
Signature de l'encadrant : M. Nicolas DAGET



Signature de l'encadrante : Mme. Saloua ZAMMELI



Dédicaces

À ma chère mère, pour ses sacrifices démesurés et son amour infini. Que Dieu la garde afin que ses prières me protègent et que ses regards suivent ma destinée. J'espère pouvoir réaliser aujourd'hui l'un de ses rêves et être toujours à la hauteur de ses espérances.

À mon cher père, Qui n'a jamais cessé de me soutenir et m'encourager, Qui a impatiemment attendu ce jour, Aucun mot ne serait assez loquace pour témoigner les sentiments de reconnaissance que j'éprouve à son égard.

À mon cher mari, pour ses conseils, ses encouragements, sa confiance, son soutien, sa tolérance, sa compréhension et sa patience

A mon cher frère et mes chères soeurs, pour leurs soutien moral.

À mes chers amis, Pour l'amitié qui nous a liée.

Remerciements

C'est avec un grand plaisir que je consacre cette page à l'expression de ma sincère gratitude à tous ceux qui ont contribué à la réussite de ce Projet de Fin d'Etude.

Ma sincère gratitude à M.Nicolas DAGET, chef d'équipe ITIM-GTB, qui m'a offert l'opportunité de mener à bien mon projet de fin d'étude.

Un grand merci à Mlle. Saloua ZAMMELI, mon encadrante à la Faculté des Sciences de Tunis (FST), pour la confiance qu'elle m'a témoignée en acceptant de diriger ce projet. Je la remercie pour ses directives, ses conseils et sa disponibilité. C'est grâce à son support, son engagement et son enthousiasme que j'ai puachever ce travail.

Je tiens également à témoigner ma gratitude et mes vifs remerciements à notre architecte technique M. David MATTHEY pour sa grande disponibilité malgré ses responsabilités.

Je ne manquerais d'exprimer mes remerciements à toute l'équipe ITIM-GTB pour leur bonne humeur, leur professionnalisme, leurs soutien quotidien ainsi que d'avoir bien voulu partager leurs connaissances avec moi.

J'adresse mes remerciements à tous mes enseignants pendant ces trois agréables années au sein de la faculté des Sciences de Tunis pour la qualité d'enseignement qu'ils nous ont offert.

Enfin, mes remerciements les plus sincères vont aussi aux membres de Jury pour avoir accepté d'évaluer ce modeste travail et pour y avoir porté avis.

Table des matières

Table des figures	8
Liste des tableaux	10
1 Cadre du projet	13
1.1 Organisme d'accueil	13
1.1.1 La Société Générale : Qui sommes nous ?	13
1.1.2 Un modèle de banque universelle	14
1.1.3 Activité	15
1.1.4 Organisation de l'équipe	17
1.2 Présentation du projet	19
1.2.1 Introduction	19
1.2.2 Contexte du projet et étude de l'existent	19
1.2.3 Critique de l'existent	21
1.2.4 Solution proposée	22
1.2.5 Comparaison des deux versions	23
1.3 Méthodologie Appliquée	24
1.3.1 Choix de la méthodologie de travail	24
1.3.2 Solution proposée et objectifs	25
1.3.3 Équipe et rôles :	27
1.3.4 Conclusion	27
2 Phase de pré-jeu	28
2.1 Spécification des besoins	28
2.1.1 Besoins fonctionnels :	28
2.1.2 Besoins non fonctionnels :	29
2.2 Analyse des besoins	30
2.2.1 Identification des acteurs	30
2.2.2 Diagramme de cas d'utilisation général de l'application	31

2.3	Diagramme de déploiement	32
2.4	Backlog du produit	32
2.4.1	Priorités	35
2.4.2	Planification des sprints	35
2.4.3	Identification des principaux risque critique	36
2.4.3.1	La non maîtrise des outils et du langage de programmation	36
2.4.3.2	La contrainte du temps	36
2.5	Choix technique et Mise en oeuvre	36
2.5.1	Introduction	36
2.5.2	Choix technologiques	36
2.5.2.1	Choix de base de données	36
2.5.2.2	Choix technologiques de la couche service	37
2.5.2.3	Choix technologiques du client web	39
2.5.3	Choix de l'architecture	40
2.5.3.1	Architecture générale	40
2.5.3.2	Architecture de site	42
2.5.4	Mise en oeuvre	42
2.5.4.1	Introduction	42
2.5.4.2	Environnement de travail	43
2.5.4.3	Environnement matériel	43
2.5.4.4	Environnement logiciel et choix technique	43
2.5.4.5	Git :Logiciel de gestion de versions :	45
2.6	Conclusion	47
3	Sprint 1 : Menu authentification et menu Operations	48
3.1	Introduction	48
3.2	But du sprint	48
3.3	Menu Authentification	48
3.3.1	Authentification WEBCLEAR	48
3.3.2	Diagramme de séquence d'Authentification	50
3.3.3	Implémentation	52
3.4	Menu Opérations	56
3.4.1	Architecture du menu Opérations	56
3.4.2	Impléménatation	57
3.5	Conclusion	62

4 Sprint 2 : Menu Balance et menu Invoicing	63
4.1 Introduction	63
4.2 But du sprint	63
4.3 Menu Balance	63
4.3.1 Architecture du menu Balance	63
4.3.2 Implémentation	64
4.4 Menu Invoicing	64
5 Sprint 3 : Menu Orders et menu Documents	66
5.1 Introduction	66
5.2 But du sprint	66
5.3 Menu Orders	66
5.3.1 Architecture du menu Ordres	66
5.3.2 Périmètre	67
5.3.3 Fonctionnalités	68
5.3.4 Profils de connexion	68
5.3.5 Heure limite de signature des ordres	69
5.3.6 Implémentation	69
5.4 Menu Documents	71
5.4.1 Architecture du menu Documents	71
5.4.2 Implémentation	72
5.5 Conclusion	73
6 Sprint 4 : Menu Tools et menu Switching Bank	74
6.1 Introduction	74
6.2 But du sprint	74
6.3 Menu Tools	74
6.3.1 Architecture du menu Tools	74
6.4 Menu Switching Bank	75
6.4.1 Principes	75
6.4.2 La cinématique globale	76
6.4.3 Le périmètre	78
6.4.4 Les intervenants	79
6.4.5 Les échanges de flux entre les intervenants	79
6.4.6 Mise en oeuvre	80
6.4.6.1 Identification des fonctions	80

TABLE DES MATIÈRES

6.4.6.2	Description détaillée des fonctions	80
6.4.7	Implémentation	83
6.5	Conclusion	84
7	Conclusion générale et perspectives	85

Table des figures

1.1	Chiffres importants à savoir	14
1.2	Activités de la société générale durant l'année 2015-2016	17
1.3	Interface principale de webclear-ancienne version	20
1.4	Interface de Suivi - ancienne version	22
1.5	Etapes de Scrum	25
2.1	Diagramme de cas d'utilisation général	31
2.2	Diagramme de déploiement	32
2.3	les différents Sprints de notre application	35
2.4	Implémentation de services web REST avec le framework Spring	39
2.5	Architecture générale de l'application	41
2.6	Architecture de site	42
2.7	le mode de fonctionnement de Git	47
3.1	Authentification du client	49
3.2	Authentification du client	51
3.3	Interface d'authentification	53
3.4	Interface Home - mode desktop	54
3.5	Interface Home - mode tablette	55
3.6	Interface Home - version desktop	56
3.7	Architecture du menu Opérations	57
3.8	Interface de no data available	58
3.9	Interface de Search criteria	58
3.10	Interface de Entries effective	59
3.11	Interface de Plus de colonnes	60
3.12	Interface de Details	61
4.1	Architecture du menu Balance	63
4.2	Interface de Account statement	64

5.1	Architecture du menu Ordres	67
5.2	Interface de File Upload	70
5.3	Interface de Input	70
5.4	Interface de Tracking	71
5.5	Architecture du menu Documents	72
5.6	Interface du menu documents - mode desktop	72
5.7	Interface du menu documents - mode tablette	73
6.1	Architecture du menu Tools	75
6.2	Interface du menu documents - mode tablette	76
6.3	Les différentes fonctionnalités des modules	80
6.4	Cinématique de saisie du mandat	81
6.5	Cinématique de saisie d'opération	81
6.6	Cinématique de Consultation du mandat flux 2 reçu	82
6.7	Cinématique de Consultation du flux 3 reçu	82
6.8	Cinématique de Consultation du flux 4	83
6.9	Interface de la page de synthèse	83

Liste des tableaux

1.1	Comparaison entre l'ancienne et la nouvelle version	23
2.1	Backlog du produit de notre projet	34
2.2	Description du Weblogic [5]	38
2.3	les commandes de bases de git	46
3.1	Étapes de l'authentification	50
3.2	Description du diagramme de séquence "S'authentifier"	52
5.1	Exemple des différents profils	69

Introduction générale

LE secteur bancaire est aujourd’hui chamboulé par l’essor de la banque mobile et la dématérialisation de la relation client. Ainsi, confrontées aux nouveaux usages et à des clients de plus en plus connectés, les banques ont fait de la transformation digitale et le phénomène de disintermédiation un axe stratégique de développement.

Principale conséquence, aujourd’hui, quel que soit le service utilisé par un client, le faire à partir d’un mobile ou une tablette est devenu quasi systématique, le recours aux applications mobiles et les réseaux sociaux sont privilégiés pour interagir avec les clients si bien que plus de 80% des contacts entre un client et sa banque se font désormais sur des supports numériques (Source site Société Générale).

C'est dans ce contexte que s'inscrit notre projet de fin d'étude qui consiste en la refonte totale de l'application WEBCLEAR utilisée par les clients Corporate de la Société Générale pour la gestion de leurs opérations bancaires.

Afin de s'adapter aux nouveaux usages numériques et répondre ainsi aux nouvelles attentes des clients, la nouvelle version de WEBCLEAR sera conçue en Full Responsive Design et totalement axée sur l'expérience utilisateur avec une interface graphique intuitive, et cela tout en respectant un niveau élevé de sécurité, de fiabilité et de disponibilité de l'application.

Le présent rapport documente et explique la démarche que nous avons adoptée pour réaliser notre projet. Le rapport s'articule autour de six chapitres. Le premier chapitre introduit le contexte général du travail, à savoir l'organisme et le département d'accueil, la description de la problématique et le choix méthodologique. Le deuxième est consacré à une étude préalable et à la spécification des besoins fonctionnels et non fonctionnels du système et ses interactions avec son environnement il permet aussi de décrire les différentes

techniques utilisées lors du développement de notre application. Les quatre chapitres suivants développent la réalisation de chaque sprint.

Ils présentent la conception détaillée et décrivent l'aspect implémentation de chaque sprint à travers quelques interfaces de l'application résultante.

Enfin, la conclusion fait le bilan et présente quelques perspectives futures.

Chapitre 1

Cadre du projet

Introduction

Ce chapitre expose le cadre général et le contexte du projet, ainsi que les objectifs à atteindre. Nous présentons dans un premier lieu l'organisme d'accueil, puis nous décrivons la problématique ainsi que les principaux objectifs du projet.

1.1 Organisme d'accueil

1.1.1 La Société Générale : Qui sommes nous ?

La création de la Société Générale remonte au 4 mai 1864, date du décret d'autorisation signé par Napoléon III. A ce titre cette banque fait partie historiquement des trois grandes banques françaises avec le Crédit lyonnais et la BNP. Les premiers statuts reflètent d'emblée les ambitions de la banque. Au fil du temps la banque s'est beaucoup développée suivant les évolutions des métiers de la banque. La Société Générale est désormais une banque où la diversité des métiers est un défi permanent et doit accompagner les évolutions majeures d'un groupe en croissance constante sur l'ensemble de la planète et prendre en compte les problématiques propres à chacun de ses métiers, tout en étant porteuse de cohésion. Avec 31 millions de clients dans 66 pays, elle est l'une des plus importantes entreprises de services financiers en Europe. [1]

La figure 1.1 tirée du site intranet de la Société Générale montre le résumé des statistiques de l'année 2015 concernant la Société Générale :

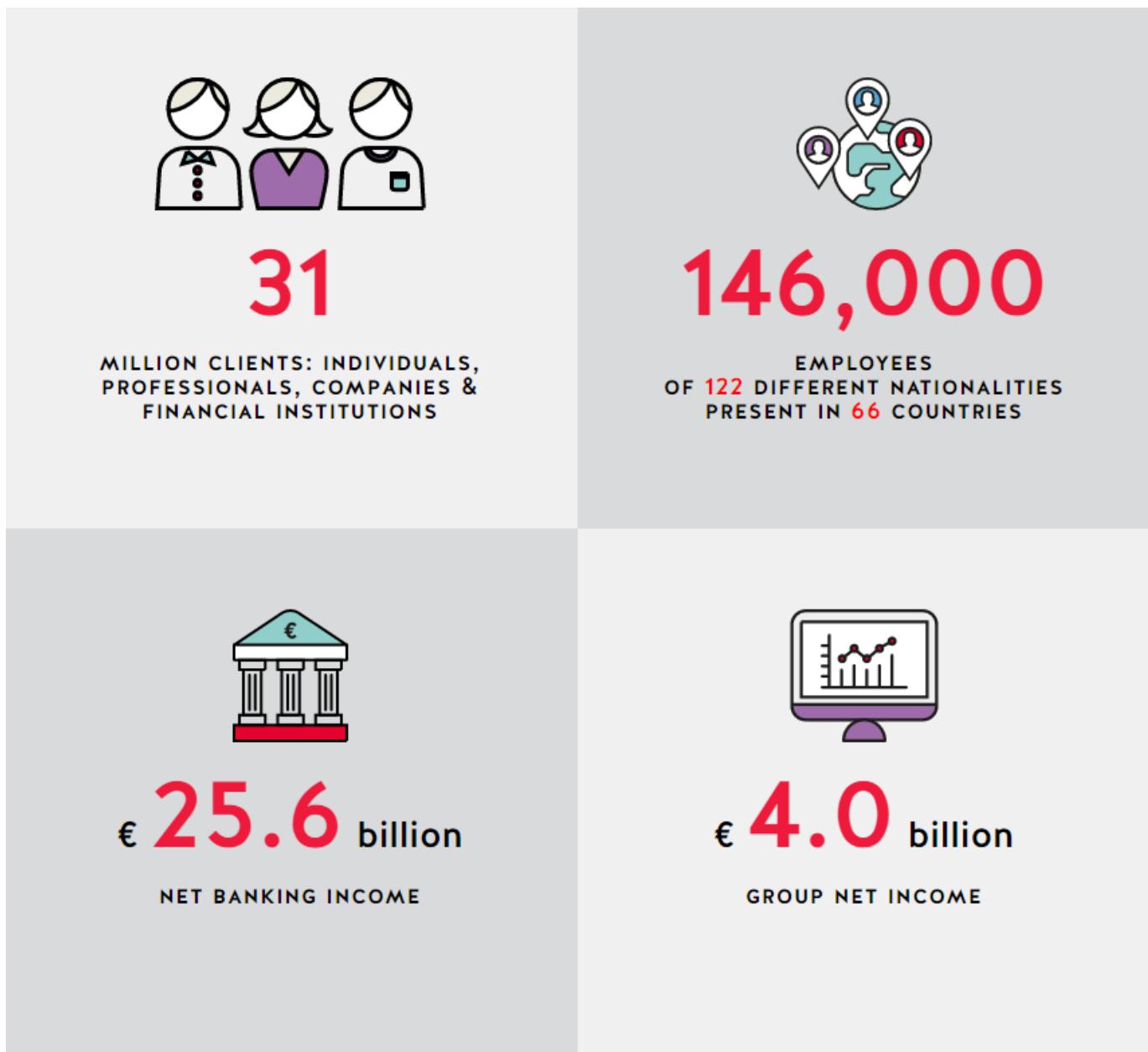


FIGURE 1.1 – Chiffres importants à savoir

1.1.2 Un modèle de banque universelle

Qu'ils soient particuliers, professionnels, entreprises ou investisseurs institutionnels, les clients de la Société Générale souhaitent avoir accès à une large gamme de services. Elle s'engage à les leur fournir à tout moment, quel que soit l'endroit où ils se trouvent. C'est pourquoi elle renforce son modèle de banque universelle, qui s'appuie sur trois pôles métiers complémentaires :

- **La banque de détail en France** avec les enseignes Société Générale, Crédit du Nord et Boursorama qui offrent des gammes complètes de services financiers avec une offre multicanal à la pointe de l'innovation digitale.
- **La banque de détail à l'international, services financiers et assurances** avec des réseaux présents dans les zones géographiques en développement et des métiers spécialisés leaders dans leurs marchés.
- **La banque de financement et d'investissement, banque privée, gestion d'actifs et métier titres** : Ce projet a été réalisé au sein de ce département qui, outre le développement, effectue les tests et la validation des applications.

1.1.3 Activité

La Société Générale propose une large gamme de produits, couvrant les besoins de ses clients particuliers, entreprises et institutions à travers le monde. Elle aide ses clients à mener à bien leurs projets personnels et professionnels, gérer les risques, faciliter le financement et l'investissement, et relever les défis qui renforcent l'économie. Ses clients, qu'ils soient des particuliers, des professionnels, des entreprises ou des investisseurs institutionnels, en France ou ailleurs, souhaitent avant tout disposer d'une large gamme de services. Les parcours proposés par la société générale sont :

- **La Banque de Détail en France** : En France, elle dispose de trois banques relationnelles complémentaires : Société Générale, Crédit du Nord et Boursorama. Ces trois réseaux partagent les meilleures pratiques, simplifient leurs processus et proposent des offres sur-mesure pour répondre aux besoins de leurs clients, tout en améliorant continuellement la qualité de leurs services.
- **Banque de Détail à l'international** : Présente dans 36 pays, avec 3 750 agences au total. Son objectif : conserver son modèle de banque universelle tout en continuant à respecter les spécificités des marchés locaux.
- **Services financiers spécialisés et assurances** : Crédit à la consommation, financement des ventes et des biens d'équipement, location longue durée et gestion de flottes de véhicules et enfin assurance : ces services sont accessibles aux particuliers et aux entreprises à travers ses banques de détail et Société Générale Corporate & Investment Banking.
- **Banque de Financement et d'investissement** : Société Générale Corporate & Investment Banking est un acteur essentiel dans le financement de l'économie à travers son rôle clé d'intermédiaire auprès de deux types de clients que sont les

émetteurs et les investisseurs.

- **Gestion d'actifs** : Lyxor Asset Management, filiale du groupe Société Générale, a été fondé en 1998. Avec 600 collaborateurs de 30 nationalités différentes, et présent dans 12 pays en Europe, en Amérique et en Asie Pacifique, Lyxor entretient un positionnement unique en tant que pionnier et leader dans ses domaines d'expertise.
- **Banque Privée** : Société Générale Private Banking offre des solutions de gestion de fortune à une clientèle d'entrepreneurs et de particuliers fortunés. Ses services comprennent la gestion de patrimoine, les services fiduciaires, une expertise globale dans les produits structurés, les fonds spéculatifs, les solutions d'investissement immobilier et l'accès aux marchés de capitaux.
- **Services aux investisseurs** : Société Générale Securities Services offre une gamme complète de services à la pointe des dernières évolutions des marchés financiers et de la réglementation : compensation, conservation, banque dépositaire, gestion des liquidités, administration de fonds et asset servicing, agent de transfert et services aux émetteurs.
- **Les Directions Groupe** : Les différentes Directions Groupe définissent l'orientation future et les politiques globales de Société Générale. Elles jouent un rôle de pilotage, d'animation et de conseil vis-à-vis des différents métiers du Groupe. Les 8 340 collaborateurs des Directions Groupe garantissent le professionnalisme et la cohérence de chacune des activités du groupe.

Par ses différents parcours, la Société Générale est donc présente dans différents domaines d'activités comme la montre la figure 1.2 tirée du site intranet de la Société Générale.

