



INSTITUT SUPÉRIEUR DES ÉTUDES TECHNOLOGIQUES DE NABEUL  
المعهد العالي للدراسات التكنولوجية بنابل

## Département **Technologie de l'Informatique**

Code : DS/CR/22

### PARCOURS

### **DEVELOPPEMENT DES SYSTEMES D'INFORMATION**

## **PROJET DE FIN DE PARCOURS**

### INTITULE

### **DEVELOPPEMENT D'UNE APPLICATION BANQUE DIGITALE**

### ORGANISME D'ACCUEIL

**BFI**

### REALISE PAR

**CHARGUI RAMI**

### ENCADRANT ACADEMIQUE

**GUERBOUJ SONIA**

### ENCADRANT PROFESSIONNEL

**ASSAS ANIS**

---

**ANNEE UNIVERSITAIRE  
2021/2022**

<b>Nom de l'entreprise</b>	BFI
<b>Ville</b>	Djerba, Houmet Souk
<b>Nom de l'encadrant industriel</b>	Assas Anis
<b>Date début et date fin du stage</b>	Du 1 <sup>er</sup> février jusqu'à 1 <sup>er</sup> mai
<b>Nombre de jours effectifs passés dans les locaux de l'entreprise</b>	1 mois
<b>Les principales tâches réalisées durant le stage (maximum 3)</b>	<ul style="list-style-type: none"> <li>• Conception des interfaces.</li> <li>• Modélisation des diagrammes UML.</li> <li>• Développement des applications Web et mobile.</li> </ul>
<b>Le stage contient-il une période d'observation ? Si oui combien de jours ça a duré.</b>	Non
<b>Le stage contient-il une partie conception ? Si oui combien de jours ça a pris</b>	Oui, 12 jours.
<b>Le stage contient-il une partie développement ? si oui combien de jours ça a pris</b>	Oui, 3 mois.
<b>Quels sont les prérequis pour ce stage en termes de compétences techniques ?</b>	<ul style="list-style-type: none"> <li>• Langages de programmation : Dart, Java et JavaScript .</li> <li>• Framework: Flutter , Spring Boot et ReactJS</li> </ul>
<b>Les outils et logiciels utilisés durant le stage</b>	Visual Studio Code, MySQL , Postman , JIRA, StarUML , adobe XD.
<b>Le matériel et la machinerie (autre que les PCs) utilisés durant le stage</b>	Aucun.
<b>Méthodologies utilisées durant le stage</b>	Scrum
<b>Les deux principales qualifications-compétences comportementales acquises à travers ce stage</b>	Gestion du temps et gestion du stress.
<b>Les deux principales qualifications-compétences techniques acquises à travers ce stage</b>	Dart, Flutter et ReactJS



## **Fiche synthétique**

# Dédicaces

*Je dédie ce travail à toute ma famille :*

*Mes parents*

*Vous m'avez soutenu toute ma vie, je vous remercie de tout mon cœur pour les sacrifices et les efforts que vous faites.*

*Mon frère et ma sœur*

*Je vous suis reconnaissant pour tous les conseils qui vous me donnez.*

*J'ai toujours espéré vous rendre heureux et fiers de moi.*

*Je dédie aussi ce travail à tous mes amis, en souvenir des beaux moments qu'on a passés ensemble.*

*Je vous remercie pour vos encouragements et votre aide précieuse.*

***Rami Chargui***

# **Remerciements**

*Je tiens, avant de présenter mon travail, à exprimer ma grande gratitude envers les personnes qui m'ont apporté leurs soutiens pour accomplir ce travail.*

*Je tiens à remercier mon encadrante à l'ISET, Madame **Sonia Guerbouj** pour le temps qu'elle a consacré à l'encadrement et le suivi de ce travail ; les conseils qu'elle m'a prodigués après ses minutieuses lectures et pour les réunions qui ont rythmées les différentes étapes de la rédaction de ce rapport.*

*Je remercie également mon encadrant externe, Monsieur **Anis Assas** pour l'expérience enrichissante et pleine d'intérêt qu'il m'a fait vivre durant la période du stage et pour tous les conseils et les informations qu'il m'a prodigués. Mes sincères remerciements s'adressent aux employés de la société BFI pour leur accueil et leur coopération lors de mon stage.*

*En plus, j'exprime ma reconnaissance bien évidemment à tous les enseignants qui m'ont formé durant mon cursus universitaire à l'Institut Supérieurs des Etudes Technologiques de Nabeul*

*Enfin, mes respects sont adressés à tous les membres du jury pour l'évaluation de mon projet.*

# Sommaire

Introduction générale.....	1
Chapitre 1 : Contexte général du projet.....	2
Introduction .....	2
1. Présentation de la société.....	2
2. Etude de l'existant .....	2
2.1. Problématique.....	2
2.2. Description de l'existant.....	3
2.3. Critique de l'existant .....	4
2.4. Solution proposée .....	5
3. Méthodologie de développement .....	5
4. Choix Technologiques.....	6
4.1. Outils logiciels.....	6
4.2. Langages de programmation .....	7
4.3. Frameworks .....	8
4.4. Technologies .....	9
Conclusion.....	9
Chapitre 2 : Product Backlog .....	10
Introduction .....	10
1. Spécification des besoins.....	10
1.1. Besoins fonctionnels.....	10
1.2. Besoins non fonctionnels.....	13
2. Product Backlog .....	13
2.1. Identification des acteurs.....	13
2.2. Eléments du product backlog .....	14
2.3. Planning des sprints.....	19
Conclusion.....	19
Chapitre 3 : Sprint 1 – Configuration des comptes bancaires .....	20
Introduction .....	20
1. Backlog du sprint 1.....	20
2. Description fonctionnelle des histoires du sprint 1 .....	22
2.1. Diagrammes de cas d'utilisation.....	23

2.2.    Description textuelle des cas d'utilisations .....	25
3.    Description comportementale des histoires du sprint 1 .....	28
4.    Diagramme de classes du sprint 1 .....	34
5.    Review du sprint 1 .....	35
5.1.    Espace administrateur.....	35
5.2.    Espace client.....	39
Conclusion.....	43
 Chapitre 4 : Sprint 2 – Module transactions .....	44
Introduction .....	44
1.    Backlog du sprint 2.....	44
2.    Description fonctionnelle des histoires du sprint 2 .....	45
2.1.    Diagrammes de cas d'utilisation.....	45
2.2.    Description textuelle des cas d'utilisation.....	47
3.    Description comportementale des histoires du sprint 2.....	50
4.    Diagramme de classes du sprint 2 .....	55
5.    Review du sprint 2.....	55
5.1.    Espace administrateur.....	56
5.2.    Espace client.....	58
Conclusion.....	60
 Chapitre 5 : Sprint 3 – Module communication .....	61
Introduction .....	61
1.    Backlog du sprint 3.....	61
2.    Description fonctionnelle des histoires du sprint 3 .....	63
2.1.    Diagrammes de cas d'utilisation.....	63
2.2.    Description textuelle des cas d'utilisation.....	66
3.    Description comportementale des histoires du sprint 3 .....	68
4.    Diagramme de classes du sprint 3 .....	71
5.    Review du sprint 3 .....	71
5.1.    Espace administrateur.....	71
5.2.    Espace client.....	74
Conclusion.....	77
 Conclusion générale .....	78
Webographie .....	79

# Liste des figures

Figure 1: Logo de la société BFI .....	2
Figure 2: Application Attijari Mobile Tunisie.....	3
Figure 3:Application STB Bank.....	4
Figure 4: Méthodologie agile : Scrum.....	6
Figure 5:Diagramme de cas d'utilisation global du sprint 1 .....	23
Figure 6: Diagramme de cas d'utilisation détaillé de l'administrateur au sprint 1 .....	24
Figure 7:Diagramme de cas d'utilisation détaillé du client au sprint 1.....	25
Figure 8: Diagramme de séquence du cas d'utilisation « S'inscrire » .....	29
Figure 9: Diagramme de séquence du cas d'utilisation « S'authentifier » .....	30
Figure 10 : Diagramme de séquence du cas d'utilisation « Modifier un type de compte ».....	31
Figure 11: Diagramme de séquence du cas d'utilisation « Rechercher un contact » .....	32
Figure 12:Diagramme de séquence du cas d'utilisation « supprimer un type de compte » .....	33
Figure 13:Diagramme de classe du sprint 1 .....	34
Figure 14:Interface d'authentification - admin .....	35
Figure 15: Interface Dashboard - admin.....	36
Figure 16: Interface liste des clients .....	36
Figure 17: Interface liste des comptes .....	37
Figure 18: Interface modifier type de compte .....	38
Figure 19: Interface supprimer un service.....	38
Figure 20: Interface authentification - client .....	39
Figure 21: Interface récupération de mot de passe .....	40
Figure 22:Interface inscription .....	40
Figure 23: Interface compte bancaire .....	41
Figure 24: Interface liste des contacts .....	42
Figure 25: Interface ajout d'un contact .....	42
Figure 26: Interface détail contact.....	43
Figure 27 : Diagramme de cas d'utilisation globale sprint 2 .....	46
Figure 28:Diagramme de cas d'utilisation détaillé du client.....	46
Figure 29:Diagramme de cas d'utilisation détaillé de l'administrateur .....	47
Figure 30 : Diagramme de séquence « Effectuer un retrait ».....	51
Figure 31 : Diagramme de séquence « Valider un dépôt » .....	52
Figure 32 : Diagramme de séquence « Annuler une transaction ».....	53

Figure 33 : Diagramme de séquence « Effectuer virements vers un autre compte » .....	54
Figure 34 : Diagramme de classes du sprint 2.....	55
Figure 35 : Interface Dashboard - admin - sprint 2 .....	56
Figure 36 : Interface liste des transferts .....	57
Figure 37 : Interface validation d'un dépôt.....	57
Figure 38 : Interface dépôt et retrait .....	58
Figure 39 : Interface virement.....	59
Figure 40:Interface historique .....	59
Figure 41 : Interface détails d'une opération .....	60
Figure 42: Diagramme de cas d'utilisation globale du sprint 3 .....	63
Figure 43: Diagramme de cas d'utilisation détaillé de l'administrateur.....	64
Figure 44: Diagramme de cas d'utilisation détaillé du client.....	65
Figure 45 : Diagramme de séquence « Consulter un rendez-vous » .....	68
Figure 46 : Diagramme de séquence « Envoyer un fichier » .....	69
Figure 47: Diagramme de séquence « Télécharger un fichier ».....	70
Figure 48: Diagramme de classe sprint 3 .....	71
Figure 49 : Interface réclamation - admin .....	72
Figure 50 : Interface liste des fichiers - admin .....	72
Figure 51 : Interface rendez-vous - admin .....	73
Figure 52 : Interface messagerie - admin .....	73
Figure 53: Interface menu - client .....	74
Figure 54: Interface réclamation – client.....	74
Figure 55 : Interface fichier - client.....	75
Figure 56:Interface rendez-vous - client.....	76
Figure 57: Interface messagerie - client .....	76
Figure 58 : Interface notification.....	77

# Liste des tableaux

Tableau 1: Product Backlog .....	14
Tableau 2 :Planning des sprints.....	19
Tableau 3 :Backlog du sprint 1.....	20
Tableau 4 : Description textuelle de cas d'utilisation "S'authentifier".....	26
Tableau 5:Description textuelle du cas d'utilisation "S'inscrire" .....	26
Tableau 6 : Description textuelle de cas d'utilisation "Modifier un type de compte" .....	27
Tableau 7:Description textuelle de cas d'utilisation « Chercher un contact ».....	27
Tableau 8 :Description textuelle de cas d'utilisation « Supprimer un type de compte » .....	28
Tableau 9: Backlog du sprint 2.....	44
Tableau 10: Description textuelle du cas d'utilisation « Valider un dépôt » .....	47
Tableau 11: Description textuelle du cas d'utilisation « Effectuer un retrait ».....	48
Tableau 12: Description textuelle du cas d'utilisation « Annuler une transaction».....	49
Tableau 13: Description textuelle du cas d'utilisation « Effectuer un virement vers un autre compte»	49
Tableau 14: Backlog du sprint 3.....	61
Tableau 15:Description textuelle du cas d'utilisation « Consulter un rendez-vous ».....	66
Tableau 16:Description textuelle du cas d'utilisation « Envoyer un fichier » .....	66
Tableau 17:Description textuelle du cas d'utilisation « Télécharger un fichier ».....	67
Tableau 18:Description textuelle du cas d'utilisation « Consulter une réclamation ».....	67

## Introduction générale

Le marché des technologies a connu une énorme évolution ces dernières années et avec l'intégration d'internet il y'a eu des changements sur la façon et la rapidité d'accès aux informations. Plusieurs secteurs se sont adaptés à ce changement et encore plus ont vu le jour pour satisfaire ces nouveaux clients.

Les banques représentent l'un des domaines que connaissent un nombre énorme de clients et qui souffrent pour les satisfaire. Ils doivent alors proposer leurs services d'une façon plus rapide et sécurisé mais aussi à distance afin de pouvoir éviter la congestion dans les agences.

Avec l'apparition des appareils mobiles, comme les smartphones qui ont connu une révolution technologique, il est devenu facile pour une banque de gérer ce conflit en proposant une application mobile capable de procurer des services indispensables rapidement et à tout moment.

Dans ce contexte, la société BFI nous a proposé de développer une application mobile qui propose de nombreuses fonctionnalités à aux clients d'une banque ainsi qu'une application Web pour administrer et gérer les services clientèle.

Ce rapport décrit les étapes de développement de notre projet. Il contient 5 Chapitres : Le premier chapitre sera consacré à présenter l'organisme d'accueil et à exposer l'analyse de l'existant et la méthodologie de développement. Dans le chapitre suivant, serons détaillées les diverses fonctionnalités dans le product backlog et la répartition des sprints. Les trois derniers chapitres présenteront le travail détaillé des trois sprints du projet, à savoir le backlog de chaque sprint, l'étude conceptuelle et la réalisation.

# Chapitre 1 : Contexte général du projet

## Introduction

Ce chapitre présente l'entreprise d'accueil ainsi que la problématique. Ensuite, il contient l'analyse de l'existant afin de proposer une solution adéquate. Enfin, il décrit la méthodologie de développement et liste tous les outils logiciels, les frameworks et les langages de programmations utilisés.

### 1. Présentation de la société

**BFI** est une entreprise tunisienne créée en 1994, spécialise dans l'édition et l'intégration de solution logicielles destinées aux banques.

Cette entreprise se consacre essentiellement au marché africain de la gestion bancaire dont elle occupe une place importante. Elle a, d'ailleurs avec une réputation de très haut niveau en possédant plus de 200 clients répartis dans différentes nations.



Figure 1: Logo de la société BFI

### 2. Etude de l'existant

#### 2.1. Problématique

D'habitude les gens doivent se rendre aux agences bancaires, se mettre en files d'attente et perdre un temps précieux pour effectuer leurs transactions retrait, versement, transfert, demande de crédit). Les besoins ayant évoluer aussi que les exigences des clients. De nos jours, du à l'évolution technologique et l'apparition des smartphone les gens sont devenus de plus en plus

attachés à leurs smartphones qui leurs permet de faire toutes sortes d'opérations et courses quotidiennes d'une façon plus facile et rapide. Peut-on alors apporter les services bancaires au smartphone des clients ?

## 2.2. Description de l'existant

Dans cette section nous allons étudier les différentes applications bancaires.

### ▪ Attijari Mobile Tunisie :

L'application de Attijari Bank offre ses services suivants :

- Consulter le solde et les mouvements des comptes en temps réel.
- Commander un chéquier
- Consulter les cours des devises.



Figure 2: Application Attijari Mobile Tunisie

### ▪ STB Bank :

L'application de STB Bank offre divers services tels que :

- Consulter les soldes et les mouvements de ses comptes.
- Parcourir l'historique de ses opérations de dépôt ou de retrait.
- Gérer ses cartes bancaires.

- Consulter les opérations monétique.
- Consulter ses placements.
- Consulter ses impayés.
- Réaliser un transfert d'argent pour ses proches.
- Planifier un rend – vous avec le chef d'agence ou le charge clientèle.
- Lancer une demande de chéquier ou de carte.
- Suivre les cours de devise et l'historique de TMM.
- Déposer une opposition de chèque ou carte bancaire.



Figure 3: Application STB Bank

### 2.3. Critique de l'existant

Il existe un bon nombre d'applications mobiles bancaires en Tunisie. Mais ces applications ne réussissent pas à proposer à la fois un grand nombre de fonctionnalités et une interface moderne et conviviale.

En fait, l'application de « STB » offre un grand nombre de fonctionnalité mais l'interface n'est pas bien organisée.

L'application « Attijari mobile Tunisie » propose une interface pas trop moderne et elle manque aussi de quelques fonctionnalités comme la géolocalisation des agences bancaires.

## 2.4. Solution proposée

Suite à ces études on a eu l'idée de développer une application mobile ; **BFI Bank** une application riche en termes de fonctionnalité, développée en utilisant des technologie rapide et moderne et avec des interfaces dynamiques modernes et faciles à utiliser.

Notre application permettra aux clients de :

- S'inscrire et gérer son profile
- Créer un ou plusieurs comptes bancaires et les gérer.
- Gérer la liste de ses contacts.
- Effectuer et consulter des virements bancaires.
- Effectuer et consulter des dépôts et retraits d'argent.
- Consulter l'historique des opérations effectuées
- Transférer des fichiers à la banque.
- Envoyer des réclamations au service clientèle.
- Communiquer par messagerie avec un agent de la banque.
- Prendre un rendez vous
- Recevoir et consulter les notifications

Tandis que l'application web permettra au responsable d'une banque de bénéficier des fonctionnalités suivantes :

- La gestion des agences
- La gestion des clients
- La gestion des comptes bancaires.
- La gestion des services et des types de comptes bancaires.
- La gestion des transactions bancaires effectuées par les clients.
- La gestion des réclamations des clients.
- La gestion des fichiers transférés par les clients.

## 3. Méthodologie de développement

Pour le développement de notre application on a opté pour scrum. C'est une méthode agile de gestion de projets informatiques privilégiant la communication, et facilitant les réorientations opportunes.

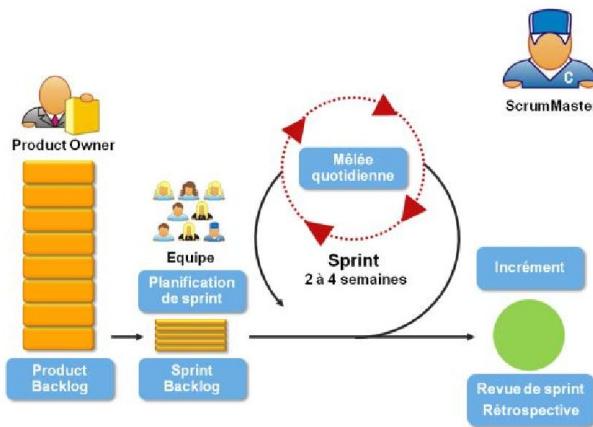


Figure 4: Méthodologie agile : Scrum

En effet, en tant que méthode agile, elle facilite l'expression des besoins et aide à l'acceptation du changement, elle est une approche itérative, qui est conduite dans un esprit collaboratif elle peut intégrer le client. Elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins des clients, pour monter en surface les besoins non prévisibles de notre projet. [1]

## 4. Choix Technologiques

### 4.1. Outils logiciels

On commence par décrire les outils logiciels utilisés tout au long de la conception et développement de notre projet.



