|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| République Tunisienne |  |  |  | **Ingénieur en:**  Génie Informatique Industrielle  **Option :**  Réseaux et Contrôle Industriels  ***Projet de Fin d’Etudes***  **N° d’ordre:** ING-GII-26- 15 |
| Ministère de l’Enseignement Supérieur  et de la Recherche Scientifique |
|  |
| Université de Sfax |
|  |
| Ecole Nationale d’Electronique et des Télécommunications de Sfax |

**MEMOIRE**

***présenté à***

L’Ecole Nationale d’Electronique et des Télécommunications de Sfax

***en vue de l’obtention du***

Diplôme National d’Ingénieur en :

Génie Informatique Industrielle

Option :

Réseaux et Contrôle Industriels

***par***

**Bilel MAKNI**

**&**

**Intissar BEN BOUZID**

**Développement d’un Mobile Banking sur Android**

***soutenu le 18 juin 2015, devant la commission d'examen:***

|  |  |  |
| --- | --- | --- |
| Mme. | Kaouthar FAKHFAKH | Président |
| M. | Achraf MTIBAA | Examinateur |
| Mme. | Nebrasse ELLOUZE | Encadrant |
| M. | Mohamed Salim BESSI | Encadrant |

***Dédicaces***

*Du profond de mon cœur, je dédie ce travail à tous ceux qui me sont chers,*

*A MES CHERS PARENTS, que ce travail soit l’expression de ma reconnaissance pour vos sacrifices consentis, votre soutien moral et matériel que vous n’avez cessé de prodiguer. Vous avez tout fait pour mon bonheur et ma réussite. Que dieu vous préserve en bonne santé et vous accorde une longue vie.*

*A MON CHER FIANCE, pour l'amour et le respect qu’il me porte, pour son aide, son encouragement et sa disponibilité.*

*A MES FRERES ET SŒURS, vous étiez toujours présents pour m’aider et m’encourager. Sachiez que vous serez toujours dans mon cœur.*

*A MON BINOME, pour le frère agréable qu’il était et qu’il restera pour moi.*

*A TOUS MES AMIS, qui n’ont cessé de m’encourager et de me soutenir.*

***Intissar***

*Avec tout respect et amour, je dédie ce travail à tous ceux qui me sont chers*

*A mes chers parents, en témoignage de leur affection, leur amour et leurs sacrifices,*

*A mon frère et mes sœurs, pour tout leur soutien moral, leur amour et leurs précieux conseils,*

*A ma chère coéquipière pour tout ce qu’elle a fait pour la réussite de ce stage,*

*A tous mes amis, en souvenirs des plus beaux instants qu'on a passés ensemble,*

*Aussi bien à tous ceux qui m'ont aidé*

*Merci...*

***Bilel***

***Remerciements***

La réalisation de ce projet a été possible grâce à l'intervention de plusieurs personnes à qui nous voudrions témoigner toute notre reconnaissance.

Nous tenons d'abord à exprimer nos sincères remerciements à notre encadrant au sein de l'Ecole Nationale d'Electronique et de Télécommunications de Sfax, ***Mme. Nebrasse ELLOUZE***, pour sa collaboration, sa générosité ainsi que ses précieuses recommandations.

Nos remerciements s'étendent également à ***Mr. Mohamed Selim BESSI***, responsable de notre stage, pour ses remarques sur le travail et pour tout le temps et l’effort qu’il nous a fournis pour que ce projet voie le jour. Sans oublier tous ceux qui ont été remarquablement coopératifs au sein de la banque BFI et qui ont assuré l'achèvement de ce stage dans les meilleures conditions.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce modeste travail.

Qu’ils puissent tous, trouver ici, l’expression de notre profonde gratitude et de notre grand respect.

Table des matières

[**Introduction Générale** 1](#_Toc421799390)

[***Chapitre 1 : Cadre général du projet*** 3](#_Toc421799391)

[1. Présentation de l’organisme d’accueil «BFI » 4](#_Toc421799392)

[2. Le Mobile Banking: actualités et perspectives 5](#_Toc421799393)

[2.1 Definition du Mobile Banking 5](#_Toc421799394)

[2.2 Connaissances et intérêt pour le Mobile Banking 5](#_Toc421799395)

[2.3 Le Mobile Banking en Tunisie 6](#_Toc421799396)

[2.4 Exemples d'applications Mobile-Banking existantes en Tunisie 9](#_Toc421799397)

[2.5 Risques de l'utilisation des applications mobile dans le secteur bancaire 10](#_Toc421799398)

[***Chapitre2 : Analyse des besoins et spécifications*** 12](#_Toc421799399)

[1. Spécification des besoins 13](#_Toc421799400)

[1.1 Spécification des besoins fonctionnels 13](#_Toc421799401)

[1.2 Spécification des besoins non fonctionnels 13](#_Toc421799402)

[2. Spécification d'architecture 14](#_Toc421799403)

[3. Analyse des besoins 14](#_Toc421799404)

[3.1 Description détaillée des cas d'utilisation 14](#_Toc421799405)

[3.1.1 Diagrammes de cas d'utilisation relatifs à l'administrateur 16](#_Toc421799406)

[3.1.2 Diagrammes des cas d'utilisation relatifs à l'abonné 16](#_Toc421799408)

[3.2 Description détaillée des diagrammes de séquences 22](#_Toc421799409)

[3.2.1 Diagramme de séquence « Authentification» 23](#_Toc421799410)

[3.2.2 Diagramme de séquence « Consultation de solde » 23](#_Toc421799411)

[3.2.3 Diagramme de séquence « Historique des opérations » 24](#_Toc421799412)

[3.2.4 Diagramme de séquence « Demande de carte » 24](#_Toc421799413)

[3.2.5 Diagramme de séquence « Demande de chéquier » 25](#_Toc421799414)

[3.2.6 Diagramme de séquence « Demande virement » 25](#_Toc421799415)

[***Chapitre 3 : Conception*** 27](#_Toc421799416)

[1. Diagrammes d’activités 28](#_Toc421799417)

[1.1 Diagramme d’activité «Authentification» 29](#_Toc421799418)

[1.2 Diagramme d’activité « Consultation de solde » 29](#_Toc421799419)

[1.3 Diagramme d’activité « Demande carte » 30](#_Toc421799420)

[1.4 Diagramme d’activité « Demande chéquier » 31](#_Toc421799421)

[1.5 Diagramme d’activité « Demande virement interbancaire» 32](#_Toc421799422)

[1.6 Diagramme d’activité « Demande virement intra bancaire » 33](#_Toc421799423)

[2. Diagramme de classes 34](#_Toc421799424)

[3. Etude technique de l'application 35](#_Toc421799425)

[3.1 Architecture avec une base de données externe 35](#_Toc421799426)

[3.2 Choix de technologie Middleware « REST » 36](#_Toc421799427)

[3.3 Choix de technologie Services Web « JSON » 37](#_Toc421799428)

[***Chapitre 4 : Réalisation*** 39](#_Toc421799429)

[1. Environnement de travail 40](#_Toc421799430)

[1.1 Architecture de l’application 40](#_Toc421799431)

[1.2 Modèle de données 41](#_Toc421799432)

[2. Environnement logistique 42](#_Toc421799433)

[2.1 Java EE 42](#_Toc421799434)

[2.2 Eclipse 42](#_Toc421799435)

[2.3 Android 43](#_Toc421799436)

[2.3.1 Le SDK Android 43](#_Toc421799437)

[2.3.2 ADT pour Eclipse 43](#_Toc421799438)

[2.4 Système de gestion de base de données (MYSQL) 43](#_Toc421799439)

[2.5 Serveur application (Tomcat 8) 44](#_Toc421799440)

[3. Implémentation 44](#_Toc421799441)

[3.1 Authentification 44](#_Toc421799442)

[3.2 Menu Principal 46](#_Toc421799443)

[3.3 Liste des comptes 46](#_Toc421799444)

[3.4 Consulter solde 47](#_Toc421799445)

[3.5 Historiques des opérations 48](#_Toc421799446)

[3.6 Demande carte 48](#_Toc421799447)

[3.7 Demande chéquier 50](#_Toc421799448)

[3.8 Demande Virement 50](#_Toc421799449)

[3.9 Déconnexion 52](#_Toc421799450)

[4. Déroulement du stage 52](#_Toc421799451)

[**Netographie** 55](#_Toc421799452)

[**Bibliographie** 56](#_Toc421799453)

[**Annexes**  57](#_Toc421799453)

**Table des figures**

[**Figure 1.** *Progression du mobile banking en Afrique* 6](#_Toc422091686)

[**Figure 2.** *Degré de popularité du mobile Banking en Tunisie (%)* 7](#_Toc422091687)

[**Figure 3.** *Degré d’intérêt au Mobile Banking (%)* 7](#_Toc422091688)

[**Figure 4.** *Les motifs du mobile Banking (%)* 8](#_Toc422091689)

[**Figure 5.** *Les facteurs de résistance au mobile Banking (%)* 8](#_Toc422091690)

[**Figure 6.** *Nombre de Trojans bancaires détectés pour appareils mobiles* 10](#_Toc422091691)

[**Figure 7.** *Diagramme du cas d’utilisation global* «*Administrateur*» 16](#_Toc422091692)

[**Figure 8.** *Diagramme du cas d’utilisation global* « *Abonné* » 16](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091693)

[**Figure 9.** *Description du cas d’utilisation « s’authentifier »* 17](#_Toc422091694)

[**Figure 10.** *Description du cas d’utilisation « consultation de solde»* 18](#_Toc422091695)

[**Figure 11.** *Description du cas d’utilisation « consultation de l'historique des opérations »* 18](#_Toc422091696)

[**Figure 12.** *Description du cas d’utilisation « Demande de carte »* 19](#_Toc422091697)

[**Figure 13.** *Description du cas d’utilisation « Demande de chéquier»* 20](#_Toc422091698)

[**Figure 14.** *Description du cas d’utilisation « Demande de virement »* 21](#_Toc422091699)

[**Figure 15.** *Diagramme de séquence « Authentification»* 23](#_Toc422091700)

[**Figure 16.** *Diagramme de séquence « Consultation de solde »* 23](#_Toc422091701)

[**Figure 17.** *Diagramme de séquence « Historique des opérations »* 24](#_Toc422091702)

[**Figure 18.** *Diagramme de séquence « Demande de carte »* 24](#_Toc422091703)

[**Figure 19.** *Diagramme de séquence « Demande de chéquier »* 25](#_Toc422091704)

[**Figure 20.** *Diagramme de séquence « Demande virement »* 25](#_Toc422091705)

[**Figure 21.** *Diagramme d’activité «Authentification»* 29](#_Toc422091706)

[**Figure 22.** *Diagramme d’activité « Consultation de solde »* 29](#_Toc422091707)

[**Figure 23.** *Diagramme d’activité « Demande carte »* 30](#_Toc422091708)

[**Figure 24.** *Diagramme d’activité « Demande chéquier »* 31](#_Toc422091709)

[**Figure 25.** *Diagramme d’activité « Demande virement interbancaire »* 32](#_Toc422091710)

[**Figure 26.** *Diagramme d’activité « Demande virement intra bancaire »* 33](#_Toc422091711)

[**Figure 27.** *Diagramme de classes* 34](#_Toc422091712)

[**Figure 28.** *Modèle de la communication entre l’application mobile et la base de données* 35](#_Toc422091713)

[**Figure 29.** *Liaison d'une application Android avec base de données* 36](#_Toc422091714)

[**Figure 30.** *Architecture REST* 37](#_Toc422091715)

[**Figure 31.** *Architecture à trois niveaux* 40](#_Toc422091716)

[**Figure 32.** *Modèle de données* 41](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091717)

[**Figure 33.** *Interface chargement ouverture application* 45](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091719)

[**Figure 34.** *Interface d’authentification* 45](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091718)

[**Figure 35.** *Vérification de l'Id et du code personnel* 45](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091720)

[**Figure 36.** *Interface échec d'authentification* 45](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091721)

[**Figure 37.** *Menu principal* 46](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091722)

