

Remerciements

Résumé

Table des figures

1.1	Logo Android	9
1.2	Déférence entre les surcouches EMUI, MIUI et ONE UI	9
1.3	L'interface utilisateur d'Android 10 sans l'application des surcouches	10
1.4	L'architecture d'Android	10
1.5	Logo iOS	11
1.6	L'interface de iOS	11
1.7	L'architecture de iOS	12
1.8	Classement des meilleures applications dans le monde pour juillet 2020 par téléchargements	13
1.9	Les trois types d'applications mobile	14
2.1	Logo de Flutter	16
2.2	Exemple de l'application Reflectly développée avec le framework Flutter	17
2.3	L'architecutre du framework Flutter	17
2.4	Logo de Firebase	20
2.5	Logo de Parse	22
2.6	Exemple de la tarification d'une partie des services Firebase	23
2.7	Logo de Google APIs	24
2.8	Exemple de l'integration des services Google Maps dans une application mobile	25
3.1	Logo de TripAdvisor	27
3.2	Interface de TripAdvisor lors de la recherche des restaurants et les avis	27
3.3	Logo de Google Maps	28
3.4	Exemple de différents ecrans lors d'une recherche d'un restaurant	28
3.5	Logo de LaFourchette	29
3.6	Enchainement des actions pour le lancement de l'application	30
3.7	Enchainement des actions pour la recherche des restaurants	30
3.8	Enchainement des actions pour l'affichage de la page restaurant et son menu	31
3.9	Diagramme d'actions	31
3.10	Logo de Balsamiq	32
3.11	Interface du logiciel Balsamiq	32
3.12	Ecran principal	33
3.13	Ecran de connexion	33
3.14	Ecran de creation du compte	34
3.15	Ecran de confirmation du compte	34

3.16 Ecran de réinitialisation du mot de passe	35
3.17 Ecran d'affichage de la fenêtre du restaurant	35
3.18 Page de détails du restaurant	36
3.19 Page de détails du menu	36
3.20 Page d'avis	37
3.21 Page de recherche	37
3.22 Page du profil et paramètres	38
3.23 Page du filtre	39
3.24 Page des restaurants favoris	39
3.25 Exemple de composantes du Material Design	40
3.26 Logo de Adobe XD	41
3.27 interface du logiciel Adobe XD	41
3.28 Interfaces utilisateur de l'application	44
3.29 Logo de Visual studio code	46
3.30 Interface du logiciel Visual Studio Code	46
3.31 Logo de Firebase	47
3.32 Interface de la console de Firebase	47

Liste des tableaux

Table des matières

Remerciements	1
1 Generalités sur les applications mobiles	8
1.1 Introduction	8
1.2 Les systèmes d'exploitation mobiles	8
1.2.1 Définition	8
1.2.2 Le système Android	9
1.2.3 Le système iOS	11
1.3 Les applications mobiles	13
1.3.1 Definition	13
1.3.2 Les types des applications mobiles	14
1.3.3 Le cycle de vie d'une application mobile	15
2 Les technologies utilisées	16
2.1 Introduction	16
2.2 Flutter	16
2.2.1 Principes de base	18
2.2.2 Pourquoi utiliser flutter ?	19
2.3 Firebase	20
2.3.1 Les services de Firebase	20
2.3.2 Pourquoi utiliser Firebase ?	22
2.4 Google APIs	24
2.4.1 Google Maps	24
3 Conception et implémentation de l'application	26
3.1 Introduction	26
3.2 Conception	26
3.2.1 Objectif du projet	26
3.2.2 Applications déjà existantes	26
3.2.3 Les diagrammes UML	29
3.2.4 Maquette fonctionnelle (Wireframe)	32
3.2.5 Prototypage et interface utilisateur	40
3.3 Implementation	45
3.3.1 Outils Materiels	45
3.3.2 Outils Logiciels	46

3.3.3 Developpement	47
4 Conclusion et perspectives	49

Chapitre 1

Generalités sur les applications mobiles

1.1 Introduction

Dans ces derniers temps, la possession d'un smartphone ou une tablette est devenue indispensable dans notre vie quotidienne, ces derniers sont gérés par de nombreux systèmes d'exploitation.

Les systèmes d'exploitation sont des logiciels conçus pour faire fonctionner un smartphone en permettant à l'utilisateur de réaliser plusieurs tâches comme passer un appel, prendre des photos et même télécharger des applications pour mieux exploiter son smartphone.

Dans ce chapitre nous allons nous focaliser sur les différents systèmes d'exploitation et le développement des applications mobiles , nous commençons par une définition d'Android et IOS, ensuite nous allons faire une comparaison entre les deux systèmes d'exploitation et pour finir , nous allons parler sur le développement des applications mobiles et les différents outils utilisés dans ce domaine.

1.2 Les systèmes d'exploitation mobiles

1.2.1 Définition

Un système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface. [8]

Les systèmes d'exploitation mobiles sont généralement installés sur les smartphones. Il existe un grand nombre de systèmes d'exploitation tel que : Windows Phone, iOS, Android, BlackberryOs, Symbian OS et Bada.

Les deux systèmes d'exploitation les plus présents en ce temps sont Android de l'entreprise Google et iOS d'Apple. Ces deux systèmes offrent de manière régulière des mises à jour qui améliorent la qualité et les performances du système en plus de renforcement de la sécurité de l'appareil.

1.2.2 Le système Android

Android est un système d'exploitation créé par Google, ce système permet d'utiliser son smartphone, le personnaliser et même télécharger des applications pour faciliter les tâches quotidiennes de l'utilisateur et mieux exploiter son appareil.

Lancé en juin 2007 à la suite du rachat par Google en 2005 de la startup du même nom⁶, le système avait d'abord été conçu pour les smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs comme les télévisions (Android TV), les voitures (Android Auto), les Chromebook (Chrome OS qui utilise les applications Android) et les smartwatch (Wear OS). [1]

Le système Android a été développé et ce jusqu'à la version 10 appelée Android Q. Ce système est distribué vers les smartphones et les tablettes sous trois formes :

- Un système d'exploitation customisé en appliquant une surcouche logicielle qui va avec les modèles de chaque constructeur comme One UI de Samsung, MIUI de Xiaomi et EMUI de Huawei. cela apporte une meilleure expérience utilisateur



FIGURE 1.1 – Logo Android

FIGURE 1.2 – Différence entre les surcouches EMUI, MIUI et ONE UI

- Un système d'exploitation sans l'application d'une surcouche. cette version est installé par exemple sur les smartphones Android Go, Ces derniers sont mis à jour rapidement vers de nouvelles versions de ce système.
- Le système Android est installé aussi sous la forme de versions alternatives appelées Custom ROM. On installe ces ROM par exemple lorsqu'un smartphone est incompatible avec une version Android, ce ROM s'installe avec la version courante d'Android mais avec une interface d'un système différent.

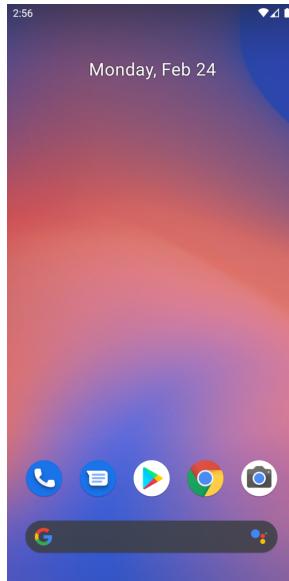


FIGURE 1.3 – L’interface utilisateur d’Android 10 sans l’application des surcouches

L’architecture d’Android :

L’architecture du système Android est sous forme d’une pile de composants comme on le voit dans la figure suivante :

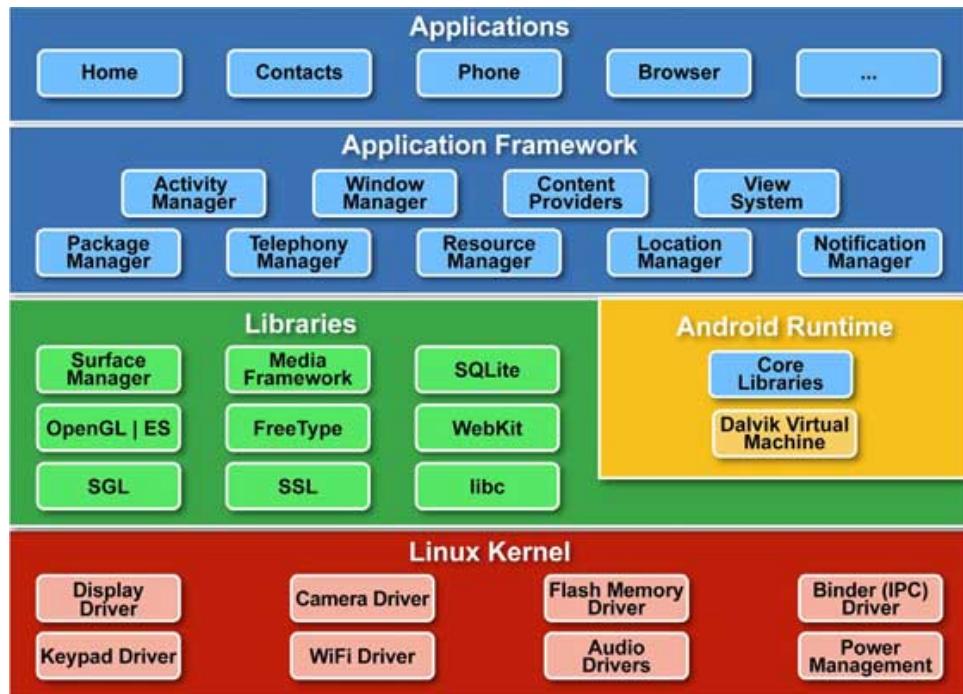


FIGURE 1.4 – L’architecture d’Android

En lisant la figure de bas en haut, on trouve dans la première couche que ce système est basé sur le noyau (ou Kernel) de Linux conçu spécialement pour l’environnement mobile, ce

composant joue le rôle d'un pont entre les composants matériels et logiciels. Il contient donc tous les pilotes, ces derniers utilisent des fonctions qui permettent de contrôler le matériel.

Au-dessus du kernel il y a le "Hardware abstraction layer" qui permet de séparer la plate-forme logique du matériel. Au-dessus de cette couche d'abstraction on retrouve les librairies C/C++ utilisées par un certain nombre de composants du système Android. On retrouve ensuite l'Android Runtime, cette couche contient les librairies coeurs du Framework ainsi que la machine virtuelle exécutant les applications. Au-dessus la couche "Android Runtime" et des librairies coeurs on retrouve le Framework permettant au développeur de créer des applications. Enfin au-dessus du Framework il y a les applications.

1.2.3 Le système iOS

iOS, anciennement iPhone OS , est le système d'exploitation mobile développé par Apple pour plusieurs de ses appareils. Il est dérivé de macOS dont il partage les fondations (le noyau hybride XNU basé sur le micro-noyau Mach, les services Unix et Cocoa, etc.).

iOS comporte quatre couches d'abstraction, similaires à celles de macOS : une couche « Core OS », une couche « Core Services », une couche « Media » et une couche « Cocoa ». Le système d'exploitation occupe au maximum 3 Go de la capacité mémoire totale de l'appareil, selon l'appareil. [15]

Ce système d'exploitation est connu par sa rapidité et fluidité sur les appareils d'Apple vu qu'il est développé spécialement pour les iphones contrairement à Android.



FIGURE 1.5 – Logo iOS

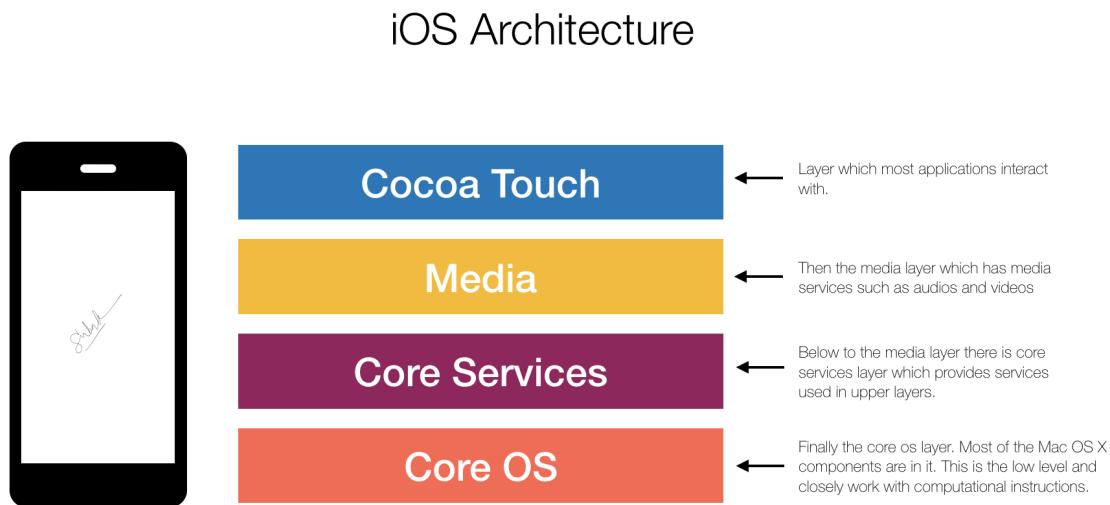


FIGURE 1.6 – L'interface de iOS

L'architecture d'iOS :

Comme on l'a déjà mentionné dans la partie précédente, iOS est basé sur macOS. Son architecture est composée de 4 éléments : le BaseBand, le BootLoader, le Firmware et le SeckPack.

