République algérienne démocratique et populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Mouloud MAMMERI de Tizi-Ouzou
Faculté de Génie électrique et Informatique
Département Informatique

Mémoire *de fin d'étude*

En vue de l'obtention d'un Master en Informatique

Thème

Conception et réalisation d'une application banque en ligne

Proposé et dirigé par :

Mr SI MOHAMMED

Réalisé par :

AIT ZEGGANE Belaid

OULMANE Hakim

Remerciements

Louange à Allah tout puissant qui nous a guidés pour l'accomplissement de ce modeste travail.

C'est un grand plaisir pour nous autant qu'un devoir de remercier toutes les personnes qui, de près ou de loin, nous ont aidés à mettre en terme ce mémoire.

Notre reconnaissance et nos vifs remerciements vont tout particulièrement à notre promoteur ***Mr SI MOHAMMED*** *pour son encadrement, ses nombreux conseils, ses compétences, ses permanentes disponibilités et ses encouragements jusqu'à la finalisation de ce travail.*

Nous tenons ensuite à remercier les membres du jury pour nous avoir fait l'honneur d'accepter de juger ce travail.

Un grand merci à nos familles qui nous ont apporté le soutien, la confiance et l'amour indéfectibles que nous espérons rendre pareillement et transmettre à notre tour.

Table des matières

INTRODUCTION GENERALE

CHAPITRE 1 : RESEAUX MOBILES

1. Introduction 14
2. Définition 15
3. Les défis des réseaux mobiles 15
4. Evolution des réseaux mobiles 16
   - 4.1 Première génération (1G) 16
   - 4.2 Deuxième génération (2G) 16
   - 4.3 Troisième génération (3G) 17
   - 4.4 Quatrième génération (4G) 18
   - 4.5 Cinquième génération (5G) 18
   - 4.6 Les réseaux de nouvelle génération (NGN) 19
5. Classification des réseaux sans-fil 20
   - 5.1 Les réseaux personnels sans-fil (WPAN) 20
   - 5.2 Les réseaux locaux sans-fil (WLAN) 20
   - 5.3 Les réseaux métropolitains sans-fil (WMAN) 21
     - 5.3.1 WiMAX 21
   - 5.4 Réseaux étendus sans-fil (WWAN) 21
     - 5.4.1 GSM 21
     - 5.4.2 GPRS 22
     - 5.4.3 UMTS 22
     - 5.4.4 LTE 22

6. Architecture des différents réseaux mobiles 23
6.1 Architecture du réseau 4G LTE 23
6.2 Fonctionnement d'un réseau LTE 26
6.2.1 Le contrôle d'information dans un réseau LTE 26
6.2.2 Les flux de données dans un réseau LTE 27
7. Comparatif des réseaux mobiles 28
8. Conclusion 30

CHAPITRE 2 : DEVELOPPEMENT D'APPLICATIONS MOBILES

1. Introduction 32
2. Définitions 33
2.1 Application 33
2.2 Développement applications mobiles 33
3. Type d'applications mobiles 33
3.1 Applications natives 33
3.1.1 Avantages 34
3.1.2 Inconvénients 34
3.2 Applications hybrides 34
3.2.1 Avantages 35
3.2.2 Inconvénients 35
3.3 Applications Web progressives 35
3.3.1 Avantages 36
3.3.2 Inconvénients 36
3.4 Tableau comparatif 37
4. Développement application Android 39
4.1 Définition 39
4.2 Caractéristiques 40

4.3 Architecture 41
5. Développement application iOS 42
5.1 Définition 42
5.2 Liste des versions 42
5.3 Les dispositifs mobiles compatibles 43
5.4 Caractéristiques 43
5.5 Architecture 43
5.6 Outils 45
6. Différences entre Android et iOS 45
6.1 Différences techniques 45
6.2 Différences marketing 46
7. Conclusion 47

CHAPITRE 3 : ANALYSE ET CONCEPTION

1. Introduction 49
2. Les objectifs de l'application 50
3. Banque en ligne 50
3.1 Définition 50
3.2 Services 51
4. Généralités sur UML 52
4.1 Les diagrammes d'UML 52
4.1.1 Diagrammes structurels ou statiques 52
5. Analyse 53
5.1 Définitions 54
5.1.1 Acteur 54
5.1.2 Cas d'utilisation 54
5.1.3 Scénario 54

5.2 Spécification des besoins....54
5.3 Identification des acteurs....55
5.4 Diagramme de contexte....56
5.5 Identification des cas d'utilisations....56
5.6 Diagramme de cas d'utilisation....59
5.6.1 Diagramme de cas d'utilisation général « client et visiteur »....60
5.6.2 Diagramme de cas d'utilisation général « administrateur »....61
5.6.3 Quelques diagrammes de cas d'utilisation détaillé....62
6. Conception....65
6.1 Diagramme de séquence....65
6.1.1 Définition....65
6.1.2 Diagramme de séquence de cas d'utilisation « Valider client »....66
6.1.3 Diagramme de séquence de cas d'utilisation « Ouvrir un nouveau compte » 67
6.2 Diagramme d'activités....68
6.2.1 Définition....68
6.2.2 Diagramme d'activités du cas d'utilisations « Débloquer carte »....69
6.2.3 Diagramme d'activités du cas d'utilisation « Effectuer virement »....70
6.3 Diagramme de classe....71
6.4 Conception de la base de données relationnelle....73
7. Conclusion....82
1. Introduction....84

CHAPITRE 4 : REALISATION

2. Architecture de l'application....84
3. Environnement de travail....85
3.1 Environnement logiciel....85

3.1.1 Système d'exploitation 85
3.1.2 Environnement de développement (IDE).....85
3.1.3 Hébergeur web.....86
3.1.4 Base de données.....86
3.1.5 Framework.....86
3.1.6 Langage de programmation.....87
3.2 Environnement matériel 89
3.3 Technologie et architecture 90
4. Présentation de l'application 91
4.1 Exemples de document imprimable 91
4.2 Exemple de graphes imprimable.....91
4.3 Captures d'écran (Seront disponibles prochainement) 96
4.3.1 Page d'accueil du site de la banque.....96
4.3.2 Application web coté client 97
4.3.3 Application web coté administrateur.....99
4.3.4 Applications mobiles 101
4.3.5 Site d'achat 107
5. Conclusion 109

Liste des figures

Figure 1: Architecture du réseau 4G LTE 24
Figure 2: Déploiement des stations de base sur un territoire 26
Figure 3: Un réseau cellulaire 26
Figure 4: L'architecture complète d'Android 41
Figure 5: Architecture du système d'exploitation iOS.....44
Figure 6: Diagramme de contexte.....56
Figure 7: Diagramme de cas d'utilisation « client et visiteur » 60
Figure 8: Diagramme de cas d'utilisation «administrateur».....61
Figure 9: Diagramme de cas d'utilisation « inscription » 62
Figure 10: Diagramme de cas d'utilisation « Commander une carte » 63
Figure 11: Diagramme de séquence de cas d'utilisation « Valider client ».....66
Figure 12: Diagramme de séquence de cas d'utilisation « Ouvrir un nouveau compte » .67
Figure 13: Diagramme d'activités du cas d'utilisation « Débloquer carte ».....69
Figure 14: Diagramme d'activités du cas d'utilisation « Effectuer virement » 70
Figure 15: Diagramme de classe général.....72
Figure 16: Architecture globale de l'application.....84
Figure 17: Capture des écrans page accueil.....96
Figure 18: capture d'écran page espaces comptes coté client web.....97
Figure 19 : capture d'écran page espace carte coté client web 98
Figure 20 : capture d'écran page consulter client coté admin 99
Figure 21: capture d'écran page statistiques coté admin 100
Figure 22: capture d'écran espace comptes version Android native 101
Figure 23:capture d'écran espace comptes version hybride 101
Figure 24: capture d'écran carte version hybride.....102
Figure 25: capture d'écran espace cartes version iOS native 102
Figure 26: capture d'écran statistiques version hybride.....103
Figure 27: capture d'écran statistiques version Android native.....103
Figure 28: capture d'écran virement hybride 104
Figure 29: capture d'écran virement version iOS native 104
Figure 30: capture d'écran espace activités version Android native 105
Figure 31: capture d'écran gestion plafond carte version iOS native.....105

Figure 32: capture d'écran page accueil version iOS native.....106
Figure 33: capture d'écran virement programmé version Android native 106
Figure 34: L'accueil de site d'achat.....107
Figure 35: Le formulaire de paiement avec la carte de crédit.....107
Figure 36: Formulaire de retrait.....108
Figure 37: Espace client de guichet.....108

Liste des tableaux

Tableau 1: Comparatif des réseaux mobiles 29
Tableau 2: Comparatif des types d'applications 38
Tableau 3: Caractéristiques de système d'exploitation Android 40
Tableau 4: Cas d'utilisations pour chaque acteur.....59
Tableau 5: Table admin.....73
Tableau 6: Table categorie 73
Tableau 7: Table carte.....74
Tableau 8: Table client.....75
Tableau 9: Table client_api.....75
Tableau 10: Table communes 76
Tableau 11: Table compte.....76
Tableau 12: Table dépôt.....76
Tableau 13: Table distributeur 76
Tableau 14: Table gerer_carte.....77
Tableau 15: Table gerer_client 77
Tableau 16: Table gerer_compte.....77
Tableau 17: Table gerer_incident 77
Tableau 18: Table hist_virement_prog 78
Tableau 19: Table incident_carte.....78
Tableau 20: Table message_client.....78
Tableau 21: Table message_visiteur.....79
Tableau 22: Table paiement.....80
Tableau 23: Table pays 80
Tableau 24: Table reponse_client 80
Tableau 25: Table reponse_visiteur.....81
Tableau 26: Table retrait 81
Tableau 27: Table virement 81
Tableau 28: Table virement_prog.....82
Tableau 29: Table visiteur.....82
Tableau 30: Table wilayas 82

Introduction générale

Ces dernières années ont vu émerger ce que les médias de n'ont pas hésité à nommer « la révolution des nouvelles technologies ». La démocratisation de l'ordinateur et de l'accès au réseau Internet nous a fait pleinement rentrer dans l'ère de l'informatique grand public. L'usage de ces technologies s'impose tant dans la vie domestique que professionnelle. Cette évolution touche également le secteur bancaire.

Le monde de la banque et des services financiers ne cesse de connaitre des changements, il est l'un des secteurs les plus innovateurs qui suit et qui rénove, incessamment, son activité, en profitant de 1a richesse variée des nouvelles technologies.

La banque en ligne est l'une des plus grandes révolutions de ce monde. Que ce soit pour les établissements bancaires ou les clients, les codes du domaine bancaire ont changé et désormais la banque en ligne apparaît comme une réelle alternative à la banque traditionnelle.

Avec les banques en ligne, à distance et à n'importe quelle heure, le client peut surveiller ses comptes, effectuer des virements, surveiller ses activités... De plus, des applications sur téléphone mobile ou tablette sont disponibles et elles offrent au client la possibilité de gérer ses finances sans avoir besoin d'un ordinateur.

Dans ce contexte, nous avons conçu et développé des applications web et mobiles pour la réalisation d'une banque en ligne « Banque Virtuelle d'Algérie », avec l'objectif de réaliser un développement comparable aux applications professionnelles.

Notre mémoire se présente comme suit :

- Le premier chapitre « Réseaux mobiles » donne un aperçu sur les différentes générations des technologies mobiles et l'apport de chacune ainsi que leurs architectures.
- Le deuxième chapitre s'intitule « Développement d'applications mobiles », et est consacré à la présentation des différents types d'applications mobiles en se focalisant sur les deux systèmes d'exploitations Android et iOS.
- Le troisième chapitre est dédié à l'analyse et la conception de notre application à l'aide des outils d'UML.
- Le quatrième et dernier chapitre expose le choix de l'environnement et les outils de développements ainsi que la présentation de l'application.

Nous examinons pour finir, en conclusion, les résultats ainsi que les perfections de notre travail.

Chapitre 1
Réseaux mobiles

1. INTRODUCTION

La possibilité d'accéder à Internet à partir d'un téléphone a profondément modifié nos vies, c'est pourquoi il est intéressant de raconter comment nous en sommes arrivés là. D'autant que nous sommes maintenant à la pointe de la prochaine génération des données mobiles, et qui dit données mobiles, dit forcement réseaux mobiles.

Le monde des réseaux mobiles n'a pas vu la lumière du jour au lendemain, l'histoire de la technologie mobile a commencé avec l'utilisation d'émetteur-récepteur radio mobile (comme les talkie-walkie) et a évolué vers l'état actuel avec les smartphones, tablettes, smartwatchs …etc.

Les réseaux cellulaires ont été confrontés à beaucoup de problèmes, notamment en qualité de signal et en sécurité, par conséquence, beaucoup de protocoles ont été mis en œuvre pour répondre aux exigences, et le monde aujourd'hui est témoin de l'évolution et de l'apport des générations des réseaux mobiles durant les 40 dernières années.

Ces réseaux ont évolué à travers une série de générations, chaque génération est venue pour innover et améliorer la génération précédente, on a démarré avec la 1G, et aujourd'hui, on est avec la 4G et la 5G est sur les portes.

Dans ce chapitre, nous verrons ensemble l'évolution de ces différentes générations et l'apport technologique de chacune d'elles et leurs architectures globales, ensuite, nous proposerons une classification des réseaux sans-fils selon leurs périmètres géographiques, et enfin, nous terminons avec un tableau comparatif des différents réseaux cellulaires.

2. DEFINITION

Un réseau est un ensemble d'équipements interconnectés pour échanger de l'information.

Un réseau mobile est un réseau où les terminaux sont des téléphones sans-fil mobiles, et les communications sont assurées par le biais des fréquences radios via les antennes relais ou ce qu'on appelle les stations de transmission de base (**BTS**).[1]

3. LES DEFIS DES RESEAUX MOBILES

- **Un débit faible** : à l'apparition des réseaux mobiles, un réseau sans-fil offrait un débit plus faible que son équivalent filaire. Il fallait donc minimiser la bande passante utilisée pour la gestion du réseau, afin de réserver la bande passante pour les communications.
- **Une atténuation rapide du signal** : plus la distance est grande, plus le signal s'affaiblit.
- **Erreurs de transmissions et interférences** : les liens radios ne sont pas isolés et le nombre de fréquence disponibles est limité, il faut donc les partager, ce qui provoque des interférences entre des fréquences trop proches et même subir des bruit parasites par l'environnement. Par conséquence, les taux d'erreurs de transmission dans les réseaux radios sont plus élevés que dans les réseaux filaires.
- **Une faible sécurité physique** : les réseaux mobiles sont vulnérables à l'écoute, en effet, il est facile pour tout nœud à portée radio d'un nœud émetteur de capter l'information d'une manière passive.
- …etc.

[1] Source Wikipedia

4. EVOLUTION DES RESEAUX MOBILES

Avant la première apparition des téléphones mobiles, il existait des radios téléphonies qu'on trouvait dans des camions ou des voitures. Ils permettaient juste les communications vocales. Cette technologie a été utilisée pendant une longue période et elle a connue beaucoup de problèmes surtout en termes de sécurité, il suffit d'ajuster la fréquence pour écouter la conversation entre deux entités, en plus elle n'assure pas une communication bidirectionnelle (un parle et l'autre écoute).

4.1 PREMIERE GENERATION (1G)

Il s'agit de la première génération des téléphones sans-fil. À l'époque (1980), c'était un exploit technologique, elle est basée principalement sur le Standard **NMT** (Nordic Mobile Telephone) dans les pays européens nordiques, **AMPS** (Advanced Mobile Phone System).

Ce type de réseau utilise le signal hertzien comme moyen de communication, il permet de passer des appels vocaux dans un même pays, un transfert de donnée limité à une vitesse de 2,4 Kb/s.

L'inconvénient avec cette technologie, c'est que le signal utilisé est facilement intercepté avec un récepteur radio tout bande, il est donc un sujet à la distorsion (déformation). En plus, cette technologie n'est pas standardisée (elle a connu plus de 5 versions), et par conséquent, des problèmes d'incompatibilité. On peut citer aussi la mauvaise qualité vocale due au faible taux de transfert de données.

4.2 DEUXIEME GENERATION (2G)

Elle est apparue en 1990 pour la première fois, elle a changé complètement le principe des réseaux mobiles par rapport à ces prédécesseurs grâce au passage de l'analogique vers le numérique (système binaire). Initialement, elle était basée sur **GSM** (Global System for Mobile Communications) qui est un protocole standard des réseaux mobiles pour assurer la compatibilité. Ce qui a vraiment marqué cette première version, c'était l'apparition des appels vocaux numériques (à n'importe quelle destination) qui ont permis d'avoir une meilleure communication vocale, sans oublier aussi le lancement de service SMS (limité à 80 caractères) et le service MMS (message multimédia), les débits pouvant aller en moyenne jusqu'à 64 kbits.

Malgré ses innovations, des problèmes ont été signalés, dont le problème de signal numérique faible et l'incapacité de gérer des données complexe comme les vidéos. En outre, le **GSM** était amélioré grâce à l'apparition de la norme **GPRS** (General Packet Radio Service), qu'est considérée comme une extension de **GSM** (on la qualifie souvent de **2,5G**). Cette extension a apporté des nouveautés et des améliorations remarquables. En termes de débit, on a passé de 64 kb/s à 144 kb/s, et en termes de service, l'apparition de la messagerie mail et la navigation sur le web.

La norme **EDGE** (Enhanced Data Rates for GSM Evolution) fait son entrée pour rectifier certaines limites de **GPRS** avec ses corrections et ses améliorations grâce au service haut débit (jusqu'à 500 kb/s) et l'amélioration de la qualité de signale en zone rurale ou en zone suburbaine. Cette norme était la transition vers une nouvelle génération des réseaux mobiles (**3G)**.

4.3 TROISIEME GENERATION (3G)

Le premier réseau **3G** a été introduit en 2000, il s'agit de l'évolution de la **2G** pour une vitesse internet plus rapide, et pour un ensemble de standard unifiées dans le monde entier.

La 3G s'appuie dans sa première apparition sur la norme **UMTS** (Universal Mobile Telecommunications Service) elle fournit des services multimédias avec une meilleure qualité grâce au débit élevé qu'elle propose à partir de 200Kb/s jusqu'à 2Mb/s, GPS, conférence vidéo, télévision streaming, les jeux 3D …etc. Ce qui est intéressant avec cette génération c'est qu'elle permet une interopérabilité avec le **GSM**.

L'évolution de la 3G a boosté c'est capacité en adoptant **HSPA** (High Speed Packet Access), il s'agit d'une fusion de deux protocoles mobiles **HSDPA** (High Speed Downlink Packet Access) et **HSUPA** (High Speed Uplink Packet Access ou aussi appeler **3G+**), cette nouveauté a amélioré les performances des réseaux de télécommunication mobiles 3G existants principalement le débit, on a atteint 14 Mb/s en liaison descendante (downlink) et 5.76 Mb/s en liaison ascendante (uplink).

Une autre amélioration a été publiée en 2008, il s'agit de **HSPA+** (Evolved High Speed Packet Access, nommé **H+** ou **3G++**), cette nouvelle norme a permis d'atteindre un débit théorique de 337 Mb/s en liaison descendante et à 34 Mb/s en liaison ascendante mais ces vitesses sont rarement atteintes dans la pratique.

Ce qui rend le réseau 3G désavantageux pour ses utilisateurs est le fait que les téléphones cellulaires équipés par cette technologie sont plus chers par rapport à ceux qui ne partagent pas la même technologie, de plus on a cherché à atteindre des débits plus élevés ce qui a donné naissance à une nouvelle génération (**4G**).

4.4 QUATRIEME GENERATION (4G)

La **4G** est conçue pour atteindre un plus haut débit (une vitesse plus élevée) que ses prédécesseurs **3G** et **2G**, dans certains cas, surpassant même les vitesses offertes par la connexion internet à domicile.

Dans sa première version, elle était basée sur la technologie **LTE** (Long Term Evolution) que nous détaillerons plus loin et **WiMAX** (Worldwide Interoperability for Microwave Access). Ces deux normes ont permis d'atteindre une vitesse de 100 Mb/s jusqu'à 1Gb/s selon la localisation et le nombre d'utilisateurs connectés au réseau, une technologie qui fournit un service multimédia étendu comme le service télévision 3D et mobile TV Haute Définition, dans la téléphonie, le service **VoIP** (Voice over IP) qui permet de transmettre la voix sur des réseaux IP filaires ou sans-fil toute en assurant une meilleure qualité grâce à l'architecture de la **4G** orienté IP.

4.5 CINQUIEME GENERATION (5G)

Le monde prévoit le déploiement d'une nouvelle génération **5G** en 2020, elle devrait constituer une plate-forme **WWWW** (World Wide Wireless Web) idéale pour se connecter n'importe où sur la planète. Un véritable monde sans-fil, où nous pouvons accéder via Internet sans rencontrer d'obstacles ni de restrictions d'espace ni de temps, cette génération vise à atteindre un débit 30Gb/s, mettre en place une connexion qui supporte au moins 1 million d'appareils connectés par kilomètre carré pour activer le support IoT (Internet of Things) avec un temps de réponse minimal 4ms (en comparant à 20ms pour **LTE**), accessibilité au réseau sans rupture par les appareils mobiles déplaçant à une grande vitesse qui peut aller jusqu'à 500Km/h …etc.

Le système de communications cellulaires mobiles 5G constituera un changement majeur dans le fonctionnement des réseaux de communications mobiles. Pour atteindre cet objectif, un nouveau réseau d'accès radio (**5G New Radio**) et un nouveau réseau central sont nécessaires pour fournir les performances requises.

En utilisant la technologie 5G NR, le système est en mesure de répondre aux besoins différents et invariables des utilisateurs mobiles, qu'il s'agisse d'un petit nœud IoT, ou d'un utilisateur de données de grande taille, fixe ou mobile.

Bien que les déploiements initiaux de la 5G utilisent le réseau principal de réseaux LTE, voire même 3G, l'objectif est de disposer d'un nouveau réseau capable de gérer des volumes de données beaucoup plus importants tout en étant capable de fournir un niveau de latence[2] beaucoup plus bas. **5G NextGen Core Network** permet de répondre à ce besoin.

4.6 LES RESEAUX DE NOUVELLE GENERATION (NGN)

NGN (New Generation Network) ou les réseaux de nouvelle génération sont une nouvelle architecture de réseau qui peut être utilisée à la fois pour la téléphonie et les données toute en supportant la mobilité. Initialement, le terme réseau de nouvelle génération était utilisé pour désigner la transformation du réseau central en IP (orienté IP). Il transporte toutes les informations et tous les services (voix, données et toutes sortes de supports tels que la vidéo) en les encapsulant dans des paquets IP, similaires à ceux utilisés sur Internet.

En effet, **NGN** est une transition des anciens réseaux basés sur la communication radio vers des réseaux communicants à base de paquets, une technologie qui n'est pas encore mature, il fixe encore les règles et les protocoles de cette technologie mais les experts affirment que la **NGN** est une nouvelle ère pour les réseaux, qui permettra d'atteindre des dizaines de débit voir même des centaines de gigabits, la création des nouveaux services et l'innovation des anciens services et bien d'autres avantages.

[2] **Latence :** le temps de réponse

5. CLASSIFICATION DES RESEAUX SANS-FIL

Les réseaux sans-fil peuvent être classifiés selon le périmètre géographique offrant une connectivité (appelé zone de couverture). Autrement dit selon la distance séparant les unités mobiles. On distingue, quatre catégories : réseaux personnels sans-fil, réseaux locaux sans-fil, réseaux métropolitains sans-fil et réseaux étendus sans-fil.

5.1 LES RESEAUX PERSONNELS SANS-FIL (WPAN)

Les réseaux sans-fil personnel **WPAN** (Wireless Personal Area Network), appelés également réseaux domestiques, ils sont des réseaux sans-fil à très faible portée (de l'ordre d'une dizaine de mètres). Ce type de réseau sert à :

- Faire communiquer deux ou plusieurs équipements présents sur une personne (par exemple : kit mains libres et un téléphone portable).
- Relier des périphériques (par exemple pour relier une imprimante, un scanner ou un PDA à un ordinateur sans liaison filaire).
- Assurer la liaison sans-fil entre deux machines très peu distantes.

Il existe plusieurs technologies appartenant à ce type de réseau, on peut citer le Bluetooth, infrarouge, ZigBee…etc.

5.2 LES RESEAUX LOCAUX SANS-FIL (WLAN)

Les réseaux locaux sans-fil ou **WLAN** (Wireless Local Area Network) sont généralement utilisés à l'intérieur d'entreprises, d'universités ou chez des particuliers. Ce réseau permet de relier des terminaux présents dans une portée d'environ une centaine de mètres. Les principales technologies utilisées dans cette catégorie sont : HiperLan1, HiperLan2, IEEE 802.11 (Wi-Fi) …etc.

Prenant exemple la norme IEEE 802.1, c'est la plus répondue dans les réseaux locaux sans-fil. Elle offre un débit allant jusqu'à 54 Mb/s sur une distance de plusieurs centaines de mètres. La norme initiale 802.11 a connu de nombreuses révisions notées 802.118, 802.11b, 802.11g pour les principales. Ces révisions visent essentiellement une amélioration du débit et/ou une amélioration de la sécurité.