[**Figure 38.** *Interface contact* 46](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091723)

[**Figure 39.** *Interface liste des comptes* 47](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091724)

[**Figure 40.** *Echec connexion à la base de données* 47](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091725)

[**Figure 41.** *Interface consulter solde* 47](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091726)

[**Figure 42.** *Interface consultation les historiques des opérations* 48](#_Toc422091727)

[**Figure 43.** *Interface demande carte* 49](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091729)

[**Figure 44.** *Confirmation demande carte* 49](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091730)

[**Figure 45.** *Interface de l’historique de demande carte* 49](#_Toc422091728)

[**Figure 46.** *Interface demande chéquier* 50](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091731)

[**Figure 47.** *Consultation l'état des demandes chéquier* 50](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091732)

[**Figure 48.** *Demande virement interne* 51](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091733)

[**Figure 49.** *Demande virement externe* 51](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091734)

[**Figure 50.** *Message d'échec si la code validation incorrect* 51](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091735)

[**Figure 51.** *Confirmation demande virement* 51](file:///C:\Users\Ballou\Downloads\rapport-PFE-Mobile_BankingVersion-Finale%20(1).docx#_Toc422091736)

[**Figure 52.** *Déconnexion* 52](#_Toc422091737)

**Table des tableaux**

[**Tableau 1**. *Applications similaires existantes* 9](#_Toc421797747)

[**Tableau 2**. *Les exigences techniques* 13](#_Toc421797748)

[**Tableau 3**. *Description des cas d’utilisation* 15](#_Toc421797749)

[**Tableau 4**. *Fiche de description du cas d’utilisation : « S'authentifier »* 17](#_Toc421797750)

[**Tableau 5**. *Fiche de description du cas d’utilisation : « Consultation de solde »* 18](#_Toc421797751)

[**Tableau 6**. *Fiche de description du cas d’utilisation:«Consulter historique des opérations»* 19](#_Toc421797752)

[**Tableau 7**. *Fiche de description du cas d’utilisation « Demande de carte »* 20](#_Toc421797753)

[**Tableau 8**. *Fiche de description du cas d’utilisation « Demande de chéquier»* 21](#_Toc421797754)

[**Tableau 9**. *Fiche de description du cas d’utilisation « Demande de virement »* 22](#_Toc421797755)

[**Tableau 10**. *Chronogramme du travail* 53](#_Toc421797756)

# Introduction Générale

Notre époque connaît l'avènement des nouvelles technologies qui impactent de jour en jour de nombreux secteurs d'activités. Comme le secteur de communication, le secteur informatique, industriel, commercial, etc., le secteur bancaire s'est influencé par ces nouvelles technologies et a connu une mutation importante en matière de distribution des services. Aux développement et perfectionnements des automates bancaires, des serveurs vocaux et des services par internet, est venue s'ajouter l'offre des services par mobile, communément appelé Mobile-banking.

En effet, Mobile-banking est d’une importance stratégique pour un opérateur de banque et de finance, un secteur où la concurrence s’accroît considérablement et nécessite des approches variées et innovantes pour acquérir plus de parts de marché.

Dans cette optique, certains clients de BFI[[1]](#footnote-1) ne possèdent pas un service Mobile-Banking et d’autres ont des solutions ne couvrant pas la totalité des fonctionnalités souhaitées ou qui ne suivent pas l'évolution technologique côté matériels et logiciels. Pour faire face à cette situation, BFI a décidé de développer un Mobile-banking pour satisfaire les besoins de ses clients. Néanmoins, un tel système est généralement appuyé sur des technologies hétérogènes comme il doit communiquer avec un système d’information bancaire pour achever son fonctionnement.

Dans ce contexte, l’objectif de notre travail consiste à concevoir et réaliser une application qui se chargera de servir tout client possédant un compte bancaire et souhaitant bénéficier des services bancaires à distance sans avoir à se soucier de l’interception de ses données personnelles. Afin de réaliser notre objectif, nous profitons de ce qu'offre l’architecture basée sur les web services pour assurer l’interopérabilité et la fluidité de la communication de notre application avec les systèmes hétérogènes.

Ce rapport est organisé en quatre chapitres. Nous commençons par un premier chapitre qui décrit le cadre général de notre projet pour une meilleure compréhension du contexte. Le deuxième chapitre est consacré à analyser et spécifier les besoins de notre application ainsi que l’architecture de notre solution. Dans le troisième chapitre, nous sommes chargés de présenter les différents aspects conceptuels de l’application.

Le dernier chapitre porte sur la partie réalisation. Il illustre en premier temps les détails de l’environnement matériel et logiciel utilisé afin de réaliser notre application. De même, il comporte les interfaces de notre application qui permettent à l’utilisateur d’y accéder. A la fin de ce chapitre, nous présentons le chronogramme de travail. Ce rapport est clôturé par une conclusion rappelant notre problématique ainsi que les éventuelles évolutions.

# *Chapitre 1 :* *Cadre général du projet*

**Introduction**

Dans ce chapitre, nous présentons l'organisme d'accueil en précisant les tâches effectuées et les services offerts aux clients. Nous comprendrons par la suite la notion du Mobile Banking, son intérêt, ses avantages et ses inconvénients pour entamer après sa situation en Tunisie.

# Présentation de l’organisme d’accueil «BFI »[[2]](#footnote-2)

 BFI est spécialisée depuis 1994 dans l’édition et l’intégration de solutions logicielles destinées aux banques et institutions financières. Bien implantée sur le marché africain, BFI a forgé une réputation de très haut niveau, en équipant plus de 180 clients dans plus de 15 pays différents notamment avec des institutions de renommée mondiale. Avec une expérience de plus que 20 ans d’installations en Afrique et d’une politique de proximité auprès de ses clients, BFI offre la mise en œuvre de projets couvrant : le suivi et le pilotage, l’intégration de systèmes, la méthodologie de test et de recette, la migration des données, la formation et le transfert de compétences, le conseil et l’accompagnement et enfin le support et la maintenance.

BFI allie la maîtrise des nouvelles technologies de l’information à la connaissance du métier de ses clients. Ses équipes pluridisciplinaires maîtrisent un large éventail de compétences, toutes au service d’un même objectif : optimiser la croissance de ses clients. BFI s’engage auprès de ses clients comme un véritable partenaire, en leur assurant la réussite de leurs projets dans le respect des délais de mise en œuvre et du budget. C’est ainsi que BFI a obtenu la confiance de ses clients et des principaux bailleurs de fonds mondiaux, tels que la Banque Mondiale (BM), la Banque Africaine de Développement (BAD) et Millenium Challenge Corporation (MCC).

# Le Mobile Banking: actualités et perspectives

Afin de faciliter la vie quotidienne, la plupart des gens sont dotés d'un Smartphone et emploient ainsi plusieurs types d'applications. De ce fait, pour accompagner cette croissance technologique et innover les services destinés à sa clientèle, les banques ont de leur côté, cherché à créer leur propre application pour Smartphone et essayé de permettre à leurs utilisateurs de gérer leurs comptes bancaires à partir de leur téléphone [1N].

## Definition du Mobile Banking

Le mobile Banking repose sur la réalisation d’opérations financières via l’utilisation du téléphone portable, offertes par les banques. Ces services se présentent le plus souvent sous les formes suivantes :

* Echanges d’informations financières : utilisées soit par l’institution financière pour alerte d’échéances de prêts, offres commerciales, confirmations d’opérations…, soit par le bénéficiaire dans le cas de services de consultation de solde, demande de relevés…
* Transactions financières : transferts d’argents, paiements des factures, dépôts, retraits, ou encore virements.

Le mobile Banking est lié alors aux opérations de « banque à distance », dématérialisés, pouvant être offerts avec ou sans compte bancaire, utilisant les nouvelles technologies (NTIC[[3]](#footnote-3)), en particulier le téléphone mobile et via des SMS ou des applications sur Smartphone [1B].

## Connaissances et intérêt pour le Mobile Banking

L’intérêt croissant que nous portons aujourd’hui au mobile banking résulte du succès spectaculaire des premières expériences de ce service généralisées à de nombreux pays en développement. De nos jours, plus de 74 pays offrent au moins une plateforme de mobile banking[[4]](#footnote-4). Grâce à ce service, les clients peuvent effectuer leurs opérations bancaires à tout moment et surtout rapidement.  Le mobile banking est un outil puissant permettant d’offrir des services aux milliards de personnes dans le monde qui disposent d’un téléphone portable. Parce qu’il s’affranchit des contraintes géographiques, le mobile banking présente des avantages par rapport à la banque traditionnelle. Il permet aux institutions bancaires et de la micro finance de s’appuyer sur ses services comme opportunité pour diminuer les coûts de leurs opérations ou pour mettre en place une structure évolutive à faible coût.

Pour la facilité que donne ce service, l’intérêt que porte le monde entier au mobile banking progresse de jour en jour. C’est le cas du continent africain, puisque la croissance exponentielle de la pénétration de la téléphonie mobile crée une opportunité de développer les services bancaires.

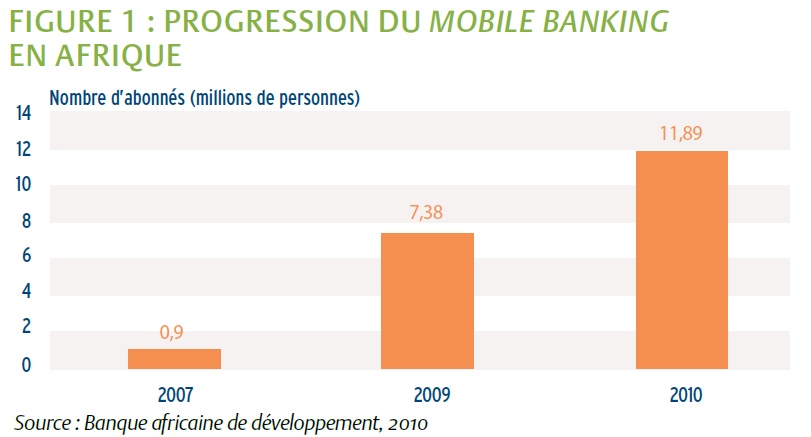


Figure 1. *Progression du mobile banking en Afrique[[5]](#footnote-5)*

Avec l’avènement du paiement par mobile, nous commençons à utiliser le mobile comme outil bancaire. En http par exemple, les utilisateurs essayent de se familiariser progressivement pour consultation de comptes et d’historique, effectuation des demandes de cartes, de chéquiers…

Selon les résultats donnés par http Mobile en Janvier 2012, deux français sur trois ont déjà utilisé leur téléphone mobile pour réaliser une opération bancaire et considèrent que ces applications vont se développer rapidement et être utilisées par un grand nombre de personnes dans un avenir proche [2B].

## Le Mobile Banking en Tunisie

Face à la croissance de l’utilisation de la technologie en Tunisie, les institutions financières ont lancé leurs produits de mobile banking et d’internet banking. Essayant de servir sa clientèle, les banques tunisiennes possèdent des sites Web et offrent des services bancaires à distance via un numéro vert, les téléphones mobiles ou grâce à Internet. Ces services concernent principalement la consultation du solde et des dernières opérations effectuées, la demande d’un chéquier, l’opposition sur les cartes égarées ou encore la recharge des téléphones portables.

En vue de cerner les potentialités du mobile banking, une enquête[[6]](#footnote-6) a été effectuée au cours du deuxième semestre de l’année 2012 auprès de 100 personnes habitant le grand Tunis. Cette enquête vise à :

* Dresser l’état du mobile Banking en Tunisie.
* Déceler ensuite les clients potentiels des services du mobile Banking.
* **Les résultats de l’enquête**

La population étudiée est assez équitablement partagée en termes de genre (55% d’hommes et 45% de femmes), appartenant majoritairement à la classe d’âge 18-39 (61%).