- **Visual Studio Code**

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégrer.[2]

- **MySQL**

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il fait partie des logiciels de gestion de base de données les plus utilisés au monde.[3]

- **Postman**



Postman est un environnement de développement d'API qui aide les utilisateurs à créer, tester, documenter, surveiller et publier la documentation de leurs API.[4]

- **Adobe XD**



Un outil de conception d'expérience utilisateur basé sur des vecteurs pour les applications Web et les applications mobiles, développé et publié par Adobe Inc.[5]



- **Jira**

JIRA est une plateforme multifonction qui vise à faciliter la gestion de projet en aidant à suivre les tâches, identifier les blocages et partager des informations entre les membres d'une équipe.[6]



- **StarUML**

StarUML est un logiciel de modélisation open source qui prend en charge UML (Unified Modeling Language) pour la modélisation de systèmes et de logiciels. Il fournit onze types de diagrammes différents et il accepte la notation UML 2.0. Il soutient activement l'approche MDA (Model Driven Architecture).

- **Firebase Cloud Messaging**



FCM ou Firebase Cloud Messaging est la fonctionnalité de Firebase qui permet aux utilisateurs d'envoyer des notifications push sur des appareils Android, iOS et Web.[7]

## 4.2. Langages de programmation

- **Dart :**

Dart est un langage de programmation optimisé pour les applications sur plusieurs plateformes. Il est développé par Google et est utilisé pour créer des applications mobiles, de bureau, de serveur et web.[8]



- **Java :**

Java est un langage de programmation et une plate-forme informatique créée par Sun Microsystems. Utilisé par 9 millions de développeurs dans le monde, ce langage est nécessaire pour le fonctionnement de beaucoup d'applications et de sites Web. Il est la base de la plupart des applications en réseau : des jeux, des téléphones portables à Internet, du contenu web et des logiciels entreprises.[9]



- **JavaScript :**

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web.[10]



### 4.3. Frameworks

- **Flutter :**

Flutter est un kit de développement logiciel (SDK) d'interface utilisateur open-source basé sur Dart créé par Google. Il est utilisé pour développer des applications pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code.[11]



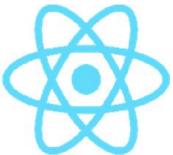
- **Spring boot :**

Spring Boot est un framework de développement JAVA. C'est une déclinaison du framework classique de Spring qui permet essentiellement de réaliser des microservices.[12]



- **ReactJS :**

React est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.[13]



## 4.4. Technologies

- **JWT :**

Un JSON Web Token est un access token (jeton d'accès) aux normes RFC 7519 qui permet un échange sécurisé de donnée entre deux parties. Il contient toutes les informations importantes sur une entité.[14]



- **WebSocket :**

L'API WebSocket est une technologie évoluée qui permet d'ouvrir un canal de communication bidirectionnelle entre un navigateur (côté client) et un serveur.[15]



## Conclusion

Tout au long de ce chapitre, nous avons évoqué le contexte du projet. Nous avons commencé par une présentation de l'organisme d'accueil qui a été suivie par une étude de l'existant, Ceci nous a permis de comprendre les besoins et d'envisager la solution la plus adéquate aux attentes du client. Dans le prochain chapitre, nous détaillerons la planification de notre projet.

# Chapitre 2 : Product Backlog

## Introduction

Ce chapitre est consacré à l'analyse et la spécification des besoins. Nous commençons par définir les besoins fonctionnels et non fonctionnels et par la suite définir notre product backlog.

### 1. Spécification des besoins

Cette partie comporte extraction des besoins fonctionnels et non fonctionnels.

#### 1.1. Besoins fonctionnels

Les besoins fonctionnels sont les fonctionnalités offertes par le système afin de répondre à une demande d'un utilisateur. Dans ce qui suit, nous présentons les besoins fonctionnels de notre projet qui comporte une application mobile et une application Web.

##### 1.1.1. Besoins fonctionnels de l'application mobile

Toute l'application mobile est disponible au client pour lui permettre après l'authentification par son login et mot de passe d'avoir accès aux différents services :

- **Inscription :**

L'application offre aux clients la possibilité de s'inscrire dans l'application afin de créer un profile utilisateur et pouvoir effectuer les opérations bancaires.

- **Gestion des comptes bancaire :**

Le client a le droit de créer un ou plusieurs comptes bancaires, de les consulter et demander à les fermer.

- **Gestion des contacts :**

Le client peut créer des contacts, les rechercher, les consulter, les modifier, les supprimer, les trier et il a la possibilité de les ajouter dans une liste de contact favoris,

- **Gestion des transactions :**

L’application offre aux clients la possibilité d’effectuer des transactions bancaires (retrait, dépôt et virement) entre ses comptes ou avec ses contacts, de consulter l’historique de ses activités bancaires et leurs états et il a la possibilité d’annuler une opération.

- **Gestion des réclamations :**

L’application offre la possibilité d’envoyer des réclamations au service clientèle et consulter leurs réponses.

- **Recevoir des messages :**

L’application permet aussi au client de communiquer avec le service clientèle par échange de messages.

- **Gestion des rendez-vous :**

L’application offre au client la possibilité de prendre un rendez-vous avec le service clientèle, le consulter, le modifier ou l’annuler.

- **Gestion des fichiers :**

Le client peut transférer des fichiers au service clientèle, les consulter, les télécharger ou les supprimer.

- **Réception et consultation des notifications :**

L’application permet au client de recevoir des notifications lors d’une activation de compte ou à la réponse d’une réclamation ainsi que lors des validations des transactions bancaires.

- **Gestion du profile utilisateur :**

L’application permet au client de consulter son profile et modifier ses données personnelles et son mot de passe.

### **1.1.2. Besoins fonctionnels de l’application web**

L’application web offre à l’administrateur d’effectuer les fonctionnalités suivantes :

- **Gestion des agences :**

L'administrateur a la possibilité de consulter la liste des agences ,ajouter une agence , il peut aussi modifier et supprimer les agences .

- **Gestion des comptes bancaires :**

L'administrateur a la possibilité de consulter la liste des comptes et leurs détails comme le type de compte, il peut aussi les activer et les désactiver.

- **Gestion des types des comptes bancaires :**

L'administrateur a le droit de créer des types de comptes bancaires, les consulter, les modifier et les supprimer.

- **Gestion des services des comptes bancaires :**

L'administrateur peut créer des services de comptes bancaires, les consulter, les modifier, les supprimer, et choisir les types des comptes associés au service.

- **Gestion des clients :**

Seul l'administrateur peut consulter la liste des clients, et voir les détails de chacun.

- **Gestion des transactions :**

L'application offre à l'administrateur la possibilité de consulter les détails des transactions (retrait, dépôt ou virement) de tous les comptes, et la possibilité des valider ou annuler une transaction.

- **Gestion des réclamations :**

L'administrateur peut consulter la liste des réclamations, répondre aux réclamations et les archiver,

- **Gestion des messages :**

L'application offre à l'administrateur la possibilité de discuter avec les clients par messagerie.

- **Gestion des rendez-vous :**

L'administrateur peut consulter la liste des rendez-vous, les confirmer ou les annuler.

- **Gestion des fichiers :**

L'application permet à l'administrateur de consulter la liste des fichiers transférés par les clients.

## 1.2. Besoins non fonctionnels

Un besoin non fonctionnel spécifie les contraintes liées à l'environnement et à l'implémentation, il exprime les exigences en matière de performance. Les besoins non fonctionnels sont indispensables et permettent l'amélioration de la qualité logicielle d'un projet.

- **La sécurité :** C'est un besoin très important à assurer lorsqu'on parle de banque. C'est pour cela qu'entre autre l'authentification nécessaire à l'accès, chaque opération nécessite une validation d'identité par un code secret.
- **La performance :** Etant donné le nombre d'utilisateur simultané, l'application doit être en mesure de satisfaire tous les clients sans diminuer les performances
- **L'ergonomie :** L'application doit offrir des interfaces conviviales, simples à manipuler et à comprendre
- **Compatibilité :** L'application doit être fonctionnelle sur n'importe quelle architecture et système d'exploitation.
- **Extensibilité :** L'application doit pouvoir être extensible telle que d'autres modules peuvent être ajoutés de manière transparente aux utilisateurs et sans altérer le fonctionnement actuel.

## 2. Product Backlog

### 2.1. Identification des acteurs

Notre application sera utilisée principalement par deux acteurs :

- Client : c'est l'utilisateur de l'application mobile et qui possède un compte bancaire.
- Admin : c'est un employé de la banque qui gère l'application web à travers laquelle elle administrer l'application mobile.

## 2.2. Eléments du product backlog

Le **backlog** est une liste de tâches priorisées définissant les caractéristiques d'un produit. Il est un des éléments fondamentaux de la méthodologie Scrum. Il s'agit de l'outil de travail principal du Product Owner qui se charge de recueillir les besoins auprès des parties prenantes et de les transformer en liste de fonctionnalités prêtes à être développées par l'équipe de développement.[16]

Dans cette partie, nous présenterons chaque « sprint » avec ses différentes « fonctionnalités » en indiquant les « priorités » selon leurs importances et les « user story » chacune avec son « ID » et son « Point d'histoire ».

Tableau 1: Product Backlog

Module	Fonctionnalité	ID	User story	Priorité	Story point
<b>Module configuration des comptes bancaires</b>	Gérer les comptes bancaires	1	En tant que admin je veux consulter la liste des tous les comptes bancaires et les activer et désactiver.	1	3
		2	En tant que client je veux créer des comptes et les consulter et demander à les fermer.		2
	Gérer les types des comptes bancaire	3	En tant que admin je veux créer des types des comptes bancaires les consulter, les modifier et les supprimer.	2	2
		4	En tant que client je veux choisir un type de compte bancaire.		1

	Gérer les services bancaires.	5	En tant que admin je veux créer des services bancaire, les consulter, les modifier et les supprimer.	3	3
		6	En tant que client je veux choisir un service.		2
	Gérer les clients.	7	En tant que admin je souhaite consulter la liste des comptes et pouvoir bloquer et débloquer un utilisateur. Je veux aussi pouvoir m'authentifier, consulter et modifier mon profile et récupérer mon mot de passe.	4	3
		8	En tant que client je veux créer un compte, s'authentifier et consulter mon profile et changer mes données personnelles.		4
	Gérer les contacts	9	En tant que client je veux pouvoir créer des contact, les consulter, les modifier et supprimer et ajouter des contacts favoris .	5	5
	Gérer les utilisateurs	10	En tant qu'admin je veux créer des comptes utilisateurs, les consulter et les supprimer.	6	4

	Gérer les agences	11	En tant que admin je veux créer des agences , les consulter, les supprimer et les modifier.	7	3
		12	En tant que utilisateur je veux consulter les agences et choisir une agence .		1
<b>Module transactions</b>	Gérer les retraits d'argent	1	En tant que admin je veux consulter la liste des retraits d'argent et pouvoir valider ou annuler une transaction.	8	3
		2	En tant que client je veux pouvoir effectuer des retraits d'argent, l'annuler et consulter l'historique de mes transactions et leurs états.		4
	Gérer les dépôts d'argent	3	En tant que admin je veux consulter la liste des dépôts d'argent et pouvoir valider ou annuler une transaction.	9	3
		4	En tant que client je veux pouvoir effectuer des dépôts d'argent, l'annuler et consulter l'historique de mes transactions et leurs états.		4

<b>Module Communication</b>	Gérer les virements bancaires	5	En tant que admin je veux consulter la liste des virements bancaires et pouvoir valider ou annuler une transaction.	10	3
		6	En tant que client je veux pouvoir effectuer des virements bancaires, l'annuler et consulter leurs historiques et leurs états.		4
		7	En tant que client je veux consulter l'historique de mes transactions et leurs états.		2
	Gestion des réclamations	1	En tant que admin je veux consulter la liste des réclamations, et pouvoir répondre et changer leurs états pour les archiver.	12	4
		2	En tant que client je veux pouvoir envoyer des réclamations au service clientèle et les consulter.		3
	Gestion des messages	3	En tant que client je veux discuter avec l'admin en envoyant et lisant des messages.	13	3
		4	En tant que admin je veux discuter avec le client en envoyant et lisant des messages.		2

	Gestion des rendez-vous	5	En tant que clients je veux planifier un rendez-vous le modifier ou l'annuler et consulter la liste de mes rendez-vous.	14	3
		6	En tant que admin je veux consulter la liste des rendez-vous, accepter un rendez-vous ou suggérer une autre date ou l'annuler.		3
	Gestion des fichiers	7	En tant que client je veux envoyer des fichiers a l'administrateur, les consulter , les modifier ou les supprimer	15	1
			En tant que admin je veux consulter la liste des fichiers, les télécharger et les supprimer.		1
	Gestion des notifications	8	En tant que client je veux recevoir des notifications dans l'application, par SMS et par email, les consulter et les supprimer.	16	5

## 2.3. Planning des sprints

Nous avons choisi de diviser le projet en 3 sprints, tel que dans chaque sprint sera développé un module de l'application. La durée et l'intitulé du module de chaque sprint sont indiqués dans le tableau

Tableau 2 :Planning des sprints

Sprints	Module	Durée
Sprint 1	Module configuration des comptes bancaires	4 semaines.
Sprint 2	Module transaction	5 semaines.
Sprint 3	Module communication	3 semaines.

## Conclusion

Dans ce chapitre nous avons défini la liste des besoins fonctionnels et non fonctionnels et les acteurs et de ce projet. Ensuite, nous avons présenté les éléments du backlog du produit, puis, la planification des sprints. Dans ce qui suit, nous détaillerons les étapes chaque sprint en commençant par le sprint 1.

# Chapitre 3 : Sprint 1 – Configuration des comptes bancaires

## Introduction

Dans ce chapitre, nous entamons le premier sprint qui traite le module consacré à la gestion des comptes. Pour cela, nous traiterons la description fonctionnelle **de** ce sprint **à** travers les diagrammes de cas d'utilisation, puis nous passerons à la description comportementale **à** l'aide des diagrammes de séquences. Enfin, nous présenterons le diagramme de classes juste avant les interfaces de réalisation.

## 1. Backlog du sprint 1

Dans le tableau ci-après, nous représentons le backlog de ce sprint avec les tâches à réaliser pour chaque user story.

Tableau 3 :Backlog du sprint 1

Fonctionnalité	ID	User Story	Tâches	Story point
<b>Gérer les comptes bancaire</b>	1	En tant que admin je veux consulter la liste des tous les comptes bancaires et les activer et désactiver.	- Créer la classe compte bancaire - Ajouter les méthodes consulter , supprimer et modifier , activer et désactiver dans la classe compte bancaire .	3
	2	En tant que client je veux créer des comptes et les consulter et demander à les fermer.	- Donner seulement le droit de créer , de consulter , et demander de fermeture .	2

<b>Gérer les types des comptes bancaire</b>	3	En tant que admin je veux consulter les types des comptes bancaires les modifier et les supprimer.	- Créer la classe type de compte - Ajouter les méthode consulter , supprimer et modifier a la classe type de compte . - Donner le droit de créer , de consulter , de modifier et de supprimer	2
	4	En tant que client je veux choisir un type de compte bancaire.	- Donner seulement le droit de choisir un type de compte bancaire .	1
<b>Gérer les services bancaires</b>	5	En tant que admin je veux créer des services bancaire, les consulter, les modifier et les supprimer.	- Créer la classe service bancaire . - Ajouter ses attribut - Ajouter les méthode de consultation , suppression et modification .	3
	6	En tant que client je veux choisir un service.	- Donner seulement le droit de choisir un service de compte bancaire .	2
<b>Gérer les clients</b>	7	En tant que admin je souhaite consulter la liste des comptes et pouvoir bloquer et débloquer un utilisateur. Je veux aussi pouvoir m'authentifier, consulter et modifier mon profile et récupérer mon mot de passe.	- Créer la classe utilisateur. - Ajouter les méthode de consultation , blocage et déblocage des utilisateur et les méthode de récupération de mot de passe et de modification du profile .	3

	8	En tant que client je veux créer un compte, s'authentifier et consulter mon profile et changer mes données personnelles.	- Donner le droit de s'authentifier, de consulter son profile et de modifier son compte .	4
<b>Gérer les utilisateurs</b>	10	En tant qu'admin je veux créer des comptes utilisateurs, les consulter et éditer mon profile .	- Donner le droit de créer des comptes utilisateur .	4
<b>Gérer les contacts</b>	9	En tant que client je veux pouvoir créer des contact, les consulter, les modifier et supprimer et ajouter des contacts favoris .	- Créer la classe contact. - Ajouter des attributs à cette classe - Ajouter les méthode de consultation , d'ajout , suppression et modification des la classe contact .	5
<b>Gérer les agences</b>	11	En tant que admin je veux créer des agences , les consulter, les supprimer et les modifier.	- Créer la classe agence . - Ajouter les méthode d'ajout , consultation , suppression et modification a la classe agence .	3
	12	En tant que utilisateur je veux choisir une agence	- Donner seulement le droit de choisir une agence .	1

## 2. Description fonctionnelle des histoires du sprint 1

Pour donner une vision globale du comportement fonctionnel du système, on utilise les diagrammes de cas d'utilisation. Dans cette partie, nous commençons par présenter le diagramme de cas d'utilisation global correspondant au sprint 1, par la suite, nous raffinerons par acteur

## 2.1. Diagrammes de cas d'utilisation

Un diagramme de cas d'utilisation est une représentation fonctionnelle du système, il permet de modéliser les attentes des utilisateurs afin de réaliser une bonne délimitation du système et également d'améliorer la compréhension de son fonctionnement. [17]

### 2.1.1. Diagramme de cas d'utilisation global du sprint 1

La figure 5 décrit les fonctionnalités offertes par le sprint 1 aux deux acteurs.

