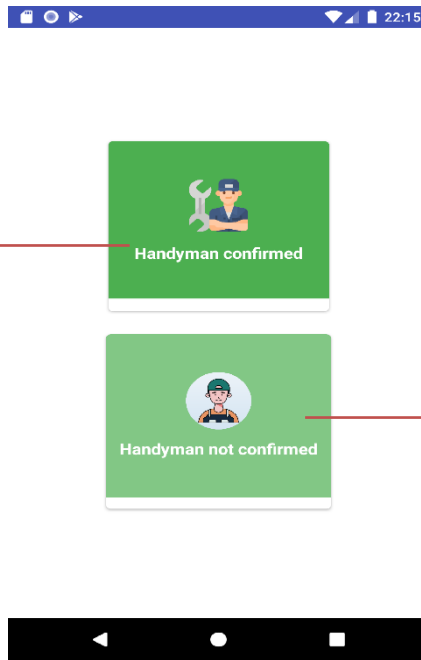


Figure 5.12 Interface de liste des bricoleurs.

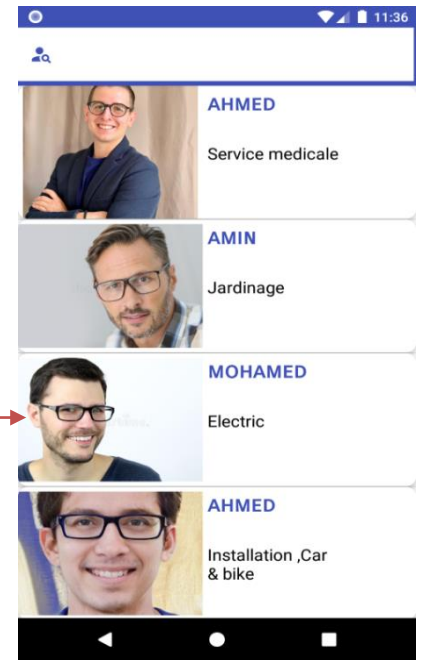


Figure 5.13 Interface de liste des bricoleurs confirmés

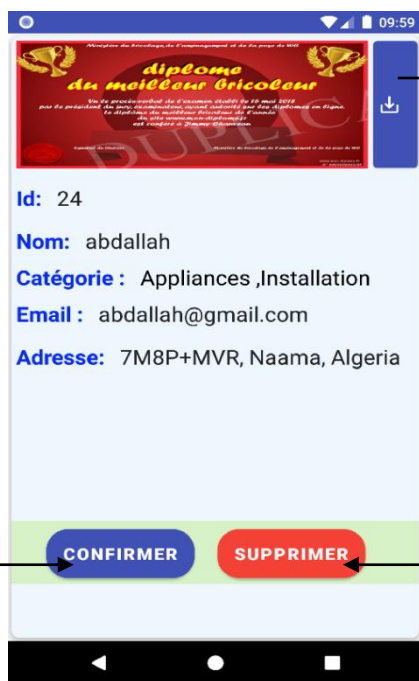


Figure 5.14 fiche d'information.

Télécharger le diplôme

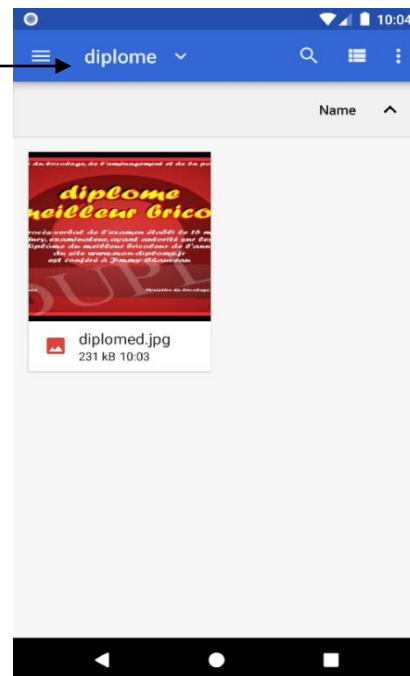


Figure 5.15 Interface de téléchargement le diplôme

Confirmé le
le bricoleur

Refusé le
le bricoleur

Chapitre 5 : Implémentation

➤ Interface de la recherche d'un bricoleur

La recherche d'un bricoleur par l'administrateur par nom de bricoleur (Figure 5.16)et(Figure 5.17).