45

55

Figure 2. *Degré de popularité du mobile Banking en Tunisie (%)*

25

14

28

33

**Figure 3.** *Degré d’intérêt au Mobile Banking (%)*

59,6

24,6

14,0

1,8

Figure 4. *Les motifs du mobile Banking (%)*

34,1

29,5

22,7

9,1

2,3

2,3

Figure 5. *Les facteurs de résistance au mobile Banking (%)*

* **Discussion des résultats**

Comme d’autres études réalisées dans certains pays en développement, les résultats de cette enquête montrent qu’en dépit d’un nombre limité d’adhésions aux services du mobile banking, nous remarquons une grande prédisposition de la population tunisienne jeune, instruite et ayant des connaissances préalables à adopter les services financiers mobiles. Les principaux motifs sont le gain de temps et la commodité des services. Les opposants au mobile banking s’attachent à leurs habitudes de paiement et au contact personnel avec les partenaires commerciaux ainsi que la peur face à l’insécurité des opérations.

La Tunisie a donc intérêt de renforcer les premières expériences du mobile banking dont les résultats restent modérés. Les institutions financières devraient ainsi cibler leurs démarches de communication en cherchant à atteindre les jeunes ayant des habitudes de paiement en ligne ; de plus, les autorités publiques sont appelées à sensibiliser les agents aux avantages des services financiers à distance en renforçant le cadre juridique sur la sécurité et l’anonymat des opérations.

## Exemples d’applications Mobile-Banking existantes en Tunisie

Tableau 1. *Applications similaires existantes*

**Critères**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Application** | **Open Source** | **Portable** | **Plateforme** | **Services** |
| **BT Mobile[[7]](#footnote-7)** | Oui | Oui | Android  IOS | *Espace public :*   * Géo localisation des agences et des distributeurs * Consultation des cours des devises * Consultation des cours en bourse * Simulateur de crédits   *Espace adhérant :*   * Consultation : comptes, liste des chéquiers, encours de crédits, échéances, porte feuille titres, alertes configurées par le client * Affichage des mouvements du compte * Affichage du détail des transactions * Lancement d’une demande de chéquier * Mise en opposition d’une carte * Gestion des cartes * Emission de virements * Paiement des factures * Lancement d’une réclamation |
| **Banque de l’Habitat[[8]](#footnote-8)** | Oui | Oui | Android  IOS  Windows Phone  J2ME  BlackBerry | * Consultation : la liste des comptes et carte bancaire, historique des transactions * Effectuation des requêtes clients * Suivie des Alertes de BH * Demande de chéquier |
| **Attijari** **Mobile[[9]](#footnote-9)** | Oui | Oui | Android  IOS | * Consultation : soldes des comptes (Chèque, épargne et titre), dernières opérations des comptes * Effectuation des virements compte à compte * Commande de chéquier |
| **Banque Zitouna[[10]](#footnote-10)** | Oui | Oui | IOS  Android  BlackBerry  J2ME | * Consultation : liste des comptes, historique des transactions * Recherche multicritères de mouvements * Demande de chéquier * Consultation : messagerie, coordonnées des différentes agences de la Banque en Tunisie * Simulation des financements * Recherche de l’agence la plus proche * Contact de la Banque |

## Risques de l’utilisation des applications mobile dans le secteur bancaire

Considéré comme délicat, le secteur bancaire présente des dangers spécifiques aux services mobiles. Ces dangers se présentent le plus souvent sous les formes suivantes :

• *Attaques par hameçonnage :* un criminel repère les données d’accès et parvient de cette manière à accéder à des comptes.

• *Attaque «man in the browser» :* des codes malveillants permettent au cyberpirate de voler des données pour espionner pendant le login ou manipuler les données de transaction.

• *Perte de l’appareil :* une personne malintentionnée accède à l’application bancaire installée.

D’ailleurs, une enquête réalisée par Kaspersky Lab[[11]](#footnote-11) pendant le 2ème trimestre de l’année 2014, a dévoilé des données concernant l’évolution des menaces informatiques. En effet, le nombre total d’objets malveillants pour appareils mobiles a augmenté (1,7 fois plus qu’au trimestre précédent), ce qui s’illustre à travers le schéma d’après. Depuis le début de l’année 2014, le nombre de Trojans bancaires a presque quadruplé tandis qu’il a été multiplié par 14,5 fois si on le compare au chiffre d’il y a un an (juin 2013)[2N].

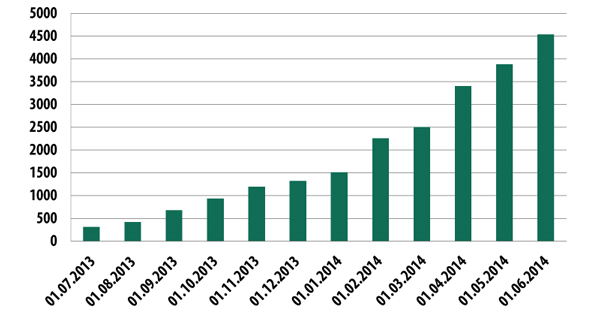
[](http://www.viruslist.com/fr/images/vlpub/q2-2014-mw_report_8.jpg)

Figure 6. *Nombre de Trojans bancaires détectés pour appareils mobiles*

Dans ce contexte, une importante mesure de sécurité est nécessaire pour protéger ce genre de menaces. Il convient par exemple de paramétrer le système d’exploitation pour empêcher l’installation de logiciels dont la source n’est pas fiable. Il est aussi nécessaire de mettre son Smartphone régulièrement à jour et, dans la mesure du possible, de le protéger à l’aide d’un antivirus et d’un pare-feu. De même, il est recommandé de s’armer de bon sens lorsque l’on navigue sur Internet depuis son Smartphone.

Par ailleurs, pour éviter que des personnes non autorisées accèdent aux données et aux applications de son dispositif mobile, il est nécessaire d’activer le verrouillage de l’appareil et de veiller à saisir le code de déverrouillage à l’abri des regards indiscrets.

**Conclusion**  
 Après avoir mis l’accent sur l’importance du Mobile Banking et son avenir dans le monde et en particulier en Tunisie, nous allons analyser et spécifier les besoins de notre application.

# *Chapitre 2 : Analyse des besoins et spécifications*

**Introduction**

Une étape essentielle de tout cycle de développement logiciel ou conceptuel consiste à effectuer une étude préalable. Le but de cette phase est de comprendre les besoins de notre projet en déterminant les fonctionnalités et les acteurs les plus pertinents et en identifiant les cas d’utilisation initiaux.

# Spécification des besoins

## Spécification des besoins fonctionnels

L’application à réaliser est une application de gestion d’opérations bancaires, elle permet à un abonné de réaliser toutes sortes d’opérations liées à son compte bancaire tels que :

* Consultation du solde de son ou ses comptes
* Suivi de l’historique des opérations
* Demande de carte ou chéquier
* Demande de virements

## Spécification des besoins non fonctionnels

L’application doit être en mesure de répondre à un certain nombre d’exigences techniques. Celles-ci se résument dans les objectifs présentés dans le tableau suivant :

Tableau 2. *Les exigences techniques*

|  |  |
| --- | --- |
| **Objectif** | **Description** |
| La fiabilité | -Les données fournies par l’application doivent être fiables. |
| Une solution ouverte et évoluée | -Possibilité d’amélioration de l’application par l’ajout d’autres modules pour assurer la souplesse, l’évolutivité et l’ouverture de la solution. |
| Confidentialité | -Les utilisateurs du système sont identifiés par leurs noms et leurs données personnelles. |
| Sécurité des transactions | -Toutes les données échangées entre le client et le serveur doivent être sécurisées. |

# Spécification d’architecture

La spécification d’architecture consiste à créer une représentation virtuelle d’une réalité de manière à mettre en valeur les points auxquels on s’intéresse.

 Dans le cadre de notre projet on a utilisé la méthodologie UML[[12]](#footnote-12) pour la modélisation des différents diagrammes puisqu’il convient pour toutes les méthodes objet et se prête bien à la représentation de l’architecture du système.

Le langage UML utilisé quotidiennement est un langage graphique qui permet la modélisation des données. En fait, avec UML, il est possible de modéliser toutes les étapes du développement d’une application informatique, de sa conception à la mise en route, grâce à des diagrammes [3B].

# Analyse des besoins

Dans cette section nous présentons la description détaillée des cas d’utilisation et les diagrammes de séquences associés.

## Description détaillée des cas d’utilisation

Un cas d’utilisation permet de décrire l’interaction entre un acteur et le système étudié.

* Acteur : un acteur est une entité externe qui interagit avec le système, comme une personne humaine ou un robot.
* Système : l’activité du système a pour objectif de satisfaire les besoins de l’acteur. Il fixe les limites de l’application en relation avec les acteurs qui l’utilisent (en dehors du système) et les fonctions qu’il doit fournir (à l’intérieur du système).

Dans ce qui suit, nous présentons un tableau englobant les cas d’utilisation des deux acteurs principaux : l’administrateur et le client abonné.

Tableau 3. *Description des cas d’utilisation*

|  |  |  |
| --- | --- | --- |
| **Acteur** | **Cas d’utilisation** | **Description** |
| Administrateur | Gérer compte abonné | Activer un nouveau compte d’abonné.  Désactiver un compte. |
| Gérer bénéficiaires de virement | Ajouter des bénéficiaires.  Supprimer des bénéficiaires |
| Abonné banque  (utilisateur mobile) | Consultation de solde | Connaissance du montant exacte du solde bancaire |
| Consultation des opérations | Consulter l’historique des opérations effectuées pendant une durée déterminée |
| Demande carte | Effectuer une demande de carte bancaire à partir de l’application et suivre l’état des demandes en cours |
| Demande chéquier | Effectuer une demande d’un ou plusieurs chéquiers et suivre l’état des demandes en cours |
| Demande virement | Effectuer une demande d’un virement interbancaire ou intrabancaire. |

### Diagrammes de cas d’utilisation relatifs à l’administrateur

L’administrateur est un agent bancaire connecté au système de la banque.

* **Diagramme de cas d’utilisation global**

****

Figure 7. *Diagramme du cas d’utilisation global* «*Administrateur*»

**🡺** Dans notre projet, on est demandé de s’intéresser uniquement à la partie relative au client abonné.

### Diagrammes des cas d’utilisation relatifs à l’abonné

* **Diagramme de cas d’utilisation global**

******



Figure 8. *Diagramme du cas d’utilisation global* « *Abonné* »

* **Diagrammes de cas d’utilisation détaillés**
* Description du cas d’utilisation **« s’authentifier »**

Figure 9. *Description du cas d’utilisation « s’authentifier »*

🡺 Le client est demandé d’introduire son login ainsi que son mot de passe pour s’authentifier à l’application.