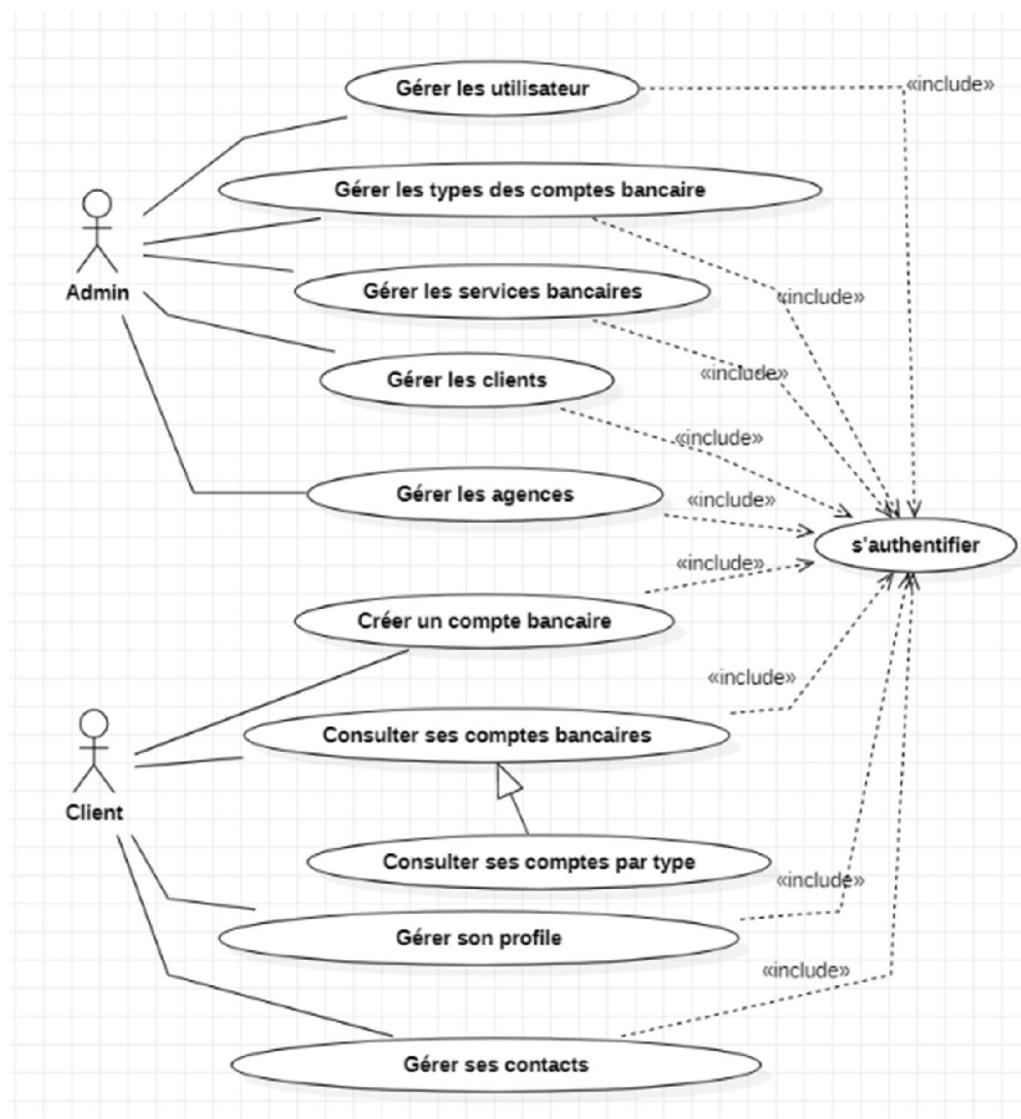


Figure 5:Diagramme de cas d'utilisation global du sprint 1

## 2.1.2. Diagramme de cas d'utilisation détaillé de l'administrateur

La figure 6 détaille les fonctionnalités offertes par le premier module à l'administrateur.

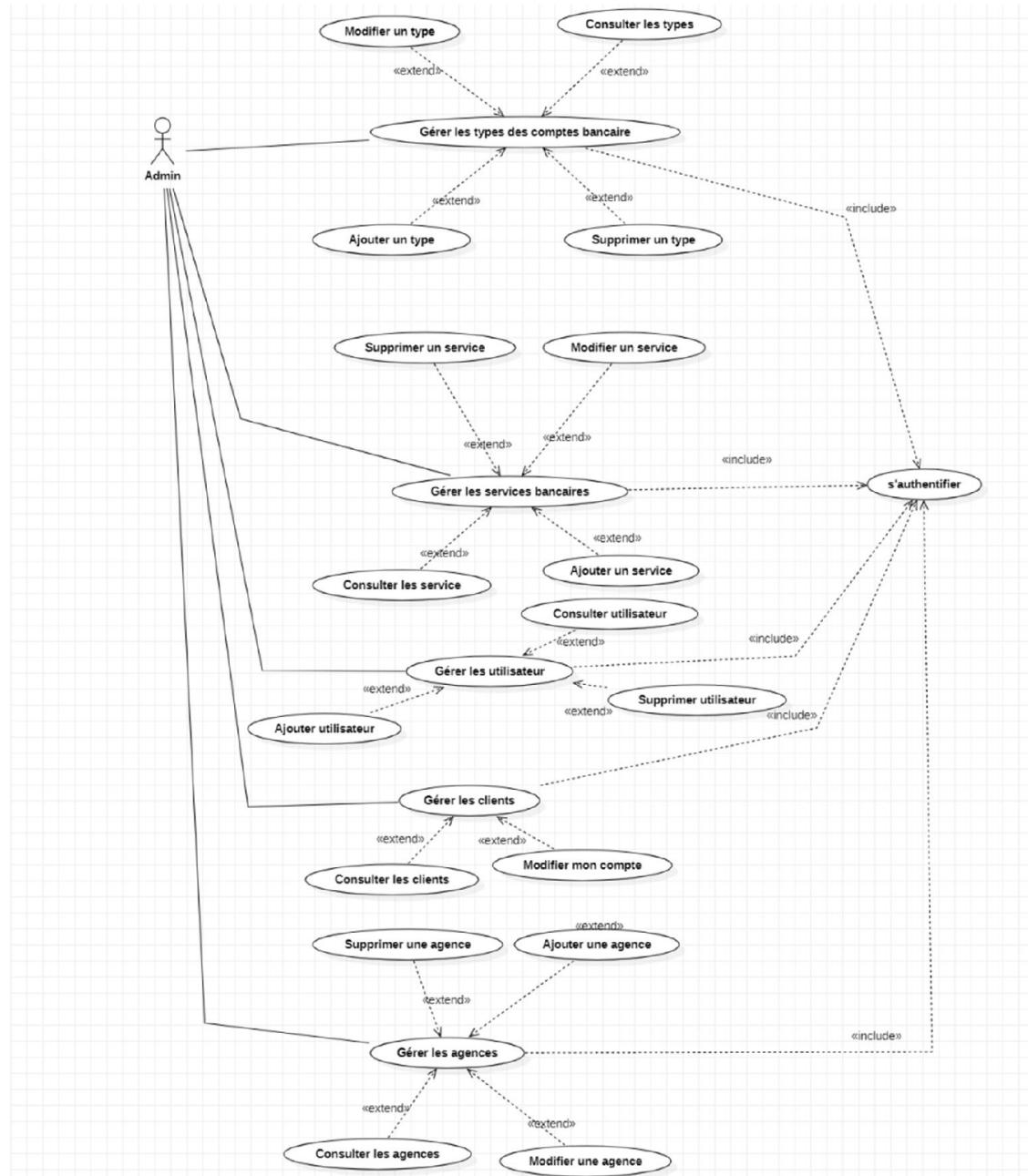


Figure 6: Diagramme de cas d'utilisation détaillé de l'administrateur au sprint 1

### 2.1.3. Diagramme de cas d'utilisation détaillé du client

La figure 7 détaille les fonctionnalités offertes par le premier module au client.

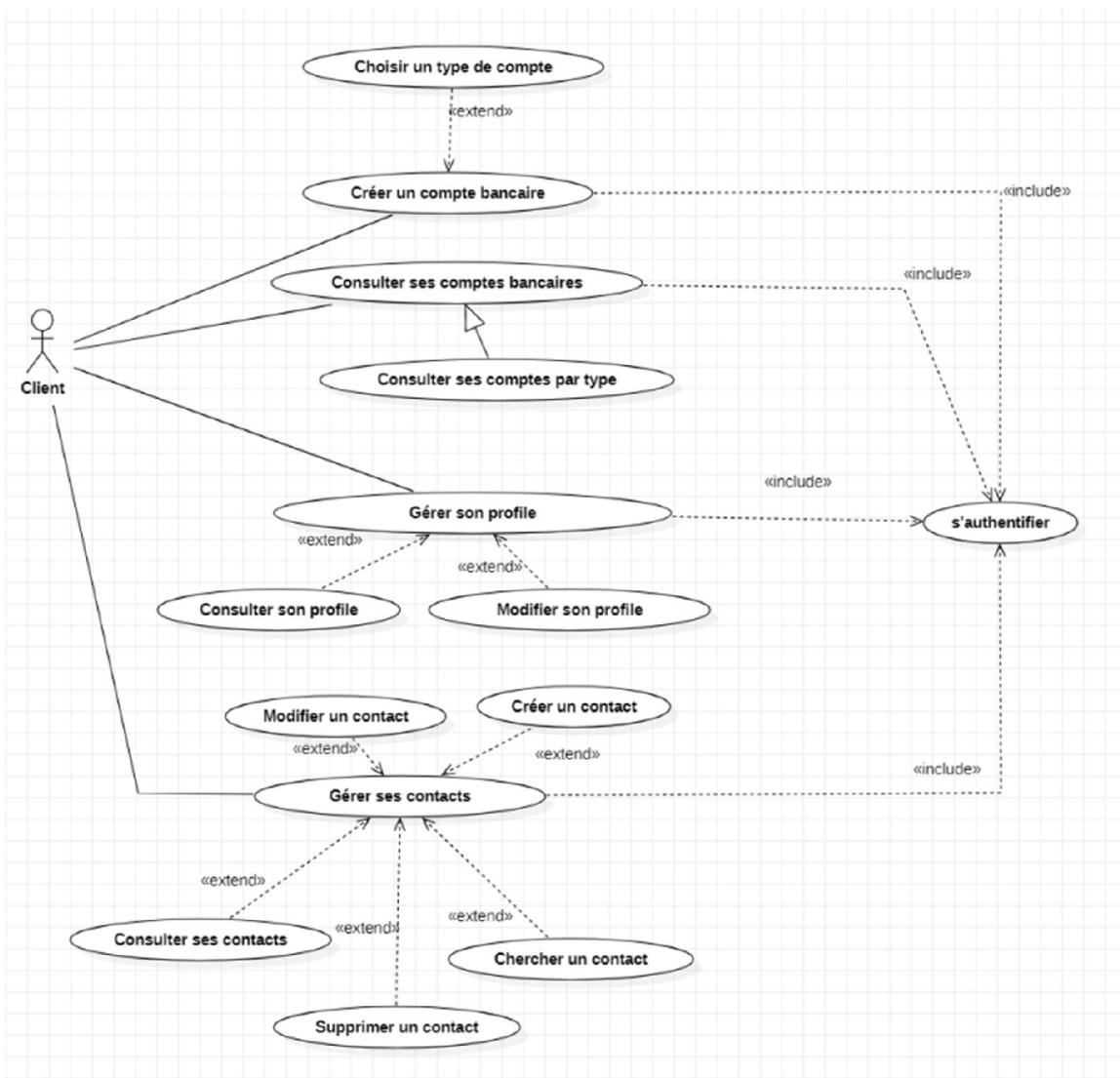


Figure 7: Diagramme de cas d'utilisation détaillé du client au sprint 1

## 2.2. Description textuelle des cas d'utilisations

Dans cette partie nous décrivons textuellement les principaux cas d'utilisation qui apparaissent dans le diagramme de cas d'utilisation global.

Tableau 4 : Description textuelle de cas d'utilisation "S'authentifier"

<b>Cas d'utilisation :</b> S'authentifier
<b>Acteur initiateur :</b> Admin, client
<b>But du cas :</b> Accéder aux fonctionnalités avancées de l'application
<b>Précondition :</b> L'utilisateur doit posséder un compte et connaître son login et mot de passe
<b>Description du scénario :</b>
<p><i>Scénario principale (succès) :</i></p> <ul style="list-style-type: none"> <li>1- Le système affiche le formulaire d'authentification.</li> <li>2- L'utilisateur introduit son login et le mot de passe et valide.</li> <li>3- Le système vérifie les données introduites et affiche l'interface d'accueil.</li> </ul> <p><i>Scénario alternatif :</i></p> <ul style="list-style-type: none"> <li>3.1- L'utilisateur introduit des données incorrectes.</li> <li>3.2- Le système affiche un message d'erreur et demande à l'utilisateur de recommencer.</li> </ul>

Tableau 5:Description textuelle du cas d'utilisation "S'inscrire"

<b>Cas d'utilisation :</b> S'inscrire
<b>Acteur initiateur :</b> Admin, client
<b>But du cas :</b> Créer un compte sur la plateforme
<b>Précondition :</b> Le login n'existe pas déjà
<b>Description du scénario :</b>
<p><i>Scénario principale (succès) :</i></p> <ul style="list-style-type: none"> <li>1- Le système affiche le formulaire d'inscription.</li> <li>2- L'utilisateur entre les informations nécessaires puis valide.</li> <li>3- Le système vérifie les données introduites et affiche le succès de la création du compte, génère un login et envoie un email à l'utilisateur contenant le login générer.</li> </ul> <p><i>Scénario alternatif :</i></p> <ul style="list-style-type: none"> <li>3.1- L'utilisateur introduit des données incorrectes.</li> <li>3.2- Le système affiche un message d'erreur pour aider l'utilisateur à corriger .</li> </ul>

Tableau 6 : Description textuelle de cas d'utilisation "Modifier un type de compte"

<b>Cas d'utilisation :</b> Modifier un type de compte
<b>Acteur initiateur :</b> Administrateur
<b>But du cas :</b> Accéder aux données d'un type pour les modifier
<b>Précondition :</b> L'administrateur est authentifié, le type existe déjà dans la base .
<b>Description du scénario :</b>
<p><i>Scénario principale (succès) :</i></p> <ul style="list-style-type: none"> <li>1- L'admin choisit d'afficher toutes les types des comptes.</li> <li>2- Le système affiche la liste.</li> <li>3- L'admin choisit un type à modifier.</li> <li>4- Le système affiche le formulaire de modification.</li> <li>5- L'admin effectue les modifications puis valide.</li> <li>6- Le système enregistre la modification dans la base de données et réaffiche la liste des types.</li> </ul> <p><i>Scénario alternatif :</i></p> <ul style="list-style-type: none"> <li>5.1- L'admin introduit des données incorrectes.</li> <li>5.2- Le système affiche un message d'erreur pour aider l'admin à corriger .</li> </ul>

Tableau 7:Description textuelle de cas d'utilisation « Chercher un contact »

<b>Cas d'utilisation :</b> Chercher un contact
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Accéder aux information d'un contact
<b>Précondition :</b> Le client est authentifié, le contact existe déjà dans la base .
<b>Description du scénario :</b>
<p><i>Scénario principale (succès) :</i></p> <ul style="list-style-type: none"> <li>1- Le système affiche une zone de recherche des contacts.</li> <li>2- Le client introduit le nom du contact souhaité et demande à le chercher.</li> <li>3- Le système affiche le résultat de la recherche dans la liste des contacts.</li> </ul> <p><i>Scénario alternatif :</i></p> <ul style="list-style-type: none"> <li>3.1- Le client introduit un contact qui n'existe pas.</li> <li>3.2- Le système affiche un message d'erreur que le contact n'existe pas le</li> </ul>

Tableau 8 :Description textuelle de cas d'utilisation « Supprimer un type de compte »

<b>Cas d'utilisation :</b> Supprimer un type de compte
<b>Acteur initiateur :</b> Administrateur
<b>But du cas :</b> Effacer un type de la base
<b>Précondition :</b> L'administrateur est authentifié, le type existe déjà dans la base .
<b>Description du scénario :</b>
<p><i>Scénario principale (succès) :</i></p> <ul style="list-style-type: none"> <li>1- L'admin choisit d'afficher toutes les types des comptes.</li> <li>2- Le système affiche la liste.</li> <li>3- L'admin choisit un type à supprimer.</li> <li>4- Le système demande la confirmation de la suppression.</li> <li>5- L'admin confirme la suppression.</li> <li>6- Le système efface le type de la base de données.</li> </ul> <p><i>Scénario alternatif :</i></p> <ul style="list-style-type: none"> <li>5.1- L'admin annule la suppression.</li> <li>5.2- Le système réaffiche la liste .</li> </ul>

### 3. Description comportementale des histoires du sprint 1

Pour illustrer l'aspect comportemental du release de ce premier sprint, nous avons eu recourt aux diagrammes de séquences. Un diagramme de séquence permet de représenter les échanges entre les différents objets et acteurs du système en fonction du temps afin d'illustrer un cas d'utilisation, autrement dit, l'une des histoires d'un acteur du système.[18]

Dans la suite, nous présentons les diagrammes de séquences correspondant aux principaux cas d'utilisation de notre sprint1.

- **Diagramme de séquence « S'inscrire »**

La figure 7 illustre l'enchaînement des séquences nécessaires à la réalisation du cas d'utilisation s'inscrire pour le client.

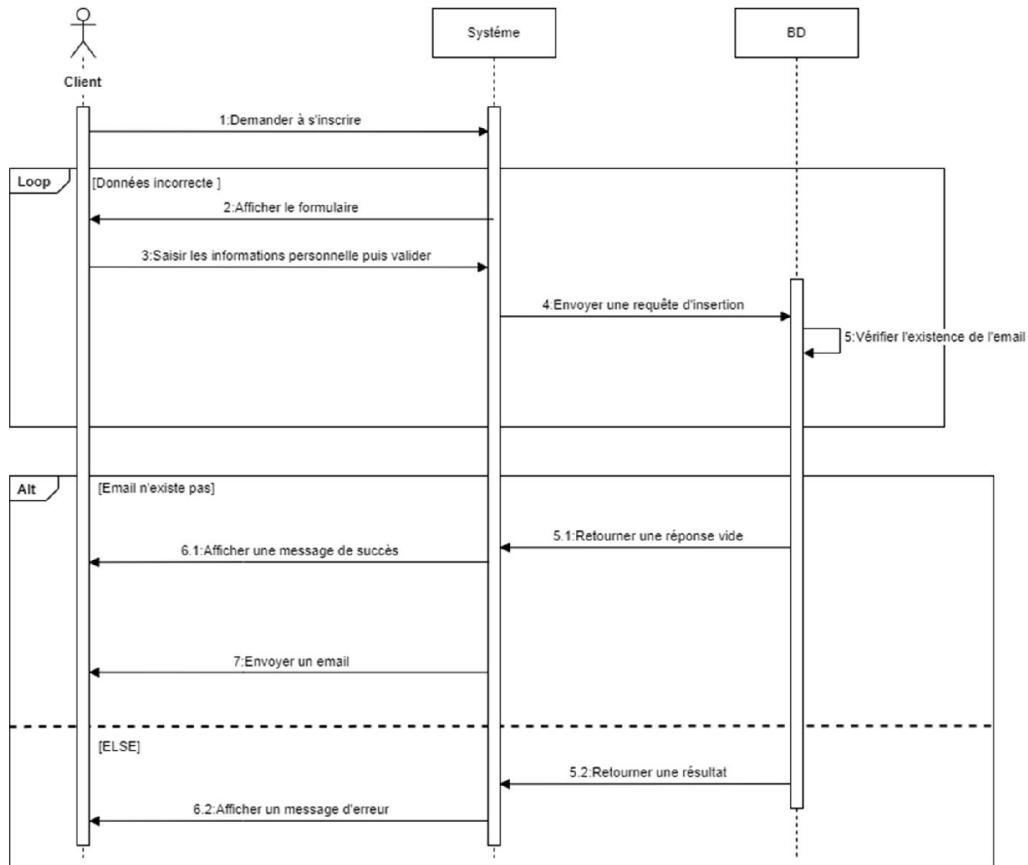


Figure 8: Diagramme de séquence du cas d'utilisation « S'inscrire »

- **Diagramme de séquence « S'authentifier »**

La figure 14 montre l’interaction entre le système et l’utilisateur pour la réalisation du cas d’utilisation s’authentifier.