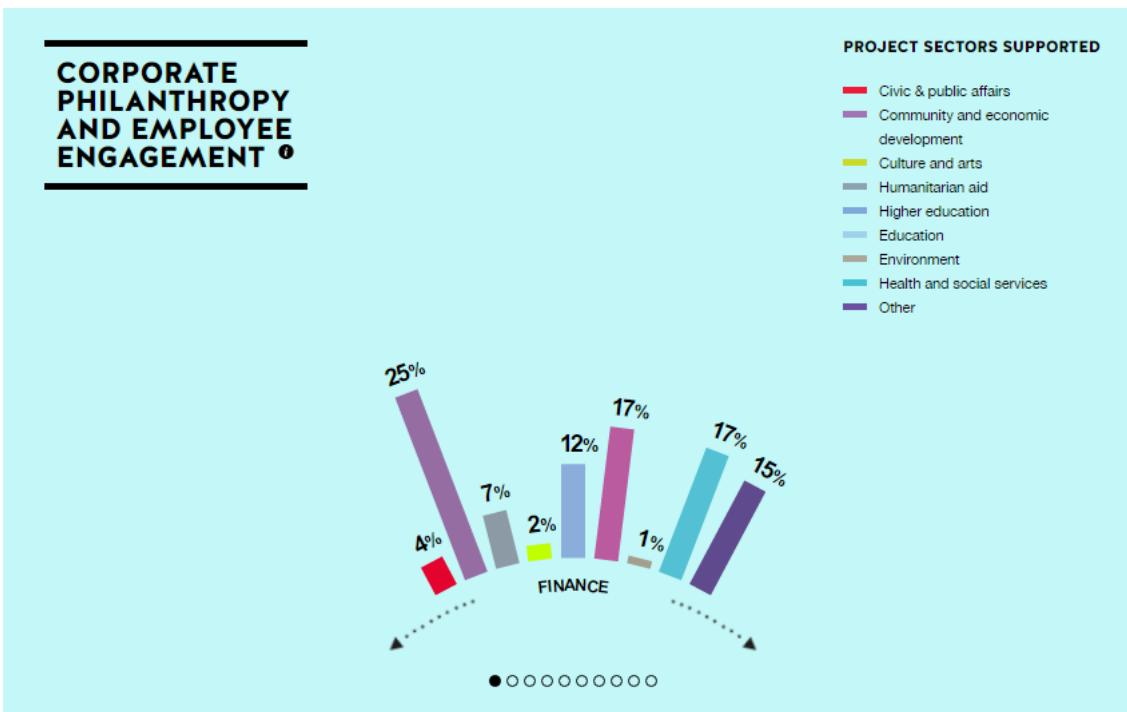


FIGURE 1.2 – Activités de la société générale durant l’année 2015-2016

1.1.4 Organisation de l’équipe

Chez la Société Générale, la palette de métiers SI s’organise autour de deux grands services aux utilisateurs :

- **La conception, le développement et la maintenance d’applications,**
- **La production et le support technique.**

Les différents rôles dans l’équipe sont :

- **Architecte des Systèmes d’Information** : Il garantit la bonne urbanisation du système d’information : architecture fonctionnelle et applicative, au service d’un SI agile et performant (optimisant le rapport coût informatique / valeur créée pour le métier). Il assure l’évolution de l’ensemble du Système d’information dans le respect des objectifs de l’entreprise, des processus métiers, des risques et des contraintes externes et internes, en exploitant au mieux les possibilités de l’état de l’art (technologie, existant, concepts d’architecture) ; il propose des scénarios d’évolution du SI en tenant compte des facteurs susceptibles d’avoir un impact sur celui-ci.
- **Architecte technique** : Il définit l’architecture technique du Système d’Information et assure l’évolution cohérente des composants techniques dont il a la charge en exploitant au mieux les possibilités de l’état de l’art. Il définit les normes et standards, propose les scénarios d’évolution et évalue la pertinence et la cohérence

des solutions par rapport à l'existant et à l'architecture cible.

- **Ingénieur études et développement** : Il analyse, réalise, assemble, valide techniquement et fonctionnellement et livre des composants logiciels dans le respect des besoins exprimés, des estimations de charges, des délais annoncés et des normes et procédures en vigueur. Il assure également la maintenance évolutive et corrective des composants logiciels existants.
- **Responsable de projet** : Il assure la responsabilité de la conception, de la planification et de la réalisation des livrables dans le respect des besoins exprimés, des budgets, des délais alloués et des critères de qualité, de sécurité et de performance exigés. Il anime et coordonne tout ou partie des acteurs du projet, assure la relation avec le client interne ou son représentant et les fournisseurs externes.
- **Ingénieur systèmes** : Il garantit le bon fonctionnement des socles techniques (matériels et logiciels) d'exploitation. Il assure la continuité des services des plates-formes techniques, contrôle l'optimisation des systèmes, les évolutions matérielles logiciels, participe aux études et conseille les ingénieurs de développement. Il intervient dans les diverses applications et techniques de pointe, traitant de gros volumes comme les portails, les applications de e-banking, les services mobiles et le Data-Warehouse.
- **Ingénieur réseaux et télécommunication** : Il définit les configurations, sélectionne les fournisseurs, garantit les installations et assure la maintenance des composants logiciels et matériels réseaux. Il garantit le bon fonctionnement, le bon dimensionnement, la sécurité et la flexibilité des solutions mises en oeuvre. Il assure le rôle de conseil, d'assistance et d'information, ainsi que la veille technologique. Il procède aux études techniques et participe au choix des solutions à mettre en oeuvre.
- **Expert en sécurité des SI** : Il participe à l'élaboration de la politique et des normes de sécurité des Systèmes d'Information et implémente celles-ci dans leur périmètre. Il est garant du bon fonctionnement des solutions mises en oeuvre et assure le rôle de conseil, d'assistance, d'alerte et d'information. Il réalise les études techniques, préconise les solutions à mettre en oeuvre et assure la veille technologique.
- **Maître d'ouvrage** : Il identifie, analyse et formalise les besoins des utilisateurs en matière de Systèmes d'Information sous la forme d'un cahier des charges. Il assure la responsabilité des livrables conformément au cahier des charges en ce qui concerne la qualité, les performances, le coût et le délai.
- **Responsable de la relation client** : Il est en charge d'accompagner et d'orienter

le client dans ses choix (sur l'offre de services : catalogue ou spécifique), de coordonner les changements et assurer le suivi des projets afin de garantir la cohérence du SI du groupe, la qualité et la sécurité des services pour les clients et le respect des engagements financiers.

1.2 Présentation du projet

1.2.1 Introduction

C'est dans le cadre de notre stage de Projet de Fin d'Etude que vient s'inscrire ce projet se déroulant au sein de la banque Société Générale. Il représente ainsi un complément de la formation universitaire acquise au sein de la Faculté de Science de Tunis. Il a été réalisé sous la direction de Monsieur Nicolas DAGET, chef du projet WEBCLEAR.

1.2.2 Contexte du projet et étude de l'existant

La Société Générale a plusieurs clients banques résidents et non résidents. Ces derniers accèdent à leurs comptes par le biais d'une application nommée WEBCLEAR qui leurs permet également de saisir ou de télécharger des virements commerciaux interbancaires.

Le projet, initié début 98, a nécessité la collaboration de la SSII Ifatec et de l'architecture technique de la Société Générale pour l'hébergement. Mis en service le 4 janvier 1999 parallèlement à la naissance de l'Euro, ce système est conçu comme un service à valeur ajoutée. [2]

Le service WEBCLEAR de la Société Générale met ainsi à la disposition des établissements financiers une tenue de leurs comptes en ligne. Par cette application, la Société Générale entend devenir la première banque compensatrice sur l'Euro en ligne.

Cette application web permet aux clients d'envoyer des ordres de paiement SWIFT d'une manière aussi efficace que s'ils étaient sur une station SWIFT.

Elle est compatible avec les types de messages MT103, MT200, MT202 et MT202COV. Ils peuvent initier des ordres de paiement en Euro mais également dans toute une gamme d'autres devises.

Différents niveaux d'habilitation sont disponibles :

- Un profil Saisie permet d'initier des ordres.

- Un profil Confirmation autorisera un collègue ou superviseur à vérifier et confirmer ces ordres.
- Un profil Consultation est également disponible pour visualiser les ordres validés ou en attente de validation.

D'autres fonctions sont incluses, comme la possibilité de sauvegarder un ordre pour le réutiliser comme modèle, de l'annuler ou encore de le renvoyer à son initiateur pour modification. Il est également possible d'exporter un ordre au format pdf (pour l'imprimer ou l'archiver, par exemple).

Les ordres sont vérifiés, filtrés et exécutés de manière strictement identique à tous les autres ordres qui utilisent le système Société Générale.

La figure 1.3 montre l'interface principale de l'ancienne version de l'application WEB-CLEAR.

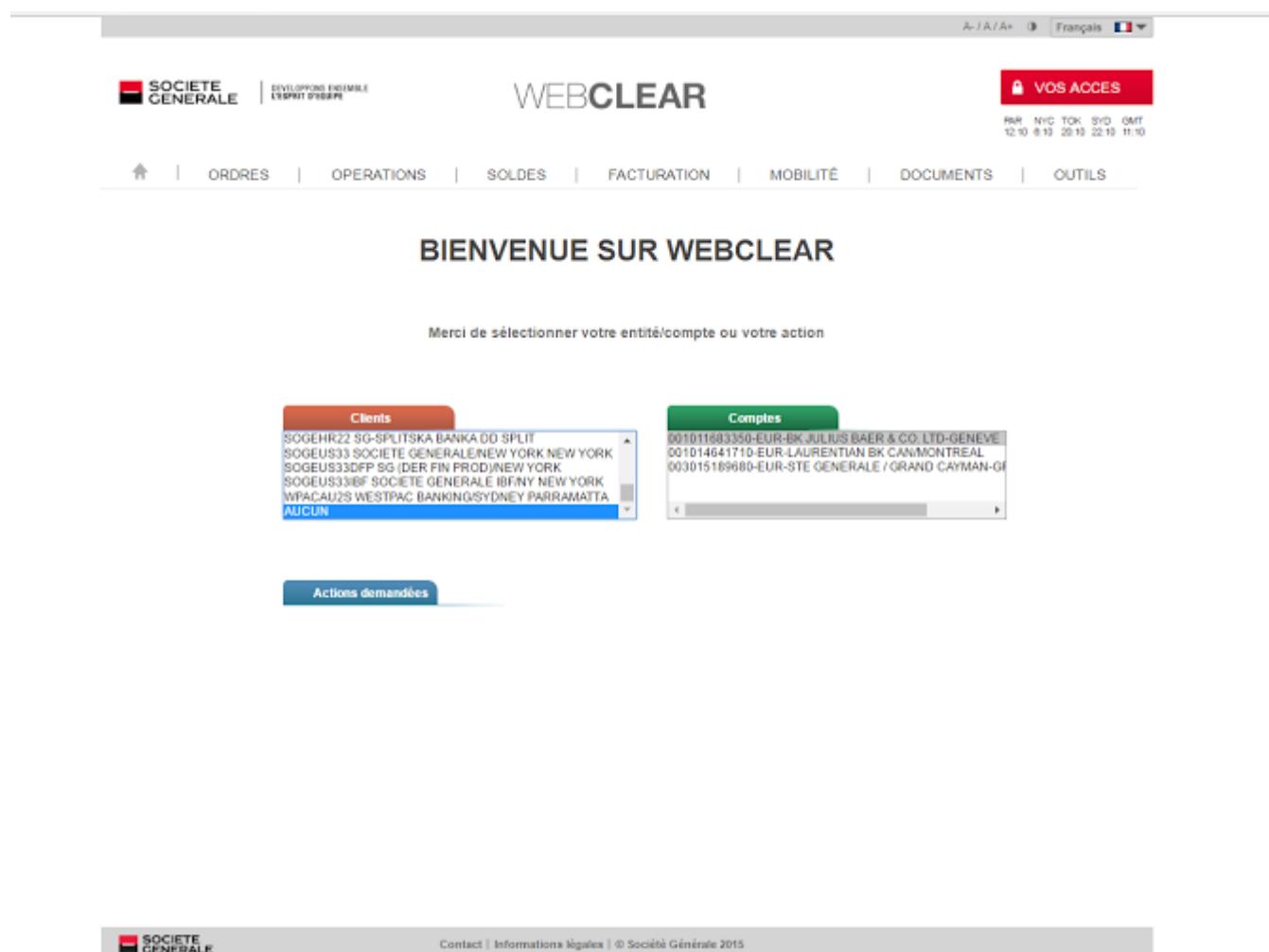


FIGURE 1.3 – Interface principale de webclear-ancienne version

1.2.3 Critique de l'existant

L'ancienne version de WEBCLEAR est une version web qui n'est pas compatible avec tous les terminaux, on ne peut l'utiliser que sur un ordinateur ce qui pose des soucis pour les utilisateurs de cette application surtout que l'utilisation des tablettes et smartphones domine. En plus, les technologies utilisées dans cette versions sont anciennes, ce qui peut poser des problèmes au futur si on veut la changer par ce qu'on ne va pas trouver les profils spécialisés qui maîtrisent ces anciennes technologies.

Aussi, WEBCLEAR n'est pas en Responsive Design et ne comporte pas la partie Mobilité Bancaire.

La figure 1.4 est une imprime écran d'une des interfaces de l'ancienne version, on peut remarquer qu'elle n'est pas responsive : Il y a plusieurs menus qui disparaissent dès qu'on passe à une résolution plus petite que celle du desktop, une partie du tableau ne s'affiche non plus.

Reference du virement	Date de règlement	Montant	Devise	Type d
WEB SG16700003534	2015-06-16	50.00	EUR	
WEB SG15900004644	2016-06-07	0.00	EUR	
WEB SG14800004601	2016-05-27	666.00	EUR	

FIGURE 1.4 – Interface de Suivi - ancienne version

1.2.4 Solution proposée

La nouvelle version de WEBCLEAR contient deux sous projets :

- **WebClear en mode responsive design** : Ce sous projet comporte les fonctionnalités de l'ancienne version avec les nouvelles technologies et en mode responsive design c'est à dire que nous avons fait la refonte totale et en marque blanche de la plateforme WEBCLEAR tout en mettant en place l'approche responsive design afin

- d'adapter l'application aux nouveaux usages et supports (Web, Mobile, Tablette)
- **La Mobilité Bancaire :** Le nouveau service de Mobilité Bancaire permet la prise en compte des nouvelles coordonnées bancaires par les émetteurs, de prélèvements et de virements dans un délai maximum de 22 jours ouvrés. Le service de Mobilité Bancaire est basé sur le principe de notification automatisée des organismes émetteurs (de prélèvement ou de virement) des changements de domiciliation bancaire. Étant donné les délais de prise en compte, ce service nécessitera des dispositifs automatisés de transmission d'information entre les PSP d'arrivée, les PSP de départ et les PSP des émetteurs.

Dans ce mode de fonctionnement, le client peut se reposer complètement sur son PSP d'arrivée sans avoir à lui fournir la liste des émetteurs et sa référence client chez ces émetteurs. Les PSP doivent informer leurs clients du nouveau service de Mobilité Bancaire. Le service de Mobilité Bancaire est proposé gratuitement et sans conditions par les PSP d'arrivée

1.2.5 Comparaison des deux versions

Le tableau 1.1 présente quelques points de différences entre l'ancienne et la nouvelle version

Ancienne version	Nouvelle version
Struts 1	-AngularJs -Bootstrap -Spring
Une instance de serveur pour faire back et front	- 1 serveur pour le front - 1 serveur pour le back
version non responsive	version en mode responsive design
Absence de la partie mobilité	Existence de la partie mobilité

TABLE 1.1 – Comparaison entre l'ancienne et la nouvelle version

1.3 Méthodologie Appliquée

Après avoir présenté le travail demandé, il était crucial de choisir une méthodologie adéquate en vue d'organiser le travail, d'assurer le bon déroulement de notre projet et de garantir le meilleur résultat.

1.3.1 Choix de la méthodologie de travail

Le choix de la méthode agile comme méthodologie de pilotage pour ce projet s'est basé sur les atouts de ce dernier. Ils se résument comme suit :

- Plus de souplesse et de réactivité.
- Grande capacité d'adaptation au changement grâce à des itérations courtes.
- Les besoins de notre projet sont bien définis dès le départ.
- La nature de cette méthode qui est bien organisée et permet de tester et valider chaque partie du projet avant de passer à une autre.

Nous avons choisi la méthodologie agile Scrum dans le cadre de notre mission en se basant sur plusieurs avantages. En effet, la méthodologie Scrum nous a amené à valoriser les individus et leurs interactions plus que les processus et les outils, des logiciels opérationnels plus qu'une documentation exhaustive, la collaboration avec les clients plus que la négociation contractuelle et enfin l'adaptation au changement plus que le suivi d'un plan. Cette valorisation a révélé plusieurs avantages tels que la simplicité des processus, la satisfaction des utilisateurs et l'amélioration de la communication entre le réalisateur du projet et le client. Nous justifions notre choix par le fait que les besoins de notre futur système, quoi que nombreux et définis dès le départ, risquent de changer. De plus, Scrum est la méthodologie référence utilisée au sein de la Société Générale pour conserver un maximum de contrôle sur la gestion du projet.

Afin de comprendre cette méthodologie, nous avons établi le tableau 1.2 pour définir les mots clés qui vont nous servir tout au long du projet.

La figure 1.5 donne une vue globale de Scrum[3] en utilisant les mots ci-dessus.

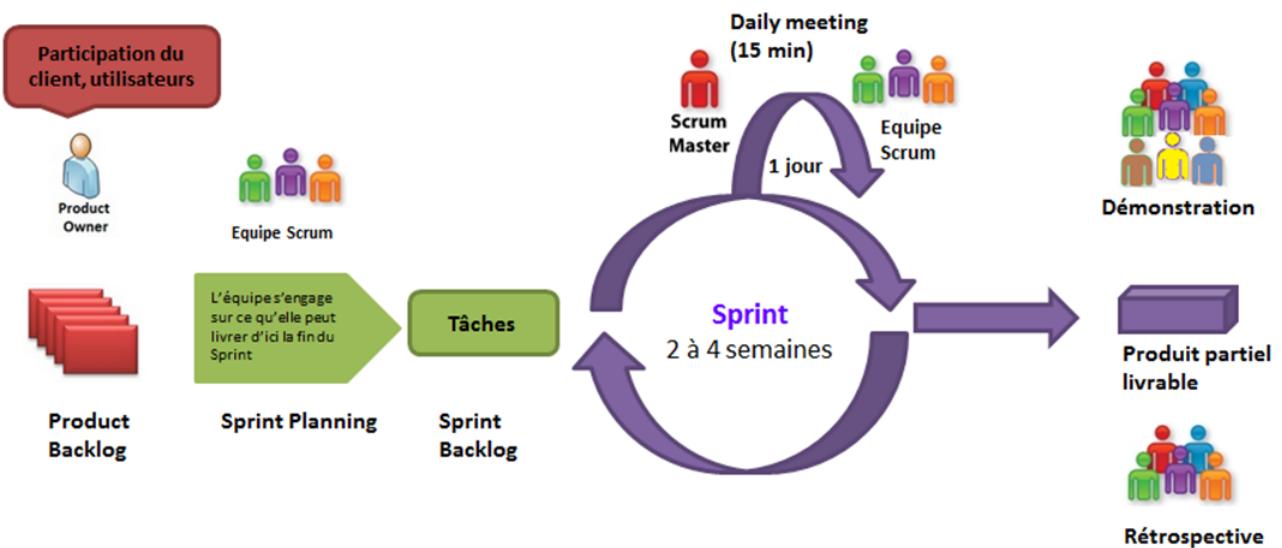


FIGURE 1.5 – Etapes de Scrum

1.3.2 Solution proposée et objectifs

Pour la réalisation du projet, nous avons opté pour la méthodologie agile Scrum en respectant plusieurs phases qui succèdent à la mise en place du Backlog :

Phase 1 : Le "Quoi" Une fois que le Product Backlog est suffisamment complet et ordonné, on peut planifier un sprint. Le Product Owner revoit alors avec l'équipe de développement la vision du produit, la roadmap, le plan de livraison (jalons et deadline), l'objectif du sprint et le Product Backlog. L'équipe de développement vérifie les estimations, confirme qu'elles sont exactes et sélectionne en haut du Product Backlog les exigences qu'elle se sent capable de convertir en fonctionnalités utilisables d'ici la fin du sprint (il s'agit d'une prévision et non pas d'un engagement).

Phase 2 : Le "Comment" L'équipe de développement fait ensuite l'inventaire des tâches qui permettront de convertir les exigences sélectionnées en fonctionnalités utilisables d'ici la fin du sprint. Toutes les exigences n'ont pas nécessairement besoin d'être découpées en tâches. En cas de manque de temps, l'équipe de développement peut se contenter de découper celles qui seront réalisées au cours des premiers jours du sprint (elle découpera en cours de sprint les autres exigences). Elle doit cependant aller

suffisamment loin dans l'effort de conception pour pouvoir vérifier sa prévision. Si elle constate après analyse des exigences sélectionnées, que sa prévision est erronée, elle peut réajuster avec le Product Owner la liste des exigences sélectionnées.

Les tâches de développement sont centralisées dans le Sprint Backlog et ajoutées au tableau des tâches physique. L'idéal est de parvenir à un découpage relativement fin (inférieur ou égal à une journée de travail). Chacun peut personnaliser les colonnes de son tableau des tâches, ou code couleur des post-it.

Sprint (2 à 4 semaines) Au cours du Sprint, l'équipe se concentre sur l'accomplissement des tâches du Sprint Backlog. En cas de retard, des exigences ou tâches seront retirées du Sprint Backlog en cours de route en essayant de préserver l'objectif du sprint (pour cela, il est conseillé d'ordonnancer les exigences au sein du sprint). Et inversement, si l'équipe avance plus vite que prévu, des exigences ou tâches y seront ajoutées. En accord avec le Product Owner dans les deux cas. Les développeurs doivent éviter de trop paralléliser les exigences et encore moins les tâches de développement.

Le tableau des tâches physique rempli de post-it est pratiquement indispensable. Il permet d'avoir une vision claire du travail à accomplir, en cours et terminé. Il peut également s'avérer précieux lors des réunions quotidiennes, Le tableau facilite aussi l'affectation des tâches par l'équipe en ayant une vision d'ensemble du sprint en un coup d'œil.

Mêlée quotidienne ou "stand-up meeting " Durée maximum : 15 minutes.

Chaque personne répond à 3 questions :

- Qu'ai-je fait hier qui a aidé l'équipe de développement à atteindre l'objectif Sprint ?
- Que vais-je faire aujourd'hui pour aider l'équipe de développement à atteindre l'objectif Sprint ?
- Est ce que je vois des obstacles susceptibles de m'empêcher ou d'empêcher l'équipe de développement d'atteindre l'objectif du Sprint ?