Tableau 4. *Fiche de description du cas d’utilisation : « S’authentifier »*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | S’authentifier | |
| But | L’abonné s’identifie pour profiter des services de l’application | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
| *Pré conditions* | | *Post conditions* |
| * Client non identifié | | * Authentification |
| **Scénario nominal** | | |
| * Le client saisit ses coordonnées pour se connecter. * Si les données sont valides, le client est bien connecté à l’application. | | |

* Description du cas d’utilisation **« consultation de solde**»



Figure 10. *Description du cas d’utilisation « consultation de solde»*

Tableau 5. *Fiche de description du cas d’utilisation : « Consultation de solde »*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | Consulter solde | |
| But | L’abonné a la possibilité de consulter le solde de chacun de ses comptes | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
| *Pré conditions* | | *Post conditions* |
| * Client est authentifié * Choisir le numéro de compte | | * Affichage solde |
| **Scénario nominal** | | |
| * Le client se connecte à son application mobile par login et mot de passe. * Le client valide la requête d’affichage du solde via le bouton « Consulter solde » * Le système expose l’affichage du solde du compte correspondant. | | |

* Description du cas d’utilisation **« consultation de l’historique des opérations** »



Figure 11. *Description du cas d’utilisation « consultation de l’historique des opérations »*

Tableau 6. *Fiche de description du cas d’utilisation : « Consulter historique des opérations »*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | Consulter historique des opérations | |
| But | Consulter l’historique des opérations effectuées par le client | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
| *Pré conditions* | | *Post conditions* |
| * Client authentifié | | * Affichage de l’historique |
| **Scénario nominal** | | |
| * Le client se connecte à son application mobile par login et mot de passe. * Le client valide la requête d’affichage de l’historique via le bouton « Consulter l’historique des opérations» * Le système expose l’affichage des dernières opérations du mois. | | |

* Description du cas d’utilisation **« Demande de carte** »

**Figure 12.** *Description du cas d’utilisation « Demande de carte »*

Tableau 7. *Fiche de description du cas d’utilisation « Demande de carte »*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | Demande carte | |
| But | L’abonné effectue une demande de carte bancaire. | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
| *Pré conditions* | | *Post conditions* |
| * Client est authentifié * Choisir le type de la carte | | * Message de confirmation de prise en charge de la demande de carte. |
| **Scénario nominal** | | |
| * Le client se connecte à son application mobile par un login et un mot de passe. * L’utilisateur lance la commande de la carte via l’icône « demande de carte » * Le système affiche un message de succès. | | |

* Description du cas d’utilisation **« Demande de chéquier** »



Figure 13. *Description du cas d’utilisation « Demande de chéquier»*

Tableau 8. *Fiche de description du cas d’utilisation « Demande de chéquier»*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | Demande de chéquier | |
| But | L’abonné peut commander un ou plusieurs chéquiers. | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
|  | | |
| *Pré conditions* | | *Post conditions* |
| * Client authentifié * Déterminer type et nombre de chèques | | * Message de confirmation de prise en charge de la demande de chéquier. |
| **Scénario nominal** | | |
| * Le client se connecte à son application mobile par login et mot de passe. * L’utilisateur lance la commande de chéquiers via l’icône « demande de chéquier » * Le client doit déterminer le nombre de chèques et le type de chéquier à commander. * Le système affiche un message de succès. | | |

* Description du cas d’utilisation **« Demande de virement** »



Figure 14. *Description du cas d’utilisation « Demande de virement »*

Tableau 9. *Fiche de description du cas d’utilisation « Demande de virement »*

|  |  |  |
| --- | --- | --- |
| **Sommaire** | | |
| Titre | Demande virement | |
| But | L’abonné effectue une demande de virement | |
| Acteur | Abonné | |
| **Description des enchainements** | | |
| *Pré conditions* | | *Post conditions* |
| * Client est authentifié * Déterminer le type de virement | | * Message de confirmation de prise en charge de la demande client de virement. |
| **Scénario nominal** | | |
| * Le client se connecte à son application mobile par login et mot de passe. * L’utilisateur lance la demande de virement via l’icône « demande de virement » * Le client doit déterminer le type de virement : interbancaire ou intra bancaire. * Pour le type inter, le client doit entrer le code de validation de bénéficiaire. * Le système affiche un message de succès. | | |

## Description détaillée des diagrammes de séquences

Un diagramme de séquences est une représentation graphique et temporelle des interactions entre les acteurs et le système dans la formulation UML et dans le cadre d’un scénario d’un [diagramme des cas d’utilisation](http://fr.wikipedia.org/wiki/Diagramme_des_cas_d%27utilisation).

Pour simplifier la représentation, on place l’acteur principal à gauche du diagramme et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs et les objets.

### Diagramme de séquence « Authentification»



Figure 15. *Diagramme de séquence « Authentification»*

### Diagramme de séquence « Consultation de solde »



Figure 16. *Diagramme de séquence « Consultation de solde »*

### Diagramme de séquence « Historique des opérations »

Figure 17. *Diagramme de séquence « Historique des opérations »*

### 

### Diagramme de séquence « Demande de carte »



Figure 18. *Diagramme de séquence « Demande de carte »*

### Diagramme de séquence « Demande de chéquier »

Figure 19. *Diagramme de séquence « Demande de chéquier »*

### Diagramme de séquence « Demande virement »



Figure 20. *Diagramme de séquence « Demande virement »*

**Conclusion**

Au cours de ce chapitre, nous avons décrit les besoins fonctionnels et techniques attendus de notre application en présentant les diagrammes des cas d’utilisation et des séquences. Nous pouvons ainsi entamer la prochaine étape qui consiste à présenter les différents aspects conceptuels de l’application.

# *Chapitre 3 : Conception*

**Introduction**

Comme étant une phase fondamentale dans le cycle de vie d’un logiciel, la phase de conception a pour objectif de fixer les choix techniques et de préparer les solutions pour trouver comment le problème est résolu.

Ceci sera illustré par des diagrammes d’activité et des diagrammes de classes.

# Diagrammes d’activités

Le diagramme d’activité est un moyen graphique pour donner une vision d’ensemble sur des scénarios alternatifs et/ou d’exception. S’agissant d’un diagramme comportemental d’UML, il permet de modéliser un processus interactif, global ou partiel pour un système donné ([logiciel](http://fr.wikipedia.org/wiki/Logiciel), [système d’information](http://fr.wikipedia.org/wiki/Syst%C3%A8me_d%27information)).  On peut attacher un diagramme d’activités à n’importe quel élément de modélisation afin de visualiser, spécifier, construire ou documenter le comportement de cet élément.

Dans la phase de conception, les diagrammes d’activités sont particulièrement adaptés à la description des cas d’utilisation. Plus précisément, ils viennent illustrer et consolider la description textuelle des cas d’utilisation. De plus, leur représentation sous forme d’organigrammes les rend facilement intelligibles.

Les diagrammes d’activités permettent de spécifier des traitements a priori séquentiels et offrent une vision très proche de celle des langages de programmation impératifs comme C++ ou Java. Ainsi, ils peuvent être utiles dans la phase de réalisation, car ils permettent une description si précise des opérations qu’elle autorise la génération automatique du code.

Dans ce qui suit, nous présentons les diagrammes d’activités pour les cas d’utilisation de notre système.

## Diagramme d’activité «Authentification»

Figure 21. *Diagramme d’activité «Authentification»*

## Diagramme d’activité « Consultation de solde »



Figure 22. *Diagramme d’activité « Consultation de solde »*

## Diagramme d’activité « Demande carte »



Figure 23. *Diagramme d’activité « Demande carte »*

## Diagramme d’activité « Demande chéquier »



Figure 24. *Diagramme d’activité « Demande chéquier »*

## Diagramme d’activité « Demande virement interbancaire»



Figure 25. *Diagramme d’activité « Demande virement interbancaire »*

## Diagramme d’activité « Demande virement intra bancaire »



Figure 26. *Diagramme d’activité « Demande virement intra bancaire »*

# Diagramme de classes

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d’une telle modélisation.

Contrairement au diagramme de cas d’utilisation qui montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d’utilisation. Sous forme de schéma, il est utilisé pour présenter les [classes](http://fr.wikipedia.org/wiki/Classe_(informatique)) et les [interfaces](http://fr.wikipedia.org/wiki/Interface_(informatique)) des systèmes ainsi que les différentes relations entre celles-ci. Ce [diagramme](http://fr.wikipedia.org/wiki/Diagramme) fait partie de la partie [statique](http://fr.wikipedia.org/wiki/Statique) d’[UML](http://fr.wikipedia.org/wiki/Unified_modeling_language) car il fait abstraction des aspects temporels et [dynamiques](http://fr.wikipedia.org/wiki/Dynamique).

Le diagramme de classes ci-dessous nous permet d’avoir une vue statique de l’application. Il nous montre les relations entre les différentes entités composant notre application et il nous mènera vers la solution finale. À partir de ce schéma, nous retrouvons les corps des différentes classes de notre application.



Figure 27. *Diagramme de classes*

# Etude technique de l’application

Les différents utilisateurs de notre application y accèdent à travers les divers navigateurs via l’internet. La gestion des demandes de l’utilisateur est effectuée par le serveur d’application qui permet la communication avec le serveur de la base de données.

## Architecture avec une base de données externe

Pour faire communiquer l’application Android et notre base de données nous avons utilisés un Middleware. Ce Middleware, par définition, va organiser, adapter et traiter les échanges entre l’application et la base de données.[[13]](#footnote-13)

Ces échanges sont possibles grâce à des Web Services, avec des langages d’échanges adaptés tels que JSON, ou XML [2N].



**

Figure 28. *Modèle de la communication entre l’application mobile et la base de données*

Le Middleware, comme étant une entité qui tourne sur un serveur, va permettre de récupérer ou d’insérer des informations de la BD, et de répondre aux demandes de diverses sources (par exemple notre application Android). Celui-ci peut être développé avec des technologies comme JavaEE, PHP, .NET, ...

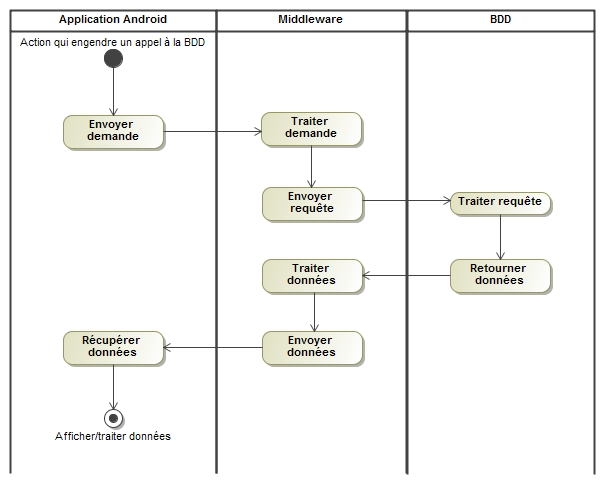
 Le schéma ci-dessous illustre comment s’organise la procédure d’interaction entre les deux entités.

Figure 29. *Liaison d’une application Android avec base de données*

## Choix de technologie Middleware « REST[[14]](#footnote-14) »

REST est un style d’architecture réseau pas standard utilisé pour Web Services. Il utilise des standards puisqu’il met l’accent sur la définition de ressources identifiées par des URI. Il utilise les messages du protocole http[[15]](#footnote-15) pour définir la sémantique de la communication client/serveur.

REST offre plusieurs avantages dont nous citons :

– L’utilisation du http, un protocole applicatif et non protocole de transport.

– Cache réseau : en respectant les entêtes et les requêtes préconisés dans la norme http, nous assurons une utilisation performante du serveur cache entre le serveur et les clients de l’application.

– Interface uniforme : chaque Web Service REST a une interface orientée autour des messages de http.

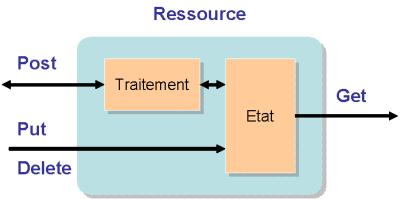
 Le protocole http, sur lequel se base l’architecture REST, comporte deux méthodes principales GET et POST et deux manières de transmettre des paramètres, soit dans l’URI, soit dans les données d’un formulaire.

Figure 30. *Architecture REST*

## Choix de technologie Services Web « JSON[[16]](#footnote-16) »

JSON est un format léger d’échange de données ouvert. Il est indépendant de la plateforme et propose un large éventail d’implémentations. Il est basé sur un sous-ensemble du langage de programmation JavaScript. Puisqu’il s’agit principalement d’un format texte complètement indépendant de tout langage, le format JSON peut être utilisé dans pratiquement tous scénario dans lequel les applications doivent échanger ou stocker des informations structurées sous forme de texte, s’agissant ainsi d’un langage d’échange de données idéal.

 A partir d’un texte JSON, nous pouvons analyser les valeurs, les chaines, les objets et les tableaux, comme nous pouvons sérialiser les types valeur en texte JSON à l’aide de fonctionnalités de l’espace de noms [**Windows.Data.Json**](https://msdn.microsoft.com/fr-fr/library/windows/apps/xaml/windows.data.json.aspx). Il permet de représenter de l’information structurée comme le permet [XML](http://fr.wikipedia.org/wiki/Extensible_Markup_Language) par exemple [4B].

**Les avantages de JSON :**

* La simplicité de mise en œuvre.
* La vitesse de traitement.
* Le fait d’être peu verbeux, ce qui le rend lisible aussi bien par un humain que par une machine.
* La facilité à apprendre, car sa syntaxe est réduite et non extensible.