5.3 LES RESEAUX METROPOLITAINS SANS-FIL (WMAN)

Un réseau métropolitain sans-fil ou **WMAN** (Wireless Metropolitan Area Network), également appelés boucle locale radio (**BLR**) permet à un particulier ou une entreprise d'être relié à son opérateur (téléphonie fixe, Internet, télévision...) via les ondes radio. Les réseaux WMAN sont basés sur la norme IEEE 802.16. Ils offrent un débit utile de l à 10 Mb/s pour une portée de 4 à 10 kilomètres. Comme exemple de WMAN, on trouve le **WiMAX** (Worldwide Interoperability for Microwave Access), et il y a d'autres ...etc.

5.3.1 WiMAX

Il s'agit d'un standard de réseau sans fil métropolitain, son objectif est de fournir une connexion internet à haut débit sur une zone de couverture de plusieurs kilomètres de rayon allant à des débit montants et descendants de 70 Mb/s avec une portée de 50 kilomètres. Le standard WiMAX possède l'avantage de permettre une connexion sans fils entre une station de base **BTS** et des milliers d'abonnés sans nécessiter de ligne visuelle directe **LOS** (Line Of Sight) ou **NLOS** (Non Line Of Sight). Dans la réalité le WiMAX ne permet de franchir que de petits obstacles tels que des arbres ou une maison mais ne peut en aucun cas traverser les collines ou les immeubles. Le débit réel lors de la présence d'obstacles ne pourra ainsi dépasser les 20 Mb/s.[3]

5.4 RESEAUX ETENDUS SANS-FIL (WWAN)

Les réseaux sans-fil étendus ou **WWAN** (Wireless Wide Area Network) sont également connus sous le nom de réseau cellulaire mobile. Les WWANs regroupent notamment les différents réseaux téléphoniques et les réseaux satellitaires. Les principales technologies : **GSM**, **GPRS**, **UMTS, LTE**.

5.4.1 GSM

GSM permet l'envoi de courts messages (SMS) et la transmission de données avec un débit allant de 1 à 9,6 Kb/s. Elle propose aussi un ensemble de services complémentaires :

- Renvoi d'appel.
- Présentation d'appel en instance.

[3] D'après la source

- Conférence à 3.
- Mise en attente (en garde) d'un appel.
- Présentation du numéro de l'appelant.

Nous verrons tout de suite que le réseau GSM est une base pour la mise en place des réseaux GPRS et UMTS.

5.4.2 GPRS

GPRS (General Packet Radio Service) est une extension de GSM, il vient d'améliorer son prédécesseur en terme de débit et de service. Elle ajoute par rapport à ce dernier la transmission par paquet (commutation de paquets, que pour les appels) toute en gardant l'interopérabilité avec le GSM. Elle propose un débit allant jusqu'à 144 Kb/s, grâce à cette vitesse, les internautes peut s'échanger des fichiers de taille moyennes (pièce jointes dans des mail, MMS et FTP …). La nouveauté qui a marqué cette innovation, c'est la navigation sur le web qui a été la cause de lancement d'une multitude de service en ligne (E-commerce, E-payement, E-business, …) à grand échelle dans le monde.

5.4.3 UMTS

UMTS (Universal Mobile Telecommunications System) est la troisième génération des téléphones mobiles, basée sur le standard GSM. UMTS vise à atteindre un débit de 200 Kb/s jusqu'à 2 Mb/s, elle a pour but d'assurer l'interopérabilité avec ses prédécesseurs (GSM, GPRS), transférer simultanément la voix et des données numériques à haut débit. Il a fait apparaitre de nouveaux services comme le mobile TV, GPS, les appels vidéos, les jeux 3D …etc.

5.4.4 LTE

LTE (Long Term Evolution) est la norme utilisée pour la quatrième génération des réseaux mobiles, elle permet d'obtenir un débit de 100Mb/s, elle supporte environ 200 utilisateurs actifs simultanément dans une cellule (zone de couverture) qui peut être en mouvement avec une vitesse de 350 km/h.

6. ARCHITECTURE DES DIFFERENTS RESEAUX MOBILES

L'architecture de tous les réseaux mobiles (que ce soit 2G/3G/LTE) est toujours composée de deux grandes parties qui sont : la partie radio (ou bien appeler aussi partie accès) et la partie cœur.

Le réseau LTE est uniquement basé sur l'IP (Internet Protocol), par contre, la 3G est un réseau basé sur le CS (Circuit Switch[4]) et PS (Packet Switch[5]), mais ce qui concerne la 2G, on parle seulement du CS.

Généralement, pour la 2G la partie radio est composée du BTS (station de base), BSS (c'est l'entité responsable du contrôle des différents BTS). Pour la partie cœur, on a le MSC (Mobile Switching Center), le HLR (Home Location Register) et le VLR (Visitor Location Register).

Pour la 3G, on parle du NB (Node B, c'est pareil que la BTS) et on a aussi le RNC qui est l'entité la plus intelligente dans le réseau (il est responsable de tout ce qui est management des ressources radio). Pour le cœur réseau, on a introduit le SSGN et GGSN pour la commutation (switching) de tout ce qui est paquet.

6.1 ARCHITECTURE DU RESEAU 4G LTE

Comme nous l'avons vu, tous les réseaux mobiles se décomposent en deux parties, la partie radio ou appelé aussi réseau d'accès et la partie cœur, dans le cas du réseau 4G, la partie réseau est appelée eUTRAN (evolved Universal Terrestrial Radio Acces Network) et le réseau cœur appelé EPC (Evolved Packet Core).

La première partie « eUTRAN », est simplifiée comparée à celles des réseaux 2G et 3G, par l'intégration dans les stations de base « eNode B » des fonctions de contrôles qui étaient auparavant localisées dans les RNC (Radio Network Controller) des réseaux 3G UMTS.

La partie radio d'un réseau LTE se compose donc des eNode B, d'antennes locales ou distantes, de liaisons en fibres optiques vers les antennes distantes et des liaisons IP reliant les eNode B entre eux, et avec le cœur de réseau

[4] **Circuit Switch** est la méthode originale de transfert de données développée pour les systèmes de téléphone mobiles, Il a un fonctionnement similaire à celui d'un appel vocal où une bande de fréquence radio est allouée.

[5] **Packet Switch** est un procédé de regroupement des données qui sont transmises sur un réseau numérique en paquets. Les paquets sont constitués d'un en- tête et d'une charge utile. Les données contenues dans l'en-tête sont utilisées par le matériel réseau pour diriger le paquet vers sa destination.

La seconde partie EPC utilise des technologies « full IP », c'est-à-dire basées sur les protocoles internet pour la signalisation, le transport de la voix et des données. Ce cœur de réseau permet l'interconnexion via des routeurs avec les eNodeB distants, les réseaux des autres opérateurs mobiles, les réseaux de téléphonie fixe et le réseau Internet.

L'EPC d'un opérateur LTE est principalement constitué de passerelles de services, SGW (Serving GateWays) qui transportent le trafic de données et concentrent le trafic de plusieurs eNodeB, de MMEs qui gèrent la signalisation (plan de contrôle) et donnent l'accès aux bases de données (HSS / HLR). Un (ou plusieurs) PGW servent de passerelles vers le réseau Internet.

L'EPC permet aussi, via d'autres passerelles, l'accès des terminaux au cœur de réseau LTE, en utilisant d'autres types d'accès radio : des points d'accès Wi-Fi ou des Femtocells généralement connectés via des box ADSL ou FTTH.

L'utilisation du protocole IP de bout-en bout dans le cœur de réseau permet, par rapport aux réseaux 3G, des temps de latence réduits pour l'accès internet et les appels vocaux LTE.

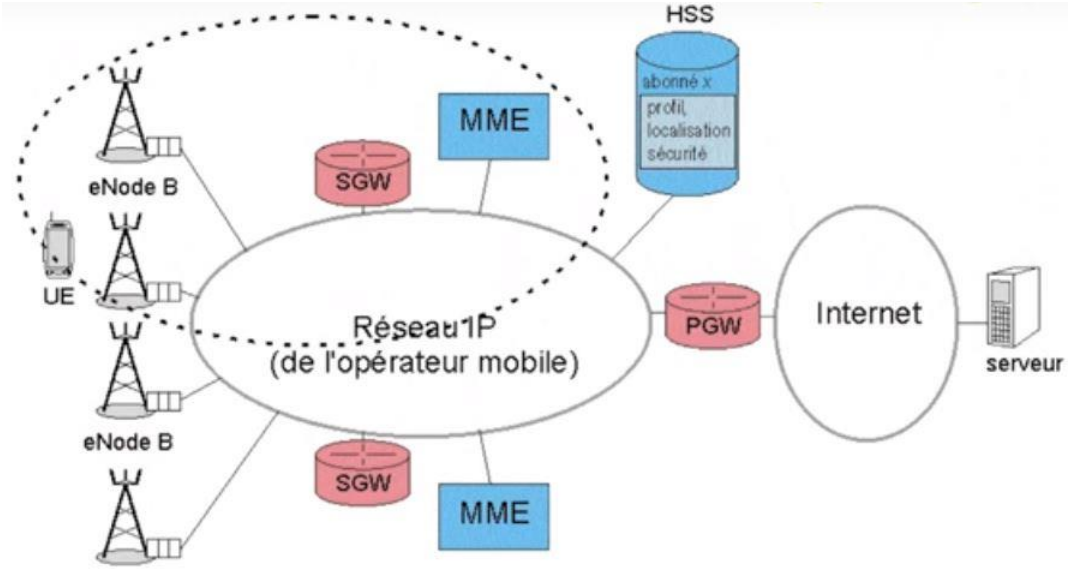


Figure 1: Architecture du réseau 4G LTE

- **UE** : (User Equipment) il peut s'agir de tout appareil capable d'établir des fonctions de communication telles que des téléphones mobiles, des ordinateurs, …etc.
- **eNode B** : (« evolved » évolution par rapport à la 3G, Node Base station) station de base composée d'émetteurs-récepteurs qui réalisent la passerelle entre les terminaux mobiles, les antennes radio et le cœur du réseau des opérateurs LTE. Elles intègrent aussi des fonctions de contrôles, on peut trouver jusqu'à 10000 dans un pays.
- **PGW** : (Packet GateWay) il constitue une porte d'entrée unique entre le réseau IP de l'opérateur et Internet. Il achemine donc les données internet vers le terminal et réciproquement les données du terminal vers Internet. Il assure également quelques fonctions de sécurité, et attribue les adresses IP aux terminaux LTE.
- **SGW** : (Serving GateWay) il constitue une passerelle régionale reliée au PGW. Il collecte les données à envoyer vers le PGW, données provenant des terminaux via les stations de base. Il participe aussi à la transmission des données en sens opposé, depuis le PGW vers les terminaux.
- **HSS** : (Home Subscriber Server) contient toutes les informations sur tous les abonnés de l'opérateur réseau dans une base de données centrale.
- **MME** : (Mobility Managment Entity) l'équipement qui gère la signalisation (plan de contrôle) entre les terminaux (UE) et le cœur de réseau LTE. Il est chargé de dialoguer avec le HSS de manière à consulter et stocker les profils et les données de sécurité des terminaux situés dans une grande région donnée.

6.2 FONCTIONNEMENT D'UN RESEAU LTE

Les stations de base composant la partie réseau sont déployées le plus régulièrement possible de façon à couvrir un territoire, en ayant une faible densité dans les zones rurales et une forte densité dans les zones urbaines en suivant le concept de la décomposition en cellule, où chaque cellule contient son annotaine qui couvre sa superficie (figure 2 et 3),
Elles sont reliées à un réseau IP déployé par un opérateur mobile ce qui fait qu'elles peuvent communiquer entre elles

Figure 2: Déploiement des stations de base sur un territoire

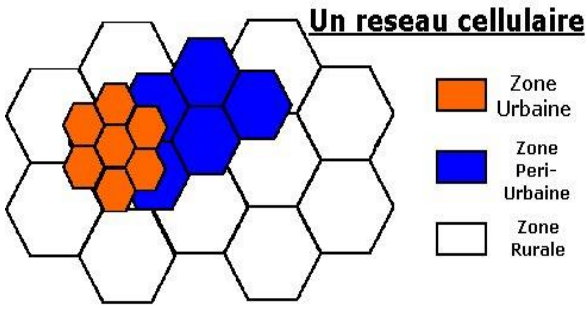


Figure 3: Un réseau cellulaire

Pour accéder à ce réseau l'utilisateur doit posséder un terminal (User Equipment) doté de la technologie 4G ainsi que d'une carte Sim qui contient des données d'abonnement unique au monde, et se trouver à proximité des stations de base pour pouvoir communiquer avec elles à travers un signal radio.
Chaque station de base diffuse continuellement un signal appelé « voie balise » qui indique son existence, les terminaux mesurent la puissance avec laquelle ils reçoivent le signal et se permutent entre les stations en fonction de cette dernière.

6.2.1 Le contrôle d'information dans un réseau LTE

Au moment de l'allumage du terminal, celui-ci envoi un signale au eNodeB, qui va le transmettre au MME de sa part le MEE va vérifier s'il possède ses coordonnés si ce n'est pas le cas il va les avoir chez le HSS qui contient le profil de tous les

abonnés du réseau ainsi que leurs droits et services auxquels ils sont souscrits et les données de sécurité.
Ensuite pour tous les accès ultérieurs le MEE va pouvoir contrôler localement (ce profile est une sorte de mémoire cache), donc il est responsable de l'authentification des utilisateurs, il aussi responsable de la traçabilité des utilisateurs; savoir a tous moments où ils se trouvent et connaitre leurs états (au repos, connecté).
De manière générale le MME et le HSS ne s'occupent que de la gestion de la signalisation qu'est un ensemble de messages échangés pour gérer le réseau (informations de contrôle qui sont utiles pour l'opérateur, contrôle d'accès et suivie du terminal en cas de mobilité).

6.2.2 Les flux de données dans un réseau LTE

En envoyant une requête à partir d'un smartphone, le paquet des données se transmet par voie radio vers la station de base la plus proche, et celle-ci le transmet au PGW à travers le réseau IP de l'opérateur, le PGW relié aussi au réseau internet l'achemine vers le serveur, en plus de son rôle de passerelle entre les deux réseaux il assure aussi la sécurité du réseau en permettant de remplir les fonctions d'un pare-feu.
Pour éviter des reroutages fréquents, à cause de la mobilité des terminaux, on utilise des SGW qui couvre une zone géographique, et assure la gestion de ce reroutage en étant un intermédiaire entre les eNode et les PGW, donc il est responsable du transfert d'un relais à un autre, il est capable d'effectuer ce transfert vers un réseau 2G ou 3G.

7. COMPARATIF DES RESEAUX MOBILES

Nous présentons ci-dessous un tableau comparatif des différentes générations des réseaux mobiles

Génération	Définition	Débit	Technologie	Services offerts
1G	Analogique	2.4-14.4 Kb/s	AMPS, NMT	Les téléphones sans-fil sont utilisés uniquement pour la voix
2G	Données numériques à bande étroite	14.4-64 Kb/s	GSM	Plusieurs utilisateurs dans un seul canal, Service SMS et MMS
2.5G	Données par paquets	64-144 Kb/s	GPRS	Service multimédia et mail, navigation web
3G	Données par paquets numériques à large bande	200 Kb/s 2 Mb/s	UMTS, EDGE	Télévision streaming, conférence vidéo, GPS
3.5G	Données par paquets	1-3 Mb/s	HSPA, HSPA+	Débit plus élevé, amélioration des services existants

4G	Paquets numériques haut débit (tout IP)	100-300 Mb/s	WiMAX, LTE, LTE+	Télévision 3D, service VoIP
5G	World Wide Wireless Web	Probablement dizaine Gb/s	Pas encore	Très haut débit, large bande passante

Tableau 1: Comparatif des réseaux mobiles

8. CONCLUSION

Dans ce chapitre, nous avons vu comment les générations de la téléphonie mobile ont progressé à travers le temps, et comment ces réseaux sont classés selon le déploiement géographique toute en expliquant quelques normes utilisées par chaque type de réseau.
En parallèle avec l'évolution des réseaux mobiles, il y a eu l'apparition de différents services en ligne comme E-learning, E-payement, E-commerce …etc.

Grâce à l'utilisation des nouvelles technologies surtout les réseaux et les applications mobiles, le monde a connu un énorme changement qui a redéfini l'économie mondiale tel qu'elle est connue auparavant et il a ouvert des nouvelles perspectives dans le domaine économique et le secteur du marché, un phénomène qui a permet aux professionnels et aux clients de se mettre en contact direct, et de manière instantanée.

Aujourd'hui, le client d'une banque (exemple Societe Generale) n'est pas obligé de se présenter au guichet pour effectuer un paiement, la majorité des banques ont mis des dispositifs spécialisés, on peut citer le TPE (Terminal de paiement électronique) qui permet d'effectuer un payement n'importe où, la seule condition est de pouvoir capter un réseau de téléphonie mobile GPRS ou 3G (selon le TPE), on peut citer aussi E-wallet (Porte-monnaie électronique), un dispositif qui peut stocker de la monnaie sans avoir besoin d'un compte bancaire et d'effectuer directement des paiements sur TPE, …et d'autres.

Dans ce qui suit, nous allons aborder le développement des applications et des services cités auparavant, notamment le développement des applications mobiles vu que 98.79% (selon StatCounter GlobalStats) des appareils en Algérie fonctionnent sous ces deux systèmes d'exploitations, et nous allons voir ensemble comment se déroule le développement de ces applications.

Chapitre 2
Développement d’applications mobiles

1. INTRODUCTION

Chaque jour, des nouveaux appareils arrivent sur le marché avec des options innovantes grâce à la technologie en croissance. L'évolution de la technologie de développement d'applications mobiles avec de nouveaux appareils a grandement simplifié notre vie.

Il existe une demande rapide d'applications mobiles sur les deux systèmes d'exploitation Android ainsi que sur iOS. Selon les dernières statistiques de GlobalStats statcounter, Android couvre 75.39 % et iOS 22.35%, ce qui doit certainement augmenter à l'avenir, de même que la demande d'applications.

Le nombre est spectaculaire et il y a une croissance dans le secteur de la téléphonie mobile. Etre disponible sur un appareil connecté à Internet est nécessaire pour chaque entreprise, ce qui a donné le coup d'envoi au développement d'applications mobiles.

Dans ce chapitre, nous verrons : ce qu'est une application mobile, comment elle est développée et les différents types d'applications mobiles. Ensuite nous allons détailler les deux systèmes d'exploitations mobiles les plus utilisés, Android et iOS, puis nous terminons par un tableau comparatif de ces derniers.

2. DEFINITIONS

2.1 APPLICATION

Programme ou ensemble de programmes destiné à aider l'utilisateur d'un ordinateur pour le traitement d'une tâche précise.[6]

2.2 DEVELOPPEMENT APPLICATIONS MOBILES

Le développement d'applications mobiles consiste essentiellement à concevoir et à développer des logiciels pour terminaux mobiles, appelés applications mobiles. Le développement d'applications mobiles est fondamentalement la demande, l'exigence de créer l'application nécessaire, telle que demandée par l'utilisateur et compatible avec le système d'exploitation, pour effectuer sa tâche.
Exemples : applications Facebook, Uber, Google MAP et bien plus encore.

Quand on parle de développement d'applications, on parle généralement de développement pour les appareils mobiles, cela inclut les smartphones, les tablettes. Il existe également des applications pour le Web et des dispositifs portables tels que les smartwatchs, mais pour les besoins de ce chapitre, nous resterons dans les limites des appareils mobiles tels que les smartphones et les tablettes, plus précisément, nous allons nous concentrer sur l'un des deux plus grands systèmes d'exploitation : iOS et Android

3. TYPE D'APPLICATIONS MOBILES

3.1 APPLICATIONS NATIVES

Il s'agit d'applications développées pour être utilisées sur une plate-forme ou un système d'exploitation particulier, tel qu'Android, iOS, etc. Les applications natives sont généralement écrites dans des langages acceptés par la plate-forme. Ils sont également conçus à l'aide de l'environnement de développement intégré (IDE)

[6] D'après le dictionnaire Larousse

spécifique pour le système d'exploitation donné, tels que Android Studio pour les applications Android et XCode pour les applications iOS.
Le fait de développer une application de ce type permet généralement d'utiliser toutes les fonctionnalités liées au système d'exploitation visé (GPS, accéléromètre, appareil photo, etc.) et permet également de proposer des applications généralement plus riches que les web applications en HTML5. Une fois téléchargées et installées certaines applications peuvent par ailleurs être utilisées sans connexion Internet, exemple des applications météorologique.

3.1.1 Avantages

Les applications natives présentent un certain nombre d'avantages importants par rapport aux autres solutions qu'on verra prochainement, on peut citer :

- Elles offrent aux utilisateurs l'expérience la plus rapide, la plus fiable et la plus réactive
- Facile à exploiter dans une fonctionnalité plus large de l'appareil Native facilite l'utilisation des gestes de la caméra, du microphone, de la boussole, de l'accéléromètre
- Peuvent utiliser les notifications push pour alerter les utilisateurs lorsque leur attention est requise dans l'application.

3.1.2 Inconvénients

- Les applications iOS ne fonctionneront pas sur Android et inversement, donc il faut travailler avec des bases de code différentes pour chaque plate-forme.
- Elles peuvent coûter plus cher et prendre plus de temps à créer, vu que La plupart des développeurs se spécialisent dans une plate-forme (Android ou iOS). Par conséquent, le développement de l'application sur les deux plates-formes nécessite deux développeurs (ou équipes) distincts

3.2 APPLICATIONS HYBRIDES

Les applications hybrides sont à mi-chemin entre Web App (Voir 3.3) et les applications natives, Elles sont développées pour être utilisées sur plusieurs plates-

formes, c.-à-d. pouvant être déployées sur les plates-formes iOS et Android. Les applications mobiles hybrides sont conçues de la même manière que les sites Web. Les deux utilisent une combinaison de technologies telles que HTML, CSS et JavaScript. Cependant, au lieu de cibler un navigateur mobile, les applications hybrides ciblent une WebView[7] hébergée sur une application native. Cela leur permet de faire des choses comme accéder aux capacités matérielles du périphérique mobile.

3.2.1 Avantages

- Adaptable à plusieurs plates-formes, le même code pouvant être réutilisé pour Android, iOS et Windows.
- Développement unifié et moins coûteux, car l'application ne doit être développée qu'une seule fois à l'aide d'un code de base.

3.2.2 Inconvénients

- Performances d'application plus lentes que les applications natives, car la structure hybride sert de pont pour communiquer avec les fonctionnalités natives du téléphone.
- Les applications comportant de nombreuses animations et effets sonores ne sont pas aussi transparentes que leurs homologues natifs.

3.3 APPLICATIONS WEB PROGRESSIVES

Une application Web progressive (PWA) ou Web App est une application Web qui utilise des fonctionnalités Web modernes pour offrir aux utilisateurs une expérience similaire à celle d'une application sans qu'ils soient obligés d'installer une application à partir de l'App Store / Play Store. Ils sont généralement accessibles via une URL Web, qui peut toujours être épinglée ou enregistrée sur l'écran d'accueil du téléphone. Les PWA sont généralement construits en utilisant HTML, CSS,

[7] **WebView :** est un mini navigateur intégrer dans une application mobile pour afficher de contenu web

JavaScript également, même si parfois des web App correspondent simplement à un site web responsive (qui s'adapte à la taille de l'écran utilisé).

3.3.1 Avantages

- Facile à construire.
- Facile à maintenir.
- Fonctionne hors ligne et fonctionne bien sur les réseaux de faible qualité.
- Créez une seule application pour toutes les plates-formes - iOS, Android, etc. tant qu'elle peut exécuter un navigateur.

3.3.2 Inconvénients

- Nécessite un navigateur pour fonctionner. Les utilisateurs doivent faire un autre pas pour taper l'URL de l'application, ce qui constitue une expérience utilisateur médiocre.
- Les applications Web sont moins interactives et intuitives que les applications natives.

3.4 TABLEAU COMPARATIF

	Web App	**Application native**	**Application hybride**
Compatibilité sur les plateformes	(+) Compatible sur tous les terminaux	(-) Compatible sur 1 seule plateforme	(+) Compatible sur tous les terminaux (nécessite une configuration particulière pour chacune)
Expérience utilisateur	(-) Limitée (même ergonomie et graphisme pour toutes les plateformes)	(+) Sur mesure	(-) Limitée (même ergonomie et graphisme pour toutes les plateformes)
Performances	(-) Réduites (dépend de la qualité du réseau et du navigateur utilisé)	(+) Optimales	(-) Réduites (dépend de la qualité du réseau)
Hors connexion	(-) Non disponible	(+) Disponible	(+) Disponible

Distribution et mise à jour	Via Internet (+) Mises à jour transparentes (-) Fidélisation plus compliquée qui influe sur le trafic de l'application	Sur un store (-) Nécessite une Republication et validation à chaque mise à jour (un peu long pour iOS)	Sur un store (+) Mises à jour transparentes pour la partie web
Monétisation	(-) Impossible de vendre le téléchargement de l'application (+) Publicité	(+) L'application peut être payante sur le store (mais un pourcentage est prélevé) (+) Achats in-app (+) Publicité	(+) L'application peut être payante sur le store (mais un pourcentage est prélevé) (+) Publicité
Exemple	Ouedkniss	Outlook	Facebook

Tableau 2: Comparatif des types d'applications

4. DEVELOPPEMENT APPLICATION ANDROID

4.1 DEFINITION

Android est un système d'exploitation ouvert (Open Source) pour tablettes tactiles, terminaux mobiles et TV connectées. Il a été conçu en 2007, par la société Android, une start-up rachetée par Google, c'est un système d'exploitation fondé sur un noyau Linux, disponible grâce à une licence Apache, Android inclut tous les utilitaires requis par un constructeur pour le mettre en œuvre dans un téléphone portable.il est proposé à tous les fabricants de téléphones mobiles, pour faciliter son adoption.