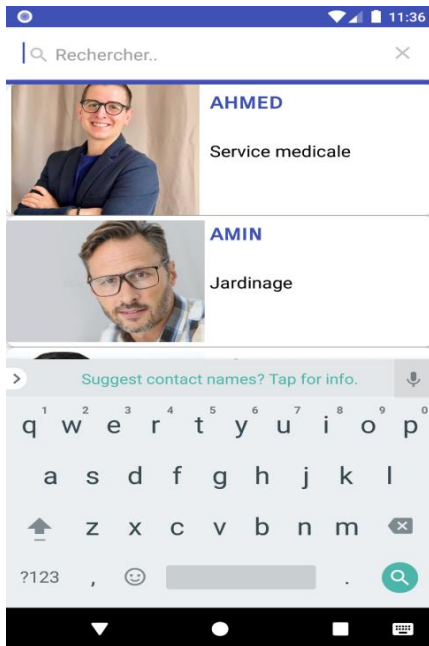


Figure 5.16 Interface de recherche par l'administrateur

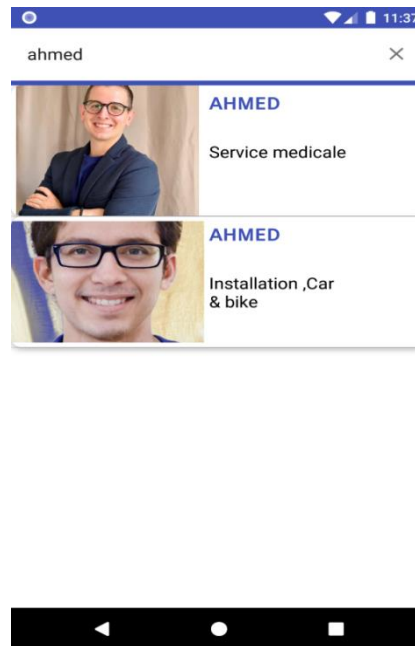


Figure 5.17 Interface de recherche par Nom de bricoleur.

5.3.4. Espace Client

➤ Interface de choix

C'est la première interface qui s'affiche après le lancement de l'application. (Figure 5.18).



Figure 5.18 Interface de choix.

Chapitre 5 : Implémentation

➤ Interface de d'inscription et d'Authentification de client

La figure 5.19 représente interface de l'inscription du client, et Figure 5.20 représente interface d'Authentification (saisir leurs mot de passe et leurs login afin d'accéder à leurs interface).

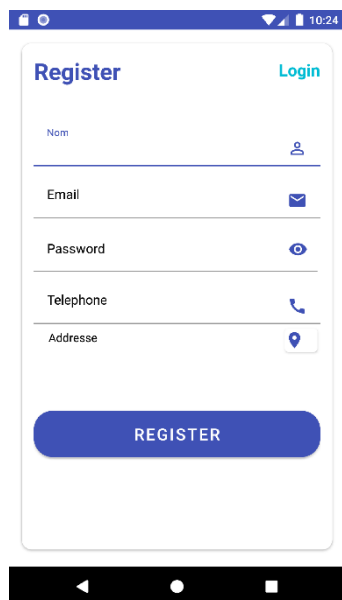


Figure 5.19 Interface d'inscription de client

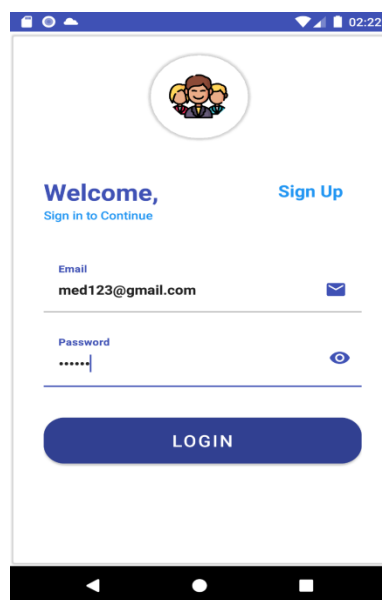


Figure 5.20 Interface d'Authentification de client.

➤ Interface de client après l'authentification

Cette activité représente l'interface de client après l'authentification, elle contient deux fragments ; une pour les catégories qui existe dans l'application et l'autre contient Google maps personnalisée.