Revue de Sprint : Durée maximum : 1 heure par semaine de sprint (autrement dit : 2 heures pour des sprints de 2 semaines).

L'objectif de la revue de sprint est d'inspecter l'incrément produit au cours du sprint écoulé, faire un point sur l'avancement de la Release et adapter au besoin le plan et le Product Backlog.

La livraison à la fin de chaque sprint n'est pas obligatoire.

Rétrospective de sprint Durée maximum : 45 minutes par semaine de sprint (autrement dit : 1 heure 30 pour des sprints de 2 semaines).

Cette réunion est généralement animée par le ScrumMaster qui s'adresse à son équipe. Elle a pour but d'améliorer continuellement le processus de développement de l'équipe en mettant les idées de chacun à contribution.

1.3.3 Équipe et rôles :

L'équipe a un rôle capital dans la méthodologie Scrum, elle est constituée avec le but d'optimiser la flexibilité et la productivité, Scrum définit ainsi trois rôles qui sont :

- **Le Product Owner** : C'est la personne qui prépare la liste d'exigences fonctionnelles et non fonctionnelles du produit à réaliser, généralement c'est un expert dans le domaine.
- **Le Scrum Master** : C'est la personne qui doit assurer le bon déroulement des différents sprints et qui doit impérativement maîtriser Scrum.
- **Le Scrum Team** : Constitué des MOA qui seront chargés d'implémenter les différents besoins du clients sous forme de spécifications et de MOE qui sont les développeurs qui seront chargés de codes ces besoins.

Dans le contexte de notre projet, M. Philippe LE MOULEC sera le product owner, M. David MATTHEY sera le scrum master, M. Thierry Guedé, M. Vincent FONTAINE et Mme. Françoise GHADA seront les MOA, ils forment le scrum team avec moi et 6 autres développeurs.

1.3.4 Conclusion

DANS ce chapitre, nous avons passé en revue les différentes notions nécessaires pour mieux comprendre le sujet telles que la présentation de l'organisme d'accueil et la méthodologie à suivre. Aussi, la compréhension du contexte général de ce projet et l'étude de l'existant qu'ont été essentielles pour bien exprimer la problématique du travail et en tirer les objectifs à réaliser. Une étude des besoins fera l'objet du prochain chapitre.

Chapitre 2

Phase de pré-jeu

Introduction

La réussite de tout projet dépend en grande partie de la qualité de son départ. De ce fait, les étapes d'analyse et de spécification constituent la base de départ du travail. En effet, elle assure l'adéquation de toute l'application à réaliser aux besoins des utilisateurs et aux traitements envisagés au niveau de ses opérations ainsi que son utilité future. Pour assurer ces objectifs, il est essentiel de parvenir à une vue claire des différents besoins escomptés du projet.

Dans ce chapitre, nous explicitons les besoins fonctionnels, non fonctionnels et techniques de notre système, et nous effectuons une spécification formelle des besoins, présentée par des diagrammes des cas d'utilisation.

2.1 Spécification des besoins

L'analyse de la problématique permet de dégager les fonctionnalités qui sont mises à la disposition de l'utilisateur. Ces besoins sont classés en besoins fonctionnels et non fonctionnels.

2.1.1 Besoins fonctionnels :

”Un besoin fonctionnel est un besoin spécifiant une action qu'un système doit être capable d'effectuer, sans considérer aucune contrainte physique : besoin spécifiant un comportement d'entrée/sortie d'un système.”[Jacobson 1996].

L'outil Internet WEBCLEAR doit assurer les fonctionnalités suivantes :

- Authentification : permet l'identification de l'utilisateur en un mode précis selon les droits qui lui sont accordés.
- La saisie d'un ordre de virement.
- La signature d'un ordre de virement.
- Le suivi des ordres de virements initiés à l'aide du web.
- La sauvegarde d'un ordre de virement.
- Le renvoi d'un ordre à un utilisateur de profil saisie afin de modifier.
- L'annulation d'un ordre
- L'export au format pdf d'un ordre de virement pour impression et/ou archivage.
- Écriture et lecture des opérations bancaires.
- Avoir la documentation nécessaire.
- L'accès aux relevés de comptes
- La notification
- Proposer à un client de gérer sa mobilité bancaire au sein de WEBCLEAR
- Pouvoir utiliser l'application durant les déplacements : Adapter la visualisation sur tablette et différents devises

2.1.2 Besoins non fonctionnels :

”Un besoin non fonctionnel est un besoin spécifiant des propriétés du système, telles que les contraintes liées à l'environnement et l'implémentation, les exigences en matière de performance, de dépendances de plate-forme, de facilité de maintenance, d'extensibilité et de fiabilité.”[Jacobson1996]

Pour assurer la qualité de l'application, Elle doit garantir l'ensemble de besoins non fonctionnels suivants :

- **Modularité** : Le système doit être conçu et développé dans une architecture modulaire afin d'améliorer la performance, faciliter la maintenance et répondre à l'objectif principal du projet.
- **Utilisabilité** : Le temps de réponse et le chargement des données doivent être minimaux. La navigation d'un écran à un autre ne doit pas gêner l'utilisateur.
- **Fiabilité** : Le système doit être disponible à tout moment pour l'utilisateur.
- **Simplicité** : Le système doit présenter une interface compréhensible, facile à manipuler permettant d'accroître la rentabilité et l'efficacité du système.
- **Extensibilité** : Le système doit permettre d'étendre ses capacités en ajoutant des modules ou mises à jour.

- **Evolutivité** : Le systèmes doit garantir la capacité d'un service ou d'une application à évoluer afin de satisfaire aux exigences de performance croissantes.

2.2 Analyse des besoins

Après avoir déterminé les besoins fonctionnels et les différentes contraintes qui doivent être considérés, nous allons les présenter dans un diagramme de cas d'utilisation général. Pour chaque cas d'utilisation qui représente les interactions entre les acteurs et le système, il s'agit de faire une description textuelle. L'intérêt de cette étape est d'étudier l'interaction entre les acteurs et le système.

2.2.1 Identification des acteurs

Un acteur représente un ensemble de rôles joués par un utilisateur en interaction avec notre application. Il interagit avec notre système pour avoir un résultat, il peut être un homme, une machine ou un autre système. Dans ce contexte, nous identifions les acteurs suivants :

► **les acteurs fonctionnels :**

- **les clients bancaires** : qui sont les clients banques de la Société Générale résidents et non résidents.

pour la partie Mobilité Bancaire, on a trois types de clients bancaires :

- **La banque d'arrivée** : Adresse à l'établissement de départ un message par lequel il lui demande de lui envoyer les informations relatives aux opérations concernées sur les 13 derniers mois de fonctionnement du compte.
- **La banque de départ** : Recense les opérations concernées et adresse à l'établissement d'arrivée un ou plusieurs messages comprenant les informations demandées.
- **La banque d'émetteur** : Route les modifications de domiciliation bancaire à l'émetteur

- **Les gestionnaire de WEBCLEAR** : Ils administrent le site WEBCLEAR.

► **Les acteurs du projet :**

- **ITIM-GTB** : C'est l'équipe qui a développé WEBCLEAR.
- **GTPS** : C'est le métier ; il fournit les besoins.

- **Exploitants** : Ils maintiennent la production.
 - **L'homologueur** : Il a pour rôle de tester l'application.
- **Les acteurs techniques du site :**
- **ClearVision** : C'est le moteur du Batch qui alimente la base de données.
 - **L'équipe Portail** : Elle attribue les droits d'accès, génère le clavier virtuel, gère le système de signature d'ordre et de séquences, Elle gère aussi l'authentification.
 - **Le Socle** : Pour la partie mobilité, il joue un rôle d'intermédiaire entre les banques d'arrivée, de départ et d'émetteur.
- **L'équipe réseau** : Vérifie les firewall applicatifs et maintient l'infrastructure

2.2.2 Diagramme de cas d'utilisation général de l'application

Un cas d'utilisation décrit un ensemble de scénarios, y compris des variantes, qu'un système exécute pour produire un résultat tangible pour un acteur. Il définit ainsi les activités attendues des différents utilisateurs par rapport à l'application. Ainsi, les différentes fonctionnalités décrites dans la section précédente sont regroupées par le diagramme de cas d'utilisation général donné par la figure 2.1.

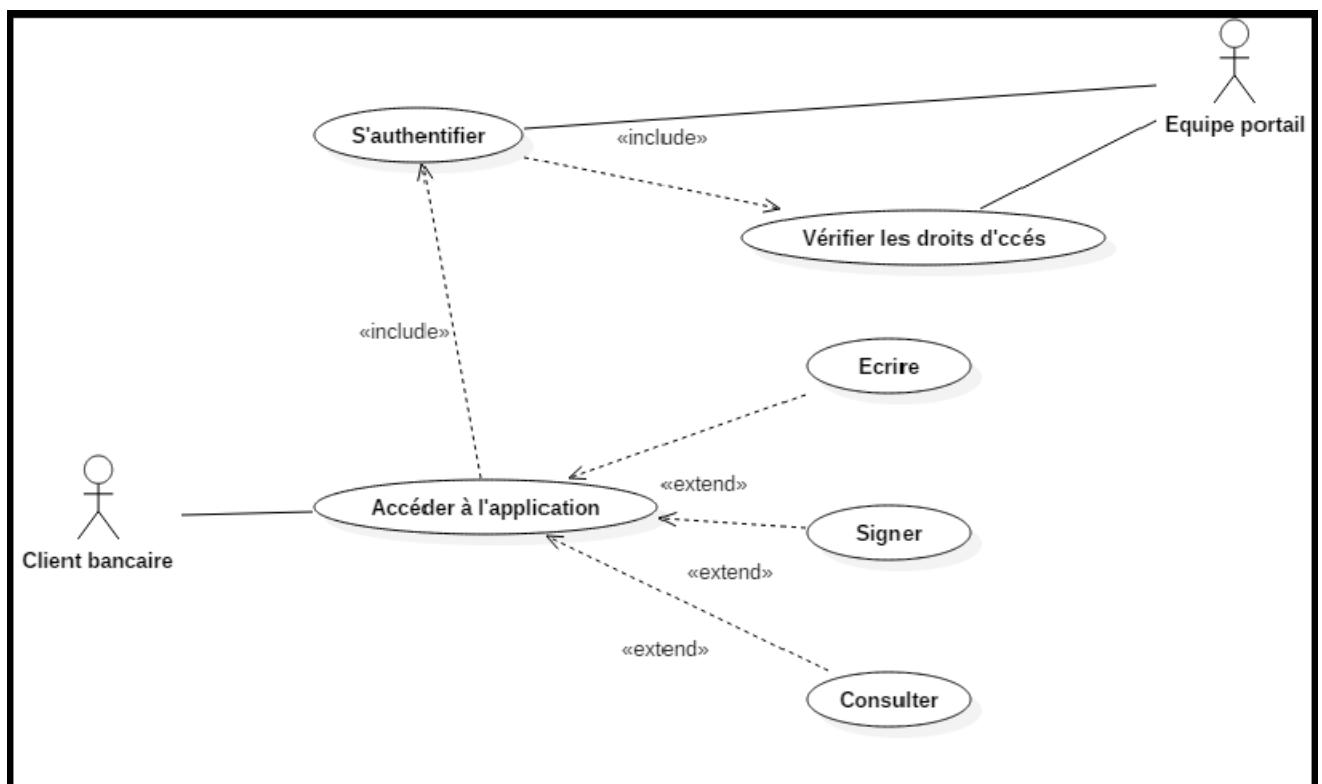


FIGURE 2.1 – Diagramme de cas d'utilisation général

2.3 Diagramme de déploiement

Le diagramme de déploiement décrit la disposition physique des matériels qui composent un système donné. Cette description aide à cerner les moyens matériels mis en place, et informe le développeur sur la solution technique à adopter. Elle est le reflet de l'architecture utilisée par l'application. La figure 2.2 représente le diagramme de déploiement de notre projet qui est valable pour tous les sprints, contenant les matériels qui interviennent lors du fonctionnement de notre solution logicielle.

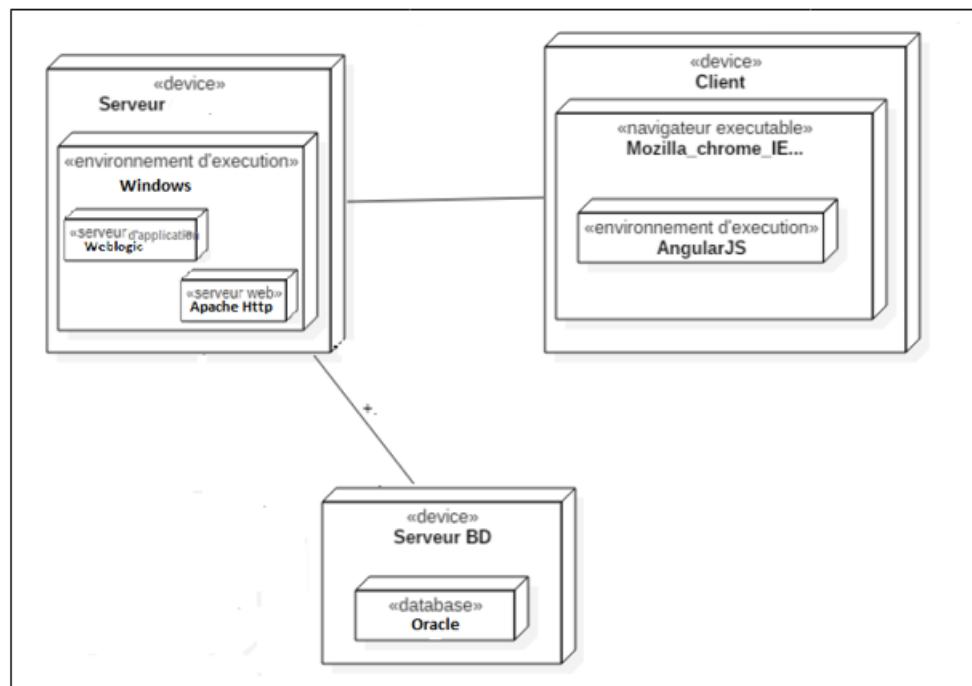


FIGURE 2.2 – Diagramme de déploiement

- Le serveur d'application : C'est la machine sur laquelle s'exécute l'environnement java, cette machine communique avec le serveur de base de données et la machine client.
- Serveur de BD : C'est la machine sur laquelle est installé le SGBD et contenant les données des utilisateurs (Authentification LDAP, etc)
- Client : C'est la machine qui exécute l'environnement FrontEnd Angularjs qui présente la gestion de vue de l'application.

2.4 Backlog du produit

Les équipes agiles collectent les fonctions essentielles et les raffinent à travers un outil de collecte qui s'appelle le backlog de produit.

Le backlog du produit est l'artefact le plus important de Scrum, c'est l'ensemble des caractéristiques fonctionnelles ou techniques qui constituent le produit souhaité.

Le Tableau 2.1 résume le backlog du produit de notre application.

Dans ce tableau chaque user story est caractérisée par ses risques et sa priorité. Pour le traitement des user story nous choisissons de commencer avec les cas d'utilisation les plus prioritaires et ayant le risque le moins élevé. Chaque histoire utilisateur possède un effort (vitesse) qui est l'estimation initiale sur la quantité de travail nécessaire pour implémenter cette exigence. Cet effort est calculé en point d'histoire qui correspond aux jours hommes idéaux.

ID Tâche	User story	Nom Tâche	Risque	Priorité
1	En tant que développeur, avant de commencer le développement je dois avoir les spécifications techniques nécessaires	Rédactions des spécifications techniques	Élevé	Élevé
2	En tant qu'utilisateur, je dois pouvoir me connecter via l'application.	s'authentifier	Élevé	Élevé
3	En tant qu'utilisateur, je dois pouvoir saisir un ordre de virement.	La saisie d'un ordre de virement	Élevé	Élevé
4	En tant qu'utilisateur, je dois pouvoir signer un ordre de virement.	La signature d'un ordre de virement	Élevé	Élevé
5	En tant qu'utilisateur, je dois pouvoir sauvegarder un ordre de virement.	La sauvegarde d'un ordre de virement	Élevé	Élevé
6	En tant qu'utilisateur, je dois pouvoir renvoyer un ordre à un utilisateur de profil saisie afin de modifier.	Le renvoi d'un ordre	Élevé	Élevé
7	En tant qu'utilisateur, je dois pouvoir annuler un ordre.	L'annulation d'un ordre	Élevé	Élevé

8	En tant qu'utilisateur, je dois pouvoir exporter au format pdf un ordre de virement pour impression et/ou archivage.	L'export d'un ordre de virement .	faible	Élevé
9	En tant qu'utilisateur, je dois pouvoir Accéder à mes opérations bancaires annulées.	Accès aux opérations bancaires annulées	moyen	moyen
10	En tant qu'utilisateur, je dois pouvoir Accéder à mes opérations bancaires effectuées.	Accès aux opérations bancaires effectuées	moyen	moyen
11	En tant qu'utilisateur, je dois pouvoir Accéder aux détails de mes opérations bancaires annulées.	Détails des opérations bancaires annulées	moyen	moyen
12	En tant qu'utilisateur, je dois pouvoir Accéder aux détails de mes opérations bancaires effectuées.	Détails des opérations bancaires effectuées	moyen	moyen
13	En tant qu'utilisateur, je dois pouvoir Exporter au format excel mes opérations bancaires	Export des opérations bancaires	moyen	moyen
14	En tant qu'utilisateur, je dois pouvoir Accéder aux relevés de comptes.	Relevés de comptes	moyen	moyen
15	En tant qu'utilisateur, je dois pouvoir Accéder aux soldes en valeurs.	Soldes en valeurs	moyen	Élevé
16	En tant qu'utilisateur, je dois notifier si il y a des changements ou des perturbations .	Notifier	moyen	moyen
17	En tant qu'utilisateur, je dois avoir un guide	Guide	faible	faible
18	En tant qu'utilisateur, je dois avoir un glossaire	Glossaire	faible	faible
19	En tant qu'utilisateur, je dois pouvoir faire un transfert automatisé de mes domiciliations bancaires ainsi que des opérations.	La mobilité bancaire	Élevé	Élevé

TABLE 2.1 – Backlog du produit de notre projet

2.4.1 Priorités

Généralement, un cas d'utilisation A est plus prioritaire que B, si sa réalisation accélère la stabilisation du système. Le choix des priorités dans cette section s'est basé sur la dépendance entre les fonctionnalités de l'application. Nous avons classé donc l'authentification via l'application et la fonctionnalité "ajout" pour chaque module en priorité haute. Par la suite la fonctionnalité "modifier" pour chaque module en priorité moyenne, et finalement la fonctionnalité "supprimer" pour chaque module en priorité faible.

2.4.2 Planification des sprints

La réunion de planification des Sprints est l'événement le plus important dans Scrum. Le but de cette réunion est de préparer le planning de travail et d'identifier le backlog des Sprints. L'un des produits de cette réunion est basé sur la durée des Sprints qui diffère selon la complexité des tâches, la taille de l'équipe et qui ne doit pas dépasser 30 jours. La figure 2.3 représente les différents Sprints de notre application.

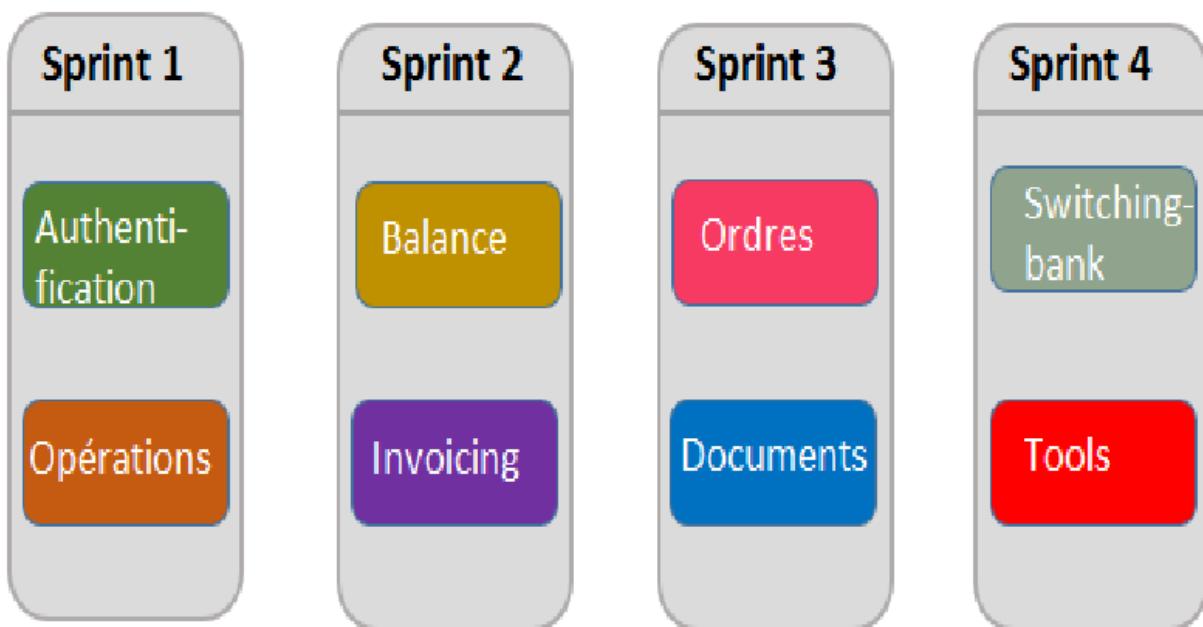


FIGURE 2.3 – les différents Sprints de notre application

Le premier sprint est dédié pour faire la page d'authentification et le menu Opérations. Le menu Balance et le menu Invoicing ont été réalisé dans le deuxième sprint. Dans le troisième sprint nous avons réalisé les menus Ordres et Documents et enfin nous avons le menu tools et on a passé à la partie mobilité bancaire dans le dernier sprint.