**Conclusion**

Avant la réalisation de notre projet et pour faciliter la compréhension du système, une étude technique a été nécessaire afin de déterminer les solutions adoptées ainsi que les choix techniques fixés.

# 

# *Chapitre 4 : Réalisation*

**Introduction**

Dans ce chapitre, nous présentons l’architecture à partir de laquelle nous avons développé notre application, les différents outils utilisés ainsi que les composantes applicatives réalisées.

# Environnement de travail

## Architecture de l’application

Pour la récupération des données, l’application embarquée Mobile Banking se connecte à un serveur de bases de données distant, via Internet. Ce qui requiert ainsi l’intégration d’un serveur web entre l’application client et le serveur de bases de données.

C’est la raison pour laquelle l’architecture de notre application est à 3 niveaux (*architecture 3-tiers*), et elle est partagée entre :

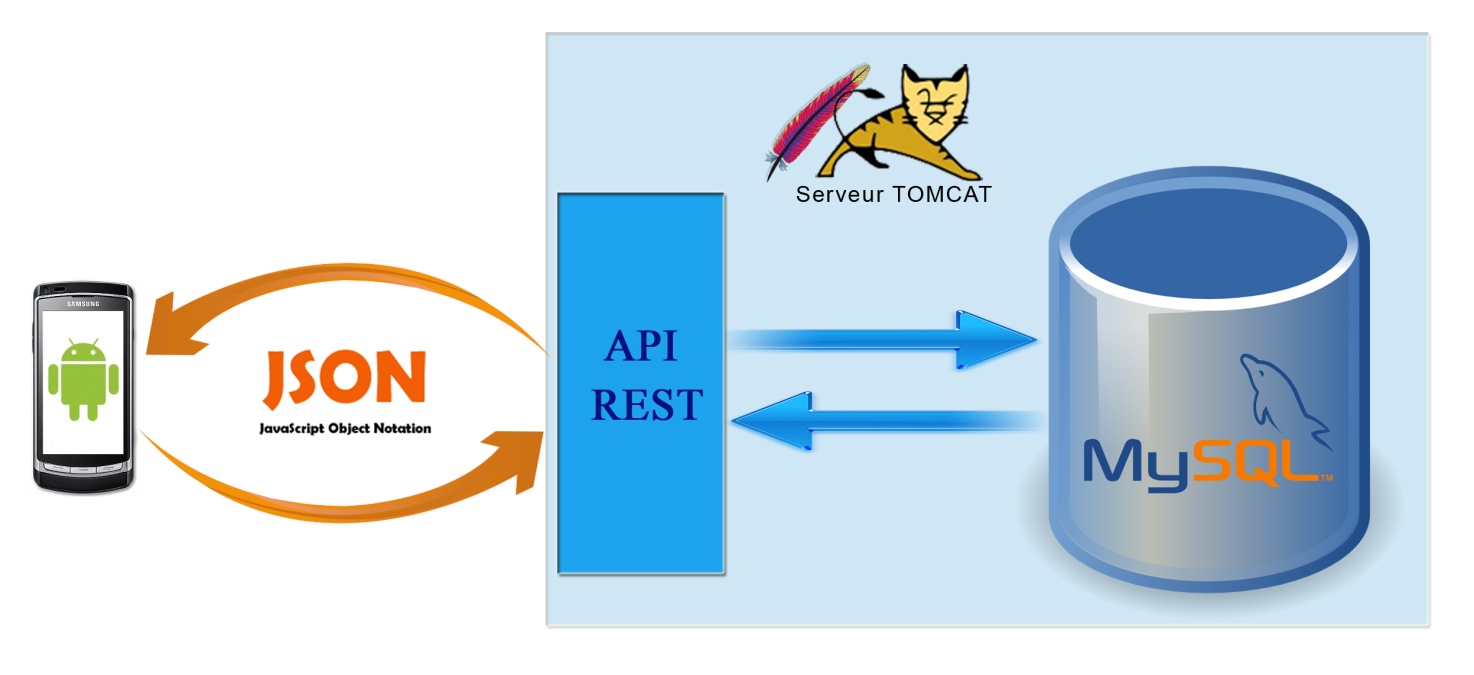
* Le client Android : conteneur d’application et demandeur de ressources.
* Le serveur de base de données : fournisseur des données au serveur web.
* Le serveur Web : gestionnaire de la communication entre deux environnements hétérogènes : le client Android et le serveur de base de données.

Figure 31. *Architecture à trois niveaux*

## Modèle de données

Le modèle conceptuel des données permet de représenter la structure du système d’information, du point de vue des données, et définit également les dépendances ou relations entre ces différentes données.

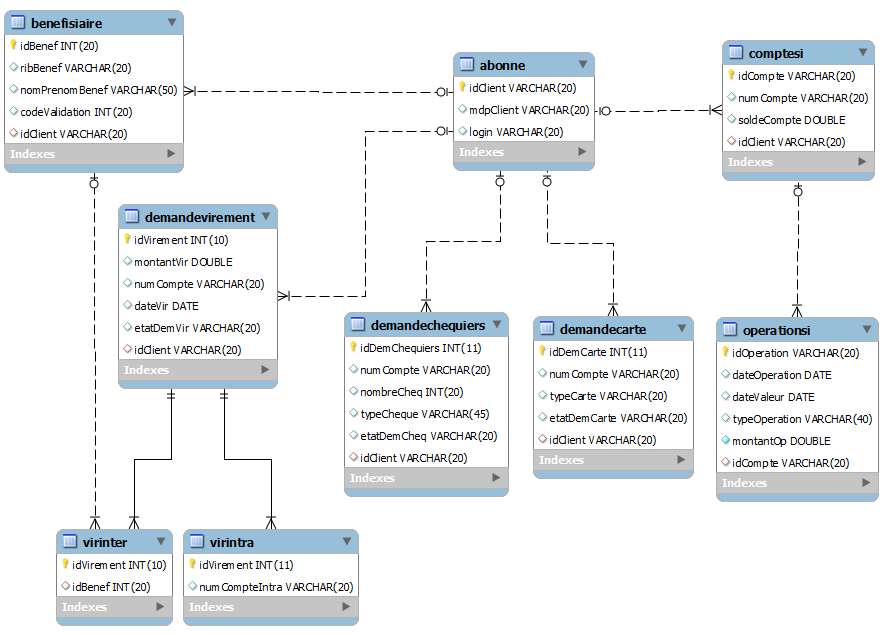
Dans notre cas, la base de données comporte les neuf tables nécessaires lors de la réalisation de notre travail. La figure ci-dessous définit les relations entre ces différentes tables.

Figure 32. *Modèle de données*

# Environnement logistique

## Java EE [[17]](#footnote-17)

C’est une spécification pour la technique Java   d’[Oracle](http://fr.wikipedia.org/wiki/Oracle_(entreprise)) plus particulièrement destinée aux applications d’entreprise. L’utilisation de Java EE pour développer et exécuter une application offre plusieurs avantages dont nous citons :

* Faciliter le développement de nouvelles applications à base de composants
* Une architecture d’applications basée sur les composants qui permet un découpage de l’application et donc une séparation des rôles lors du développement
* Intégration avec les systèmes d’informations existant.
* La possibilité de choisir les outils de développement et le ou les serveurs d’applications utilisés qu’ils soient commerciaux ou libres [5B].

## Eclipse

Eclipse se définit comme une plateforme universelle, l’IDE[[18]](#footnote-18) le plus largement utilisé pour la programmation Java. Figurant parmi les grandes réussites de l’open-source, il agit sur le monde du développement Java EE. C’est est un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d’aide à la programmation. La plateforme Eclipse a été conçue dans le but de faciliter la conception et l’utilisation des nouveaux langages de programmation apparus suite au développement d’Internet et des nouvelles technologies.

Dans le cadre de notre projet, nous avons utilisé la version Eclipse Luna, avec le plugin ADT[[19]](#footnote-19) de Google.

## C:\Users\Intissar\Desktop\images (1).jpgAndroid

Android est un système d’exploitation open source édité par Google pour appareils embarqués et mobiles, comme les Smartphones ou les tablettes.

Avant de démarrer le développement d’applications mobiles Android, il faut d’abord intégrer et paramétrer Android SDK[[20]](#footnote-20) dans la plateforme de développement Eclipse pour Java grâce au plugin ADT pour Eclipse.

### Le SDK Android

C’est un ensemble d’outils d’aide à la programmation pour concevoir des logiciels, jeux, applications mobiles, etc. pour un terminal et/ou un système d’exploitation spécifique. Il permet de télécharger tous les outils indispensables au développement d’applications.

### ADT pour Eclipse

ADT s’agit d’un plugin, c'est à dire un module qui complète un logiciel hôte pour lui apporter de nouvelles fonctionnalités. Ce plugin joue le rôle d’un pont entre Eclipse (environnement de développement) et Android. ADT permet de simplifier la construction des interfaces graphiques des applications et permet la conception graphique d’interfaces utilisateur, le debug distant sur un téléphone, la gestion de l’architecture de fichiers d’une application etc.

## Système de gestion de base de données (MYSQL)

MySQL est un Système de Gestion de Bases de Données Relationnelles (SGBDR) Open Source qui permet de gérer des bases de données. 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. Développé dans un souci de performances élevées en lecture, il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est [multi-thread](http://fr.wikipedia.org/wiki/Processus_l%C3%A9ger) et multi-utilisateur et utilise pour cela le langage SQL.

## C:\Users\Intissar\Desktop\téléchargement.jpgServeur application (Tomcat 8)

Apache Tomcat est un conteneur web libre de servlets et JSP Java EE figurant parmi un des nombreux projets de l’Apache Software Foundation. Il implémente les spécifications des servlets et des JSP du Java Community Process1, est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur http. De plus, a été écrit en [langage Java](http://fr.wikipedia.org/wiki/Java_(langage)). Il peut donc s’exécuter via la [machine virtuelle Java](http://fr.wikipedia.org/wiki/Machine_virtuelle_Java) sur n’importe quel [système d’exploitation](http://fr.wikipedia.org/wiki/Syst%C3%A8me_d%27exploitation) la supportant.

# Implémentation

Cette section nous permet de montrer quelques résultats de notre application dans sa phase d’essai.

## Authentification

Une étape primordiale pour la création d’application consiste en l’authentification. Un client désirant s’inscrire à l’application doit faire une demande auprès de l’agent de la banque pour avoir un identifiant et un code personnel.

Afin d’accéder aux fonctionnalités de l’application, un écran d’authentification s’offre à l’utilisateur dès l’exécution de cette dernière où il fera entrer ses données personnelles obtenus préalablement.

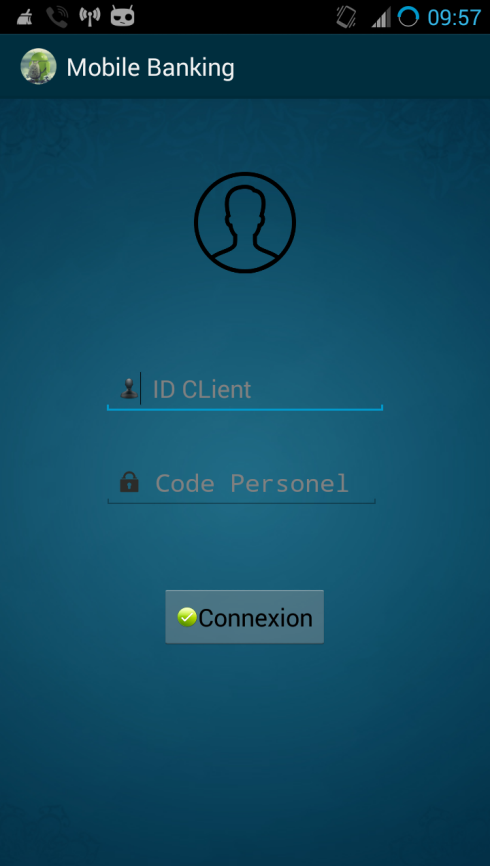


Figure 34. *Interface d’authentification*

Figure 33. *Interface chargement ouverture application*

Après l’insertion des données nécessaires, le système fait la vérification d’après la base de données. Au cas où l’utilisateur ne remplit pas tous les champs affichés, un message d’erreur apparaît. Ce message s’affiche aussi en l’absence de connexion ou lorsque l’utilisateur fournit de fausses données.