Chapitre 5 : Implémentation

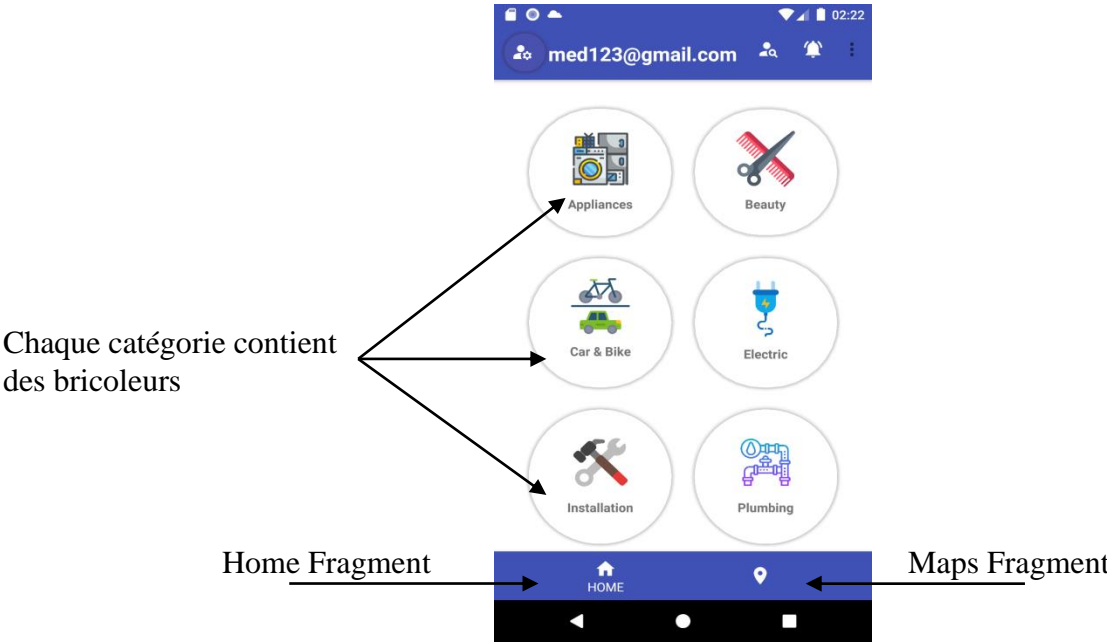


Figure 5.21 Interface des service bricolage.

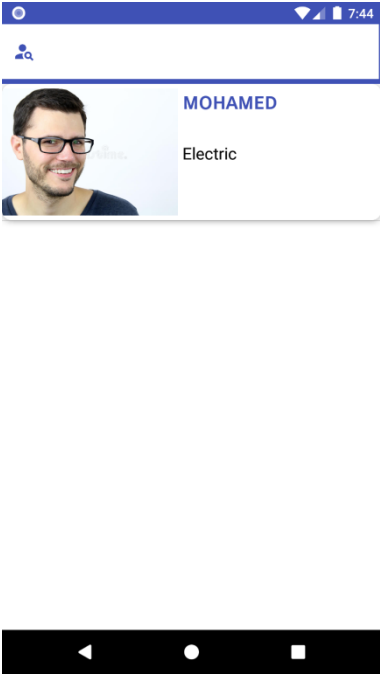


Figure 5.22 Interface de catégorie "Electric".

Chapitre 5 : Implémentation

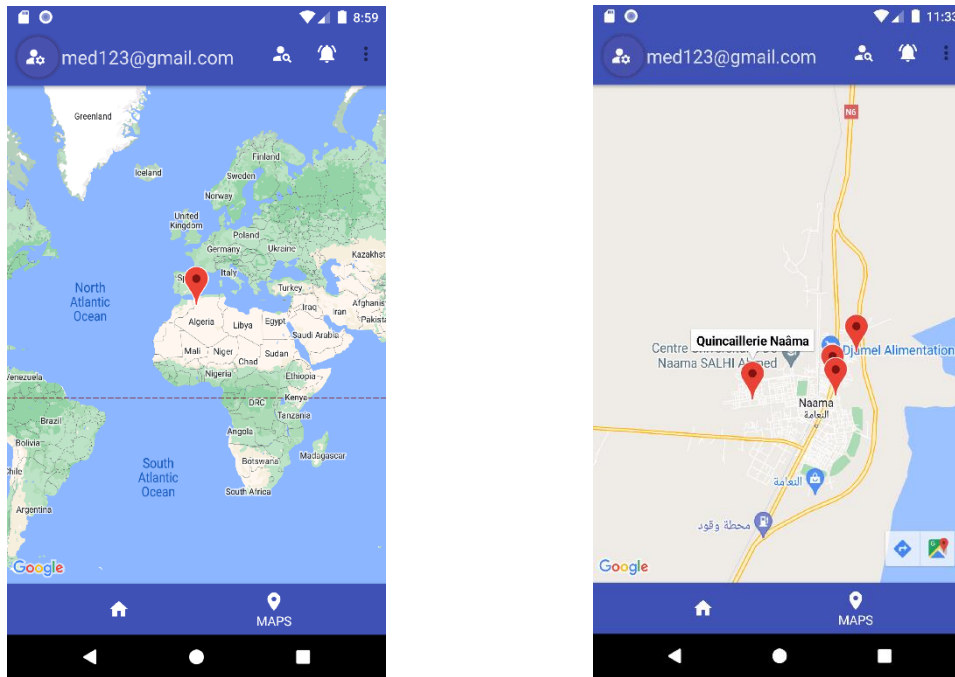


Figure 5.23 Interface de Google maps personnalisée.

➤ Interface de profil client

Cette interface représente le profil de client. Il peut être modifié et supprimer ses informations.