- Le BaseBand est un micrologiciel autonome qui gère en temps réel tous les périphériques de communication de l'appareil. Le BaseBand est considéré donc comme le BIOS de l'iPhone.
- Le BootLoader est le programme chargé du démarrage du système iOS, c'est le premier processus exécuté lorsqu'on allume l'iphone
- Le Firmware est un programme interne dans l'iPhone, ce programme prend contrôle de la partie systématique de l'appareil comme la caméra, l'écran et le clavier tactile.
- Le SeckPack est une partie de la mémoire flash de l'appareil contenant entre autres des informations sur le verrouillage de celui-ci. Le Seckpack peut être considéré comme un mot de passe : en effet, si un SeckPack correct est fourni au BootLoader lors du lancement, alors l'utilisateur a la possibilité d'utiliser le BaseBand, et donc les fonctionnalités de téléphonie et d'Internet. [15]



All Rights Reserved To Anuradha Caldera

FIGURE 1.7 – L'architecture de iOS

1.3 Les applications mobiles

1.3.1 Définition

Une application mobile est un logiciel installable sur les appareils mobiles par exemple une tablette tactile ou un smartphone, pour exécuter des tâches spécifiques.

La plupart des applications mobiles sont distribuées au public par le biais de plateformes de téléchargement généralement gérée par les fabricants de appareils mobiles comme Google Play Store pour Android/Google ou encore App Store pour les produits Apple. Ces applications sont disponibles soit en version gratuite et qui contient généralement des publicités soit en version payante.

Sur certaines plateformes, les applications peuvent aussi être installées à partir de sources tierces, via un site non affilié au distributeur d'origine. Sur Android, cela est possible en activant le mode développeur. Sur iOS, cette manipulation est possible soit en étant développeur Apple, soit en possédant un appareil jailbreaké. [11]

les applications mobiles se regroupent en plusieurs séries, on trouve :

- Les jeux mobiles.
- Les applications à but éducatif.
- Les applications de géolocalisation.
- Les applications d'écoute de musiques ou de radios et de streaming vidéo.
- Les applications pour la consultation d'Internet.
- Les applications des réseaux sociaux.

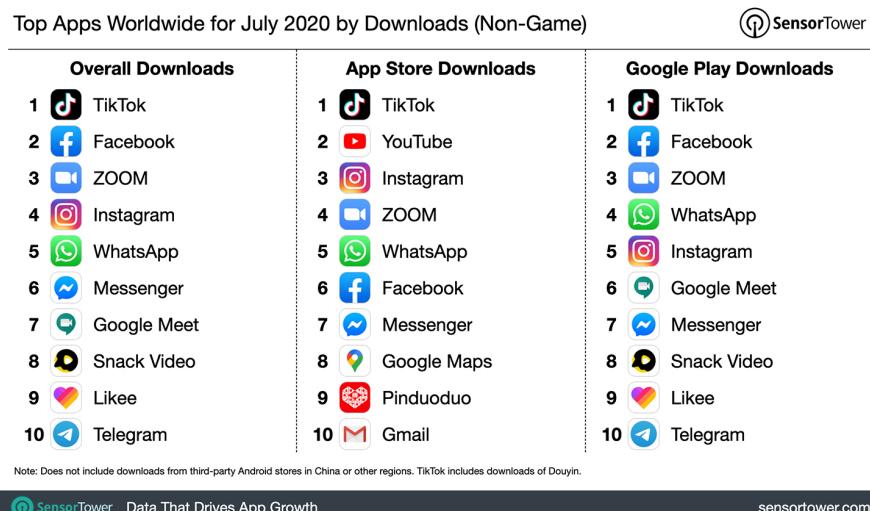


FIGURE 1.8 – Classement des meilleures applications dans le monde pour juillet 2020 par téléchargements

1.3.2 Les types des applications mobiles

Il existe dans le domaine des applications mobiles trois types d'applications en parlant du langage de programmation et les tâches que fait l'application. On trouve donc :

Application native :

Les applications natives sont des applications conçues spécialement pour les systèmes d'exploitation fiables par les smartphones. Ces applications utilisent chaque un langage de programmation spécifique pour chaque système d'exploitation, comme Java pour Android et Swift pour iOS.

Application web :

Une application web est une application conçue avec le langage HTML et CSS, ce type d'application fonctionne de manière flexible sur tous les navigateurs internet dans les appareils mobiles.

Application hybride :

Les applications hybrides regroupent les caractéristiques des applications web et applications mobilité, elles sont donc accessibles via toutes les plateformes des applications. ce type d'application réduit la durée et les charges du développement.

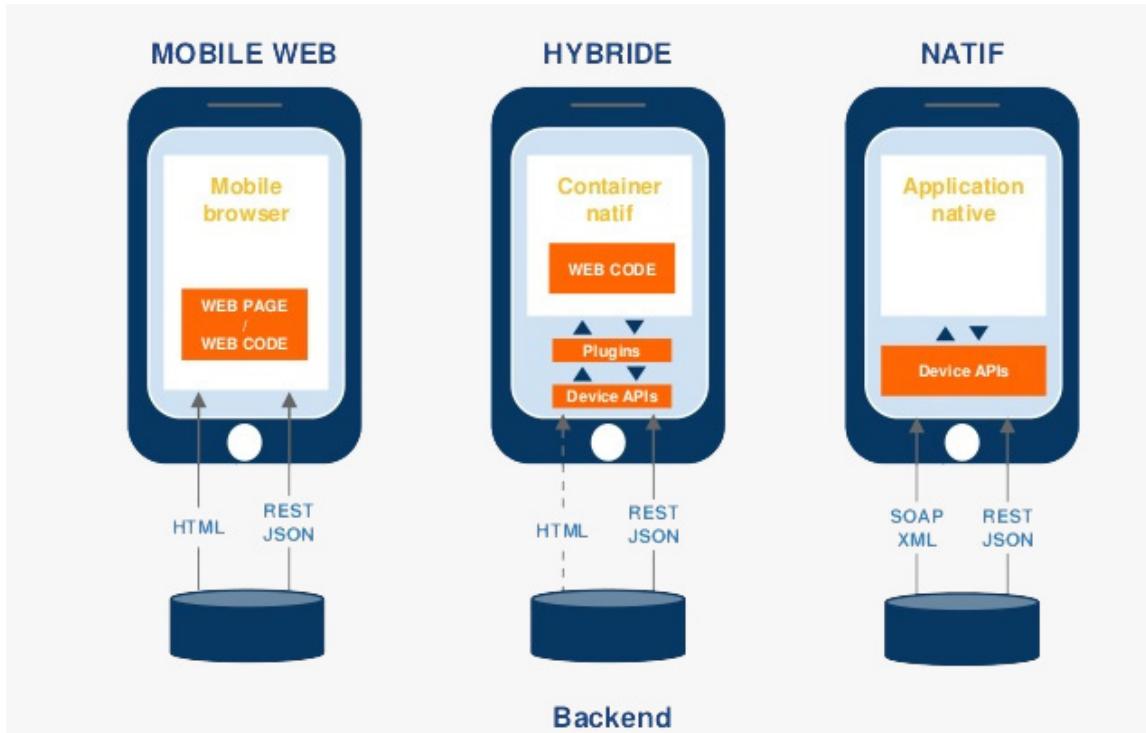


FIGURE 1.9 – Les trois types d'applications mobile

1.3.3 Le cycle de vie d'une application mobile

Le cycle de vie d'une application mobile se compose de 6 phases de travail [2], citons-les :

1- Planification et analyse :

La première étape de la conception consiste à analyser la situation pour tenir compte des contraintes, des risques et de tout autre élément pertinent et assurer un ouvrage ou un processus répondant aux besoins du client.

2- Spécification technique :

Dans cette étape, On détermine le type des opérations ainsi que les systèmes pour lesquels on souhaite créer l'application mobile. On établit à la fin un cahier de charge.

3- Prototypage et conception :

À cette étape, des diagrammes UML ainsi que des diagrammes d'action sont créé puis on crée un prototype de l'application qui aide le client pour valider l'idée et commencer la phase suivante

4- Développement :

Le processus de développement d'applications est divisé en deux phases : front-end et backend. Cette étape consiste à coder les modules de l'application un par un et effectuer des tests sur place avant d'entamer le prochain module. Cependant, le développement de la partie backend n'est pas toujours nécessaire si on utilise des techniques cloud pour le traitement et le stockage des données

5- Tests et assurance de qualité :

La phase de tests et assurance de qualité est une partie très importante pour la création de l'application. cette phase comprend des tests des exigences, interfaces et de la sécurité de l'application et des données ainsi que les ressources du bas niveau.

6- Maintenance et mise à jour :

Après la phase de tests, on est dans la phase finale où l'application est prête pour la distribution dans les plateformes d'achat des applications, mais le travail ne s'arrête pas là, l'application doit être mise à jour de façon régulière pour répondre aux nouveaux besoins des utilisateurs.

Chapitre 2

Les technologies utilisées

2.1 Introduction

Ces dernières années, le secteur du développement des applications mobiles a connu une majeure évolution avec l'apparition des outils de développement cross-platform. Ces outils permettent la création d'une application mobile avec une base de code unique sur les différents environnements Ios et Android.

Au cours de ce chapitre nous allons nous présenterons d'abord Flutter, un des frameworks utilisés pour le développement des applications multiplate-forme, ensuite nous allons parler de Firebase, un service Google utilisé dans le côté Backend des applications. Pour finaliser nous allons définir les api Google ainsi que google maps.

2.2 Flutter

Flutter est un SDK d'application permettant de créer des applications haute performance et haute fidélité pour ios, Android, Web (version bêta) et de bureau (aperçu technique) à partir d'une base de code unique. [9] L'objectif est de permettre aux développeurs de fournir des applications hautes performances qui semblent naturelles sur différentes plateformes notamment appelés les applications multi-platformes.

Flutter s'appuie sur le langage de programmation DART (à l'origine appelé Dash), créé également par Google et présenté au public en 2011.

Les applications Flutter s'exécutent dans une machine virtuelle, cette dernière offre une fonctionnalité du recharge rapide de l'application sans avoir besoin de recompiler le projet, les applications sont donc compilées directement en code machine, soit en Javascript si elles sont destinées au web ou en instructions Intel X64 ou ARM.



FIGURE 2.1
– Logo de
Flutter

Le framework Flutter est open source, avec une licence BSD permissive, et dispose d'un écosystème florissant de packages tiers qui complètent les fonctionnalités de base de la bibliothèque.

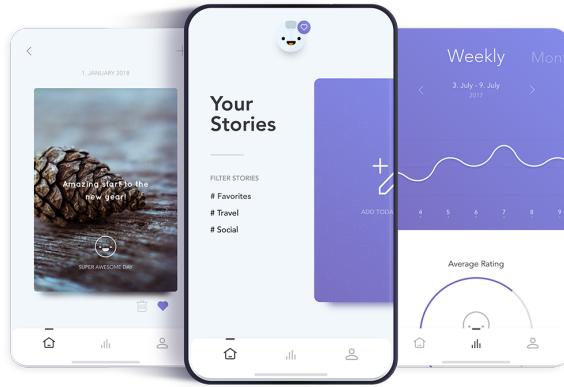


FIGURE 2.2 – Exemple de l'application Reflectly développée avec le framework Flutter

Flutter est conçu comme un système extensible en couches. Il existe sous la forme d'une série de bibliothèques indépendantes qui dépendent chacune de la couche sous-jacente. Aucune couche n'a un accès privilégié à la couche ci-dessous, et chaque partie du niveau de structure est conçue pour être facultative et remplaçable. [4]

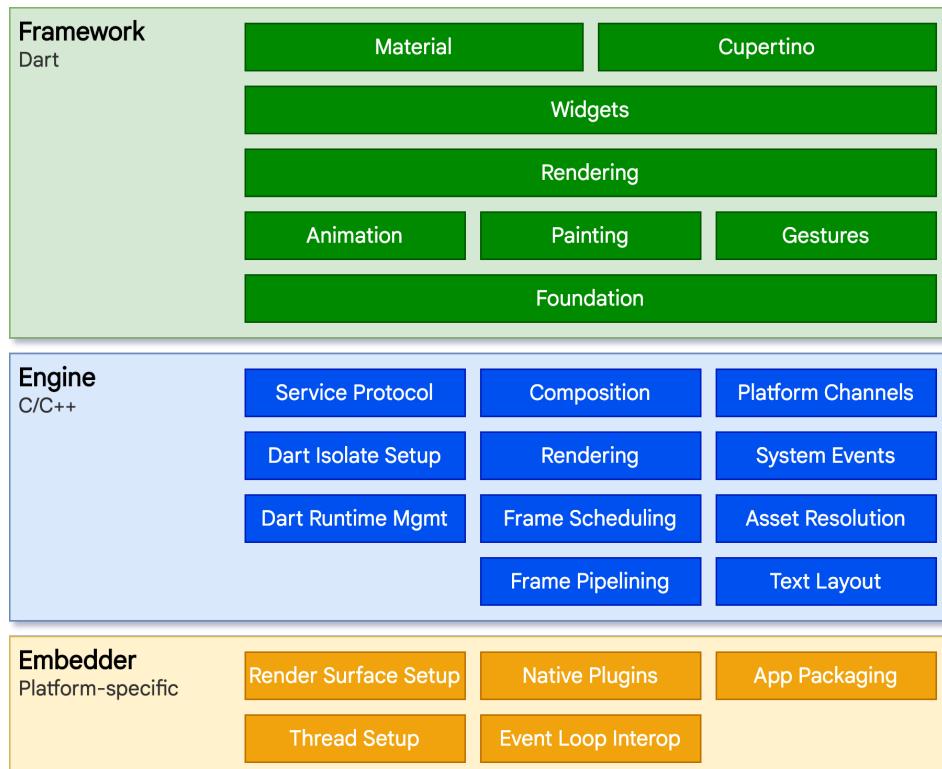


FIGURE 2.3 – L'architecutre du framework Flutter

2.2.1 Principes de base

Flutter comprend un framework de style react moderne, un moteur de rendu 2D, des widgets prêts à l'emploi et des outils de développement. Ces composants fonctionnent ensemble pour vous aider à concevoir, créer, tester et déboguer des applications. Tout est organisé autour de quelques principes fondamentaux.

