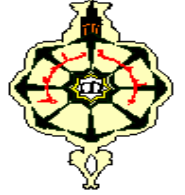


République Algérienne Démocratique et Populaire

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



UNIVERSITE ABOU BEKR BELKAID DE TLEMCCEN
FACULTE DE TECHNOLOGIE



DEPARTEMENT DE GENIE ELECTRIQUE ET ELECTRONIQUE



MEMOIRE

Présenté pour l'obtention du diplôme Master en Génie Industriel

Application Mobile pour la gestion des restaurants

Intitulé:

Présenté par:

M^r. SAHEL Alaeddine

M. SOUALHI Nacera

Soutenu le 19 septembre 2018 devant le Jury :

Mr. MALIKI Fouad.

MAA

Encadrant/ EPST/Tlemcen

Mr. Besnoussi. H

MAA

Examineur

Mr. Benkrouf

MAA

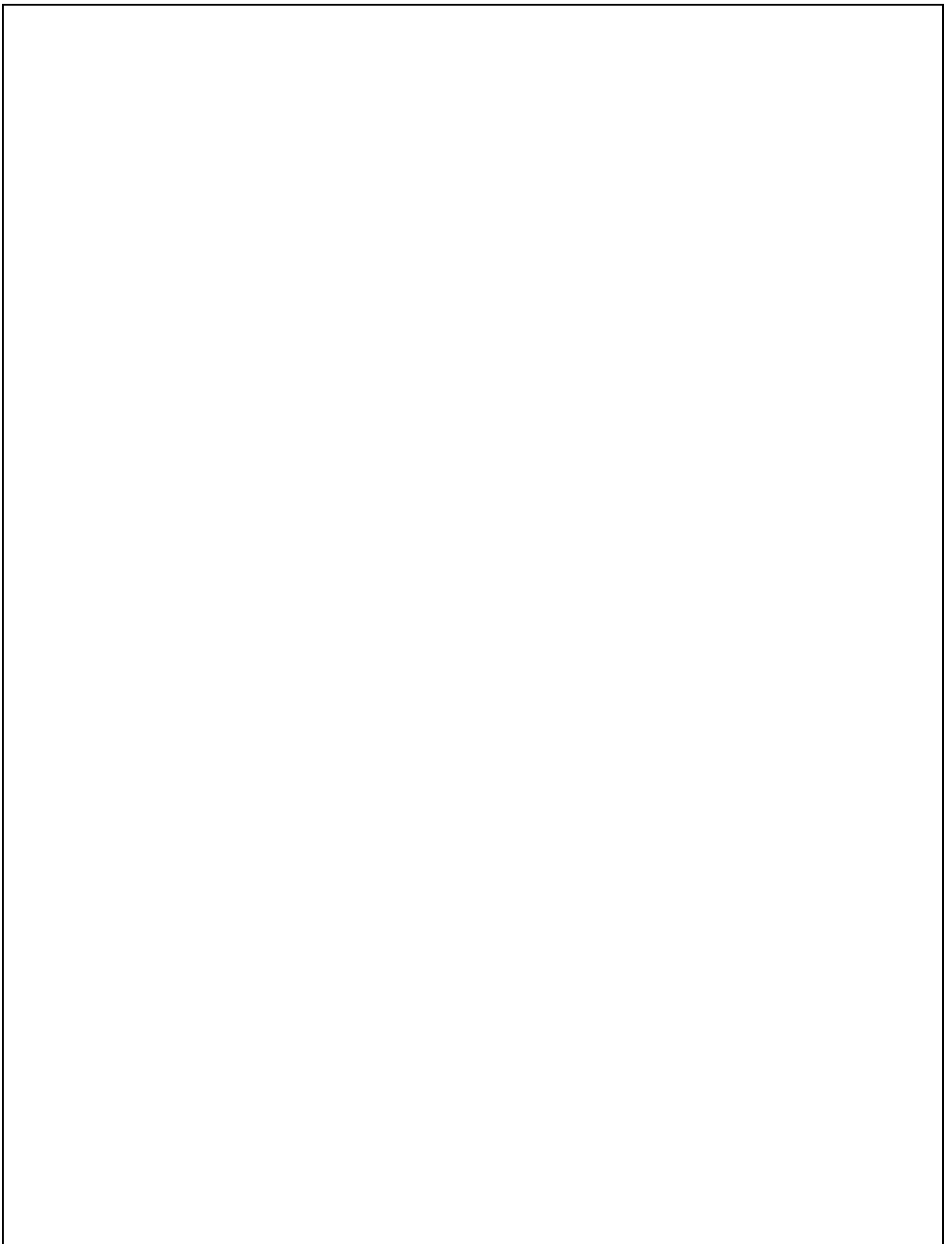
Président

Mr. Mékamecha

MAA

Examineur

Année universitaire: 2017/2018



Remerciements

Louanges à Dieu le tout puissant, grâce à qui nous avons pu faire tout ce travail. Nos remerciements s'adressent à toutes les personnes qui, d'une manière ou d'une autre nous ont permis de mener à son terme ce mémoire et plus particulièrement :

A notre encadrant, Mr Maliki Fouad qui nous a grandement aidés à définir avec circonspection le sujet de notre travail, et qui par ses conseils et idées, nous a permis d'améliorer la qualité du contenu. Son aide technique a été primordiale pour avancer dans notre projet.

Nos vifs remerciements vont aux membres de Jury, pour avoir accepté de juger notre travail.

A nos enseignants, à qui nous devons notre formation, qu'ils trouvent ici l'expression de nos sentiments les plus respectueux et de notre profonde gratitude pour leur gentillesse, leur disponibilité et leur soutien.

A nos chers parents, nos frères et sœurs et nos familles qui nous ont toujours soutenus le long de notre formation.

A tous nos amis, pour leur soutien morale et aide précieuse.

A Fayçal pour son aide précieuse et pour le temps qu'il a bien voulu nous accorder.

Dédicace

Moi S. Alaeddine,

Je dédie ce travail à tous ceux qui ont cru en moi et m'ont soutenue : A mes très chers parents, mes deux soeurs Hadia et Hanae, Mon frere Yakin, que j'aime énormément,

A tous mes cousins et cousines. A toute ma famille Sahel.

A mes amis : Aziz, Fayçal, Khalil, Mustapha, Ahmed T, Akrem, Adlene, Hamza, Sidali, Mono, Dahmoun, Zahi, Aslonie, Abdou, Abdelkhalek, Hadjer, Ghizlan, Hadjira, Titouni, Sarah, Yasser, Bouftou et Sidou, que je n'oublierais jamais.

Moi S. Nacera, je dédie ce modeste travail à : Mes très chers parents pour leur soutien et leur noble sacrifice afin de me voir réussir.

Mon frère « Mohamed lamine », et mes sœurs « Nedjma », « Souad » et « Houda », que j'aime beaucoup.

A mes amis : Hasna, Nour el Houda, Thiziri, Khadidja, Khawla, Imen, Nadir, Amani, Soumia, Ilies, Yahia, Mina, Hiba, que je n'oublierais jamais.

A toute ma famille, ma belle-famille Tous ceux qui ont supporté mon absence durant toutes ces années Tous ceux qui ont contribué de près ou de loin à l'aboutissement de cette thèse A tous mes amis Tous ce qui m'aime

Liste Des Figures

Figure 1.1 : iPhone et ces applications	5
Figure I.2: IOS (operating System)	6
Figure 1.3 : Evolution des versions d'Android.....	7
Figure 1.4: Windows phone (Operating System)	7
Figure 1.5: BlackBerry (Operating System).....	8
Figure 1.6: Symbian (Operating System)	9
Figure 1.7: Architecture Android.....	13
Figure 2.1 : Diagramme des cas d'utilisation global.....	28
Figure 2.2 : Diagramme de séquence d'inscription.....	29
Figure 2.3 : Diagramme de séquence d'authentification.....	30
Figure 2.4 : Diagramme de séquence : commander.....	31
Figure 2.5 : Diagramme de séquence : Modifier/supprimer un sandwich.....	32
Figure 2.6 : Diagramme de séquence : ajouter un sandwich.....	33
Figure 2.7 : Diagramme de séquence : Répondre aux clients.....	34
Figure 2.8 : Diagramme d'activité inscription.....	35
Figure 2.9 : Diagramme d'activité authentification.....	36
Figure 2.10 : Diagramme de classe.....	37
Figure 3.1 : Android Studio.....	43
Figure 3.2: Structure d'un projet Android	44
Figure 3.3 : SDK Manager.....	45
Figure 3.4: FirebaseConsole.....	46
Figure 3.5 : GENYMOTION.....	47
Figure 3.6: Architecture de la solution globale	48
Figure 3.6: Connexion ou inscription	50
Figure 3.8 : Création d'un nouveau compte	51

Figure 3.9 : Vérification des données.....	52
Figure 3.10 : Liste des Restaurants.....	53
Figure 3.11: Liste des Sandwichs.....	54
Figure 3.12 : Choisir un Sandwich.....	55
Figure 3.13 : La Barre de Menu.....	56
Figure 3.14 : Les Sandwich commandées.....	57
Figure 3.15 : Ajouter un commentaire et confirmer.....	58
Figure 3.16: Le statut de la commande.....	59
Figure 3.17 : Liste des Sandwichs coté du restaurant.....	60
Figure 3.18 : Ajouter un nouveau Sandwich.....	61
Figure 3.19 : Liste des commandes.....	62
Figure 3.20 : Changer le statut de la commande.....	63

Liste des Tableaux

Tableau 2.1 : les diagrammes utilisé.....	23
---	----

Introduction générale

Introduction générale

La technologie de la téléphonie mobile continue à connaître des progrès et un développement surprenant : ce petit appareil, qui a trouvé sa place dans notre vie quotidienne en quelques années et qui est devenu l'appareil le plus vendu au monde.

Bref, même les services fournis par cette appareil est allée si loin. Là où le mobile n'est plus un moyen de communication et d'échange de messages uniquement, mais il est devenu un outil polyvalent en raison des applications intelligentes qui ont été utilisées dans de nombreux domaines de notre vie, et parmi ces domaines la commande en ligne des repas.

Alors, 94% des utilisateurs des Smartphones choisissent les lieux à proximité et 90% d'entre eux se déplacent dans les 24 heures, ce qui a causé une augmentation du marché des applications qui font des commandes en ligne, les revenus des restaurants ont augmenté de 160 millions de dollars en 2016. Les gens commandent des collations au moyen d'appareils mobiles et beaucoup de gens ont tendance d'acheter de la nourriture par Internet.

Notre projet consiste à mettre en place une application mobile dont le but est d'offrir la possibilité aux fonctionnaires, étudiants, habitants...etc. de commander des sandwichs sans être obligés à se déplacer ou perdre du temps en attendant que la commande soit prête. On va entamer cette application dans la ville de Tlemcen. Dans notre prototype on a utilisé que cinq (5) restaurants (L'échappatoire, Maxi, Le gourmet, Chick'n, Cyno), chacun son propre menu. Pour ce faire, notre choix s'est porté sur la plateforme mobile Android, qui est gratuite et qui englobe une communauté importante par rapport à d'autres plateformes.

Notre application mobile aide également les clients à voir les offres, à savoir quand recevoir les sandwichs et à bien parcourir la liste avant de commander en sachant tout ce qui concerne les sandwichs. Elle aide aussi les restaurants pour ne pas avoir un encombrement (sur tout dans les week-ends), et les rapproche de ses clients.

Le présent mémoire est divisé en trois chapitres décrivant les étapes suivies pour la réalisation de notre application mobile.

- le premier chapitre consiste à étudier les technologies mobiles en s'appuyant sur les applications mobiles et Android.
- Le deuxième chapitre aura comme but de présenter les différentes étapes de conception de notre l'application mobile.

- Nous décrivons dans le chapitre trois l'environnement de développement de notre application et les langages utilisés suivi d'une présentation des différentes interfaces de notre application mobile.

Enfin, nous clôturons ce mémoire par une conclusion générale résumant les points essentiels de notre projet et dégageons quelques perspectives envisagées pour notre application mobile.

Chapitre I :

La Technologie Mobile

1.1 Introduction :

Aujourd'hui, le téléphone mobile ne se limite pas aux appels et à l'échange de messages courts comme autre fois, mais De nouveaux usages sont apparus tels que les jeux, la lecture audio, la musique etc..... Sur un plan plus pratique, Le téléphone mobile dispose des variétés d'utilisations et l'appareil est devenu plus utile que jamais auparavant. L'essor du mobile a connu une évolution considérable en particulier dans les pays en développement. Les réseaux de télécommunication mobiles sont également en expansion : le nombre d'utilisateurs de mobile est en constante progression et la couverture territoriale est largement répandue.

Avec l'utilisation généralisée des téléphones mobiles à large bande et le développement de logiciels et de réseaux, les applications mobiles peuvent répondre à un large éventail de besoins ainsi qu'à un développement technologique continu qui les rend plus importants. Les applications mobiles sont devenues une partie importante de notre vie quotidienne. Il existe plusieurs types, les applications de téléphonie mobile couvrent un large éventail de domaines, ces applications sont considérées comme l'étendue de la communication avec le monde extérieur.

Dans ce chapitre nous allons parler sur la technologie mobile. Ensuite, nous présentons les différentes notions des applications mobiles et ses Caractéristiques, types, Statistiques sur le marché, Les différentes plateformes mobiles, Les domaines utilisant les applications mobiles, Les différents types d'applications mobiles, Eléments de comparaison entre les applications mobiles, Puis, les réseaux sociaux comme application mobile et après l'application mobile de restaurant. On terminera par une simple conclusion.

1.2 La technologie mobile :

Né dans les années 1980, le téléphone mobile a connu une diffusion très rapide. Aujourd'hui, pour une population mondiale de 7 milliards de personnes, on compte plus de 6.8 milliards d'abonnés au téléphone mobile. Soit environ 96.2% des habitants (UIT-2016). [1]

C'est véritablement un nouveau secteur de l'industrie mondiale qui s'est créé, regroupant notamment constructeurs de circuits électroniques, de terminaux mobiles, d'infrastructures de réseaux, développeurs d'applications et de services et opérateurs de réseaux mobiles. [1]

Les progrès technologiques dans le domaine des réseaux de télécommunications mobiles, ont vu l'apparition des technologies numériques au début des années 1990. En Europe (**GSM**), au Japon (**PDC**) et aux Etats Uni (**PCS**) [2]. L'évolution du réseau radio mobile analogique

(1G) vers GSM (dit de 2ème génération « 2G ») après l'UMTS (dit de 3ème génération « 3G ») ensuite vers la « 4G » (4ème génération) passe par des générations intermédiaires comme le **GPRS**, **HSCSD** ou **EDGE** (dites « 2.5G »), **HSDPA** (3.5G) et **HSUPA** (3.75G) et prochainement la 5G pas encore mise en œuvre. Ces différentes générations exploitent des terminaux modernes tels que les Smartphones.

Le Smartphone est un compagnon de tous les instants et un véritable ordinateur de poche pour gérer le quotidien. Ce dernier est passé en l'espace de quelques années d'un outil professionnel à un objet de divertissement grand public. Il désigne un téléphone portable multifonctions. Selon le principe d'un ordinateur, il peut exécuter divers applications grâce à un système d'exploitation spécialement conçu pour mobiles, et donc, en particulier, fournir des fonctionnalités en plus de celles des téléphones mobiles classiques comme : la navigation sur le Web, la géolocalisation, la cartographie numérique, etc.[3].

L'arrivée sur le marché de cette nouvelle catégorie de terminaux puissants et financièrement adorables est le principal vecteur de plusieurs innovations dans divers domaines et notamment dans le domaine médical, sauvetage, commerce, sécurité routière, ...

Les technologies mobiles prennent de plus en plus de place sur le marché. Les Smartphones sont considérés comme des petits ordinateurs dotés d'un système d'exploitation s'appuyant sur un noyau Linux. Cependant, ils diffèrent des ordinateurs classiques par le cycle de vie d'une application. Sous Android, une application est composée d'une ou plusieurs activités. Une activité est la base d'un composant pour la création d'interfaces utilisateur. Afin de faciliter la cinématique de l'application, il est préconisé de n'avoir qu'une interface visuelle par activité.

1.3 Les applications mobiles :

1.3.1 Définition :

Une application mobile est un logiciel applicatif développé pour être installé sur un appareil électronique mobile, tel qu'un téléphone portable « Smartphone » [4]. Comme elle est un programme téléchargeable de façon gratuite ou payante et exécutable à partir du système d'exploitation du téléphone. [5] Ce petit logiciel s'appuie d'une manière générale sur le principe de widgets que nous connaissons sur nos ordinateurs.

Pour télécharger une application sur un téléphone mobile, il existe différentes possibilités :

- Transfert depuis un ordinateur via un câble de connexion,

- À partir d'un service mobile,
- À partir d'un service mobile,
- Via une boutique logicielle accessible depuis un téléphone mobile (App Store d'Apple, Windows Market Place, Nokia OVI, AndroidMarket, etc.),
- Le cas échéant l'application est dite native ; elle est déjà dans le téléphone lors de l'achat du téléphone (l'opérateur ou le fabricant l'a ajouté comme fonction de base). [6]



Figure 1.1 : iPhone et ces applications en détail [7]

1.3.2 Caractéristiques d'application mobile

Des contraintes techniques qu'il est nécessaire de prendre en compte lors de la conception d'une application mobile :

- Tailles d'écrans variables, pouvant dans certains cas être assez réduite,
- Possibilité limitée de saisie de données,
- Puissance du processeur, pouvant être limité sur les premiers Smartphones,
- Tailles de la mémoire pouvant varier,
- Autonomie du Smartphone,
- Débits variables de la bande passante Internet.

Outre ces aspects techniques, il est également important de prendre en compte l'ergonomie de l'application mobile à réaliser, et c'est un point crucial à ne pas négliger. Une application mobile doit respecter certaines règles :

- Utiliser des images petites et légères,
- Utiliser des éléments facilement accessibles,
- Maitriser l'utilisation du JavaScript pour économiser la batterie,
- Adapter le mode de saisie des informations. [8]

1.3.3 Les différentes plateformes mobiles :

Le marché des téléphones mobile est aujourd'hui dominé par cinq grandes entreprises de technologie Smartphone qui sont Apple, **RIM**, Google, Microsoft et Nokia qui développent respectivement les systèmes d'exploitation Ios, BlackBerry OS, Android, Windows Phone 7 et SymbianOS. Dans ce volet, nous allons présenter brièvement chacun des systèmes, voir leurs avantages et inconvénients pour connaître le leader dans le marché des Smartphones et déterminer le système qui pourra répondre le plus aux besoins de l'application.

❖ IOS :

Est le système d'exploitation mobile développé par Apple pour l'iPhone, l'iPod touch, et iPad. Il est dérivé de Mac OSx dont il partage les fondations (le Keren hybride XNU basé sur le micronoyau Mach, les services Unix et Cocoa, etc.). IOS comporte quatre couches d'abstraction, similaires à celles de Mac OS X : une couche « Core OS », une couche « Core Services », une couche « Media » et une couche « Cocoa ». [9]



Figure I.2: IOS (operating System) [9]

❖ Android OS :

Android a été développé par l'Open Handset Alliance. Il a été annoncé en 2007 et il est devenu une plateforme ouverte en 2008. Android est un OS gratuit et complètement ouvert.