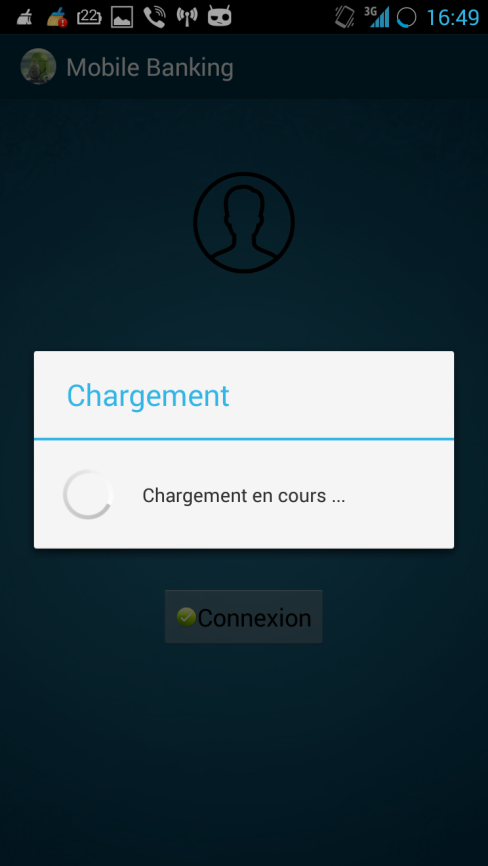
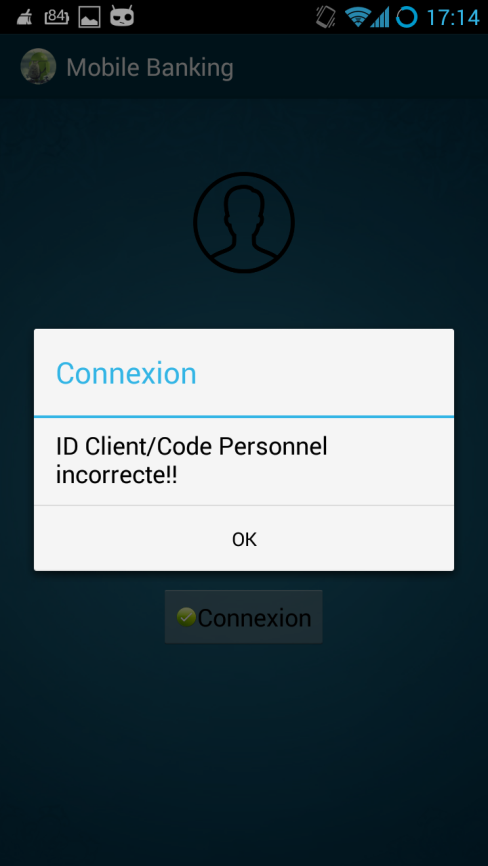


Figure 35. *Vérification de l'Id et du code personnel*

Figure 36. *Interface échec d'authentification*

## Menu Principal

Une fois les données sont valides, l’utilisateur accède à une interface qui contient le menu, où il peut gérer toutes les fonctionnalités de l’application. En plus de ce menu, il a la possibilité d’accéder à une interface appelée contact où il trouve toutes les informations concernant sa banque.

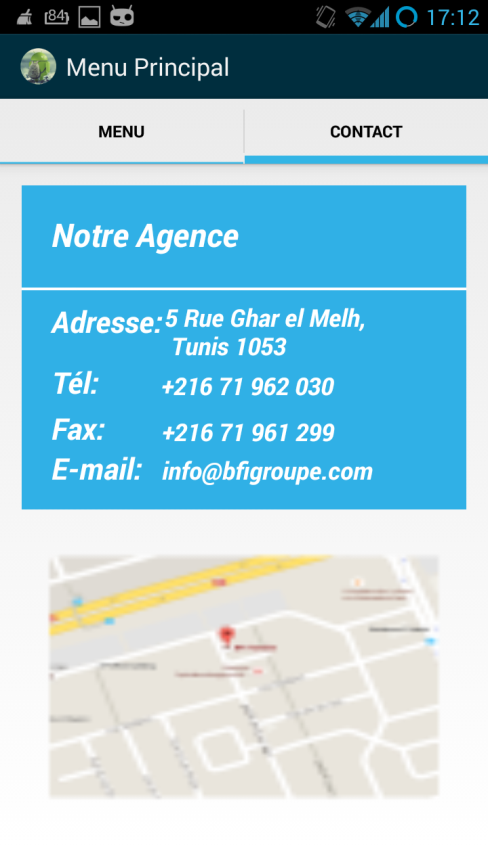
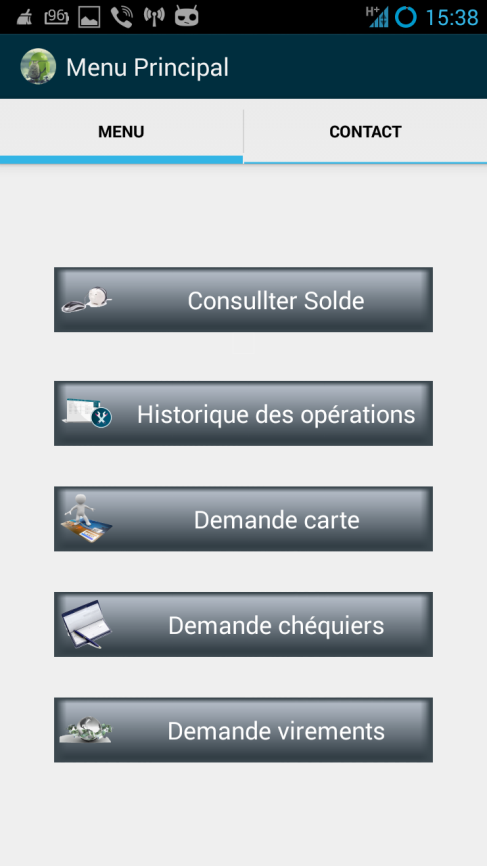
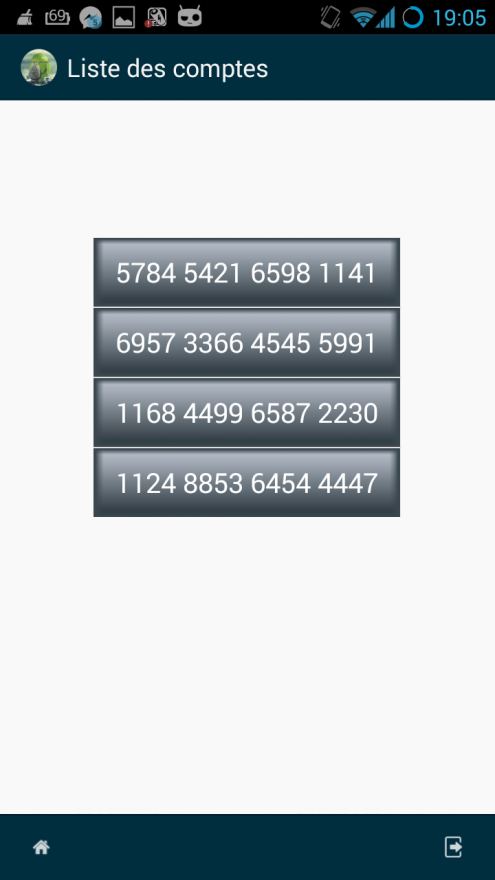


Figure 38. *Interface contact*

## Liste des comptes

Figure 37. *Menu principal*

Quand l’utilisateur choisit l’un des services possibles affichés dans le menu principal, le système affiche la liste des comptes correspondants à l’utilisateur connecté. Ce dernier est appelé à choisir le compte qu’il veut utiliser pour une éventuelle action.



**Figure 39.** *Interface liste des comptes*

## 

## Consulter solde

Après avoir sélectionné le numéro du compte, l’abonné, et en cliquant sur « Consulter Solde», ouvre une page qui lui affiche le solde du compte en question. En cas de problème, le système affiche un message d’erreur.



Figure 41. *Echec connexion à la base de données*

Figure 40. *Interface consulter solde*

## Historique des opérations

L’utilisateur peut consulter toutes les opérations effectuées sur son compte en cliquant sur « Historique des opérations ». Il accède alors à une interface qui affiche les détails comme la date, le type et le montant de chaque opération correspondant au numéro du compte sélectionné.



Figure 42. *Interface consultation les historiques des opérations*

## Demande carte

En cliquant sur «Demande carte», l’utilisateur accède à une interface qui lui permet d’obtenir une nouvelle carte bancaire correspondant au numéro du compte sélectionné préalablement. La demande d’une nouvelle carte consiste à spécifier son type (s’agissant d’une carte interne, une carte Visa ou une carte Gold). Ensuite, le client est demandé à confirmer le choix effectué. Il peut tout de même l’annuler.

En cas de confirmation de la demande, un message s’affiche pour indiquer le succès de l’opération.

******

Figure 43. *Interface demande carte*

Figure 44. *Confirmation demande carte*

L’interface «Demande carte» permet aussi à l’utilisateur de vérifier l’état de la demande. Il suffit de cliquer sur Historique pour afficher toutes les demandes de carte effectuées auparavant ainsi que leurs états.

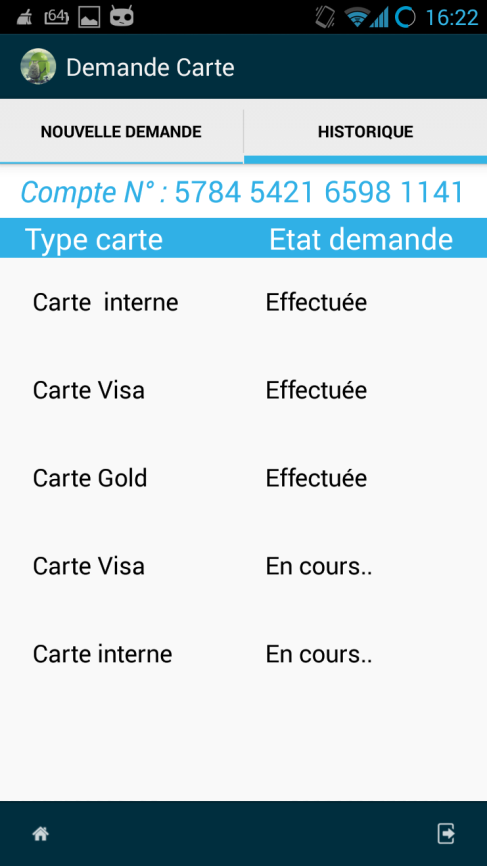


Figure 45. *Interface de l’historique de demande carte*

## Demande chéquier

Après avoir sélectionné le compte pour lequel la demande va s’effectuer, la demande d’un nouveau chéquier s’effectuera en précisant le nombre de chèques ainsi que leurs types. Ensuite, le client peut confirmer le choix effectué comme il peut l’annuler.

En cliquant sur historique, une liste de l’historique des demandes effectuées avec leurs états s’affiche à l’utilisateur.

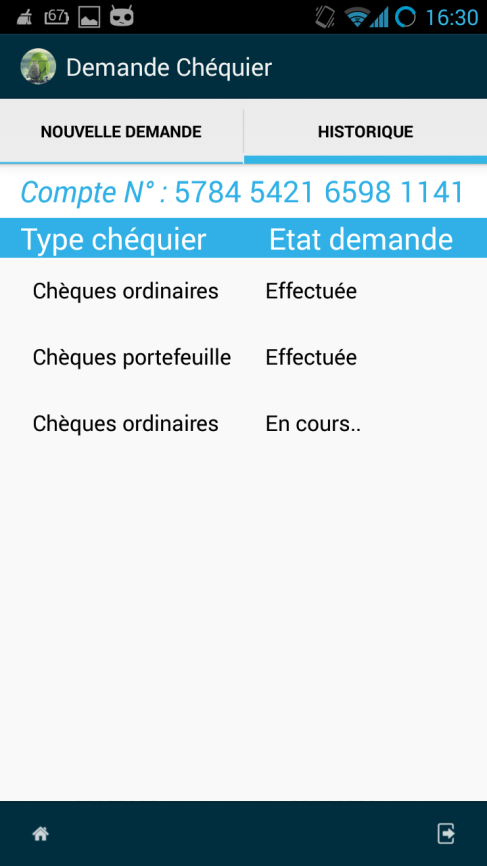


Figure 47. *Consultation l'état des demandes chéquier*

Figure 46. *Interface demande chéquier*

## Demande Virement

Il existe deux types de virements : le virement intra c’est-à-dire entre les propres comptes bancaires du client, et le virement inter qui se fait vers un bénéficiaire de la même banque ou d’une autre banque.