1. Tout est un widget :

Les widgets sont les éléments de base de l'interface utilisateur d'une application Flutter. Chaque widget est une déclaration immuable d'une partie de l'interface utilisateur. Contrairement aux autres frameworks qui séparent les vues, les contrôleurs de vue, les présentations et d'autres propriétés, Flutter possède un modèle objet cohérent et unifié : le widget. Un widget peut définir :

- un élément structurel (comme un bouton ou un menu).
- un élément stylistique (comme une police ou un jeu de couleurs).
- un aspect de la mise en page (comme le rembourrage).
- etc...

2. L'utilisation du language Dart

Le Dart est un langage de programmation structuré, open source et évolutif qui fonctionne à l'aide d'une machine virtuelle. Le projet de google à travers le Dart est de faciliter le développement web, de combler les carences du Javascript et d'offrir de meilleures performances, surtout pour les gros projets. Le Dart est destiné à fonctionner sur les navigateurs web modernes ainsi que sur les serveurs et il embarque également un convertisseur Javascript. Il y a plusieurs manières d'exécuter du code Dart ; soit en utilisant un navigateur qui supporte directement ce code, soit en le compilant. En plus du navigateur Web et de la conversion javascript, le code peut être exécuté en ligne de commande, hébergé dans une machine virtuelle ce qui permet au client et au serveur d'avoir les applications écrites dans le même langage. [6]

3. Hot Reload :

La fonctionnalité Hot Reload fonctionne en injectant des fichiers de code source mis à jour dans la machine virtuelle Dart (VM) en cours d'exécution. Une fois que la machine virtuelle a mis à jour les classes avec les nouvelles versions de champs et de fonctions, le framework Flutter reconstruit automatiquement l'arborescence des widgets, vous permettant de visualiser rapidement les effets de vos modifications. Le build des applications est très rapide, ce qui rend quasiment invisible le temps de compilation. Un gain de temps pour les développeurs .

2.2.2 Pourquoi utiliser flutter ?

-Multi-plateforme (Cross-platform)

Flutter permet d'avoir une seule base de code pour tous les systèmes d'exploitation choisis. Puisque tout dans Flutter est un widget, le code est un balisage.

-Compatibilité avec toutes les version Android/iOS

La prise en charge des applications mobiles par tous les appareils et toutes les versions du système d'exploitation est devenue un gros problème.

Ce problème a été résolu par Flutter qui a son propre moteur et des widgets pris en charge à la fois par les composants matériels pour android et Cupertino pour IOS. Les applications développées avec flutter sont pris en charge depuis la version IOS 8 et Android jelly Bean jusqu'à la dernière version.

-Performance et stabilité

Le code de Flutter est compilé dans le code ARM du processeur. Avec son propre moteur de rendu, les applications Flutter ne sont affectées par aucune mise à jour du système d'exploitation ou personnalisation du système. Ils auront toujours la même apparence en termes d'interface même après une mise à jour du système iOS ou Android.

La compatibilité des versions est un autre aspect de l'influence de la stabilité. En tant que boîte à outils en croissance rapide, Flutter ne change pas son API et ses approches de développement. Le code écrit il y a deux ans pourrait être réutilisé dans les applications nouvellement créées.

2.3 Firebase

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application, 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. [13]



FIGURE 2.4 – Logo de Firebase

2.3.1 Les services de Firebase

La plateforme Firebase peut fournir trois types de services :

1 - Construction des meilleures applications :

 **Cloud Firestore** Permet de stocker et synchroniser les données entre utilisateurs et appareils - à l'échelle mondiale - à l'aide d'une base de données NoSQL hébergée dans le cloud. Cloud Firestore offre donc une synchronisation en direct et une assistance hors ligne, ainsi que des requêtes de données efficaces. Son intégration avec d'autres produits Fi-rebase permet de créer de véritables applications sans serveur.

 **Kit ML BETA** Ce service Apporte de puissantes fonctionnalités d'apprentissage automatique au applications mobile. Firebase ML aide à déployer des modèles ML personnalisés optimisés pour l'inférence sur l'appareil, ce qui réduit la taille d'installation initiale de l'application et permet d'effectuer plus facilement des mises à jour. on peut également utiliser AutoML Vision Edge pour entraîner nos propres modèles de classification d'images personnalisés ou accéder aux API Cloud AI Vision pour une solution plus clé en main.

 **Fonctions cloud** Grace a ce service on peut devlopper les applications avec un code backend personnalisé sans avoir à gérer et à mettre à l'échelle nos propres serveurs. Les fonctions peuvent être déclenchées par des événements, qui sont émis par les produits Firebase, les services Google Cloud ou des tiers, à l'aide de webhooks.

 **Authentification** Ce service permet de gérer les utilisateurs de manière simple et sécurisée. Firebase Auth propose plusieurs méthodes pour s'authentifier, y compris l'e-mail et le mot de passe, des fournisseurs tiers comme Google ou Facebook, et en utilisant directement le système de compte existant.

 **Hebergement** Avec ce service Firebase ,l'hébergement est facile avec grace outils spécialement conçus pour les applications Web modernes. Lorsqu'on telecharge nos ressources Web, ces dernières sont automatiquement transférées vers le CDN mondial et leur fournissent un certificat SSL gratuit afin que les utilisateurs de l'application bénéficient d'une expérience sécurisée, fiable et à faible latence, où qu'ils se trouvent.

 **Stockage en ligne** Grace a Firebase on peut Stocker et partager du contenu généré par les utilisateurs de notre application, comme des images, de l'audio et de la vidéo avec un stockage d'objets puissant, simple et économique conçu pour l'échelle de Google. Les SDK Firebase pour Cloud Storage ajoutent la sécurité de Google aux

importations et aux téléchargements de fichiers pour nos applications Firebase, quelle que soit la qualité du réseau.

 **Base de données en temps réel** Realtime Database est la base de données originale de Firebase. C'est une solution efficace et à faible latence pour les applications mobiles qui nécessitent des états synchronisés entre les clients en temps réel. Le service Cloud Firestore est fortement recommandé au lieu de Realtime Database pour la plupart des développeurs qui démarrent un nouveau projet.

2 - Amélioration de la qualité des applications :

 **Crashlytics** Permet de réduire le temps de dépannage en transformant une avalanche de plantages en une liste gérable de problèmes. On obtient donc des informations claires et exploitable sur les problèmes à résoudre en premier en voyant l'impact de l'utilisateur directement dans le tableau de bord Crashlytics. Crashlytics est donc le principal reporter de crash de Firebase.

 **Suivi des performances** Grâce à ce service on peut diagnostiquer les problèmes de performances des applications survenant sur les appareils de nos utilisateurs. Donc on peut rester au courant de l'heure de démarrage de notre application et surveiller les requêtes HTTP sans écrire de code.

 **Laboratoire de test** Avec ce service on peut exécuter des tests automatiques et personnalisés pour notre application sur des appareils virtuels et physiques hébergés par Google. On peut alors découvrir tous les bugs et les incohérences afin de pouvoir offrir une excellente expérience aux utilisateurs sur un large choix d'appareils.

 **Distribution d'applications BETA** Firebase App Distribution permet aux développeurs d'envoyer des versions préliminaires de leur application à des testeurs de confiance à partir de la console ou à l'aide d'outils de ligne de commande, ainsi que de gérer les testeurs en un seul endroit.

3 - Développement de l'entreprise :

 **Messagerie intégrée à l'application BETA** Donne le pouvoir de déclencher des messages en fonction du comportement et des intérêts des utilisateurs. On peut aussi personnaliser la conception des messages intégrés à l'application en fonction de notre marque. La messagerie intégrée prend en charge une variété de cas d'utilisation et de formats.

 **Google analytics** Grâce aux informations en temps réel à partir de rapports on peut analyser les attributions et le comportement des utilisateurs et aussi exporter nos données d'événement brutes vers Google BigQuery pour une analyse personnalisée.

 **Tests A/B** L'exécution des expériences produit et marketing nous permet d'améliorer des tests A / B. Grâce à ce service on peut personnaliser les tests en fonction de nos objectifs et tester diverses mises à jour de notre application, comme la copie de messages ou de nouvelles fonctionnalités.

 **Prédictions** La puissance de l'apprentissage automatique de Google est exploitée par ce service pour obtenir des informations sur les segments d'utilisateurs susceptibles de générer des revenus ou de dépenser (ou de terminer un autre événement de conversion). Ces segments prédictifs intelligents peuvent être utilisés pour cibler d'autres produits tels que Remote Config, Cloud Messaging et In-App Messaging.

 **Messagerie Cloud** Grace à ce service on peut envoyer gratuitement des messages et des notifications aux utilisateurs sur toutes les plates-formes (Android, iOS et le Web). Les messages peuvent être envoyés à des appareils uniques, à des groupes d'appareils ou à des sujets ou segments d'utilisateurs spécifiques. Firebase Cloud Messaging (FCM) s'adapte même aux applications les plus volumineuses, fournissant des centaines de milliards de messages par jour.

 **Configuration à distance** Grace à ce service on peut personnaliser le rendu de votre application pour chaque utilisateur, Modifier l'apparence, déployer les fonctionnalités progressivement, exécuter des tests A / B, fournir du contenu personnalisé à certains utilisateurs ou effectuer d'autres mises à jour sans déployer une nouvelle version. On peut aussi surveiller l'impact des modifications et effectuer des ajustements en quelques minutes.

 **Liens dynamiques** L'utilisation des liens dynamiques offre une expérience utilisateur personnalisée pour iOS, Android et le Web. On les utiliser pour alimenter le Web mobile afin de générer des conversions d'applications natives, un partage d'utilisateur à utilisateur, des campagnes sociales et marketing, etc. Les liens dynamiques fournissent les attributions dont on a besoin pour mieux comprendre notre croissance mobile.

2.3.2 Pourquoi utiliser Firebase ?

Sachant que la majorité des la communauté des développeurs considère Firebase comme le Meilleur service BaaS (backend as a service) et l'utilise dans la majorité de leurs projets , il existe d'autre alternatives , parmi les alternatives on a :

- La construction manuelle du service de gestion de l'infrastructure Backend de notre application.

Parse

Parse est un service BaaS développé par Facebook. Parse est un serveur Open Source, il offre donc un contrôle total du code et on peut ajouter nos propres fonctions et éléments. Cependant l'implémentation du service dans l'application reste très difficile .

- Il existe d'autres alternatives comme Back4App, Backendless, Kinvey, AWS Amplify, etc.

Cependant, on a trouvé que Firebase est la meilleure solution à utiliser pour notre projet pour les raisons suivantes :

- La documentation est assez solide contrairement à Parse et les autres alternatives.
- La facilité d'implémentation des services d'authentification comme Google, Facebook et Github.



FIGURE 2.5
– Logo de Parse

- La facilité de l'intégration avec Flutter car Flutter et Firebase travaillent main dans la main pour nous aider à créer des applications mobiles en un temps record.