C'est-à-dire que le code source et les APIs sont ouvertes. Ainsi, les développeurs obtiennent la permission d'intégrer, d'agrandir et de remplacer les composants existants. [10]



Figure 1.3: Evolution des versions d'Android. [7]

❖ **Windows Mobile (Windows Phone) :**

Windows Mobile a été développé par l'entreprise Microsoft. Pour cette raison, Windows Mobile fonctionne uniquement sous « Windows », il est compatible avec tous les logiciels Windows. Il offre la possibilité de télécharger, de jouer les chansons et de regarder la TV en ligne. Windows Mobile supporte aussi beaucoup de types d'audio et de vidéo.



Figure I.6: Windows phone (Operating System). [11]

❖ BlackBerry :

BlackBerry a été créé par Research In Motion (RIM). BlackBerry fournit le service électronique pour les entreprises en utilisant BlackBerry Enterprise Server. De plus, le fournisseur offre aux abonnés des fonctions push de messagerie et d'accès mobile pratique qui a du contenu Internet pour leurs terminaux. Il comporte aussi la technologie de la pièce jointe qui supporte divers types de pièces jointes telles que les fichiers d'extensions .zip, .html, .doc, .dot, .ppt, .PDF, etc. C'est pourquoi son service de messagerie électronique est meilleur que les autres plateformes. [12].



Figure I.4: BlackBerry (Operating System). [12]

❖ Symbian OS :

Est un système d'exploitation pour téléphones portables conçu par Symbian Ltd. Il est l'héritier du système d'exploitation EPOC32 qui équipa les Psion, et est né d'un consortium entre les différents constructeurs (souvent des Nokia). Il dispose de nombreuses **API** spécifiques pour la communication mobile voix et données, et implémente la majorité des standards de communication : IPv4/IPv6, MMS, Bluetooth, **GPRS/UMTS**, Java, WAP, SyncML... Il a été adopté par différents fabricants de téléphones portables de deuxième génération (**GSM** et **GPRS**) et troisième génération (**UMTS**). [13]



Figure I.5: Symbian (Operating System) [13]

I.3.6 Les domaines utilisant les applications mobiles [14]

- Avec les possibilités matérielles incorporées aux terminaux (caméra, **GPS**, gyroscope, ...), les applications Smartphones et Tablettes peuvent intégrer des fonctionnalités spécifiques et dédiées pour les utilisateurs, permettant ainsi d'enrichir le spectre fonctionnel et imaginer des usages non couverts jusqu'à présent par les systèmes d'information.
- Géolocalisation, Itinéraires
- Scan de Code barre, Flash, QR Code
- Réalité augmentée
- M-commerce, Paiement mobile
- Push et notification
- Gestion de documents, dématérialisation, Workflow
- Analyse d'Audience
- Gestion et Sécurisation de parc et de déploiement de terminaux mobiles



Figure 1.6: domaine d'application mobile [7]

I.3.7 Les différents types d'applications mobiles :

On peut distinguer plusieurs solutions pour créer ces applications mobiles : les applications natives, les applications web et les applications hybrides. [15]

a) Application native

Une application mobile native est une application mobile spécifique à un système d'exploitation mobile, développée avec le langage et les outils associés fournis par l'éditeur du système d'exploitation mobile, et installée directement sur le mobile. Cette installation de l'application native se faisant soit au travers d'un téléchargement via Internet soit par déploiement depuis un ordinateur connecté au mobile.

Aspects dominants : Accès à toutes les fonctionnalités du mobile, rapidité d'exécution, possibilité de notifier (push) l'utilisateur, eco-système associés aux magasins (store) d'application permettant de générer des revenus.

b) Application web ou Web App

L'application mobile web est un site Internet qui a été adapté pour une restitution sur un navigateur de mobile. De la même manière que les sites web, une application mobile web est généralement placée sur un serveur **HTTP** et se manipule en actionnant des éléments d'interface XHTML à l'aide d'un navigateur web. Aspect dominant : une application mobile web adresse l'ensemble des utilisateurs de mobile, et non une population spécifique utilisant une marque précise de mobile.

c) Applications hybrides

Une application hybride est un mélange de code natif et d'affichage de vues **HTML/JavaScript**. Concrètement toutes les plateformes mobiles proposent un composant de type **WebView**, permettant d'afficher du contenu web soit sur une partie de l'écran, soit en plein écran, et en utilisant le moteur **HTML** du navigateur intégré au système. Ces applications hybrides peuvent être distribuées sur les stores des systèmes mobiles.

Plusieurs stratégies sont alors possibles, selon que l'on place le curseur plus du côté natif ou plus du côté **HTML** :

- Ne réaliser que certains écrans voir même que certains composants d'IHM en **HTML**.
- Réaliser tous les écrans en **HTML** mais garder la logique applicative en code natif, notamment les effets de transitions entre écrans et la gestion du scrolling.
- Réaliser les écrans en **HTML**, et les transitions / scrolling en **JavaScript**. Le code natif peut alors se cantonner à quelques composants techniques très ciblés. De la même manière selon les applications la logique métier peut être codée en **JavaScript** ou bien en code natif.

1.3.8 Eléments de comparaison entre les applications mobiles : [15]

❖ Coûts de mise en œuvre

Développer une application native pour plusieurs plateformes mobiles peut coûter très cher, de par la multitude de langages et technologies mises en œuvre. Selon le nombre de plateformes cibles, une technologie web ou même hybride sera souvent moins coûteuse. De plus il sera souvent plus simple de disposer de développeurs maîtrisant les technologies web, que les diverses plateformes mobiles.

❖ **Qualité, rapidité des applications**

Difficile de rivaliser avec les applications natives, Celles-ci seront presque toujours plus rapides. Mais cela dépend fortement du type d'application en jeu et de ses fonctionnalités.

❖ **Publication et mises à jour**

Une importante contrainte des applications natives est que celles-ci doivent être approuvées avant diffusion sur leur store respectif (sauf pour Android), ce qui peut s'avérer long et contraignant. Une application hybride permet de limiter ce désagrément, et un web app de s'en affranchir complètement. Le même problème se pose pour les mises à jour, il n'est souvent pas possible de diffuser un patch correctif en urgence ou même rapidement sur une application native.

❖ **Monétisation**

Les magasins d'applications permettent très facilement de vendre les applications, mêmes si Apple, Google et consorts prélèvent leur part sur les prix de vente, généralement autour de 30%. Même si les stores d'applications web comment timidement à apparaître, leur usage est encore très restreint. Les stores servent aussi de moteur de recherche et de vitrines pour les applications, et permettent ainsi de les mettre en avant et de les faire découvrir.

1.4 ANDROID

En 2015, Android est le système d'exploitation le plus utilisé dans le monde avec plus de 80 % de parts de marché dans les Smartphones.

1.4.1 ARCHITECTURE ANDROID :

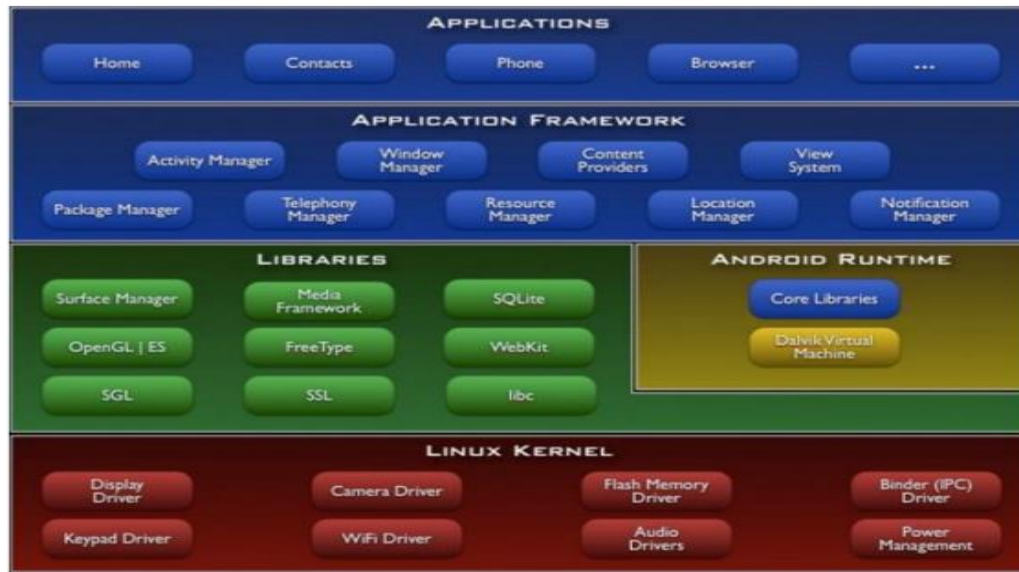


Figure 1.11: Architecture Android [16]

On remarque que la figure I.6 présente l'architecture Android qui se compose de plusieurs parties :

a) Partie applications

Toute l'application Android se trouve à la couche supérieure.

b) Partie applications framework

La couche Application Framework fournit de nombreux services de haut niveau à des applications sous la forme de classes Java

c) Partie libraries

Android fournit un ensemble de librairies C/C++ .Ces fonctionnalités sont rendues disponibles aux développeurs a travers le Framework d'application d'Android.

d) Partie Android runtime

Le runtime Android fournit également un ensemble de librairies fournissant des fonctionnalités des librairies standards java.

e) Partie Linux Kernel

Android s'appuie sur le noyau Linux 2.6 pour les services système de base tels que la sécurité, la gestion de la mémoire et des processus, le réseau et la gestion des drivers.

1.4.2 Composition d'une application

Une application Android peut être composée des éléments suivants :

- Activités (android.app.Activity): il s'agit d'une partie de l'application présentant une vue à l'utilisateur.
- Services (android.app.Service): il s'agit d'une activité tâche de fond sans vue associée.
- Fournisseurs de contenus (android.content.ContentProvider): Permettent le partage d'informations au sein ou entre applications.
- Widgets (android.appwidget.*): Une vue accrochée au Bureau d'Android.
- Intents (android.content.Intent): permettent d'envoyer un message pour un composant externe sans le nommer explicitement.
- Récepteurs d'Intents (android.content.BroadcastReceiver): permettent de déclarer être capable de répondre à des Intents.
- Notifications (android.app.Notifications): permettent de notifier l'utilisateur de la survenue d'événement

1.4.2 Les avantages et les inconvénients d'Android :

a) Avantages

Le système d'exploitation Android de Google est capable d'exécuter plusieurs tâches à la fois :

- Plusieurs applications peuvent être lancées à la fois.
- Notifications dans le temps des SMS, messagerie de Gmail.
- Les marques les plus connues tels que HTC, Samsung, LG, ou encore Sony Ericsson possèdent un système d'exploitation Android, ce qui laisse le choix à chaque personne de choisir sa marque de prédilection.
- Possibilité d'installer une Rom modifiée pour donner une touche personnalisée à votre téléphone Android.

b) Inconvénients

Android présente néanmoins quelques inconvénients tel que :

- Connexion en "mode continu" et donc en permanence.

- Les téléphones Android possèdent de faible autonomie. Ils se déchargent vite à cause de la de la connexion.
- Certaines applications sont incompatibles à des versions de Google Android.
- Bug de certaines applications.
- Des applications inutiles sont installées, ce qui entraine sur certains modèles une insuffisance d'espace.

1.5 Une application pour les restaurants

L'application mobile de restaurant va permettre de créer un lien direct avec les clients sans intermédiaire., et ainsi va proposer la commande en ligne et un tas d'information pour eux.

L'avantage de l'application est aussi d'envoyer des notifications dites push. Cela veut dire que le client recevra une notification directement sur son mobile. Contrairement à un mail le taux de vue est quasiment de 100% car le mobinaute ne peut pas rater le message. Vous pouvez donc diffuser des messages de manières directes à vos clients. Cela permet entre autres de les informer sur vos nouveautés ou promotions.

1.7 Conclusion :

Ce chapitre a présenté les technologies mobiles et les applications mobiles et ces caractéristiques et plus spécifiquement le système d'exploitation Android en étalant ces fonctionnalités, son architecture ainsi que les principaux composants du système.

Dans le chapitre suivant, on va entamer la conception de notre projet (application mobile).

Chapitre II :

Conception UML

2.1 Introduction

L'amélioration de la performance des restaurants rapides est particulièrement importante aujourd'hui, car nous constatons que les gens s'y intéressent beaucoup, leur seule préoccupation étant de satisfaire leur client.

Un restaurant est un établissement commercial où l'on sert des repas contre paiement qui offre des conditions de confort plus ou moins importante pour répondre aux besoins des clients.

Dans ce chapitre nous présentons notre application en commençant par la description de notre objectif suivie d'une description de l'architecture technique et les différentes fonctionnalités de notre application en exploitant l'UML comme outil de base. Nous proposons pour cela, d'analyser les différents cas d'utilisations afin de déterminer les différentes classes intervenant et de repérer le séquençement des flux pour chaque scénario de réalisation.

2.2 Objectif

Chaque commerçant cherche à répondre aux besoins des clients pour les satisfaire, dans le domaine de la restauration on trouve le problème de la surcharge des commandes au niveau des restaurants et la longue durée d'attente pour une commande, qui gêne la vie quotidienne des clients.

Notre travail englobe la réduction de cette durée, et la meilleure gestion de ces commandes, donc on a réalisé une application mobile pour faire des commandes par internet pour mieux les gérer.

Donc on va entamer notre application dans la ville de Tlemcen, en utilisant dans notre prototype cinq (05) restaurants : (L'échappatoire, Maxi, Chick'n, Le gourmet, Cyno), ces restaurant nous ont donné les informations nécessaires pour réaliser ce projet. Ainsi, ce sont parmi les restaurants qui ont un grand nombre de commande quotidiennement, surtout les heures de pointe, la gestion de ces commandes est une priorité majeure.

2.3 Fonctionnement de notre application :

Fonctionnement de notre application : Notre application est reliée à une base de données, dans la première interface de l'application, il y aura seulement un bouton qui nous mèneront vers une interface secondaire pour connecter ou inscrire si l'utilisateur n'a pas un compte, après un clic sur le bouton connexion l'application va comparer les informations avec cette base données et l'envoi vers une nouvelle interface qui contient une liste des restaurants, chaque restaurant

possède un menu, l'utilisateur peut créer une commande, qui va s'enregistrer à notre base de données pour que l'employer puisse la récupérer.

2.4 Le langage de modélisation

Donc pour modéliser la structure et le comportement de notre application de façon très claire et précise, nous adaptons le langage de modélisation UML (Unified Modeling Language).

2.4.1 UML :

UML, c'est l'acronyme anglais pour « Unified Modeling Language ». On le traduit par « Langage de modélisation unifié ». La notation UML est un **langage visuel** constitué d'un ensemble de schémas, appelés des **diagrammes**, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour **représenter** le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc.

Réaliser ces diagrammes revient donc à **modéliser les besoins** du logiciel à développer.

UML (la version 2.0) définit 13 (treize) diagrammes mais ceux utilisés dans l'ensemble de notre analyse sont présentés dans le tableau suivant :

Diagramme	Rôles
Cas d'utilisation	Représente les cas d'utilisation du système, les acteurs et les relations existant entre eux.
Classes	Exprime de manière générale la structure statique d'un système, en termes de classes et de relations entre ces classes.
Séquence	Montre les exemples d'historiques de communication entre les objets ou les utilisateurs.
Activité	Une forme spéciale de diagramme de transition d'états utilisés pour modéliser l'état de contrôle.

Tableau 2.1 : les diagrammes utilisés

2.4.2 Diagrammes utilisés

Nous présentons ci-dessous les 3 diagrammes UML, que nous avons mobilisé afin de construire un modèle dynamique qui facilite le développement de notre application :

- Le diagramme de cas d'utilisations.
- Le diagramme de séquence.
- Le diagramme d'activité.
- Le diagramme de classe.

2.4.2.1 Le diagramme de cas d'utilisation

Un diagramme de cas d'utilisation représente l'ensemble des cas d'utilisation qui caractérisent le comportement d'un système. Il décrit la structure des fonctionnalités nécessaires aux utilisateurs du système. L'ensemble des cas d'utilisations vise à décrire exhaustivement les exigences fonctionnelles du système [17].



Un cas d'utilisation est une manière spécifique d'utiliser un système. Les acteurs sont à l'extérieur du système ; ils modélisent tout ce qui interagit avec lui. Un cas d'utilisation réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie [18].

❖ Concepts de diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation se compose de trois éléments principaux



- **Un Acteur** : C'est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Il se représente par un petit bonhomme avec son nom inscrit dessous [19].
- **Un cas d'utilisation** : C'est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service. Il représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.
- **Les dépendances stéréotypées et la généralisation/spécialisation**. Les dépendances stéréotypées sont des dépendances dont la portée est explicitée par le nom du stéréotype. Les stéréotypes les plus utilisés sont l'inclusion et l'extension.
- **La relation d'inclusion** : Pour clarifier un diagramme UML permet d'établir des relations entre les cas d'utilisation. Il existe principalement deux types de relations :
 - **La relation d'extension** : Si le comportement de B peut être étendu par le comportement d'A, on dit alors que A étend B. Une extension est souvent soumise à condition. Graphiquement, la condition est exprimée sous la forme d'une note.
 - **La relation de généralisation ou d'inclusion** : Un cas A est une généralisation d'un cas B si B est un cas particulier de A, la consultation d'un compte bancaire via Internet est un cas particulier de la consultation. Cette relation de généralisation/spécialisation est présente dans la plupart des diagrammes UML et se traduit par le concept d'héritage dans les langages orientés objet.

2.4.2.2 Le diagramme De séquence

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Il est organisé en fonction du temps qui s'écoule au fur à mesure que nous parcourons la page. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction de moment où ils prennent part dans la séquence.

❖ Composition d'un diagramme de séquence :

Le diagramme de séquence contient les éléments suivants :

- **Les lignes de vie** : une ligne verticale qui représente la séquence des évènements, produite par un participant, pendant une interaction, alors que le temps progresse en bas de ligne. Ce participant peut être une instance d'une classe, un composant ou un acteur.
- **Les messages** : deux types de messages dans le diagramme de séquence, le premier et dit message synchrone utilisé pour représenter la communication entre des threads distincts ou la création d'un nouveau thread.
- **Les occurrences d'exécution** : représente la période d'exécution d'une opération.
- **Les commentaires** : un commentaire peut être joint à tout point sur une ligne de vie.
- **Les itérations** : représente un message de réponse suite à une question de vérification. [20]

2.4.2.3 Diagramme d'activité

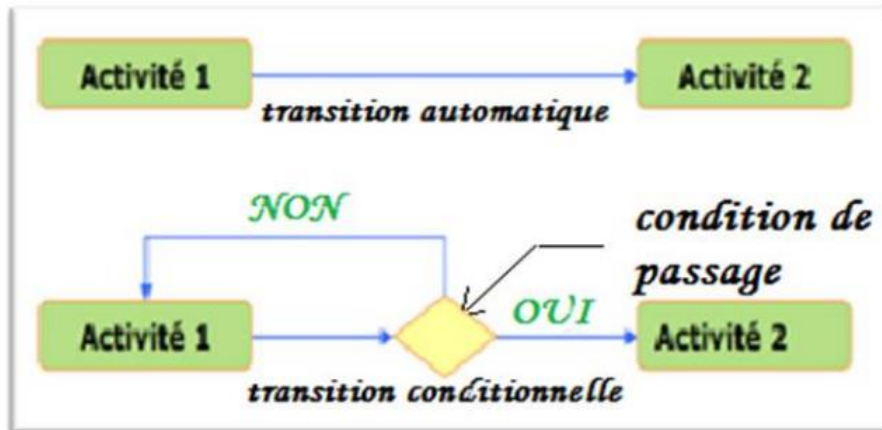
Le diagramme d'activité n'est autre que la transcription dans UML de la représentation du processus telle qu'elle a été élaborée lors du travail qui a préparé la modélisation : il montre l'enchaînement des activités qui concourent au processus. Autrement dit c'est un Diagramme associé à un objet particulier ou à un ensemble d'objets, qui illustre les flux entre les activités et les actions. Il permet de représenter graphiquement le déroulement d'un cas d'utilisation.