Le bureau est étendu sur cinq parties minimum, personnalisables. Cependant, Android est modifiable par les constructeurs. Donc, les versions varient largement d'un constructeur à l'autre.

Ces dernières ont toutes des noms de desserts et suivent une logique alphabétique. Les connues sont

- Gingerbread, sortie en décembre 2010
- Honeycomb sortie en janvier 2011
- Ice Cream Sandwich 2011
- Jelly Bean en 2012
- KitKat en 2013
- Lollipop en 2014
- Marshmallow en 2015
- Nougat en 2016
- Oreo en 2017
- Pie en 2018

Désormais, Android équipe aussi les tablettes tactiles, les téléviseurs, les netbooks, les cadres photos ou encore les autoradios.

4.2 CARACTERISTIQUES

Le Tableau ci-dessous résume les caractéristiques d'Android

Framework	Framework Java pour le développement d'application pour la plateforme Android
Machine virtuelle Dalvik	Machine virtuelle spécialement développée pour Android. Cette machine virtuelle permet d'exécuter les applications java développées avec le Framework.
Navigateur web	Navigateur web basé sur le moteur de rendu Webkit
Graphique	Librarie graphique 2D, librarie graphique 3D basé sur OpenGL ES 1.0. Accélération matériel possible.
Stockage	Base de données SQL : SQLite est utilisé pour le stockage des données
Média	Android supporte les formats audio/vidéo/image suivants : MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF
Connectivité	GSM, EDGE, 3G, Bluetooth, Wi-Fi
Support Matériel	Android est capable d'utiliser Camera, GPS, accéléromètre
Environnement de développement	Android possède un environnement de développement complet contenant : un émulateur, un débuggeur, un analyseur de mémoires et de performances et un plugin eclipse.

Tableau 3: Caractéristiques de système d'exploitation Android

4.3 ARCHITECTURE

Android bénéficie d'une architecture en couche complète faisant de lui une plateforme riche, dédiée aux appareils mobiles.

Il est à base de noyau linux profitant des services système de base tels que la sécurité, la gestion mémoire, gestion de processus, etc.

À un niveau supérieur se trouvent un ensemble de librairies écrites en C/C++ jouant le rôle d'un middleware (on en cite le système de bibliothèque C, les médiathèques, le SGL, etc.). C'est sur cette couche que se greffe l'Android Runtime, comprenant la machine virtuelle java et ses bibliothèques.

Vient ensuite la plateforme logicielle, nommée aussi Framework de développement, écrite en java et permettant de mutualiser les ressources entre applications Java. Elle offre aux développeurs la possibilité de produire des applications diverses et innovantes à travers un ensemble d'API.

Enfin, et à un niveau supérieur se situe un ensemble d'applications sous forme de paquets «. APK ».

Les applications fournies par Android sont telles qu'un navigateur web, un client mail, un calendrier, un gestionnaire de contacts, etc.

Cette architecture est synthétisée sur la figure suivante :

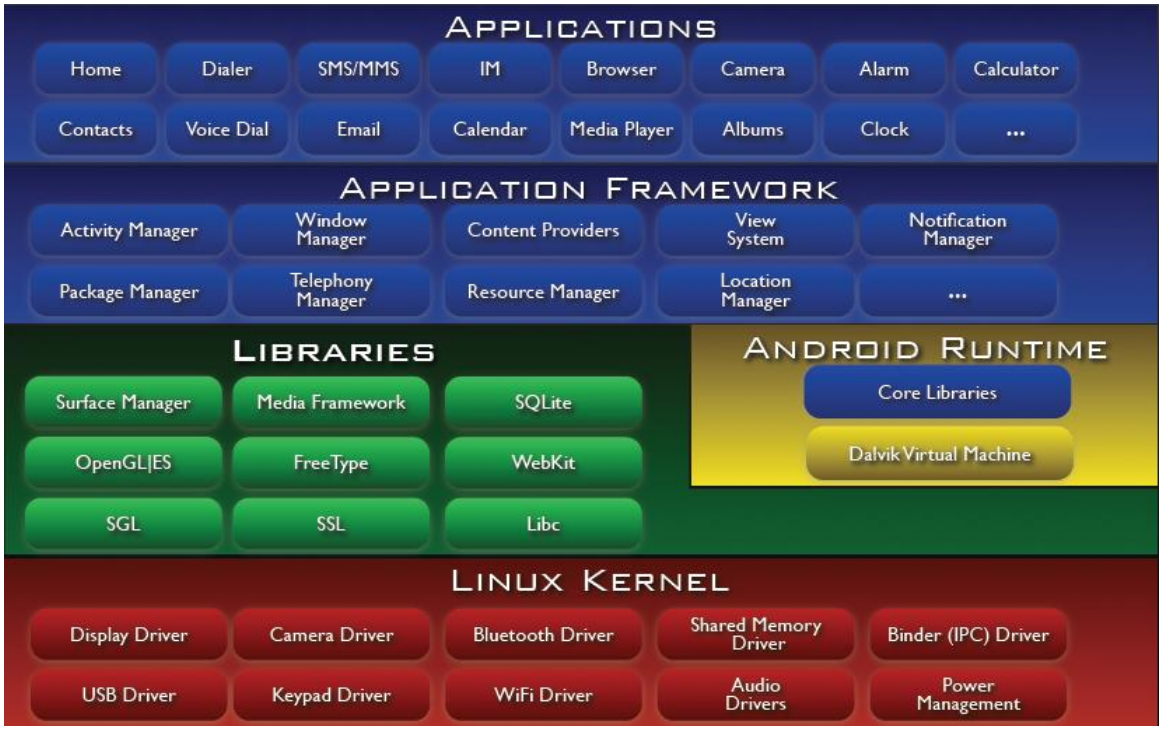


Figure 4: L'architecture complète d'Android

5. DEVELOPPEMENT APPLICATION IOS

5.1 DEFINITION

iOS est un système d'exploitation mobile créé et développé par Apple. Il se fait appeler à l'origine iPhone OS, mais il a été renommé iOS en juin 2009. L'iOS fonctionne actuellement sur l'iPhone, l'iPod touch et l'iPad. Reconnu pour sa fluidité, son ergonomie et son intuitivité : c'est le système d'exploitation le plus abouti à ce jour. Il dispose du portail App Store, avec un catalogue de plus de 200 000 applications, il s'est imposé comme une référence parmi les kiosques d'applications mobiles.

5.2 LISTE DES VERSIONS

- **iPhone, iPod touch et iPad**
  - iPhone OS 1 (6 mars 2018)
  - iPhone OS 2 (11 juillet 2008)
  - iPhone OS 3 (17 juin 2009)
  - iOS 4 (21 juin 2010)
  - iOS 5 (6 juin 2011)
  - iOS 6 (11 juin 2012)
  - iOS 7 (10 juin 2013)
  - iOS 8 (2 juin 2014)
  - iOS 9 (8 juin 2015)
  - iOS 10 (13 juin 2016)
  - iOS 11 (5 juin 2017)
  - iOS 12 (4 juin 2018).
- **Apple TV**
  - iOS 4 (1 septembre 2010)
  - iOS 5 (12 octobre 2011)
  - iOS 6 (24 septembre 2012)
  - iOS 7 (23 septembre 2013).

5.3 LES DISPOSITIFS MOBILES COMPATIBLES

Seuls les appareils mobiles d'Apple peuvent fonctionner sous le système d'exploitation iOS :

- iPhone 4, 5, 6, etc.,
- iPad, iPad Mini,
- iPod nano, iPod shuffle.

5.4 CARACTERISTIQUES

- iOS est un système d'exploitation développé par Apple.
- Il est une variante de Mac OS X, qui est le système d'exploitation pour les ordinateurs de marque Apple. Il est basé sur Unix.
- L'installation d'un matériel autre que celui d'iOS est impossible.
- Bien qu'il s'agisse d'un système d'exploitation exclusif pour les appareils Apple, les développeurs peuvent librement apporter des améliorations à son SDK.
- A moins de jailbreaker son smartphone, l'installation des applications tierces n'est pas possible.
- L'utilisation d'Adobe Flash n'est autorisée qu'à partir d'iOS 8, mais les utilisateurs d'iPhone, d'iPod Touch ou d'iPad avec une version antérieure à iOS 8 ne peuvent pas profiter de cette mise à jour.

5.5 ARCHITECTURE

Lorsque nous développons des applications pour iPhone, Apple ne nous permet pas d'accéder directement à ces équipements. En fait, toutes les interactions matérielles ont lieu exclusivement via plusieurs couches de logiciels qui servent d'intermédiaires entre les applications et le matériel de l'appareil. Ces couches constituent ce que l'on appelle un système d'exploitation. Dans le cas de l'iPhone, ce système d'exploitation est appelé iPhone OS. Afin de mieux comprendre l'environnement de développement de l'iPhone, il est intéressant d'examiner en détail les différentes couches qui composent l'OS iPhone.

Comme mentionné précédemment, le système d'exploitation iPhone comprend un certain nombre de couches logicielles différentes, chacune fournissant des cadres de

programmation pour le développement d'applications s'exécutant sur le système d'exploitation.
Ces couches du système d'exploitation peuvent être présentées sous forme de diagramme, comme illustré dans la figure suivante :

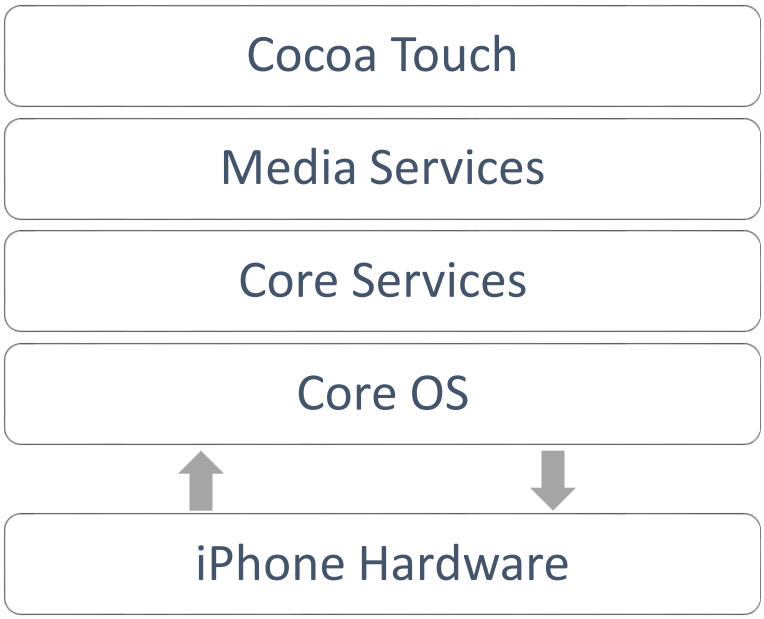


Figure 5: Architecture du système d'exploitation iOS

- **Cocoa Touch** : Il s'agit d'une couche supérieure de la pile du système d'exploitation de l'iPhone et contient les Framework les plus couramment utilisés par les développeurs d'applications iPhone. Cocoa Touch est principalement écrit en Objective-C et est basé sur le standard Mac OS X Cocoa API.
- **Media Service** : C'est la deuxième couche à partir du haut de la pile. Il fournit au système d'exploitation iPhone des fonctionnalités audio, vidéo, d'animation et de graphisme. Comme pour les autres couches de la pile du système d'exploitation iPhone, la couche Multimédia comprend un certain nombre de Framework pouvant être utilisés lors du développement d'applications iPhone.
- **Core Services** : C'est la troisième couche à partir du haut de la pile. La couche iPhone Core Services fournit en grande partie la base sur laquelle les couches ci-dessus sont construites.

- **Core OS** : La couche Core OS est la couche inférieure de la pile de systèmes d'exploitation iPhone et repose directement sur le matériel du périphérique. Cette couche fournit une variété de services, y compris la mise en réseau de bas niveau, l'accès à des accessoires externes et les services de base du système d'exploitation tels que la gestion de la mémoire, la gestion du système de fichiers et les threads.
- **iPhone Hardware** : Les périphériques matériels sont gérés par iPhone OS et fournissent les technologies nécessaires à la mise en œuvre d'applications natives sur le téléphone. Le système d'exploitation est livré avec plusieurs applications système telles que Mail, Safari, Phone, qui fournissent des services standard à l'utilisateur.

5.6 OUTILS

Depuis mars 2008, tout développeur peut utiliser gratuitement le kit de développement d'applications mobiles pour iPhone et iPod Touch. Cependant, pour pourvoir ensuite utiliser l'application sur les appareils et les vendre, il faut s'inscrire à l'iPhone Developer Program, qui est payant. Le développement d'applications mobiles pour iPhone, iPod Touch et iPad se fait uniquement sur Mac avec l'application Xcode.

6. DIFFERENCES ENTRE ANDROID ET IOS

6.1 DIFFERENCES TECHNIQUES

- **Langage de programmation** : Les système d'exploitation Android et iOS sont chacun programmés dans des langages de programmation différents. Les applications Android fonctionnent sur Java / Kotlin, tandis que les applications iOS fonctionnent sur Objective-C / Swift.
- **La phase de test de l'application** : Le contrôle d'une application est une étape essentielle du processus de développement d'une application mobile, tout comme les logiciels. S'assurer que l'application développée fonctionne parfaitement est indispensable. Le simulateur iOS et l'émulateur Android sont des exemples souvent utilisés pour effectuer ces tests. La première différence frappante est la rapidité

supérieure du simulateur iOS par rapport à son homologue Android. L'émulateur Android a par contre l'avantage d'être une machine virtuelle efficace avec un CPU qui le rend plus réaliste que son équivalent iOS. D'ailleurs, les simulateurs iOS ne transmettent souvent pas les représentations précises et réalistes des appareils Apple. C'est pour cette raison qu'il est primordial d'effectuer de nombreux tests sur de véritables appareils mobiles afin d'observer l'interactivité de l'application et de détecter d'éventuels bugs.

- **L'interface graphique** : C'est ici que se trouvent les différences visuelles lors de la création d'une application destinée à un utilisateur Android ou iOS. D'un point de vue technique, le développeur va construire des fichiers XML sur les interfaces Android. Ces fichiers sont très similaires aux fichiers XIB d'iOS. En ce qui concerne les animations, L'iOS est beaucoup mieux préparé qu'Android. Apple, mettant un accent particulier sur l'esthétique et l'expérience utilisateur, s'est toujours efforcé de proposer des animations souples, complexes et puissantes. Google pour sa part a été plus préoccupé par les paramètres de son hardware, mais a tout de même essayé d'améliorer les performances graphiques de ses dispositifs.
- **Le bouton « Retour »** : Une autre différence évidente est ce bouton de « retour », que l'on retrouve chez Android et qui n'existe pas chez iOS. C'est un petit détail qui, pourtant, conditionne le développement d'application. Ce bouton de retour sert également à la navigation dans l'environnement Android. En l'absence de ce bouton, le développeur de l'application iOS doit trouver un autre moyen d'indiquer sur l'écran comment revenir en arrière ou s'il existe une autre manière d'effectuer cette action en utilisant un autre flux d'interaction.

6.2 DIFFERENCES MARKETING

- **Le coût d'une application mobile** : Le processus de développement sur iOS coûte légèrement plus cher, simplement parce qu'il y a moins de demande chez iOS que chez Android. Ajoutons à cela que les exigences attendues sur un produit iOS sont aussi beaucoup plus élevées. Il faut également s'équiper d'un smartphone dont le prix s'élève en moyenne à 80 000,00 DA et d'un Mac à minimum 120 000,00 DA. Le coût total revient alors à environ 200 000,00 DA.
- **Retours d'investissements** : Nous pouvons remarquer que les applications iOS ont plutôt tendance à faire payer les utilisateurs pour les téléchargements et les services

payant incorporés, tandis que les applications Android sont habituellement gratuites et deviennent rentables par le biais des publicités.

7. CONCLUSION

Dans ce chapitre nous avons vu ensemble comment une application mobile est développée et les outils utilisés IDE, Framework …etc. Ensuite nous avons détaillé les deux systèmes d'exploitations mobiles Android et iOS, puis on a terminé par une comparaison des deux systèmes.

Aujourd'hui pour une PME, le développement d'une application mobile pour chaque plateforme est impossible pour des raisons de coût ou de délais, le choix de la plateforme se pose à nouveau.

Ces derniers temps, une bonne nouvelle s'annonce aux PME, Google a lancé un SDK de développement d'applications mobiles à code source ouvert. Il est utilisé pour développer des applications natives pour Android et iOS à partir d'une base de code unique. De l'autre côté, il y a aussi Microsoft qui a lancé « Xamarin » et Facebook qui lance son propre Framework « ReactNative » …etc.

Mais le choix de la plateforme reste difficile, Il y a ceux qui optent pour des Web App ou pour un site web optimisé pour mobile, des solutions alternatives qui peuvent marchées. Même si ces solutions comportent de nombreuses limites notamment en ce qui concernent les fonctionnalités du produit final, elles ont l'avantage de permettre une présence sur tous les smartphones, quel que soit leurs systèmes d'exploitations, mais avec Flutter, Xamarin et ReactNative le multiplateforme est assuré.

Le chapitre suivant portera sur l'analyse et la conception d'une application pour une banque en ligne. Nous allons proposer une solution informatique pour répondre aux exigences et aux besoins d'une telle application.

Chapitre 3
Analyse et conception

1. INTRODUCTION

La banque en ligne est l'une des plus grandes révolutions du monde de la banque. Que ce soit pour les établissements bancaires ou les clients, les codes du domaine bancaire ont changé et désormais la banque en ligne apparaît comme une réelle alternative à la banque traditionnelle.

Avec les banques en ligne, à distance et à n'importe quelle heure, le client peut surveiller ses comptes, effectuer des virements entre ses comptes épargne et votre compte courant, surveiller ses activités... De plus, des applications sur téléphone mobile ou tablette sont disponibles et elles offrent au client la possibilité de gérer ses finances sans avoir besoin d'un ordinateur.

Ce chapitre est consacré à l'étude des étapes fondamentales pour le développement d'une application de banque en ligne. Avant l'implémentation et la réalisation de toute application informatique, il convient de suivre une démarche méthodologique pour mettre en place cette dernière afin de modéliser les fonctionnalités que doit offrir notre système et représenter son architecture ainsi que les interactions entre ses différents composants.

2. LES OBJECTIFS DE L'APPLICATION

Le principal objectif de notre travail est de développer une solution informatique qui permet de répondre aux besoins d'une banque en ligne, une solution composée de deux parties :

- Une partie client qui sera développée en version web et mobiles (Android et iOS) :
  - Ouverture d'un compte en ligne et obtention d'une carte de crédit.
  - Consultation des transactions.
  - Effectuer des virements.
  - Consultation et gestion des comptes.
  - Consultation et gestion des cartes de crédits.
  - Consultation et gestion des informations personnelles (profile).
  - Consultation et gestions des factures (virement périodique).
  - Statistiques sur l'activité de client.
- Une partie administrateur qui sera développée en version web responsive adapter au mobile :
  - Consultation et gestion des clients.
  - Consultation et gestion des comptes.
  - Consultation et gestion des cartes de crédits.
  - Consultation et gestion des messages des visiteurs.
  - Statistiques sur la banque.

Mais avant de commencer notre analyse sur la conception d'une banque, on doit savoir d'abord qu'est-ce que c'est une banque en ligne, quelles sont les services qu'elle propose, et quelles sont les avantages et les inconvénients d'une telle banque ?

3. BANQUE EN LIGNE

3.1 DEFINITION

Une banque en ligne est une banque accessible par internet. Sa disponibilité permet l'accès aux comptes, d'effectuer des transactions ou d'obtenir des informations

financières récentes. Son accès peut s'effectuer depuis un Guichet Automatique Bancaire (GAB), via un ordinateur ou grâce à un smartphone via Internet. La connexion se fait par accès sécurisé.[8]

3.2 SERVICES

- Ouverture d'un compte bancaire en ligne.
- Détention d'une carte de paiement (le plus souvent, les banques en ligne la proposent gratuitement).
- Détention d'un chéquier.
- Obtention de crédits.
- Placements bancaires et financiers.

Avec une banque 100 % en ligne, les clients ne disposent pas de guichet ni d'agence physique. Toutes les relations avec la banque se passent en ligne que ce soit par :

- Mail.
- E-chat.
- Chat vidéo.
- Forum de discussion, etc.

L'absence d'agence physique représente une certaine difficulté pour les dépôts d'espèce, en revanche, pour les dépôts le client peut effectuer cette action par le billet des GAB.

[8] Depuis source

4. GENERALITES SUR UML

Durant les années 70 jusqu'à 90, un bon nombre de de méthodes objets ont été développées, nous citons la méthode OMT (Object Modeling Technique) de Rambaugh BOOCH de G BOOCH ainsi qu'Objectory d'Ivar Jacobson, mais l'inconvénient majeur de toutes ces méthodes était leur divergence sur certains concepts. C'est de la que naquit le besoin de standardisation qui a conduit à l'apparition de l'UML (Unified Modeling Language) suite à des travaux d'unification.

4.1 LES DIAGRAMMES D'UML

Le langage UML est décrit par un méta-modèle, c'est-à-dire une représentation simple, mais rigoureuse de ses constituants élémentaires, avec leurs règles d'utilisation et de syntaxe.

Un diagramme est défini par les acteurs UML comme la représentation graphique d'un ensemble sélectionné de constituants UML. Ces derniers ont élaboré treize diagrammes chacun d'eux étant dédié à la représentation des concepts particulier d'un système logiciel.

Une classification courante distingue les diagrammes qui traduisent la structure et les diagrammes qui présentent le comportement du système étudié.

4.1.1 Diagrammes structurels ou statiques

4.1.1.1 Les diagrammes structurels ou statiques :

- Diagramme de classes.
- Diagramme d'objets.
- Diagramme de composants.
- Diagramme de déploiement.
- Diagramme de paquetages.
- Diagramme de structure composite.

4.1.1.2 Diagrammes comportementaux :

- Diagramme des cas d'utilisation.
- Diagramme d'activité.
- Diagramme états-transactions.

4.1.1.3 Diagrammes d'interaction ou dynamiques

- Diagramme de séquence.
- Diagramme de communication.
- Diagramme global d'interaction.
- Diagramme de temps.

Parmi ces diagrammes nous avons choisi les quatre suivants pour la conception de notre application :

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

5. ANALYSE

L'objectif de cette partie est la spécification de manière claire et concise de l'application. On va de ce fait procéder à la définition des utilisateurs, ainsi que leurs besoins respectifs auxquels l'application est tenue d'apporter des réponses, ainsi que l'identification des acteurs principaux et les cas d'utilisations relatifs à chacun d'eux, puis l'illustration des diagrammes de cas d'utilisation et quelques scénarios associés aux acteurs.

5.1 DEFINITIONS

5.1.1 Acteur

Un acteur représente un ensemble cohérent de rôles que peut jouer l'utilisateur (entité externe) avec le système, et interagit avec celui-ci en fournissant de l'information en entrée et/ou la reçoit en sortie.

5.1.2 Cas d'utilisation

Un cas d'utilisation est un texte qui décrit l'interaction et les dialogues entre l'acteur et le système. Les cas d'utilisations sont une technique puissante pour consigner et traduire le comportement détaillé du système.

5.1.3 Scénario

Représente une succession particulière d'enchainements s'exécutant du début jusqu'à la fin du cas d'utilisation. Un enchainement étant l'unité de description de séquences d'actions. Un ensemble de scénarios pour un cas d'utilisation identifie tout ce qui peut arriver lorsque ce cas d'utilisation est mis en œuvre.

5.2 SPECIFICATION DES BESOINS

La spécification des besoins doit décrire sans ambiguïté le système logiciel à développer, l'expression des besoins doit proposer ce que le système devrait faire. Notre projet porte sur la conception d'une application Client/serveur banque en ligne, elle doit permettre :

- Visiteur
    - S’inscrire (devenir client).
    - Contacter l’administrateur.
- Client
    - Ouvrir un compte.
    - Demander une carte de crédit.
    - Consulter et modifier ses informations personnelles.
    - Consulter ses comptes.
    - Consulter et gérer ses cartes de crédits.
    - Geler sa carte.
    - Consulter ses opérations.
    - Consulter ses statistiques.
    - Effectuer, programmer des virements entre compte
    - Transferts inter-compte (virement interne).
    - Contacter l’administrateur.
- Administrateur
    - Gérer les clients.
    - Gérer les comptes.
    - Gérer les cartes de crédits.
    - Consulter les profiles des clients et leurs opérations.
    - Gérer les messages des visiteurs.
    - Gérer les messages des clients.
    - Gérer les utilisateurs de l’API.
    - Consulter des statistiques.