- La flexibilité de la tarification des services Firebase. On peut commencer gratuitement, puis on paye au fur et à mesure selon nos besoins. [3]

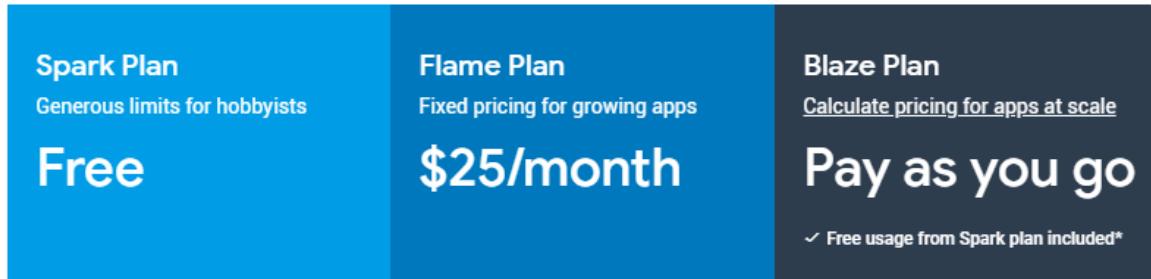


FIGURE 2.6 – Exemple de la tarification d'une partie des services Firebase

2.4 Google APIs

Les API Google sont des interfaces de programmation d'application (API) développées par Google qui permettent la communication avec les services Google et leur intégration à d'autres services. Les exemples de ceux-ci incluent la recherche, Gmail, la traduction ou Google Maps. Les applications tierces peuvent utiliser ces API pour tirer parti ou étendre les fonctionnalités des services existants.

Les API fournissent des fonctionnalités telles que l'analyse, l'apprentissage automatique en tant que service (l'API de prédiction) ou l'accès aux données utilisateur (lorsque l'autorisation de lire les données est donnée). Un autre exemple important est une carte Google intégrée sur un site Web, qui peut être réalisée à l'aide de l'API Static Maps, Places API ou de l'API Google Earth . [14]



FIGURE 2.7 – Logo de Google APIs

Dans ce projet nous avons utilisé deux APIs : Google Maps et Google Places dont on va en parler dans la section suivante.

2.4.1 Google Maps

Google Maps est un service de cartographie web, ce service disponible sur ordinateur, tablette et smartphone qui permet, à partir de l'échelle mondiale, de zoomer jusqu'à l'échelle d'une habitation. Plusieurs types de vue sont disponibles dans Google Maps : une vue en plan cartographique classique, avec les noms des rues, quartiers, villes et une vue en images satellites ou photographies aériennes, qui couvre aujourd'hui le monde entier, une vue en images obliques couvrant les grandes villes du monde et certaines villes secondaires et une vue avec le relief. [5]

Avec le SDK Maps pour Android, on peut ajouter des cartes basées sur les données de Google Maps à notre application. L'API gère automatiquement l'accès aux serveurs Google Maps, le téléchargement des données, l'affichage de la carte et la réponse aux gestes de la carte. On peut également utiliser les appels d'API pour ajouter des marqueurs, des polygones et des superpositions à une carte de base et pour modifier la vue de l'utilisateur d'une zone de carte particulière. Ces objets fournissent des informations supplémentaires sur les

emplacements de la carte et permettent à l'utilisateur d'interagir avec la carte. L'API nous permet d'ajouter ces graphiques à une carte :

- Icônes ancrées à des positions spécifiques sur la carte (marqueurs).
- Ensembles de segments de ligne (polylinques).
- Segments fermés (polygones).
- Graphiques bitmap ancrés à des positions spécifiques sur la carte (superpositions au sol).
- Ensembles d'images qui sont affichés au-dessus des tuiles de la carte de base.

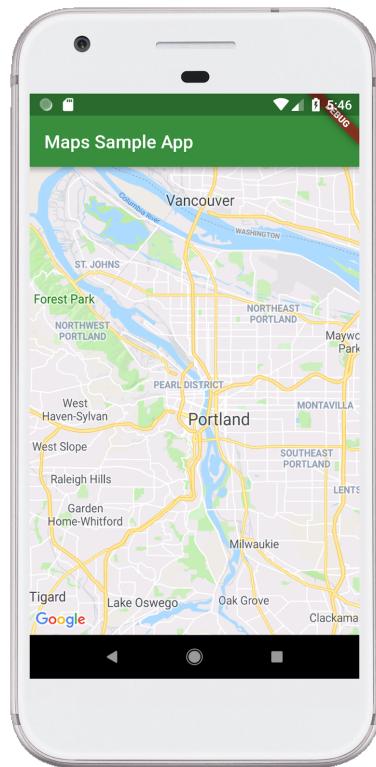


FIGURE 2.8 – Exemple de l'intégration des services Google Maps dans une application mobile

Chapitre 3

Conception et implémentation de l'application

3.1 Introduction

Après avoir étudié les différents techniques utilisés pour le développement des applications mobiles, on va entamer dans ce chapitre la partie conception et implémentation de notre application Android en fixant les objectifs de cette application suivie par une étude comparative sur des solutions déjà existantes, ensuite on va présenter les diagrammes UML ainsi que le diagramme d'action, puis les maquettes fonctionnelles et les interfaces utilisateur. À la fin on termine avec les outils matériels et logiciels utilisés au long du cursus du développement de l'application.

3.2 Conception

3.2.1 Objectif du projet

Ce projet a pour but de développer une application mobile sous la plateforme Android. qui permettra d'afficher dans une carte géographique des restaurants qui sont à proximité de l'utilisateur avec les informations de ces derniers ainsi que les menus qui lui seront accessible tous ça a pour but de rendre la recherche sur les restaurants et les menus plus facile et efficace .

3.2.2 Applications déjà existantes

Il existe peu d'applications mobiles dans le domaine de la restauration nous allons explorer certaines des plus populaires solutions existantes aujourd'hui.

TripAdvisor

TripAdvisor est une plate-forme disponible en web et en application mobile, cette plate-forme offre des avis et des suggestions d'hôtels, lieux de loisirs, villes et régions à l'échelle internationale, compare les prix et affiche les meilleures offres du moment.

TripAdvisor couvre aussi les services de restauration et fournit des informations sur les différents restaurants qui existent dans le monde entier.

Le service qu'offre cette plate-forme pour le secteur de la restauration a des avantages :

- L'utilisation de ce service est gratuite.
- Donne un aperçu sur les restaurants et offre des informations sur la fourchette des prix..
- Des évaluations et des avis sont fournis par les utilisateurs eux-mêmes.

TripAdvisor a ses avantages mais aussi ses inconvénients :

- On ne peut pas parcourir les restaurants en utilisant une carte géographique.
- L'utilisation d'une application tierce est indispensable pour avoir l'itinéraire vers les restaurants.
- Les informations offertes par cette plate-forme sont limitées au avis fourni par les utilisateurs.

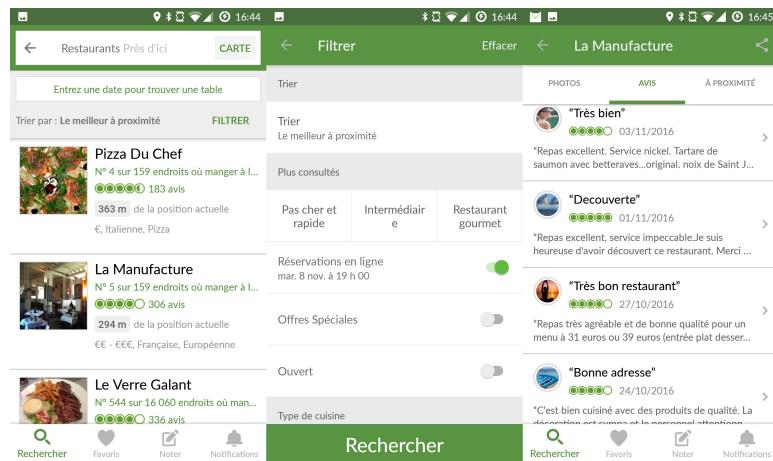


FIGURE 3.1

– Logo de TripAdvisor

FIGURE 3.2 – Interface de TripAdvisor lors de la recherche des restaurants et les avis

Google Maps

Google maps est un service de cartographie développé par l'entreprise Google et lancé en 2005. Ce service offre différents types de vues et permet aussi d'afficher les différents points d'intérêts dans le monde comme les hôtels, les lieux d'attractions, etc. Ce qui nous intéresse dans notre domaine de recherche est la géolocalisation des restaurants, une fonctionnalité qu'offre ce service.

En accédant au site web ou l'application mobile, l'utilisateur aura accès à un large choix de restaurants proposés grâce à un système de suggestions, aussi développé par Google. L'utilisateur pourra donc accéder aux informations des restaurants ainsi que des photos et des avis généralement postés par la communauté des utilisateurs de Google Maps.



FIGURE 3.3

– Logo de Google Maps

Google Maps présente de nombreux avantages, nous en citerons :

- L'utilisation de ce service est gratuite.
- La plupart des restaurants sont affichés dans la carte.
- Les images des restaurants sont parfois affichées.
- Les utilisateurs peuvent donner des avis et des notes.
- Des itinéraires précis sont proposés.

Néanmoins ce service présente aussi des inconvénients :

- Les menus détaillés ne sont pas affichés dans l'interface.
- On ne peut pas filtrer les types de restaurants.

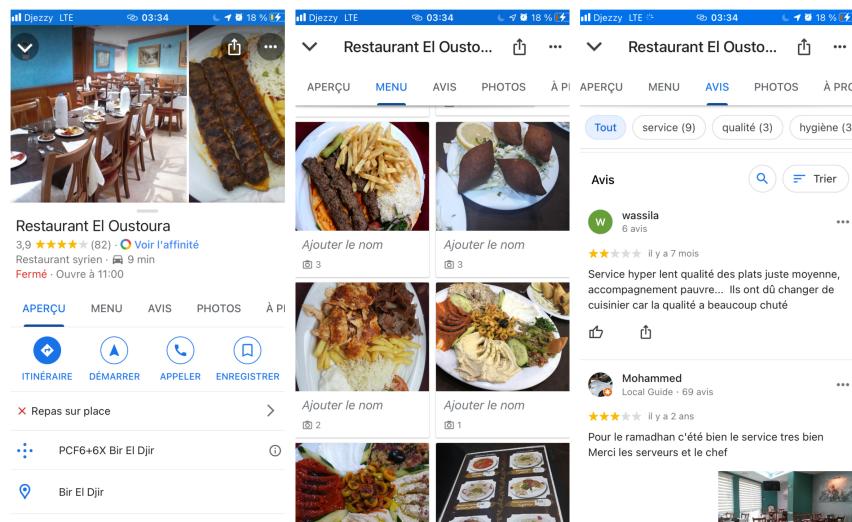


FIGURE 3.4 – Exemple de différents écrans lors d'une recherche d'un restaurant

LaFourchette

TheFork de son ancien nom "LaFourchette" est une plate-forme française de réservation de restaurants en ligne, accessible par un site web et par une application mobile disponible sur Android et iOS. Elle est présente dans plus de vingt pays en Europe et en Amérique latine. [17]



TheFork a ces avantages et ces inconvénients, parmi ces avantages on a :

- La base de données regroupe plus de 80000 restaurants à travers le monde. [7]

- L'application est multiplate-forme et disponible en version web et mobile.

- L'interface utilisateur est simple qui permet d'utiliser le service de manière efficace.

- La présence de la fonction d'exploration des restaurants via une carte.

Quant aux Inconvénients nous citeront :

- Le service n'est pas totalement gratuit.

- Cette application est dédiée qu'aux réservations de restaurants et ne donne pas des informations détaillées sur les menus.

- L'application ne prends pas encore en charge les restaurants algériens.

FIGURE 3.5 –
Logo de LaFourchette

3.2.3 Les diagrammes UML

Le langage UML est un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent.

L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l'analyse et la conception orientées objet.

Pour bien expliquer le fonctionnement de notre application et modéliser les diagrammes. On a utilisé l'outil Visual Paradigm Online qui est facile à manipuler afin de réaliser ces diagrammes.