❖ Composition d'un diagramme d'activités :

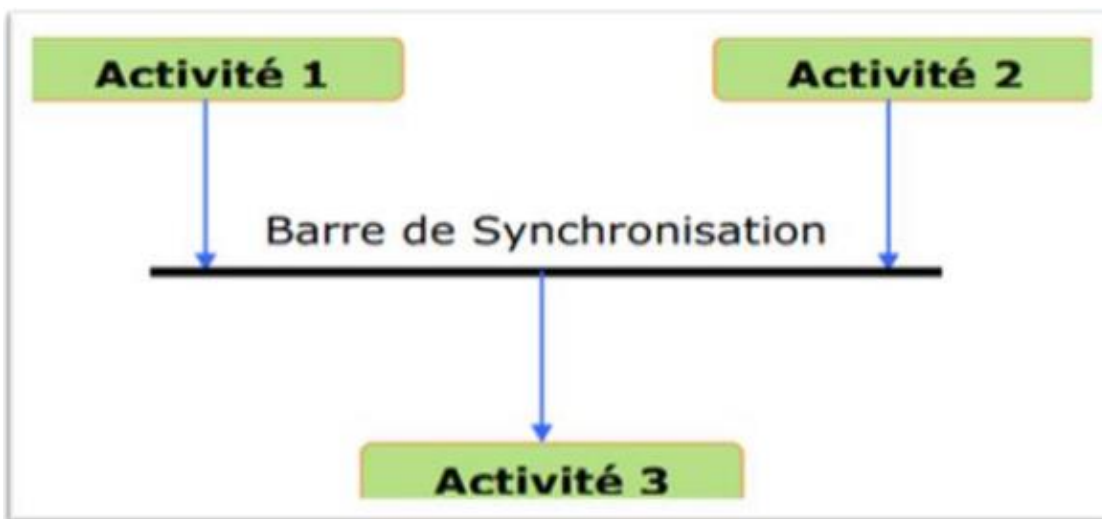
Ce diagramme comporte les éléments suivants :



- **Une activité** : représente une exécution d'un mécanisme, autrement dit, un déroulement d'étapes séquentielles.



- **Une transition** : elle est représenté le passage d'une activité vers une autre. Cette transition peut être automatique, qui se déclenche par la fin d'une activité, provoquent le début immédiat d'une autre ou conditionnelle, qui ne se déclenche qu'après la satisfaction de la condition qu'on appelle aussi garde.
- **Les gardes** : qui représentent la condition de passage d'une activité à une autre dans les transitions conditionnelles ils sont symbolisés par des losanges comme dans la figure ci-dessus.



- **Les barres de synchronisation** : sont des barres représentées par une ligne épaisse, le rôle de cette barre est de synchroniser le départ de plusieurs transitions qui arrivent de différentes activités, aboutissant toutes à une activité commune.

2.4.2.4 Diagramme de classes

Le diagramme de classes exprime la structure statique du système en montrant les objets dans le système, les relations entre les objets, les attributs et les opérations qui caractérisent chaque classe d'objet. Il intervient au niveau de la branche fonctionnelle de notre processus.

❖ Identification des classes :

Le diagramme de classe comporte les éléments suivants : Une classe, un ensemble commun de propriétés (les attributs), de comportements (les opérations) et de relations avec d'autres objets (les associations et les agrégations).

- **Les classes** : c'est une description formelle d'un ensemble d'objets ayant une sémantique et des caractéristiques communes. Elle est représentée par un rectangle divisé en trois sections. La section supérieure est le nom de la classe, la section centrale définit les propriétés de la classe alors que la section du bas énumère les méthodes de la classe.
- **Les associations** : c'est une relation structurelle entre deux classes (association binaire) ou plus (association n'aire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc que des liens peuvent exister entre des instances des classes associées.
- **Les attributs** : les attributs représentent les données encapsulées dans les objets des classes. Chacune de ces informations est définie par un nom, un type de données, une visibilité et peut être initialisé. Le nom de l'attribut doit être unique dans la classe. [22]

2.5 Analyse et conception

Ce diagramme illustre bien le fonctionnement de l'application, il représente le diagramme global des cas d'utilisation.

2.5.1. Identifications des acteurs

Dans le cadre de notre analyse, les acteurs que nous avons pu identifier sont :

- **Le client** : c'est l'utilisateur principal du système. Cette unité extérieure interagit avec le système via une interface graphique. Son travail est de faire introduire dans l'application toutes les informations sur toutes les opérations effectuées durant la réservation du plat.

- **L'employé dans un restaurant** : il contrôle toutes les opérations qui se passent dans l'application et vérifie leur conformité avec les données introduites dans l'application de gestion.

2.5.2. Diagramme de cas d'utilisation globale

Ci-dessous, nous présentons le diagramme de cas d'utilisation pour la compréhension du fonctionnement du système.

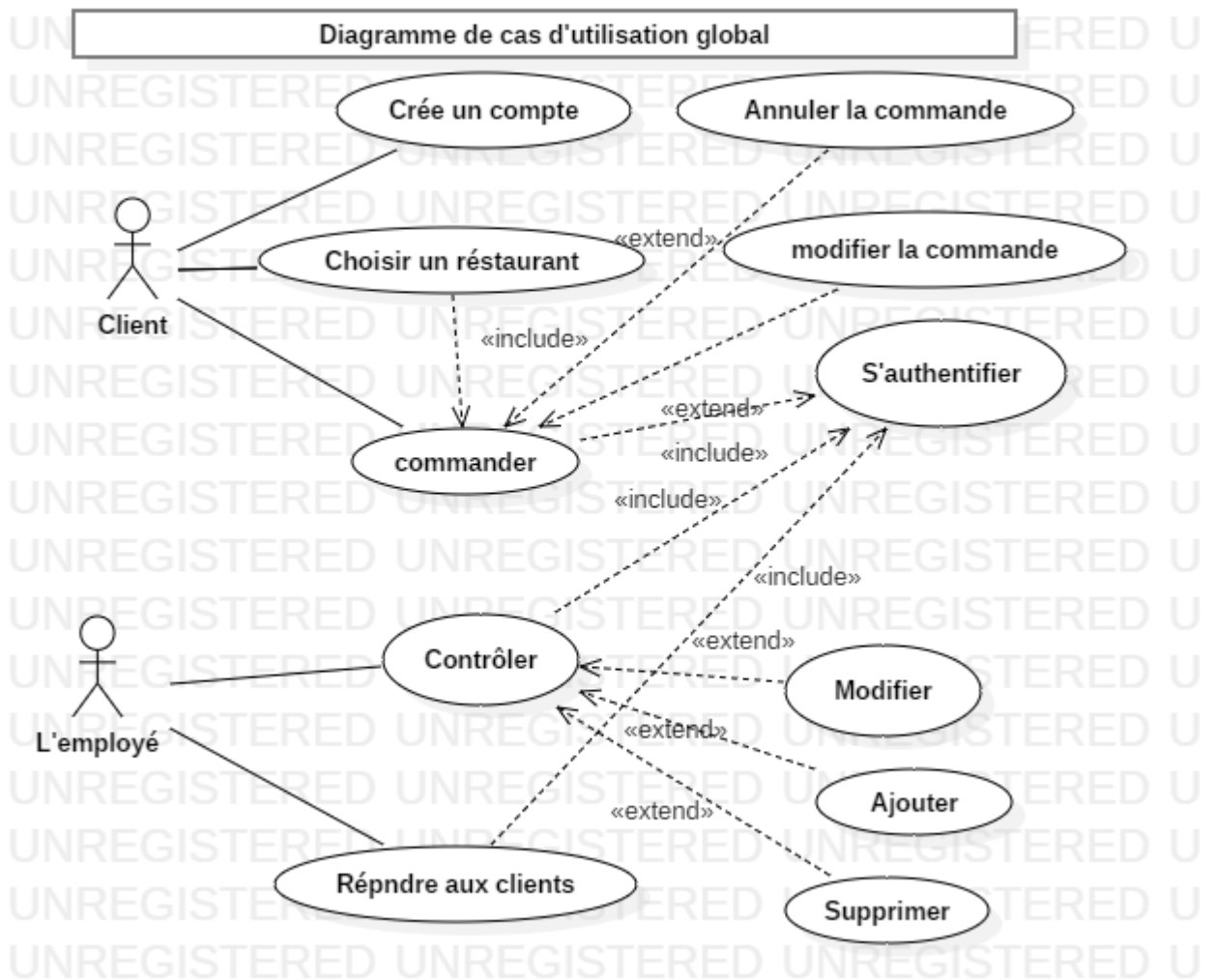


Figure 2.1 : Diagramme des cas d'utilisation global

Il ressort du contexte de l'application les cas d'utilisation suivants :

- **Cas 1** : Créé un compte (inscription)
- **Cas 2** : Authentification
- **Cas 3** : Choisir un restaurant

- Cas 4 : Commander
- Cas 5 : gérer le menu
- Cas 6 : Répondre aux clients

2.5.3 Diagramme de séquence du cas d'utilisation 1 : 'Inscription'

Un client ou utilisateur doit d'abords entamer la phase d'inscription avec succès et pour cela il faut qu'il passer par l'ensemble des séquences que nous allons simplifier par le schéma suivant :

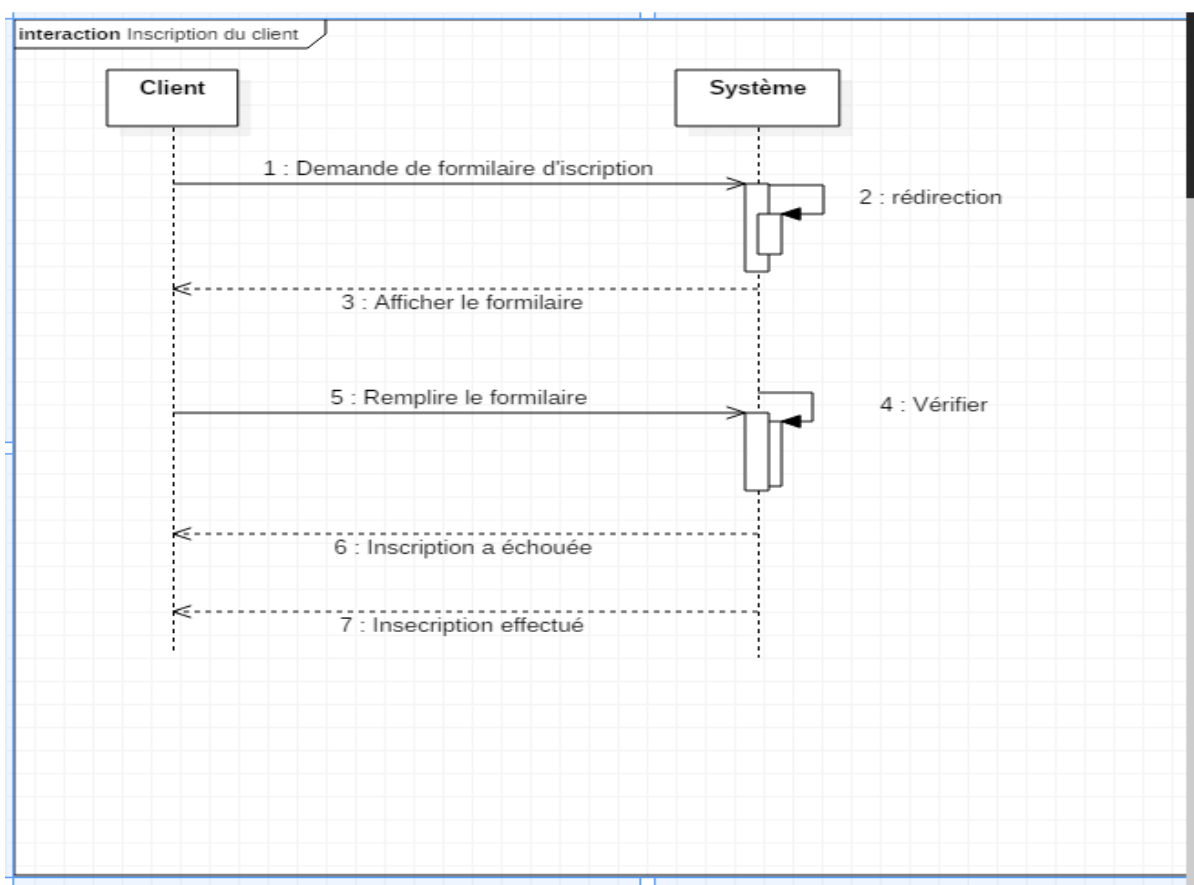


Figure 2.2 : Diagramme de séquence d'inscription

- Le client demande le formulaire d'inscription.
- Le formulaire s'affichera.
- Le client remplit le formulaire.
- Une vérification de l'existence du client dans la base se lance.
- Si le client existe déjà un message d'erreur s'affiche.

- Si c'est un nouvel client l'inscription est effectué.

2.5.4 Diagramme de séquence du cas d'utilisation 2 : 'Authentification'

Avant d'atteindre la phase d'authentification, notre client est un utilisateur présente sur notre application d'une façon anonyme, d'où il devient indispensable d'entrer son login et son mot de passe pour accéder les services dont il autorisé.

Le schéma suivant va vous représenter les séquences à effectuer pour entamer la phase d'authentification :

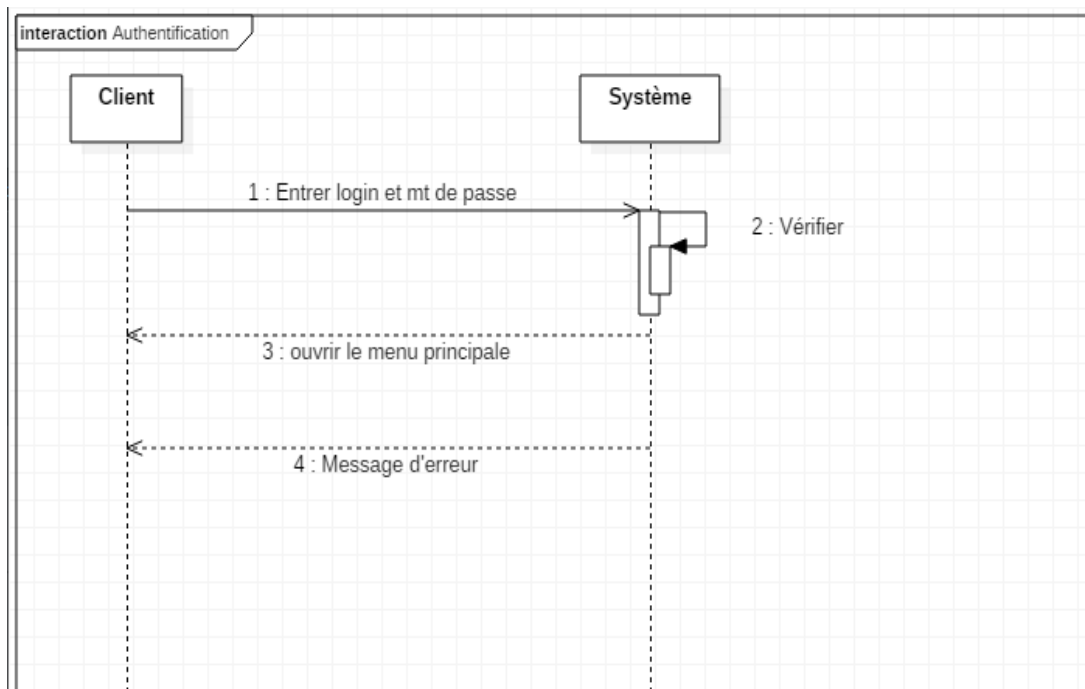
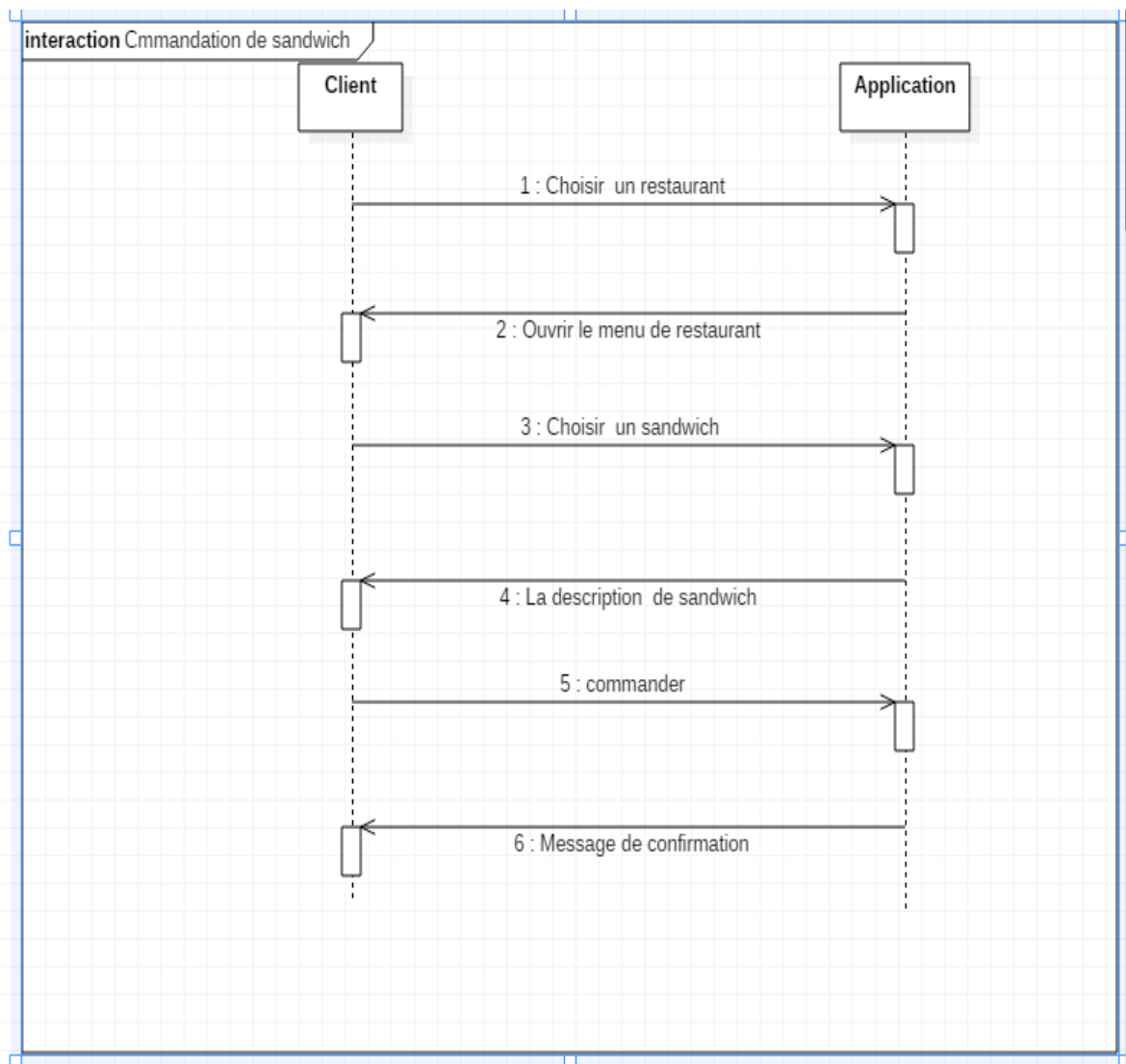


Figure 2.3 : Diagramme de séquence d'authentification

- Le client entre son login et son mot de passe.
- Une vérification sera lancée au niveau de la base de données.
- Après un temps de réponse, l'authentification sera validée ou bien un message d'erreur sera affiché.