2.4.3 Identification des principaux risque critique

2.4.3.1 La non maîtrise des outils et du langage de programmation

Notre expérience étant limitée en développement orienté web, nous essayerons d'apprendre ce mode de développement au fur et à mesure que nous construisons notre application.

2.4.3.2 La contrainte du temps

La contrainte Temps est un risque à tenir en considération car l'application à concevoir et à développer est assez dense en terme des besoins à identifier et à modéliser.

2.5 Choix technique et Mise en oeuvre

2.5.1 Introduction

Cette section chapitre est décomposé en deux parties importantes. La première partie est consacrée au choix de la solution technique à adopter. Pour ce faire, nous allons justifier les différents choix technologiques effectués. Dans la deuxième partie, nous commençons par détailler l'étape de conception de notre système, ensuite, nous présentons l'environnement matériel et logiciel sur lesquels nous avons réalisé notre travail. Enfin, nous donnerons un aperçu sur les modules réalisés.

2.5.2 Choix technologiques

L'étude fonctionnelle déjà traitée, consistait à dégager les exigences purement fonctionnelles, tout en étant abstrait du niveau technologique. Cette section est dédié à l'étude technique du projet qui a été menée en parallèle à l'étude fonctionnelle pour rester fidèle au cycle de développement adopté. En effet, elle consiste à capturer les exigences techniques du projet.

2.5.2.1 Choix de base de données

Comme beaucoup de sites dynamiques, celui-ci a besoin d'une base de données, et comme il s'agit d'une refonte de l'application et pour des raisons de temps et de financement on n'a pas changer la base de données qui est Oracle 11g [4].

En effet la base de données Oracle a plusieurs avantages parmi lesquels :

- Richesse fonctionnelle

- Fonction d'audit évolué
- Row level storage security (RLSS) : permet de ne faire apparaître que certaines lignes des tables pour un utilisateur/une application donné.
- Intégration LDAP, SSL, Unicode ; réPLICATION intégrée ; capable de mapper un fichier plat en table
- Parallélisme, caches nommés ; haute disponibilité ; grande possibilité de tuning
- Procédures stockés en PL-Sql (langage propriétaire Oracle) ou ... en JAVA (depuis la 8.1.7) ce qui peut s'avérer utile pour les équipes de développement.
- Assistants performants via Oracle Manager Server, possibilité de gérer en interne des tâches et des alarmes
- Gestion centralisée de plusieurs instances
- Concept unique de retour arrière (Flashback)
- Pérennité de l'éditeur : avec plus de 40% de part de marché, ce n'est pas demain qu'Oracle disparaîtra
- Réglages fins : dans la mesure où l'on connaît suffisamment le moteur, presque TOUT est paramétrable.
- Accès aux données système via des vues, bien plus aisément manipulable que des procédures stockées.
- Interface utilisateur remaniée et extrêmement riche, permettant - enfin ! - le tuning fin de requêtes par modification des plans d'exécution.
- Architecture Multi-Générationnelle (MGA)
- Services Web, support XML
- Ordonnanceur intégré
- Compression des données et des sauvegardes
- Support technique Orion extrêmement riche et fourni

2.5.2.2 Choix technologiques de la couche service

La couche service de notre application ou bien le Back-end comme elle est nommée chez les développeurs, c'est la logique de l'application. Nous allons présenter les technologies utilisées afin d'implémenter cette couche.

Weblogic :

Dans une architecture distribuée JEE, le rôle d'un serveur d'applications est fondamental : il doit assurer la gestion des services d'infrastructure comme le cycle de vie des composants métier et les transactions entre composants. L'administration d'un tel

serveur ne doit pas être sous-estimée car cela impacte directement les performances des applications. Ci dessous un tableau descriptif de Weblogic.

Fonction	Brique	Commentaire
Technologie	J2EE 1.3 (Java 2 Enterprise Edition)	J2EE couvre une dizaine de services techniques (accès à un annuaire, une base de données, etc.).
Langages supportés	Java	WebLogic Applications Server se limite aux environnements de développement Java (IBM, Borland, etc.).
Composants	EJB (Enterprise Java Bean).	Les composants métiers transportent la logique applicative, qu'ils encapsulent.
Moteur de script	JSP (Java Server Pages) et Servlet.	JSP et Servlet sont destinés aux applications Web.
Intégration applicative	JDBC, JCA et les Web Services.	Ces éléments constituent des infrastructures pour se connecter aux applications tierces.
Intégration BtoB	-	les langages ebXML, RosettaNet et BPML sont disponibles dans WebLogic Integrator.
Systèmes d'exploitation	Solaris, Windows NT, Linux, IBM AIX, HP-UX.	Ce panel couvre ainsi l'ensemble des principaux systèmes d'exploitation.
Sécurité	Equilibrage de charge entre serveurs et failover.	Entendez par failover le routage automatique des flux vers un serveur d'applications miroir en cas de panne.

TABLE 2.2 – Description du Weblogic [5]

Spring IoC :

Spring est un framework libre de développement d'application java. Il s'agit d'un framework dit "léger" d'applications car il permet au développeur de créer des serveurs d'application (JEE) sans se soucier des couches basses.

Le fonctionnement de Spring IoC repose sur 3 éléments clés :

- La configuration XML pour lier les implémentations à leurs interfaces.
- Les classes sont des beans et possèdent donc des getters et setters pour les champs à injecter.

- L'injection en même est effectuée par un ApplicationContext de Spring

Spring MVC :

REST (Representational State Transfer) ou RESTful est un style d'architecture reposant sur l'utilisation du protocole HTTP en tirant partie de son enveloppe et de ses en-tête sans ajout de surcouche. Ce paradigme d'architecture se veut parfaitement stateless et laisse donc le soin au client de gérer les sessions.

Dans ce projet, nous avons implémenter des services web de type REST avec le framework Spring. Le support de REST a été introduit à partir de la version 3 de Spring et il est basé sur Spring MVC.

Ceci est expliqué par la figure 2.4 [6]

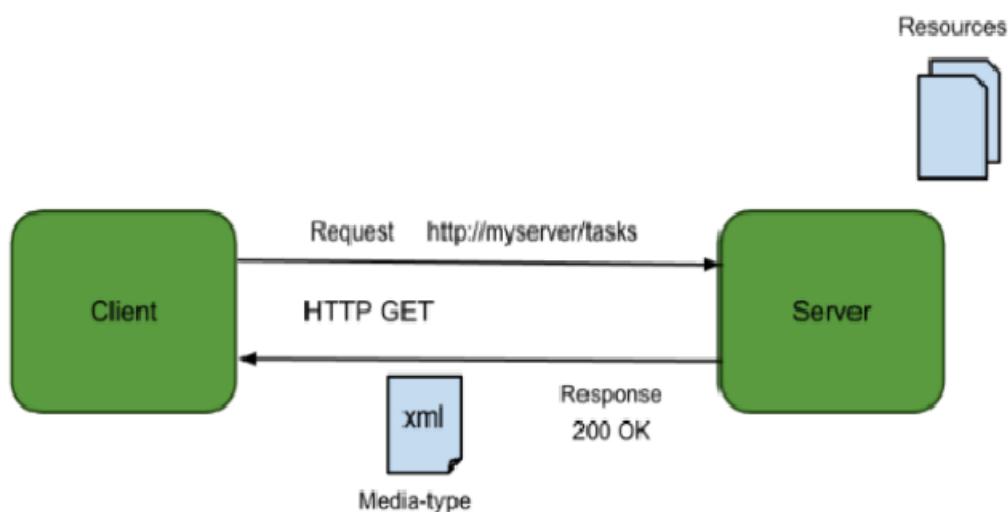


FIGURE 2.4 – Implémentation de services web REST avec le framework Spring

2.5.2.3 Choix technologiques du client web

Notre client web ou bien le Front-end comme il est nommé chez les développeurs, c'est la partie visible de l'application. Nous allons présenter les différentes technologies utilisées afin d'implémenter ce client.

AngularJS :

AngularJS est un Framework d'application web open-source principalement maintenu par Google et par une communauté d'individus et des entreprises afin de répondre à un bon nombre de difficultés rencontrées dans le développement d'applications. AngularJS est construit selon l'architecture MVC. Son principal avantage qui peut parfois être un défaut est l'écoute réciproque des événements entre les différentes couches, les écouteurs d'événements sont gérés automatiquement. C'est exactement ce qu'il fallait pour pouvoir faire une interface qui interagisse avec le côté serveur via l'API REST ou encore faire des filtres de recherche très rapidement et simplement.

Bootstrap :

Bootstrap est un Framework CSS/JS sous licence Apache développé par la société californienne Twitter. Il est une collection d'outils utiles à la création de sites Web et d'applications Web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option .

Bootstrap offre du code CSS bien organisé et structuré. Il offre aussi des plugins jQuery de qualité pour enrichir les pages. Il permet d'avoir des pages, du code CSS propre convivial et bien organisé et le plus important c'est que grâce à bootstrap nous avons pu faire une application en mode responsive design.

2.5.3 Choix de l'architecture

2.5.3.1 Architecture générale

Le choix de l'architecture est décisif, du fait que la performance ainsi que l'évolutivité de l'application en sont dépendantes. Ainsi, il nous a déjà fallu définir un modèle architectural permettant de satisfaire les contraintes et répondre aux besoins spécifiés dans la section précédente. Le modèle le plus adapté à notre application est une architecture N-tiers.

L'architecture multi-niveaux appelée aussi architectures distribuées est une variante du modèle client-serveur dans laquelle le serveur se subdivise en plusieurs sous-composantes indépendantes.

Cette architecture est bâtie sur les services web en mettant l'accent sur les communications entre les différents tiers. En effet, nous pouvons distinguer les différentes couches de l'application :

- Client web : cette couche fournit à chaque utilisateur les interfaces graphiques nécessaires assurant la consultation et l'usage des services disponibles.
- Services web : ils présentent les services exposés par le serveur à distance. Et ils assurent la liaison entre le client et le serveur qui sont dans notre cas les services web Rest.
- La base de données : elle est responsable du stockage des données provenant de l'application.
- Serveur externe : il répond aux besoins de ses clients web et leurs fournit les réponses nécessaires.

La figure 2.5 illustre l'architecture globale de notre application.

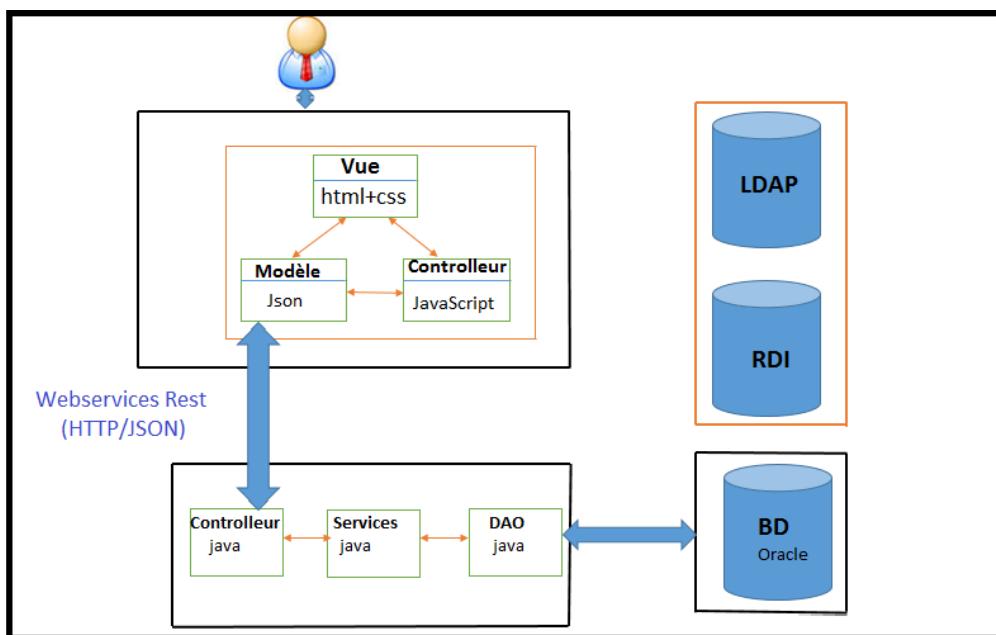


FIGURE 2.5 – Architecture générale de l'application

Coté client, c'est du **MVC**, un patron de conception qui permet d'avoir une stricte séparation entre la vue, le modèle et le contrôleur. Il ne s'agit pas de mettre l'ensemble de code au sein du même contrôleur. Il s'agit donc de développer plusieurs petits contrôleurs qui vont agir sur différents éléments de la page. Ceci aura comme avantage de séparer le code et de s'échapper de l'encombrement.

Coté serveur, l'exécution est simple, le serveur weblogic prend en charge le traitement Backend de l'application ainsi que les requêtes avec la base de données.

2.5.3.2 Architecture de site

Le site web de notre application est décrit dans la figure 2.6.

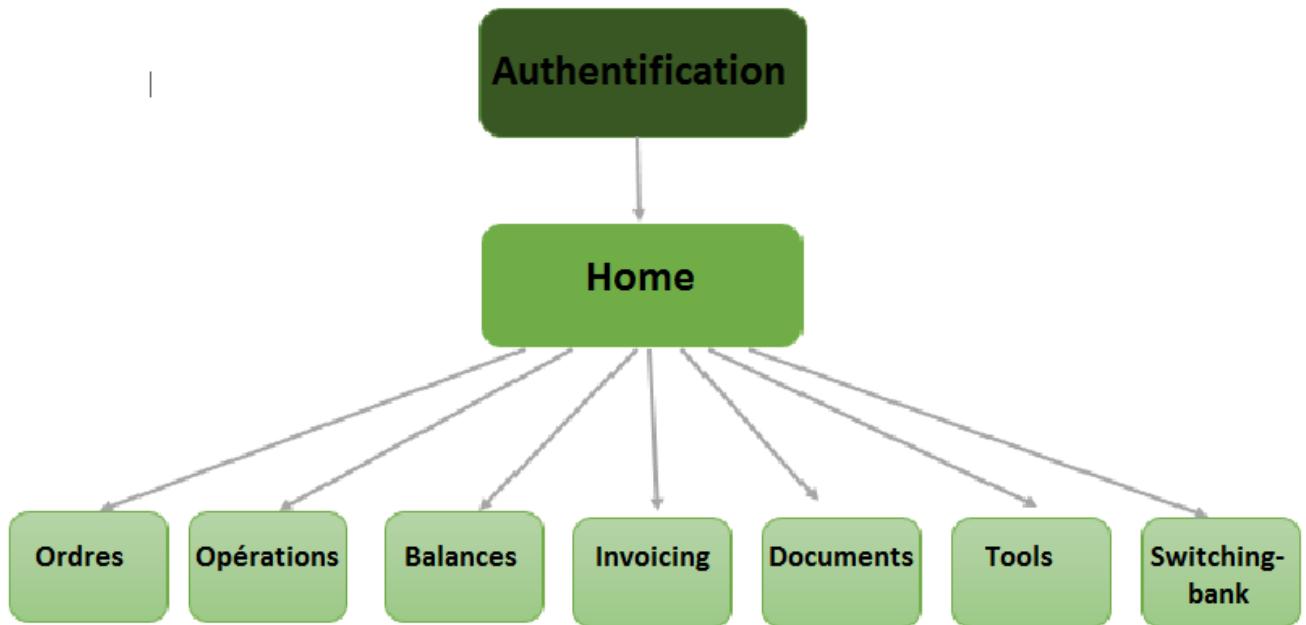


FIGURE 2.6 – Architecture de site

Notre application est complète et simple à utiliser. En ouvrant cette application, on se trouve sur la page d'authentification rapide, puis une page Home ayant différents onglets. Ces modules vont être détaillés plus tard.

2.5.4 Mise en oeuvre

2.5.4.1 Introduction

La phase de réalisation se base essentiellement sur la phase de conception pour transformer les besoins et les spécifications en fonctionnalités et de les délivrer à l'utilisateur final. Pour mener cette tâche convenablement, il est nécessaire de se donner les moyens organisationnels, matériels et logiciels nécessaires.

Dans cette partie, nous allons évoquer les aspects généraux de l'organisation de notre travail. Ensuite, nous donnerons un aperçu sur les modules réalisés.

2.5.4.2 Environnement de travail

Afin d'implémenter le système que nous avons conçu, nous avons eu recours à un environnement matériel ainsi qu'à un environnement logiciel particulier. Nous consacrons cette partie à l'exposition des architectures matérielle et des logiciels utilisés tout au long de l'élaboration de notre travail.

2.5.4.3 Environnement matériel

Afin de bien mener ce projet, il a été mis à notre disposition un ordinateur de bureau dont les caractéristiques sont les suivantes :

- RAM : 16 Go.
- Disque dur : 118 Go.
- 4 Disques réseau : 1000 Go.
- Micropprocesseur Intel i7-3770 3.40GHz
- Système d'exploitation : Windows 7 Entreprise

2.5.4.4 Environnement logiciel et choix technique

IDE :



Nous avons choisi Eclipse vu qu'il est un environnement de développement Open Source par des logiciels réalisés en langage Java. Cela est dû à sa richesse du point de vue fonctionnalité et extensibilité par les plu-gins. De plus, il est capable de supporter n'importe quel serveur d'application sans tenir compte de sa version ou son édition. [7]

Editeur de texte :



Notepad++ est un programme conçu spécifiquement pour l'édition de code source. nous avons choisi cet éditeur vu sa compatibilité avec plusieurs langages de programmation. [8]

SGBD : SQL Developer :

Afin de construire notre base de données, nous avons utilisé comme SGBD SQL Developer. Il permet de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. [9]

StartUML 2.7.0 :

StartUML est un logiciel de conception qui offre différentes techniques de modélisation, et permet de réaliser tous les types de modèle informatiques comme : Merise, UML, Data Warehouse, et processus métier. Simple d'utilisation, personnalisable et dotée d'une interface intuitive, cette application optimise la productivité individuelle et collective. [10]

Jenkins :

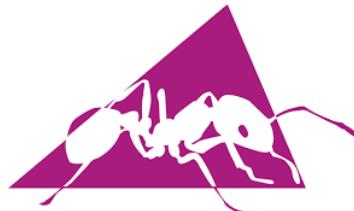
Jenkins, est un outil Open-Source d'Intégration Continue, écrit en JAVA. Pour faire simple, le principe est de vérifier, idéalement à chaque modification de code source, que le résultat de ces modifications de produit pas de régression sur l'application. [11]

Maven :

Maven est un outil de gestion de projet qui comprend un modèle objet pour définir un projet, un ensemble de standards, un cycle de vie, et un système de gestion des dépendances. Il embarque aussi la logique nécessaire à l'exécution d'actions pour des phases bien

définies de ce cycle de vie, par le biais de plug-in. L'utilisation de Maven, décrit le projet selon un modèle objet de projet clair. Il peut alors lui appliquer la logique transverse d'un ensemble de plug-in (partagés ou spécifiques). [12]

Ant :



Ant est un logiciel qui vise à automatiser les opérations répétitives du développement de logiciel telles que la compilation, la génération de pages HTML de documentation (Javadoc), la génération de rapports, l'exécution d'outils annexes (checkstyle, FindBugs, etc.), l'archivage sous forme distribuable (JAR, etc.). [13]

2.5.4.5 Git :Logiciel de gestion de versions :



Nous commençons par définir c'est quoi logiciel de gestion de version : Un logiciel de gestion de versions est un logiciel qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus. Il permet notamment de retrouver les différentes versions d'un lot de fichiers connexes. Il existe aussi des logiciels et services de gestion de versions décentralisé (distribué) (ou DVCS en anglais, pour Distributed Version Control System).

Pour gérer le code source de notre application, nous avons utilisé un logiciel de gestion de versions Git qui est un logiciel de gestion de versions décentralisé. Il permet ainsi de travailler tout en gardant une trace des modifications apportées successivement, et de retrouver ainsi l'état antérieur de ses données. Il est ainsi plus aisé de savoir quel développeur a effectué une modification, quand il a effectué, et surtout en quoi consistait

cette modification.

Fonctionnement

Afin de mieux expliquer comment git est utilisé dans le cadre du projet, il convient tout d'abord de définir quelques concepts importants :

- Un dépôt : un espace où sont conservées toutes les modifications apportées au code source.
- Un commit : une structure de donnée permettant de sauvegarder les informations concernant les changements qui ont été apportés dans le code source.
Un commit contient d'autres informations importantes comme le nom du développeur, la date, l'état du projet, ainsi qu'un message permettant d'éclaircir le but du commit.