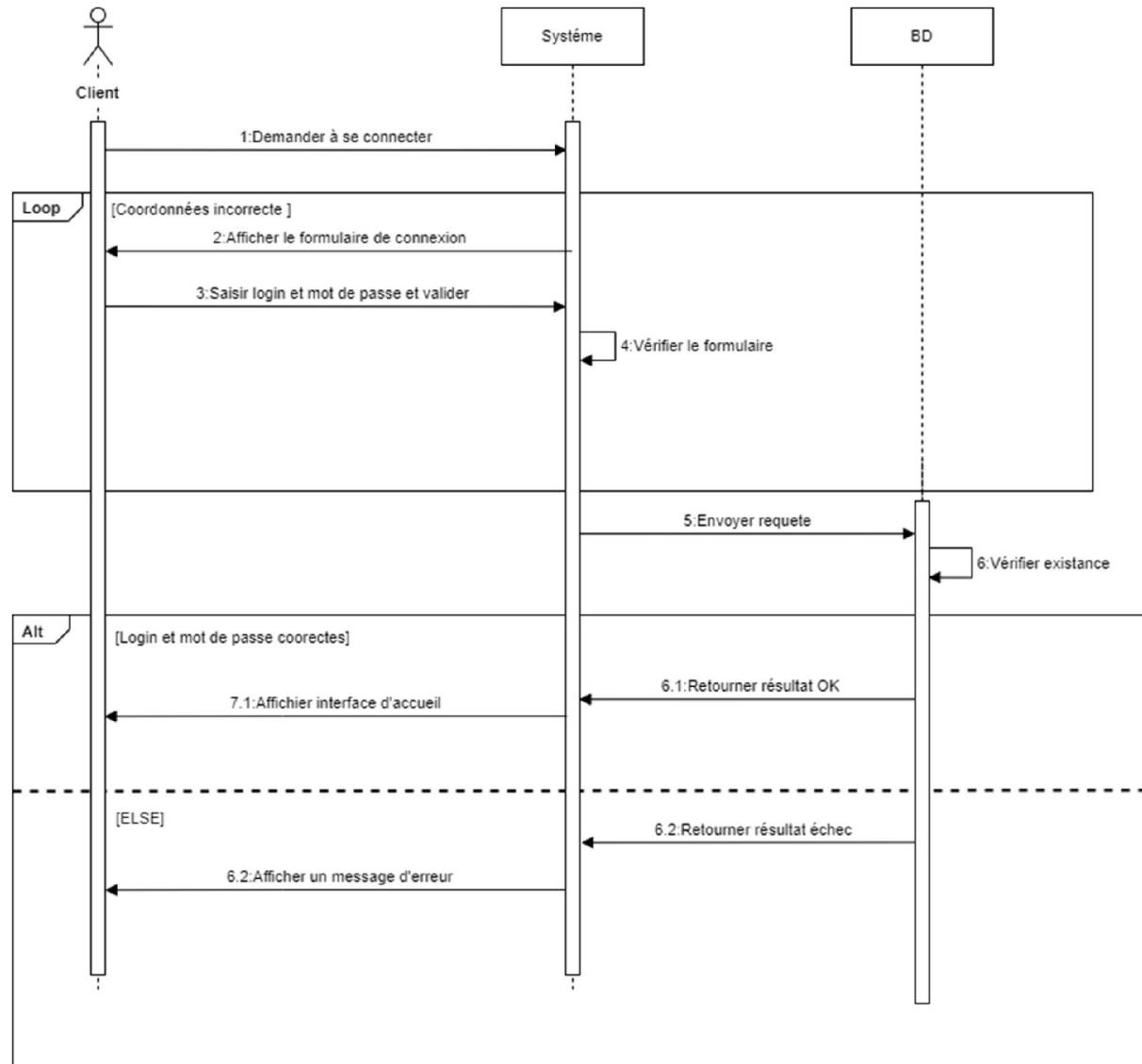


Figure 9: Diagramme de séquence du cas d'utilisation « S'authentifier »

- **Diagramme de séquence « Modifier un type de compte »**

La figure 15 présente les séquence l'interaction entre le système l'administrateur pour la réalisation du cas d'utilisation modifier un établissement.

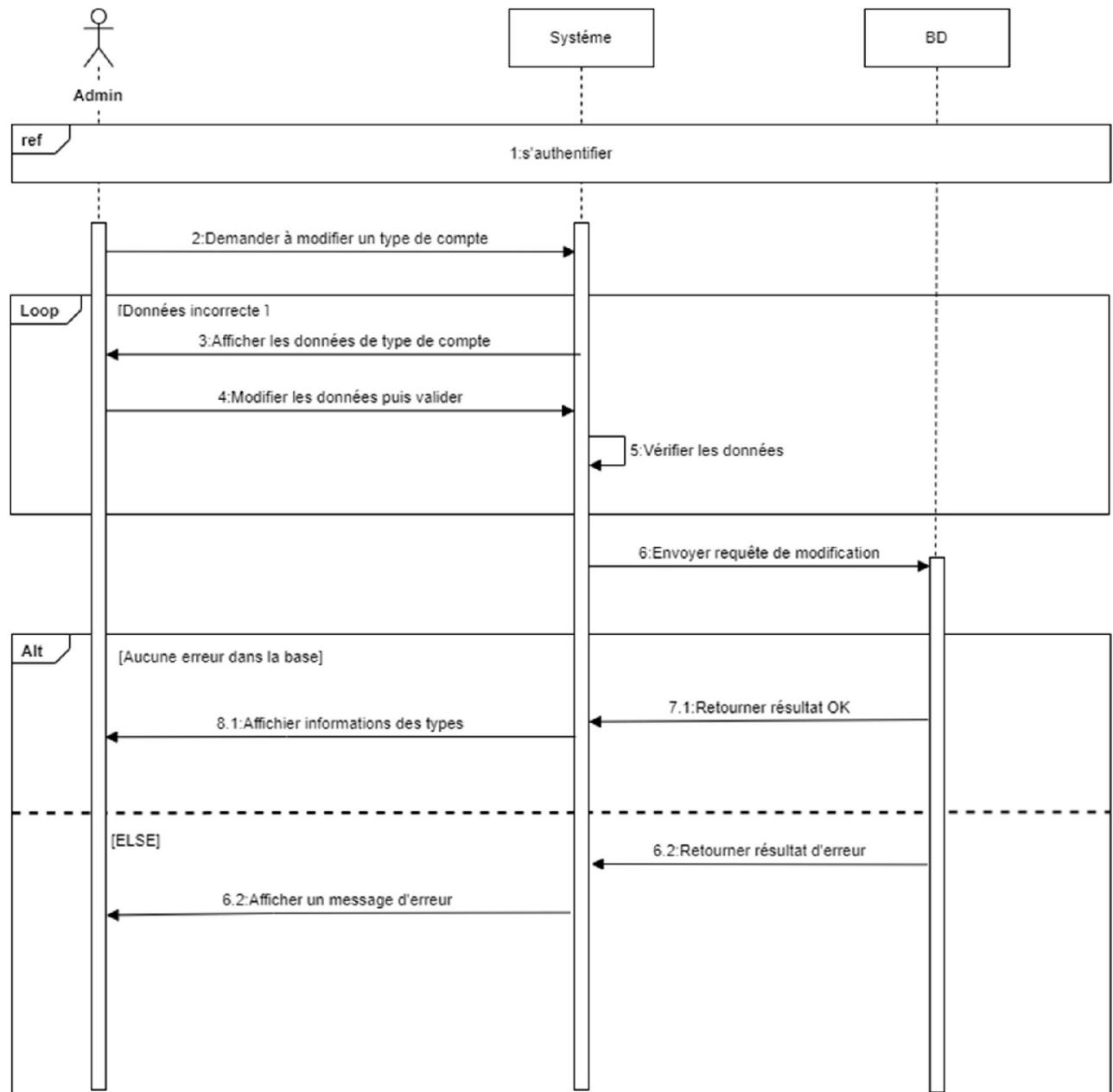


Figure 10 : Diagramme de séquence du cas d'utilisation « Modifier un type de compte »

- **Diagramme de séquence « chercher un contact»**

La figure 16 illustre l'enchaînement des séquences nécessaires à la réalisation du cas d'utilisation rechercher un contact.

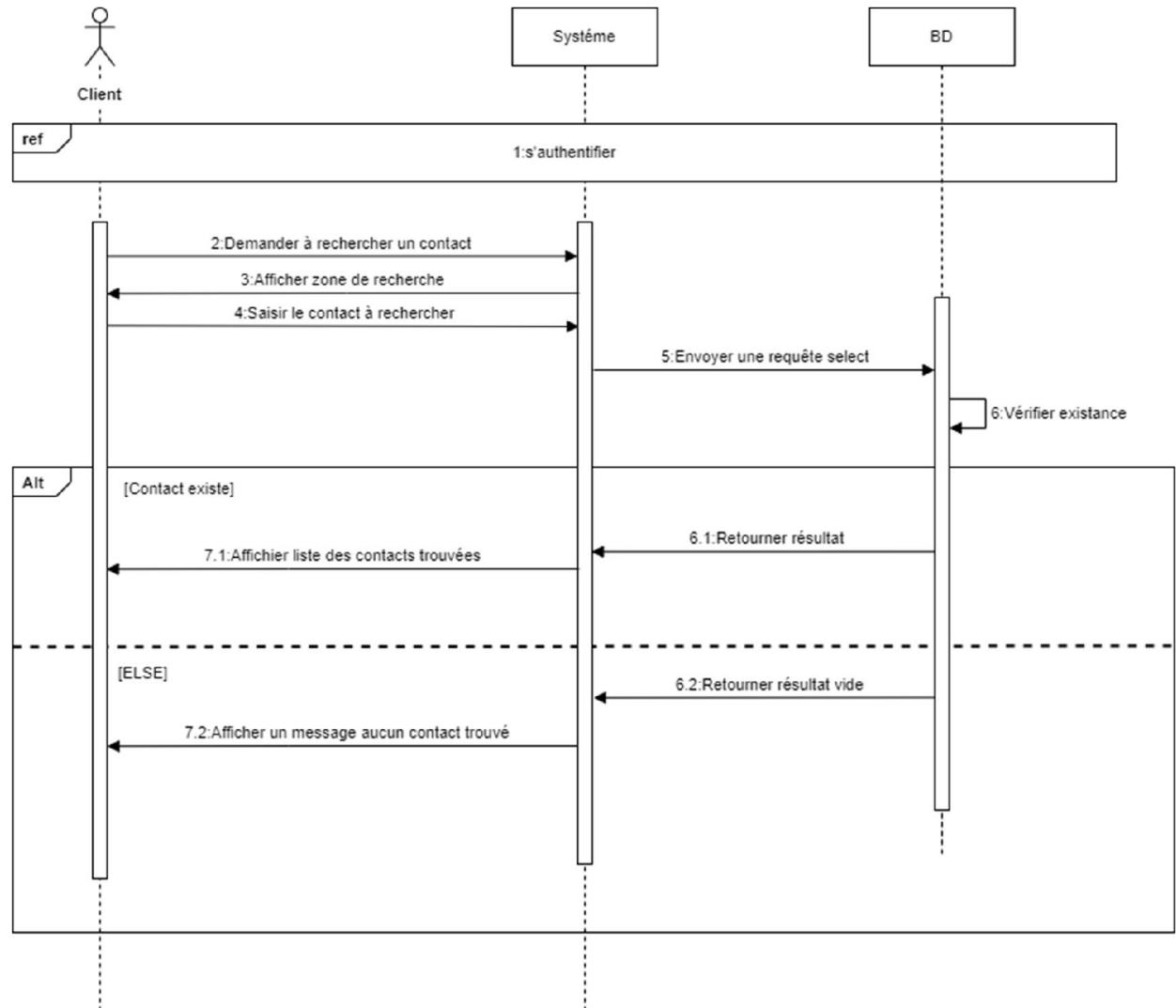


Figure 11: Diagramme de séquence du cas d'utilisation « Rechercher un contact »

- **Diagramme de séquence « supprimer un type de compte»**

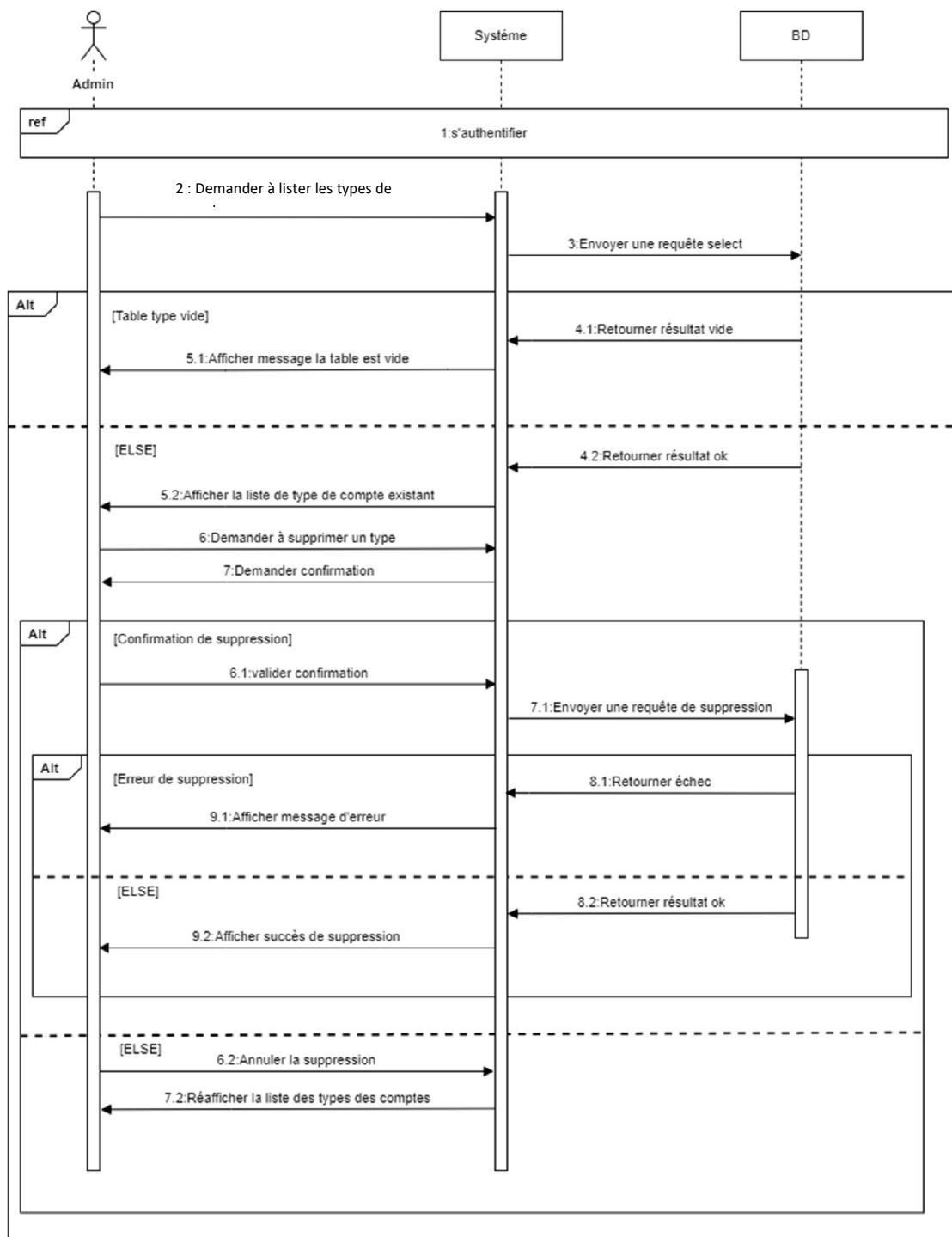


Figure 12:Diagramme de séquence du cas d'utilisation « supprimer un type de compte »

## 4. Diagramme de classes du sprint 1

Le diagramme de classes est une représentation statique des éléments qui composent un système et de leurs relations. Il représente les classes intervenant dans le système.[19]

Pour le premier sprint de notre projet, nous avons établi le diagramme de classes illustré par la figure 18. Il comporte 8 classes

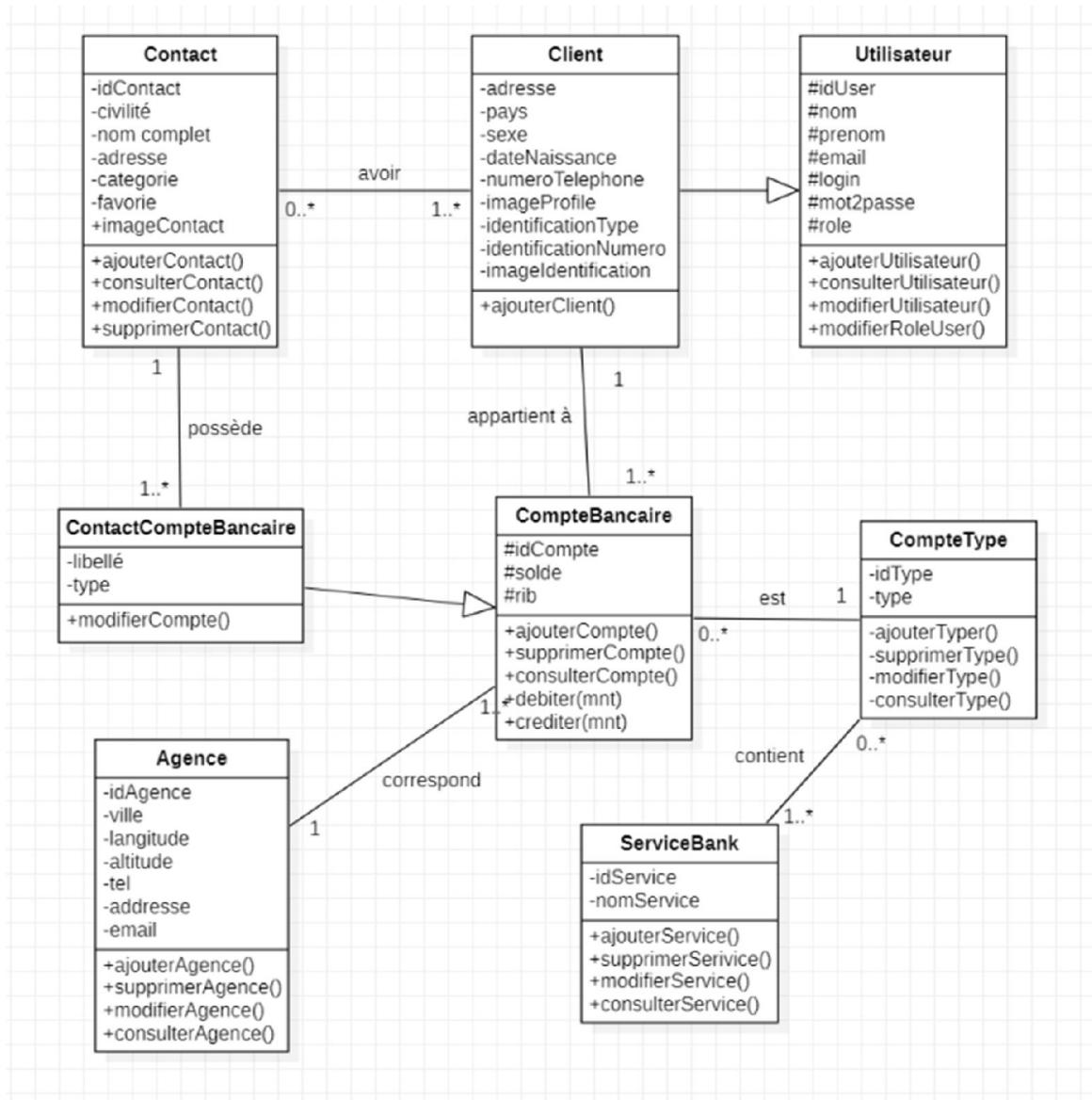


Figure 13:Diagramme de classe du sprint 1

## 5. Review du sprint 1

Pour illustrer la réalisation du premier sprint, nous présenterons les interfaces des principales fonctionnalités offertes par le module configuration des comptes bancaires.