2.5.5 Diagramme de séquence du cas d'utilisation 3 : 'Choisir et commander un restaurant'**Figure 2.4 : Diagramme de séquence : commander**

- Après la connexion vers notre menu principale, le client choisit un restaurant.
- Une fois que le client fait le choix, le système lui affichera puis il choisit le sandwich.
- L'application affiche la description de sandwich comme le prix, les compositions de sandwich, le numéro de sandwich.
- Et après le client clique sur une carte pour commander et attend la réponse de restaurant qui choisit.

2.5.6 Diagramme de séquence du cas d'utilisation 4 : 'Gérer le menu'

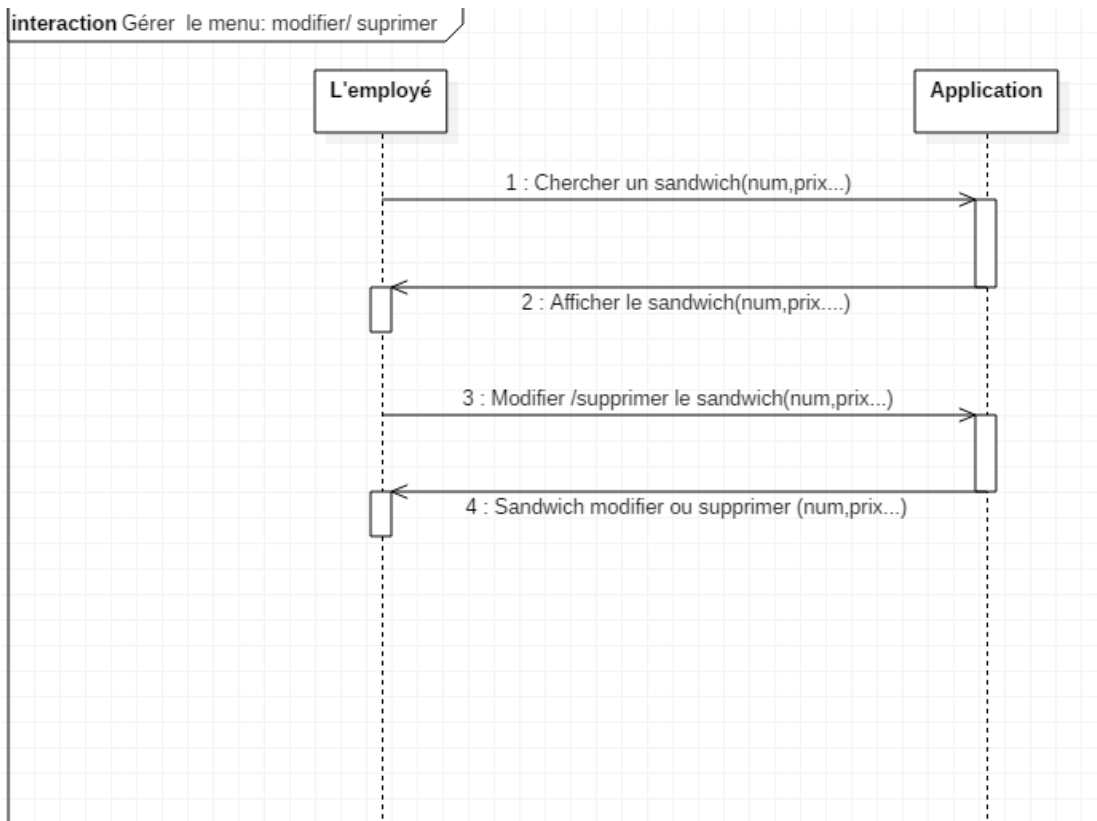


Figure 2.5 : Diagramme de séquence : Modifier/supprimer un sandwich

Ce diagramme de séquence décrit deux cas d'utilisation à la fois : le cas de la modification d'un sandwich et le cas de suppression d'un sandwich. Ces deux scénarios s'effectuent à l'aide de l'interface dédiée à la gestion des sandwiches finis. Pour la modification, l'employé cherche et sélectionne le sandwich qui fait l'objet de cette opération où il peut modifier ces informations (sa description ou son prix dans le menu par exemple). Dans le cas où le restaurant décide que ce sandwich ne fait plus partie de son menu, l'employé le supprime de la base de données.

2.5.7 Diagramme de séquence supprimer/ Ajouter sur un menu de restaurant

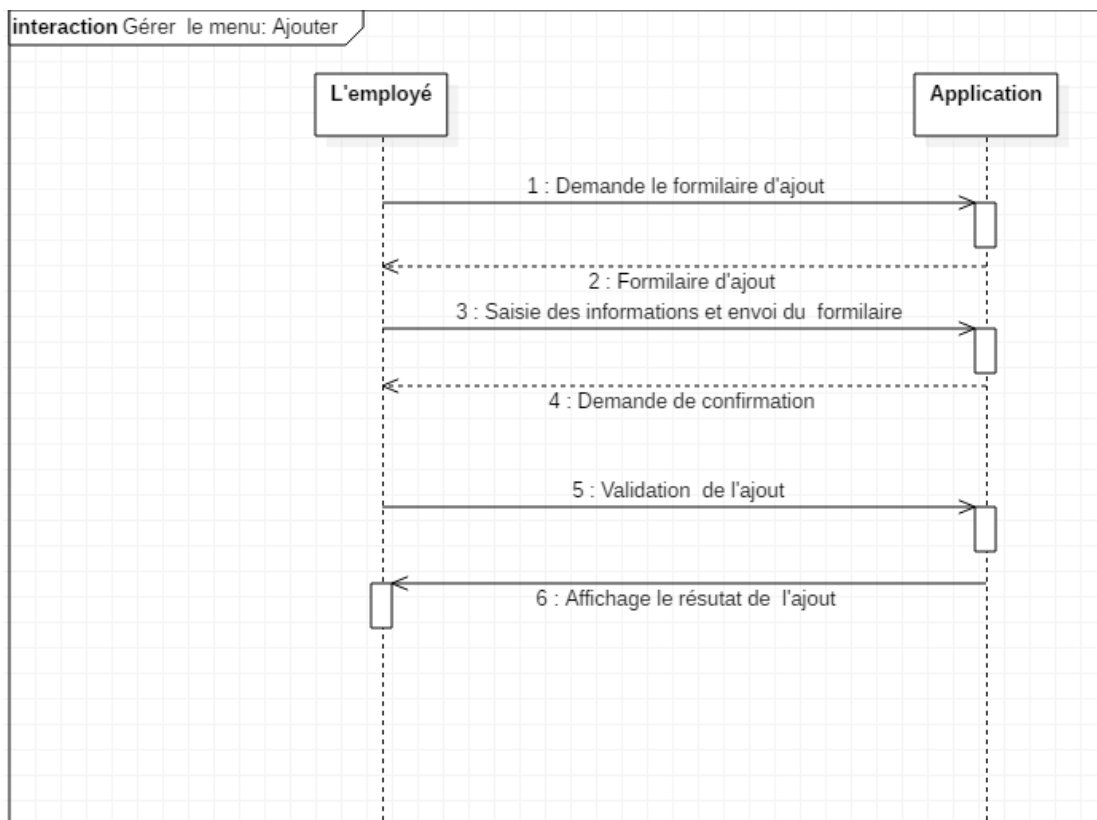


Figure 2.6: Diagramme de séquence : ajouter un sandwich

Ce diagramme représente l'opération effectuée par l'employé sur le système quand un nouveau sandwich. L'employé introduit les informations de ce dernier (nom, numéro, prix, ...).

2.5.8 Diagramme de séquence : 'Répondre aux clients'

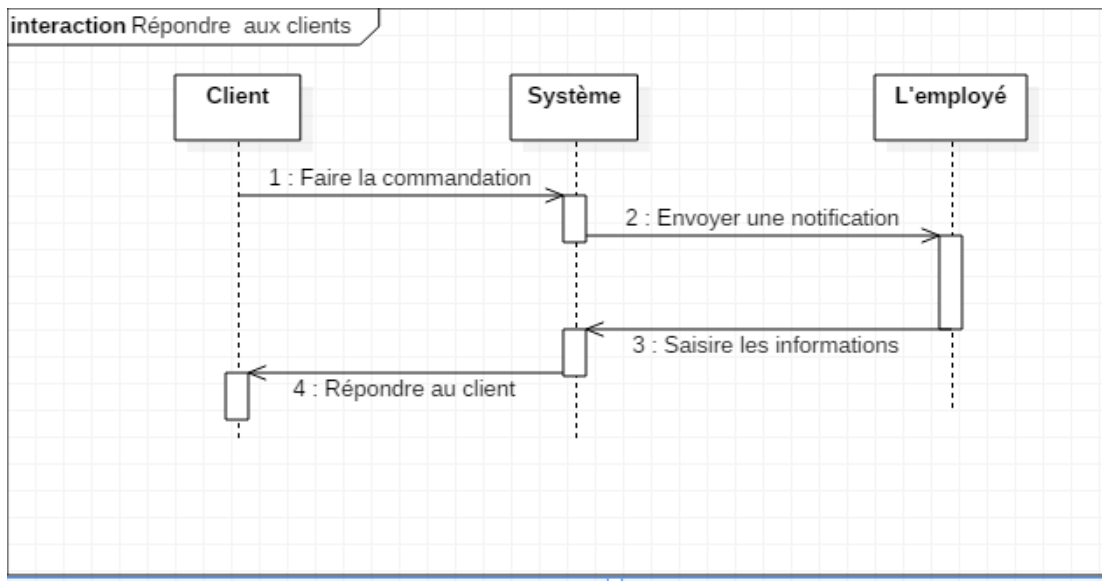


Figure 2.7 : Diagramme de séquence : Répondre aux clients

- Afin de faire la commande par le client, le système envoie les informations à la base de données à l'aide des requêtes SQL.
- Puis l'employé récupère la commande, et la mise en œuvre.

2.5.9 Diagramme d'activité

- ❖ **L'inscription** : pour passer de notre réseau social, un client doit devenir membre et cela ne se fait qu'après l'inscription. L'application fournit au client une interface lui permettant de s'inscrire. Lorsque le client choisit inscription, l'application affiche le formulaire dédié, le client remplit les champs et valide son action, puis l'application sauvegarde les informations fournies par le client dans la base de données.

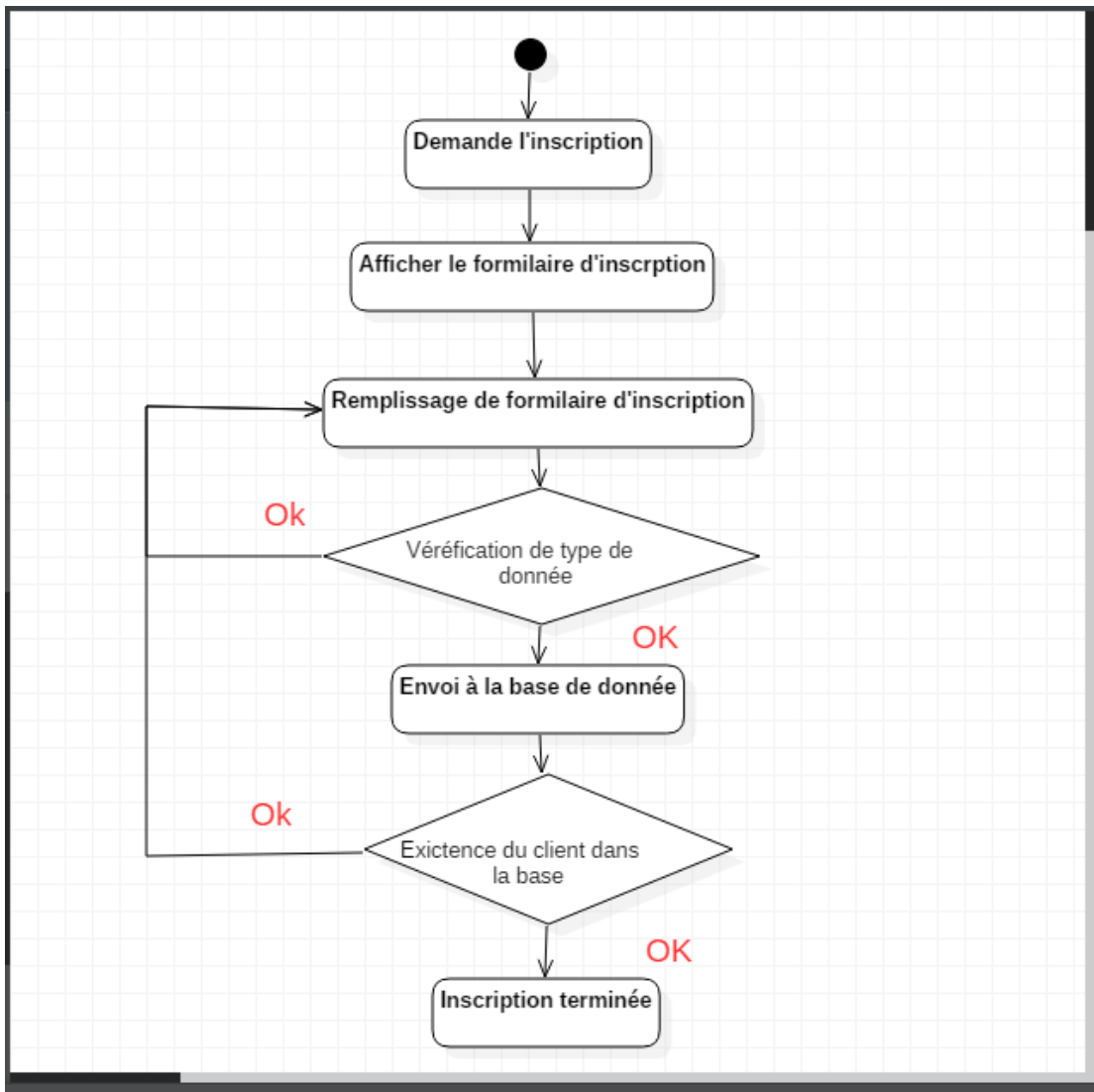


Figure 2.8: Diagramme d'activité inscription

- Le client demande l'inscription.
- Le formulaire d'inscription s'affichera.
- Le client remplit les champs demandés dans le formulaire.
- Le système vérifie les données entrées.
- Si les données sont acceptées, le système les envoyer à la base de données, si non, il revient à l'étape précédente.
- Le serveur vérifie l'existence du client dans la base de données.
- Si le client existe déjà, un message d'erreur sera affiché.
- Si le client n'existe pas, l'inscription terminera avec succès.
- ❖ **L'authentification** : c'est une procédure qui consiste pour un système informatique, à vérifier l'identité d'un client afin d'autoriser son accès aux systèmes. Elle permet donc de

valider l'authentification de l'entité en question. Le client demande l'accès au système (demande le formulaire d'authentification), ce dernier lui répond par l'affichage du formulaire d'authentification comme le client peut saisir son login et son mot de passe, ceci permet au système de vérifier l'existence de ce client, s'il figure dans la base de données donc il est authentifié, comme il aura accès à la page d'accueil (menu d'application). Si non le système renvoie un message d'erreur (pseudo ou mot de passe incorrect).

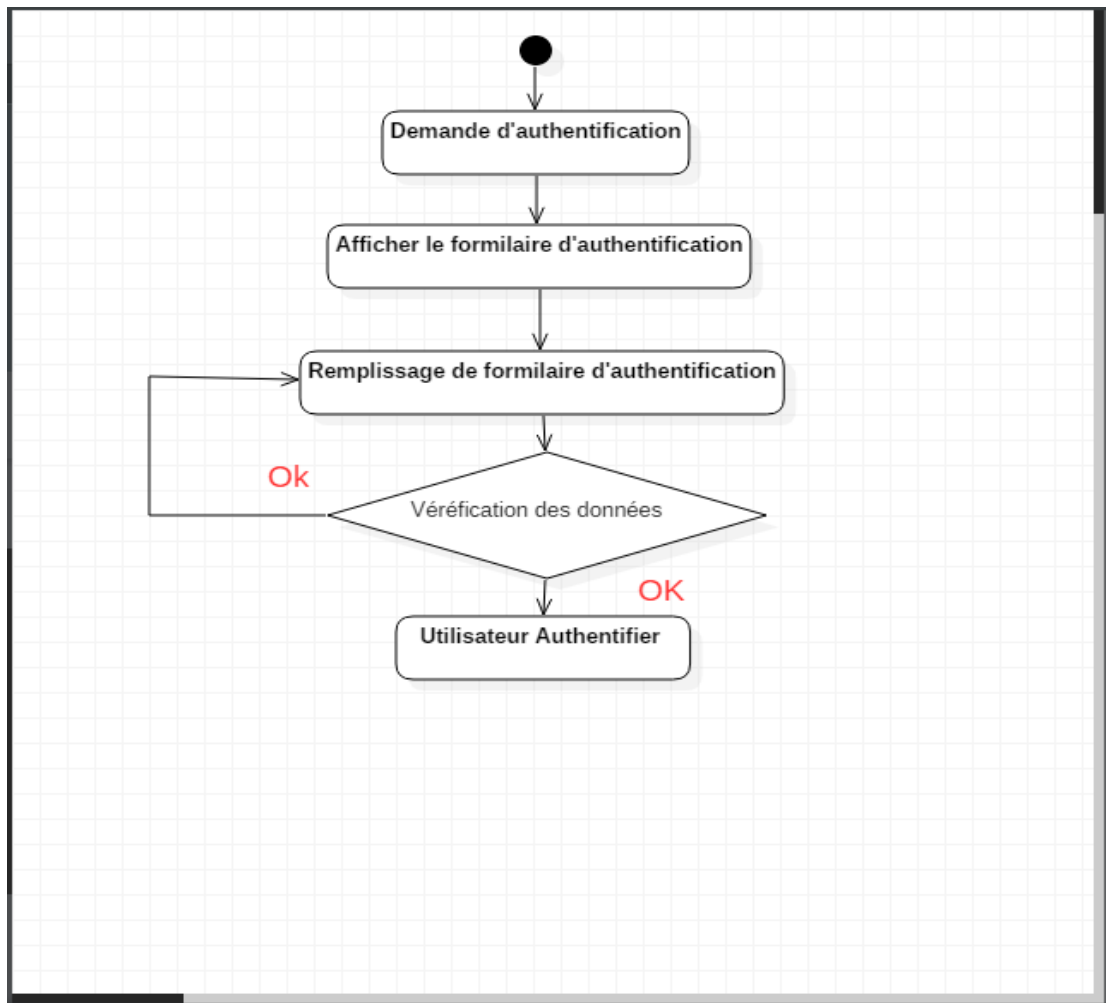


Figure 2.9: Diagramme d'activité authentification

- Une fois que le client demande l'accès à notre réseau social, le système lui affichera la page d'authentification.
- Le client entre son nom d'utilisateur et son mot de passe.
- Le système vérifie les coordonnées sur la base de données.
- La confirmation du succès ou d'échec sera envoyée au client.