5.3 IDENTIFICATION DES ACTEURS

Un acteur : représente un élément externe qui interagit avec un système, dans le but de le faire fonctionner et d'en tirer profit. Généralement il s'agit d'un utilisateur du système ou bien d'une entité qui coopère avec le système, dans notre application les acteurs qui interagissent avec le système sont les visiteurs, les clients et l’administrateur de la banque.

- **Le visiteur :** C'est toute personne pouvant consulter le site dans le but d'avoir des renseignements.
- **Le client :** la personne qui s'inscrit auprès de la banque et qui a le droit d'accéder à l'espace client après authentification pour consultation ou pour effectuer des opérations.
- **L'administrateur :** est la personne qui a le droit d'accéder à l'espace administrateur, son rôle est la gestion de la banque.

5.4 DIAGRAMME DE CONTEXTE

Le diagramme de contexte permet de spécifier le nombre d'instances d'acteurs connectés à ce système à un moment donné. La définition des besoins présentée ci-dessus nous a permis d'extraire le diagramme de contexte suivant :

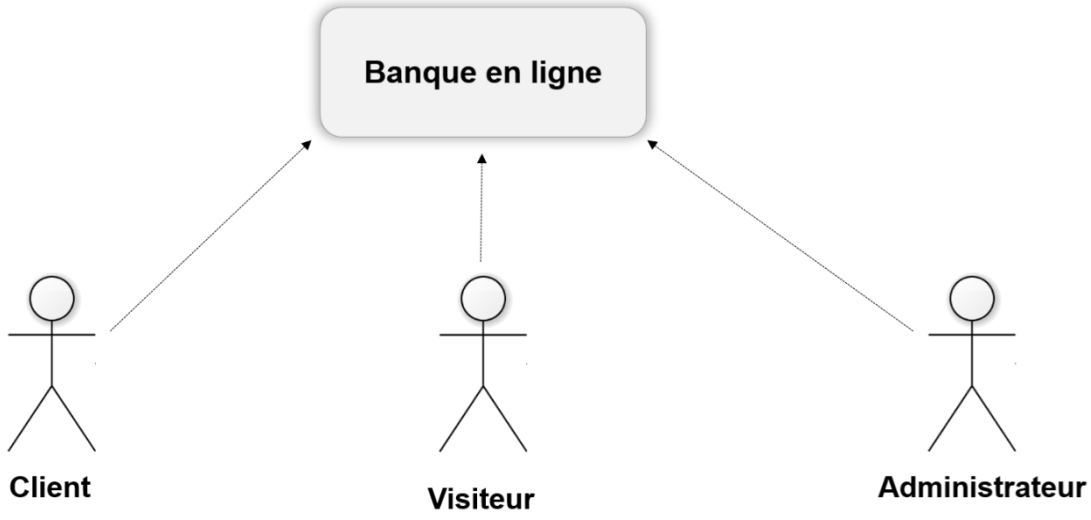


Figure 6: Diagramme de contexte

5.5 IDENTIFICATION DES CAS D'UTILISATIONS

Pour chaque acteur identifié précédemment, il convient de rechercher les différentes intentions selon lesquelles il utilise le système ou bien son rôle dans le système. Le tableau suivant récapitule les cas d'utilisation de chaque acteur de notre application.

Acteur	Besoins	Cas d'utilisations
Visiteur	S'inscrire (devenir client)	➢ Recueillir les données de l'état civile, cordonnées et profession de client. ➢ Ouvrir un compte. ➢ Attribuer une carte de crédit.
	Contacter l'administrateur	➢ Envoyer un message à l'administrateur
Client	Gestion du profil	➢ Consulter le profil. ➢ Modifier les informations personnelles.
	Gestion des comptes	➢ Consulter les informations des comptes. ➢ Imprimer le RIB[9]. ➢ Demander un nouveau compte.
	Gestion des cartes de crédits	➢ Consulter les informations des cartes de crédits. ➢ Demander la modification de code. ➢ Geler/Activer une carte. ➢ Définir et modifier le plafond d'une carte. ➢ Valider une carte (en saisissant le CVV[10]). ➢ Demander une nouvelle carte de crédit.

[9] **RIB :** Relevé d'Identité Bancaire est un document fourni par la banque qui permet l'identification du titulaire d'un compte bancaire.

[10] **CVV :** (Card Validation Value) Code de validation est une caractéristique de sécurité pour les cartes de crédit composé de 3 chiffres qu'on trouve au dos de la carte. Ce code permet de vérifier si la carte est bien en votre possession lors de la transaction parce que ça permet de constater si une carte de crédit est physiquement présente.

		➢ Demander le renouvellement d'une carte.
	Consulter des transactions	➢ Consulter les transactions. ➢ Faire une recherche avancé.
	Effectuer des virements	➢ Effectuer un transfert (inter-compte). ➢ Effectuer un virement multi-comptes (entre-comptes, solde minimal autorisée est 1 000 DA). ➢ Programmer un virement.
	Afficher des statistiques	➢ Consulter des statistiques sur l'activité de client.
	Contacter l'admin	➢ Contacter l'admin
	Consulter l'aide	➢ Consulter le guide d'utilisation
Administrateur	Gestion des clients	➢ Consulter la liste des clients. ➢ Rechercher un client. ➢ Bloquer et débloquer un client. ➢ Valider un nouveau client. ➢
	Gestion des comptes	➢ Consulter la liste des comptes. ➢ Rechercher un compte. ➢ Bloquer et débloquer un compte.

		➢ Valider un nouveau compte (doit être alimenté d'un montant minimal de 5 000.00 DA plus autorisation de la banque).
	Gestion des cartes de crédits	➢ Consulter la liste des cartes. ➢ Rechercher une carte. ➢ Bloquer et débloquer une carte. ➢ Valider une carte.
	Gestion service client	➢ Consulter la liste des messages des clients et visiteurs. ➢ Répondre aux messages. ➢ Ignorer des messages. ➢ Bloquer clients et visiteurs.
	Gestion client API	➢ Consulter la liste des clients. ➢ Ajouter un client. ➢ Modifier un client. ➢ Bloquer un client.
	Afficher des statistiques	➢ Consulter des statistiques sur la banque.

Tableau 4: Cas d'utilisations pour chaque acteur

5.6 DIAGRAMME DE CAS D'UTILISATION

Tout système peut être décrit par un certain nombre de cas d'utilisation correspondant aux besoins exprimés par l'ensemble des utilisateurs. À chaque utilisateur, vu comme acteur, correspondra un certain nombre de cas d'utilisation du système. L'ensemble de ces cas d'utilisations sont représentés sous forme d'un diagramme.

5.6.1 Diagramme de cas d'utilisation général « client et visiteur »

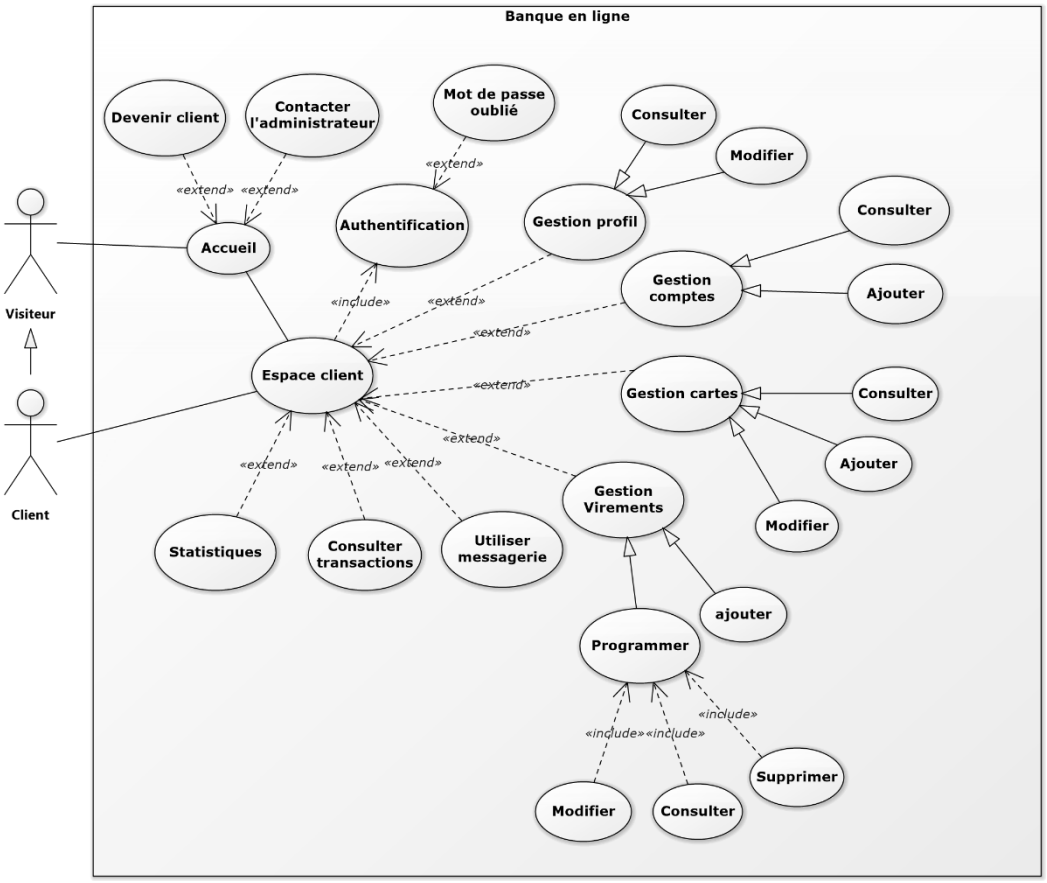


Figure 7: Diagramme de cas d'utilisation « client et visiteur »

5.6.2 Diagramme de cas d'utilisation général « administrateur »

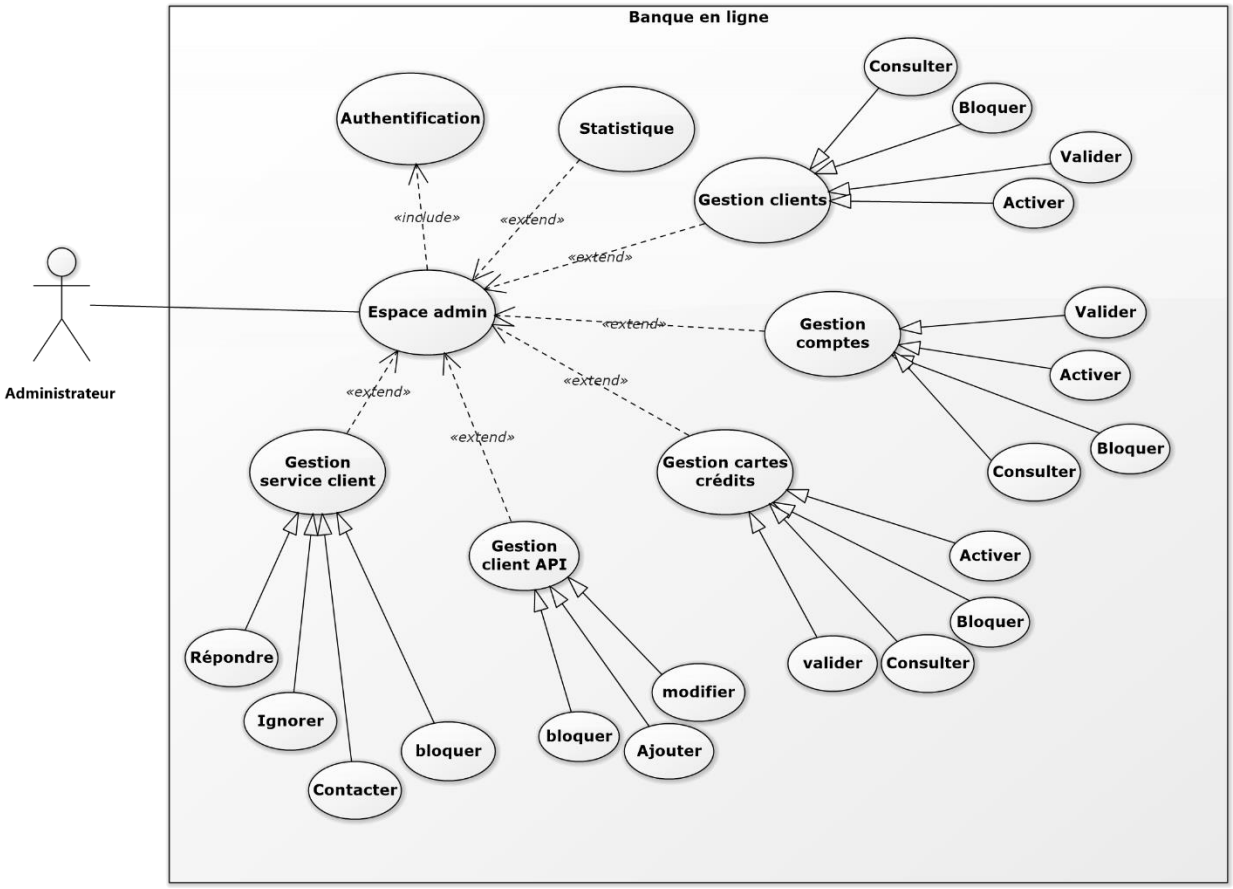


Figure 8: Diagramme de cas d'utilisation «administrateur»

5.6.3 Quelques diagrammes de cas d'utilisation détaillé

5.6.3.1 Description de cas d'utilisation « inscription »

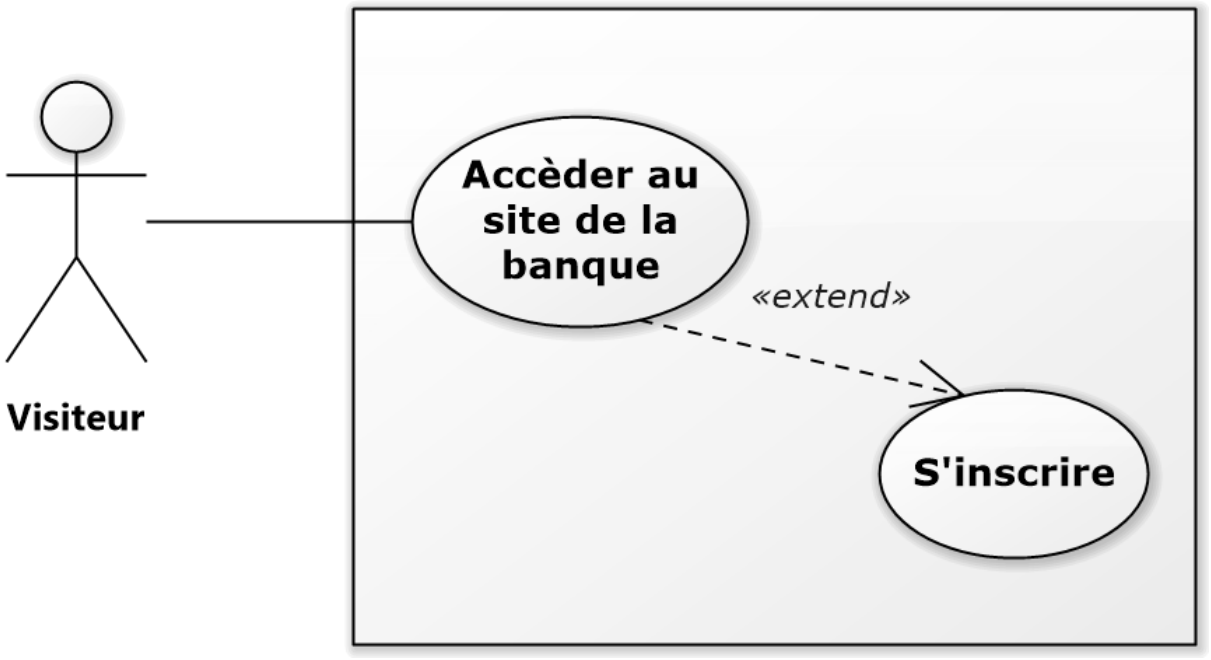


Figure 9: Diagramme de cas d'utilisation « inscription »

Scénario nominal :

- L'utilisateur accède au site de la banque
- Appuie sur le bouton devenir client
- Remplit le formulaire d'inscription, pour le justificatif il doit uploader soit :
    - Pièce d'identité national
    - Permis de conduire
    - Passeport.
- Il appuie sur le bouton « Terminer » et à la fin de l'inscription, une boite modale s'affiche à l'écran lui demandant de saisir un code à 4 chiffres qu'il recevra dans sa boite mail.
- Une fois fait, il sera invité à lire les conditions d'utilisations et valider son inscription.

- Si toutes les conditions sont satisfaites, un message s'affiche lui indiquant la réussite de l'inscription, et qu'il doit attendre la confirmation de l'admin[11] pour utiliser son compte.

5.6.3.2 Description de cas d'utilisation « Commander une carte »

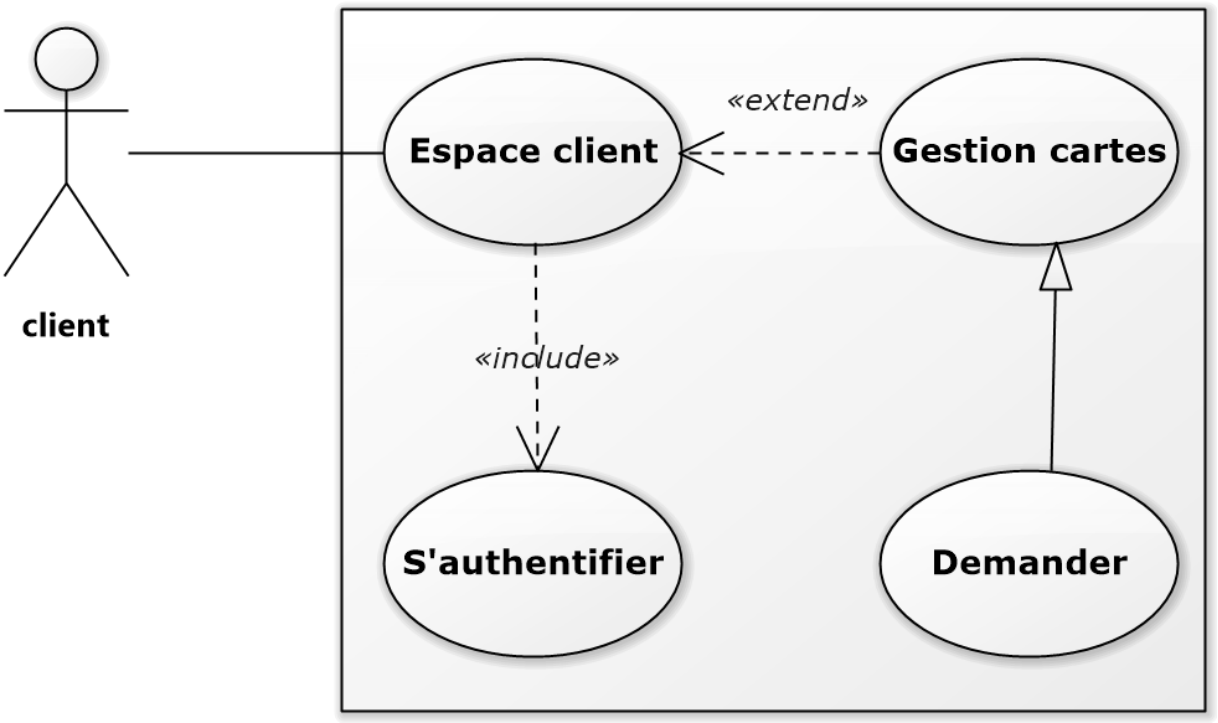


Figure 10: Diagramme de cas d'utilisation « Commander une carte »

Scénario nominal :

- L'utilisateur accède à l'onglet « Mes cartes ».
- Appuie sur le bouton « Ajouter carte ».
- Une boite modale s'affiche à l'écran lui demande de choisir le compte pour lequel il veut ajouter une carte.
- Il sélectionne un de ses comptes, et saisi son mot de passe.
- Appuie sur le bouton « valider », si le compte possède deux cartes, un message d'avertissement lui sera affiché et l'opération s'échoue sinon l'opération s'effectuera avec succès et l'utilisateur sera invité à continuer le reste de la procédure après la validation l'admin.
- Une fois reçue, il accède à l'onglet « Mes cartes ».

[11] **Admin :** veut dire administrateur

- Sectionne la nouvelle carte.
- Appuie sur le bouton « valider la carte » (cas application web) ou double clic sur la carte (cas application mobile).
- Saisit le CVV qui se trouve sur le verso de la carte.
- Appuie sur « valider ».

6. CONCEPTION

La conception est l'étape qui suit l'analyse. Elle consiste à modéliser et à détailler tous les éléments de modélisation issus de la phase analyse.
Dans cette étape, nous illustrons les diagrammes de séquences de quelque cas d'utilisations et le diagramme de classe.

6.1 DIAGRAMME DE SEQUENCE

6.1.1 Définition

Le diagramme de séquence représente des échanges de messages entre objets. Il permet de représenter un processus de façon simplifiée, en se centrant sur les échanges entre les acteurs ou entre acteurs et le système d'information.
Parmi les objets utilisés, on a utilisé l'objet interface qui représente l'interface entre l'acteur et le système tels que les pages web ou les écrans de saisie.

Dans ce qui suit nous allons traduire quelques scénarios en diagrammes de séquences :

- Valider client
- Ouvrir un nouveau compte

6.1.2 Diagramme de séquence de cas d'utilisation « Valider client »

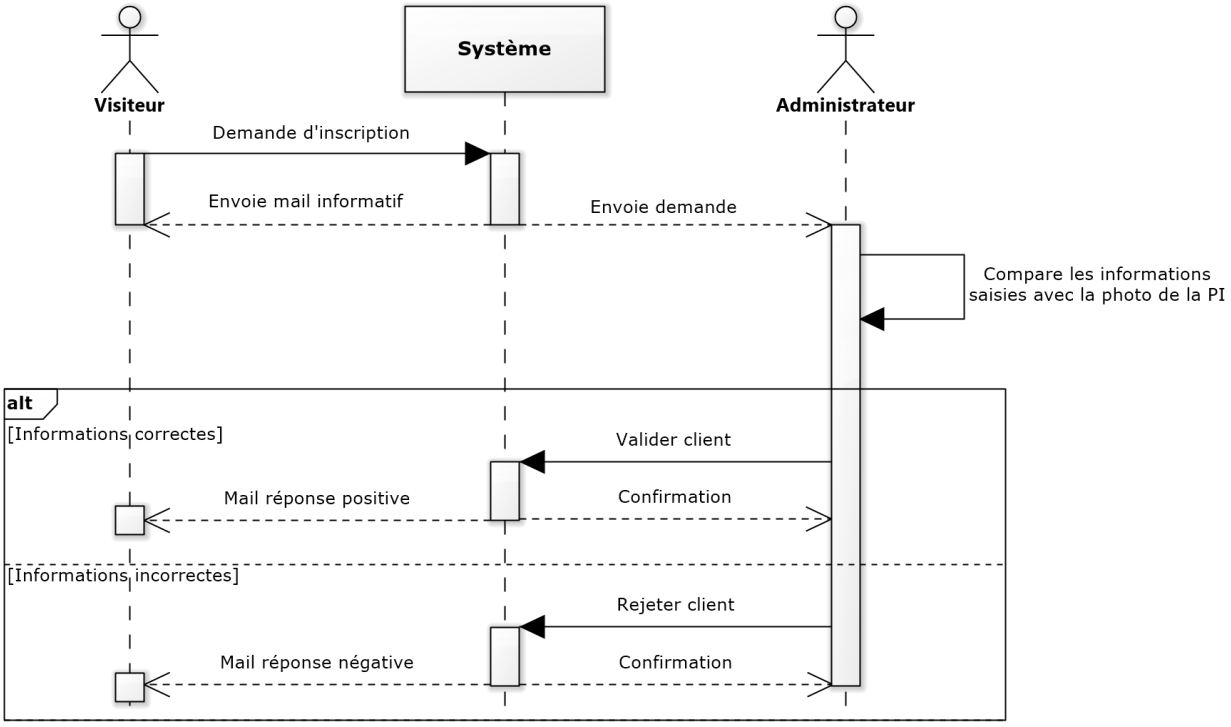


Figure 11: Diagramme de séquence de cas d'utilisation « Valider client »

Scénario nominal :

- Le visiteur envoie sa demande d'inscription
- Le système lui envoie un mail informatif et affiche un message de confirmation d'inscription puis le redirige vers l'accueil si les informations sont valides.
- L'admin accède à « valider client », la demande d'inscription s'affiche, il compare les informations saisies avec la photo de la pièce d'identité.
- Si les informations sont correctes, l'admin valide la demande, sinon, la rejette.
- Un message de confirmation sera affiché et un mail informatif de la réponse de l'admin sera envoyé au client.