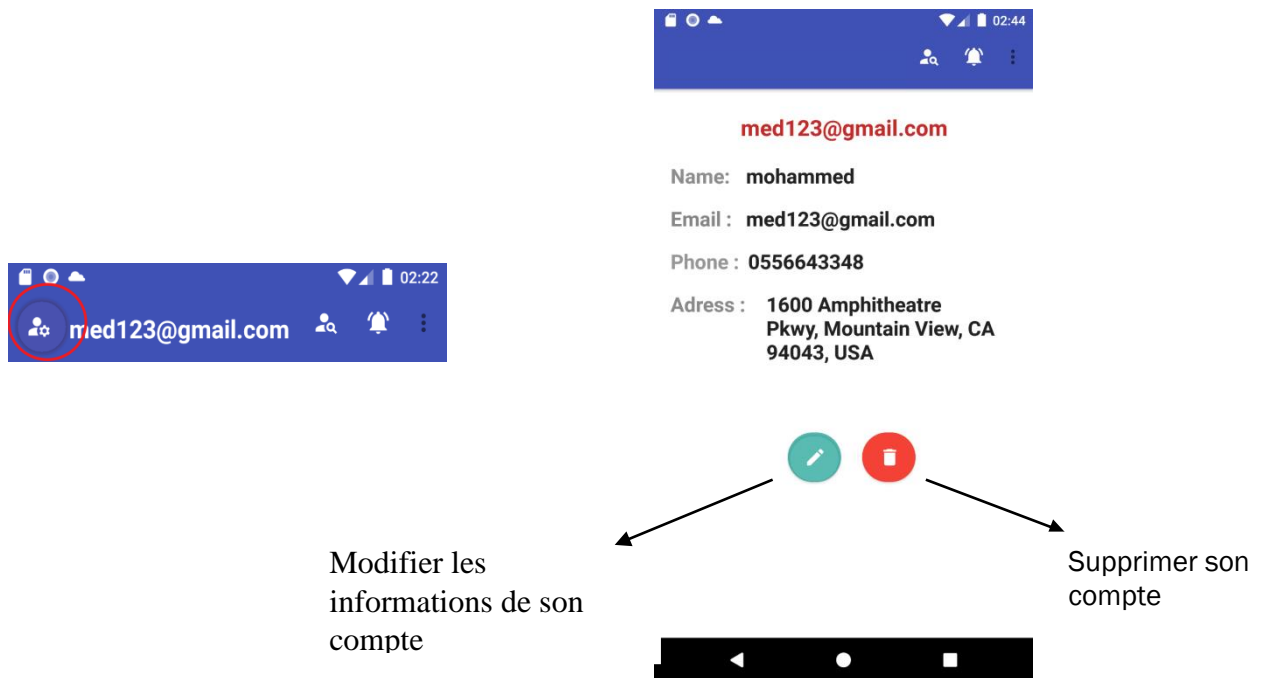


Figure 5.24 Interface de profil de client

Chapitre 5 : Implémentation

➤ Interface de la recherche d'un bricoleur

La recherche d'un bricoleur par client par nom de bricoleur (Figure 5.25).

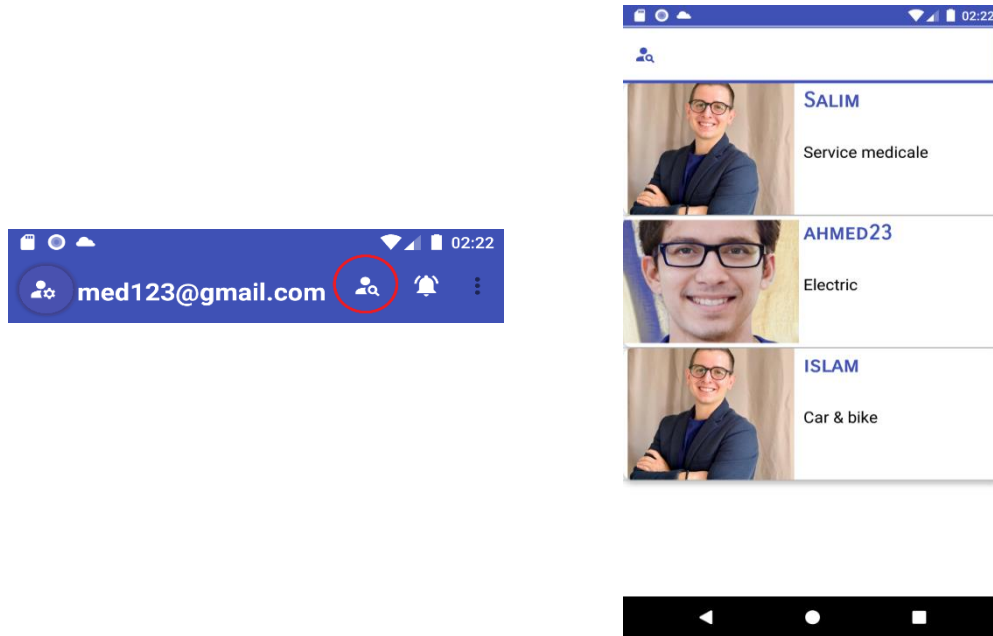


Figure 5.25 Interface de recherche par client.