2.5.11 Diagramme de classes :

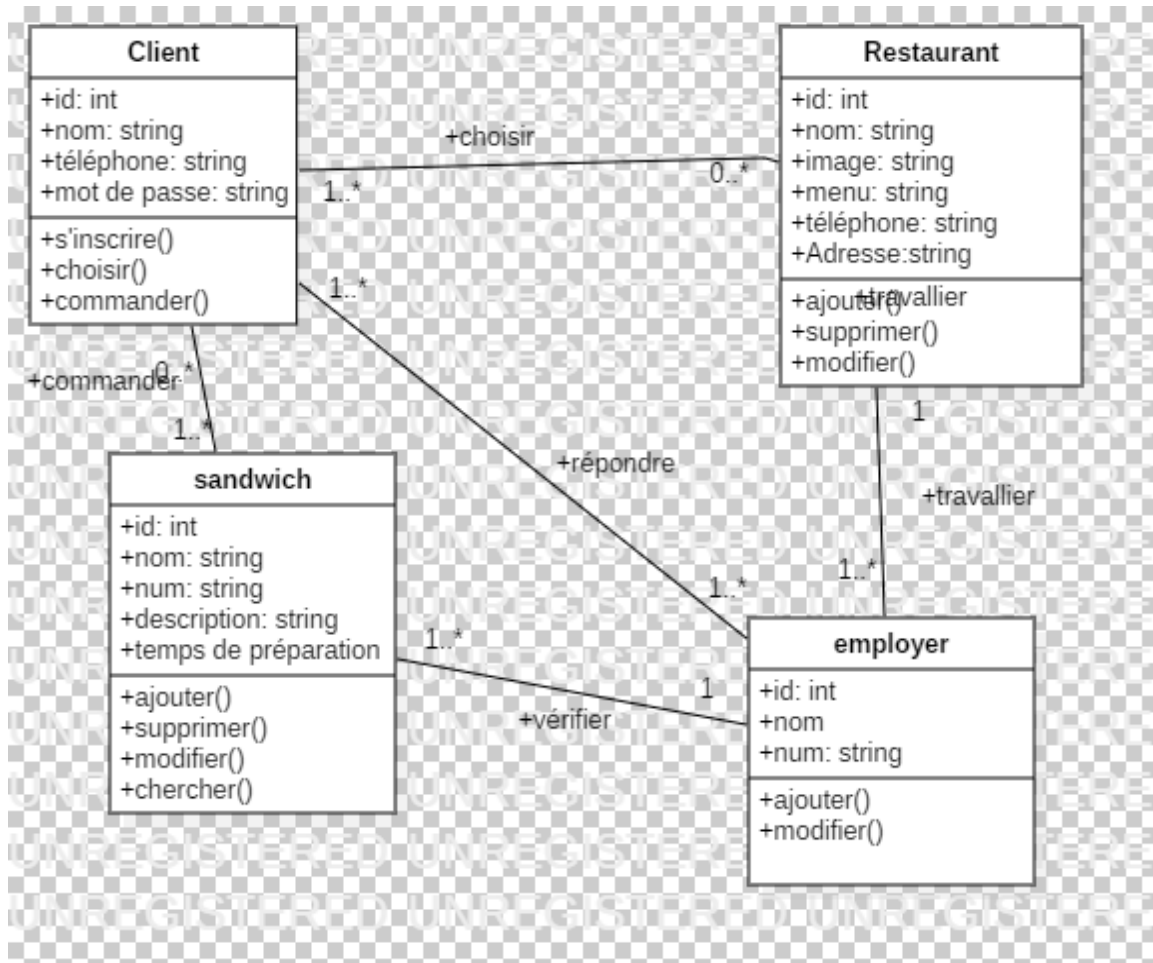


Figure 2.8 : Diagramme de classe

2.6 Conclusion

Tout au long de ce chapitre, nous avons détaillés la conception de notre application à travers le diagramme de cas d'utilisation ainsi que les diagrammes de séquence associées et diagramme de classe afin que la phase réalisation et la mise en place de l'application soit plus souple et plus aisée le chapitre suivant mettra en évidence, le fruit de ce passage et les différents résultats du développement de l'application demandée.

Chapitre III :
Conception et mise en œuvre

3.1 Introduction

Ce chapitre est consacré à la partie pratique de la réalisation de notre application mobile. Il comporte la présentation de l'environnement de développement, une description des outils de développement et langages de programmation utilisés ainsi qu'une présentation de notre application mobile.

3.2 Environnement de développement :

3.2.1 Environnement matériel :

Notre plateforme a été développée sur une machine Intel(R) Core(TM) I33110M CPU (2.4 GHz) RAM 4 Go, sous le système d'exploitation Microsoft Windows 10 professionnel (64bits).

3.2.2 Environnement logiciel :

3.2.2-a Android Studio

Android Studio est un nouvel environnement pour le développement et programmation entièrement intégré qui a été récemment lancé par Google pour les systèmes Android, il a été conçu pour fournir un environnement de développement et une alternative à Eclipse qui est l'IDE le plus utilisé.

Android Studio permet de voir chacun des changements visuels que vous effectuez sur votre application et en temps réel, vous pourrez voir aussi son effet sur différents appareils Android.

Android Studio offre aussi d'autres choses :

- Un environnement de développement robuste.
- Une manière simple pour tester les performances sur d'autres types d'appareils.
- Des assistants et des modèles pour les éléments communs trouvés sur tous les programmeurs Android.

- Un éditeur complet avec une panoplie d'outils pour accélérer le développement de votre application. [23]

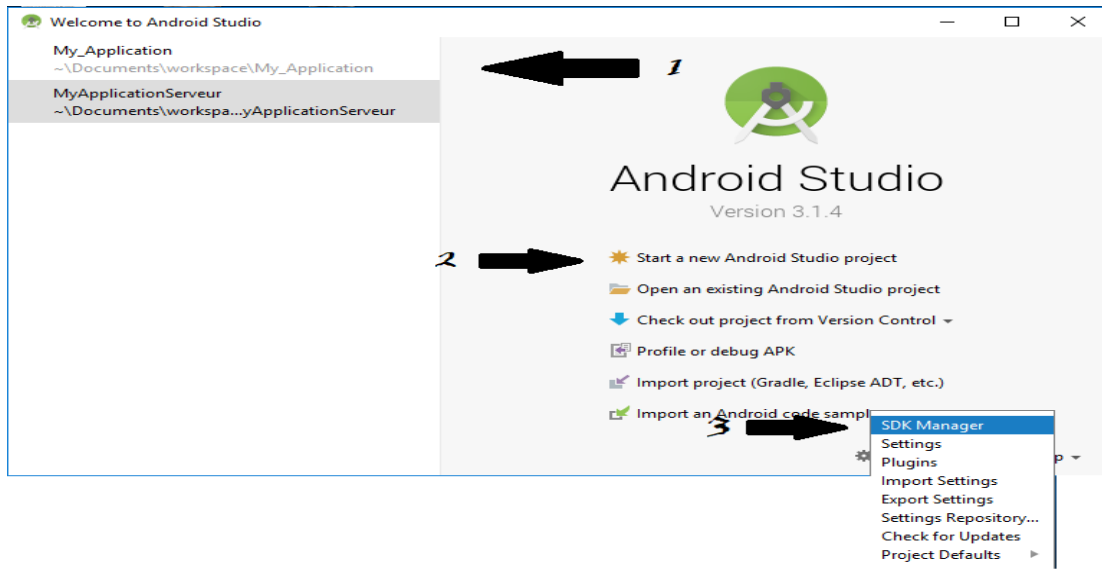


Figure 3.1 : Android Studio.

Après l'ouverture de l'Android Studio une fenêtre s'affiche avec des multiples tâches par exemple :

- 1 : Les projets récents.
- 2 : Crée un nouveau projet.
- 3 : SDK Manager (installer les outils nécessaires pour le projet : la Platform de la version android utilisé, l'intégration du service Google Play, etc.).

Les dossiers les plus utilisés sont **java** et **res**. Le premier contient le code Java qui définit le comportement de l'application (situé dans le répertoire de votre projet sous **app\src\main**) et la seconde comporte des sous dossiers (dans **app\src\main\res**) où sont stockées les ressources qui définissent l'interface de l'application (l'apparence).



Figure 3.2 :Structure d'un projet Android.

3.2.2-b SDK Manager :

SDK signifie Software Development Kit, c'est un ensemble d'outils d'aide à la programmation pour concevoir des logiciels, jeux, applications mobiles, etc. pour un terminal et/ou un système d'exploitation spécifique.

Depuis l'arrivée de Android, le système d'exploitation que Google a développé pour se lancer dans le secteur mobile, un kit de développement est désormais disponible.

Le kit de développement Android inclut les utilitaires nécessaires pour aider les développeurs Android avec les premières étapes : les différents API développés par Google à la fois pour contrôler les fonctions de l'appareil et l'intégration des services, un émulateur complet pour tester des applications, et tout le matériel de lecture nécessaire pour vous aider à effectuer les premières étapes de la programmation pour Android.

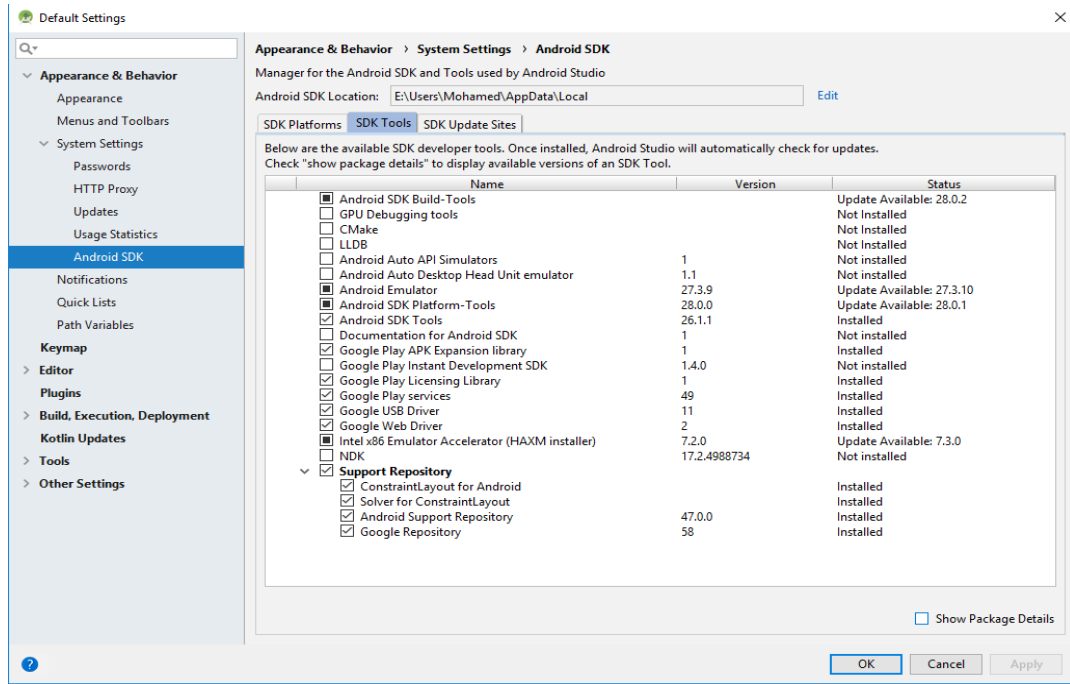


Figure 3.3: SDK Manager.

3.2.2-c Firebase:

Firebase est un ensemble des services d'hébergements pour tous types d'applications (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel. Les applications qui l'utilisent intègrent une bibliothèque qui permet les diverses interactions possibles. Il permet de développer rapidement des applications de grande qualité, d'accroître le nombre d'utilisateurs et d'augmenter vos revenus. [24]

Il a été lancé en 2011 sous le nom d'Envolv, par Andrew Lee et par James Templin, le service est racheté par Google en octobre 2014. Il appartient aujourd'hui à la maison mère de Google : Alphabet.

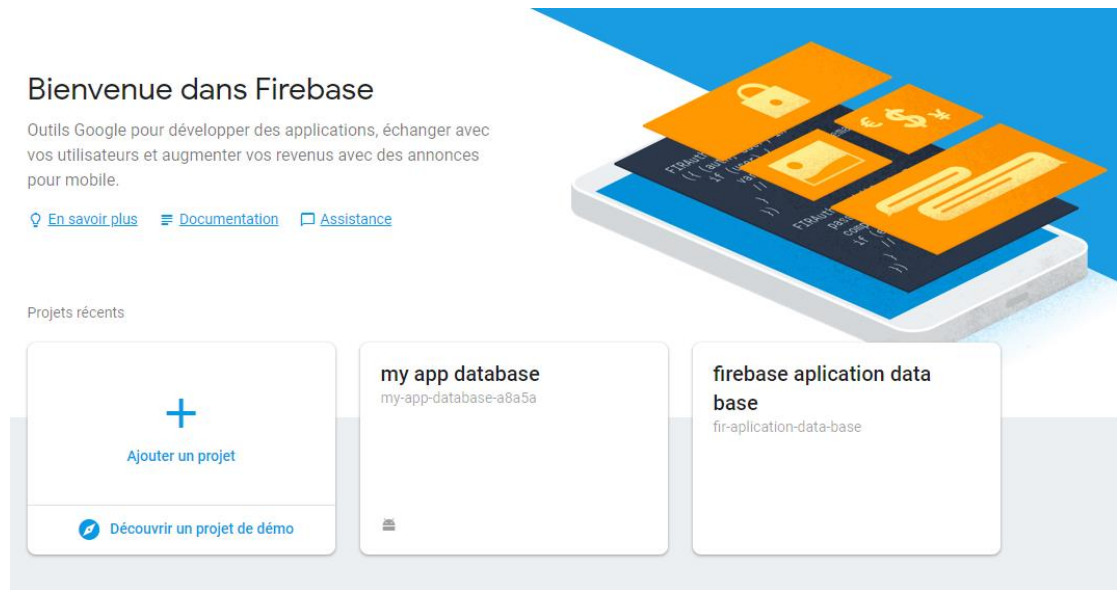


Figure 3.4 : Firebase Console.

3.2.2-c SQLite :

SQLite est un système de base de données ou une bibliothèque proposant un moteur de bases de données relationnelles. Il repose sur une écriture en C, un langage de programmation impératif, et sur une accessibilité via le langage SQL (StructuredQueryLanguage), créé au début des années 2000 par D. Richard Hipp. [25]

Il présente la particularité d'être directement intégré aux programmes et dans l'application utilisant sa bibliothèque logicielle alors que ses concurrents comme MySQL reproduisent de leur côté le schéma classique client-serveur. Avec SQLite, la base de données est intégralement stockée dans un fichier indépendant du logiciel, il propose un accès plus rapide aux données, plus structuré et avec davantage de sécurité.

3.2.2-d GENYMOTION :

Genymotion est un émulateur Android basé sur VirtualBox. Genymotion a l'avantage d'être plus rapide que l'émulateur du Kit de développement Android et fournit de meilleures performances générales. Cet outil requiert une configuration supplémentaire. [26]



Figure 3.5 : GENYMOTION.

3.3 L'architecture de l'application :

3.3.1 Type d'architecture :

L'architecture idéale pour notre système est l'architecture client /serveur-3 tiers.

Cette architecture vise à séparer très nettement trois couches logicielles au sein d'une même application ou système dont le rôle est clairement défini [27].

Le système est composé des niveaux comme nous avons cité dans le premier chapitre figure (3.6).

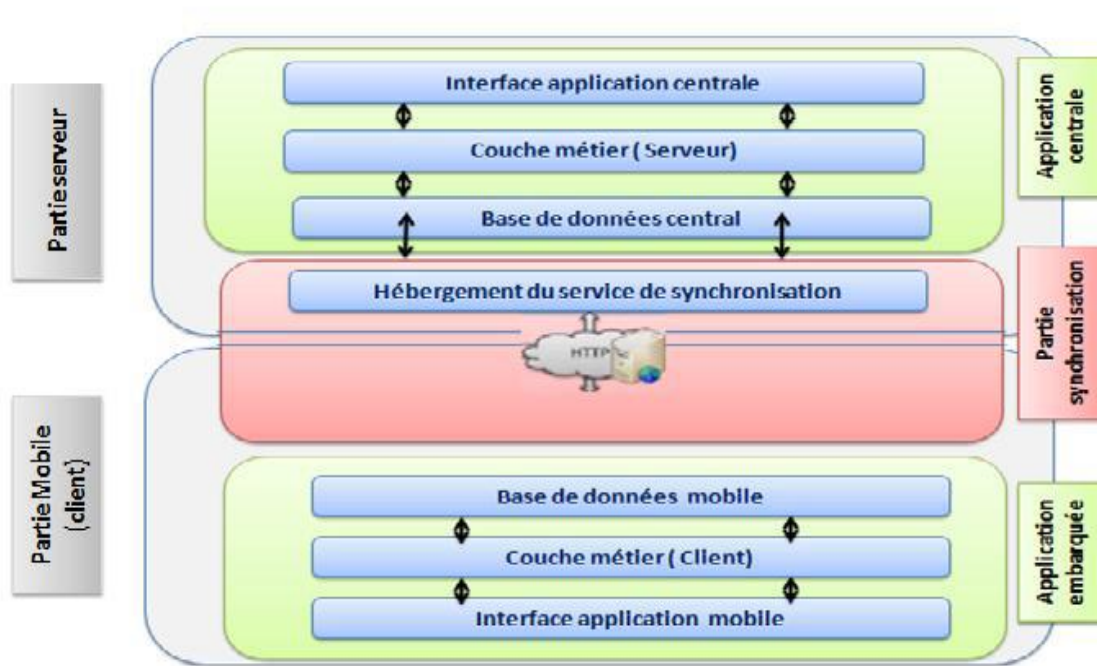


Figure 3.6 :Architecture de la solution globale.

Notre solution sera empaquetée dans les 03 parties suivantes :

1. Partie Serveur :

Elle englobe le système central qui correspond à la partie fonctionnelle du système, nous distinguons principalement la base de données centrale (Firebase) qui communique avec la couche métier où s'effectue les différents traitements (administration, transaction, les commandes des clients, menu), et qui elle-même communique avec l'interface utilisateur qui visualise les différentes fenêtres et formulaires de la solution.

2. Partie Mobile :

Elle regroupe l'application embarquée qui est composée d'une base de données locale SQLite, cette partie communique avec la couche métier qui est chargée de plusieurs tâches (effectuer des commandes, inscription, récupérer les commandes coté vendeur...).

3. Partie Synchronisation de données :

La synchronisation de données consiste à obtenir périodiquement des informations stockées dans une base de données serveur (Firebase) et de synchroniser les modifications avec une base de données client telle que (SQLite).