### 5.1. Espace administrateur

Les interfaces de cette première section concernent l'espace accessible à l'administrateur.

- **Interface d'authentification :**

Si l'administrateur n'est pas authentifié c'est l'interface d'authentification qui s'affiche en premier. L'admin saisit son login et son mot de passe, puis clique sur le bouton « se connecter ». Le système vérifie la validité des informations saisies et affiche un message d'erreur si elles sont incorrectes, l'administrateur est redirigé vers l'interface Dashboard.

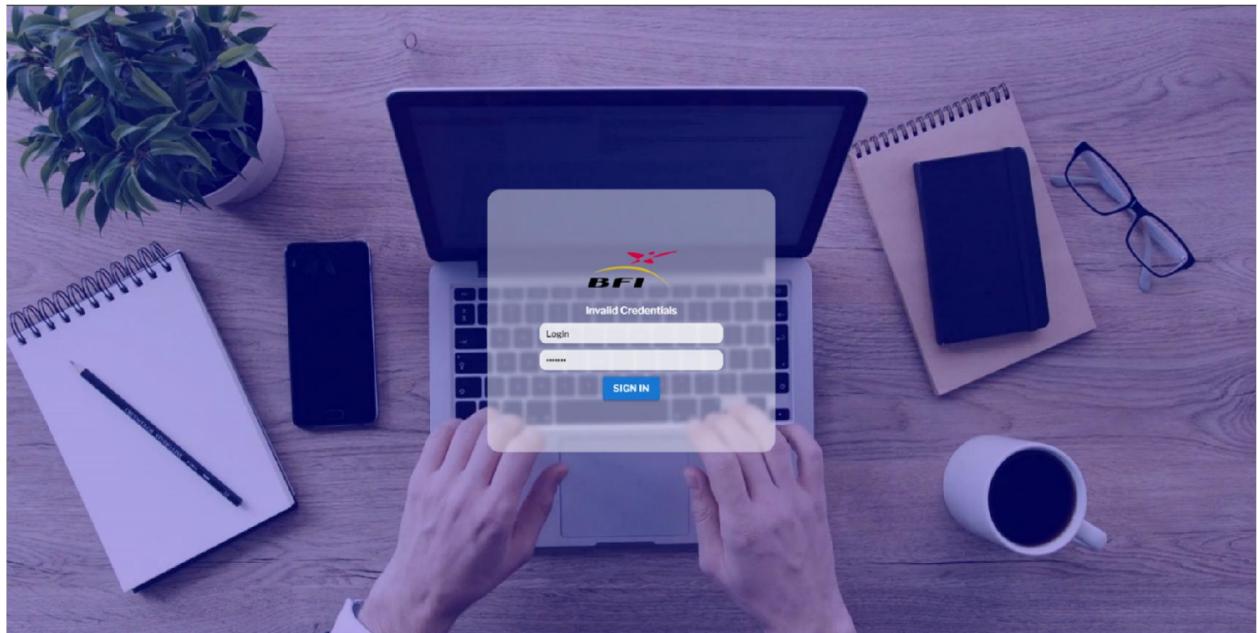


Figure 14: Interface d'authentification - admin

- **Interface Dashboard :**

Si l'administrateur est authentifié correctement, le système affiche l'interface Dashboard qui contient les informations utiles et des statistiques.



Figure 15: Interface Dashboard - admin

- **Interface liste des clients**

En tant qu'administrateur, il a la possibilité de voir tous les clients inscrits dans l'application. A partir de cette liste, il peut consulter les détails de chaque client ou de le bloquer.

Client List											
ID	Firstname	Lastname	Email	Address	Country	Gender	Birthday	Phone Number	Details	Ban	
2	rami	chargui	charguirami00@gmail.com	djerba	Tunisia	male	14/4/2022	+216 26075923			
4	user	user	user@gmail.com	Tunis	Tunisia	male	12/04/1995	+216 24242424			
5	foulen	ben foulen	foulen@gmail.com	Tunis	Tunisia	male	12/05/1997	+216 25252425			
6	foulena	ben foulen	foulena@gmail.com	Tunis	Tunisia	female	12/06/1997	+216 95252425			
7	John	Doe	john@gmail.com	New York	USA	male	12/05/1997	+1 95252425			
8	Jane	Doe	jane@gmail.com	London	United Kingdom	female	12/05/1999	+44 95252425			

Figure 16: Interface liste des clients

- **Interface liste des comptes**

L'administrateur a pour rôle de gérer les types des comptes, il peut donc consulter la liste des comptes et les activer et désactiver

ID	Balance	Type	Owner	Default Agency	Status	Activate
100026	0 DT	Saving Account	rami chargui	NULL	Active	
100030	0 DT	Saving Account	rami chargui	Tunis	Inactive	
100031	0 DT	Saving Account	rami chargui	Tunis	Inactive	
100046	0 DT	Saving Account	rami chargui	NULL	Inactive	
100047	0 DT	Saving Account	rami chargui	NULL	Inactive	
100048	0 DT	Saving Account	rami chargui	NULL	Inactive	
100049	0 DT	Saving Account	rami chargui	NULL	Inactive	
100050	0 DT	Saving Account	rami chargui	Tunis	Active	
100052	0 DT	Saving Account	user user	NULL	Inactive	
100053	0 DT	Current Account	foulen ben foulen	NULL	Inactive	
100054	0 DT	Current Account	foulen ben foulen	NULL	Inactive	
100055	0 DT	Current Account	John Doe	NULL	Inactive	
100056	0 DT	Saving Account	Jane Doe	NULL	Inactive	

Figure 17: Interface liste des comptes

- **Interface ajout d'un type de compte**

L'administrateur a pour rôle de gérer les types des comptes, il peut donc en ajouter un. Pour cela, il accède à la liste des types de compte dans le menu à gauche puis clique sur le bouton « Add type » à droite au-dessus du tableau. Une boîte de dialogue s'affiche permettant de saisir le nouveau type de compte et de valider

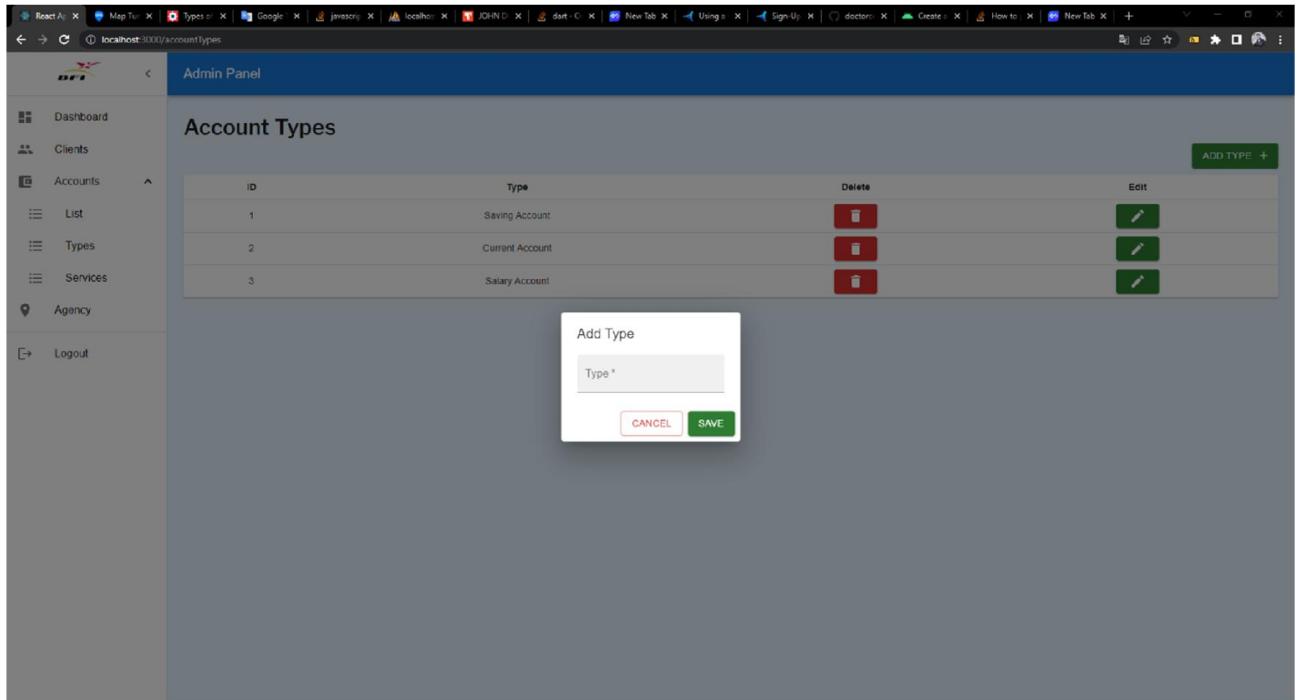


Figure 18: Interface modifier type de compte

- **Interface suppression d'un service**

Pour supprimer un service, il suffit d'accéder à la liste des services dans le menu à gauche et de cliquer sur le bouton supprimer correspondant. Une demande de confirmation est affichée et la suppression ne se fait que si l'administrateur valide ce choix. Dans ce cas, un message de succès s'affiche et la liste est mise à jour

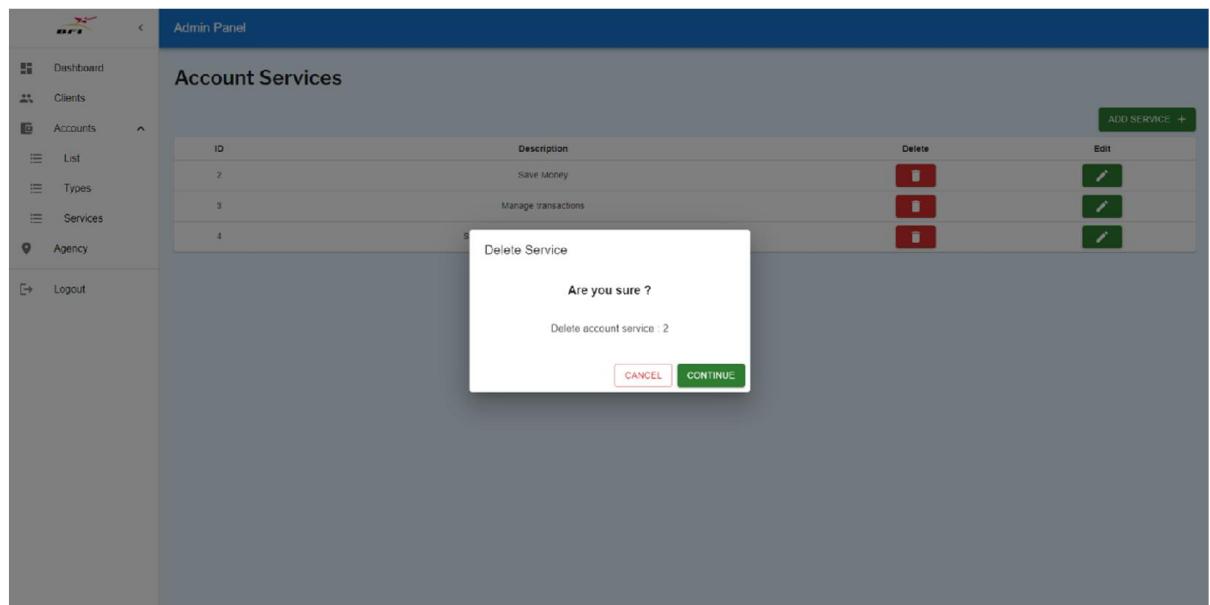


Figure 19: Interface supprimer un service

## 5.2.Espace client

- **Interface d'authentification :**

La première interface qui s'affiche lors du lancement de l'application montre le logo de la banque suivi de l'interface d'authentification. Le client saisit son login et son mot de passe, puis clique sur le bouton « se connecter ». Le système vérifie la validité des informations saisies en dirigeant vers la page d'accueil.

S'il s'agit d'un nouveau client qui ne dispose pas d'un compte, il clique sur le bouton « Sign in » afin d'afficher le formulaire d'inscription.

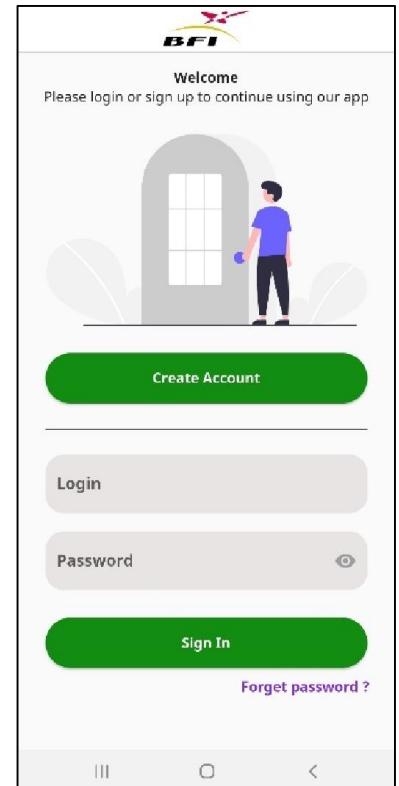


Figure 20: Interface authentification - client

- **Interface récupération de compte:**

Si le client oublie son mot de passe il peut le récupérer en cliquant sur « Forget password » dans l'interface d'authentification. Dans ce cas, un formulaire s'affiche, le client doit saisir ses coordonnées le système envoie un SMS contenant un code à saisir dans le formulaire. S'il est valide, une autre interface s'affiche, dans laquelle le client peut changer son mot de passe.

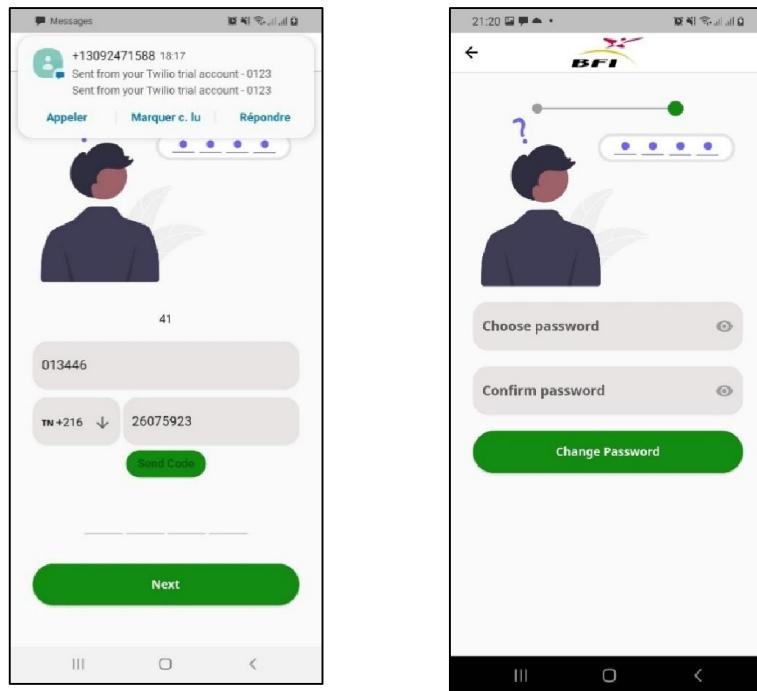


Figure 21: Interface récupération de mot de passe

- **Interface d'inscription :**

Pour l'inscription, il faut passer par quelques étapes afin d'indiquer les préférences de son compte bancaire et les détails de son profile. Si un champ n'est pas renseigné, le message d'erreur est affiché. Lorsque toutes les informations ont été complétées avec succès, le compte est créé.

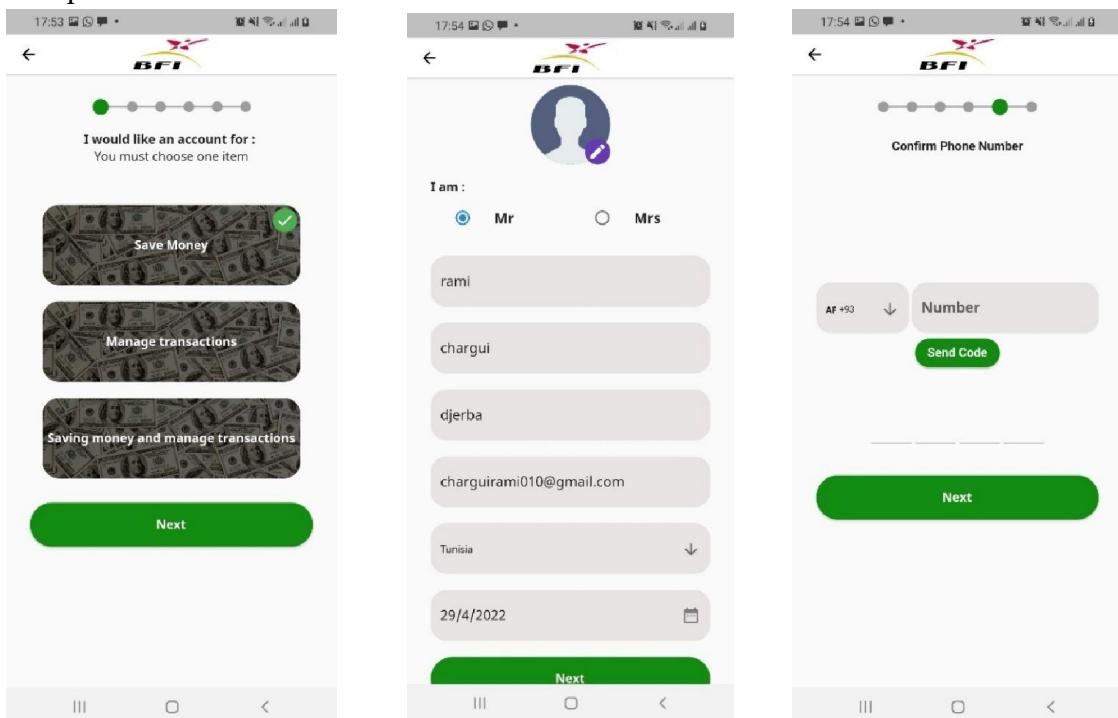


Figure 22: Interface inscription

- **Interface compte bancaire :**

Une fois authentifié, le client accède à cette interface en choisissant « Account » dans le menu à gauche. Il a donc la possibilité de consulter ses comptes et leurs détails (numéro, état, solde, ...) en lui offrant l'option de filtrer la liste selon le type de compte. Cette interface lui permet aussi de créer de nouveaux comptes bancaires en cliquant sur « add ». Une boîte de dialogue apparaît et le client peut remplir le formulaire puis valide, la liste des comptes est alors mise à jour.