Diagrammes de séquence

Enchainement des actions pour le lancement de l'application

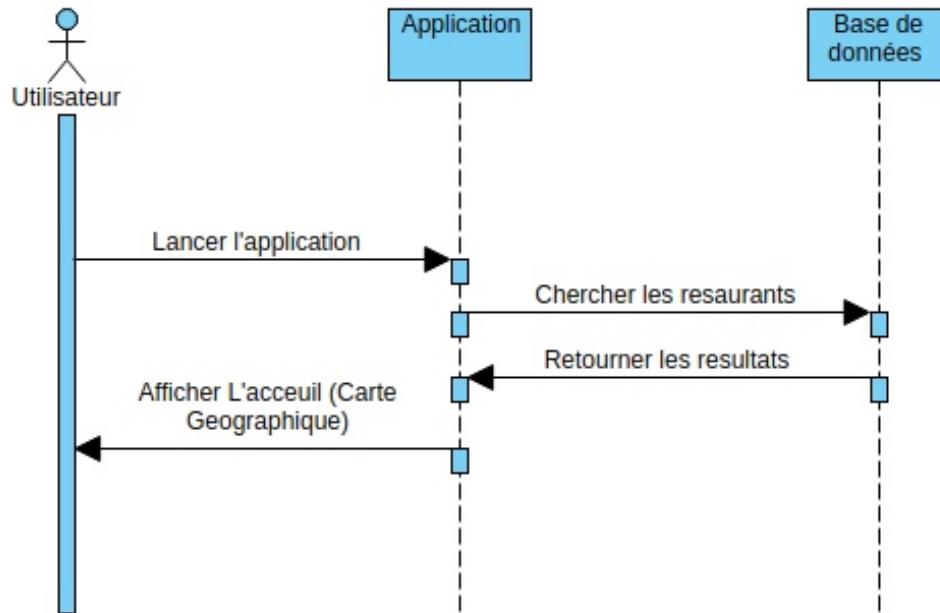


FIGURE 3.6 – Enchainement des actions pour le lancement de l'application

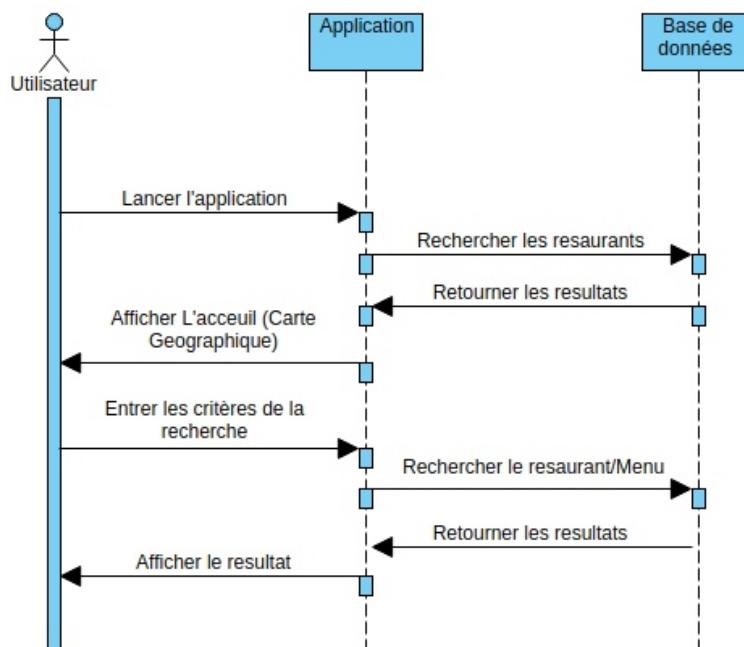


FIGURE 3.7 – Enchainement des actions pour la recherche des restaurants

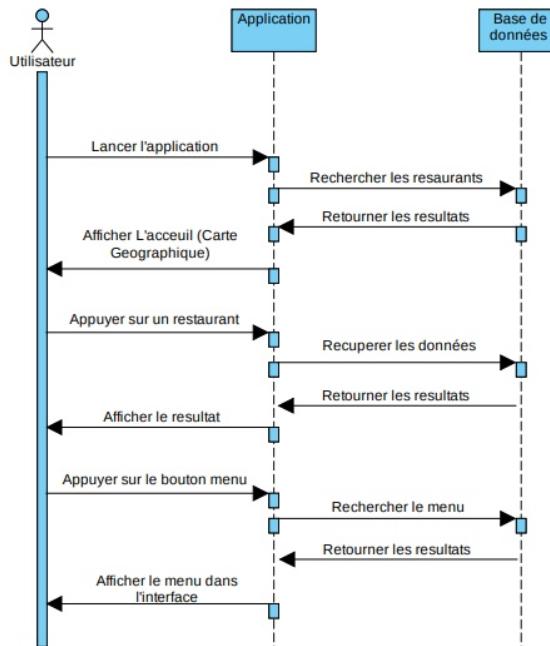


FIGURE 3.8 – Enchainement des actions pour l'affichage de la page restaurant et son menu

Diagramme d'action

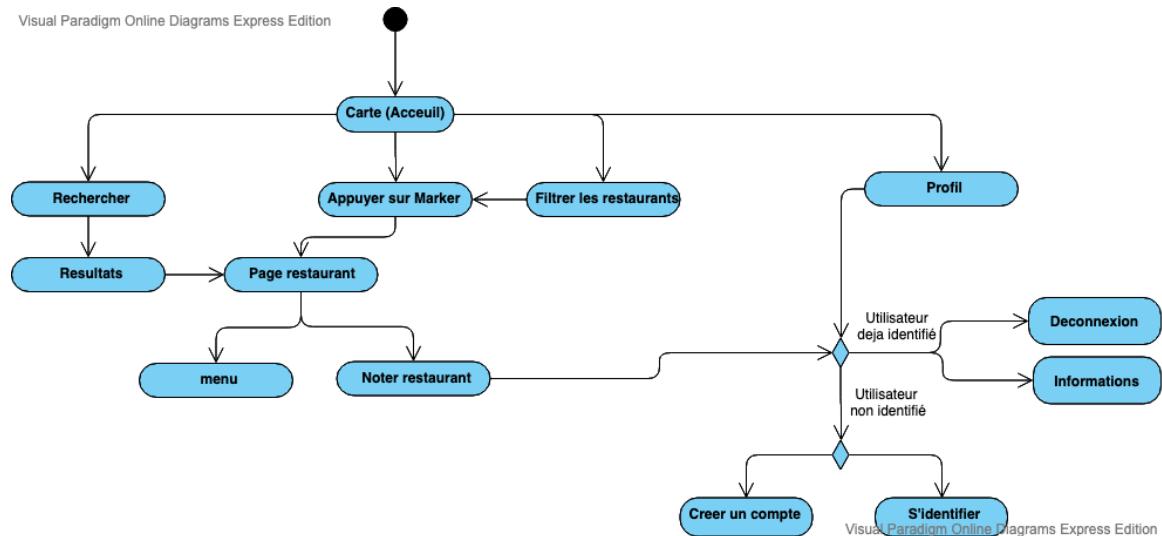


FIGURE 3.9 – Diagramme d'actions

3.2.4 Maquette fonctionnelle (Wireframe)

Définition

La maquette fonctionnelle ou Wireframe est un schéma utilisé lors de la conception des interfaces des applications pour définir les emplacements des zones de texte, des images, des vidéos, des liens, ainsi que des différents éléments graphiques.

Outil utilisé

Balsamiq Wireframes est une application de création d'interface utilisateur graphique ,elle permet au concepteur d'organiser des widgets prédéfinis à l' aide d'un éditeur WYSIWYG par glisser- déposer .L'application est proposée dans une version de bureau ainsi qu'un plug-in pour Google Drive [12].