3.4 Les interfaces graphiques :

L'interface graphique est une partie très importante pour une application mobile. Ainsi elle peut faire la différence entre deux applications même si elles ont les mêmes fonctionnalités.

Voici maintenant un ensemble de captures d'écrans sur les principaux points d'entrées de l'application :

3.4.1 Client-Side :

1. Interface d'authentification :

C'est la première interface qui apparaît une fois l'application est lancée, l'utilisateur va choisir entre connecter avec son compte ou créer un nouveau s'il n'en a pas un, cette interface contient deux boutons (inscription ou connexion).

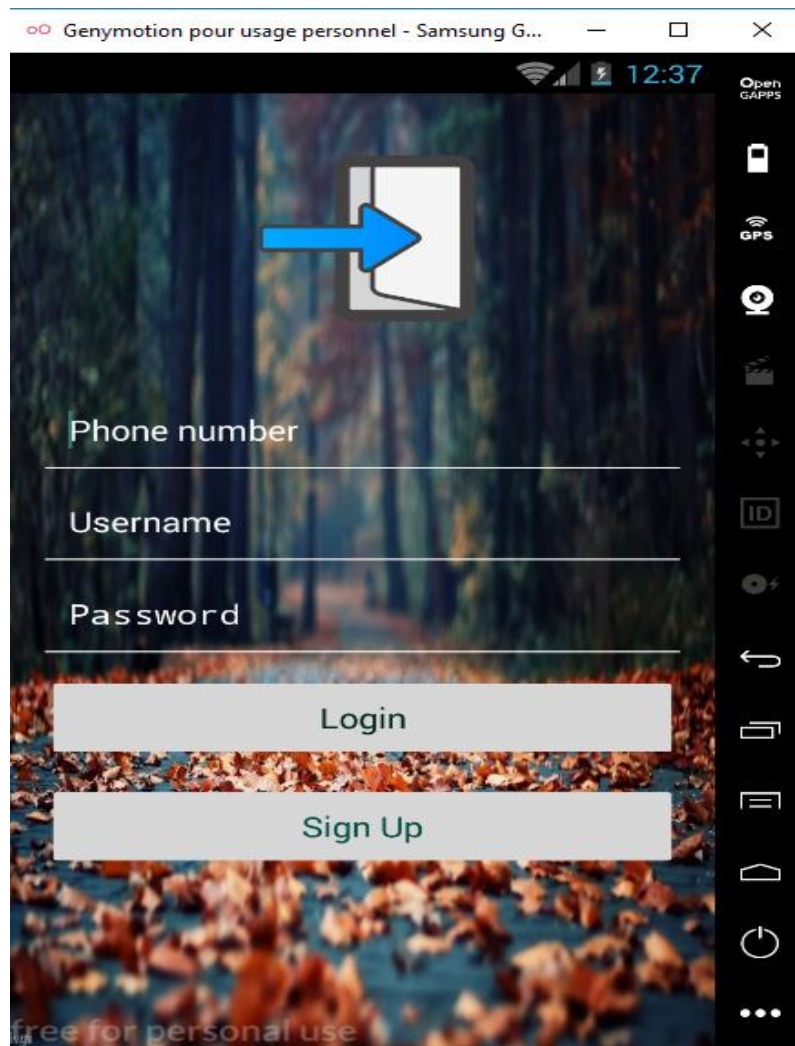


Figure 3.7 :Connexion ou inscription.

- Premier Choix (inscription):

Si l'utilisateur clique sur le bouton (SIGN UP), l'interface de l'inscription va apparaître.



Figure 3.8 :Création d'un nouveau compte.

Pour s'inscrire avec succès l'utilisateur doit remplir les champs de texte ci-dessus et en cliquant sur le bouton SIGNUP pour confirmer, le système va envoyer ses coordonnées à notre firebase.

- Deuxième Choix (Connexion) :

Si cet utilisateur clique sur le bouton (LOGIN), un algorithme va comparer ses informations avec celles de la base de données.

-En cas d'échec : un message d'erreur va s'afficher.

-Si les données sont validées : une nouvelle interface s'affiche.

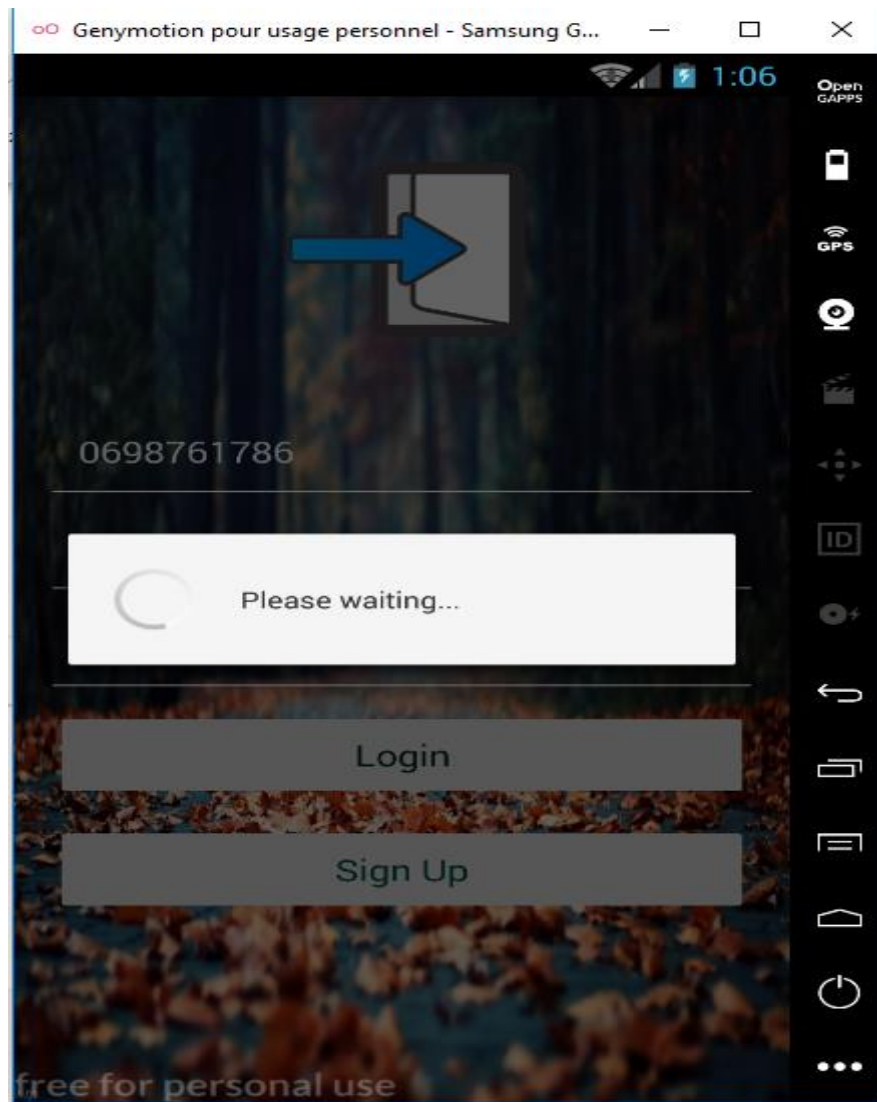


Figure 3.9 : Vérification des données.

2. Interface Home :

Cette interface contient une liste des restaurants de Tlemcen, l'utilisateur va choisir l'un des d'eux, pour choisir un sandwich du restaurant choisi.

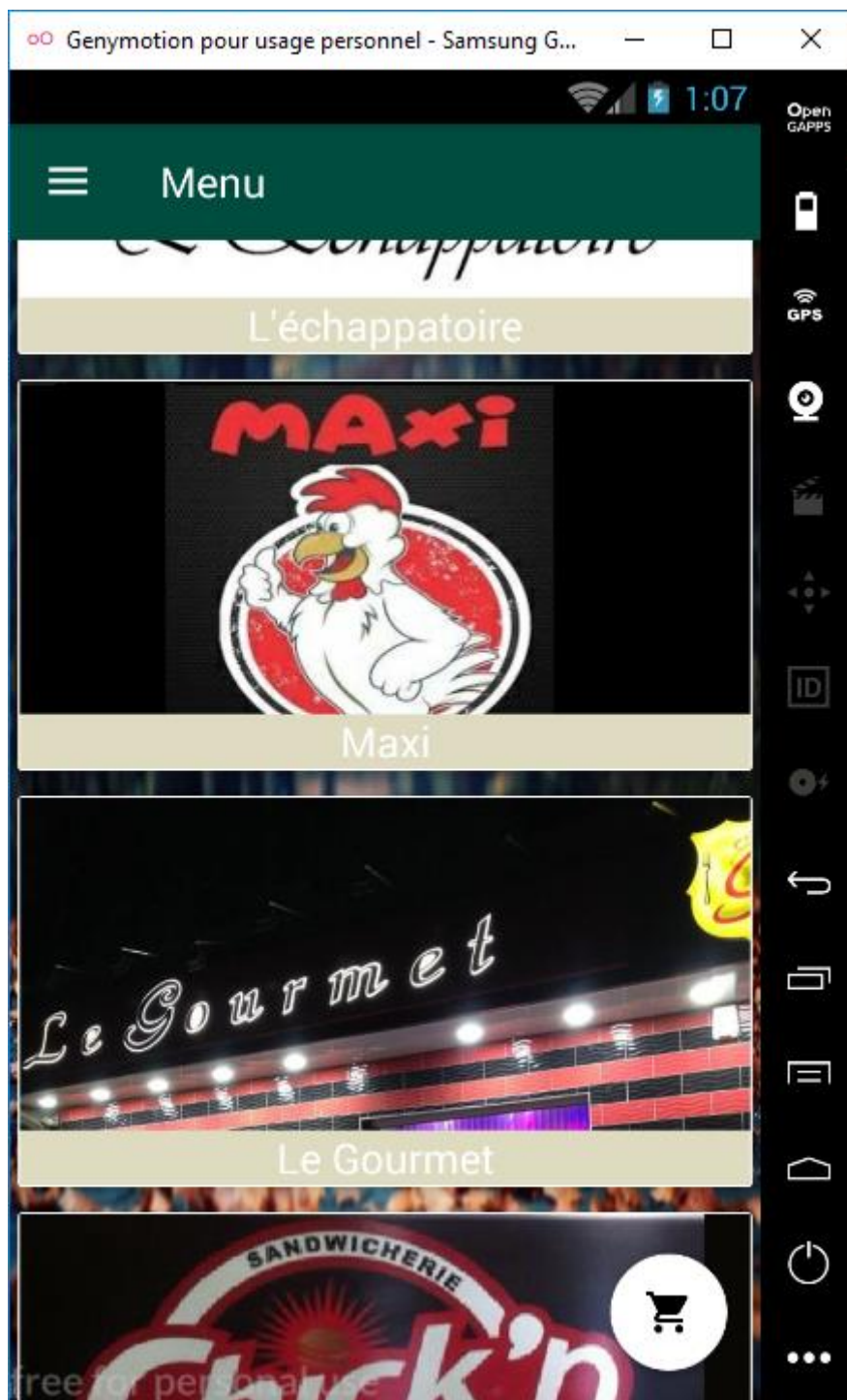


Figure 3.10 :Liste des Restaurants.

3. Interface menu du restaurant :

Cette interface contient la liste des sandwichs du restaurant choisi avec une barre de recherche, l'utilisateur va cliquer sur un sandwich voulu pour voir ces information (prix, description.....).

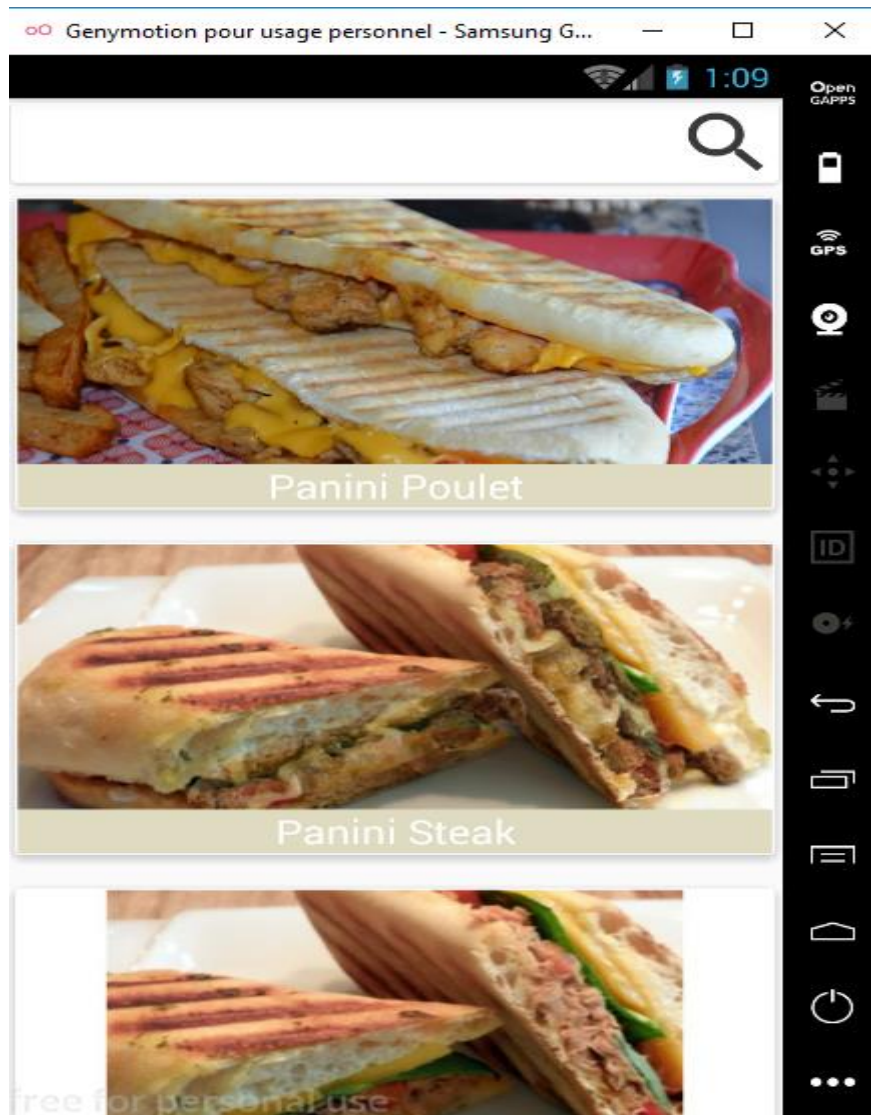


Figure 3.11 :Liste des Sandwichs.

4. Interface Détails d'un Sandwich :

Cette interface montre les différents détails du sandwich choisi et avec la possibilité de commander ce sandwich avec la quantité voulu.

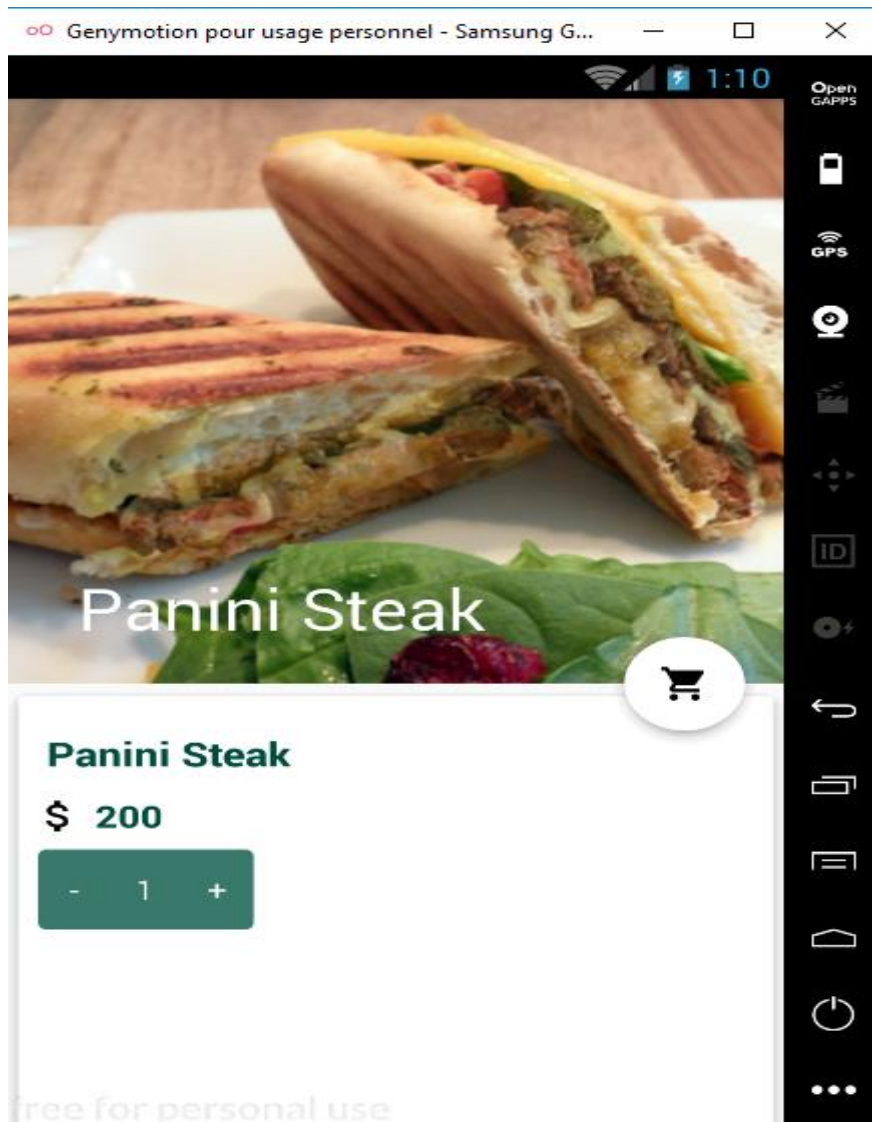


Figure 3.12 :Choisir un Sandwich.

Après être choisi les sandwichs l'utilisateur doit revenir sur l'interface home, et en cliquant sur menu une liste va s'afficher pour qu'il puisse confirmer sa commande ou la supprimer.