6.1.3 Diagramme de séquence de cas d'utilisation « Ouvrir un nouveau compte »

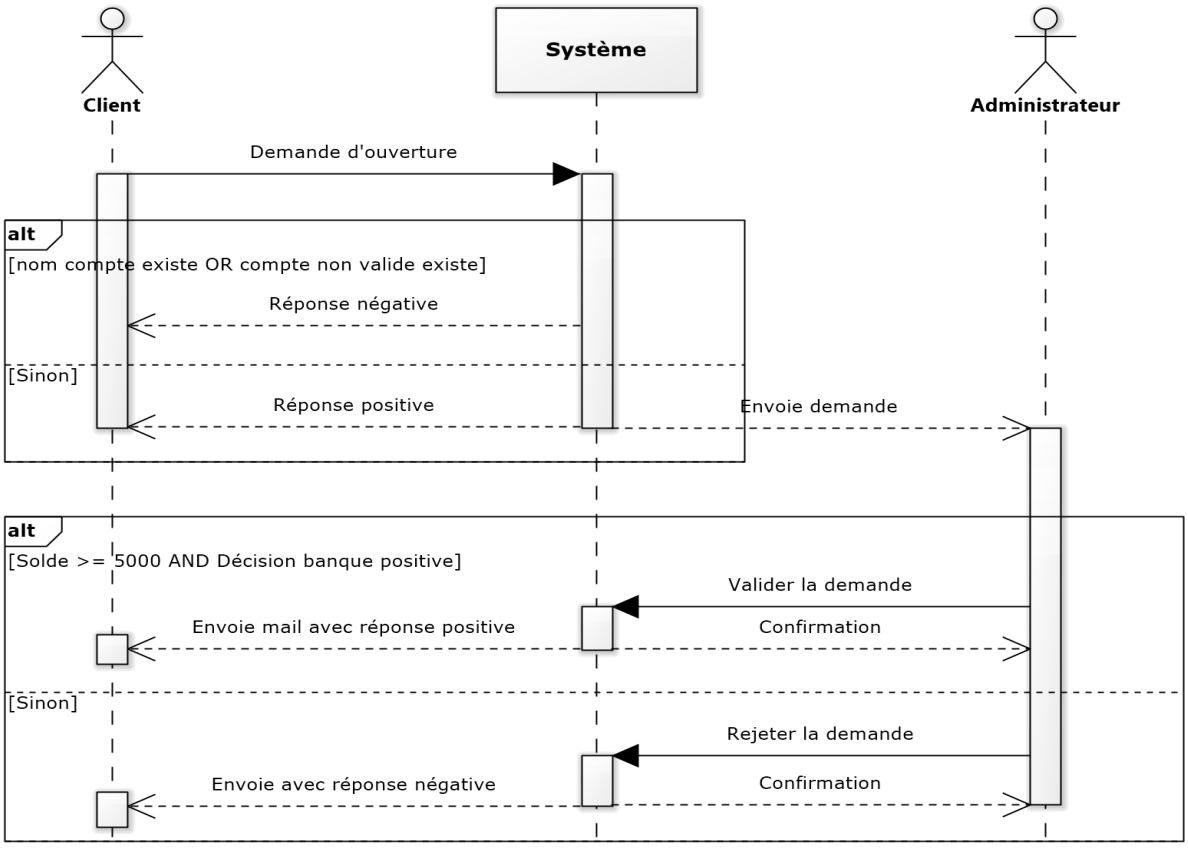


Figure 12: Diagramme de séquence de cas d'utilisation « Ouvrir un nouveau compte »

Scénario :

- Le client envoie une demande d'ouverture d'un nouveau compte.
- Le système vérifie les informations, si le nom est dupliqué ou le client possède un compte pas encore validé, la demande sera rejetée, sinon le compte sera créé.
- La demande sera envoyée à l'admin.
- Si l'admin estime que le client a besoin d'un nouveau compte et le compte est alimenté d'un montant supérieur ou égale à 5 000 DA, l'admin le valide sinon la demande sera rejetée.

6.2 DIAGRAMME D'ACTIVITES

6.2.1 Définition

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

Dans ce qui suit nous allons traduire quelques scénarios en diagrammes de d'activités :

- Débloquer carte
- Effectuer un virement

6.2.2 Diagramme d'activités du cas d'utilisations « Débloquer carte »

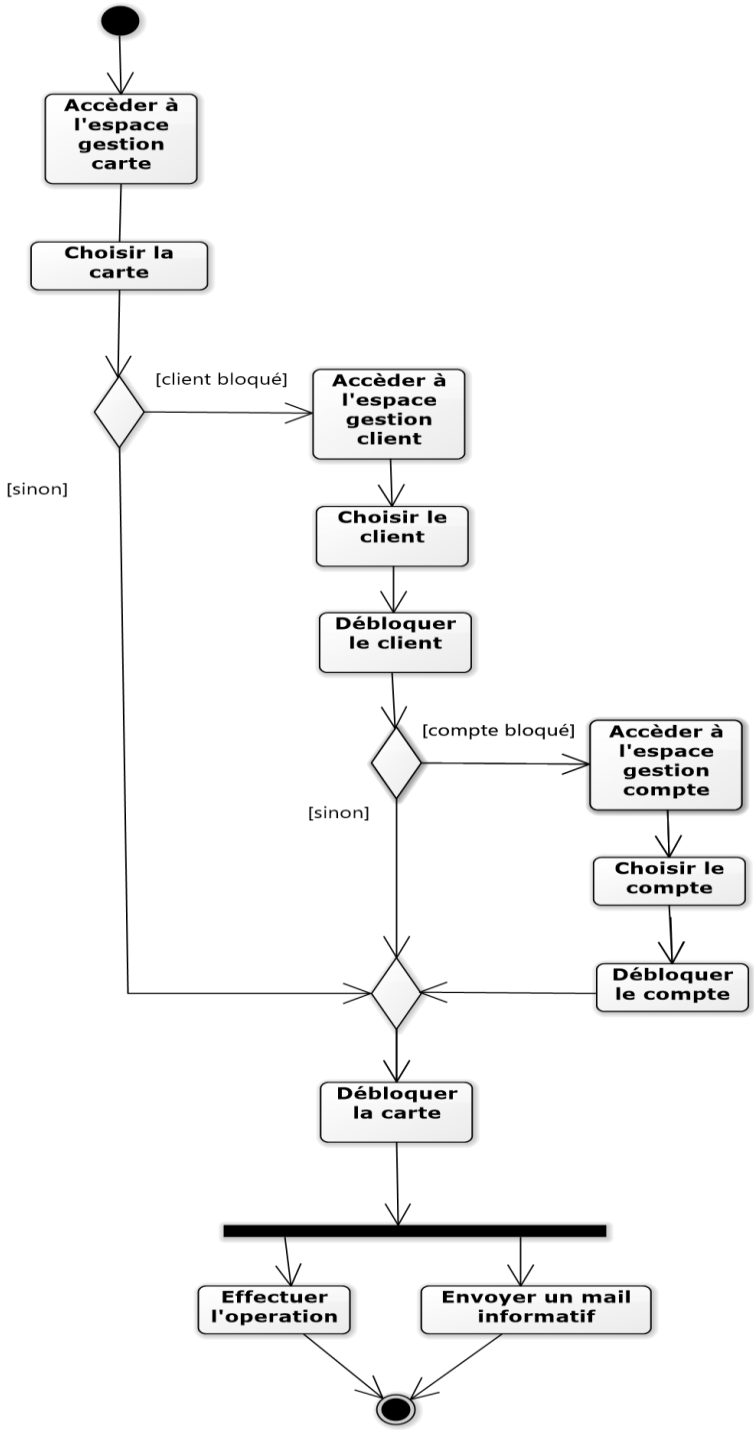


Figure 13: Diagramme d'activités du cas d'utilisation « Débloquer carte »

6.2.3 Diagramme d'activités du cas d'utilisation « Effectuer virement »

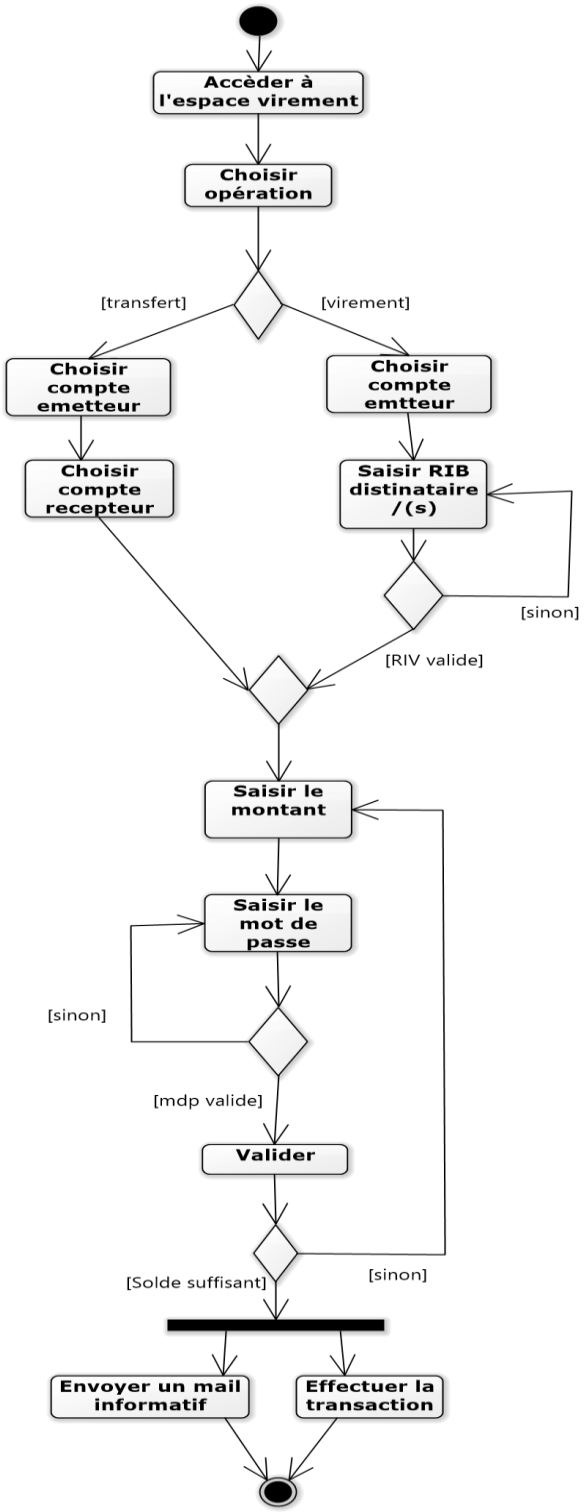


Figure 14: Diagramme d'activités du cas d'utilisation « Effectuer virement »

6.3 DIAGRAMME DE CLASSE

Le diagramme de classe montre la structure interne du système. Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation.

L'intérêt du diagramme de classe est de modéliser les entités du système d'information, il permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées (regroupées dans des classes).

Après l'analyse précédente, nous arrivons au diagramme de classe complet de notre projet, comme le montre la figure ci-dessous :

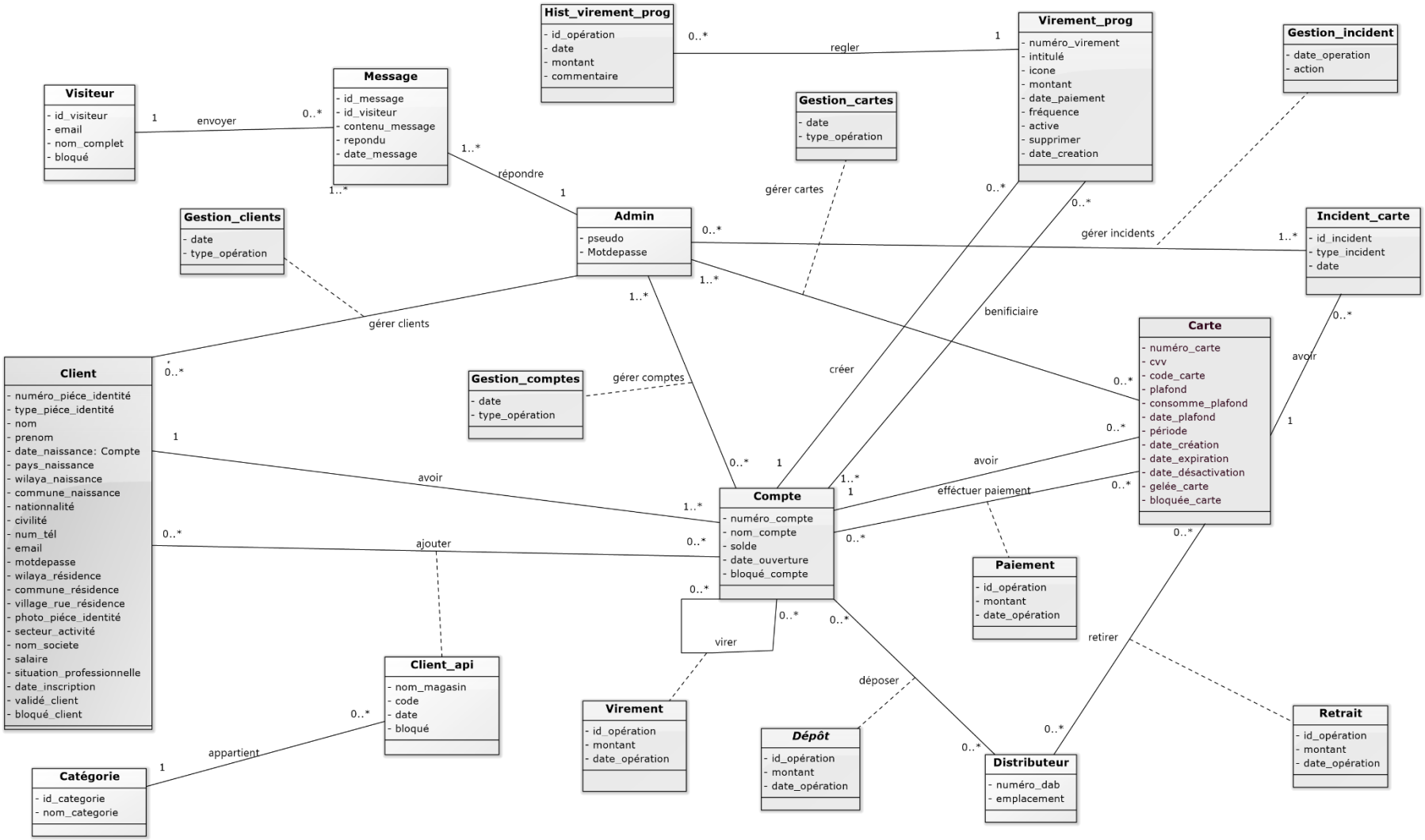


Figure 15: Diagramme de classe général

6.4 CONCEPTION DE LA BASE DE DONNEES RELATIONNELLE

Après avoir modélisé l'application avec les différents diagrammes vus précédemment (cas d'utilisation, de séquence, de classe général et détaillé), nous nous intéressons maintenant aux données nécessaires pour le fonctionnement de l'application. Pour les obtenir, nous suivons de près le déroulement de chaque cas d'utilisation et les diagrammes des classes détaillés qui nous renseignent sur les données nécessaires, qui seront stockées dans les tables de d'une base de données normalisée en 3FN.

Les Tables :

Elle contient les tables suivantes : admin, categorie, carte, client, client_api, communes, compte, depot, distributeur, gerer_carte, gerer_client, gerer_compte, gerer_incident, hist_virement_prog, incident_carte, message_client, message_visiteur, paiement, pays, reponse_client, reponse_visiteur, retrait, virement, virement_prog, visiteur, wilayas.

- **Table admin**

Nom de champ	Type	Description	Clé(s)
Pseudo	Varchar	Pseudo de l'admin	Primaire
Motdepasse	Varchar	Mot de passe de l'admin	

Tableau 5: Table admin

- **Table categorie**

Nom de champ	Type	Description	Clé(s)
Id_categorie	Int	Identifiant catégorie	Primaire
Nom_categorie	Varchar	Nom catégorie	

Tableau 6: Table categorie

- **Table carte**

Nom de champ	Type	Description	Clé(s

Numero_carte	Varchar	Numéro de la carte	Primaire
Numero_compte	Varchar	Numéro de compte	Etrangère
Cvv	Varchar	Cryptogramme de la carte	
Code_carte	Varchar	Code de la carte	
Plafond	Double	Le plafond de la carte	
Consomme_plafond	Double	La somme consomme du plafond	
Date_plafond	Date	La date du plafond	
Periode	Smallint	La période du plafond	
Date_creation	Datetime	Date création de la carte	
Date_expiration	Date	Date expiration de la carte	
Valide_carte	Bool	Si carte est validée	
Gelee_carte	Bool	Si carte est gelée	
Activee_carte	Bool	Si carte est activée	
Bloquee_admin	Bool	Si carte est bloquée	

Tableau 7: Table carte

- **Table client**

Nom de champ	Type	Description	Clé(s)
Numero_piece_identite	Varchar	Numéro de pièce d'identité client	Primaire
Type_piece_identite	Enum	Type de pièce d'identité client	
Nom	Varchar	Nom client	
Prenom	Varchar	Prénom client	
Date_naissance	Varchar	Date de naissance client	
Pays_naissance	Smallint	Pays de naissance client	
Wilaya _naissance	Int	Wilaya de naissance client	
Commune_naissance	Int	Commune de naissance client	
Nationnalite	Smallint	Nationalité client	
Civilite	Enum	Civilité client	
Num_tel	Varchar	Numéro de téléphone client	
Email	Varchar	Email client	
Motdepasse	Varchar	Mot de passe client	
Wilaya_residence	Int	Wilaya de résidence client	

Commue_residence	Int	Commune de résidence client	
Village_rue_residence	Varchar	Village ou rue de résidence client	
Photo_piece_identite	Varchar	Lien de la photo de la pièce d'identité	
Secteur_activite	Varchar	Secteur d'activité client	
Nom_societe	Varchar	Nom de la société client	
Salaire	Double	Salaire client	
Situataion_professionnelle	Varchar	Situation professionnelle client	
Date_inscription	Datetime	Date inscription client	
Valide_client	Bool	Si client est validé	
Blocage_client	Bool	Si client est bloqué	
Bloque_chat	Bool	Si chat client est bloqué	
Recevoir_mail	Bool	Si client reçoit notification	

Tableau 8: Table client

- **Table client_api**

Nom de champ	**Type**	**Description**	**Clé(s)**
Numero_compte	Varchar	Numéro de compte de client	Primaire
Pseudo	Varchar	Pseudo de l'administrateur	Primaire
Categorie	Int	Identifiant catégorie	Etrangère
Nom_magasin	Varchar	Nom du magasin	
Code	Varchar	Code de client	
Date	Datetime	Date d'adhésion	
Bloque	Bool	Si client est bloqué	

Tableau 9: Table client_api

- **Table communes**

Nom de champ	**Type**	**Description**	**Clé(s)**
Id	Int	Identifiant commune	Primaire
Code postal	Varchar	Code postal commune	
Nom	Varchar	Nom commune	
Wilaya_id	Int	Identifiant de la wilaya	Etrangère

Tableau 10: Table communes

- **Table compte**

Nom de champ	Type	Description	Clé(s)
Numero_compte	Varchar	Numéro de compte	Primaire
Numero_piece_identite	Varchar	Numéro de la pièce d'identité	Etrangère
Nom_compte	Varchar	Nom du compte	
Solde	Double	Solde du compte	
Code_api	Varchar	Code d'accès à l'api	
Date_ouverture	Datetime	Date ouverture compte	
Valide_compte	Bool	Si compte est validé	
Bloque_admin	Bool	Si compte est bloqué	

Tableau 11: Table compte

- **Table depot**

Nom de champ	Type	Description	Clé(s)
Id_opération	Int	Identifiant opération	Primaire
Numero_dab	Tinyint	Numéro de distributeur	Etrangère
Numero_compte	Varchar	Numéro de compte	Etrangère
Montant	Double	Montant de l'opération	
Date_operation	Datetime	Date de l'opération	

Tableau 12: Table dépôt

- **Table distributeur**

Nom de champ	Type	Description	Clé(s)
Numero_dab	Tinyint	Numéro de distributeur	Primaire
Emplacement	Varchar	Emplacement de distributeur	

Tableau 13: Table distributeur

- **Table gerer_carte**

Nom de champ	Type	Description	Clé(s)

Pseudo	Varchar	Pseudo admin	Primaire
Numero_carte	Varchar	Numéro de carte	Primaire
Date	Datetime	Date de l'opération	Primaire
Type_operation	Tinyint	Type de l'opération	

Tableau 14: Table gerer_carte

- **Table gerer_client**

Nom de champ	Type	Description	Clé(s)
Pseudo	Varchar	Pseudo admin	Primaire
Numero_piece_identite	Varchar	Numéro de la pièce d'identité	Primaire
Date	Datetime	Date de l'opération	Primaire
Type_operation	Tinyint	Type de l'opération	

Tableau 15: Table gerer_client

- **Table gerer_compte**

Nom de champ	Type	Description	Clé(s)
Pseudo	Varchar	Pseudo admin	Primaire
Numero_compte	Varchar	Numéro de compte	Primaire
Date	Datetime	Date de l'opération	Primaire
Type_operation	Tinyint	Type de l'opération	

Tableau 16: Table gerer_compte

- **Table gerer_ incident**

Nom de champ	Type	Description	Clé(s)
Pseudo	Varchar	Pseudo admin	Primaire
Id_incident	Varchar	Identifiant incident	Primaire
Date	Datetime	Date de l'opération	Primaire
Action	Tinyint	Action prise	

Tableau 17: Table gerer_incident

- **Table hist_virement_prog**

Nom de champ	Type	Description	Clé(s)
Numero_virement	Int	Numéro de virement	Primaire
Date	Datetime	Date de l'opération	Primaire
Montant	Double	Montant opération	
Commentaire	Varchar	Commentaire opération	

Tableau 18: Table hist_virement_prog

- **Table incident_carte**

Nom de champ	Type	Description	Clé(s)
Id_incident	Int	Identifiant incident	Primaire
Numero_carte	Varchar	Numéro de carte	Etrangère
Type_incident	Enum	Type de l'incident	
Date	Datetime	Date de l'incident	

Tableau 19: Table incident_carte

- **Table message_client**

Nom de champ	Type	Description	Clé(s)
Id_message	Int	Identifiant message	Primaire
Numero_piece_identite	Varchar	Numéro pièce d'identité	Etrangère
Contenu_message	Text	Contenu message	
Date_message	Datetime	Date message	
Action	Tinyint	Action prise par l'admin	

Tableau 20: Table message_client

- **Table message_visiteur**

Nom de champ	Type	Description	Clé(s)
Id_message	Int	Identifiant message	Primaire
Id_visiteur	Int	Identifiant de visiteur	Etrangère
Contenu_message	Text	Contenu message	
Date_message	Datetime	Date message	
Action	Tinyint	Action prise par l'admin	

Tableau 21: Table message_visiteur

- **Table paiement**

Nom de champ	Type	Description	Clé(s)
Id_operation	Int	Identifiant opération	Primaire
Numero_carte	Varchar	Numéro de carte	Etrangère
destinataire	Varchar	Numéro de béficiaire	Etrangère
Montant	Double	Montant opération	
Date_operation	Datetime	Date opération	

Tableau 22: Table paiement

- **Table pays**

Nom de champ	Type	Description	Clé(s)
Id	Smallint	Identifiant pays	Primaire
Code	Int	Code pays	
Nom	Varchar	Nom pays	

Tableau 23: Table pays

- **Table reponse_client**

Nom de champ	Type	Description	Clé(s)
Id_reponse	Int	Identifiant réponse	Primaire
Id_message	Int	Identifiant message client	Etrangère
Pseudo	Int	Pseudo de l'admin	Etrangère
Contenu_reponse	Text	Contenu réponse	
Date_reponse	Datetime	Date réponse	
Supprimee	Bool	Si réponse est supprimée	

Tableau 24: Table reponse_client

- **Table reponse_visiteur**

Nom de champ	Type	Description	Clé(s)
Id_reponse	Int	Identifiant réponse	Primaire
Id_message	Int	Identifiant message visiteur	Etrangère
Pseudo	Int	Pseudo de l'admin	Etrangère

Contenu_reponse	Text	Contenu réponse	
Date_reponse	Datetime	Date réponse	

Tableau 25: Table reponse_visiteur

- **Table retrait**

Nom de champ	Type	Description	Clé(s)
Id_operation	Int	Identifiant opération	Primaire
Numero_carte	Varchar	Numéro de carte	Etrangère
Numero_dab	Tinyint	Numéro de distributeur	Etrangère
Montant	Double	Montant opération	
Date_operation	Datetime	Date opération	

Tableau 26: Table retrait

- **Table virement**

Nom de champ	Type	Description	Clé(s)
Id_operation	Int	Identifiant opération	Primaire
Numero_compte	Varchar	Numéro compte	Etrangère
Destinataire	Varchar	Récepteur virement	Etrangère
Montant	Double	Montant opération	
Date_operation	Datetime	Date opération	

Tableau 27: Table virement

- **Table virement_prog**

Nom de champ	Type	Description	Clé(s)
Numero_virement	Int	Numéro de virement	Primaire
Numero_compte	Varchar	Numéro de compte	Etrangère
Intitule	Varchar	Intitulé virement	
Icone	Varchar	Nom icone	
Montant	Double	Montant virement	
Date_paiement	Date	Date paiement virement	
Frequence	Enum	Période paiement virement	

Active	Bool	Si virement est activée	
Supprime	Bool	Si virement est supprimée	
Date_creation	Datetime	Date création virement	

Tableau 28: Table virement_prog

- **Table visiteur**

Nom de champ	Type	Description	Clé(s)
Id_visiteur	Int	Identifiant visiteur	Primaire
Email	Varchar	Email visiteur	
Nom_complet	Varchar	Nom & prénom visiteur	
Bloque	Bool	Si visiteur est bloqué	

Tableau 29: Table visiteur

- **Table wilayas**

Nom de champ	Type	Description	Clé(s)
Id	Int	Identifiant wilaya	Primaire
Code	Int	Code wilaya	
Nom	Varchar	Nom wilaya	

Tableau 30: Table wilayas

7. CONCLUSION

Dans ce troisième chapitre, nous avons proposé une démarche de modélisation pour développer notre application à l'aide des outils UML.