Les commandes de bases de Git

Git est un logiciel fonctionnant en ligne de commandes, le tableau 2.3 présente quelques-unes qui sont couramment utilisées pour récupérer ou enregistrer des modifications du code source.

Commandes	Signification
git init	Créer un dépôt git vide ou initialiser un existant.
git status	Afficher les chemins qui ont des différences entre le fichier d'index et le HEAD courant .
git add <file>	Mettre à jour l'index en utilisant le contenu actuel trouvé dans l'arbre de travail, pour préparer le contenu mis en pour le prochain commit.
git rm <file>	Supprimer des fichiers de l'index.
git commit -m <msg>	Stocker les contenus actuel de l'index dans une nouvelle commit avec un message décrivant les changements.
git pull	Intégrer les changements depuis un dépôt distant dans la branche courante
git push	Envoyer les commits vers le dépôt distant

TABLE 2.3 – les commandes de bases de git

La figure 2.7 illustre le mode de fonctionnement de Git

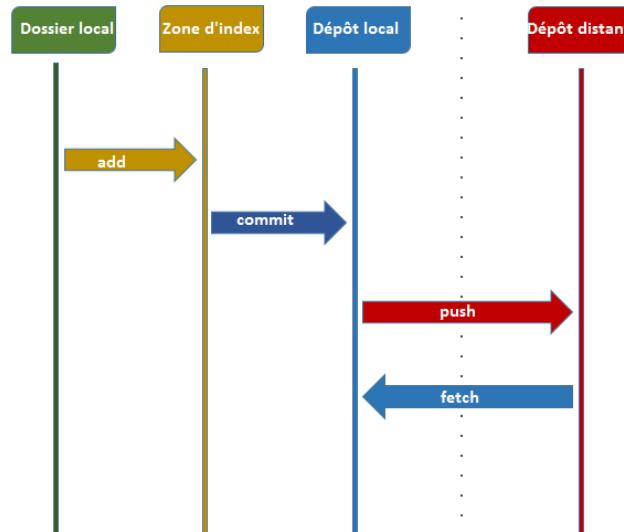


FIGURE 2.7 – le mode de fonctionnement de Git

2.6 Conclusion

Dans ce chapitre nous avons préparé notre backlog du produit, et nous avons présenté les différents Sprints de notre application. Nous avons présenté aussi les technologies récentes de développement web ainsi que le logiciel de gestion de versions Git. Dans les chapitres suivants, nous détaillons tous les sprints, mais chaque sprint sera étudié d'une manière différente des autres vu les spécifications de chaque menu. Dans le chapitre suivant, nous commençons par détailler le premier sprint de notre projet.

Chapitre 3

Sprint 1 : Menu authentification et menu Operations

3.1 Introduction

Après avoir entamé le chapitre phase de pré-jeu, nous pouvons maintenant passer aux travaux nécessaires pour produire le premier sprint ”menu authentification et menu Opérations”. Tout au long de ce chapitre, nous allons donner une vision statique à travers les diagrammes de cas d'utilisation détaillés. Ensuite nous présentons une vue dynamique à travers les diagrammes de séquence et, pour terminer, nous présentons la réalisation et les tests.

3.2 But du sprint

Avant de se lancer dans le premier sprint, l'équipe Scrum doit obligatoirement définir le but de ce dernier. Il s'agit de répondre à une question fondamentale ”pourquoi faisons-nous ce sprint ?”. Suite à une conversation entre le Product Owner et l'équipe Scrum, nous avons décidé le but suivant : ”Permettre à un client de s'authentifier et accéder à ses opérations”.

3.3 Menu Authentification

3.3.1 Authentification WEBCLEAR

Le client saisit son identifiant et son mot de passe en utilisant un clavier virtuel(Le terminal peut mobile ou desktop) qui seront envoyés au **serveur http** qui les transmet au

serveur **LDAP** qui se charge de vérifier l'identité. Si l'authentification réussit, le serveur http se charge après de vérifier les droits d'accès, il envoie donc une requête au **RDI** (Référentiel De l'International) pour récupérer le numéro du profil de l'utilisateur. Après la récupération de ce numéro il va demander au serveur **WebLogic** d'accéder à la base de données pour récupérer les droits de ce client en utilisant son numéro de profil.

Selon les droits de cet utilisateur le serveur applicatif choisit ce qui doit être affiché au client et envoie au serveur http pour lui dire qu'est ce qu'il doit afficher en authentifiant le client.

L'authentification du client est bien décrite par la figure 3.1 et le tableau 3.1 :

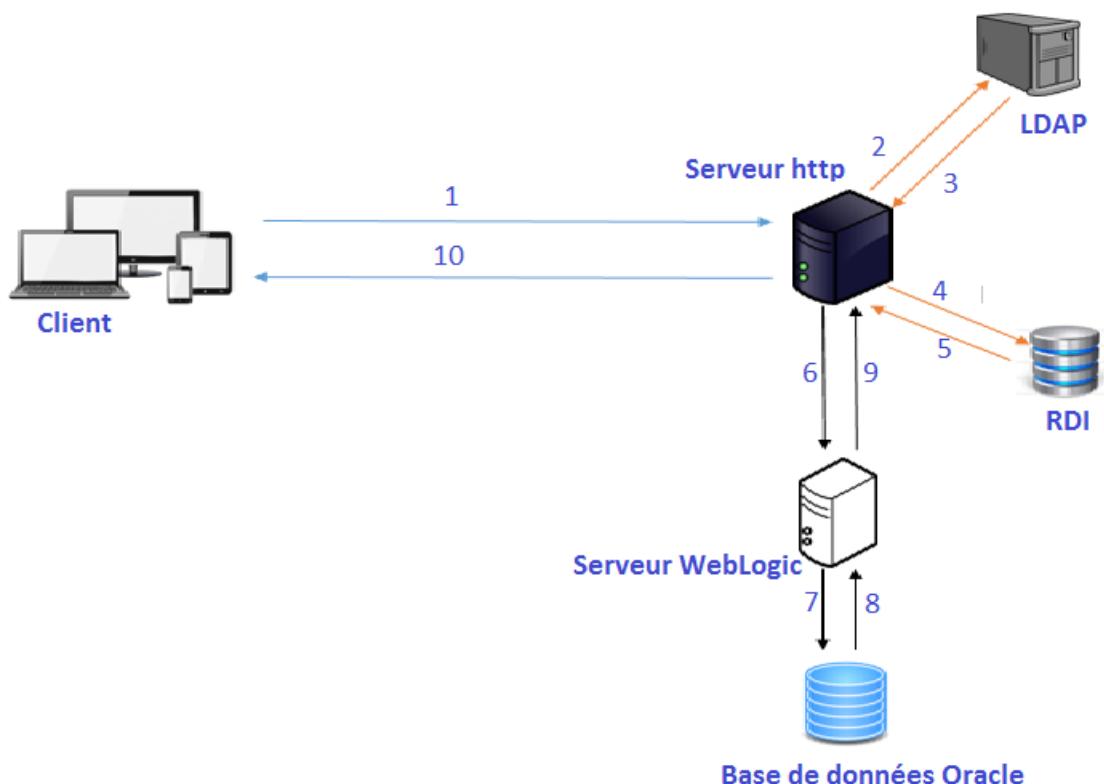


FIGURE 3.1 – Authentication du client

Etape	Signification
1	Demande d'authentification
2	Demande de vérification de l'identité
3	Réponse
4	Demande du numéro du profil du client
5	Envoi du numéro du profil du client au serveur http
6	Envoi du numéro du profil du client au serveur weblogic
7	Demande des droits du client
8	réponse
9	réponse
10	Authentification + Habilitation

TABLE 3.1 – Étapes de l'authentification

3.3.2 Diagramme de séquence d'Authentification

Chaque opération du diagramme de cas d'utilisation va donner lieu à une étude dynamique sous la forme d'un diagramme d'activité ou d'un diagramme de séquence.

Un diagramme de séquence est une forme de diagramme comportemental permettant de spécifier les interactions qui existent entre les objets, il permet de voir comment les objets s'utilisent mutuellement. Les principaux objets dans ce diagramme sont :

coté client :

- Route : Elle spécifie les classes contrôleur associées aux vues et représente l'intermédiaire entre le navigateur et l'interface.

- Le contrôleur : Il contrôle les données.

coté Serveur :

- Route : Elle définit la communication entre contrôleurs coté client et les contrôleurs coté serveur en utilisant les API de serveur.

- Le contrôleur : coordonne les modifications au modèle

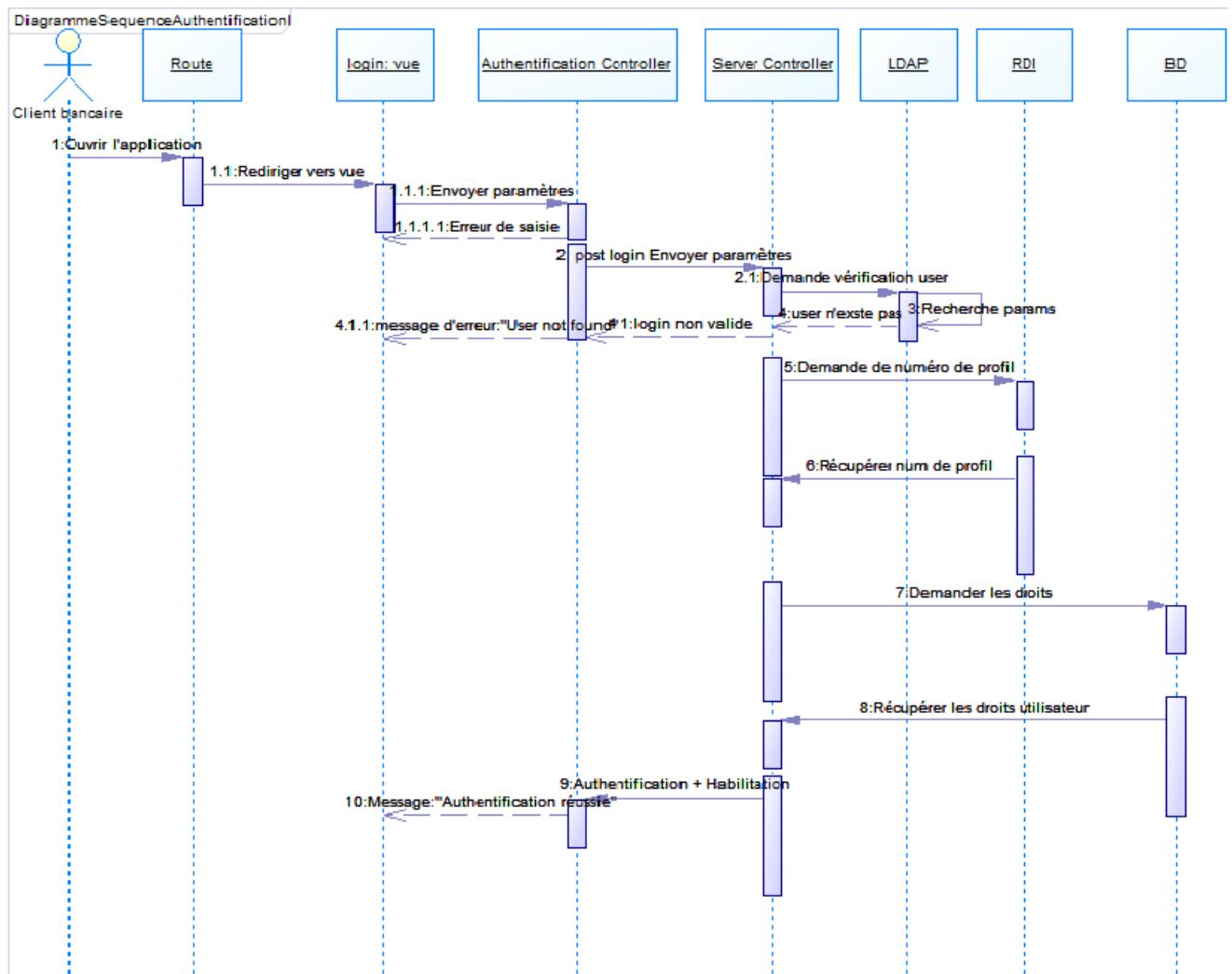


FIGURE 3.2 – Authentification du client

Nous décrivons par la suite une description textuelle du cas d'utilisation d'authentification dans le tableau 3.2

Titre	S'authentifier
Résumé	permet à un client d'accéder à son compte
Acteurs	Client bancaire
Préconditions	Acteur a un login et un mot de passe. Acteur sur l'écran de login.
Scénario nominal	1. Acteur saisit son login et son mot de passe. 2. Acteur clique sur le bouton "Connexion". 3. Le système vérifie le login et le mot de passe et affiche l'écran d'accueil de l'application.
Enchainements alternatifs	A1 : L'utilisation n'a pas rempli un champ obligatoire L'enchaînement A1 démarre au point 2 du scénario nominal. Le système informe l'acteur qu'il doit saisir toutes les informations obligatoires. Le scénario nominal reprend au point 1.
Enchainements d'erreur	E1 : Login et/ou mot de passe incorrecte L'enchaînement E1 démarre au point 3 du scénario nominal. Le système informe l'utilisateur que les informations saisies sont incorrectes. Le scénario nominal reprend au point 1.
Post-Conditions	Page d'accueil de l'application s'affiche.

TABLE 3.2 – Description du diagramme de séquence "S'authentifier"

3.3.3 Implémentation

L'implémentation est le résultat de la conception, son importance est cruciale pour la réalisation du système. Dans cette partie, nous illustrons à ce niveau les captures d'écran réalisés au niveau de la première partie du sprint 1 qui est l'authentification .

La figure 3.3 illustre l'interface d'authentification au niveau de laquelle l'utilisateur accède à l'application. Il doit donc s'authentifier en saisissant son identifiant et son code secret à travers un clavier virtuel.

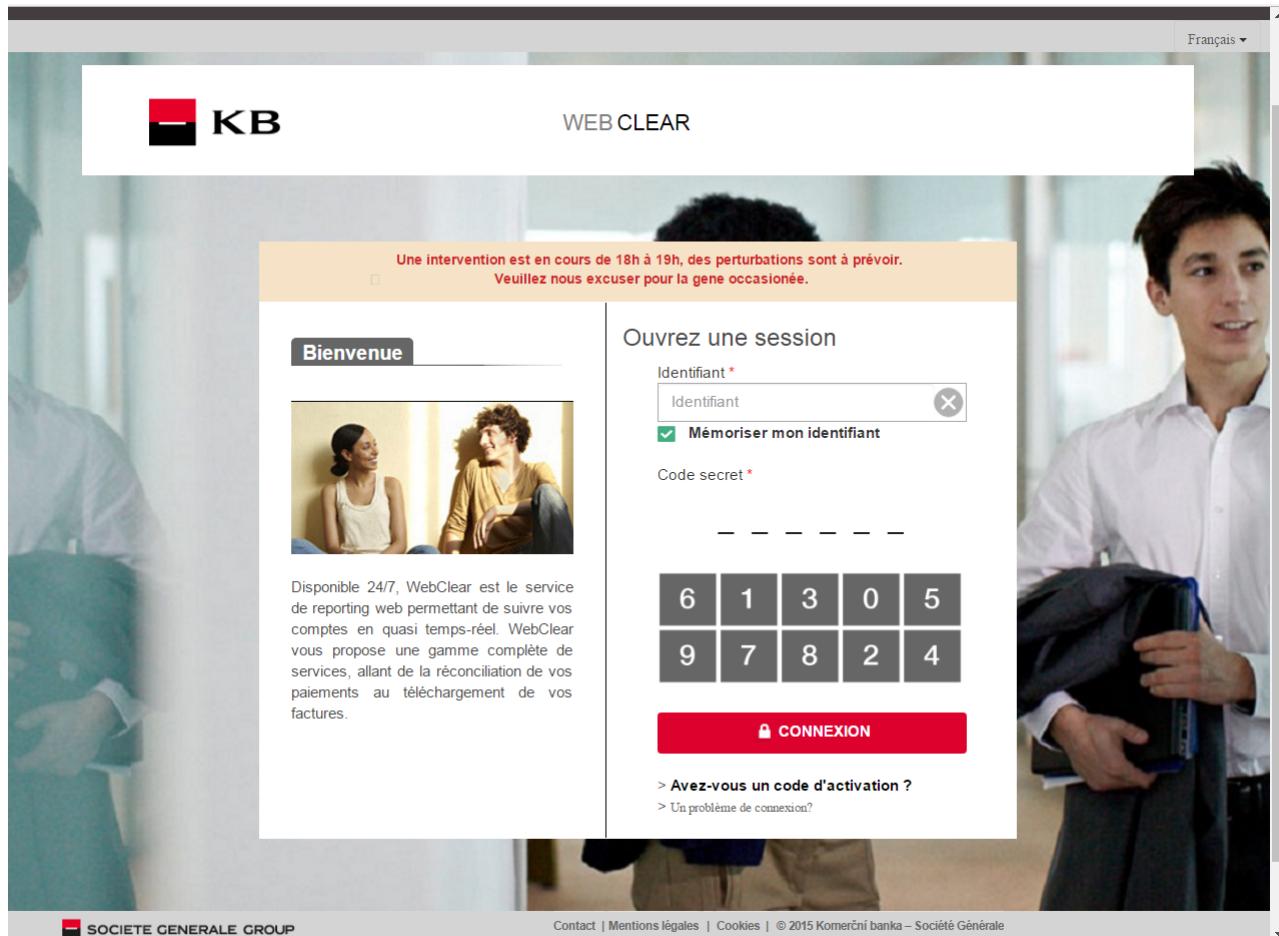


FIGURE 3.3 – Interface d’authentification

Après l’authentification en mode desktop, la page d’accueil s’affiche, elle contient les différents clients bancaire de la Société Générale, si on clique sur un client à gauche, il s’affiche à la colonne qui est à droite les différents comptes qu’il possède. Cette page contient aussi en haut les différents menus de l’application.

La figure 3.4 montre l’interface home en mode desktop

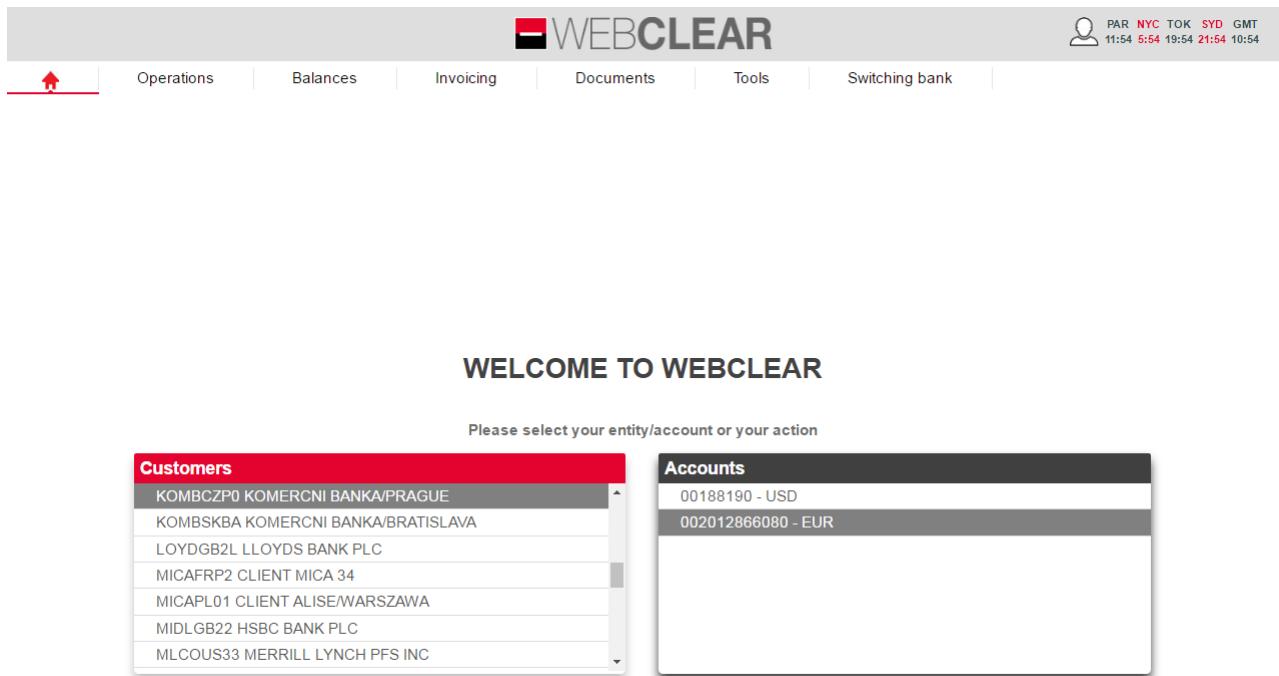


FIGURE 3.4 – Interface Home - mode desktop

En mode tablette, l’interface home n’a plus le même aspect, les deux colonnes ”Customers” et ”Accounts” sont maintenant l’une en haut et l’autre en bas. La figure 3.5 montre l’interface home en mode tablette et la figure.