➤ Interface de profil de bricoleur

Cette interface représente le profil de bricoleur avec les informations (nom, catégorie, email, téléphone et adresse). peut être prendre rendez-vous et faire appel téléphonique avec lui.

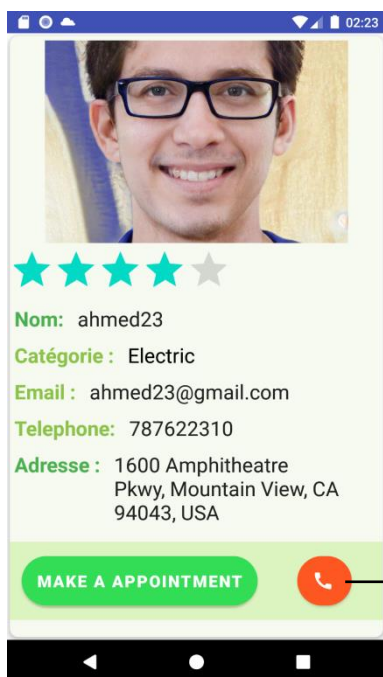


Figure 5.26 Interface de profil de bricoleur.

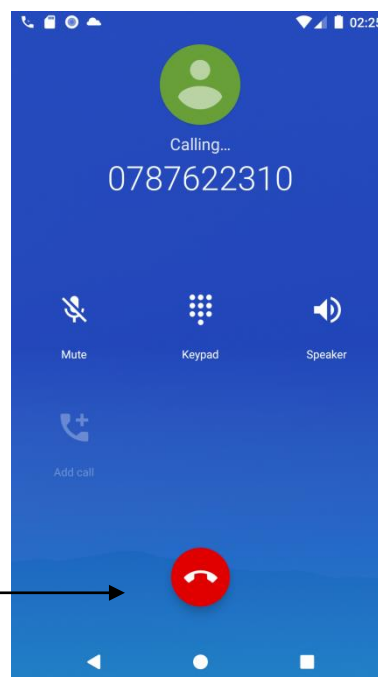


Figure 5.27 Interface de l'Appelle téléphonique

Chapitre 5 : Implémentation

➤ Interface de prendre rendez-vous (l'appointment)

le client peut être prendre rendez-vous avec bricoleur (Bouton Make a appointment) ,En attente d'approbation de bricoleur. Si le bricoleur est disponible, il envoie au client le rendez-vous (jour et heure) et celui-ci apparaît dans le bouton de notifications.

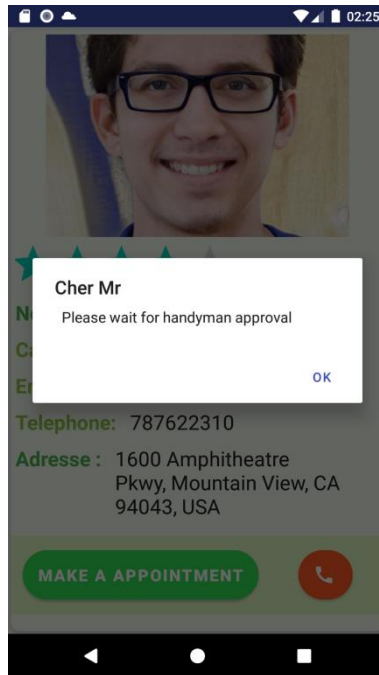


Figure 5.28 Interface de message d'attente d'approbation de rendez-vous.

Après approbation, le rendez-vous apparaîtra dans bouton de notification.