En premier lieu, nous avons défini les besoins, les acteurs et les divers cas d'utilisations en phase d'analyse. Par la suite en phase de conception, nous avons construit les diagrammes de séquences, les diagrammes d'activités et enfin nous avons élaboré le diagramme de classe.

Reste à définir l'environnement de développement de notre application, ce qui sera l'objet du chapitre suivant, qui est la réalisation.

Chapitre 4
Réalisation

1. INTRODUCTION

Dans ce chapitre nous décrierons l'architecture globale de notre application « Banque Virtuelle d'Algérie », ainsi que l'environnement et les outils de développement de notre application ensuite on passera aux architectures et aux normes que nous avons suivi pour la réaliser, et on terminera par quelques captures d'écran de notre application et celles d'autres sites que nous avons simulés pour tester son fonctionnent.

2. ARCHITECTURE DE L'APPLICATION

La figure suivante représente l'architecture globale de notre application :

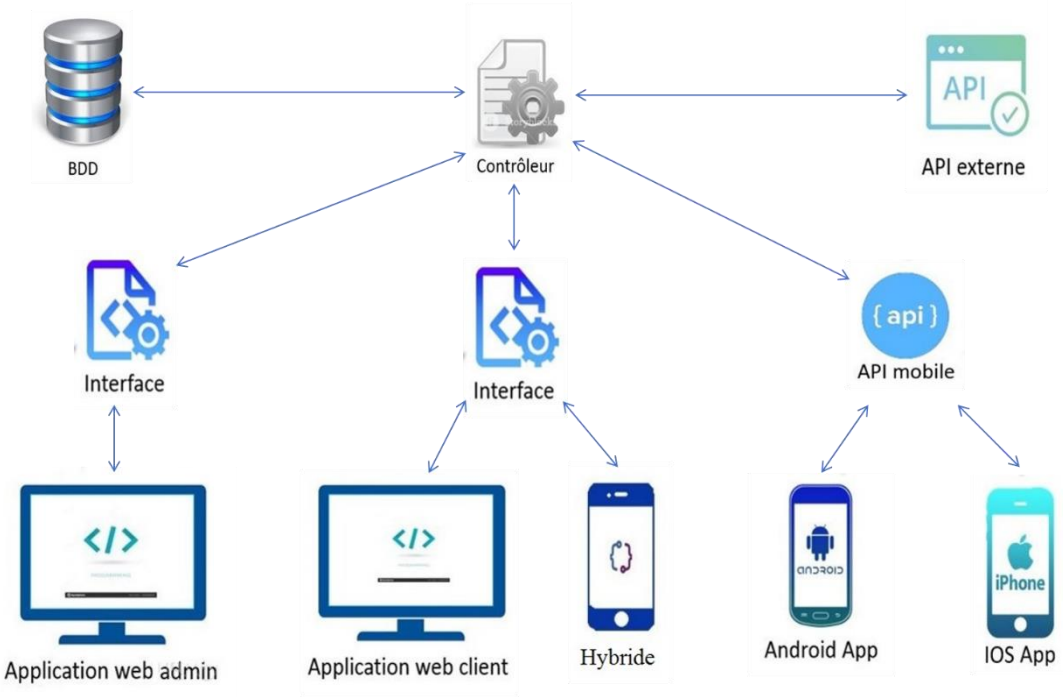


Figure 16: Architecture globale de l'application

- **Contrôleur :** Contient toutes les fonctions utilisées par l'ensemble de l'application, il s'agit d'un intermédiaire entre les différentes interfaces de l'applications et la base de données

- **API Mobile :** C'est l'interface responsable des communications entre le contrôleur et les applications mobiles, elle fournit un ensemble de fonction pour assurer le fonctionnement des deux applications mobiles Android et iOS.
- **Interface :** elle fournit les fonctions et les commandes nécessaires pour répondre aux besoins de l'application, pour l'admin elle contient essentiellement les fonctions de gestions de la banque (activer une carte, débloquer un compte, …etc.), et pour le client, elle implémente les différentes opérations faites par le client.
- **API Externe :** c'est l'interface qui fournit un ensemble de fonctions pour les clients qui souhaitent profiter des services de paiement en ligne. Grâce à cette interface, la banque peut interagir avec d'autres banques et échanger de l'informations.

3. ENVIRONNEMENT DE TRAVAIL

3.1 ENVIRONNEMENT LOGICIEL

3.1.1 Système d'exploitation

- Windows version (7, 8.1 et 10)
- Mac os sierra

3.1.2 Environnement de développement (IDE)

3.1.2.1 IDE

- Brackets
- Xampp
- Android Studio
- Intellij
- Visual Studio (version mac)
- XCode

3.1.2.2 Autres

- FileZilla (client FTP)
- Photoshop

- La plateforme GitHub
- Git
- Software Ideas Modeler

3.1.3 Hébergeur web

GoDaddy : est un hébergeur de sites Web et un registraire de domaine Internet. GoDaddy Group Inc fondée en 1997, est devenue la plus grande société de registres de domaines au monde accréditée par l'ICANN et gère plus de 40 millions de noms de domaine.

En plus des services d'hébergement et d'enregistrement, la société fournit également des solutions et des services de commerce électronique tels que : courrier électronique, développement de sites Web, services de marketing Internet, services de sécurité et de SSL.

3.1.4 Base de données

MySQL : est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, PostgreSQL et Microsoft SQL Server.

3.1.5 Framework

Matérialize : est une bibliothèque de composants d'interface utilisateur créé avec CSS, JavaScript et HTML. Matérialiser réutilisables composants d'interface utilisateur aide à la construction de pages Web attrayantes, cohérentes et fonctionnelles et des applications web tout en adhérant aux principes de conception de sites Web modernes comme la portabilité du navigateur, l'indépendance de l'appareil, et la dégradation gracieuse.

Flutter est un Framework de développement d'applications mobiles open source créé par Google. Il est utilisé pour développer des applications pour Android et iOS et constitue également la méthode principale de création d'applications pour Google Fuchsia

JQuery : C'est un Framework JavaScript sous licence libre qui permet de faciliter des fonctionnalités communes de JavaScript. L'utilisation de cette bibliothèque permet de gagner du temps de développement lors de l'interaction sur le code HTML d'une page web, l'AJAX ou la gestion des évènements. JQuery possède par la même occasion l'avantage d'être utilisable sur plusieurs navigateurs web (cf. Internet Explorer, Firefox, Chrome, Safari ou Opéra).

Highcharts : Highcharts est une bibliothèque de graphiques multiplateformes basée sur SVG, qui a fait ses preuves et qui a été activement développée depuis 2009. Elle facilite l'ajout de graphiques interactifs optimisés pour les mobiles aux projets Web et mobiles. Il propose une documentation robuste, une réactivité avancée et une prise en charge de l'accessibilité inégalée.

3.1.6 Langage de programmation

PHP : (Personal Home Page Hypertext Préprocesseur) est un langage de script orienté objet permettant la programmation de serveurs web. Il permet également la réalisation de sites web dynamiques en gérant les bases de données comme MySQL.

DART : Dart (initialement appelé Dash) est un langage de programmation web développé par Google. Son but initial est de remplacer JavaScript, néanmoins la priorité actuelle des développeurs est que le code Dart puisse être converti en code JavaScript compatible avec tous les navigateurs modernes. Dart peut aussi être utilisé pour la programmation côté serveur, ainsi que le développement d'applications mobiles (via l'API Flutter).

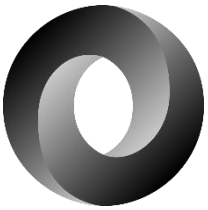
- **SQL :** C'est un langage informatique mis au point par IBM, destiné à interroger ou piloter une base de données. La première version commerciale fut disponible en 1979.
  Exemple de requête SQL:
  « SELECT nom FROM étudiant WHERE âge > 20 ORDER BY âge »,
  Cette requête donne le nom des étudiants dont l'âge est supérieur à 20 ans, les noms des étudiants étant classés par âge croissant.

- **Java script** : C'est un langage de scripts incorporé aux balises Html, permettant d'améliorer la présentation et l'interactivité des pages Web. Il est plus simple à mettre en œuvre car c'est du code que vous ajouterez à votre page écrite en Html, il est donc une extension du code Html des pages Web en permettant d'exécuter des commandes du côté client, c'est-à-dire au niveau du navigateur et non du serveur web.

- **HTML :** « HyperText Markup Language » est un langage (un code) qui permet de structurer une page internet de manière à ce qu'elle puisse être rendue dans les navigateurs Web. Le html est constitué de balises qui entourent le texte et lui donne ainsi un sens (titre, paragraphe, image, lien…). Une balise commence par <xxx> et se termine par </xxx>, xxx étant le nom de la balise.

- **CSS :** « Cascading Style Sheets », il est utilisé pour décrire la présentation d'un document écrit en HTML ou XML. Il décrit les couleurs, la police… La feuille de style fournit la mise en forme des éléments de la page, qui auront été écrits en Xhtml. Elle s'applique à une ou plusieurs pages du site. Le terme « en cascade » indique que la mise en forme d'une page peut faire appel à plusieurs feuilles de style.

- **JSON :** (JavaScript Object Notation) est un format de données textuelles dérivé de la notation des objets du langage JavaScript. Il permet de représenter de l'information structurée comme le permet XML par exemple.



3.2 ENVIRONNEMENT MATERIEL

- PC HP pavilion Intel(R) Core (TM) i7-4500u, 16 GO RAM
- PC Toshiba, 8 GO RAM
- MAC
- Samsung Galaxy s5
- iPhone 6S

3.3 TECHNOLOGIE ET ARCHITECTURE

3.3.1.1 Service REST

REST signifie « REPresentational State Transfer », REST est une architecture basée sur les normes Web et utilise le protocole HTTP. Il s'articule autour d'une ressource où chaque composant est une ressource et où une interface commune accède à une ressource à l'aide de méthodes standard HTTP.

Dans l'architecture REST, un serveur REST fournit simplement un accès aux ressources et le client REST accède et modifie les ressources. Ici, chaque ressource est identifiée par des URI / ID globaux. REST utilise diverses représentations pour représenter une ressource telle que text, JSON, XML. JSON est le plus populaire

3.3.1.2 API

Signifie « Interface de programmation d'application », une API est un ensemble de commandes, fonctions, protocoles et objets que les programmeurs peuvent utiliser pour créer des logiciels ou interagir avec un système externe. Il fournit aux développeurs des commandes standard pour effectuer des opérations courantes afin qu'ils n'aient pas à écrire le code à partir de zéro.

Exemple : Google Maps, YouTube API, Messenger API …etc.

3.3.1.3 MVC

Le MVC (« modèle vue contrôleur ») est une architecture de développement visant à séparer le code source en modules.

En effet, ce modèle très répandu, consiste à séparer distinctement l'accès aux données (bases de données), la vue affichée à l'utilisateur et la logique métier.

Cette architecture est le plus communément retrouvée au sein d'applications web mais existe également au niveau des applications lourdes.

4. PRESENTATION DE L'APPLICATION

4.1 EXEMPLES DE DOCUMENT IMPRIMABLE

- Relevé identité bancaire [RIB client]. (Page 92)
- Relevé des transactions. (Page 93)
- Liste des clients. (Page 94)

4.2 EXEMPLE DE GRAPHES IMPRIMABLE

- Dépenses et revenus d'un client. (Page 95)

Banque Virtuelle d'Algérie

Relevé d'Identité Bancaire (RIB)

Titulaire : NABI Kamel

Code banque	Code guichet	N° Compte	Clé
115	15000	0000000003	78

Le 23/06/2019

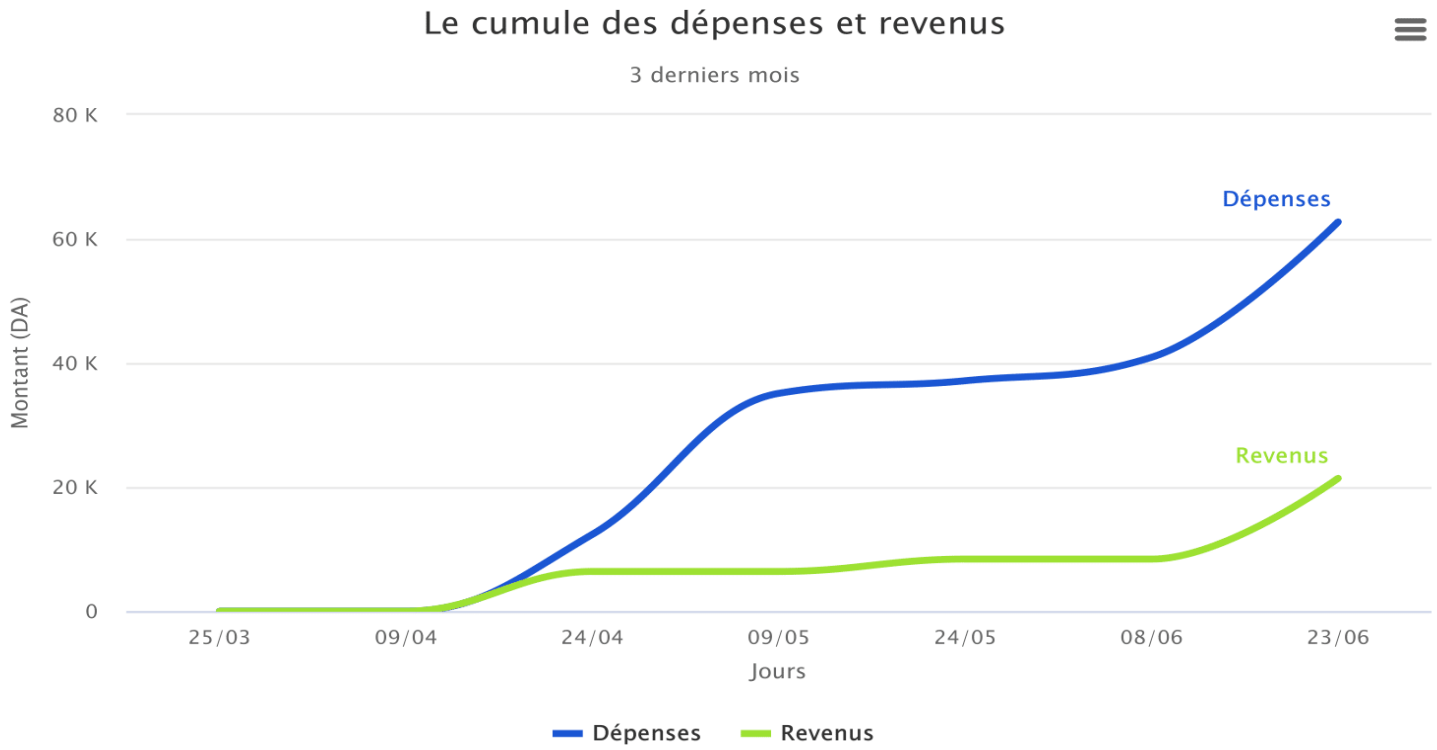
Transactions compte N° 9720190366

De	À	Type opération	Montant (DA)	Date opération
L'intéressé 6387 5203 8933 5630	Chlef DAB N°13	Retrait	-2 000.00	31/05/2019 14:49
L'intéressé 6387 5203 8933 5630	Chlef DAB N°13	Retrait	-11 000.00	31/05/2019 14:35
L'intéressé 6387 5203 8933 5630	Annaba DAB N°12	Retrait	-3 000.00	31/05/2019 14:34
L'intéressé 6387 5203 8933 5630	Bab el oued DAB N°11	Retrait	-2 000.00	31/05/2019 04:02
Tizi ouzou DAB N°15	L'intéressé	Dépôt	+3 000.00	31/05/2019 04:01
L'intéressé	NABI Kamel 0000000003	Virement	-1 000.00	31/05/2019 03:58
COMPTE COURANT 9720190366	COMPTE FAMILIAL 6369423502	Transfert	1 000.00	31/05/2019 03:57
L'intéressé 6387 5203 8933 5630	TECHNOSHOP 0000000003	Paiement	-39 900.00	31/05/2019 03:56

Liste des clients

N° piéce d'identité	Titulaire	Date naissance	Solde totale (DA)	Date inscription	Statut
164926335	OULMANE Feriel	12/04/1997	5 000.00	07/05/2019	Actif
545555452	BELAID Ait zeggane	03/05/2000	49 000.00	07/05/2019	Actif
132345445	SIGUER Toufik	10/05/1995	3 244 092.00	30/04/2019	Actif
123654789	NABI Kamel	05/10/1995	12 496 895.50	10/04/2019	Actif
165456465	OULMANE Hakim	10/05/1995	120 871.00	08/04/2019	Actif

EAV : En attente de validation


Le cumule des dépenses et revenus
3 derniers mois
80 K
60 K
40 K
20 K
0
Montant (DA)
Dépenses
Revenus
25/03
09/04
24/04
09/05
24/05
08/06
23/06
Jours
Dépenses
Revenus