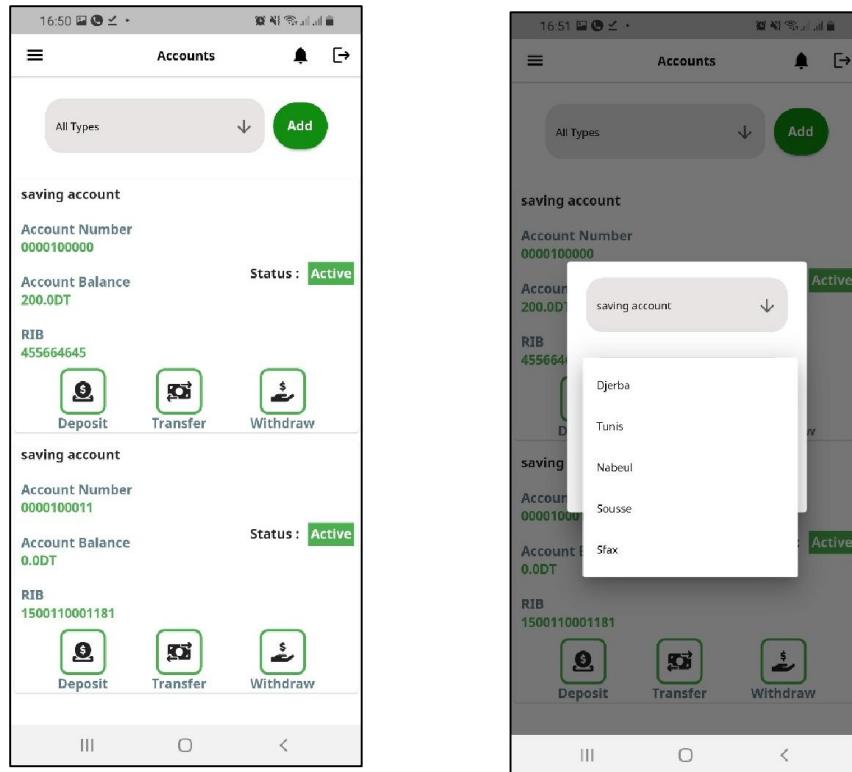


Figure 23: Interface compte bancaire

- **Interface Contact :**

L'application permet aux clients de saisir une liste de contacts avec leurs comptes bancaires afin de leur transférer de l'argent.

Le client peut accéder à l'interface contact en cliquant sur « mes contacts » dans le menu à gauche. Une interface s'affiche et lui permet de consulter la liste des contacts existants et favoris, il peut aussi ajouter un client en cliquant sur « + ».

A partir de la liste de contacts, le client a aussi la possibilité de consulter la liste de tous les contacts en cliquant sur « others ». Dans l'interface qui s'affiche, le client doit saisir le nom du

contact à chercher un contact, soit trier les contacts selon l'ordre alphabétique ou la date d'ajout du plus récent au plus ancien.

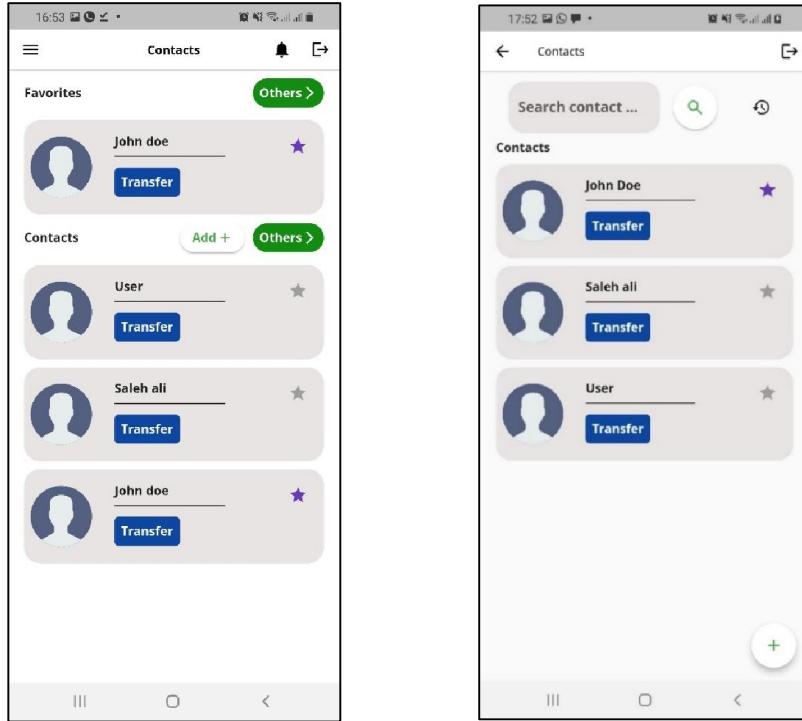


Figure 24: Interface liste des contacts

Pour ajouter un client et après avoir cliqué sur « Add + » dans la première interface de la liste des contacts, il faut remplir le formulaire avec les informations du contact et les détails de ses comptes bancaires.

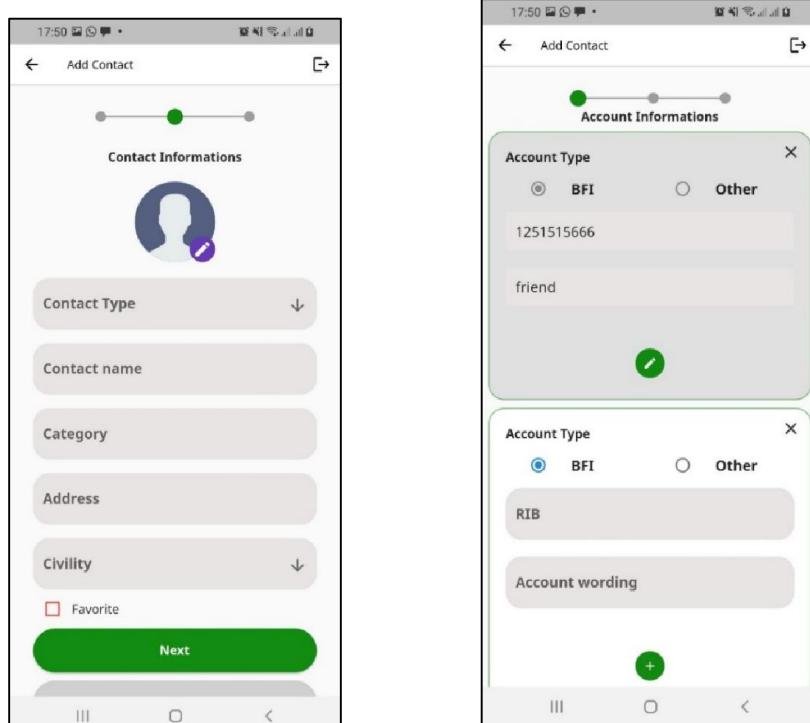


Figure 25: Interface ajout d'un contact

Le client peut aussi consulter le détails d'un contact en cliquant sur son avatar ou son nom. Une nouvelle interface apparait, à partir de laquelle le client peut consulter les informations concernant le contact et ses comptes bancaires. Il peut également modifier ou supprimer ce contact grâce aux boutons Edit et Delete.

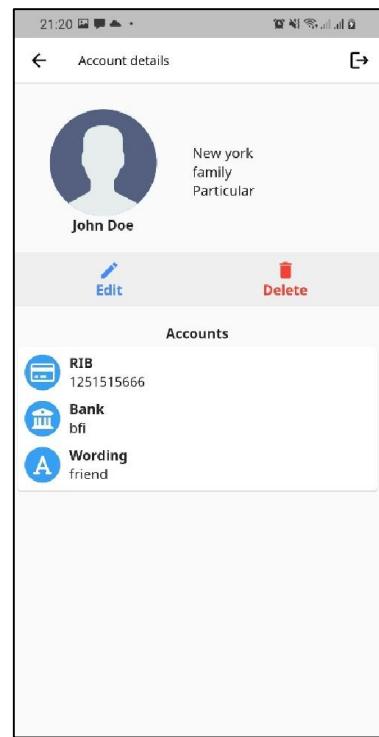


Figure 26: Interface détail contact

## Conclusion

Dans ce chapitre nous avons établi le 1er sprint, et nous avons défini ses différentes fonctionnalités réalisées à travers ses descriptions fonctionnelles et comportementale des histoires. Finalement nous avons défini son diagramme de classe suivi de ses réalisations en terme d'interface réalisées dans ce sprint. Dans le chapitre suivant nous allons décrire le 2eme sprint sous forme de chapitre 4.

# Chapitre 4 : Sprint 2 – Module transactions

## Introduction

En passant au deuxième sprint, nous suivons le même enchaînement que le sprint précédent, et nous commençons par le backlog du sprint suivi de l'étude fonctionnelle et comportementale, pour finir avec le diagramme de classes et la réalisation.

## 1. Backlog du sprint 2

Tableau 9: Backlog du sprint 2

Fonctionnalité	ID	User story	Taches	Sprint point
<b>Gérer les retraits d'argent</b>	1	En tant que admin je veux consulter la liste des retraits d'argent et pouvoir valider ou annuler un retrait.	4- Créer la classe retrait et dépôt 5- Donner le droit à l'admin de consulter et valider les retrait	3
	2	En tant que client je veux pouvoir effectuer des retraits d'argent, les annuler et consulter leurs états.	6- Donner le droit au client d'effectuer des retraits, et de pouvoir les consulter.	4
<b>Gérer les dépôts d'argent</b>	3	En tant que admin je veux consulter la liste des dépôts d'argent et pouvoir valider ou annuler un dépôt.	7- Donner le droit à l'admin de consulter et valider les dépôts	3
	4	En tant que client je veux pouvoir effectuer des dépôts d'argent, l'annuler et consulter leurs états.	8- Donner le droit au client d'effectuer des dépôts , et de pouvoir les consulter et les annuler.	4

<b>Gérer les virements bancaires</b>	5	En tant que admin je veux consulter la liste des virements bancaires et pouvoir valider ou annuler un virement.	9- Créer la classe virements. 10- Ajouter les attributs nécessaires. 11- Donner le droit à l'admin de consulter et valider les virements	3
	6	En tant que client je veux pouvoir effectuer des virements bancaires, l'annuler et consulter leurs états.	12- Donner le droit au client d'effectuer des virements , et de pouvoir les consulter et les annuler.	4
<b>Consulter l'historique des transactions</b>	7	En tant que client je veux consulter l'historique de mes transactions et leurs états.	13- Donner le droit au client de consulter l'historique de ses transactions	2

## 2. Description fonctionnelle des histoires du sprint 2

Nous décrirons les histoires du module transaction en présentant les diagrammes de cas d'utilisation, leurs descriptions textuelles ainsi que les diagrammes de séquences correspondants.

### 2.1. Diagrammes de cas d'utilisation

Dans cette section, nous commençons par présenter le diagramme de cas d'utilisation global correspondant au sprint 2, par la suite, nous raffinerons par acteur.

#### 2.2.1. Diagramme de cas d'utilisation global du sprint 2

Pour ce second sprint, nous présentons le diagramme de cas d'utilisation global qui illustre les fonctionnalités offertes par le module transactions

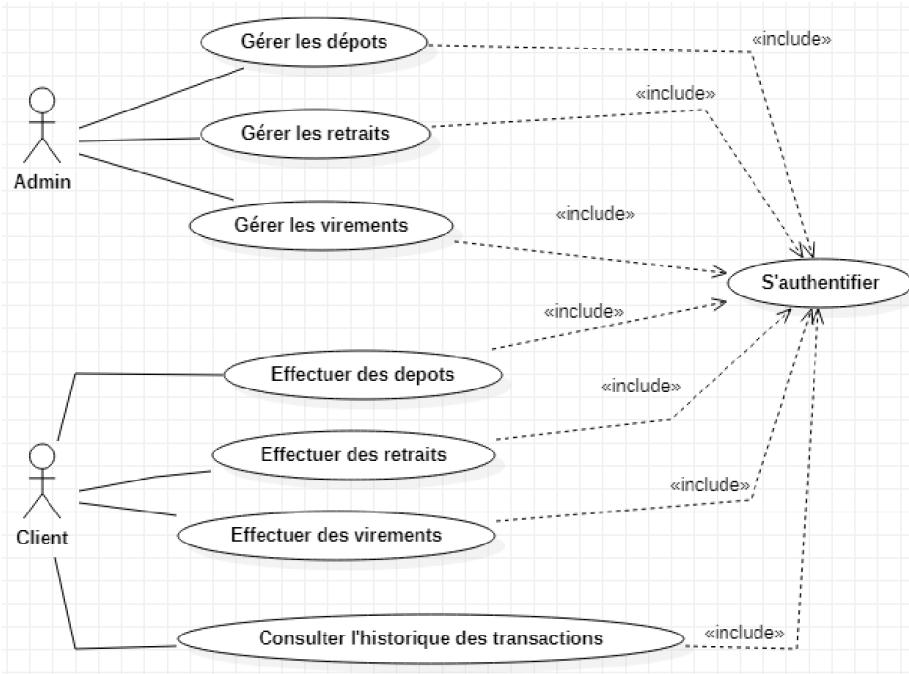


Figure 27 : Diagramme de cas d'utilisation globale sprint 2

### 2.2.2. Diagramme de cas d'utilisation détaillés du sprint 2

On passe au raffinement des diagrammes par acteur.

- **Diagramme de cas d'utilisation détaillé du client**

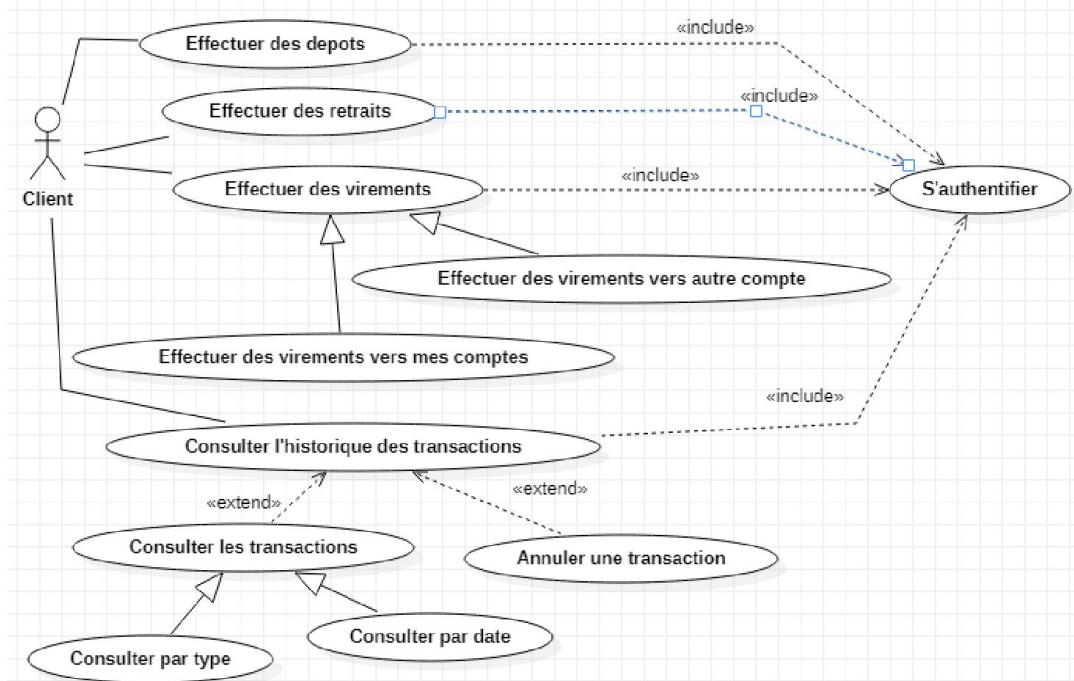


Figure 28: Diagramme de cas d'utilisation détaillé du client

- **Diagramme de cas d'utilisation détaillé de l'administrateur**

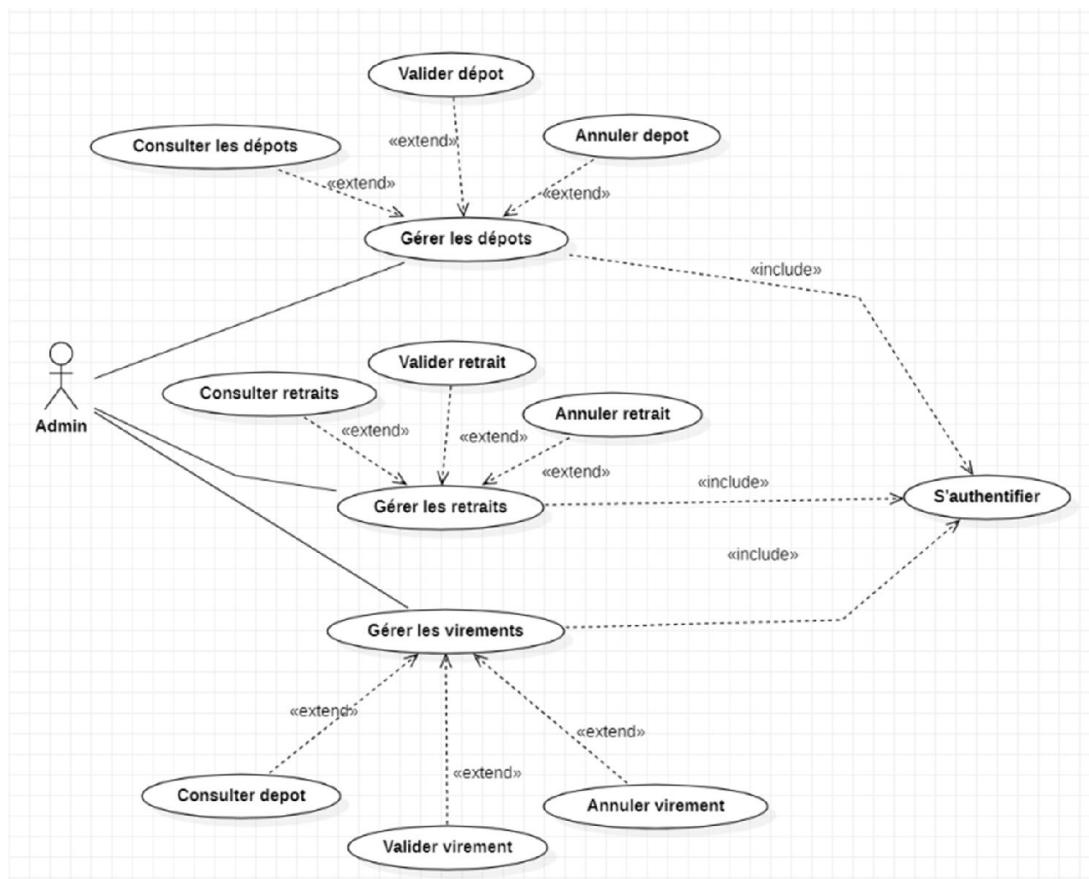


Figure 29:Diagramme de cas d'utilisation détaillé de l'administrateur

## 2.2. Description textuelle des cas d'utilisation

Dans cette partie nous décrivons textuellement les principaux cas d'utilisation qui apparaissent dans les digrammes de cas d'utilisation.

Tableau 10: Description textuelle du cas d'utilisation « Valider un dépôt »

<b>Cas d'utilisation :</b> Valider un dépôt
<b>Acteur initiateur :</b> Administrateur
<b>But du cas :</b> Mettre à jour l'état du dépôt en le confirmant
<b>Précondition :</b> L'administrateur est authentifié, la demande dépôt existe déjà dans la base.
<b>Description du scénario :</b>
<i>Scénario principale (succès) :</i>
1- L'administrateur demande la liste de dépôts.

- 2- Le système affiche la liste des dépôts existants.
- 3- L'administrateur choisit un dépôt à valider
- 4- Le système met à jour l'état du dépôt dans la base de données

*Scénario alternatif :*

- 2.1- La table dépôt est vide.