FIGURE 3.10 – Logo de Balsamiq

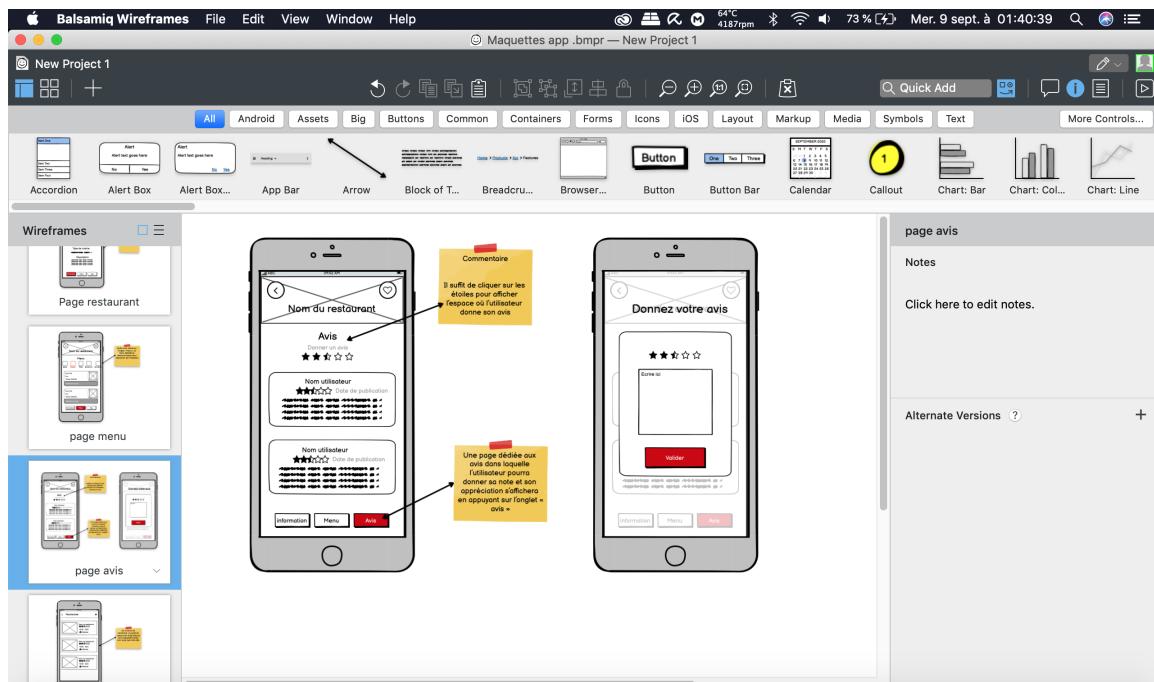


FIGURE 3.11 – Interface du logiciel Balsamiq

Les différents écrans



FIGURE 3.12 – Ecran principal

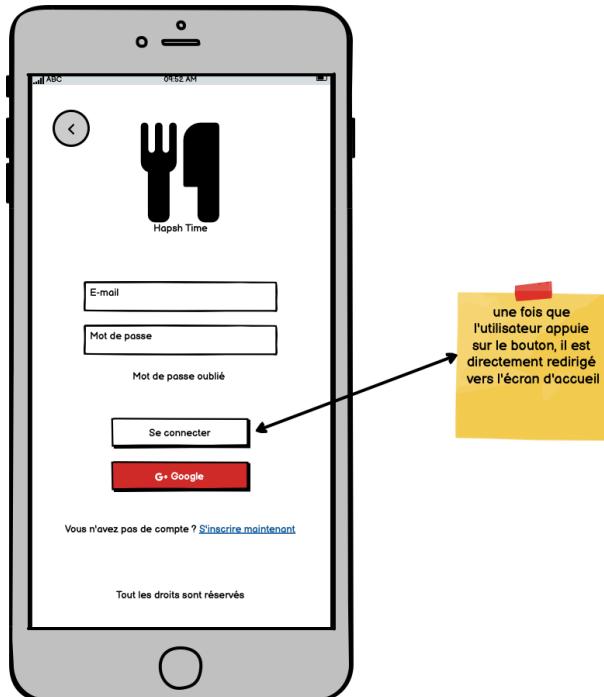


FIGURE 3.13 – Ecran de connexion

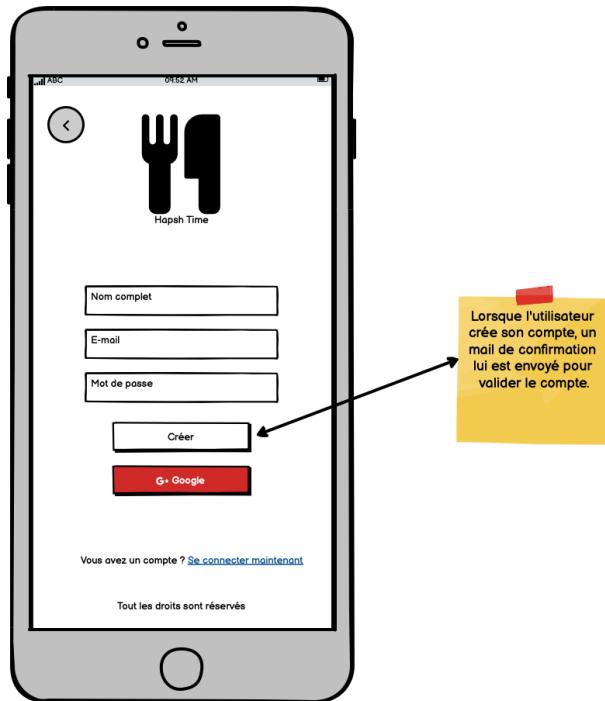


FIGURE 3.14 – Ecran de creation du compte

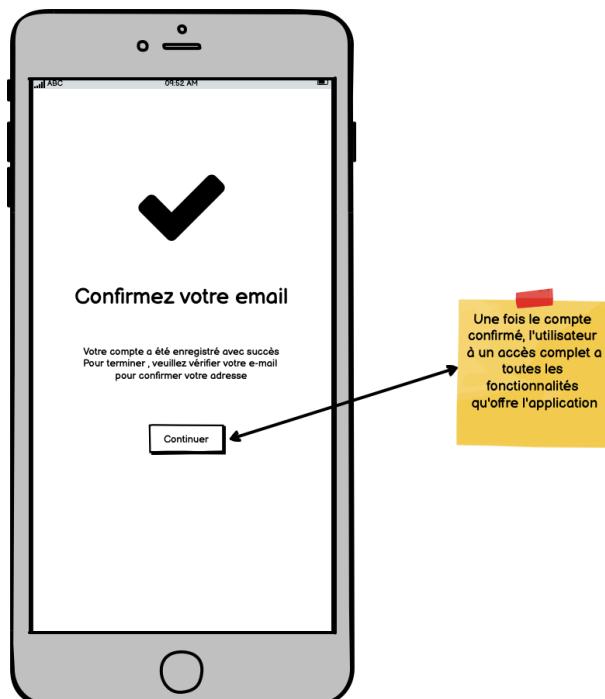


FIGURE 3.15 – Ecran de confirmation du compte



FIGURE 3.16 – Ecran de réinitialisation du mot de passe



FIGURE 3.17 – Ecran d'affichage de la fenêtre du restaurant

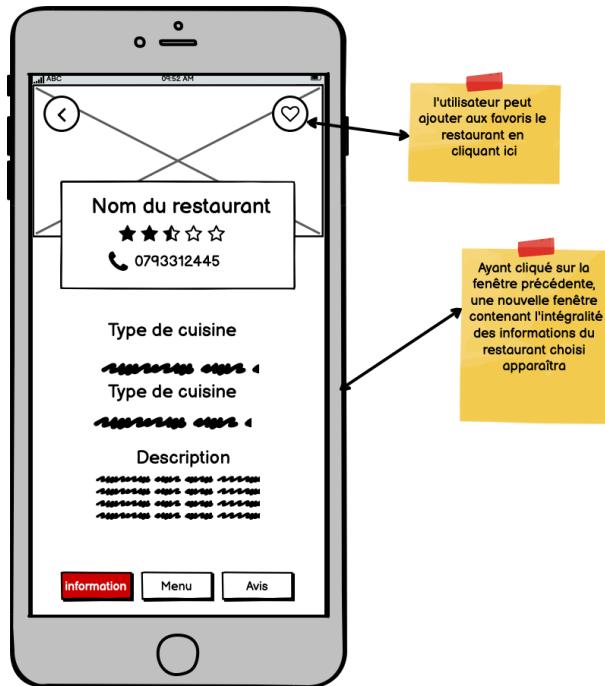


FIGURE 3.18 – Page de détails du restaurant

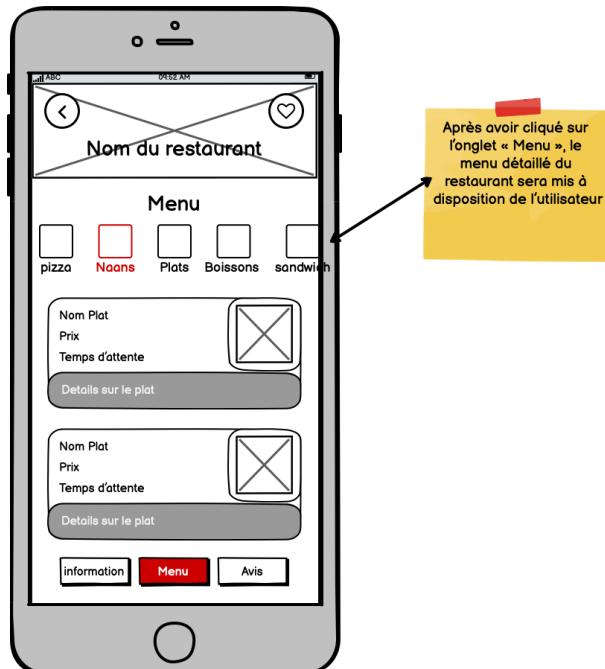


FIGURE 3.19 – Page de détails du menu

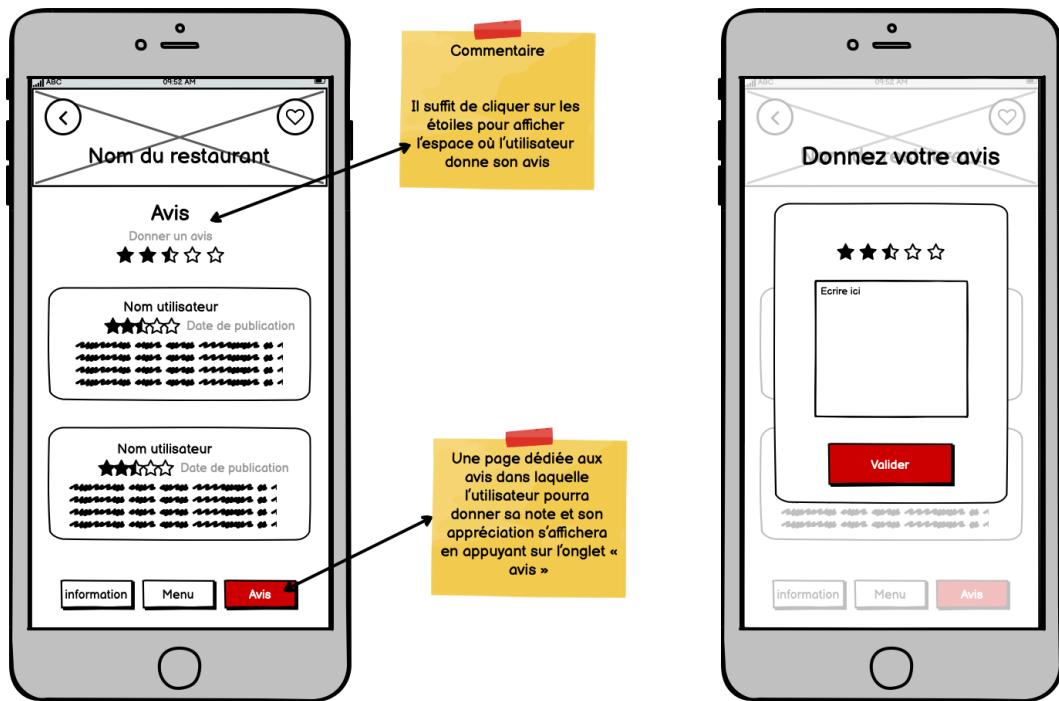


FIGURE 3.20 – Page d'avis

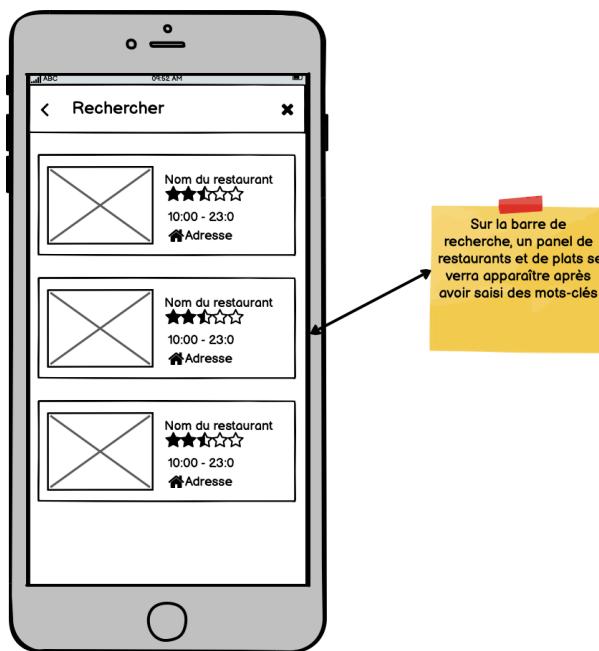


FIGURE 3.21 – Page de recherche

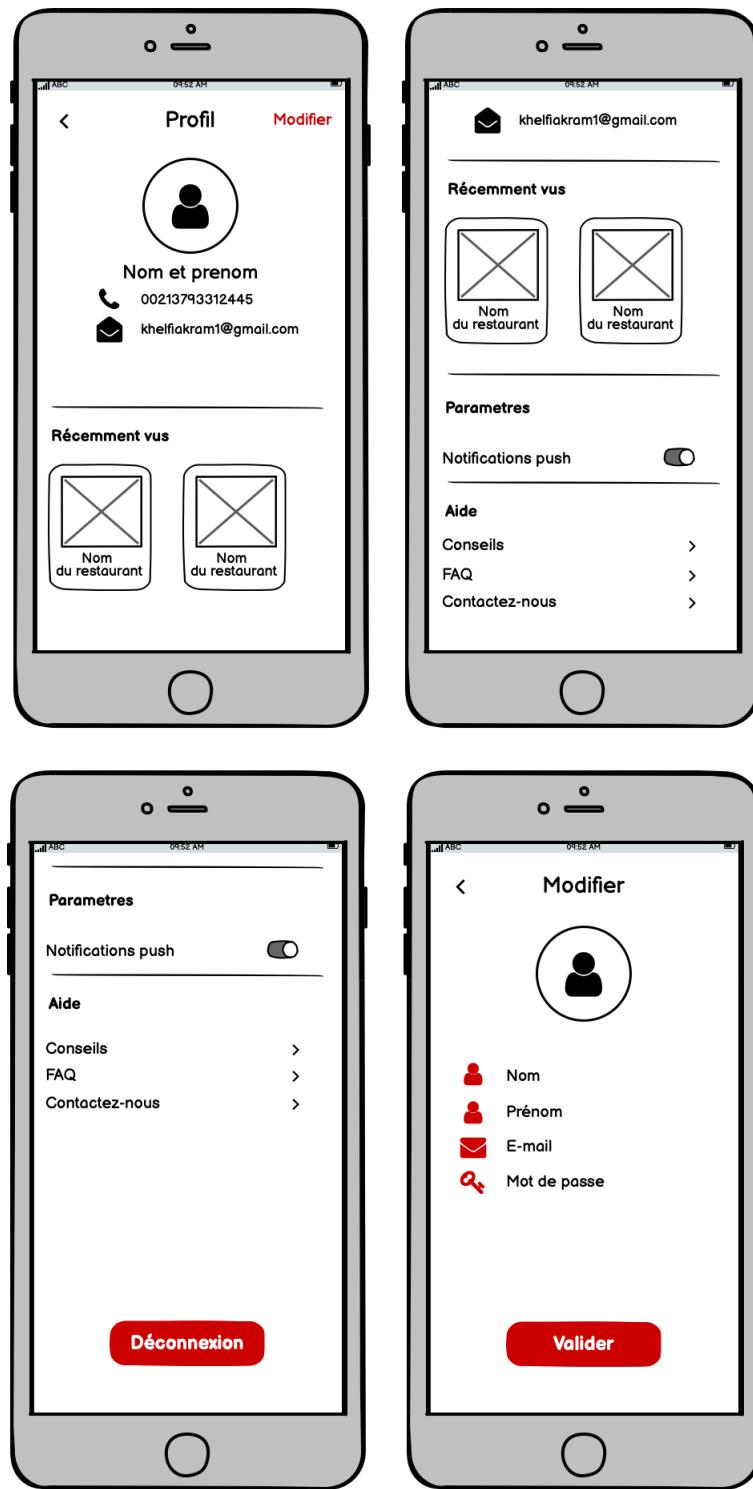


FIGURE 3.22 – Page du profil et paramètres

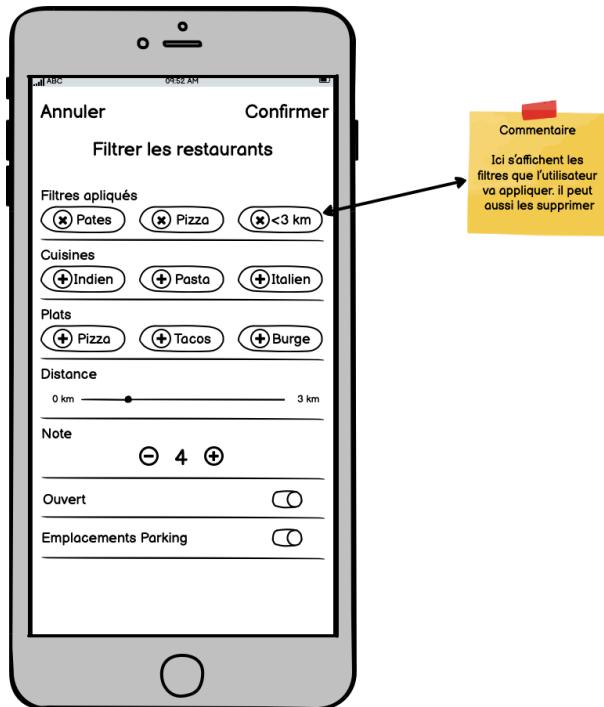


FIGURE 3.23 – Page du filtre

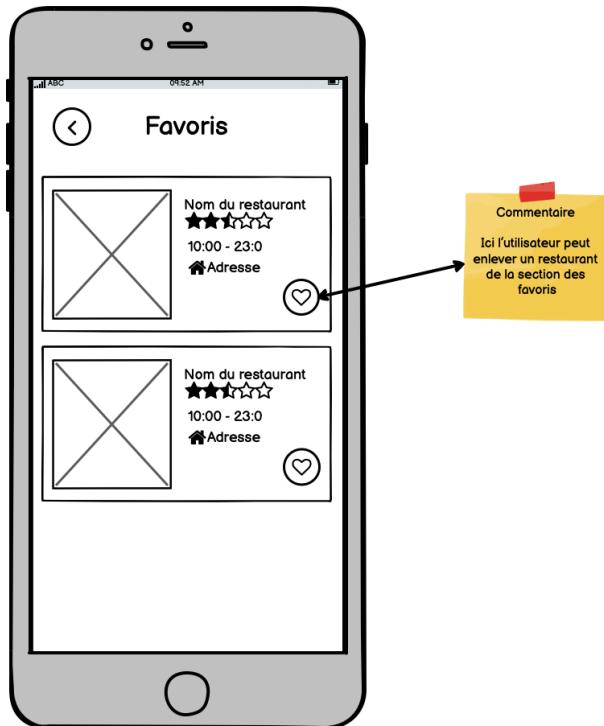


FIGURE 3.24 – Page des restaurants favoris

3.2.5 Prototypage et interface utilisateur

La phase de Prototypage et de la création de l'interface utilisateur est une étape très importante dans le développement des applications mobiles. pour ce projet on a utilisé les règles de design appelées le material design de Google qui est un design très présent dans les applications mobiles en ce temps.