4.3 CAPTURES D'ECRAN (SERONT DISPONIBLES PROCHAINEMENT)

4.3.1 Page d'accueil du site de la banque

Figure 17: Capture des écrans page accueil

4.3.2 Application web coté client

4.3.2.1 Espace comptes

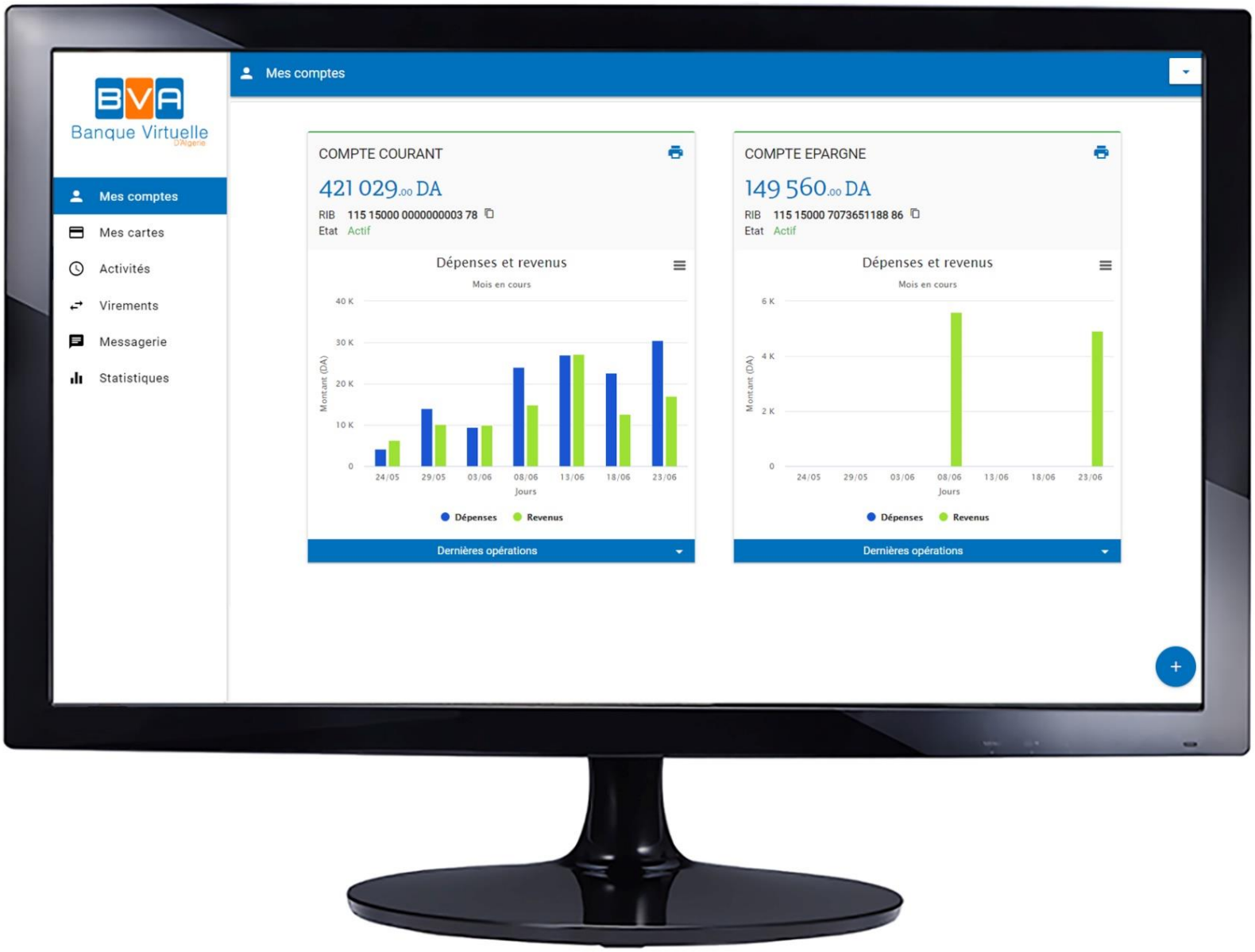


Figure 18: capture d'écran page espaces comptes coté client web

4.3.2.2 Espace cartes

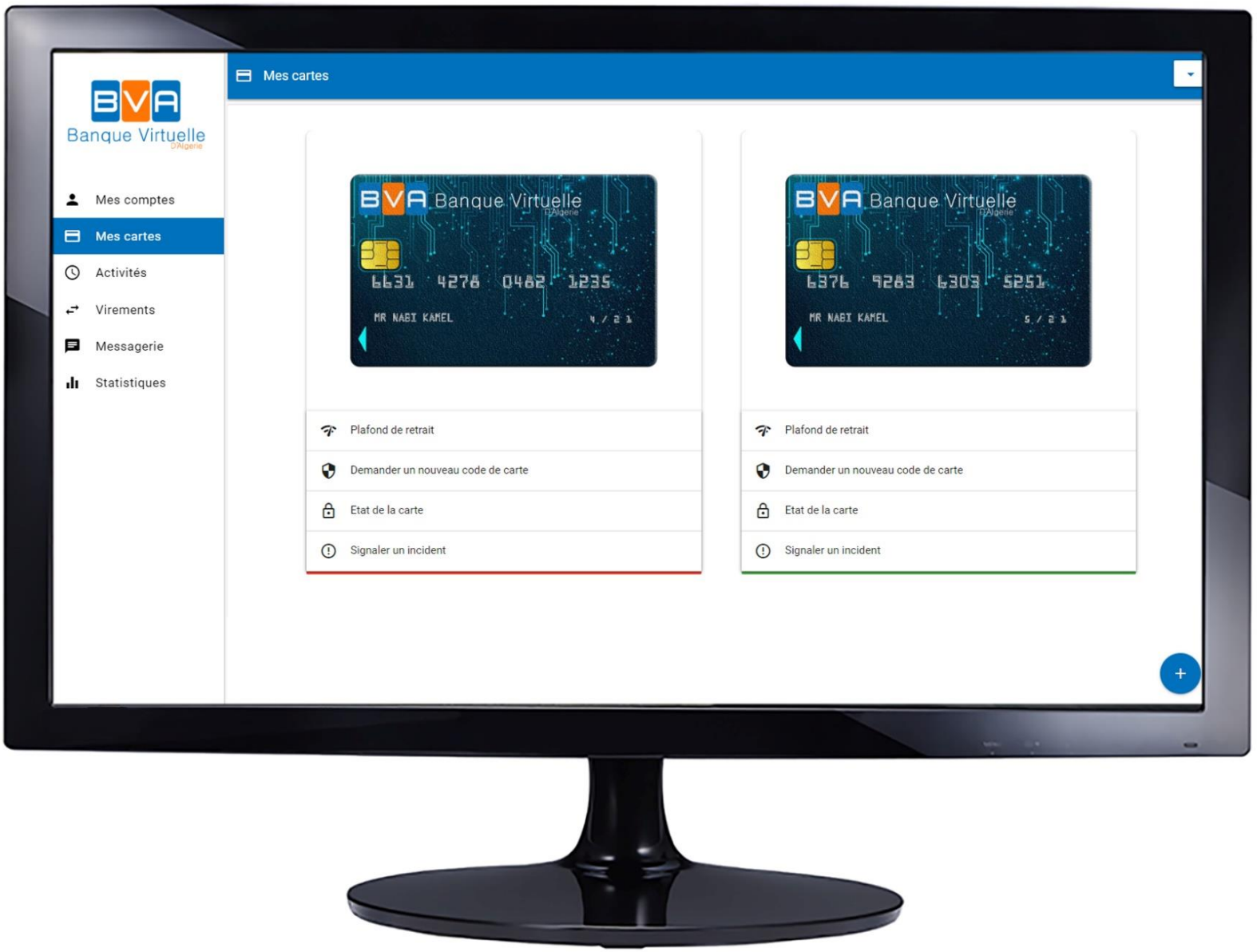


Figure 19 : capture d'écran page espace carte coté client web

4.3.3 Application web coté administrateur

4.3.3.1 Consultation liste des clients

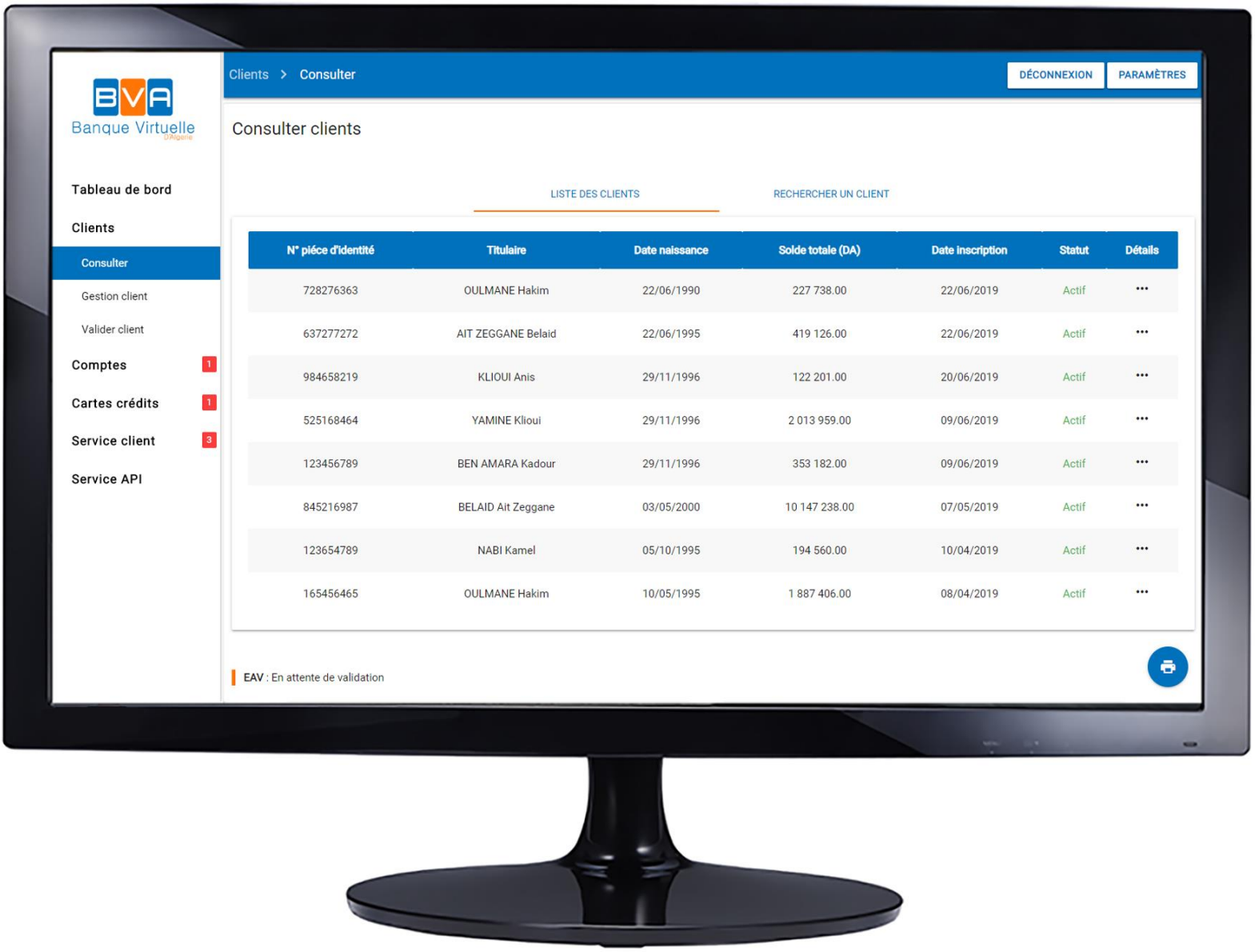


Figure 20 : capture d'écran page consulter client coté admin

4.3.3.2 *Tableau de bord (statistiques globales)*

Figure 21: capture d'écran page statistiques coté admin

4.3.4 Applications mobiles

4.3.4.1 Espace comptes

Figure 22: capture d'écran espace comptes version Android native

Figure 23:capture d'écran espace comptes version hybride

4.3.4.2 Espace cartes

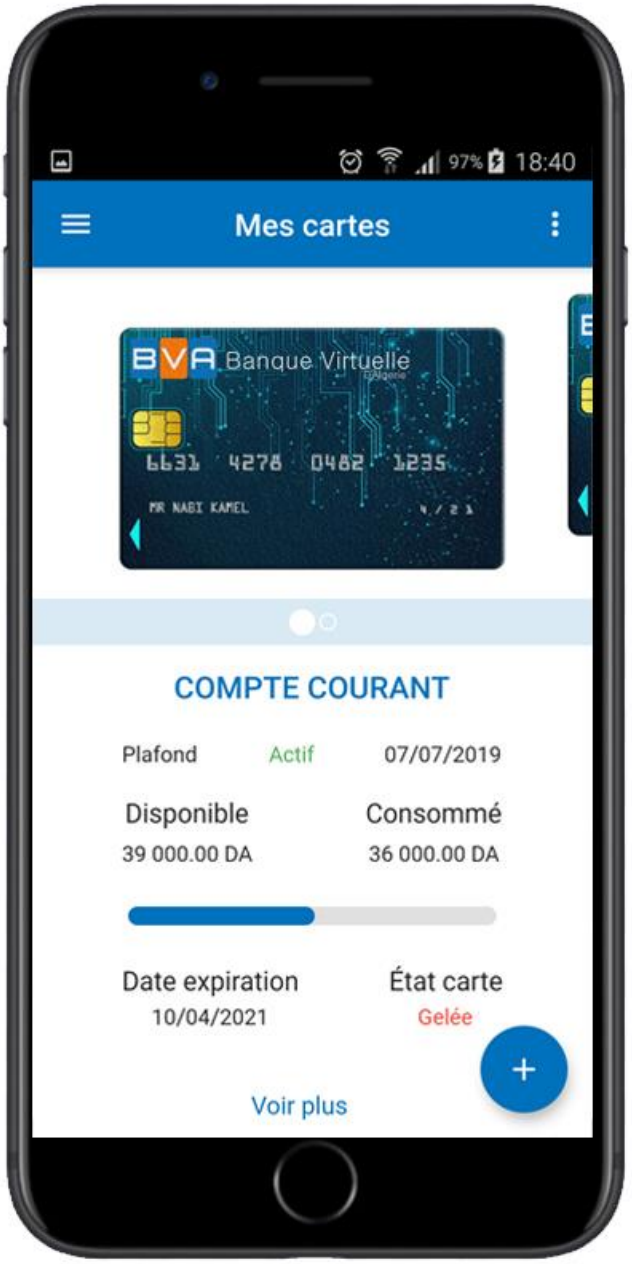


Figure 25: capture d'écran espace cartes version iOS native

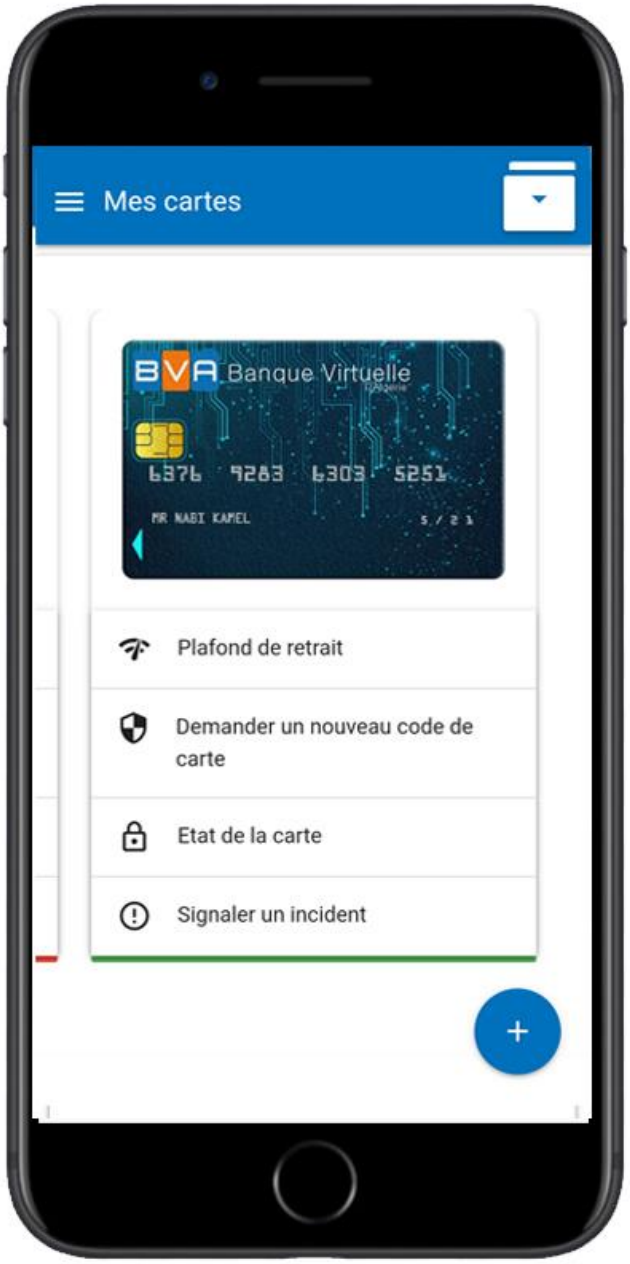


Figure 24: capture d'écran carte version hybride

4.3.4.3 Statistiques

Figure 27: capture d'écran statistiques version Android native

Figure 26: capture d'écran statistiques version hybride

4.3.4.4 Virements

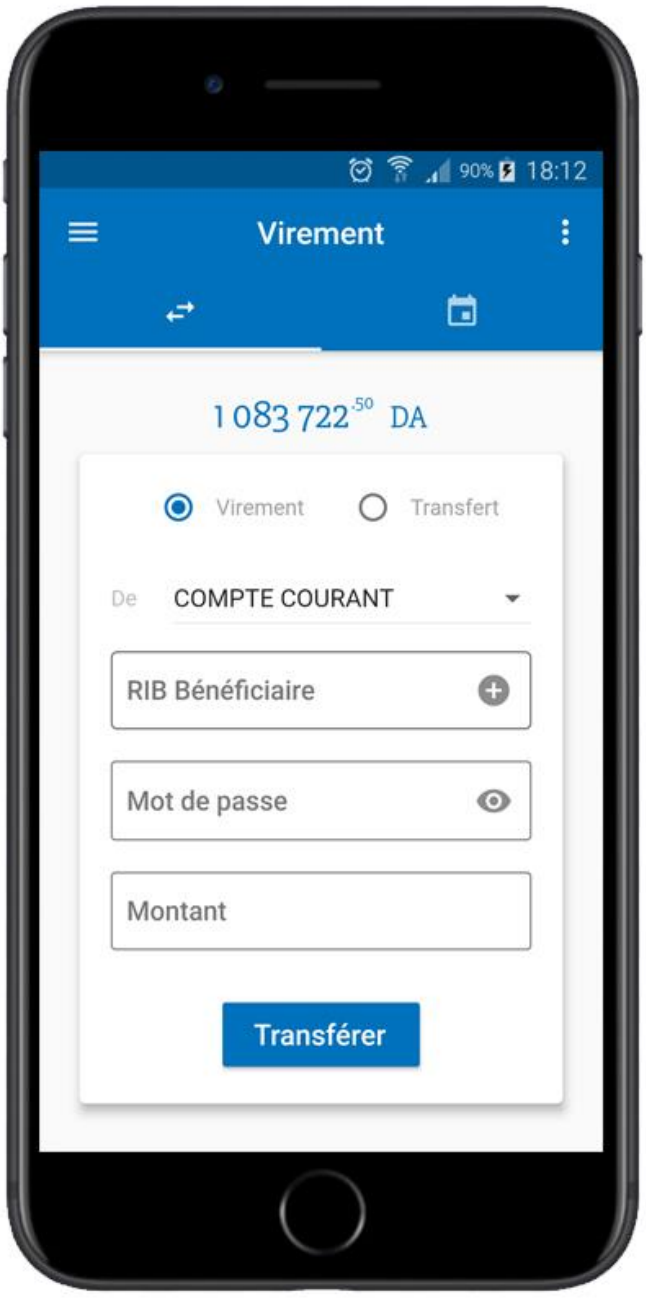


Figure 29: capture d'écran virement version iOS native

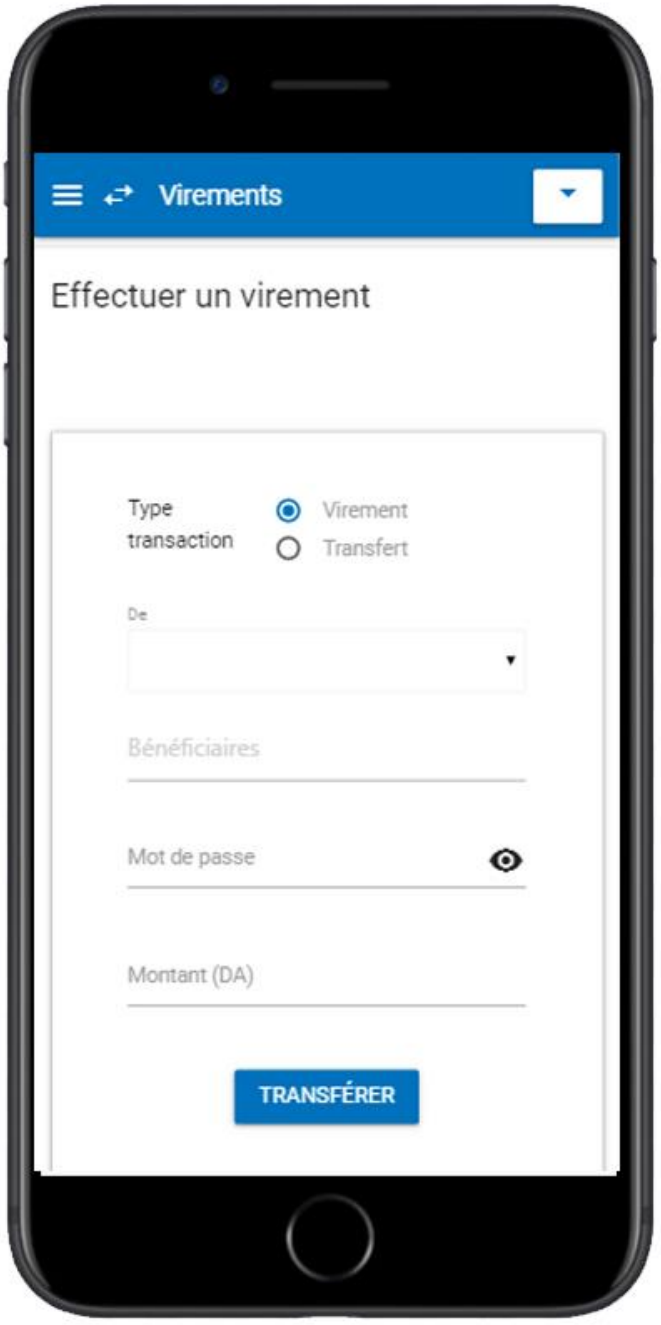


Figure 28: capture d'écran virement hybride

4.3.4.5 Autres

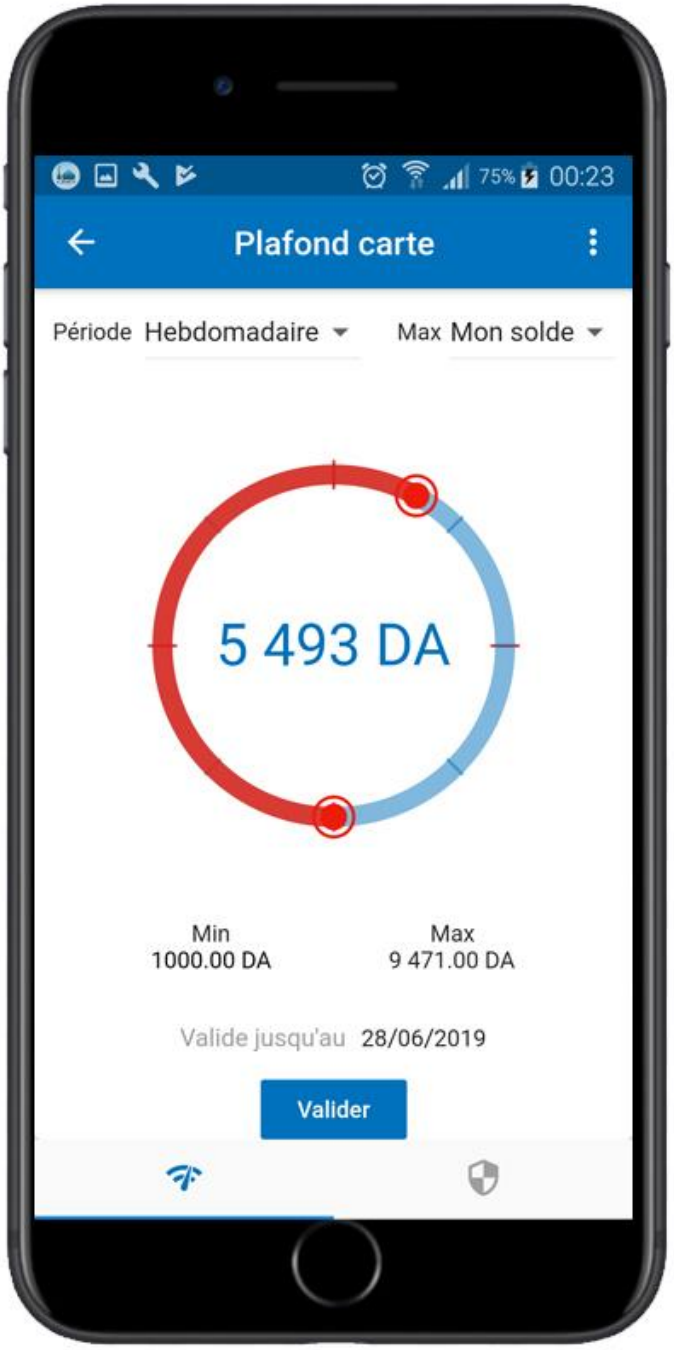


Figure 31: capture d'écran gestion plafond carte version iOS native

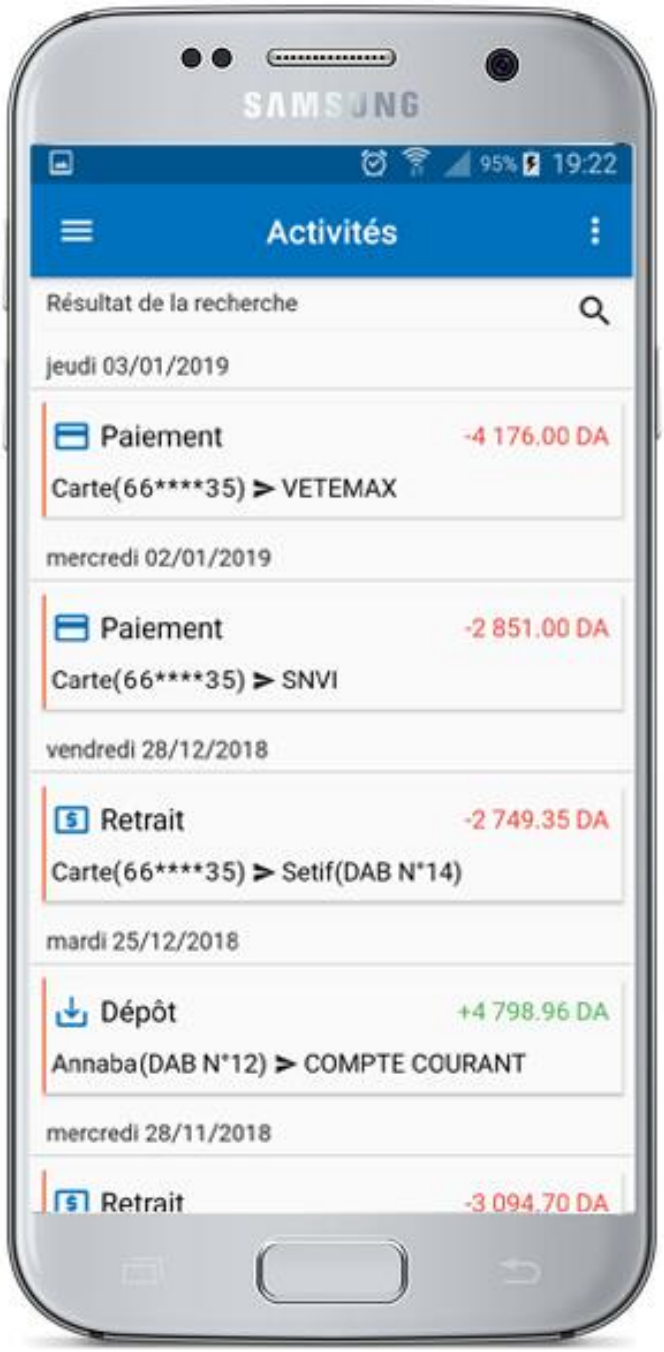


Figure 30: capture d'écran espace activités version Android native

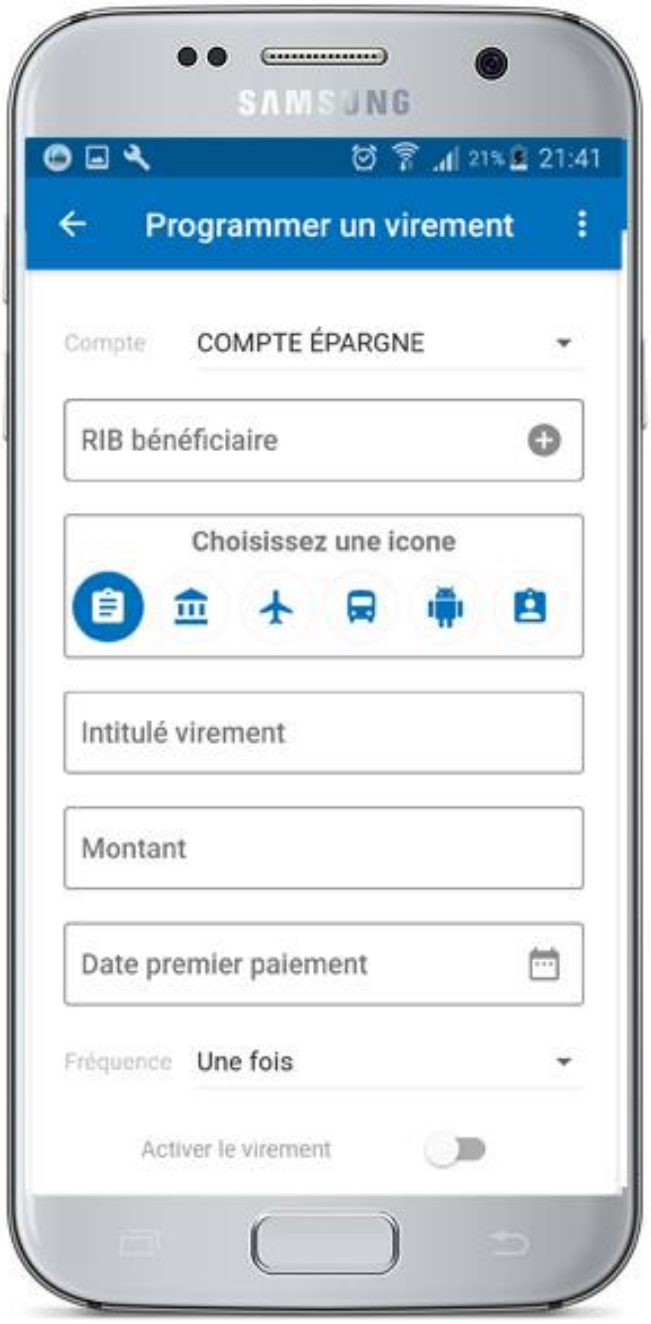


Figure 33: capture d'écran virement programmé version Android native

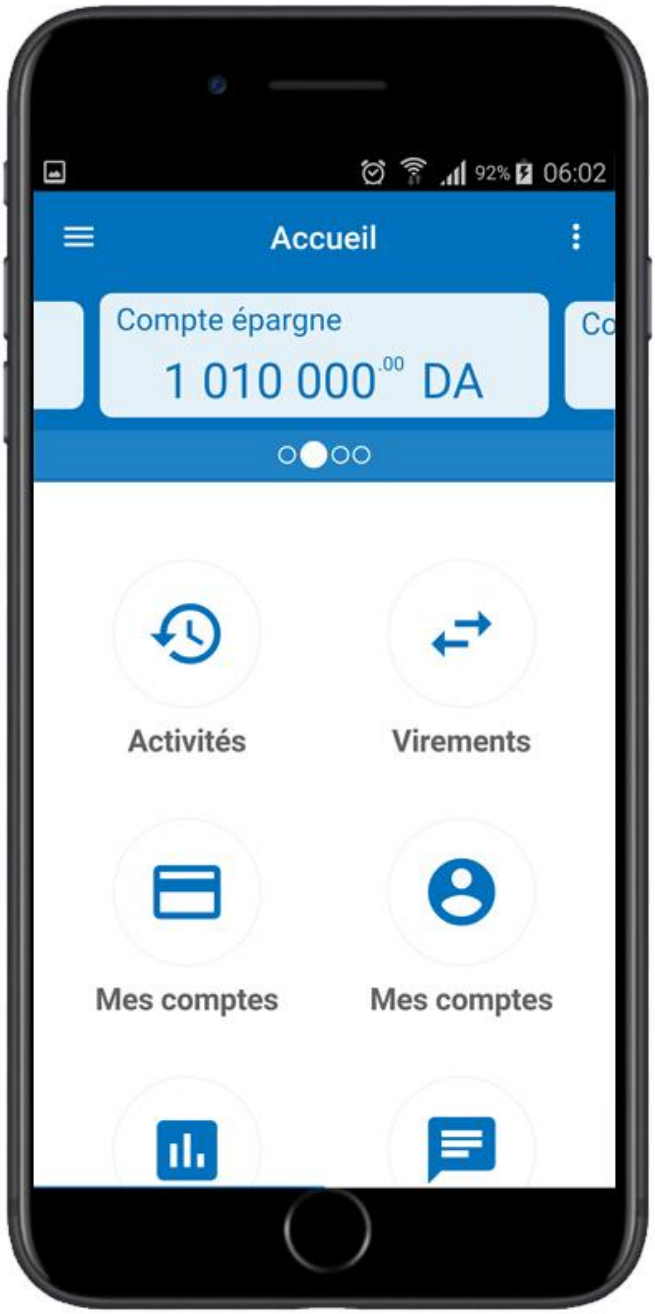


Figure 32: capture d'écran page accueil version iOS native

Afin de tester le fonctionnement de notre application, nous avons simulé un site d'achat et un guichet automatique pour interagir avec notre système.