WELCOME TO WEBCLEAR

Please select your entity/account or your action

Customers

- KOMBCZP0 KOMERCNI BANKA/PRAGUE
- KOMBSKBA KOMERCNI BANKA/BRATISLAVA
- LOYDGB2L LLOYDS BANK PLC
- MICAFRP2 CLIENT MICA 34
- MICAPL01 CLIENT ALISE/WARSZAWA
- MIDLGB22 HSBC BANK PLC
- MLCOUS33 MERRILL LYNCH PFS INC

Accounts

- 00188190 - USD
- 002012866080 - EUR

FIGURE 3.5 – Interface Home - mode tablette

La figure 3.6 montre qu'en mode tablette, la barre des menus n'est plus horizontale, elle devient verticale dans ce mode.

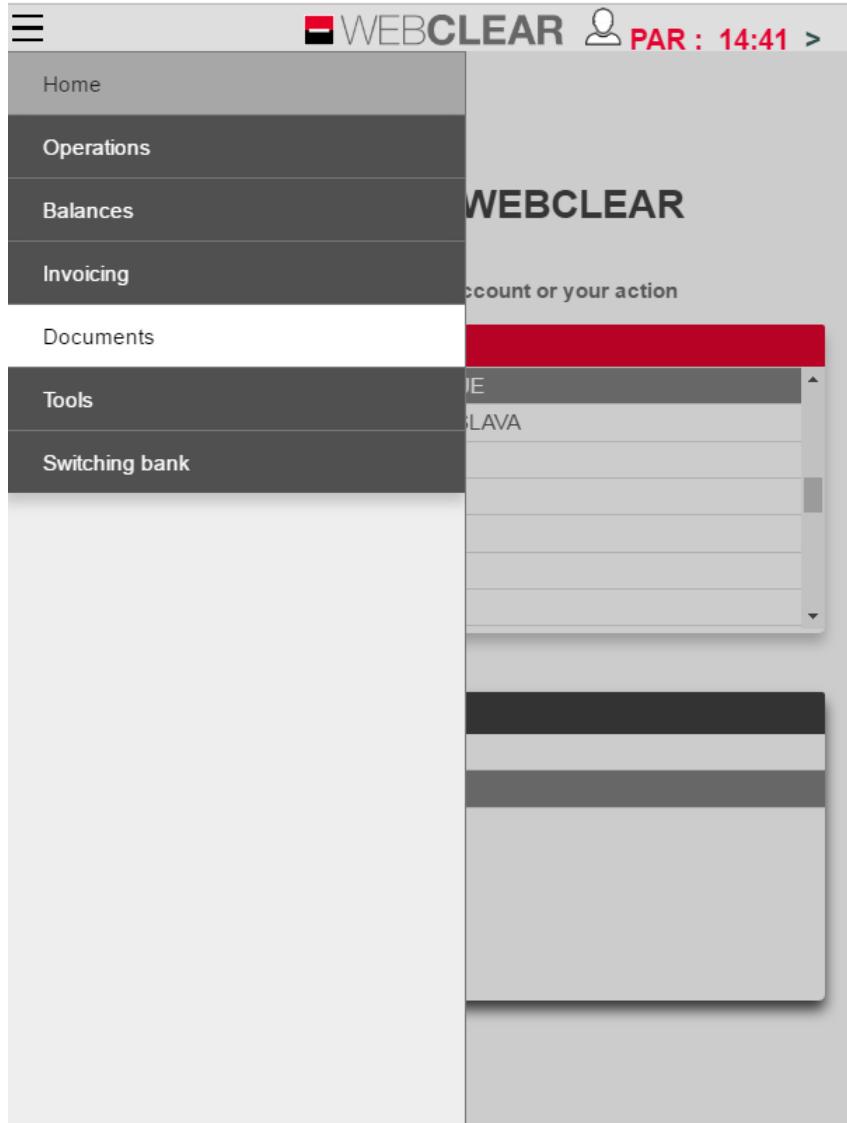


FIGURE 3.6 – Interface Home - version desktop

3.4 Menu Opérations

3.4.1 Architecture du menu Opérations

Le menu Opérations est composé de 4 sous menus : Entries, Mass payments, Reconciliations bancaires et Auto F/x.

-Le menu Entries se décompose en 2 sous menus : **Effectives** qui désigne les opérations effectuées et **Cancelled** qui désigne les opérations annulées

-Le menu Mass payments se décompose en trois sous menus ils sont liés au sepa, **Payments received** ce menu permet à un client de faire des virements de masse, c'est à dire qu'il va envoyer un seul fichier à la société générale et c'est à elle de se débrouiller pour

répartir les différentes opérations, le sous menu **Payments issued** ce sont les payments de masse émis par la société générale, et le sous menu **Details** qui donne aux clients les détails des différentes opérations effectuées.

-Le menu Reconciliations se décompose en 2 sous menus : **MT210 reconciled** ce menu permet de récupérer les messages swift de type MT210 c'est à dire les annonces et **MT210 not reconciled** il permet de récupérer les messages swift de type MT210 qui n'étaient pas réconciliés c'est à dire les messages dont on n'a pas pu trouvé des rapprochement entre eux et les messages swift de type MT202 c'est à dire qu'on n'a pas trouvé les opérations. La figure 3.7 montre l'architecture du menu Opérations.

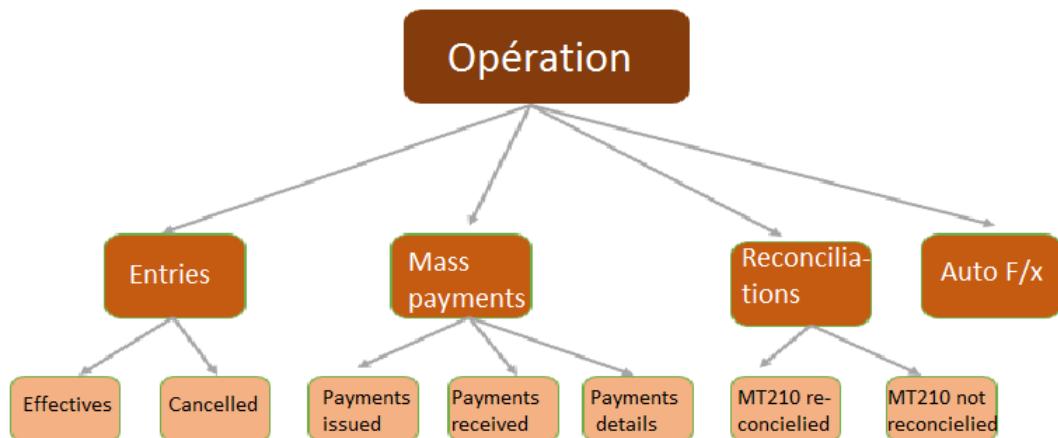


FIGURE 3.7 – Architecture du menu Opérations

3.4.2 Implémentation

Tout en cliquant sur le sous menu Entries du menu Operations, le client sera redirigé vers une page lui indiquant qu'il n'y aucune donnée qui lui permet de cliquer sur "SEARCH CRITERIA " s'il veut entrer certain critères de recherche. La figure 3.8 illustre cette page.

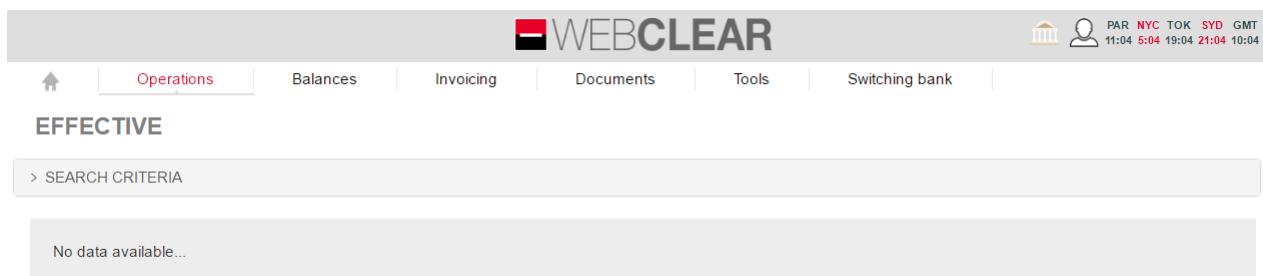


FIGURE 3.8 – Interface de no data available

La figure 3.9 est une imprime écran de la page qui permet au clients d'entrer certains critères de recherche pour accéder aux tableaux Entries.

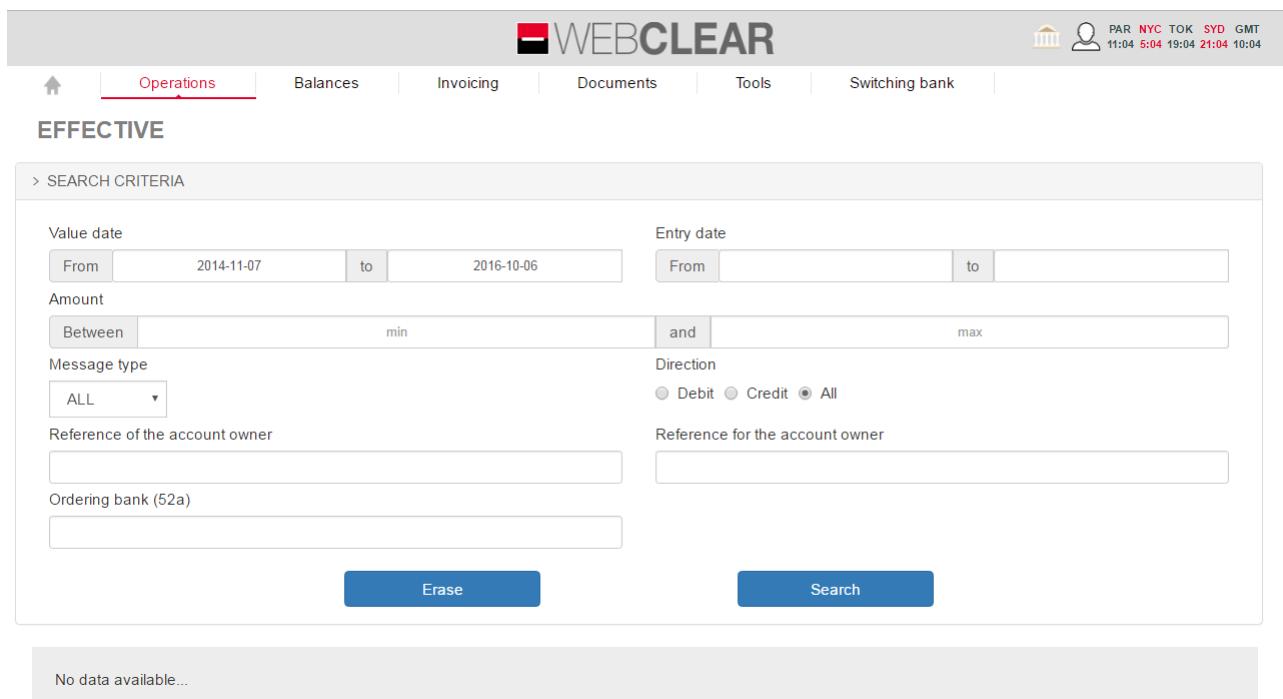


FIGURE 3.9 – Interface de Search criteria

Après la saisie des critères de recherche, les tableaux d'Entries s'affiche. La figure 3.10 est un exemple des tableaux d'Entries.

Reference of the account owner	Message type	Debit/Credit	Amount	Value date	Payment status	Clearing system	Processing Date&time	Details
0001115102211592	202	C	1,000,000.00	2015-10-26	Executed	Other	2015-10-26 07:45	
0001115102211593	202	C	1,000,000.00	2015-10-26	Executed	Other	2015-10-26 07:45	
0001W15102306385	202	C	549,675.66	2015-10-26	Executed	Other	2015-10-26 07:45	
00670529928542DO	202	C	119,578.09	2015-10-26	Executed	Other	2015-10-26 09:32	
01-4640619-00-OP	202	C	2,000,000.00	2015-10-26	Executed	Other	2015-10-26 09:31	
01-4640620-00-OP	202	C	2,000,000.00	2015-10-26	Executed	Other	2015-10-26 09:31	
019938752996854A	103	C	47,772.16	2015-10-26	Executed	Correspondent Banking	2015-10-26 09:31	
021120155092822	SDD	D	368.00	2015-10-30	Executed	Correspondent Banking	2015-10-26 00:00	
021120155092865	SDD	D	802.00	2015-10-30	Executed	Correspondent Banking	2015-10-26 00:00	
021305299285429Y	202	C	137,606.80	2015-10-26	Executed	Other	2015-10-26 09:32	
02130529928542DG	202	C	24,471.86	2015-10-26	Executed	Other	2015-10-26 09:32	
03MT151023308408	202COV	C	37,576.83	2015-10-26	Executed	Other	2015-10-26 08:08	
03MT151023309225	202COV	C	15,411.18	2015-10-26	Executed	Other	2015-10-26 08:08	

FIGURE 3.10 – Interface de Entries effective

Si on clique sur le symbole + qui à gauche, d'autres colonnes qui n'ont pas pu s'afficher à cause de la résolution de l'écran s'affiche.

La figure 3.11 représente cette interface

CHAPITRE 3. SPRINT 1 : MENU AUTHENTIFICATION ET MENU OPERATIONS

The screenshot shows the WEBCLEAR application interface. At the top, there's a navigation bar with links for Home, Operations (which is highlighted in red), Balances, Invoicing, Documents, Tools, and Switching bank. On the right side of the header, there are icons for a user profile, time zone (PAR 11:06), and other locations (NYC, TOK, SYD, GMT). Below the header, the word "EFFECTIVE" is displayed in bold capital letters. Underneath, there's a search bar with the placeholder "SEARCH CRITERIA". The main content area shows a table of transactions. The columns are: Reference of the account owner, Message type, Debit/Credit, Amount, Value date, Payment status, Clearing system, Processing Date&time, and Details. The first row of the table is expanded to show more details: "Reference for the account owner : DABAKK7052KOMBPP", "Ordering bank (52a):", "Ordering customer (50a):", "Beneficiary (59a):", and "Bulk sequence reference:". Below the table, it says "Showing 1 to 15 of 280 entries" and has navigation buttons for First, Previous, Next, and Last.

Reference of the account owner	Message type	Debit/ Credit	Amount	Value date	Payment status	Clearing system	Processing Date&time	Details
(+) 0001115102211592	202	C	1,000,000.00	2015-10-26	Executed	Other	2015-10-26 07:45	
Reference for the account owner : DABAKK7052KOMBPP Ordering bank (52a): Ordering customer (50a): Beneficiary (59a): Bulk sequence reference:								
(+) 0001115102211593	202	C	1,000,000.00	2015-10-26	Executed	Other	2015-10-26 07:45	
(+) 0001W15102306385	202	C	549,675.66	2015-10-26	Executed	Other	2015-10-26 07:45	
(+) 00670529928542DO	202	C	119,578.09	2015-10-26	Executed	Other	2015-10-26 09:32	
(+) 01-4640619-00-OP	202	C	2,000,000.00	2015-10-26	Executed	Other	2015-10-26 09:31	
(+) 01-4640620-00-OP	202	C	2,000,000.00	2015-10-26	Executed	Other	2015-10-26 09:31	
(+) 019938752996854A	103	C	47,772.16	2015-10-26	Executed	Correspondent Banking	2015-10-26 09:31	
(+) 021120155092822	SDD	D	368.00	2015-10-30	Executed	Correspondent Banking	2015-10-26 00:00	

FIGURE 3.11 – Interface de Plus de colonnes

Si on clique sur la loupe qui est à droite, le client sera redirigé vers une autre page qui contient des détails sur l'opération considérée La figure 3.12 représente cette interface

CHAPITRE 3. SPRINT 1 : MENU AUTHENTIFICATION ET MENU OPERATIONS

WEBCLEAR

PAR NYC TOK SYD GMT
18:37 12:37 2:37 4:37 17:37

EFFECTIVE

Form	Swift
Message type	202
Debit/Credit	Credit
SG Reception date	2015-10-26
Amount	1,000,000.00
Currency	EUR
Value date	2015-10-26
Entry date	2015-10-26
Original amount	
Entry status	Effective
Original currency	
Payment status	Executed
Matched with 950	No

Reference of the account owner	0001115102211592
Reference for the account owner	DABAKK7052KOMBPP
Informations to the account owner	/58//002012866080/KOMBCZPP
Additional details	
Bic sender	DABADKKXXXX
Bic receiver	SOGEFRPP
Ordering bank (52a)	

Payment status record

Title	Code	Date and time	Reason for Reject
Status: executed	06	2015-10-26 07:45:24	

OK

FIGURE 3.12 – Interface de Details

3.5 Conclusion

A la fin de ce sprint, nous avons produit un incrément ayant suffisamment de valeur pour le client et pourra être utilisé dans un environnement de production. Dans le chapitre suivant, notre effort sera consacré pour produire un nouvel incrément couvrant les soldes et les facturations.

Chapitre 4

Sprint 2 : Menu Balance et menu Invoicing

4.1 Introduction

En partant sur le même principe que le chapitre précédent, nous commençons par définir le but de ce sprint, puis nous passons à détailler les fonctionnalités

4.2 But du sprint

Le but de ce sprint est de permettre à un client bancaire d'accéder à ses soldes comptables ainsi que ses soldes en valeurs et d'accéder à sa facturation.

4.3 Menu Balance

4.3.1 Architecture du menu Balance

La figure 4.1 montre l'architecture du menu Balance

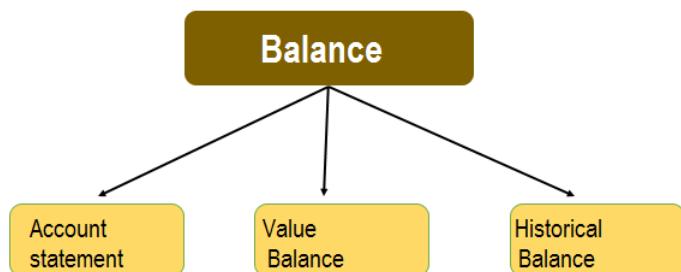


FIGURE 4.1 – Architecture du menu Balance

Le menu Balance est composé de 4 sous menus : Account statement, Value balance et Hitorical balance.

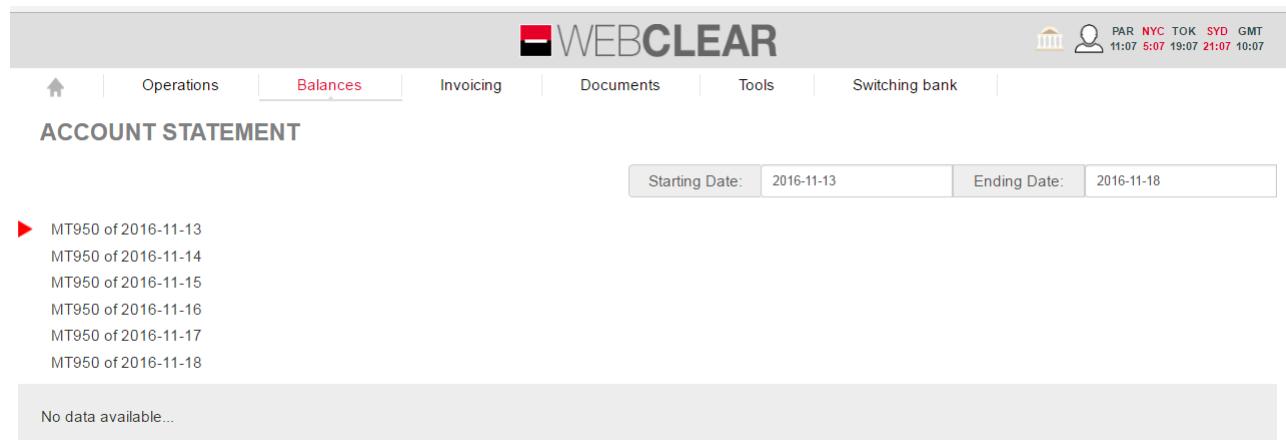
-Le menu Account statement permet de suivre la comptabilité dans la journée, c'est à dire que suite à une opération, on met à jour les soldes.

-Le menu Value balance permet aussi de suivre la comptabilité dans la journée, c'est à dire que suite à une opération, on met à jour les soldes mais c'est soldes ils ne sont pas encore sûres vu que l'opération n'a pas été encore effectuée.

-Le menu Historical balance permet de savoir l'historique des soldes de valeurs ainsi que les soldes comptables

4.3.2 Impléménatation

La figure 4.2 est une imprime écran du menu Account statement, il montre les différentes opérations effectuées entre les dates 13 Novembre 2016 et 18 Novembre 2016



The screenshot shows the WEBCLEAR software interface. At the top, there's a navigation bar with links for Operations, Balances (which is currently selected and highlighted in red), Invoicing, Documents, Tools, and Switching bank. On the far right of the top bar, there are icons for a user profile and time zone information (PAR, NYC, TOK, SYD, SYD, GMT) with specific times like 11:07, 5:07, 19:07, 21:07, and 10:07. Below the navigation bar is a search bar with fields for 'Starting Date' (set to 2016-11-13) and 'Ending Date' (set to 2016-11-18). The main content area is titled 'ACCOUNT STATEMENT' and contains a list of transaction entries. Each entry starts with a red triangle icon followed by the code 'MT950' and a date, such as 'MT950 of 2016-11-13', 'MT950 of 2016-11-14', etc., up to 'MT950 of 2016-11-18'. At the bottom of this list, there's a grey box containing the text 'No data available...'. The overall layout is clean and professional, typical of financial software interfaces.