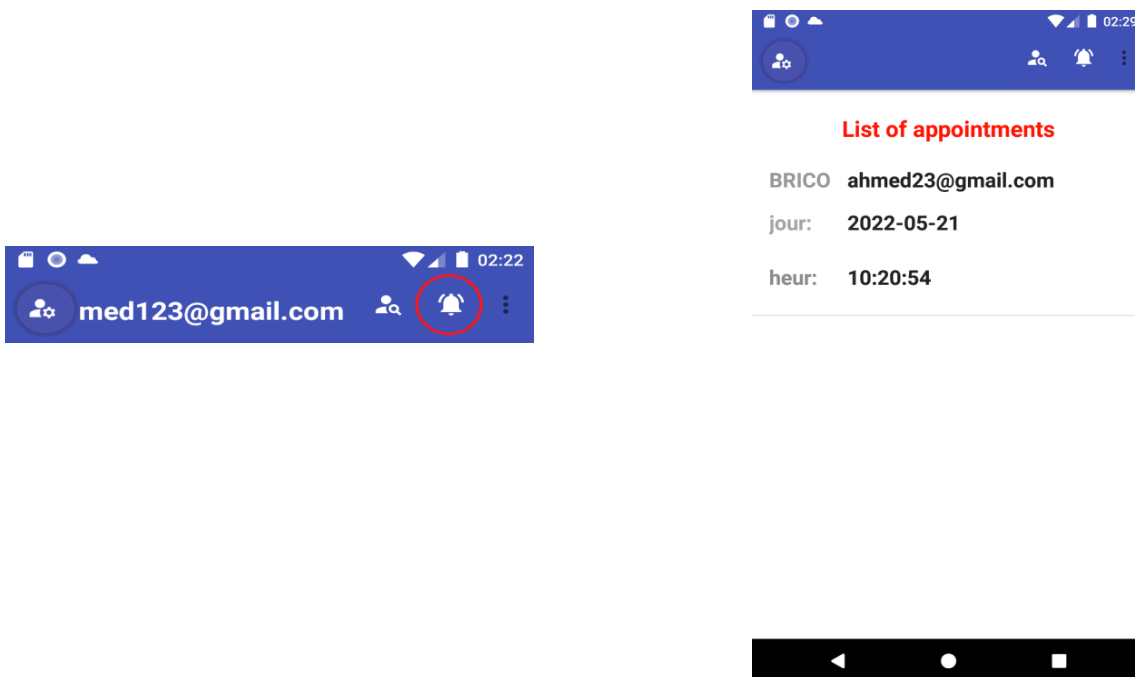


Figure 5.29 Interface de rendez-vous (jour et heure) Envoyé par le bricoleur.

Chapitre 5 : Implémentation

5.3.5. Espace Bricoleur

➤ Interface de l'inscription et l'Authentification

La figure 5.30 représente interface de l'inscription du bricoleur, et la figure 5.31 représente message d'attente de confirmation du bricoleur par l'administrateur , la figure 5.32 représente interface d'Authentification.

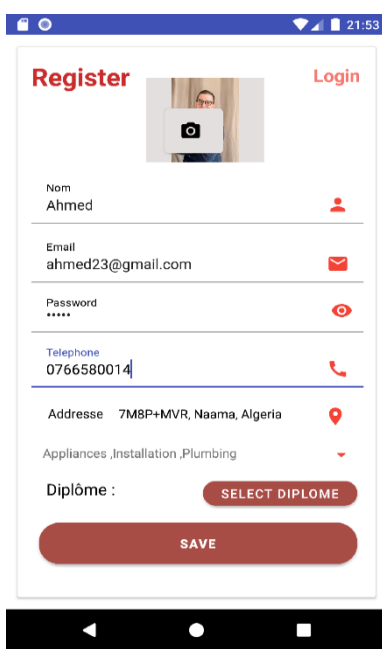


Figure 5.30 Interface d'inscription de bricoleur.

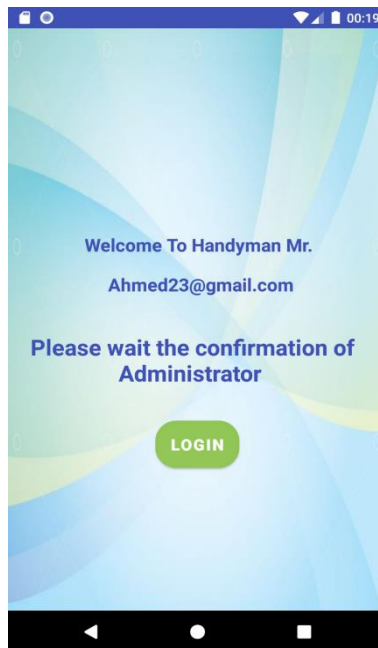


Figure 5.31 Interface de message d'attente de confirmation

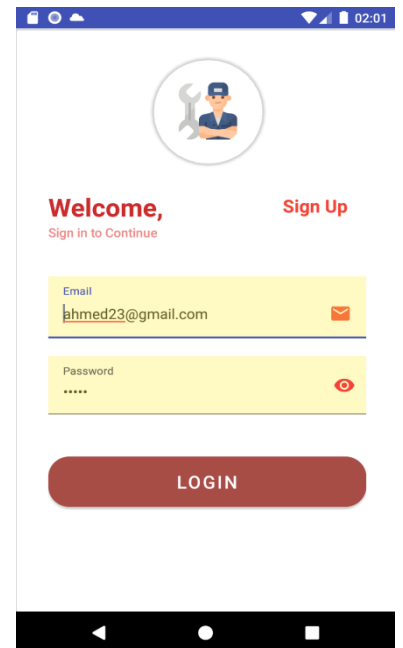


Figure 5.32 Interface d'Authentification de bricoleur après la confirmation d'admin.

➤ Interface de liste des rendez-vous

C'est la première interface affichée après l'Authentification, elle affiche le planning des rendez-vous.