4.3.5 Site d'achat

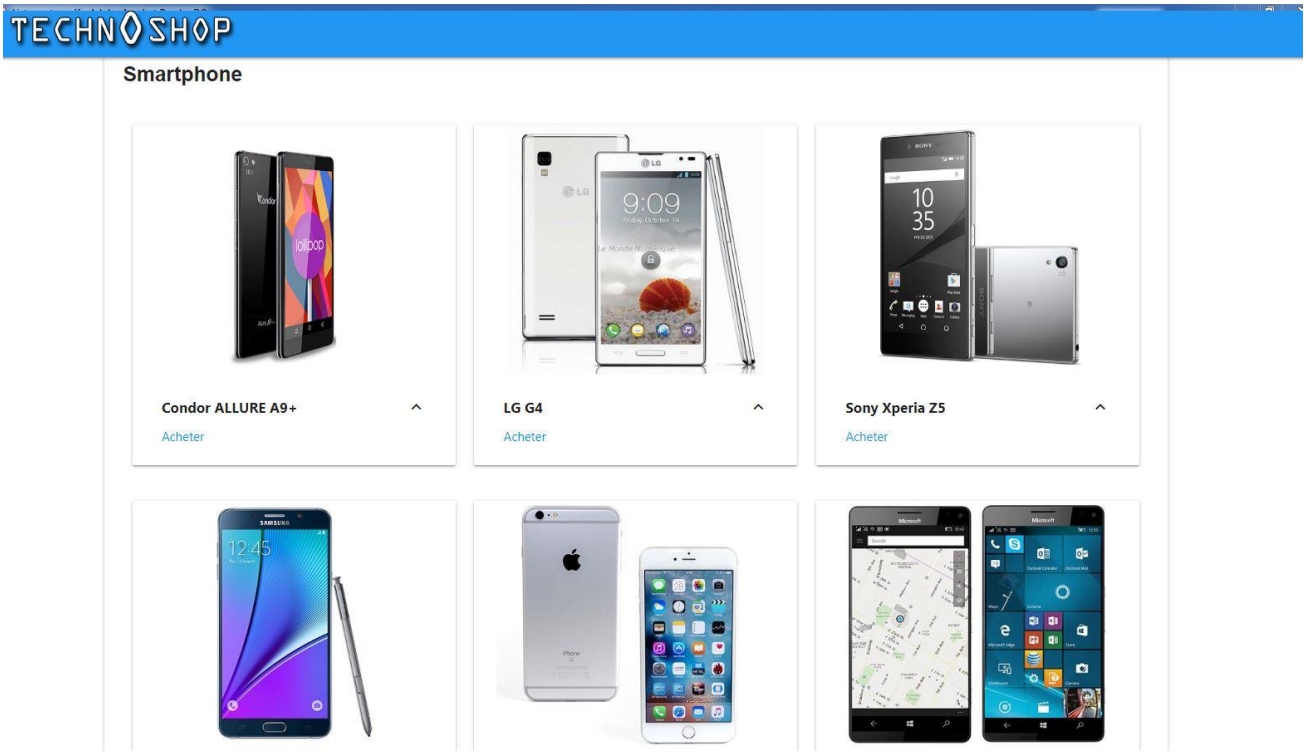


Figure 34: L'accueil de site d'achat

Figure 35: Le formulaire de paiement avec la carte de crédit

Guichet automatique

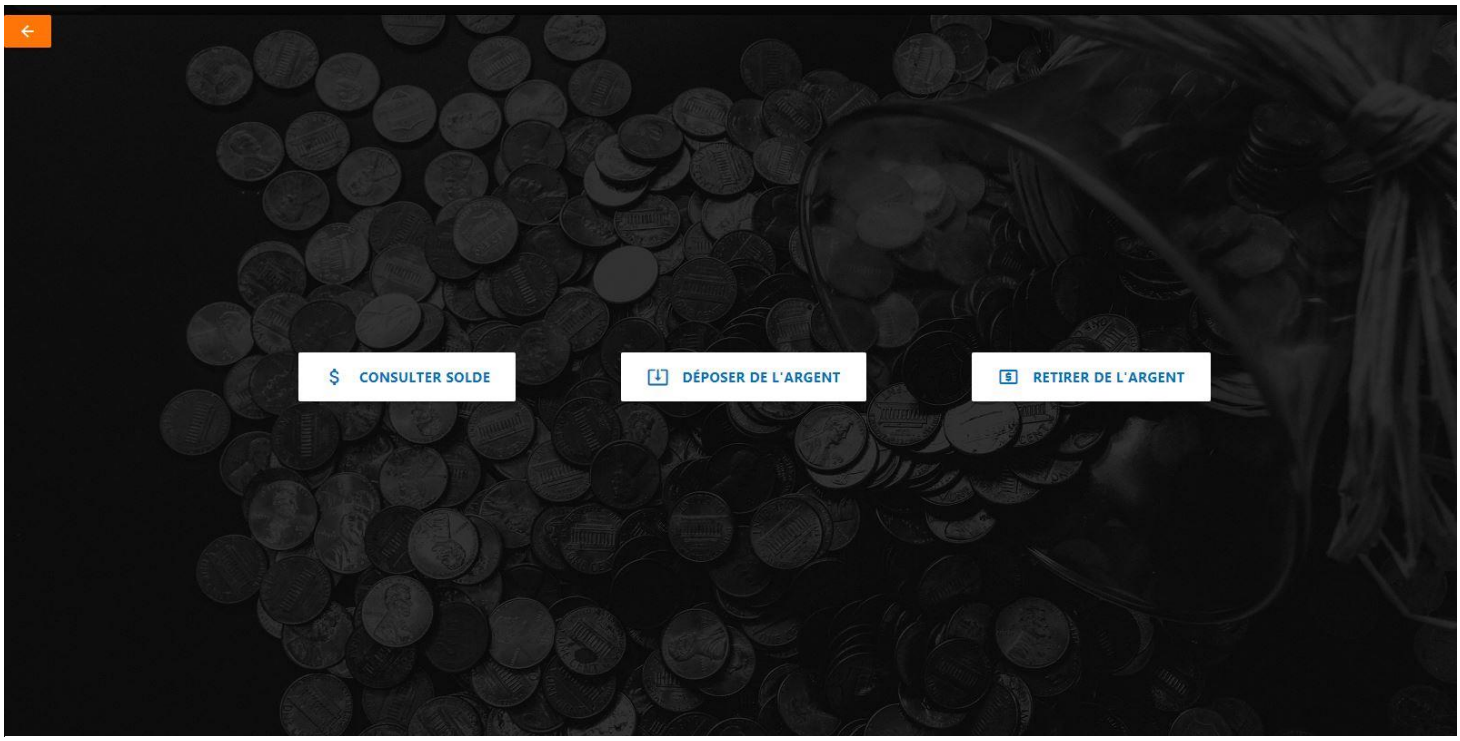


Figure 37: Espace client de guichet

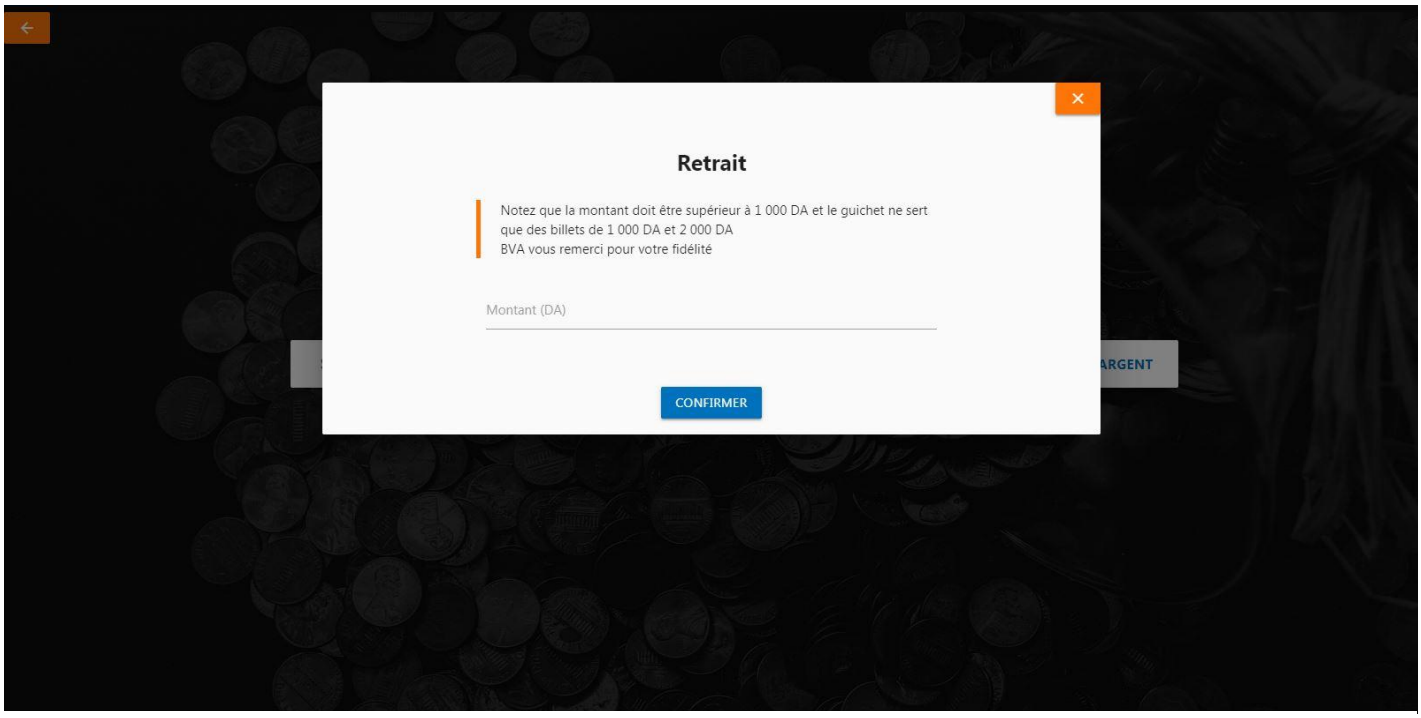


Figure 36: Formulaire de retrait

5. CONCLUSION

Dans ce dernier chapitre nous avons présenté une vue générale sur l'application que nous avons eu pour but de réaliser.

Nous avons montré les différents rendus visuels que nous offrait l'application ainsi que ses composants et le rôle de chacun.

Entant donnés les résultats obtenus, nous pouvons en déduire que tous les objectifs de départ à savoir la réalisation :

- Application web banque en ligne coté client.
- Application web gestion banque coté admin.
- Application mobile multiplateforme (Android et iOS).
- Application web progressive (Android et iOS).

ont été réalisé et implémenté.

Conclusion générale

En vue d'atteindre notre objectif qui était la conception et la réalisation d'une application banque en ligne « Banque Virtuelle d'Algérie », nous avons procédé par étapes :

- La première consistait à nous renseigner sur le fonctionnement des banques en ligne ainsi que leurs multiples services proposés.
- La deuxième étape de la réalisation notre projet était de faire une étude théorique sur les notions des bases des systèmes d'exploitations Android et iOS ainsi les technologies employées.
- Nous avons entamé par la suite l'analyse et la spécification des besoins de l'application et leurs modélisations par le diagramme de contexte et les diagrammes de cas d'utilisation
- Partant de la spécification, nous avons commencé la conception de l'application à travers les différents digrammes UML.
- Ensuite, nous sommes passés à l'implémentation des différents modules de l'application en utilisant Flutter pour le mobile et PHP coté web ainsi que les différents outils cités.
- Enfin, nous avons décrit depuis le chapitre réalisation l'environnement de développement et les outils utilisés lors de la réalisation de l'applications tout en présentant quelques interfaces.

Ce projet a fait l'objet d'une expérience intéressante, elle nous a permis l'acquisition de connaissances multiples et d'améliorer nos compétences dans le domaine de la conception et la programmation, elle a aussi permis d'avoir une première expérience professionnelle dans le développement mobile. Tout au long de l'élaboration du projet, nous avons rencontré plusieurs difficultés tant au niveau conceptuel qu'au niveau de la réalisation. Tout de même, nous avons réussi à les surpasser pour présenter en fin de compte une application totalement opérationnelle en grandeur réelle.

Comme perspective nous envisageons d'intégrer des mécanismes de sécurité et améliorer son interopérabilité avec les autres systèmes bancaires.

Nous pensons avoir atteint les objectifs fixés au départ, et souhaitons que ce travail pourra faire l'objet d'une exploitation ultérieure.

BIBLIOGRAPHIE

Cédric DEMOULIN, M. V. (s.d.). *PRINCIPES DE BASE DU FONCTIONNEMENT DU RÉSEAU GSM.* Institut Montefiore.

HACHEMI, M. (2008 -2009). *Thèse d'ingénieur : Conception et réalisation d'une application E-Commerce.* UMMTO.

IMMOUN LYNDA, K. L. (2014-2015). *Conceptoion et réalisation dun application mobile banking sous Android cas BNA Tizi Ouzou.* UMMTO.

Joseph Gabay, D. G. (s.d.). *UML 2 analyse et conception « Mise en oeuvre guidée avec études de cas Tome 1 »*. DUNOD.

Technologies et architecture Internet 2e Edition. (2003). DUNOD.

WEBOGRAPHIE

1G vs 2G vs 3G vs 4G vs 5G. (s.d.). Récupéré sur slideshare: https://www.slideshare.net/gokilabharathi/1g-vs-2g-vs-3

Android : news, Définition, photos, vidéos, fonds d'écran, membres. (s.d.). Récupéré sur maxisciences: https://www.maxisciences.com/android/

Android Architecture. (s.d.). Récupéré sur http://www-igm.univ-mlv.fr/~dr/XPOSE2008/android/archi_comp.html

API (Application Program Interface) Definition. (s.d.). Récupéré sur techterms: https://techterms.com/definition/api

Architecture reseau mobile. (s.d.). Récupéré sur slideshare: https://fr.slideshare.net/paterneson/architecture-reseau-mobile

Banque en ligne — Wikipédia. (s.d.). Récupéré sur Wikipédia: https://fr.wikipedia.org/wiki/Banque_en_ligne

Banque en Ligne : comparatif meilleures banques en ligne. (s.d.). Récupéré sur comparatifdebanque: http://www.comparatifdebanque.fr/

Banque en ligne · ING. (s.d.). Récupéré sur ing: https://www..fr/banque-en-ligne/

Banque en ligne, banque virtuelle sur internet - Ooreka. (s.d.). Récupéré sur ooreka: https://banque.ooreka.fr/comprendre/banque-en-ligne

Banques Virtuelles - Un service de paiement en ligne. (s.d.). Récupéré sur i-argent: http://www.i-argent.com/banques-virtuelles.htm

De la première aux normes de télécommunications de cinquième génération. (s.d.). Récupéré sur sicomtesting: https://www.sicomtesting.com/fr/blog/dal-1g-al-5g-il-passato-e-il-futuro-degli-standard-gsm-umts-hspa-ed-lte/

définition application mobile. (s.d.). Récupéré sur Université de lille: https://ori-nuxeo.univ-lille1.fr/nuxeo/site/esupversions/e260e333-f992-4d28-bc33-ff5895949c64

Définition d'application mobile et les types d'appli à choisir. (s.d.). Récupéré sur taktilcommunication: https://www.taktilcommunication.com/blog/applications-mobile/definition-typologie-applications-mobiles.html

Développement d'application web mobile. (s.d.). Récupéré sur appstud: https://www.appstud.com/fr/guides/agence-mobile/app059/

Développement d'applications mobiles: pourquoi iOS est plus cher? (s.d.). Récupéré sur yeeply: https://fr.yeeply.com/blog/developpement-dapplications-mobiles-ios-plus-cher-android/

Difference between 3G and 4G Technology (with Comparison Chart, Architecture, Advantages and Limitations) - Tech Differences. (s.d.). Récupéré sur techdifferences: https://techdifferences.com/difference-between-3g-and-4g-technology.html

EMF - 4G (LTE) & Health - L2. (s.d.). Récupéré sur emfexplained: http://www.emfexplained.info/?Page=25526

Generation of mobile communication systems. (s.d.). Récupéré sur slideshare: https://www.slideshare.net/jincy-a/generation-of-mobile-communication-systems

Generations of network 1 g, 2g, 3g, 4g, 5g. (s.d.). Récupéré sur slideshare: https://www.slideshare.net/noorec786/generations-of-network-1-g-2g-3g-4g-5g

Interface de programmation — Wikipédia. (s.d.). Récupéré sur Wikipédia: https://fr.wikipedia.org/wiki/Interface_de_programmation

Internet mobile : 10 raisons de proposer une application plutôt qu'une version web mobile de son site. (s.d.). Récupéré sur presse-citron: https://www.presse-citron.net/internet-mobile-10-raisons-de-proposer-une-application-plutot-quune-version-web-mobile-de-son-site/

Introduction to App Development. (s.d.). Récupéré sur codeconquest: https://www.codeconquest.com/what-is-coding/mobile-programming/

Introduction To Mobile Applications. (s.d.). Récupéré sur codeburst: https://codeburst.io/introduction-to-mobile-applications-ddf9732b690a

Introduction to Mobile Technologies: Media & Communications Book Chapter. (s.d.). Récupéré sur IGI Global: https://www.igi-global.com/chapter/introduction-mobile-technologies/60208

Introduction to Mobile Technology. (s.d.). Récupéré sur codeconquest: https://www.codeconquest.com/what-is-coding/mobile-programming/

L'application native vs. l'application Web, quelle est la différence? | Symetris Développement Web. (s.d.). Récupéré sur symetris: https://www.symetris.ca/fr/blogue/lapplication-native-vs-lapplication-web-quelle-est-la-difference

L'état du multi-plateforme sur le mobile – Backelite. (s.d.). Récupéré sur backelite: https://www.backelite.com/fr/2018/07/letat-du-multi-plateforme-sur-le-mobile/

La Téléphonie Mobile & Architechture réseau 4G | SUPINFO, École Supérieure d'Informatique. (s.d.). Récupéré sur supinfo: https://www.supinfo.com/articles/single/2455-telephonie-mobile-architechture-reseau-4g

La Téléphonie Mobile & Architechture réseau 4G | SUPINFO, École Supérieure d'Informatique. (s.d.). Récupéré sur supinfo: https://www.supinfo.com/articles/single/2455-telephonie-mobile-architechture-reseau-4g

Le développement d'application mobile: Android vs iOS. (s.d.). Récupéré sur yeeply: https://fr.yeeply.com/blog/developpement-dapplication-mobile-android-ios/

Les différentes technologies GSM, GPRS, EDGE, UMTS, HSDPA, HSUPA, HSPA+, LTE, 4G. (s.d.). Récupéré sur distrimedia: http://www.distrimedia.fr/technologies-gsm-cellulaire.html

Les fréquences 2G, 3G et 4G expliquées. (s.d.). Récupéré sur YouTube: https://www.youtube.com/watch?v=BOvXV5rreDE

L'essentiel du développement d'applications iPhone. (s.d.). Récupéré sur techotopia: https://www.techotopia.com/index.php/IPhone_App_Development_Essentials

L'internet mobile. (s.d.). Récupéré sur cours-informatique-gratuit: https://cours-informatique-gratuit.fr/cours/internet-mobile/

LTE - Introduction. (s.d.). Récupéré sur univ: http://igm.univ-mlv.fr/~dr/XPOSE2010/LTE/sae.html

LTE Network Infrastructure and Elements - lteencyclopedia. (s.d.). Récupéré sur google: https://sites.google.com/site/lteencyclopedia/lte-network-infrastructure-and-elements

Memoire Online - étude et mise en place d'un réseau GSM dans une entité académique - Ladislas Elias Semajeri. (s.d.). Récupéré sur memoireonline: https://www.memoireonline.com/04/17/9824/m_tude-et-mise-en-place-d-un-reseau-GSM-dans-une-entite-academique6.html#toc16

Mobile Application Development Introduction - DeveloperBrothers. (s.d.). Récupéré sur developerbrothers: https://www.developerbrothers.com/introduction-mobile-application-development/

Native, Web or Hybrid Apps? What's The Difference? (s.d.). Récupéré sur mobiloud: https://www.mobiloud.com/blog/native-web-or-hybrid-apps/

NGMN - Home. (s.d.). Récupéré sur ngmn: https://www.ngmn.org/home.html

Nos réalisations | Beapp. (s.d.). Récupéré sur beapp: https://corp.beapp.fr/nos-realisations

Offre banque en ligne - Ooreka. (s.d.). Récupéré sur Ooreka: https://banque.ooreka.fr/comprendre/offre-banque-en-ligne

Presentation on 1G/2G/3G/4G/5G/Cellular & Wireless Technologies. (s.d.). Récupéré sur slideshare: https://www.slideshare.net/kaushal_kaith/3g-4g-5g

Que choisir entre une application Native, Hybride ou Web ? (s.d.). Récupéré sur yeeply: https://fr.yeeply.com/blog/application-native-hybride-ou-web/

Quelle est la différence entre une web app et une application native ou hybride ? | BeApp. (s.d.). Récupéré sur beapp: https://corp.beapp.fr/expertises/questions-frequentes/difference-entre-web-app-application-native-hybride

RESTful Web Services Introduction. (s.d.). Récupéré sur tutorialspoint: https://www.tutorialspoint.com/restful/restful_introduction.htm

SOAP Web Services - javatpoint. (s.d.). Récupéré sur javatpoint: https://www.javatpoint.com/soap-web-services

Technologie mobile : pourquoi et comment - Nouvelles technologiques pour les musées - Canada.ca. (s.d.). Récupéré sur https://app.pch.gc.ca/sgc-cms/nouvelles-news/francais-french/?p=4924

Téléphonie mobile - 3G et 4G expliquées. (s.d.). Récupéré sur commentcamarche: https://www.commentcamarche.net/contents/1123-telephonie-mobile-3g-et-4g-expliquees

Terminal de paiement 3G / GPRS / GSM | Vtpe. (s.d.). Récupéré sur vente-terminal-de-paiement: https://www.vente-terminal-de-paiement.com/terminal-de-paiement-mobile-tpe-gprs-3g/

The best introduction to Mobile Application Development. (s.d.). Récupéré sur sendiancreations: https://sendiancreations.com/mobile-app-development/

The history of mobile technology — and its future. (s.d.). Récupéré sur mobilebusinessinsights: https://mobilebusinessinsights.com/2018/03/the-history-of-mobile-technology-and-its-future/

Turn Your WordPress Site into a Progressive Web App (PWA). (s.d.). Récupéré sur mobiloud: https://www.mobiloud.com/blog/wordpress-to-pwa/

What is API - Application Program Interface? Webopedia Definition. (s.d.). Récupéré sur webopedia: https://www.webopedia.com/TERM/A/API.html

What's a Mobile App? (s.d.). Récupéré sur lifewire: https://www.lifewire.com/what-is-a-mobile-application-2373354

Where Do Cross-Platform App Frameworks Stand in 2019? (s.d.). Récupéré sur netsolutions: https://www.netsolutions.com/insights/cross-platform-app-frameworks-in-2019/