L’ajout du bénéficiaire se fait auprès l’agent de la banque suite à la demande du client, et chaque bénéficiaire a un code spécifique donné par cet agent.

Dans le cas de la demande interne, l’utilisateur spécifie le numéro du compte à débiter ainsi que et le montant qu’il veut transférer. Pour la demande externe, on précise l’identité du bénéficiaire, le code de validation et le montant à transférer.

Dans les deux cas, l’utilisateur doit confirmer son choix à la fin de chaque opération.

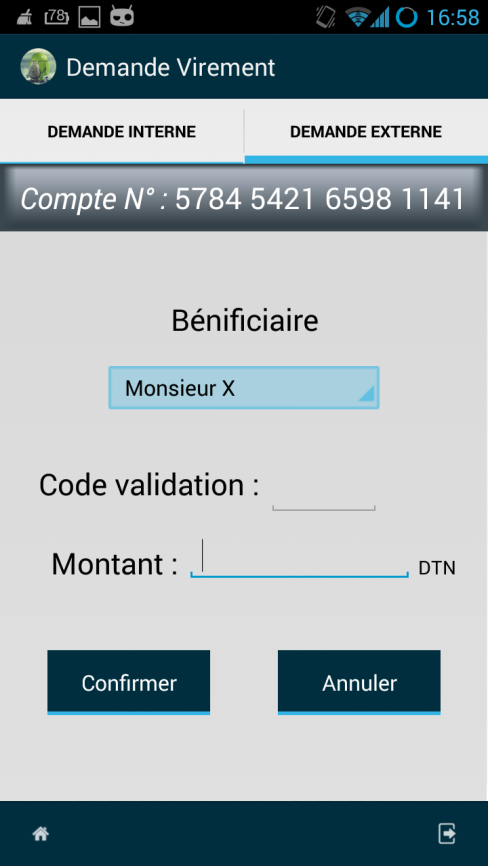


Figure 49. *Demande virement externe*

Figure 48. *Demande virement interne*

Si le code validation est incorrect, un message d’erreur s’affiche. Dans le cas contraire, un message de confirmation s’affiche, indiquant le succès de l’opération.

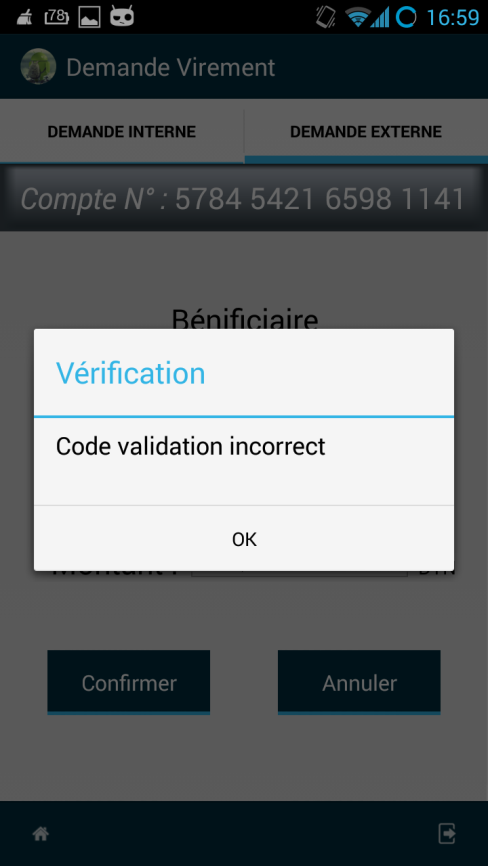


Figure 50. *Message d'échec si la code validation incorrect*

Figure 51. *Confirmation demande virement*

## E:\PC-DataPFE\PFE GII 2015\Rapport PFE ING-2015\Screenshots\Screenshot_2015-012.pngDéconnexion

Pour se déconnecter de l’application, il suffit de cliquer sur l’icône de sortie qui se trouve en bas de toutes les pages de l’application.

Un message s’affiche demandant à l’utilisateur de confirmer sa demande ou de l’annuler.



Figure 52. *Déconnexion*

# Déroulement du stage

Notre stage s’est étalé sur une durée de quatre mois et s’est déroulé en respectant les objectifs définis préalablement. Cette période est répartie de manière à décomposer le travail sur cinq intervalles dont certains se sont déroulés simultanément.

Nous présentons dans le tableau ci-dessous le chronogramme qui illustre la répartition des différentes tâches menées au cours de notre stage. Il détaille les taches effectuées ainsi que les durées correspondantes.

Tableau 10. *Chronogramme du travail*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Février | | | | | Mars | | | | | | Avril | | | | Mai | | | |
|  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| **Documentation et étude théorique** | |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| **Etude de l’existant** | |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| **Analyse et spécifications des besoins** | |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| **Conception et recherche** | Diagrammes de classes |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  |  |  |  |
| Recherche sur les technologies |  |  |  |  |  |  | |  |  |  | |  |  |  |  |  |  |  |  |
| Diagrammes d’activités |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| **Initiations, implémentation et tests** | Choix des technologies et initiation |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| Développement de l’application |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |
| Tests |  |  |  |  | |  | |  |  |  |  |  |  |  |  |  |  |  |  |
| **Rédaction du rapport** | |  |  |  |  | |  | |  |  |  | |  |  |  |  |  |  |  |  |

**Conclusion**

Dans ce dernier chapitre, nous avons entamé la réalisation de notre application. Nous avons commencé par présenter en premier temps les détails de l’environnement matériel et logiciel adopté pour réaliser notre travail. Nous avons ensuite précisé les interfaces de notre application qui permettent à l’utilisateur d’y accéder par la suite.

**Conclusion Générale**

L’idée motrice de notre projet de fin d’étude consiste à développer un Mobile banking générique souple et flexible, comme une solution permettant de couvrir une gamme de services bancaires et surtout satisfaisant les besoins des clients. De plus, elle permet de dialoguer avec n’importe quel système d’information puisqu’elle n’est pas soumise aux contraintes de l’infrastructure informatique sous-jacente de ces systèmes d’information. Ce qui permet donc de fournir à l’entreprise toute la latitude dont elle a besoin.

Ce stage fut d’un apport considérable. En effet, il nous a permis d’approfondir nos connaissances théoriques, acquises tout le long de notre formation, par la pratique des nouvelles technologies. Cette expérience nous a permis de maîtriser le langage de modélisation UML, d’utiliser dans un contexte réel les outils de développement Android à savoir le SDK Android, sous lequel, le développement n’a pas été une tâche facile.

Il nous a également permis de découvrir comment se passe l’intégration d’une application sur un serveur web distant ainsi que l’utilisation du langage JSON pour gérer la communication des données entre deux environnements hétérogènes qui sont le client Android et le serveur de bases de données.

De plus, nous avons offert à BFI une solution englobant diverses fonctionnalités. Se caractérisant par sa souplesse et sa flexibilité, cette solution est interopérable avec tout système d’information bancaire. Effectivement, elle facilite la tâche d’installation et de maintenance puisqu’elle est simplement paramétrable c’est-à-dire qu’elle s’adapte à toute modification de systèmes d’information, nous arrivons donc à un gain de temps, des coûts et des ressources.

Pour en finir, notre application reste toujours ouverte à des améliorations que ce soit en ajoutant d’autres services Mobile-Banking tel que l’accès aux opérations en bourse, ou en s’adaptant à l’évolution du domaine informatique.

# Netographie

[[1N] : http://www.memoireonline.com](file:///C:\Users\Asus\Downloads\%5b1N%5d :%20http:\www.memoireonline.com)

[2N] : <http://www.viruslist.com>

[3N] : [http://tutorielandroid.francoiscolin.fr](http://tutorielandroid.francoiscolin.fr/bdd.php)

[Dev] : http://developer.android.com

[Dev] : <http://www.openclassrooms.com>

[Dev] : <http://www.commentcamarche.net>

[Dev] : [http://www.tutos-android.com](http://www.tutos-android.com/)

[Dev] : <http://www.androidhive.info>

# Bibliographie

[1B] : Conditions générales d’accès et d’utilisation des services mobile banking

[2B] : Mobile Banking in Africa: Taking the Bank to the People, 2010

[3B] : UML Diagrammes de classes, Delphine Longuet

[4B] : [Cours Android, Jean François Lalande](http://www.univorleans.fr/lifo/Members/JeanFrancois.Lalande/enseignement/android/cours-android.pdf)

[5B] : Introduction au J2EE – Java 2 Enterprise Edition, Benjamin LIENS Société ITK.

***Annexes***

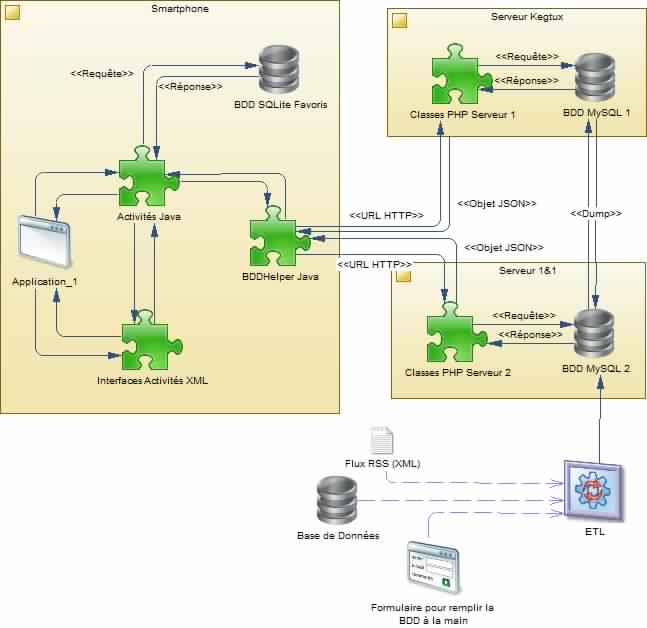
# Web service

Un Web Service est un programme informatique reposant sur une architecture réseau client serveur. La spécificité des Web Services est l'utilisation de HTTP comme support des messages entre clients et serveur. Un Web Service permet donc d'ouvrir sur le réseau une application, la modalité d'accès reposant in fine sur un protocole dont les messages seront transportés par HTTP. Présenté comme cela, les différences entre un Web Service et un site web ne semble pas évidentes... L'idée sous-jacente est qu'un Web Service va pouvoir être facilement manipulé par une interface de programmation (API), alors qu'un simple site web par opposition est fait pour être utilisé par un être humain. Les avantages architecturaux sont identiques à des architectures logicielles telles que Corba, RMI (Java) ou DCOM (Microsoft), et permettent à des composants logiciels écrits dans divers langages et tournant sur des machines différentes de communiquer.

Les autres avantages sont liés à l'utilisation d'HTTP pour le transport:

– port 80 généralement ouvert, facile à intégrer dans un réseau que ce soit en interne ou sur internet.

– HTTP : protocole bien connu et simple à aborder (protocole texte)



## Principes de REST

REST est un style d’architecture qui repose sur le protocole HTTP : On accède à une ressource (par son URI unique) pour procéder à diverses opérations (GET lecture / POST écriture / PUT modification / DELETE suppression), opérations supportées nativement par HTTP.