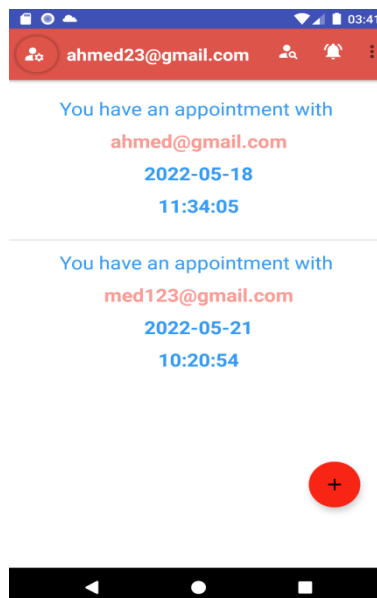


Figure 5.33 Interface de bricoleur

Chapitre 5 : Implémentation

➤ Interface de liste des demandes rendez-vous par client

Lorsque le bouton de notification est appuyé, il affiche les demandes de rendez-vous envoyées par le client.

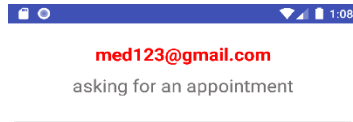


Figure 5.34 Interface de liste de demandes des rendez-vous

➤ Interface de fixer une date de rendez-vous avec client Le bricoleur fixe le rendez-vous(jour et heure) et l'envoi au client.

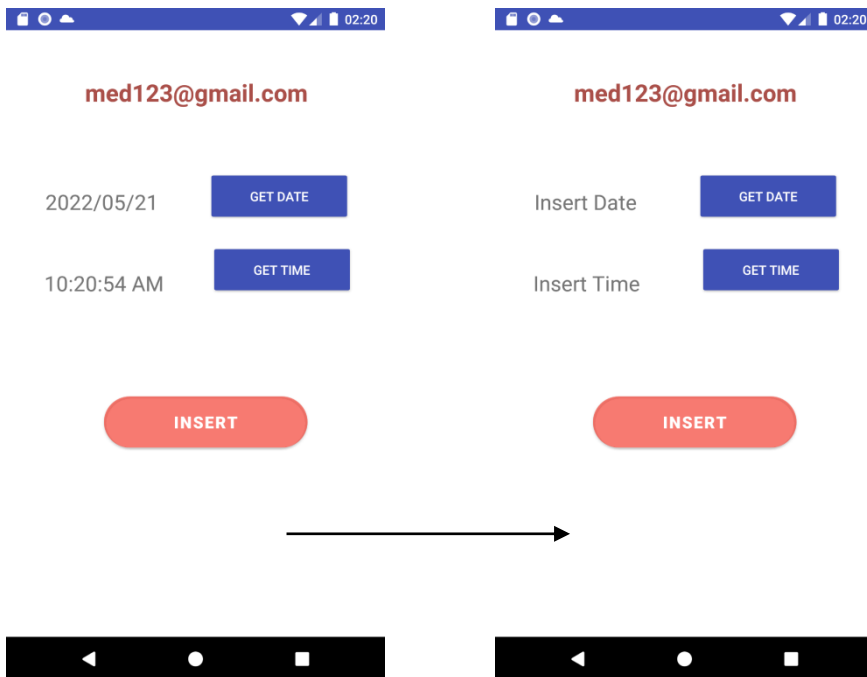


Figure 5.35 Interface de fixer une date et l'heur de rendez-vous.

Chapitre 5 : Implémentation

5.4. Conclusion

Dans ce chapitre, nous avons présenté l'environnement de développement, et les langages ainsi que les principaux outils utilisés qui nous ont permis de réaliser notre application « HEIMWERKER ZUHAUSE », et captures d'écrans sont aussi présentées afin de décrire plus en détails l'application réalisée.

Nous aurions aimé que se soit possible d'évaluer notre application. A défaut de ne pas avoir un dataset, nous n'avons pas pu proposer une expérimentation adéquate pour notre système. Une telle expérimentation pourrait nous permettre de tracer les améliorations futures possibles en se basant sur les éventuels résultats, ainsi que des feedbacks des utilisateurs.

Chapitre 6 : Conclusion et perspectives

Ce travail rentre dans le cadre de notre projet de fin d'étude pour l'obtention du diplôme de Master en informatique intitulé " Conception et Réalisation d'une Application Android : Handyman". Ce projet représente une concrétisation de cinq années d'études et de labeur. Notre objectif était d'utiliser nos connaissances afin d'aider notre communauté et faciliter quelques tâches de la vie quotidienne. D'où l'idée de créer une application Android pour les services de bricolage. Cette application vise à permettre à ses utilisateurs de rechercher un bricoleur selon leurs besoins, mais aussi à aider les bricoleurs à révéler leurs services.