FIGURE 4.2 – Interface de Account statement

4.4 Menu Invoicing

L'Invoicing est une façon pour les entreprises d'emprunter de l'argent sur la base des montants dus par les clients. Elle aide les entreprises à améliorer les banques de trésorerie, payer les employés et les fournisseurs, et réinvestir dans les opérations et la croissance plus tôt que ce qu'ils pouvaient si elles devaient attendre jusqu'à ce que leurs clients les payaient. Les banques clientes paient un pourcentage du montant de la facture pour la société générale à titre de frais pour emprunter de l'argent. Dans notre cas, l'application WEBCLEAR ne gère pas L'invoicing, elle affiche seulement les factures récupérées de la

CHAPITRE 4. SPRINT 2 : MENU BALANCE ET MENU INVOICING

part de l'équipe Actif et pour des droits de confidentialités, nous n'avons pas les droits de publier ces factures.

Chapitre 5

Sprint 3 : Menu Orders et menu Documents

5.1 Introduction

En partant sur le même principe que les 2 chapitre précédents, nous commençons par définir le but de ce sprint , puis nous passons à détailler les fonctionnalités, par contre pour la première partie de ce chapitre qui est le menu Ordres, on va avoir plus de détails que les autres menus par ce que toutesles fonctionnalités de ce menu sont fait par notre équipe.

5.2 But du sprint

Le but de ce sprint est de permettre à un client bancaire de faire, de signer ainsi que de récupérer des ordres et d'avoir la documentation nécessaire.

5.3 Menu Orders

5.3.1 Architecture du menu Ordres

La figure 5.1 montre l'architecture du menu Ordres

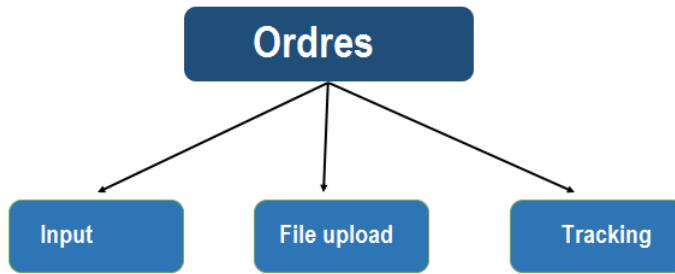


FIGURE 5.1 – Architecture du menu Ordres

Le menu Ordres présente l'ensemble des services liés aux ordres de virement. Nous trouverons les services de :

- Saisie d'ordre de virement,
- Suivi d'ordre de virement initié à l'aide du Web.

Les principes de base sont les suivants :

- La signature disjointe : un ordre saisi et validé devra dans un second temps être signé.
- Le principe des "4 yeux" : l'utilisateur qui saisit ne signe pas et inversement.
- La signature simple : un ordre signé par un seul utilisateur habilité.
- Des notifications utilisateur envoyés par e-mail :
 - faisant office de preuve de saisie ou de signature,
 - pour action.

5.3.2 Périmètre

Un utilisateur habilité au service Ordres pourra saisir des ordres de virement aux formats SWIFT suivants :

- MT103 : Virement commercial unitaire
- MT200 : Virement interbancaire pour compte propre
- MT202 : Virement interbancaire de trésorerie
- MT202 COV : Virement interbancaire de couverture

Un ordre pourra éventuellement être accompagné d'une opération de change (par exemple : un virement en USD à partir d'un compte à débiter en EUR)

5.3.3 Fonctionnalités

Les fonctionnalités proposées sont les suivantes :

- la saisie d'un ordre de virement,
- la signature d'un ordre de virement
- le suivi des ordres de virement initiés à l'aide du Web
- la sauvegarde d'un ordre de virement,
- le renvoi d'un ordre à un utilisateur de profil saisie afin de le modifier,
- l'annulation d'un ordre,
- l'export au format pdf d'un ordre de virement pour impression et/ou archivage.

5.3.4 Profils de connexion

Un utilisateur abonné à ce service est autorisé à travailler sur un périmètre de comptes et est rattaché à un profil de connexion.

Il existe 3 profils de connexion dédiés au service Ordres :

- **Profil Input** : l'utilisateur peut consulter et effectuer une saisie sans limite de montant ; cette dernière devant ensuite être signée par un utilisateur avec un profil Confirm.

Un profil Input pourra également :

- annuler un ordre avec le statut A signer.
- sauvegarder un ordre non validé afin de le reprendre plus tard pour le modifier, l'annuler ou le soumettre pour signature.
- modifier un ordre retourné par le profil Confirm suite à la détection d'un problème sur les informations saisies.

- **Profil Confirm** : l'utilisateur peut consulter et signer une saisie effectuée et validée par un profil Input.

Il pourra par ailleurs annuler un ordre mais ne pourra pas saisir un ordre ou modifier une saisie. Un plafond de montant par ordre pourra, si besoin, être attribué à un profil Confirm :

- Si le montant de l'ordre à signer est inférieur au plafond de signature de l'utilisateur, ce dernier peut signer.
- Si le montant de l'ordre à signer est supérieur au plafond de signature de l'utilisateur, ce dernier ne sera pas habilité à signer. Seul un autre profil Confirm avec un plafond plus élevé pourra signer.

- **Profil Consult** : l'utilisateur peut uniquement consulter les ordres signés ou saisis en attente de signature.

Illustration :

	Mme X	M. Y	Mme. Z
Profil de connexion	Input	Confirm - plafond 20 K euros	Confirm - plafond illimité

TABLE 5.1 – Exemple des différents profils

Les habilitations définies ci-dessus, ont pour conséquence que :

- Mme X peut saisir un ordre de virement quelque soit le montant de l'ordre.
- M. Y peut signer un ordre de virement uniquement si le montant de l'ordre est inférieur à 20 k euros.
- Mme Z peut signer un ordre de virement que soit le montant de l'ordre.

5.3.5 Heure limite de signature des ordres

- Les ordres sont transmis en temps réel par SG aux Systèmes d'échange ou aux correspondants en fonction des jours d'ouverture et des plages horaires.
- Le client peut saisir ses ordres jusqu'à 30 jours ouvrés avant la date de règlement souhaitée. Ces ordres sont stockés et transmis au système d'échange ou au correspondant le jour de la date de règlement indiquée.

5.3.6 Implémentation

On remarque que dans toutes les autres imprimés écrans, le menu Orders n'existe pas et ceci est du au fait que le client avec le client on s'est connecté n'a pas le droit d'accéder à ce menu.

Les figure 5.2, 5.3 et 5.4 sont des imprimés écrans du menu Orders Détailé ci-dessus

CHAPITRE 5. SPRINT 3 : MENU ORDERS ET MENU DOCUMENTS

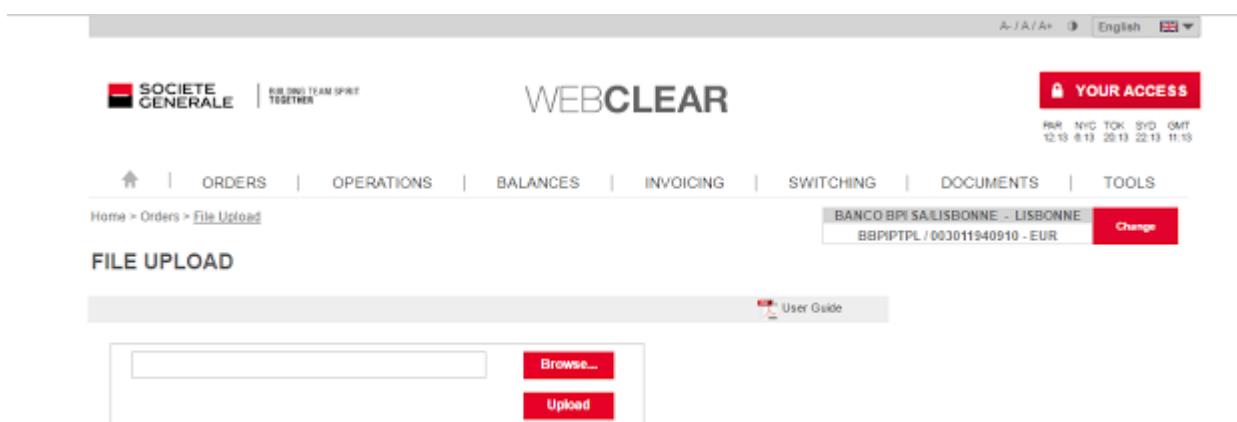


FIGURE 5.2 – Interface de File Upload

The screenshot shows the WEBCLEAR interface with a navigation bar at the top. The 'INPUT' section is displayed, containing various form fields. At the top, there are options for 'Credit transfer' (radio buttons for 'Financial Institution Transfer' and 'Customer Credit Transfer'), 'Message type' (dropdown menu showing 'MT200-Financial Institution Transfer for its Own Account'), and 'Template' (dropdown menu showing 'To Select'). Below these are fields for 'Currency (32A)' (EUR), 'Amount (32A)' (0.00), and 'Value Date (32A) (YYYY-MM-DD)' (2016-11-21). The 'Sender's Correspondent (53B)' section includes 'Account' (003011940910) and 'Location' (BBPIPTPL). There are sections for '+ Intermediary Institution (56A)', 'Account with Institution (57A)', and '+ Sender to Receiver Information (72)'. At the bottom, there are buttons for 'Discard', 'Template', 'Save', and 'Submit'. A note at the bottom left indicates that some fields are mandatory.

FIGURE 5.3 – Interface de Input

The screenshot shows the WEBCLEAR interface for tracking orders. The top navigation bar includes links for Home, ORDERS, OPERATIONS, BALANCES, INVOICING, SWITCHING, DOCUMENTS, and TOOLS. A red banner at the top right says "YOUR ACCESS". Below the navigation, a sub-menu shows "KOMERCI NI BANKA PRAGUE - PRAHA" and "KOMBCZP0 / 002012B66080 - EUR". The main content area is titled "TRACKING" and contains a table of "Orders List". The table has columns for Reference Number, Value Date, Amount, Currency, Message Type, Status, Ordering Institution (52a), Cancel limit date, and PDF. The table lists 110 orders, mostly from 2014, with various statuses like "To be amended", "Saved", and "Sent". At the bottom of the table, it says "110 orders found, displayed from 1 to 15." Below the table, there are links for "Current page: 1" and "2 3 4 5 6 7 8 ►". At the very bottom, there are "PDF Export" and "Export EXCEL" buttons.

FIGURE 5.4 – Interface de Tracking

5.4 Menu Documents

5.4.1 Architecture du menu Documents

La figure 5.5 représente l'architecture du menu Documents :

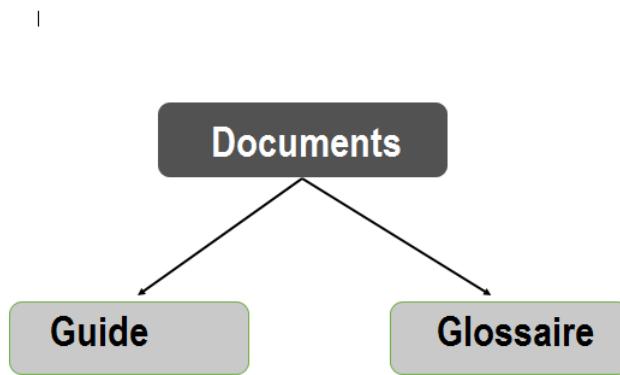


FIGURE 5.5 – Architecture du menu Documents

Le menu Documents contient deux types de documents le premier est : **Guide**, selon le profil de l'utilisateur et les menus à lesquels il a accès on a un ou plusieurs guides. Le deuxième est : **Glossaire** qui affiche à un clients l'ensemble des messages réparés et pour qu'elle raisons ils ont été réparés. Selon le profil de l'utilisateur, on trouve un ou plusieurs glossaires.

5.4.2 Implémentation

Pour des raisons de confidentialités et de sécurités, on va afficher dans ce menu des interfaces de test. La figure 5.6 est une imprime écran de l'interface Documents en mode Desktop et La figure 5.7 est celle en mode tablette :

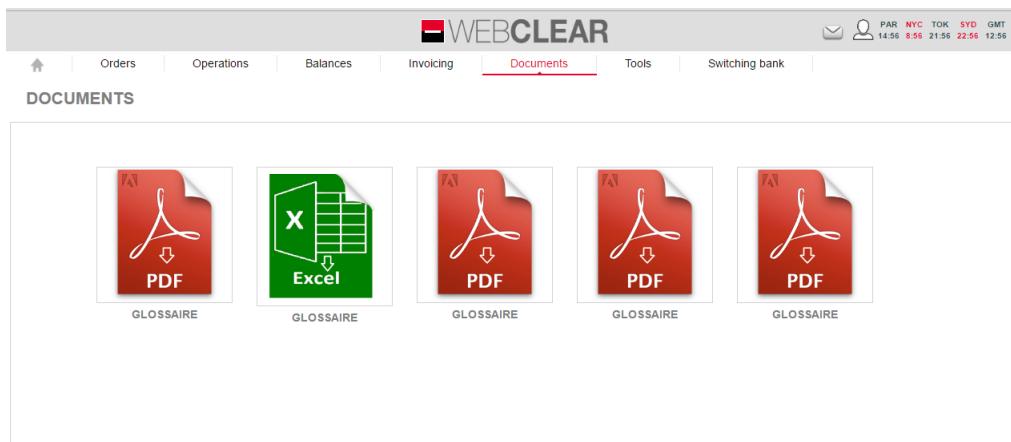


FIGURE 5.6 – Interface du menu documents - mode desktop



FIGURE 5.7 – Interface du menu documents - mode tablette

5.5 Conclusion

A la fin de ce sprint, nous avons détaillé 2 menus importants dans notre application qui sont le menu Ordres et le menu documents. Dans le chapitre suivant on va passé à deux autres menus qui sont le menu Tools et le menu Switching Bank

Chapitre 6

Sprint 4 : Menu Tools et menu Switching Bank

6.1 Introduction

En partant sur le même principe que le chapitre précédent, nous commençons par définir le but de ce sprint, puis nous passons à détailler les fonctionnalités

6.2 But du sprint

Le but de ce sprint est de permettre à un client bancaire de vérifier son IBAN ainsi que de se bénéficier d'un transfert automatisé de ses domiciliations bancaires ainsi que des opérations sur son compte.

6.3 Menu Tools

6.3.1 Architecture du menu Tools

La figure 6.1 montre l'architecture du menu Tools

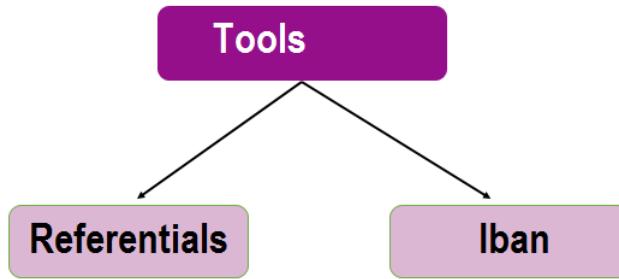


FIGURE 6.1 – Architecture du menu Tools

Le menu Tools est composé de deux sous menus : Le premier intitulé **Referentials** contient des références qu'on reçoit, elles sont stockées dans la base de données sous formes de blob. Le deuxième intitulé **Iban** permet de vérifier si l'Iban est correct.

6.4 Menu Switching Bank

En France, les banques s'étaient engagées, sous l'égide de la FBF, à offrir un service de mobilité gratuit au client consommateur. Ce service s'applique aux PSP (Prestataire des Services de Paiements) Français et aux émetteurs dont le siège est situé en France.

6.4.1 Principles

Le nouveau service de mobilité bancaire permet la prise en compte des nouvelles coordonnées bancaires par les émetteurs de prélèvements et de virements 10 dans un délai maximum de 22 jours ouvrés. Le service de mobilité bancaire est basé sur le principe de notification automatisée des organismes émetteurs (de prélèvement ou de virement) des changements de domiciliation bancaire. Étant donné les délais de prise en compte, ce service nécessitera des dispositifs automatisés de transmission d'information entre les PSP d'arrivée, les PSP de départ et les PSP des émetteurs.

Dans ce mode de fonctionnement, le client peut se reposer complètement sur son PSP d'arrivée sans avoir à lui fournir la liste des émetteurs et sa référence client chez ces émetteurs. Les PSP doivent informer leurs clients du nouveau service de mobilité bancaire. Le service de mobilité bancaire est proposé gratuitement et sans condition par les PSP d'arrivée.

6.4.2 La cinématique globale

La figure 6.2 montre la cinématique globale de la partie mobilité bancaire

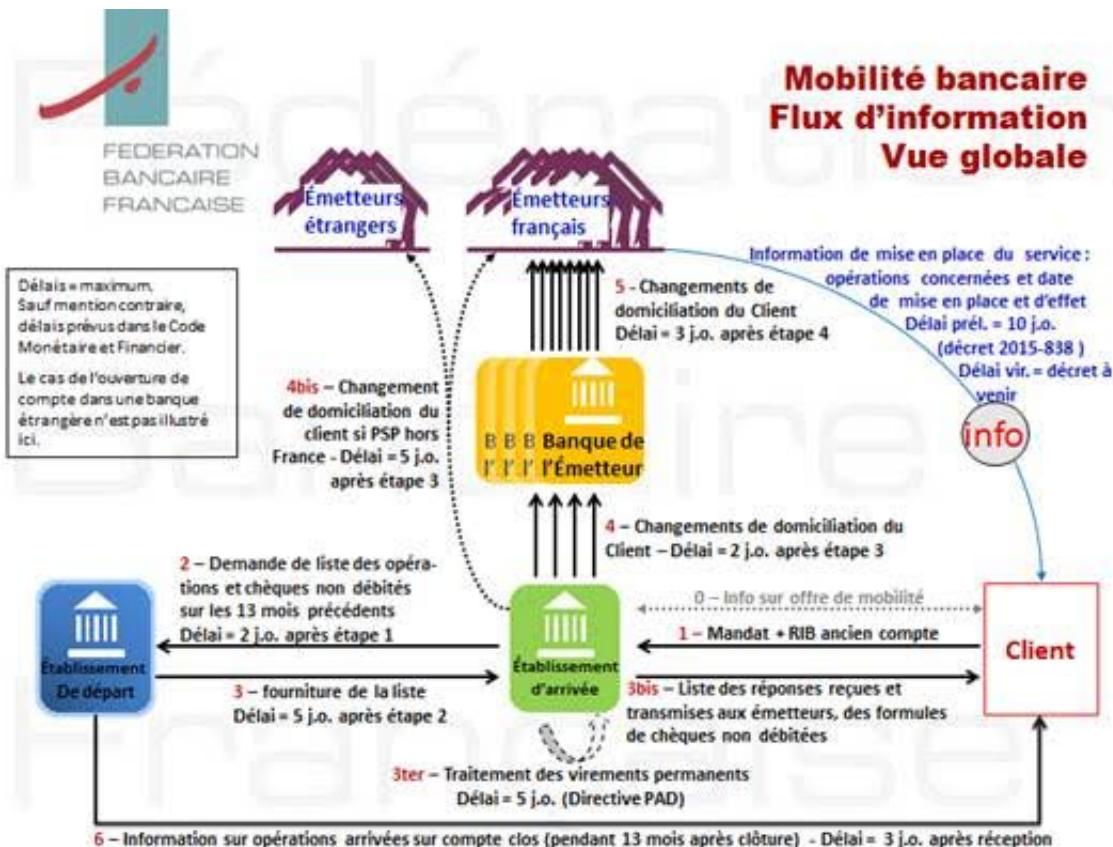


FIGURE 6.2 – Interface du menu documents - mode tablette

Etape 1

Le client ouvre un compte dans l'établissement d'arrivée. Celui-ci lui propose un service de mobilité bancaire pour transférer automatiquement ses opérations récurrentes sur ce nouveau compte. Le client donne mandat à l'établissement d'arrivée d'effectuer en son nom les formalités liées au changement de compte.

Etape 2

L'établissement d'arrivée adresse à l'établissement de départ un message par lequel il lui demande de lui envoyer les informations relatives aux opérations concernées sur les 13

derniers mois de fonctionnement du compte. L'établissement d'arrivée dispose de 2 jours ouvrés après signature du mandat pour émettre ce message.

Etape 3

L'établissement de départ recense les opérations concernées et adresse à l'établissement d'arrivée un ou plusieurs messages comprenant les informations demandées. L'établissement de départ peut aussi refuser la demande s'il détecte une incohérence. L'établissement de départ doit répondre à l'établissement d'arrivée dans un délai de 5 jours ouvrés suivant la réception de sa demande.

Etape 3 bis

L'établissement d'arrivée reçoit la liste des opérations concernées, traite les informations reçues et informe son client. Le cas échéant, il met en place les virements permanents. Cette étape ne nous concerne pas

Etape 4

Pour ce qui est des virements et prélèvements reçus, l'établissement d'arrivée classe par compte émetteur et envoie à la banque de chacun de ces émetteurs un message comprenant les opérations de l'émetteur et mentionnant le changement de domiciliation bancaire du client. La banque de l'émetteur doit recevoir ces informations au plus tard 2 jours ouvrés après que l'établissement d'arrivée a reçu la réponse de l'établissement de départ.

Etape 4 bis

Si l'établissement d'arrivée ne peut pas joindre l'émetteur via sa banque parce que celle-ci est située à l'étranger et n'est donc pas partie prenante du mécanisme français de mobilité bancaire, il se met en relation avec son client afin de l'informer de cette situation et de permettre le traitement de ce cas spécifique. Cette étape ne nous concerne pas.