Tableau 11: Description textuelle du cas d'utilisation « Effectuer un retrait »

<b>Cas d'utilisation :</b> Effectuer un retrait
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Effectuer une opération de retrait de l'un des comptes bancaires du client.
<b>Précondition :</b> Le client est authentifié, la compte existe déjà dans la base.
<b>Description du scénario :</b>
<i>Scénario principale (succès) :</i>
<ul style="list-style-type: none"> <li>1- Le client demande à effectuer un retrait.</li> <li>2- Le système affiche le formulaire du retrait.</li> <li>3- Le client choisit le compte</li> <li>4- Le système vérifie l'état du compte</li> <li>5- Le client introduit le montant à extraire et valide.</li> <li>6- Le système vérifie le solde courant du compte et enregistre la demande de retrait dans la base</li> </ul>
<i>Scénario alternatif :</i>
<ul style="list-style-type: none"> <li>3.1- Aucun compte n'est actif et le système affiche un message pour expliquer au client</li> <li>5.1- Le client introduit un montant supérieur au solde du compte sélectionné</li> <li>6.1- Le système affiche un message d'erreur pour aider le client.</li> </ul>

Tableau 12: Description textuelle du cas d'utilisation « Annuler une transaction»

<b>Cas d'utilisation :</b> Annuler une transaction
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Effacer une transactions de la base de données
<b>Précondition :</b> Le client est authentifié, transactions n'est pas encore valider.
<b>Description du scénario :</b> <i>Scénario principale (succès) :</i> 1- Le client demande l'historique de ses transactions. 2- Le système affiche la liste de ses transactions. 3- Le client choisit une transaction à annuler. 4- Le système demande la confirmation de l'annulation. 5- Le client confirme l'annulation. 6- Le système efface la demande de transaction de la base.  <i>Scénario alternatif :</i> 2.1- La table des transactions est vide. 3.1- La demande de transaction est valider. 5.1- Le client annule la suppression.

Tableau 13: Description textuelle du cas d'utilisation « Effectuer un virement vers un autre compte»

<b>Cas d'utilisation :</b> Effectuer un virement vers un autre compte
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Effectuer une opération de virements d'un compte du client vers le compte de l'un de ses contacts.
<b>Précondition :</b> Le client est authentifié, la compte du client et du contact existent déjà dans la base.

**Description du scénario :**

*Scénario principale (succès) :*

- 1- Le client demande à effectuer un virement.
- 2- Le système affiche le formulaire du virement.
- 3- Le client choisit le compte
- 4- Le système vérifie l'état du compte
- 5- Le client introduit le montant à transférer et choisit le compte du contact auquel transférer l'argent.
- 6- Le système demande la confirmation
- 7- Le client confirme le virement.
- 8- Le système vérifie le solde courant du compte et enregistre la demande de virement dans la base

*Scénario alternatif :*

- 3.1- Aucun compte n'est actif et le système affiche un message pour aider le client
- 5.1- Le client introduit un montant supérieur au solde du compte sélectionné
- 6.1- Le système affiche un message d'erreur pour aider le client à corriger.
- 5.2- Le client ne possède aucun contact.

### 3. Description comportementale des histoires du sprint 2

Dans la suite, nous présentons les diagrammes de séquences correspondant aux principaux cas d'utilisation de notre sprint 2.

- **Diagramme de séquence « Effectuer un retrait »**

La figure 29 illustre l'enchaînement des séquences nécessaires à la réalisation du cas d'utilisation « Effectuer un retrait ».

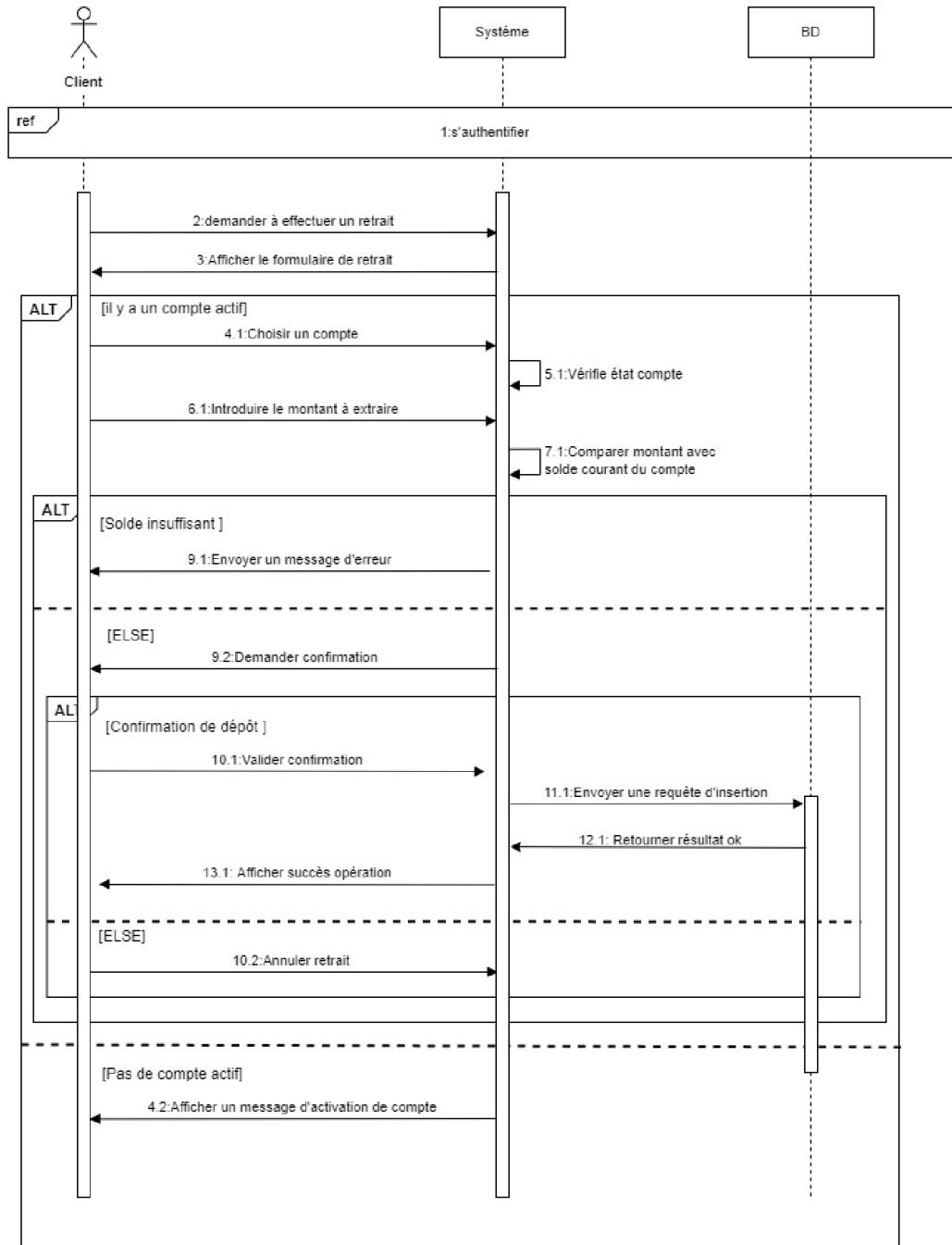


Figure 30 : Diagramme de séquence « Effectuer un retrait »

- **Diagramme de séquence « Valider un dépôt »**

La figure 30 montre l’interaction entre le système et l’administrateur pour la réalisation du cas d’utilisation « valider un dépôt ».

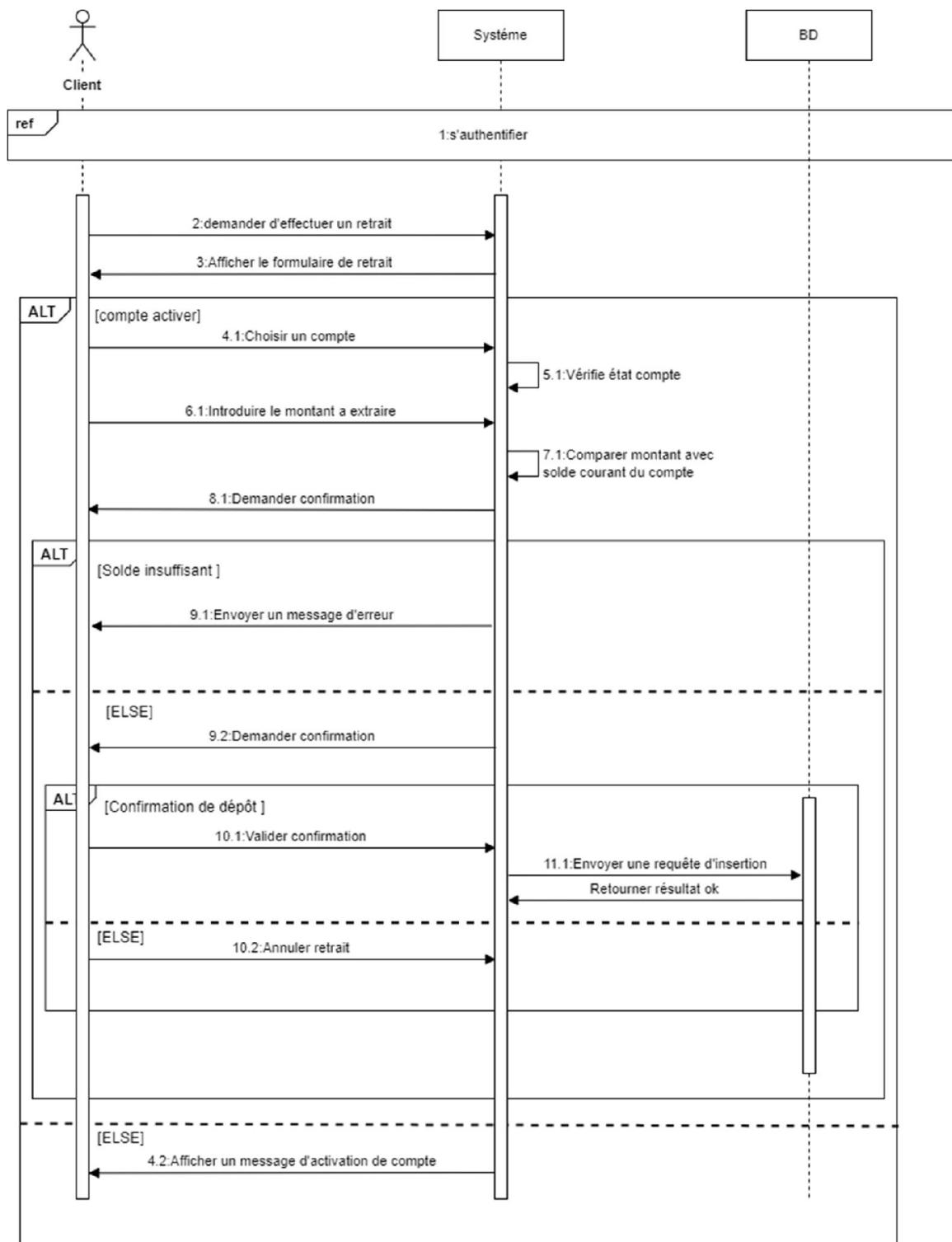


Figure 31 : Diagramme de séquence « Valider un dépôt »

- **Diagramme de séquence « Annuler une transaction »**

La figure 31 présente la réalisation du cas d'utilisation « Annuler une transaction ».

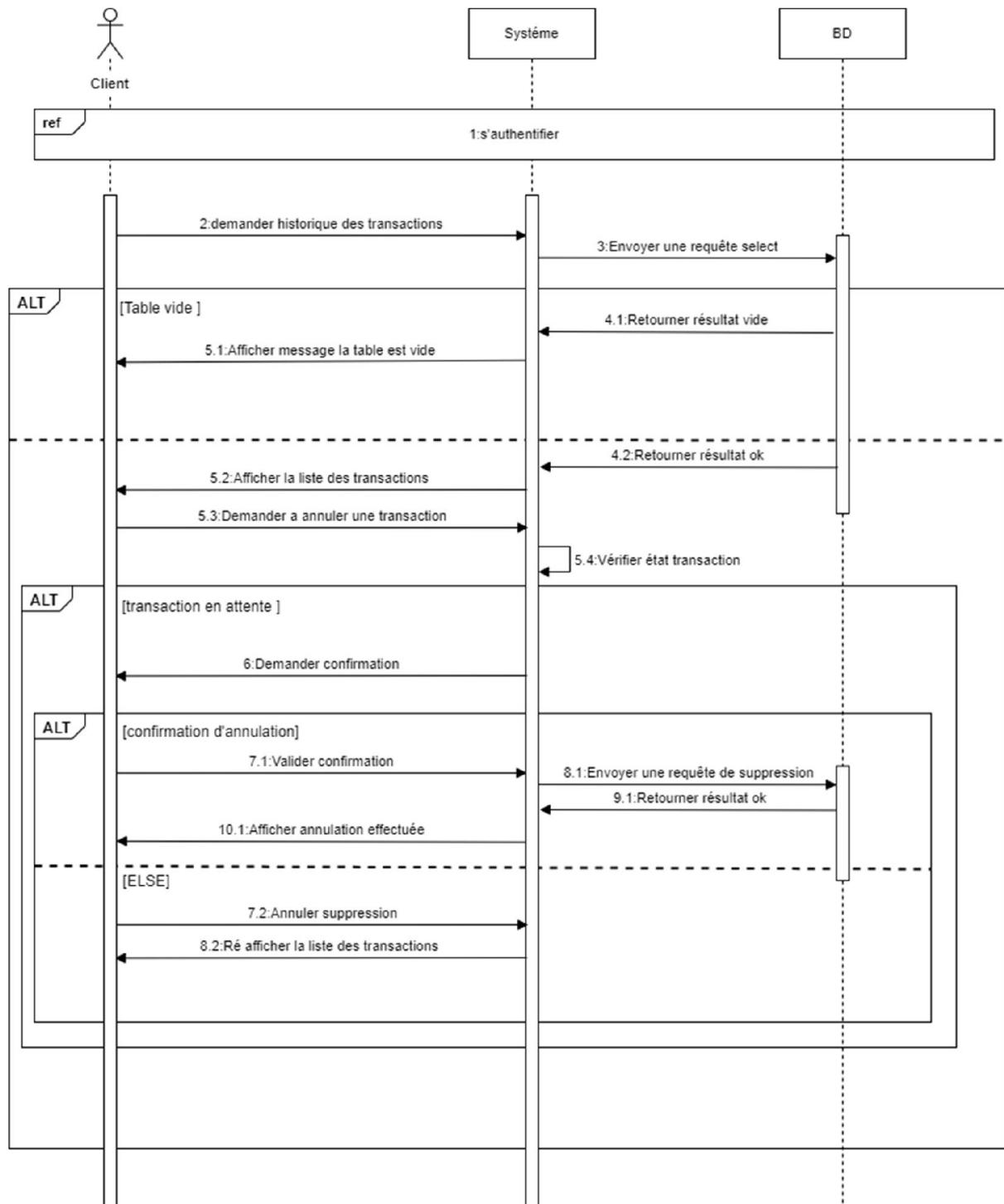


Figure 32 : Diagramme de séquence « Annuler une transaction »

- **Diagramme de séquence « Effectuer virements vers autre compte »**

La figure 32 présente la réalisation du cas d'utilisation « Effectuer virements vers un autre compte ».

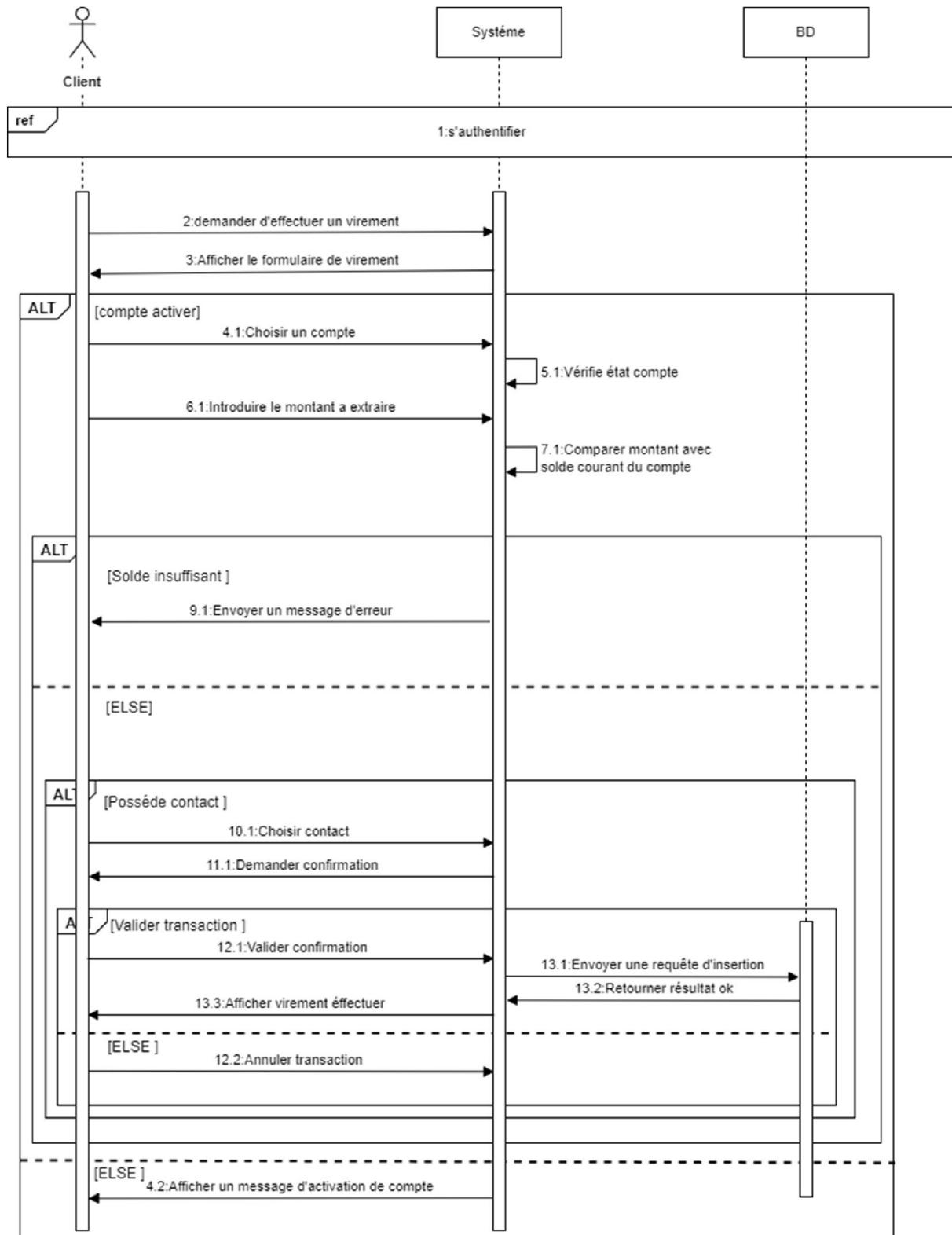


Figure 33 : Diagramme de séquence « Effectuer virements vers un autre compte »

## 4. Diagramme de classes du sprint 2

La figure 33 illustre le diagramme de classes pour le deuxième sprint de notre projet. Il compte 7 classes.