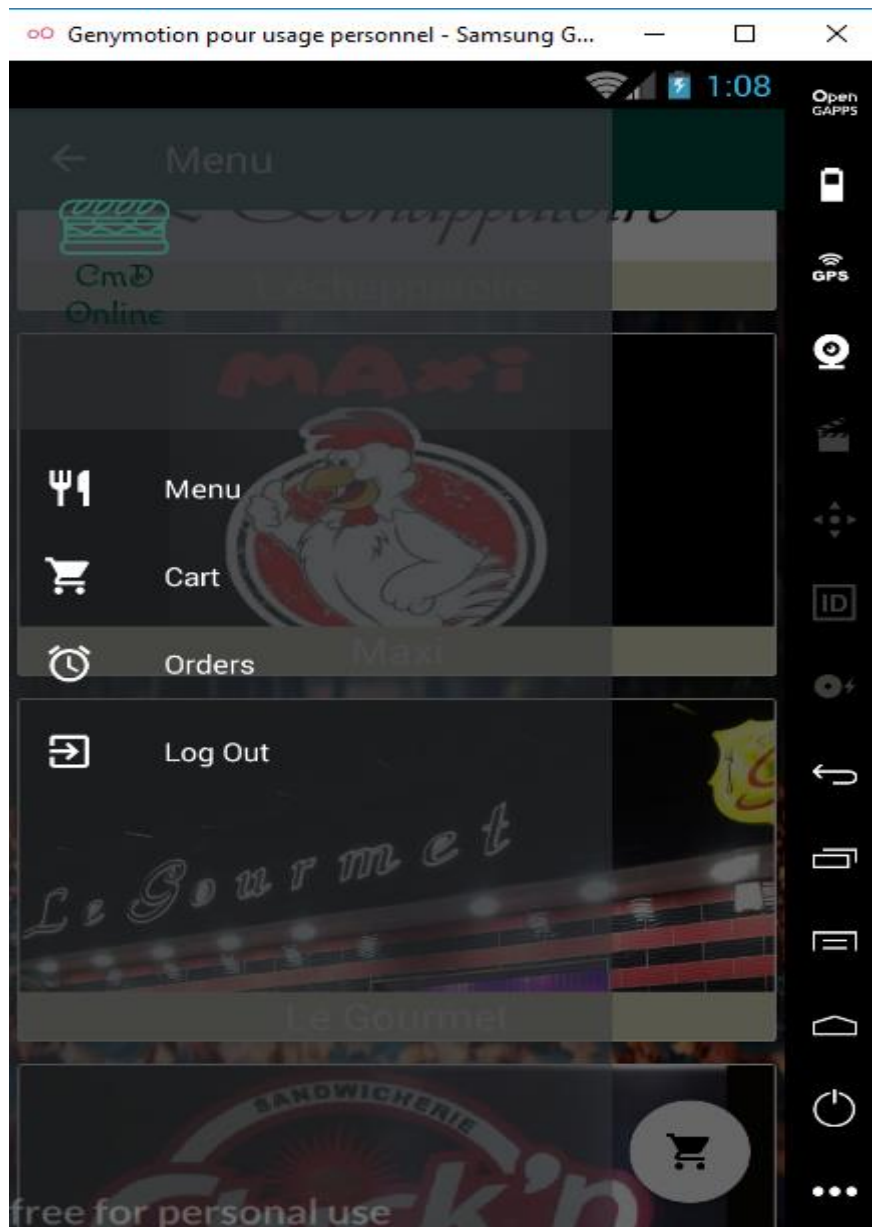


Figure 3.13 : La Barrede Menu.

5. Interface Cart :

Cette interface montre les sandwichs choisis dans cette commande aussi calcule le prix total et le temp estimé de chaque commande. L'utilisateur a une dernier étape pour confirmer sa commande et ajoute un commentaire (sans sel, les sauces voulus.....).

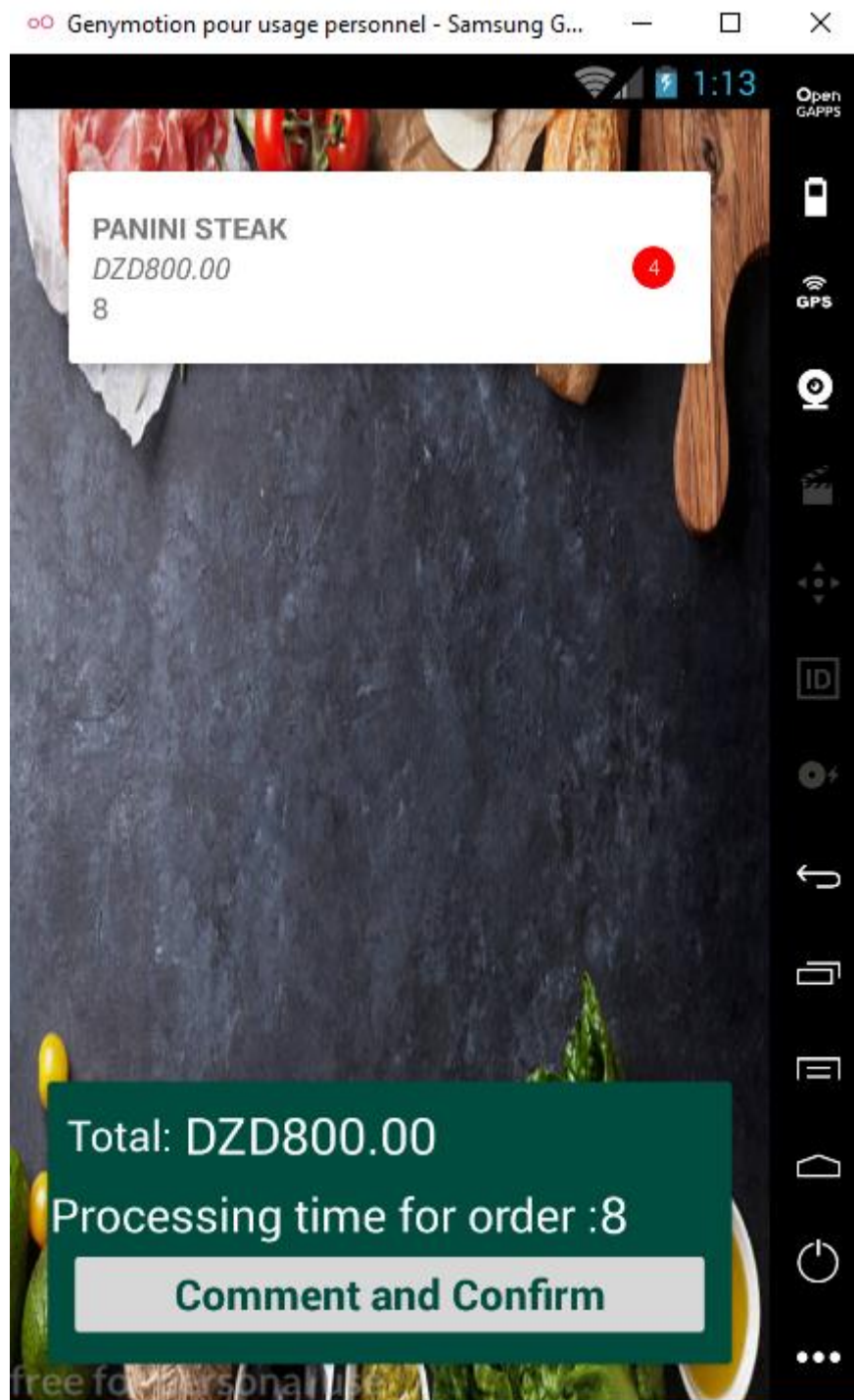


Figure 3.14 :Les Sandwich commandées.

Après une clique sur le bouton (Comment And Confirm), une petite fenêtre va s'afficher pour que l'utilisateur puisse ajouter un commentaire.

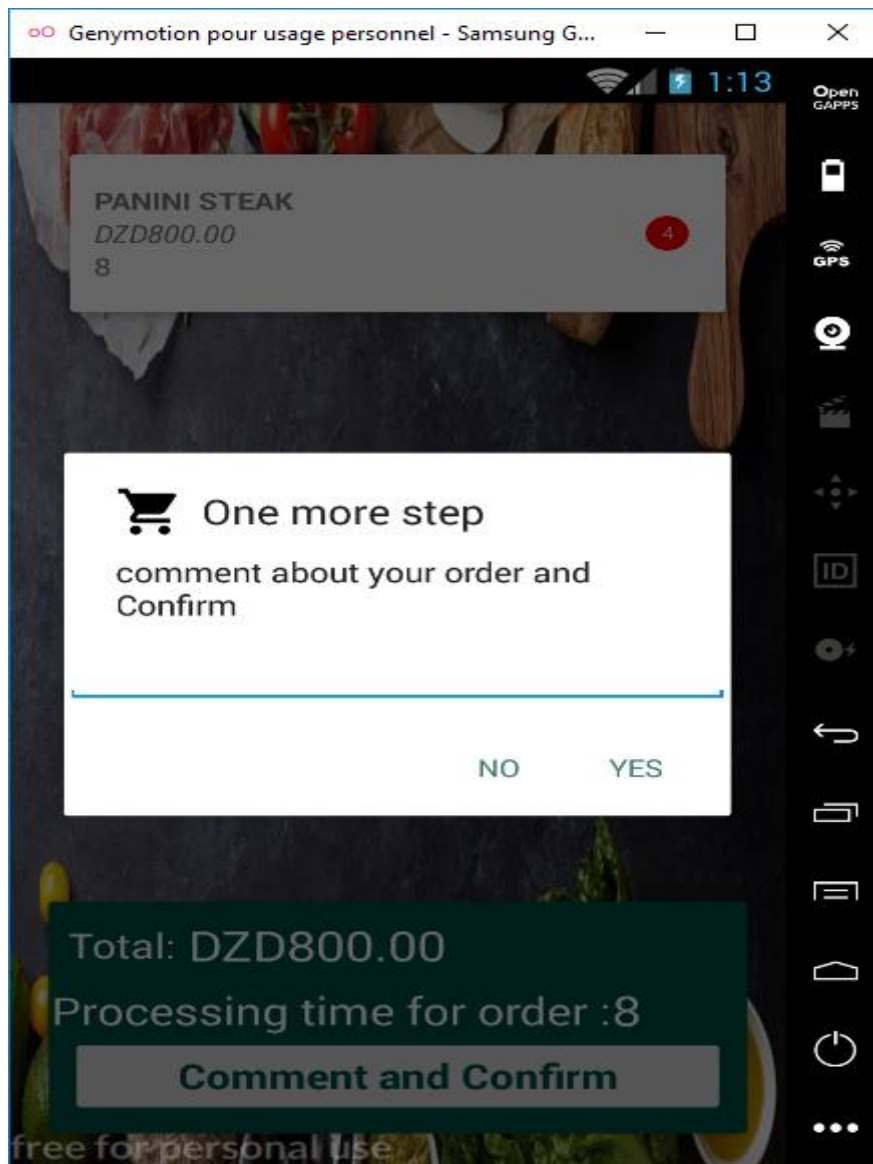


Figure 3.15 :Ajouter un commentaire et confirmer.

6. Interface Order :

Sur cette interface l'utilisateur peut connaître le temps d'attente estimé, et peut voir aussi si la commande est placée, au cour de préparation ou prête.

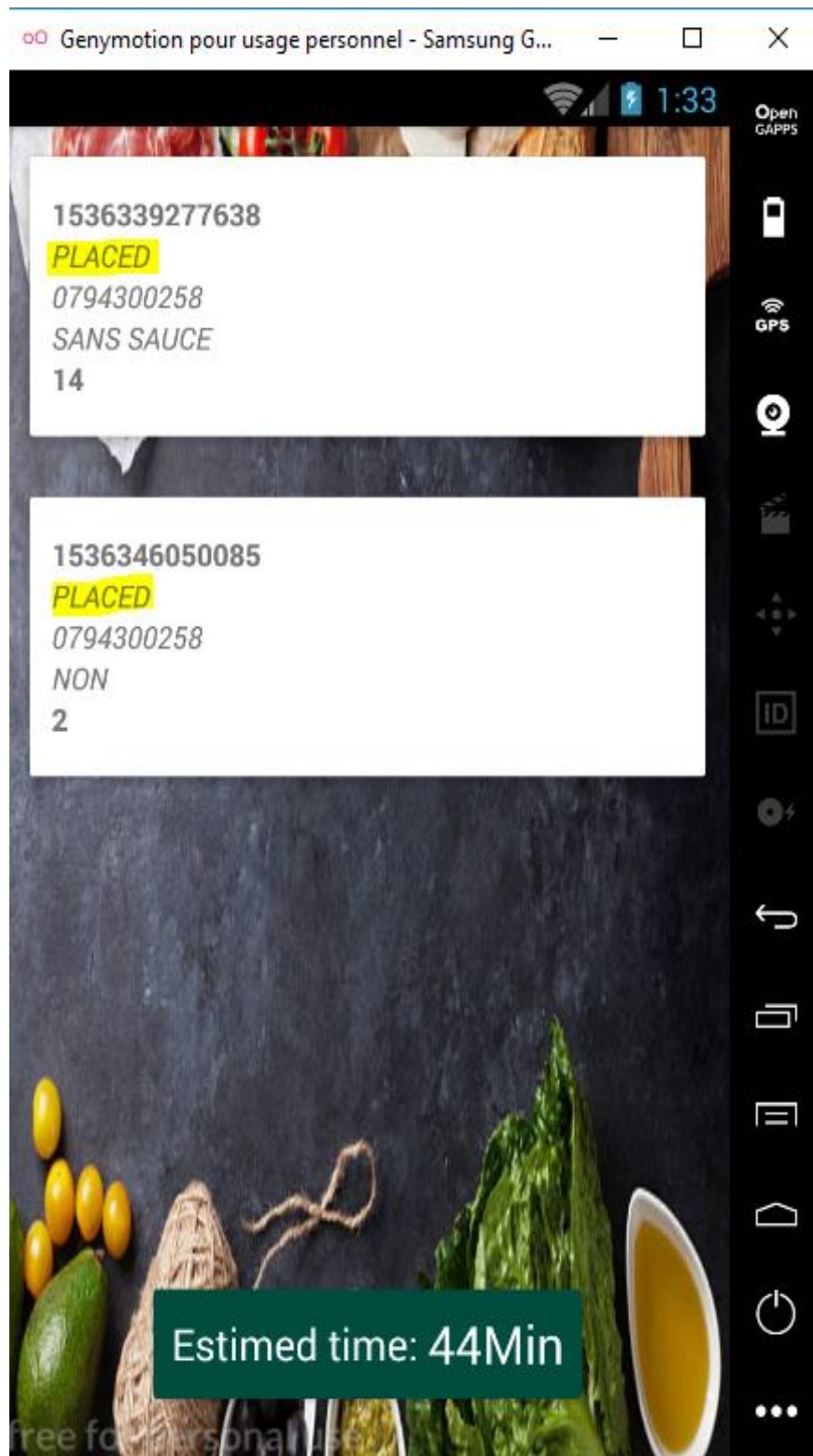


Figure 3.16 :Le statut de la commande.

3.4.2 Restaurant-Side:

Comme le côté du client la première interface et l'authentification, elle se fait avec un algorithme qui confirme si l'utilisateur est un employé d'un restaurant, après une interface s'affiche

1. Interface Menu :

L'employé de ce restaurant peut faire des plusieurs taches sur cette interface (Ajouter un sandwich, modifier ou supprimer ce dernier).

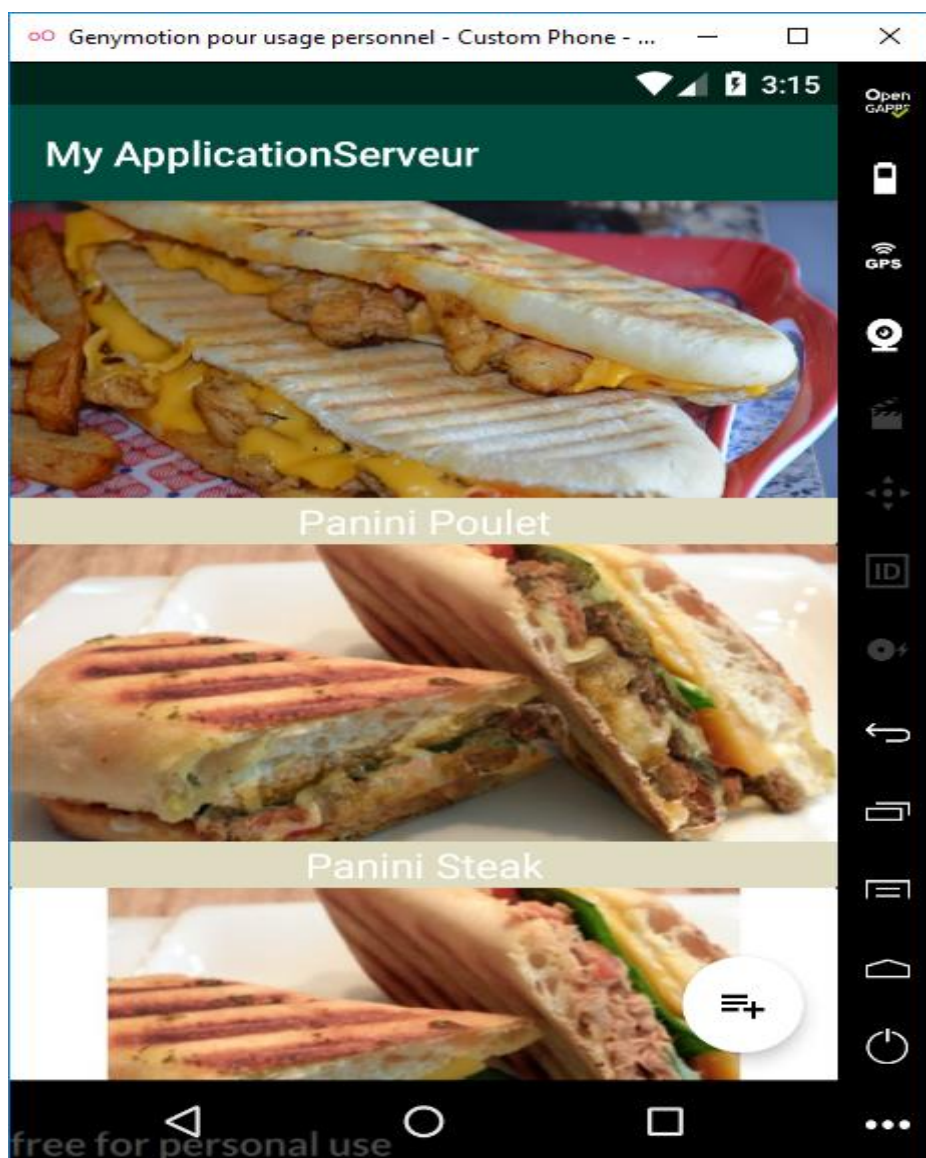


Figure 3.17: Liste des Sandwichs côté du restaurant.

En cliquant sur le bouton blanc qui donne à l'utilisateur la possibilité d'ajouter un nouveau sandwich a la base donnée en lui accordant sa propre photo choisie du téléphone et ses déférentes coordonnées.