Etape 5

La banque de l'émetteur route les modifications de domiciliation bancaire à l'émetteur. Elle dispose de 3 jours ouvrés pour ce faire.

Etape 6

L'émetteur procède au changement de domiciliation bancaire dans ses bases et informe le client dans un délai défini par le Code Monétaire et Financier.

6.4.3 Le périmètre

Le module de mobilité bancaire sera proposé à des clients (PID) ayant un compte tenu à la Société Générale, le suivi de ce compte étant effectué sur WEBCLEAR.

Les opérations concernées sont :

1. Prélèvements récurrents reçus par le client
 - Mandats non caducs
 - Mandats non révoqués
 - Créditeurs absents de liste noire de l'établissement de départ
 - Les prélèvements ponctuels (one-off) ne sont pas pris en compte
2. Virements récurrents reçus du client
 - Virements SEPA et non-SEPA (internationaux)
 - La récurrence s'entend comme la réception d'au moins 2 opérations d'un même compte du donneur d'ordre au cours des 13 derniers mois.
3. Virements permanents émis
 - Dossiers de virements permanents
 - Montant, périodicité, bénéficiaire
4. Formules de chèques non débitées
 - Issues de chéquiers utilisés sur les 13 derniers mois Issues de chéquiers remis au client au cours des 13 derniers mois et non utilisés

6.4.4 Les intervenants

-Client : personne physique n'agissant pas pour des besoins professionnels, qui souscrit au service de mobilité bancaire auprès de l'établissement d'arrivée.

-Etablissement d'arrivée : établissement dans lequel le client ouvre un nouveau compte et qui offre le service de mobilité bancaire.

-Etablissement de départ : établissement dans lequel le client détient un compte dont il souhaite transférer les opérations vers son nouveau compte dans l'établissement d'arrivée.

-Banque de l'émetteur : banque du créancier pour les prélèvements reçus sur le compte ou du débiteur pour les virements reçus sur le compte.

-Emetteur : le créancier qui émet des prélèvements vers le compte du client, ou le donneur d'ordre qui émet des virements vers le compte du client

6.4.5 Les échanges de flux entre les intervenants

Le client va voir un établissement bancaire afin d'y ouvrir un nouveau compte en lieu et place d'un compte ouvert dans un autre établissement. Il donne mandat à ce nouvel établissement afin qu'il effectue toutes les démarches liées au changement de domiciliation.

— Etablissement d'arrivée

- saisie de mandat à adresser à la Banque d'origine (saisie des informations pour l'émission du flux 2)
- consultation de la réponse reçue (consultation des informations du flux 3 reçu : listes des opérations à transférer de la banque de départ vers la banque d'arrivée)
- consultation des messages à destination des banques d'émetteurs (émission des flux 4) qui auront été produits par le socle mobilité bancaire à partir des flux 3 reçus de l'établissement de départ

— Etablissement de départ

- Consultation du mandat reçu (consultation des informations du flux 2 reçu)
- Saisie des opérations à restituer à la banque d'accueil (saisie des informations pour l'émission du flux 3)

— Banque de l'émetteur

- Consultation des messages reçus à destination d'émetteurs du PID (consultation des informations du flux 4 reçu)

6.4.6 Mise en oeuvre

6.4.6.1 Identification des fonctions

Mise à disposition des PID d'un module Mobilité Bancaire dans WEBCLEAR permettant au PID :

1. de saisir et consulter les informations de mobilité
2. de traiter les demandes dans les délais imposés (informations de mobilité en temps réel)
3. d'effectuer des recherches et d'en extraire les données (xls, PDF)
4. d'être notifié par e-mail dès mise à disposition des flux reçus.

La figure 6.3 explique les différentes fonctionnalités

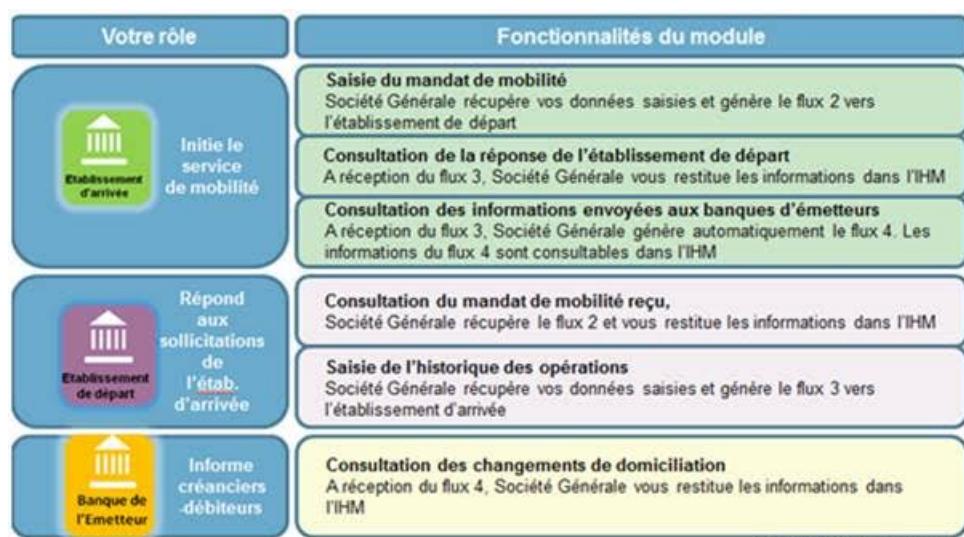


FIGURE 6.3 – Les différentes fonctionnalités des modules

6.4.6.2 Description détaillée des fonctions

Fonctions de saisie de mandat

L'acteur ici est la banque d'arrivée, la figure 6.4 montre le processus en quatre étapes qui décrit la saisie d'un mandat de mobilité en tant que banque d'arrivée.



FIGURE 6.4 – Cinématique de saisie du mandat

Saisie des opérations à restituer L'acteur ici est la banque de départ, la figure 6.5 montre le processus en quatre étapes qui décrit la saisie des opérations d'un mandat de mobilité en tant que banque de départ.

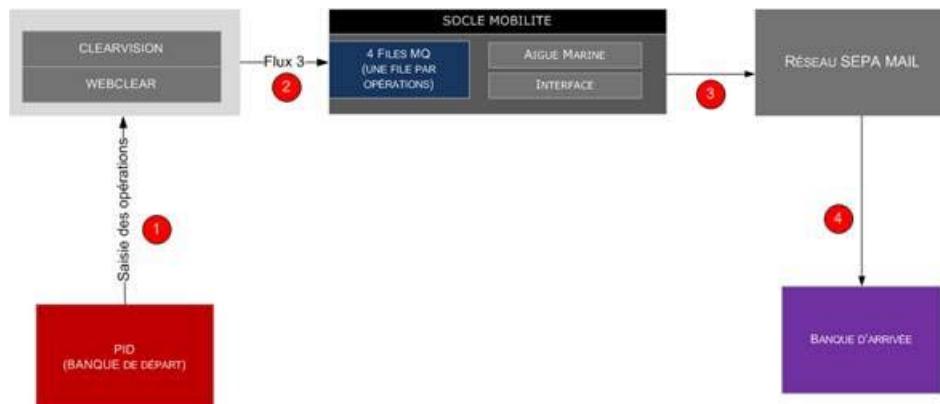


FIGURE 6.5 – Cinématique de saisie d'opération

Consultation du mandat flux 2 reçu L'acteur ici est la banque de départ, la figure 6.6 montre le processus en deux étapes qui décrit la consultation d'un mandat de mobilité en tant que banque de départ.

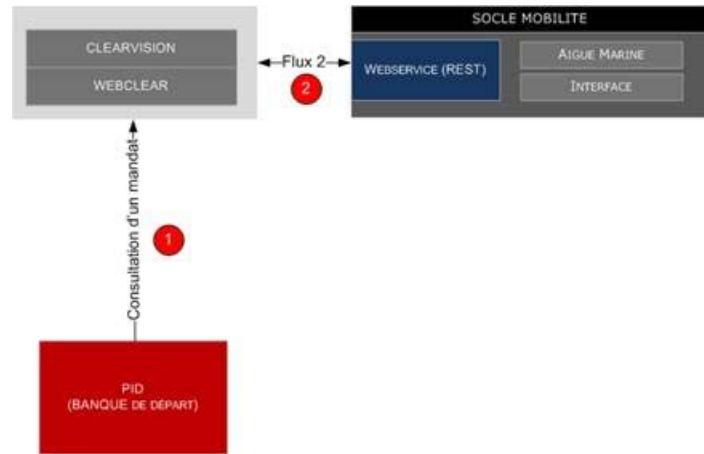


FIGURE 6.6 – Cinématique de Consultation du mandat flux 2 reçu

Consultation du flux 3 reçu L'acteur ici est la banque d'arrivée, la figure 6.7 montre le processus en deux étapes qui décrit la consultation des opérations d'un mandat de mobilité en tant que banque d'arrivée.

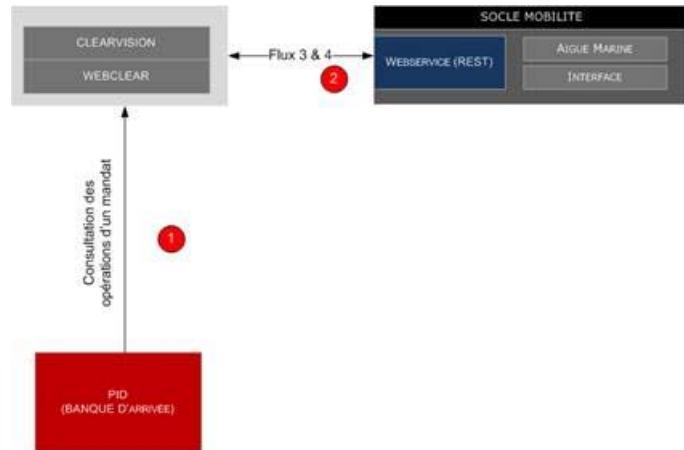


FIGURE 6.7 – Cinématique de Consultation du flux 3 reçu

Consultation du flux 4 L'acteur ici est la banque de l'émetteur, la figure 6.8 montre le processus en deux étapes qui décrit la consultation des opérations d'un mandat de mobilité en tant que banque de l'émetteur.

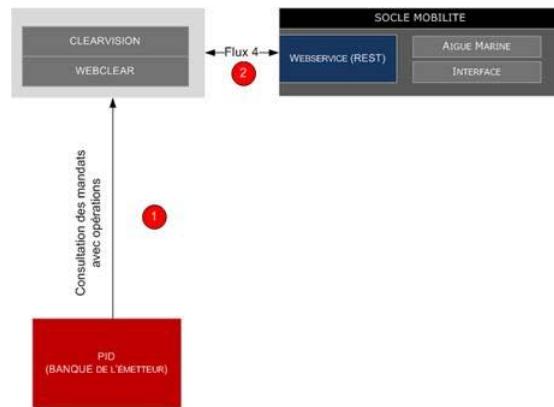


FIGURE 6.8 – Cinématique de Consultation du flux 4

6.4.7 Implémentation

La figure 6.9 est une imprime écran de la page de synthèse du menu Switching bank

L'écran affiche la page de synthèse du menu "Switching bank" de l'application "WEBCLEAR".

En haut, le menu horizontal comprend : Home, Orders, Operations, Balances, Documents, Tools, et **Switching bank** (souligné).

En haut à droite, il y a des icônes pour messagerie, calendrier, et utilisateur, ainsi que les horaires de plusieurs villes : PAR 15:08, NYC 9:08, TOK 22:08, SYD 23:08, GMT 13:08.

Le titre de la page est "SYNTHESIS".

Le contenu principal est divisé en trois sections :

- Etablissement d'arrivée :** 4 dossiers en cours.
- Etablissement de départ :** 22 dossiers en cours.
- Etablissement de l'émetteur :** 13 messages.

En dessous de ces sections, il y a une table intitulée "Etablissement de départ" :

Date Butoir	Nombre de dossiers concernés
10/02/2015	2 dossier(s) en cours
15/02/2015	7 dossier(s) en cours
22/02/2015	3 dossier(s) en cours
01/03/2015	4 dossier(s) en cours
15 /03/2015	7 dossier(s) en cours
30/03/2015	1 dossier(s) en cours
	6 dossier(s) en cours

FIGURE 6.9 – Interface de la page de synthèse

6.5 Conclusion

A la fin de ce sprint, nous avons détaillé 2 menus importants dans notre application qui sont le menu Tools et le menu Switching bank.

Chapitre 7

Conclusion générale et perspectives

Le travail présenté dans ce rapport a été réalisé au sein de la Société Générale dans le cadre de notre projet de fin d'études. Il consiste à la Refonte totale et en marque blanche de la plateforme WEBCLEAR tout en mettant en place l'approche Responsive Design afin d'adapter l'application aux nouveaux usages et aux supports (Web, Mobile, Tablette)

la nouvelle version de WEBCLEAR contient un nouveau menu qui est le menu Mobilité bancaire, il recense les fonctionnalités permettant la mise en place du transfert automatisé et transparent vis à vis du client des domiciliations bancaires ainsi que des opérations (prélèvements, virements récurrents etc) sur son compte.

Pour atteindre ces objectifs, nous avons dégagé les étapes principales de notre projet.

Nous avons fait une brève étude et critique de l'existant dans l'entreprise.

A travers l'analyse des besoins, nous avons identifié les différents acteurs et les fonctionnalités du système.

La phase de conception nous a permis de détailler les différents scénarios du système.

Nous avons adopté la méthodologie de développement Agile Scrum lors de la phase de développement pour avoir des versions partielles testables et livrables.

Enfin, ce projet nous a donné l'occasion d'intégrer le monde professionnel et aussi d'approfondir nos connaissances en développement web. Les majeurs difficultés rencontrées lors de la réalisation de ce projet étaient

- La gestion de la complexité technique de l'application ainsi que son intégration et

son interaction avec les autres applications du système d'information de la Société Générale (comme CLEARVISION, SOCLE, ACTIF, etc.).

- Acquérir et s'imprégner de la richesse fonctionnelle de l'application afin de mieux répondre aux besoins exigeants du métier.
- Respecter des délais de livraison très courts avec des sprints de trois semaines chacun tout en assurant une qualité optimale du livrable.

Comme perspectives pour notre projet, plusieurs travaux peuvent être accomplis :

- La mise en place de l'authentification forte, l'envoi d'un fax avec signature habilitée pour garantir le maximum de sécurité.
- L'ajout d'un système de notification pour notifier le client lors des opérations critiques effectuées sur son compte.
- Simplifier encore l'expérience utilisateur lors de la navigation dans certains certains menus en diminuant le nombre de clics ainsi que l'imbrication des sous menus.
- Réduire et améliorer le TIME TO MARKET c'est à dire le temps nécessaire entre l'expression du besoin client pour une nouvelle fonctionnalité et l'implémentation et la mise en production de cette fonctionnalités.

Bibliographie

- [1] Société Générale : [//mysocietegenerale.safe.socgen](http://mysocietegenerale.safe.socgen) [Site intranet de la Société Générale, consulté le 03/10/2016]
- [2] WEBCLEAR : [//www.journaldunet.com/99juil/990716webclear.shtml](http://www.journaldunet.com/99juil/990716webclear.shtml) [Consulté le 04/10/2016]
- [3] scrum : [//www.agiliste.fr/guide-de-demarrage-scrum/](http://www.agiliste.fr/guide-de-demarrage-scrum/) [Consulté le 04/10/2016]
- [4] Oracle : [//fadace.developpez.com/sgbdcmp/](http://fadace.developpez.com/sgbdcmp/) [Consulté le 06/10/2016]
- [5] Weblogic : [//www.journaldunet.com/solutions/0211/](http://www.journaldunet.com/solutions/0211/) [Consulté le 06/10/2016]
- [6] Spring : [//wayofcode.fr/implmentation-de-services-web-rest-avec-le-framework-spring/](http://wayofcode.fr/implmentation-de-services-web-rest-avec-le-framework-spring/) [Consulté le 06/10/2016]
- [7] Eclipse : [//eclipse.org/mars/](http://eclipse.org/mars/) [Consulté le 07/10/2016]
- [8] Notepad++ : [//notepad-plus-plus.org/fr/](http://notepad-plus-plus.org/fr/) [Consulté le 07/10/2016]
- [9] SQL Developer : [//www.oracle.com/technetwork/developer-tools/sql-developer/overview/index-097090.html](http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index-097090.html) [Consulté le 07/10/2016]
- [10] Start UML : [//staruml.io/](http://staruml.io/) [Consulté le 07/10/2016]
- [11] Jenkins : [//jenkins.io/](http://jenkins.io/) [Consulté le 07/10/2016]
- [12] Maven : [//maven.apache.org/](http://maven.apache.org/) [Consulté le 07/10/2016]
- [13] Ant : [//ant.apache.org/](http://ant.apache.org/) [Consulté le 07/10/2016]

Glossaire

API : Une interface de programmation(Application Programming Interface ou API)est une interface fournie par un programme informatique. Elle permet l'interaction des programmes les uns avec les autres, du point de vue technique une API est un ensemble de fonctions, procédures ou classes mises à disposition par une bibliothèque logicielle, un système d'exploitation ou un service.

BACK-END : est un terme désignant un étage de sortie d'un logiciel devant produire un résultat, dans une architecture modèle-vue-contrôleur le modèle et le contrôleur font partie du back-end (représentation et traitement des données en arrière-plan).

BNP : la Banque Nationale de Paris

BPML : Business Process Modeling Language) est un langage basé sur XML permettant de modéliser des processus métier.

Flashback : Retour en arrière.

FRONT-END : Il s'agit des éléments du site que l'on voit à l'écran et avec lesquels on peut interagir. Ces éléments sont composés de HTML, CSS et de Javascript contrôlés par le navigateur web de l'utilisateur.

HTTP : HyperText Transfer Protocol.

ISO : L'Organisation internationale de normalisation OIN(International Organization for Standardization) est un organisme de normalisation international composé de représentants d'organisations nationales de normalisation.

LDAP : Lightweight Directory Access Protocol (LDAP) est à l'origine un protocole permettant l'interrogation et la modification des services d'annuaire. Ce protocole repose sur

TCP/IP. Il a cependant évolué pour représenter une norme pour les systèmes d'annuaires.

MGA : Architecture Multi-Générationnelle.

MT103 : Message Type réservé en principe aux clients non banquiers).

MT200 : Message Type réservé à un client pour le transfert d'institutions financières pour son propre compte.

MT202 : Message Type réservé aux transferts de l'institution financière générale.

MT202COV : Message Type réservé aux transferts de l'institution financière.

MT210 : Message Type qui signifie un avis de réception.

pdf : Le Portable Document Format, est un langage de description de pages créé par la société Adobe Systems. Sa spécificité est de préserver la mise en forme d'un fichier ? polices d'écritures, images, objets graphiques, etc. ? telle qu'elle a été définie par son auteur, et cela quels que soient le logiciel, le système d'exploitation et l'ordinateur utilisé.

PID : Product Information Document.

PSP : Prestataire de Services de Paiement, est une entreprise agréée pour offrir des services de paiement. Les prestataires de services de paiement (PSP) sont : les établissements de crédit qui peuvent effectuer des opérations de banque, émettre et gérer de la monnaie électronique et fournir des services de paiement, les établissements de monnaie électronique qui peuvent émettre et gérer de la monnaie électronique et fournir des services de paiement, les établissements de paiement qui peuvent fournir des services de paiement.

RDI : Référentiel de l'International.

RLSS : Row Level Storage Security.

SEPA : Espace unique de paiement en euros (Single Euro Payments Area), s'inscrit dans le prolongement de l'adoption de la monnaie unique au sein de l'Union européenne. Son objectif est de créer une gamme de moyens de paiement en euros, afin de faciliter les échanges entre les pays européens.

SI : Un système d'Information (SI) est un ensemble organisé de ressources (matériels, logiciels, personnel, données et procédures) qui permet de regrouper, de classifier, de traiter et de diffuser de l'information sur un environnement donné.

SSII : Société de Services et d'Ingénierie en Informatique) est une société experte dans le domaine des nouvelles technologies et de l'informatique. Elle peut englober plusieurs métiers (conseil, conception et réalisation d'outils, maint.

SSL : Secure Socket Layer, c'est un système qui permet d'échanger des informations entre deux ordinateurs de façon sûre

SWIFT : Society for Worldwide Interbank Financial Telecommunication, a pour objet de gérer un réseau international de télé-transmission des messages entre les banques adhérentes, dans le monde entier. Swift est un réseau de télé-transmission par lequel transitent des ordres de paiement entre deux banques qui entretiennent de manière bilatérale des liens de correspondants, mais aussi des ordres de transferts de fonds de la clientèle des banques, des ordres d'achat et de vente de valeurs mobilières, des opérations d'ouverture de crédits documentaires.

Unicode : Unicode est un standard informatique qui permet des échanges de textes dans différentes langues, à un niveau mondial.

UML : C'est l'acronyme anglais pour(Unified Modeling Language). On le traduit par "Langage de modélisation unifié". La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc. appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique.

XML : Le XML ou eXtensible Markup Language est un langage informatique de balisage générique qui permet de décrire des données à l'aide de balises et de règles que l'on peut personnaliser.