Material design

Le Material Design est un ensemble de règles de design proposées par Google et qui s'appliquent à l'interface graphique des logiciels et applications . Il est utilisé notamment à partir de la version 5.0 (Lollipop) du système d'exploitation Android.

Google a présenté le Material Design pour la première fois lors de la conférence Google I/O, le 25 juin 2014. Ces règles de design mettent l'accent sur une utilisation accrue des mises en page basées sur une grille , des animations et des transitions, des effets de profondeur tels l'éclairage et les ombres. Selon Google ce nouveau langage de design est basé sur le papier et l'encre.

Le designer Matías Duarte explique que « contrairement au vrai papier, notre matériau numérique peut s'étirer et se modifier de manière intelligente. Le matériau contextuel a une surface physique et des bords. Les superpositions et les ombres donnent des informations sur ce que vous pouvez toucher ». [16]

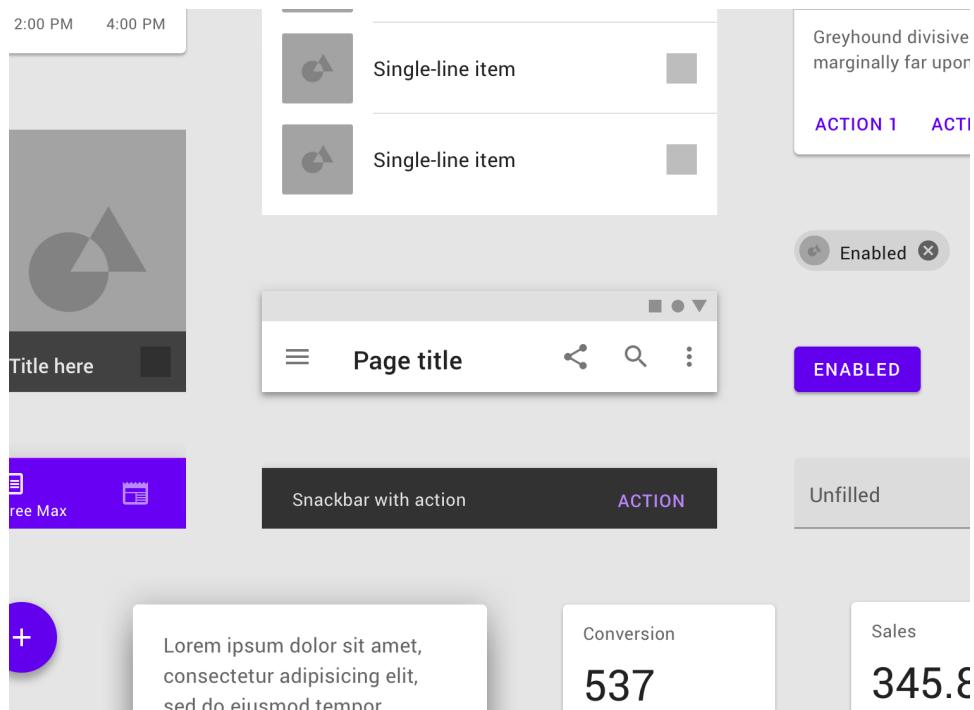


FIGURE 3.25 – Exemple de composantes du Material Design

Outil utilisé (Adobe XD)

Adobe XD est un outil de conception de l'expérience utilisateur pour les applications Web et des applications mobiles, développé et édité par Adobe Inc. Il est disponible pour macOS et Windows, bien qu'il existe des versions pour iOS et Android pour aider à prévisualiser le résultat du travail directement sur les appareils mobiles. XD prend en charge le wireframing de site Web et la création de prototypes de clic [10].



FIGURE 3.26 – Logo de Adobe XD

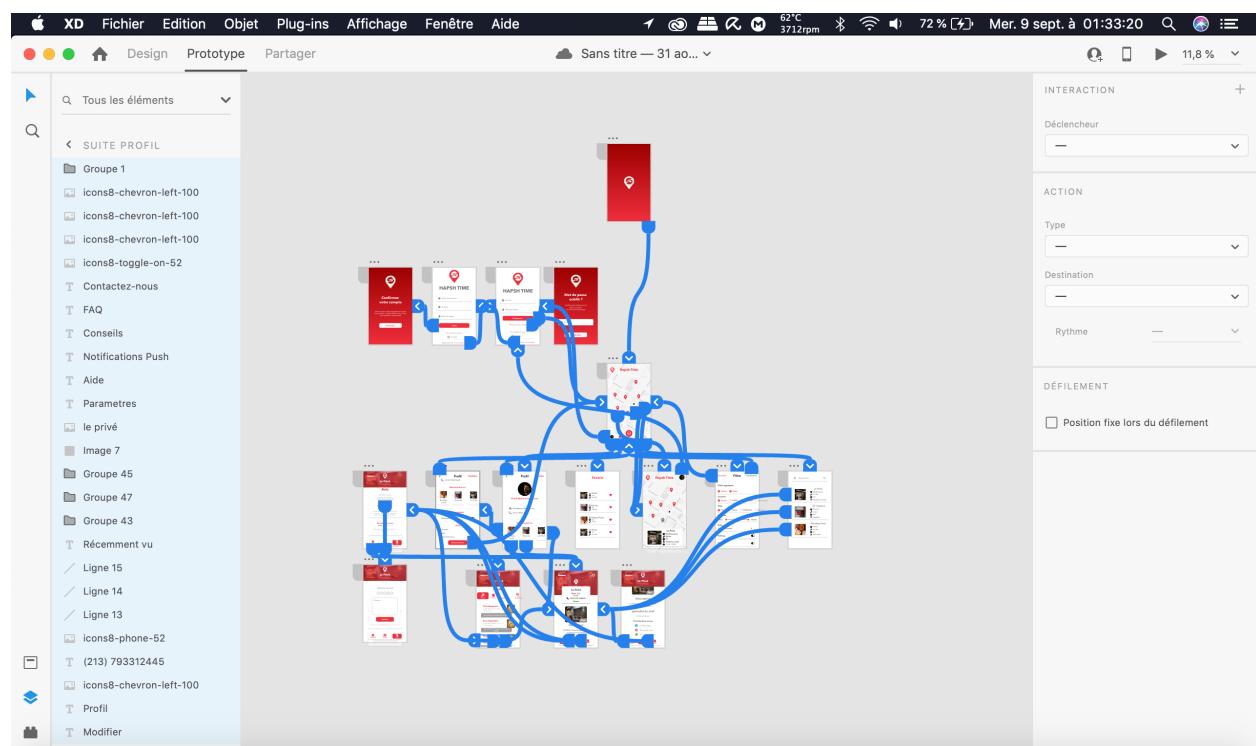
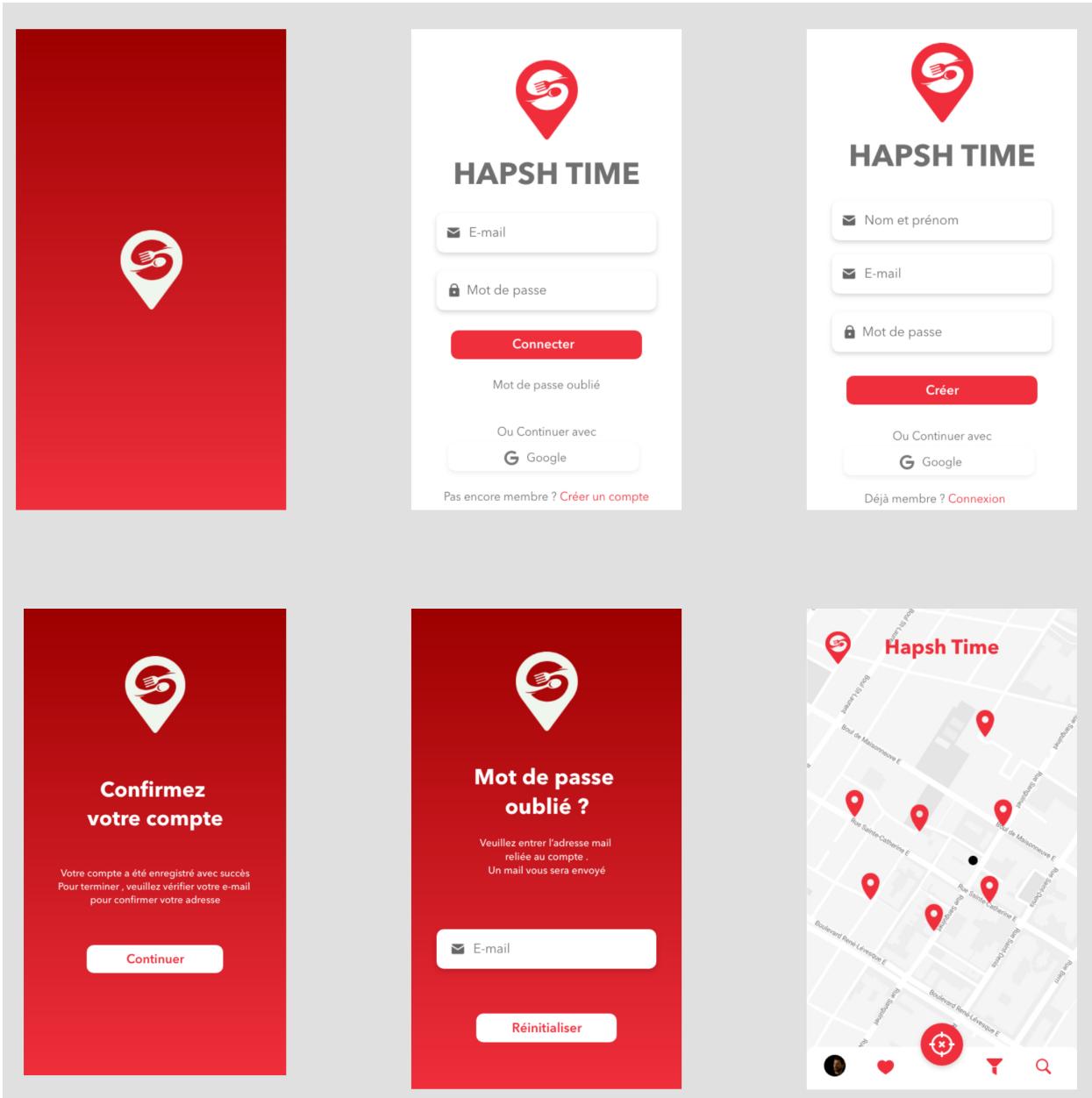


FIGURE 3.27 – interface du logiciel Adobe XD

Les Maquettes UI/UX



The screenshots illustrate the user interface of a mobile application for a restaurant named "Le Privé".

- Screenshot 1:** A map showing the location of "Le Privé" on "Rue Sainte-Catherine E" near "Boulevard René-Lévesque E". The restaurant's address is "Residence el ryad". It shows nearby landmarks like "Hapsh Time" and "Le Privé".
- Screenshot 2:** A detailed view of "Le Privé". It shows the restaurant's name, a photo of the exterior, and a summary: "Note : 4,2" (6 avis), "(213) 772 765423", and "Ouvert". Below this is another photo of the interior.
- Screenshot 3:** A detailed view of "Le Privé" with a red header. It includes a "Description" section with the text "LE PRIVÉ , Pizza Burger & Tacos comme vous en rêvez.", a "spécialité du chef" section with "Le French tacos", and a "Contactez nous" section with social media links and a phone number.
- Screenshot 4:** A menu screen titled "Menu". It features four categories: "Pizza" (with a pizza icon), "Plats" (with a plate icon), "Sandwich" (with a sandwich icon), and "Cock" (with a cocktail icon). Below these are three pizza options: "Pizza Margherita", "Pizza Napolitaine", and "Pizza Mexicaine". Each option includes a description, a photo, and a "Basilic" button.
- Screenshot 5:** An "Avis" (Reviews) screen titled "Avis". It shows a rating scale from 1 to 5 stars with "Khelfi med Akram" (4 stars) and "Belabbas Oussama" (4 stars). Each review includes a snippet and a timestamp ("il y a 5 jours" or "il y a une semaine").
- Screenshot 6:** A screen titled "Donnez un avis" (Leave a review). It features a 5-star rating scale, a text input field "Ecrire ...", and a "Valider" (Validate) button. At the bottom, there is a footer with "General", "Menu", and "Avis" buttons.