Ce mémoire est débuté par une description générale des applications mobiles. Nous avons présenté en premier lieu les stratégies préconisées pour le développement de ces applications. Nous avons ensuite introduit les principaux systèmes d'exploitation mobiles sur lesquels repose leur fonctionnement. Dans la seconde partie, nous avons analysé les travaux similaires à notre proposition selon deux classes: les travaux locaux et internationaux, Dans la troisième partie, nous avons présenté la conception du système à l'aide de formalités UML. Finalement, la concrétisation de ce qui est conçu est illustré en détaillons l'implémentation de l'application mobile sous la plateforme Android.

Ce travail nous a permis d'apprendre avec précision, les différentes étapes de la conception et réalisation d'une application Android. Ce qui a permis également d'enrichir nos connaissances dans les différents langages et outils auxquels nous avons eu recours, à savoir JAVA, PHP, Android Studio, etc.. Au début du développement de l'application, nous avons été confrontés au problème de téléchargement d'Android Studio et à la difficulté de travailler sur ce logiciel.

Finalement, l'objectif a été atteint, mais tout de même l'amélioration de notre application reste envisageable. Nous pouvons citer quelques axes:

1. L'hébergement de l'application dans Google Play. Cela va permettre une diffusion de l'utilisation en plus d'un feedback important des utilisateurs. Ce feedback sert plus tard comme support pour guider le versionning de l'application selon les exigences et préférences des utilisateurs finaux.

2. L'évaluation de l'application après diffusion selon plusieurs facteurs, en l'occurrence :

- a. Taux de rétention (Retention Rate).
- b. Taux de désabonnement (Churn Rate).
- c. Utilisateurs actifs quotidiens (Daily Active Users (DAU)).
- d. Utilisateurs actifs mensuels (Monthly Active Users (MAU)).
- e. Retour sur investissement (Return On Investment (ROI)).
- f. Coût par acquisition (Cost Per Acquisition (CPA)).
- g. Revenu moyen par utilisateur (Average Revenue Per User (ARPU)).

3. Epauler l'application par un site web officiel pour permettre aux utilisateurs non-smartphones.

4. Améliorer l'application par de nouvelles fonctionnalités, telle que:

- a. Améliorer les services offerts par l'application en ajoutant par exemple les boutiques voisines et les Institutions gouvernementales.
- b. Connectivité au web sémantique par l'immigration de notre base de données vers un graphe Linked Data. Ceci va permettre une recherche plus affinée des services et des données associées.

Bibliographie

- [1] <https://www.definitions-webmarketing.com/definition-application-mobile/>, Consulté le 17/02/2022.
- [2] <https://www.sfactory.fr/les-applications-mobiles/>, Consulté le 17/02/2022.
- [3] **Mémoire de** BAGHLI ZINEB, BENMAZOUZ HAKIM "Conception et réalisation d'une application mobile sous Android de recherche de corps médicaux" université aboubeker belkaidlemcen 2014/2015.
- [4] <https://www.taktilcommunication.com/blog/applications-mobile/site-mobile-vs-application-mobile-avantages-et-inconvenients.html>, Consulté le 17/02/2022.
- [5] <https://www.astuces-aide-informatique.info/878/definition-systeme-exploitation>, Consulté le 17/02/2022.
- [6] <https://www.zdnet.fr/actualites/android-4000082258q>, Consulté le 18/02/2022.
- [7] <https://conciergeriedugeek.fr/lhistoire-dandroid/> , Consulté le 18/02/2022.
- [8] https://mathias-seguy.developpez.com/cours/android/oldch1_1_lifecycle/
- [9] **Livre de** P. Roques, UML 2 Modéliser une Application Web, 4e édition, Eyrolles, Paris, 2008.
- [10] **Livre de** J.GABAY et D.GABAY. UML 2 analyse et conception, Dunod, Paris, 2008.
- [11] https://www.lojiciels.com/reponse-rapide-quest-ce-quandroid-studio/#Quelle_est_la_definition_drsquoAndroid_Studio , Consulté le 21/03/2022.
- [12] <https://www.ionos.fr/digitalguide/serveur/outils/tutoriel-xampp-creer-un-serveur-de-test-local/>, Consulté le 03/04/2022.
- [13] <https://www.hostinger.fr/tutoriels/quest-ce-quapache-serveur-web-apache/>, Consulté le 03/04/2022.
- [14] <https://www.phpmyadmin.net> , Consulté le 21/03/2022.
- [15] **Livre de** Michel Divay ,LA PROGRAMMATION OBJETEN JAVA Cours et exercices corrigés,Dunod, Paris, 2006.
- [16] **Livre de** Philippe Rigaux ,PRATIQUE DE My SQL ET PHP Conception et réalisation de sites web dynamiques, 4eédition,Dunod, Paris, 2009.
- [17] <https://www.zdnet.fr/lexique-it/json-une-definition-39926165.htm> , Consulté le 04/04/2022.
- [18] <https://actualiteinformatique.fr/data/definition-sql-structured-query-language> , Consulté le 04/04/2022.