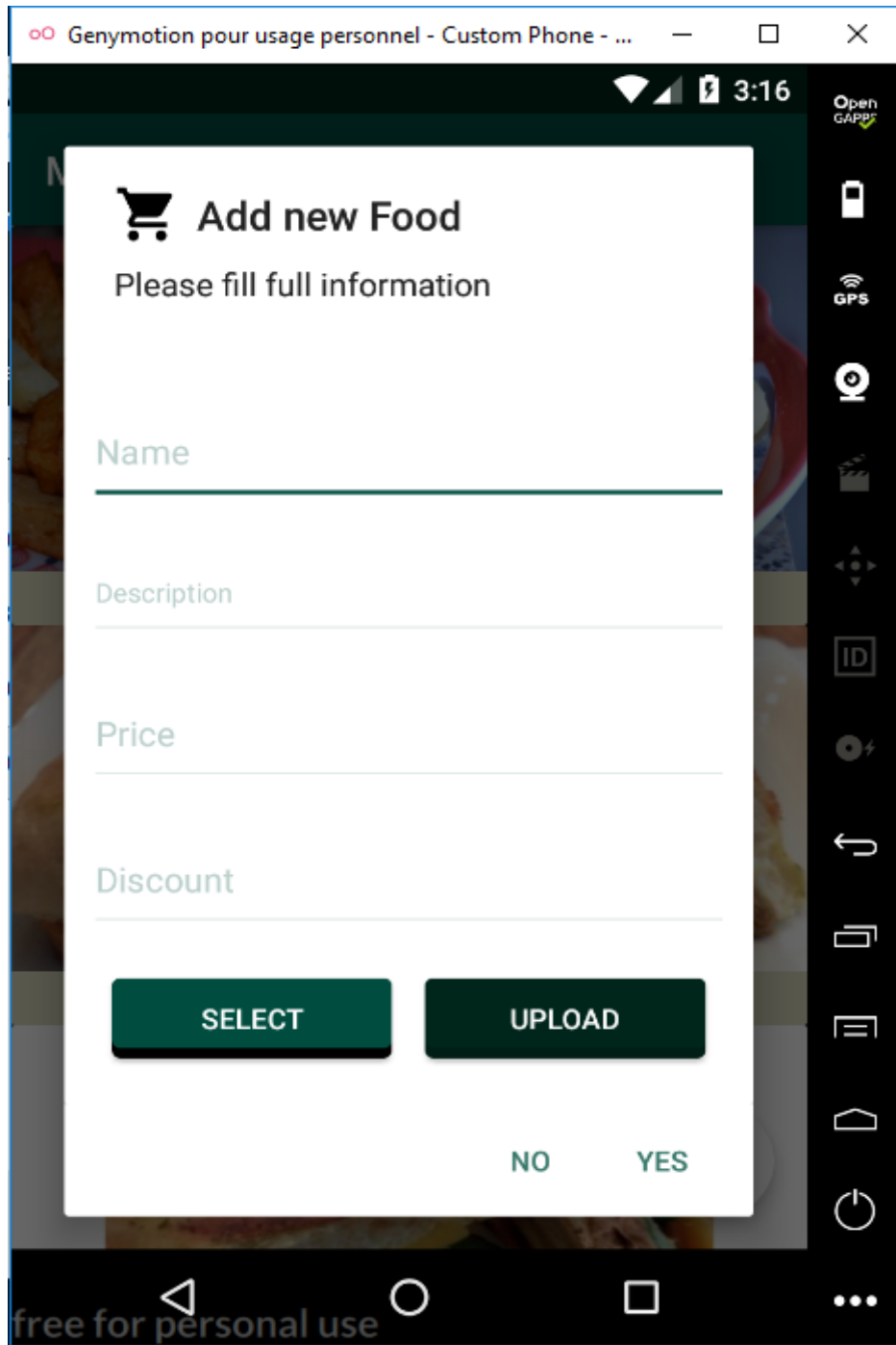


Figure 3.18 :Ajouter un nouveau Sandwich.

Il peut aussi récupérer les commandes, changer son statut (placé, en cour d'exécution, prête), dans l'interface Order.

2. Interface Order :

Dans cette interface l'employé peut faire la gestion de ces commandes.

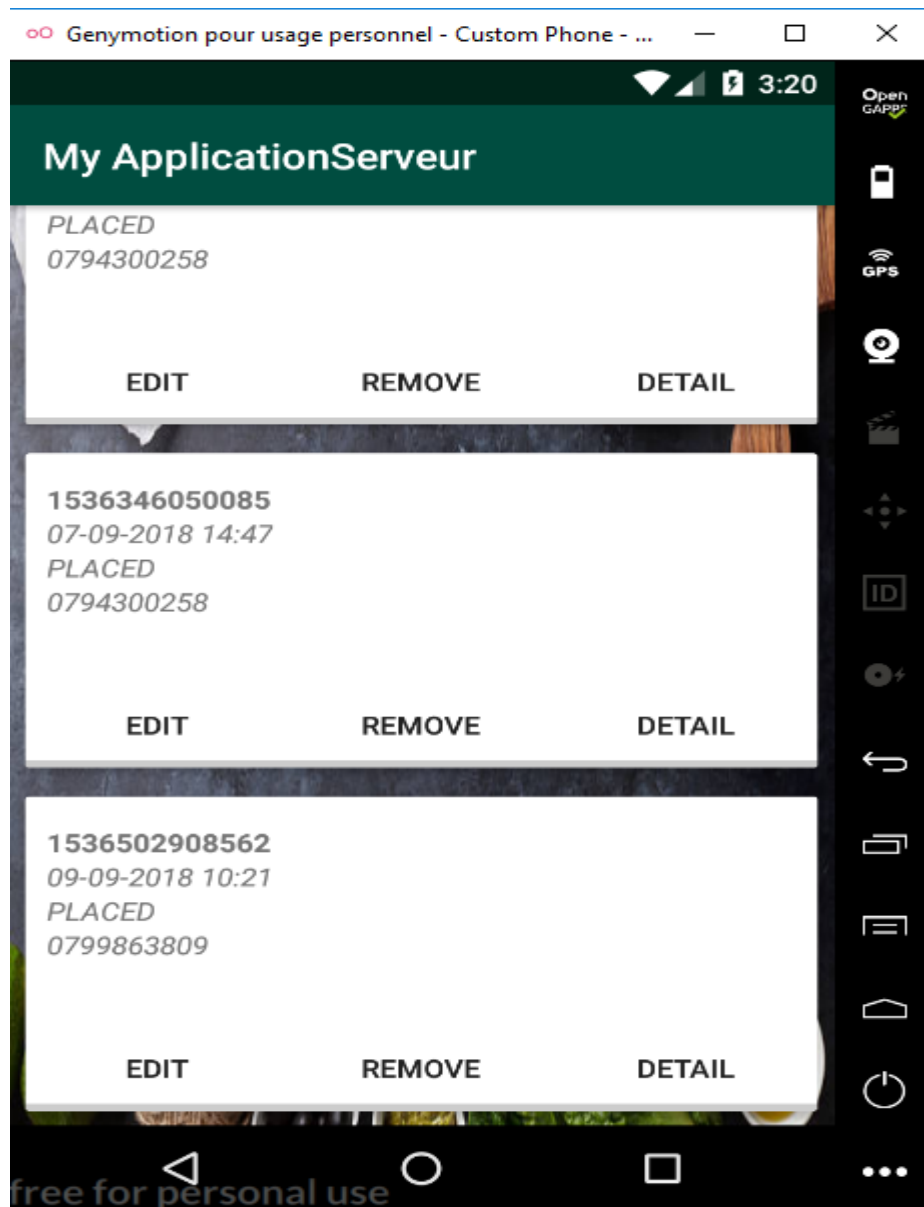


Figure 3.19 :Liste des commandes.

Chaque commande contient un ID, le numéro de téléphone de la personne pour le contacter au cas où il n'a pas venu récupérer sa commande.

-En cliquant sur EDIT : l'utilisateur pourrait changer le statut de cette commande.

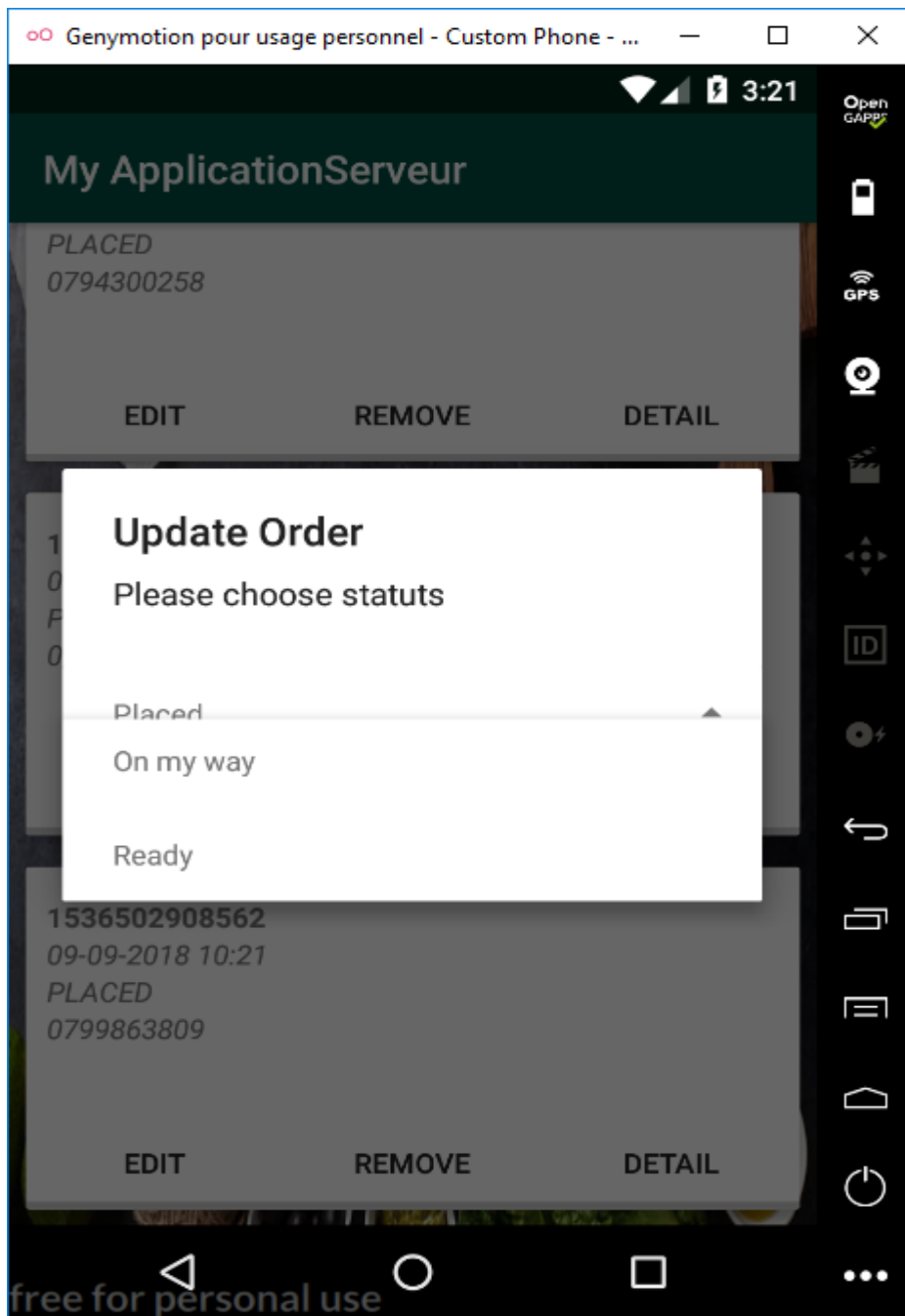


Figure 3.20 :Changer le statut de la commande.

-En cliquant sur REMOVE : l'utilisateur supprimerait la commande de la base de données, et passerait à la prochaine commande.

-En cliquant sur DETAIL : pour consulter les détails de cette commande.

3.5 Conclusion :

Dans ce chapitre nous avons présenté l'environnement matériel et logiciel utilisé dans la réalisation. Puis nous avons fourni quelques captures d'écrans décrivant des interfaces de notre application mobile. Aussi nous avons essayé d'expliquer à travers ces interfaces le fonctionnement de notre application.

Conclusion Général et perspectives

Conclusion générale perspectives

Tout au long de ce projet de fin d'étude, nous avons présenté les différentes étapes de la réalisation d'une application mobile, pour répondre aux exigences de commander de la nourriture par internet, pour les clients des restaurants de la ville de Tlemcen.

Pour bien réaliser ce projet, on a abordé l'ensemble des connaissances et des compétences acquises durant notre formation (Génie Industriel). En commençant par la technique UML qui nous a permis de mener correctement la tâche d'analyse des besoins à l'aide du diagramme de cas d'utilisation et la tâche de conception, ainsi les scénarios sont aussi détaillés afin d'expliquer toutes tâches faites. Puisque nous travaillons avec Android.

Ce projet nous a donné l'opportunité de s'imprégner de la vie professionnelle dans un milieu réel et avoir un début d'expérience significatif, par résoudre un problème très important qui est la gestion des commandes des restaurants. Il nous a appris aussi comment réussir de bonnes relations pour assurer un travail de groupe, comment compter sur soi pour résoudre les problèmes au cas où ils se présentent, comment être attentives aux indications de nos supérieurs, aussi comment être bien organisées pour accomplir dans les meilleurs délais et meilleures conditions les tâches qui nous sont confiées.

Le travail réalisé a apporté des solutions aux différents problèmes de gestion rencontrés au niveau des restaurants. Mais évidemment, ce travail étant une œuvre humaine, ce n'est pas un modèle parfait, c'est pourquoi nous restons ouverts à toutes les critiques et sommes prêts à recevoir toutes les suggestions et remarques tendant à améliorer davantage cette étude, parce que nous avons été astreints par quelques limites notamment, la contrainte du temps qui était relativement un obstacle devant l'ajout de certaines autres fonctionnalités.

Ce projet a fait l'objet d'une expérience à la fois intéressante et enrichissante, qui nous a permis d'améliorer nos connaissances et nos compétences dans le domaine du développement et de la conception de systèmes complexes.

Des perspectives d'amélioration de notre application restent toutefois indispensables. Nous envisageons ainsi d'ajouter de nouvelles fonctionnalités telles que les restaurants, les menus, la réservation des tables, Paiement via l'application.

Références bibliographiques

Bibliographie

[1] : Mémoire en Télécommunications, présenté par KEHLI Assia et BENNEZAIR Nadjet, Une application mobile Développement d'une application ANDROID pour le partage des événements sur Google Map, le 14 / 06 /2017 a Tlemcen, encadré par M. KADRI Benamar.

[2] : A. RADU, «Évaluation de la Qualité de Service par l'utilisateur final dans les systèmes mobiles», Thèse de doctorat en Informatique et Télécom de l'Université de Mame -La-Vallée, France, Mars, 2004. <http://pelleas.univ-mlv.fr/document/UMLV-2004-000235-PDF.pdf>

[3] : Définition et historique du Smartphone - SCT TELECOM <http://www.sct-telecom.fr/glossaire/smartphone/> Consulté le 20/3/2017.

[4] : http://fr.wikipedia.org/wiki/Application_mobile- Consulté le 20/3 /2017.

[5] : <http://www.definitions-webmarketing.com/Definition-Application-mobile-> Consulté le 20/3/2017.

[6]: [http://www.mobileenfrance.com/2009/08/18/application-embarquee-ou-service-mobile-%E2%80%93-partie-1-definitions/-](http://www.mobileenfrance.com/2009/08/18/application-embarquee-ou-service-mobile-%E2%80%93-partie-1-definitions/) consulté 20/3/2017

[7] : Google image.

[8] : <http://fr.clever-age.com/veille/blog/comment-aborder-un-projet-de-mobilite.html>. Consulté le 25/03/2017.

[9] : <http://fr.wikipedia.org/wiki/Smartphone>, Consulté le 28 Mars 2017

[10] : http://fr.wikipedia.org/wiki/Syst%C3%A8me_d'exploitation_mobile, Consulté le 22/08/2018.

[11] : <http://www.wikitude.com/developper/wikitude-augmented-reality-fordevelopers>, Consulté le 28/3/2017.

[12]: <http://www.developpez.net/forums/d400869/generaldeveloppement/conception/modelisation/uml/pourquoi-utiliser-uml-lanalyseapplications-web/>, Consulté le 28/3/2017.

[13] : http://fr.wikipedia.org/wiki/Syst%C3%A8me_d'exploitation_mobile, 28/3/2017.

[14] : Mémoire en informatique, présenté par Telli Fatiha et Benjeddou Khadija, Une application mobile : Une télécommande Bluetooth d'une souris optique, le 29/9/2013 a Ouargla, encadré par M. DJEDIAI Hmida.

[15]:<http://olivierguillet.com/2012/02/les-differents-types-dapplications-mobiles-natives-webappshybrides-flash/>- Consulté le 28/3/2017.

[16] : DIABETE, S. (n.d.). Guide des applications diabètes. Retrieved mars 20, 2016, from dmdpost: <http://gad.dmdpost.com/>.

[17] : Mémoire Pour l'obtention du diplôme de Master Dans le programme ESC Grande Ecole Présenté et soutenu par Manon Le Corre Le 04 juillet 2011 les réseaux sociaux dans une stratégie de communication d'une grande entreprise Directeur de mémoire David Mérieau.

[18] : DIABETE, S. (n.d.). Guide des applications diabètes. Retrieved mars 20, 2016, from dmdpost: <http://gad.dmdpost.com/>.

[19]: Thomas Pender.UML Weekend Crash Course, Wiley publishing, USA, 2002.

[20] :BenoîtCharroux, AomarOsmani, Yann Thierry-Mieg. UML 2 Pratique de la modélisation - Collection Synthex 2e édition, 2008.

[21] : Adel RAISSI, Conception et développement d'un site web de commerce pour le compte de LSAT « Nokia, Pour obtenir le mastère en nouvelles technologies de télé- communication et réseaux, Université virtuelle de Tunis, 2012/2013.

[22] : Mohamed Anouar DAHDEH, conception, développement et intégration d'une application embarquée de téléchargement des applications android 2010 – 2011.

[23] :<https://www.developpez.com/actu/168839/Android-Studio-3-0-est-disponible-avec-le-support-de-Kotlin-plus-de-fonctionnalites-Java-8-et-bien-plus/>

[24] : <https://developer.android.com/distribute/best-practices/develop/build-with-firebase?hl=fr>

[25] : <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203607-sqlite-definition/>

[26] : Utilisation de Genymotion. (2015, July 29). Consulté le mai 15, 2016, sur CA Technologies: <https://docops.ca.com/devtest-solutions/8-0-2/fr/installation/configuration-de-l-environnement-de-test-d-application-mobile/utilisation-de-genymotion>.

[27]: D. ESPOSITO, A. SALTARELLO, Microsoft .NET: Architecting Applications for the Enterprise, Microsoft 2008.

Résumé

Durant ce modeste travail, nous nous sommes intéressés aux développements d'application mobile fiable et aisée sous le système d'exploitation « Android ». Notre travail a été réalisé pour régler le problème de la surcharge des commandes au niveau des restaurants de Tlemcen.

Dans ce projet nous avons utilisé UML comme un langage de modélisation conçu pour fournir une méthode normalisée pour la conception, et la construction des documents nécessaires au bon développement. Cette conception est mise en œuvre sous l'environnement de développement Intégré (Android Studio) comme un langage de programmation, un système de gestion de base de données Firebase.

Abstract

During this modest work, we were interested in the developments of reliable mobile application under the operating system " Android ". Our work was realized to solve the problem of the overload of the orders at the level of the restaurants of Tlemcen.

In this project we used UML as a language of modelling to supply a method for the conception, and the construction of necessary documents for the better development.

This conception is implemented under the environment of development (Android Studio) as a programming language, a database management system Firebase

ملخص

خلال هذا العمل المتواضع، كنا مهتمين بتطوير تطبيقات الهاتف المحمول موثوقة وسهلة في ظل نظام التشغيل "أندرويد". لقد تم عملنا لحل مشكلة أوامر التحميل الزائد في المطاعم في تلمسان..

في هذا المشروع استخدمنا UML كلغة مصممة لتوفير طريقة موحدة لتصميم وبناء الوثائق اللازمة للتطوير السليم. يتم تنفيذ هذا التصميم تحت بيئة التطوير المتكاملة (Android Studio) كلغة برمجة، وهو نظام إدارة قاعدة بيانات Firebase

Mots clés : UML, SQL, FireBase, Android Studio, Application, Restaurant, Smartphones.