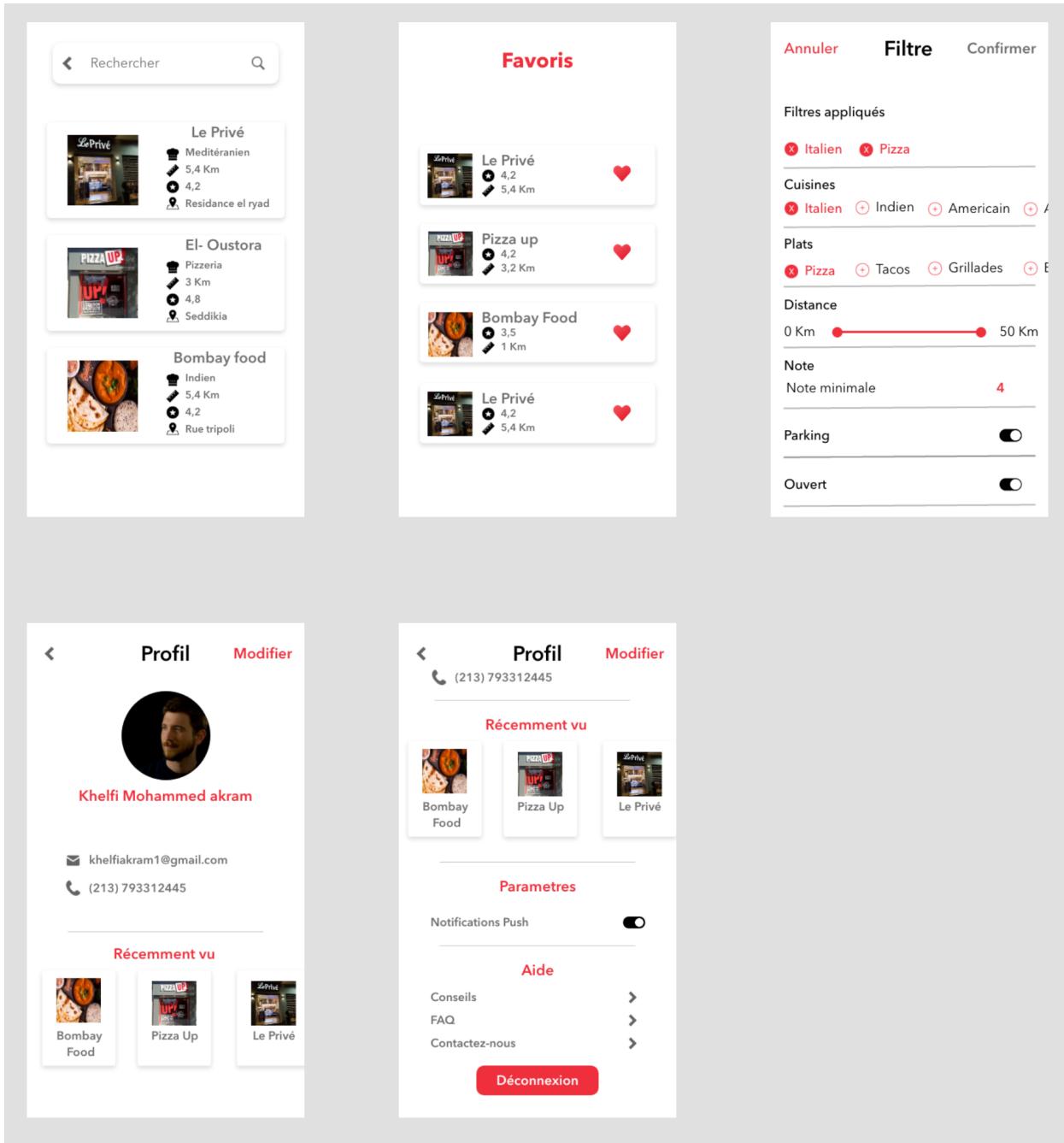


FIGURE 3.28 – Interfaces utilisateur de l'application

3.3 Implementation

3.3.1 Outils Materiels

Le développement de cette application a été effectué sur deux machines en parallèle avec les configurations suivantes :

Machine principale

- Marque : Lenovo Thinkpad
- Système d'exploitation : Windows 10
- Microprocesseur : Intel Core i5
- Mémoire vive : 8 Go
- Disque dur : 256 GO SSD

Machine secondaire

- Marque : Dell .
- Système d'exploitation : Ubuntu 19.10.
- Microprocesseur : Intel Core i5.
- Mémoire vive : 8 Go.
- Disque dur : 1 To HDD+ 480 SSD.

Les tests de cette application ont été effectués sous un Emulateur Android PIXEL 3 et un Smartphone Galaxy j3 Pro sous Android 9

3.3.2 Outils Logiciels

Visual Studio Code

Visual studio code est un éditeur de code cross-platform, open source et gratuit, supportant une dizaine de langages développés par Microsoft pour Windows, Linux et macos. Il facilite grandement le codage grâce à ses capacités à comprendre le code et aussi la navigation facile entre les différentes parties de code.



FIGURE 3.29 – Logo de Visual studio code

The screenshot shows the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Debug, Terminal, and Help. The title bar says "app.dart - pics - Visual Studio Code". The left sidebar has sections for Explorer, Open Editors, and PICS. The Explorer section shows files like main.dart, app.dart, image_list.dart, and image_model.dart. The Open Editors section shows multiple tabs for main.dart, app.dart, image_list.dart, and image_model.dart. The main workspace shows Dart code for an AppState class. The bottom status bar shows file paths, line numbers, and other development details.

FIGURE 3.30 – Interface du logiciel Visual Studio Code

3.3.3 Développement

Base de données

Pour la base de données nous avons choisi d'utiliser le service de base de données fourni par firebase avant d'aller plus dans le détail définissons firebase.

Firebase Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (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. [13]



FIGURE 3.31 – Logo de Firebase

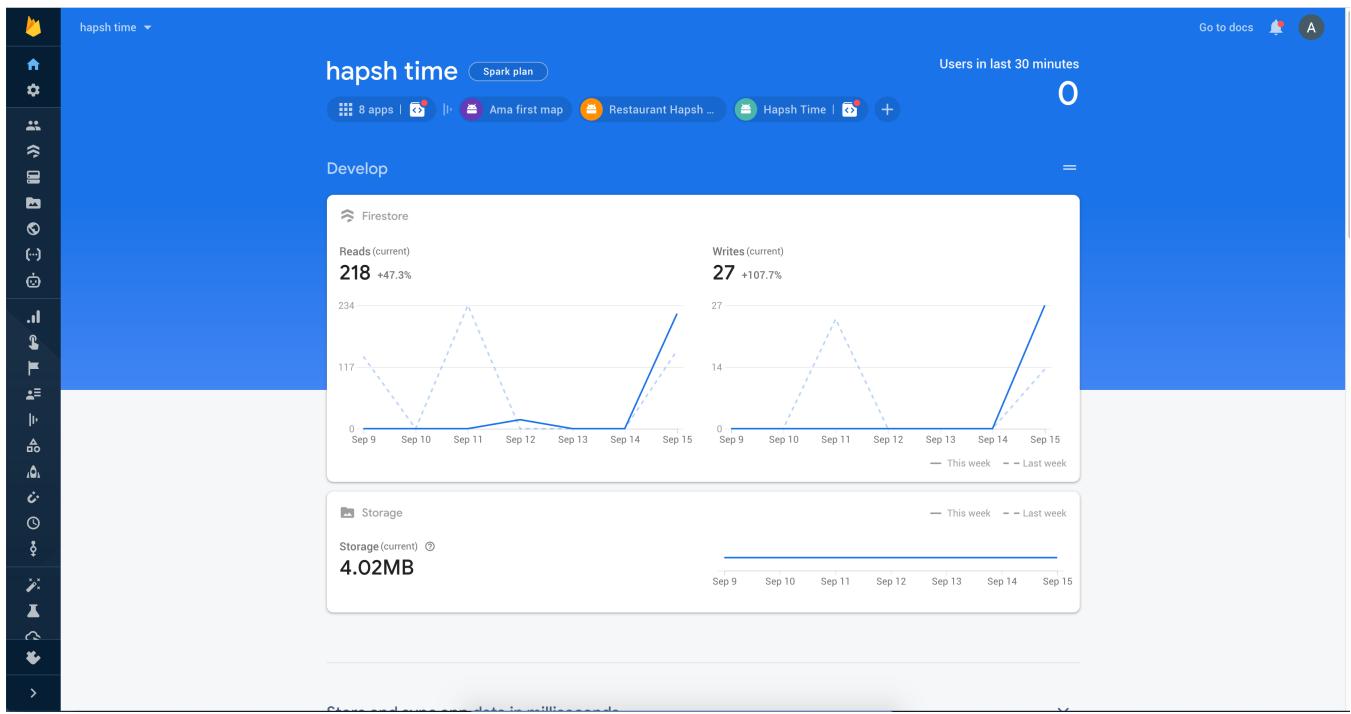


FIGURE 3.32 – Interface de la console de Firebase

Les packages utilisés :**Google maps flutter :**

Avec le plugin Google Maps Flutter, on peut ajouter des cartes basées sur les données de Google Maps à notre application. Le plugin gère automatiquement l'accès aux serveurs Google Maps, l'affichage de la carte et la réponse aux gestes de l'utilisateur tels que les clics et les traînées. On peut également ajouter des marqueurs à notre carte. Ces objets fournissent des informations supplémentaires sur les emplacements de la carte et permettent à l'utilisateur d'interagir avec la carte.

Cloud Firestore :

Ce package permet d'intégrer les services de base de données NoSQL de Firebase pour l'utilisation en temps réel. Les données sont stockées dans l'arborescence JSON sous la forme des collections, où chaque collection un nombre infini de documents, ces derniers contiennent les données de la base de données.

Geolocator :

Geolocator est un plugiciel de géolocalisation dédié à Flutter, ce dernier offre un accès facile au service de localisations spécifiques à la plateforme.

Geolocator permet de :

- Obtenir le dernier emplacement de l'utilisateur.
- Obtenir l'emplacement actuel de l'appareil.
- Vérifier si les services de localisation sont activés sur l'appareil.
- Calculer la distance (en mètres) entre deux points géographique.

Firebase auth :

Firebase auth est un plugin Flutter pour utiliser l'API d'authentification Firebase. Cet API permet de gérer les authentications des utilisateurs en offrant différentes options d'inscription et de connexion à leurs comptes.

Google sign in :

Google sign in est un plugin Flutter qui offre un système d'authentification sécurisé pour se connecter avec un compte Google sur Android et iOS.

Chapitre 4

Conclusion et perspectives

Dans ce rapport, nous avons exposé les étapes de conception et de développement de notre application qui consiste à créer une application mobile pour la géolocalisation des restaurants dans la ville d'Oran .

Notre travail s'est déroulé sur trois étapes. nous avons commencé par une étude des applications mobiles et différents systèmes d'exploitation

Dans la deuxième phase, nous avons abordé les différentes technologies utilisées, on a commencé par Flutter, ensuite Firebase et à la fin on a parlé des API de Google.

Dans la troisième étape , nous avons spécifié l'objectif du projet ainsi qu'une étude de l'existant. Ensuite on est passé à l'étape conception en utilisant des diagrammes UML et actions ainsi que les maquettes et interfaces utilisateur ,enfin l'implementation de notre code.

Ce projet se situe en effet , dans le cadre du projet de fin d'étude de licence en informatique. Ce projet était une véritable expérience de travail en collaboration, qui nous a permis de bien gérer la répartition des tâches et de renforcer l'esprit de partage de connaissances ainsi que la synchronisation de notre travail.

Cependant, nous pouvons encore améliorer cette application en créant une application web et aussi intégrer un système d'intelligence artificielle qui permet de proposer du contenu adapté à l'utilisateur .

Bibliographie

- [1] Android — Wikipédia. <https://fr.wikipedia.org/wiki/Android>.
- [2] Cycle de vie du développement d'une application mobile. <https://appinov.com/blog/article/cycle-de-vie-du-developpement-d'une-application-mobile>.
- [3] Firebase Pricing. <https://firebase.google.com/pricing/?gclid=CjwKCAjw19z6BRAYEiwAmo64Lfroa>
- [4] Flutter architectural overview. <https://flutter.dev/docs/resources/architectural-overview>.
- [5] Google Maps — Wikipédia. https://fr.wikipedia.org/wiki/Google_maps.
- [6] Le Dart | SUPINFO, École Supérieure d'Informatique. <https://www.supinfo.com/articles/single/6001-dart>.
- [7] Restaurants : le site de réservation LaFourchette change de nom pour TheFork (même en France). https://www.bfmtv.com/economie/consommation/restaurants-le-site-de-reservation-la-fourchette-change-de-nom-pour-the-fork-meme-en-france_AN-202002110099.html.
- [8] Système d'exploitation mobile — Wikipédia. https://fr.wikipedia.org/wiki/Syst%C3%A8me_d%27exploitation
- [9] Technical overview. <https://flutter.dev/docs/resources/technical-overview>.
- [10] Adobe XD. *Wikipedia*, September 2020.
- [11] Application mobile. *Wikipédia*, July 2020.
- [12] Balsamiq. *Wikipedia*, August 2020.
- [13] Firebase. *Wikipedia*, August 2020.
- [14] Google APIs. *Wikipedia*, August 2020.
- [15] iOS. *Wikipédia*, September 2020.
- [16] Material design. *Wikipédia*, February 2020.
- [17] TheFork. *Wikipédia*, September 2020.

Résumé