Avantages de REST:

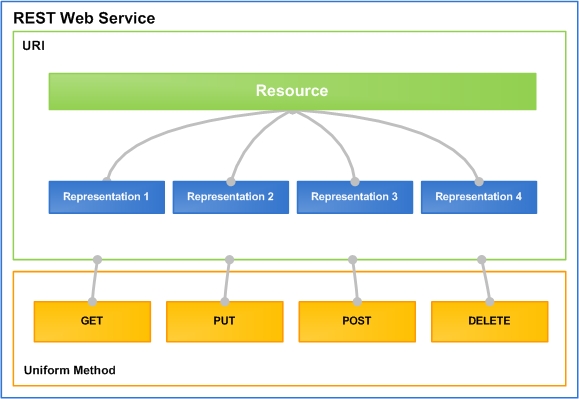
– HTTP protocole applicatif et non protocole de transport

– cache réseau : en respectant les entêtes et les requêtes préconisés dans la norme HTTP, on permet ainsi l'utilisation efficace de serveur cache entre le serveur et les clients de l'application.

– interface uniforme: chaque Web Service REST a une interface orientée autour des messages de http

### Principes d’une architecture REST[[21]](#footnote-21)

La figure ci dessous montre les principes de conception et des composants qui composent un service web REST.



## Génération d'un message

Pour générer un message, il faut suivre les étapes suivantes :

– définir la ou les ressources visées (URI collection, URI member)

– requête : définir l'action demandée, et donc le type de message : GET, PUT, POST, DELETE

– réponse: décider du code d'erreur

– code de la représentation de la ressource : ce codage se prête bien à une programmation orientée objet et à l'utilisation de bibliothèques de sérialisation/désérialisation. En suivant ces principes cela donne :

– **Pour le serveur**, décodage d'une requête (message issu d'un client) :

- identifier la ressource visée

- identifier l'action demandée par le type de requête http

- en fonction de la requête, procéder à l'analyse de la représentation des ressources fournies (PUT et POST) dans la requête

- en fonction des 2 étapes précédente, procéder à la résolution de l'action demandée

- générer la réponse : représentation de la ressource incluse dans la réponse, génération du code résultat

– **Pour le client**, décodage d'une réponse HTTP (message en provenance du serveur) :

- décodage du code résultat

- en fonction du résultat, analyse de la représentation de la ressource incluse dans la réponse

Les principes de programmation d'une application RESTful sont donc très proches d'une manipulation basique de HTTP ce qui rend relativement aisée une réalisation que ce soit dans le cadre d'une application existante ou dans le cadre d'un nouveau développement.

### Les formats d’échange[[22]](#footnote-22)

REST n’impose ni ne revendique un format d’échange entre client et serveur.

Vous êtes libre de représenter vos données en XML, en JSON, en PHP sérialisé, en MessagePack ou dans tout autre dialecte de votre propre cru (sans oublier que le but est souvent d’exposer des services vers l’extérieur).

Il n’est pas rare que les services REST permettent au client d’indiquer le format dans lequel ils souhaitent dialoguer, sous la forme par exemple d’un paramètre supplémentaire dans l’URL, ou plus simplement grâce aux en tête HTTP en spécifiant le content-type.



## JSON

Que ce soit au niveau du serveur ou au niveau des clients, une API RESTful manipule des ressources par une représentation de celles-ci. Le principe en pratique est de passer de la représentation utilisée en interne, que ce soit sur le client ou le serveur, à la représentation utilisée dans le message HTTP, requête ou réponse. Concernant la spécification du format de représentation utilisé dans le message : comme spécifié plus haut, les messages d'une application RESTful sont indépendants les uns des autres ce qui implique en particulier que le codage de la représentation des ressources peut être différent entre deux messages et donc qu'il faut spécifier dans le message le codage utilisé.

JSON est l'acronyme de JavaScript Object Notation. C'est un format texte qui permet de représenter des données et de les échanger facilement à l'instar d'XML. JSON est un sous ensemble d'ECMAScript (JavaScript) et est décrit dans le RFC 4627 [4]. Ce sous ensemble de JavaScript permet de décrire le modèle objet de JavaScript.

Deux types de structures sont disponibles :

– Objet : une collection de paire nom/valeur, i.e un tableau associatif.

– Tableau : une liste ordonnée de valeurs.

Les valeurs peuvent être des types suivants : booléen, chaîne de caractères, nombre, ou valeur nulle. (boolean, string, number, null), ou une des structures ci-dessus. La syntaxe est celle de JavaScript, simpliste.

Voici par exemple la description d'un utilisateur avec une adresse et des numéros de téléphone :

{

"nom":"Bob",

"age":34,

"addresse": { "rue":"avenue Grande", "ville":"Rio", "code":86945},

"telephone":[{"type":"maison", "numero" : 123456}, {"type":"portable", "numero": 654321}]

}

# LA SERVLET[[23]](#footnote-23)

Une servlet est un programme qui s'exécute côté serveur en tant qu'extension du serveur. Elle reçoit une requête du client, elle effectue des traitements et renvoie le résultat. La liaison entre la servlet et le client peut être directe ou passer par un intermédiaire comme par exemple un serveur http.

Même si pour le moment la principale utilisation des servlets est la génération de pages html dynamiques utilisant le protocole http et donc un serveur web, n'importe quel protocole reposant sur le principe de requête/réponse peut faire usage d'une servlet.

Ecrite en Java, une servlet en retire ses avantages : la portabilité, l'accès à toutes les API de Java dont JDBC pour l'accès aux bases de données, ...

Une servlet peut être invoquée plusieurs fois en même temps pour répondre à plusieurs requêtes simultanées.

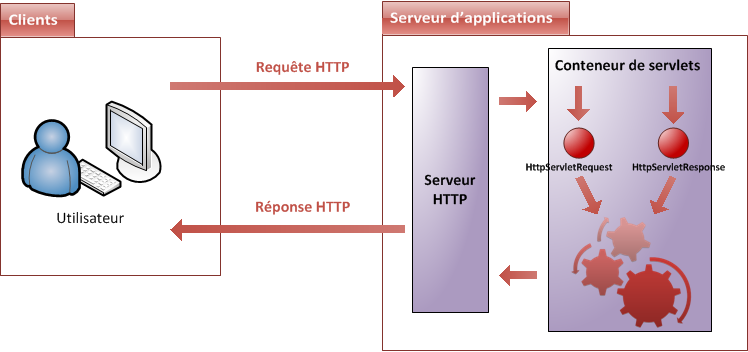
Dans une architecture Client/Serveur trois tiers, la servlet se positionne dans le tiers du milieu entre le client léger chargé de l'affichage et la source de données.

## Le fonctionnement d'une servlet

Pour exécuter une servlet, il suffit de saisir une URL qui désigne la servlet dans un navigateur.

1. Le serveur reçoit du navigateur la requête http qui a recours à une servlet
2. Si c'est la première sollicitation de la servlet, le serveur l'instancie. Les servlets sont stockées (sous forme de fichiers .class) dans un répertoire particulier du serveur. Ce répertoire dépend du serveur d'applications utilisé. La servlet reste en mémoire jusqu'à l'arrêt du serveur. Certains serveurs d'applications permettent aussi d'instancier des servlets dès le lancement du serveur.  
   Au fil des requêtes, la servlet peut être appelée par plusieurs threads lancés par le serveur. Ce principe de fonctionnement évite d'instancier un objet de type servlet à chaque requête et permet de maintenir un ensemble de ressources actives telles qu'une connexion à une base de données.
3. Le serveur crée un objet qui représente la requête http ainsi que l'objet qui contiendra la réponse et les envoie à la servlet
4. La servlet crée dynamiquement la réponse sous forme de page html transmise via un flux dans l'objet contenant la réponse. La création de cette réponse utilise bien sûr la requête du client mais aussi un ensemble de ressources incluses sur le serveur telles de que des fichiers ou des bases de données.
5. Le serveur récupère l'objet réponse et envoie la page html au client.

* [***HttpServletRequest***](http://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpServletRequest.html) : cet objet contient la requête HTTP, et donne accès à toutes ses informations, telles que les en-têtes (headers) et le corps de la requête.
* [***HttpServletResponse***](http://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpServletResponse.html) : cet objet initialise la réponse HTTP qui sera renvoyée au client, et permet de la personnaliser, en initialisant par exemple les en-têtes et le corps (nous verrons comment par la suite).[[24]](#footnote-24)



### **Développement d’un Mobile Banking sur Android**

#### Bilel MAKNI

&

#### Intissar BEN BOUZID

**Résumé**

De nos jours, les applications mobiles pour Smartphones demeurent une nouvelle technologie très répandue qui s’étale sur tous les domaines pour atteindre le domaine financier et bancaire. En effet, les banques tunisiennes cherchent à disposer d’une application mobile qui facilite la gestion des comptes bancaires de leurs clients.

Dans ce cadre, ce projet réalisé au sein de la BFI a comme objectif de développer une application sur Android qui permet de couvrir le maximum de fonctionnalités de gestion de portefeuille bancaire.

**Abstract**

Nowadays, mobile applications for Smart phones remain a widespread new technology that spans all domains to reach the financial and banking field. Indeed, Tunisian banks are looking to have a mobile application that facilitates the management of bank accounts of their clients.

In this context, this project realized within the BFI aims to design and develop an application on Android that allows covering the maximum of bank holding management features.

**تلخيص**

تمثل التطبيقات النقالة للهواتف الذكية تقنية جديدة تمتد على جميع المجالات لتشمل المجال المالي والمصرفي. البنوك التونسية تسعى لتكون لها تطبيقات تسهل إدارة الحسابات المصرفية لعملائها.

في هذا السياق، يتمثل الهدف من هذا المشروع في تطوير تطبيق على جهاز Android يمكن من تغطية أهم الوظائف المصرفية لإدارة الحسابات.

**Mots clés** : Restful, service web, Android, Mobile Banking, JSON

**Key-words** : Restful, web service, Android, Mobile Banking, JSON

1. BFI: Banque Finance Internationale. [↑](#footnote-ref-1)
2. Selon www.bfigroupe.com [↑](#footnote-ref-2)
3. Nouvelles technologies de l'information et de la communication. [↑](#footnote-ref-3)
4. Le mobile banking, une opportunité pour la réduction de l’exclusion financière en Tunisie. ZIADI Latifa. [↑](#footnote-ref-4)
5. Source: Banque Africaine de développement. [↑](#footnote-ref-5)
6. Le mobile banking, une opportunité pour la réduction de l’exclusion financière en Tunisie, ZIADI Latifa. [↑](#footnote-ref-6)
7. Selon www.bt.com.tn [↑](#footnote-ref-7)
8. Selon www.bh.com.tn/mobile [↑](#footnote-ref-8)
9. Selon www.attijariwafabank.com/AttijariMobile [↑](#footnote-ref-9)
10. Selon Android Market. [↑](#footnote-ref-10)
11. Société de solutions de sécurité informatique. [↑](#footnote-ref-11)
12. Unified Modeling Language. [↑](#footnote-ref-12)
13. BD: Base de Données. [↑](#footnote-ref-13)
14. Representational State Transfer, inventé par [Roy T. Fielding](http://www.ics.uci.edu/~fielding/). [↑](#footnote-ref-14)
15. HyperText Transfer Protocol. [↑](#footnote-ref-15)
16. JavaScript Object Notation. [↑](#footnote-ref-16)
17. Anciennement, J2EE (Java2 Enterprise Edition) [↑](#footnote-ref-17)
18. Integrated Development Environnement. [↑](#footnote-ref-18)
19. Android Development Tools. [↑](#footnote-ref-19)
20. Software Devlopment Kit. [↑](#footnote-ref-20)
21. Source : Introduction to Apache Wink [↑](#footnote-ref-21)
22. Source : http://www.croes.org/gerald/blog/qu-est-ce-que-rest/447/ [↑](#footnote-ref-22)
23. Source : http://www.jmdoudoux.fr/java/dej/chap-servlets.htm [↑](#footnote-ref-23)
24. Source : http://openclassrooms.com/courses/creez-votre-application-web-avec-java-ee/la-servlet [↑](#footnote-ref-24)