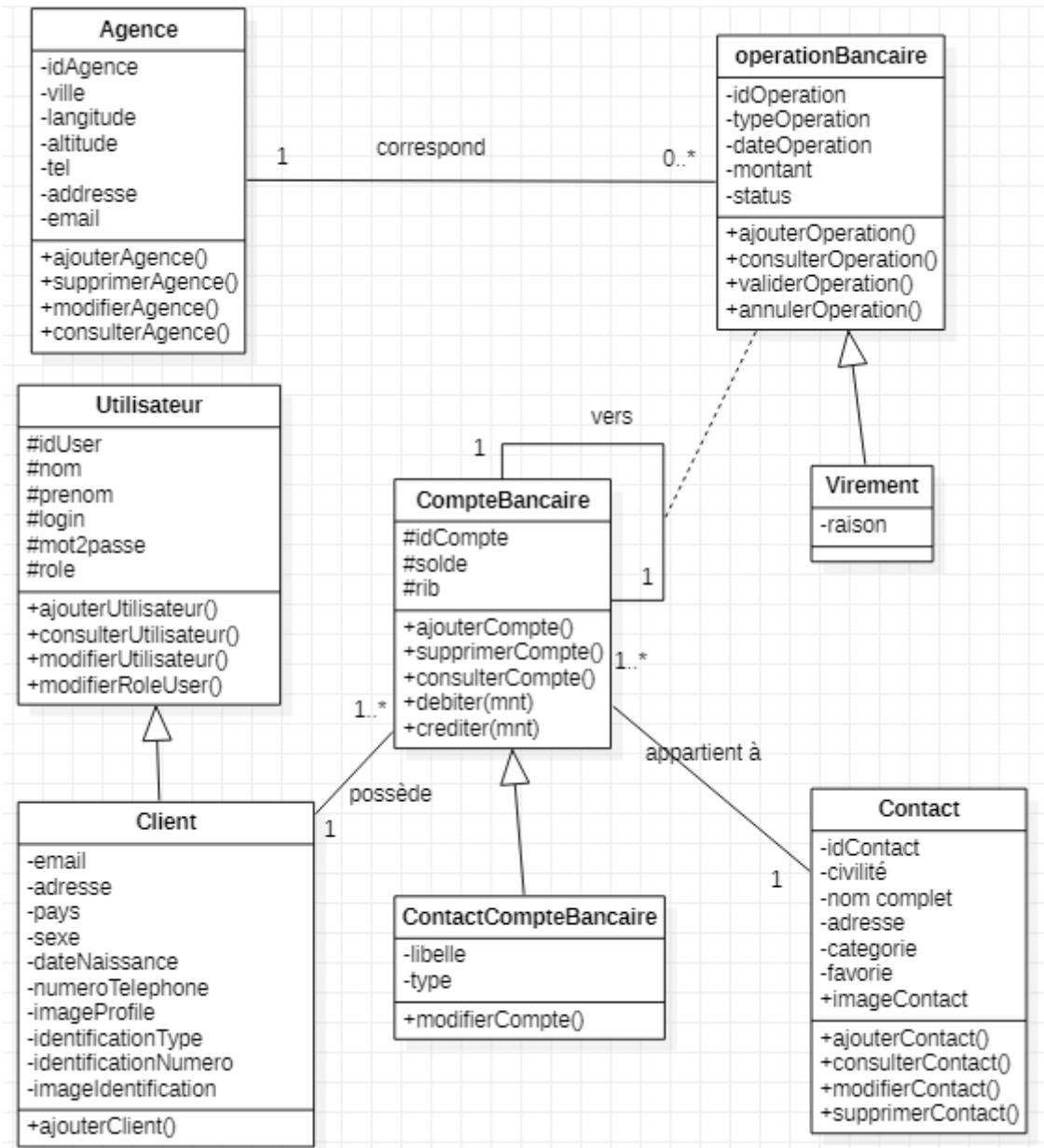


Figure 34 : Diagramme de classes du sprint 2

## 5. Review du sprint 2

Dans cette partie, nous décrirons les principales fonctionnalités offertes par le module transferts à travers des interfaces afin d'illustrer la réalisation du deuxième sprint.

## 5.1. Espace administrateur

Nous commençons par présenter les interfaces de l'espace consacré à l'administrateur.

- **Interface Dashboard**

Dans ce second sprint, l'interface d'accueil de l'administrateur permet de visualiser la liste des dernières transactions effectuées. Elle lui permet aussi de consulter les statistiques des diverses opérations.

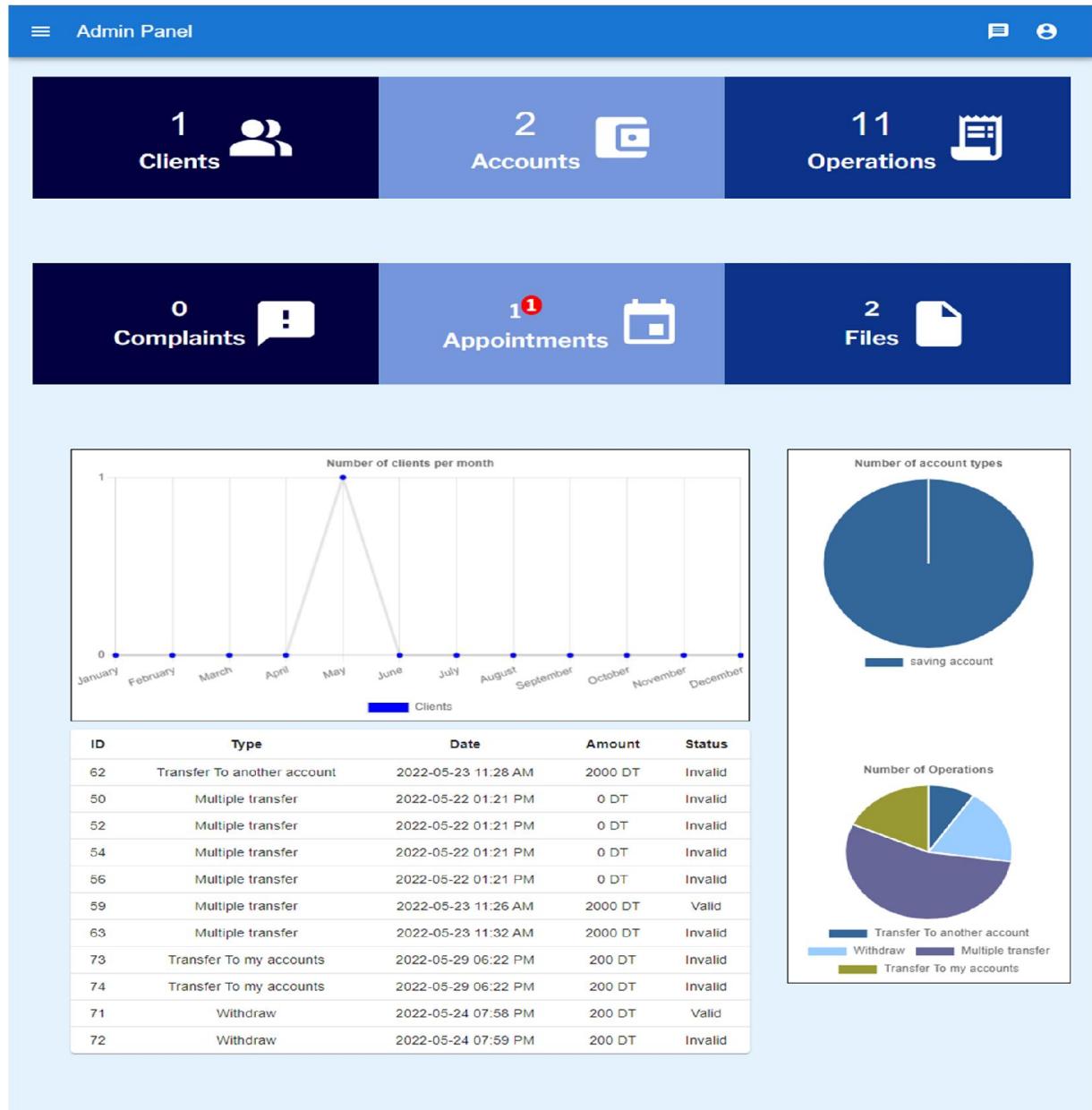


Figure 35 : Interface Dashboard - admin - sprint 2

### • Interface liste des virements

Un administrateur peut utiliser son compte pour gérer les virements en cliquant sur « Bank operations » puis « Transfer » dans le menu à gauche. La liste des virements lui permet de consulter les détails de chacun d'eux et de valider ou annuler un virement.

ID	Operation Type	Debit ID	Debit Type	Credit ID	Credit Type	Amount	Date	Reason	Status	Valid	Cancel
73	Transfer To my accounts	100000	saving account	100001	saving account	200	2022-08-26 06:22 PM	reason	Invalid		
74	Transfer To my accounts	100000	saving account	100001	saving account	200	2022-08-26 06:22 PM	NILL	Invalid		

Figure 36 : Interface liste des transferts

### • Interface validation d'un dépôt

L'administrateur peut aussi gérer les dépôts et les retraits en choisissant cette fois « Deposit/Withdraw » sous « Bank operations » du menu. A partir de la liste des dépôts et retraits, il lui est possible de valider ou annuler une opération après avoir consulté ses détails.

ID	Operation Type	Account ID	Account Type	Amount	Agency	Date	Status	Valid	Cancel
1	Deposit	100000	Saving Account	100	Tunis	2022-04-20 00:54 PM	Invalid		
2	Deposit	100000	Saving Account	500	Tunis	2022-04-20 00:55 PM	Invalid		

Figure 37 : Interface validation d'un dépôt

## 5.2. Espace client

Les interfaces de cette dernière section concernent le client.

- **Interface dépôt et retrait**

L'application permet au client d'effectuer des dépôts et des retraits en choisissant « deposit/withdraw » dans le menu. Une interface s'affiche, le client doit choisir le type d'opération à effectuer puis il doit remplir les données nécessaires, et enfin valider afin de sauvegarder l'opération.

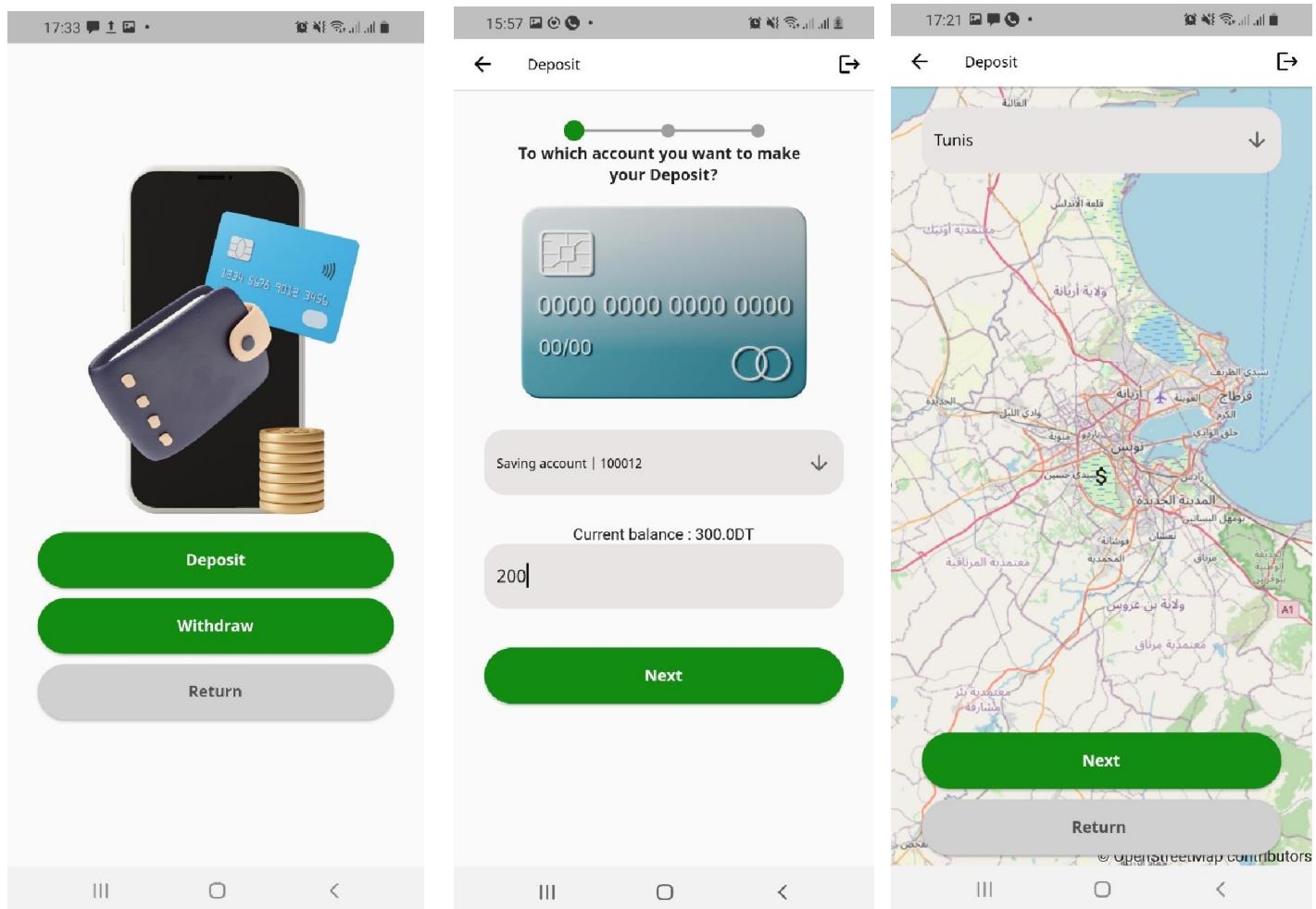


Figure 38 : Interface dépôt et retrait

- **Interface effectuer un virement**

Le client peut effectuer aussi des virements. Pour cela, il doit choisir entre transférer de l'argent vers un autre de ses comptes bancaires ou vers le compte d'un de ses contacts. Ensuite, il remplit les champs nécessaires y compris la raison de ce virement puis il valide.

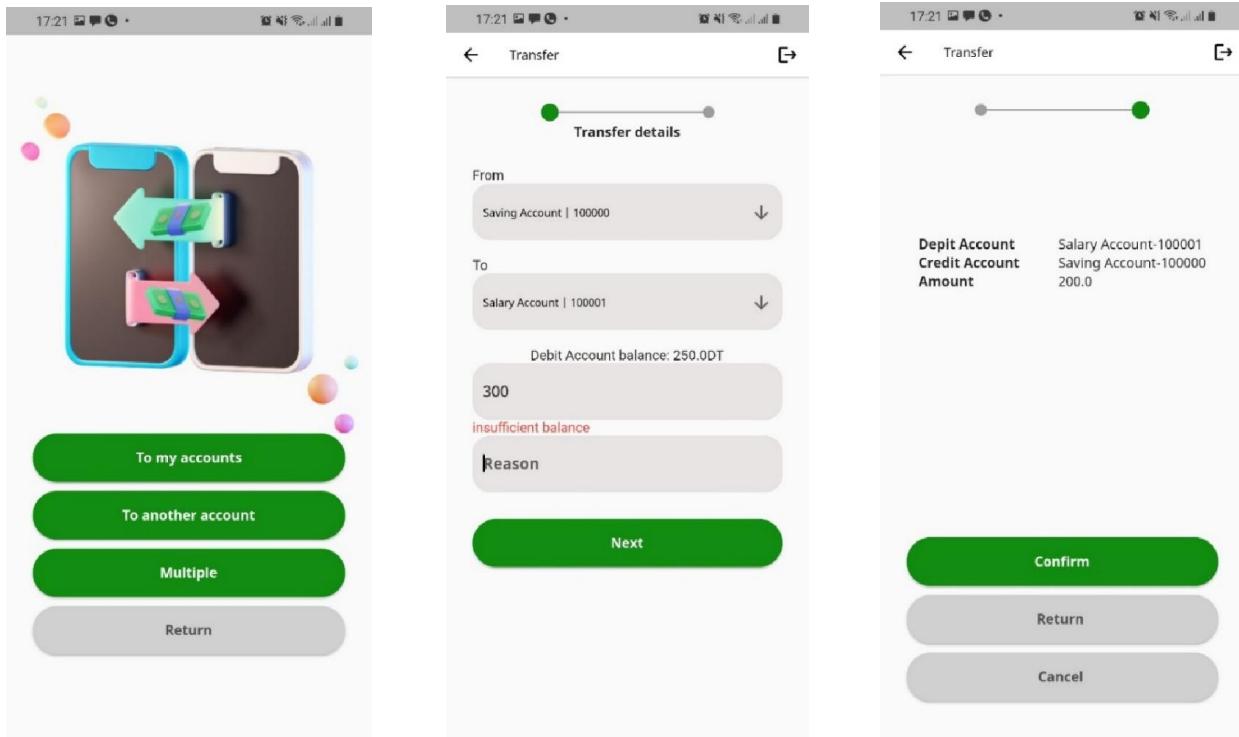


Figure 39 : Interface virement

- **Interface historique**

L'application permet au client de consulter l'historique de ses opérations. Il peut aussi les filtrer selon le type et la date .

History		
Type	Amount	Status
Transfer To another account	2000.0DT	Invalid
Multiple Transfer	0.0DT	Invalid
Multiple Transfer	0.0DT	Invalid
Multiple Transfer	0.0DT	Invalid
Multiple Transfer	2000.0DT	Valid
Multiple Transfer	2000.0DT	Invalid
Withdraw	200.0DT	Valid
Withdraw	200.0DT	Invalid

Figure 40:Interface historique

Le client peut aussi consulter les détails de chaque opération, et l'annuler si elle n'a pas encore été réalisée auprès d'une agence et donc validée par l'administrateur.

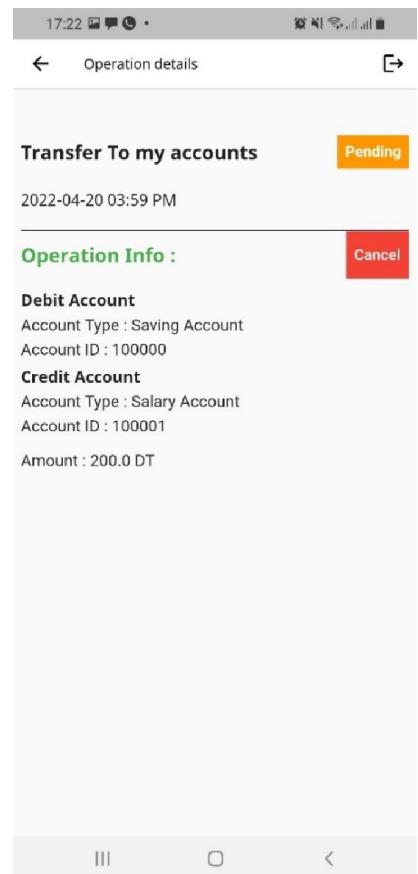


Figure 41 : Interface détails d'une opération

## Conclusion

Nous avons décrit dans ce chapitre le backlog du second sprint et listé les différentes fonctionnalités réalisées à travers les diagrammes des cas d'utilisation. Puis, nous avons présenté les diagrammes de séquences pour définir la description comportementale. Et le diagramme de classes pour décrire l'aspect statique. Enfin, nous avons présenté la partie réalisation en terme d'interfaces des fonctionnalités implémentées dans ce deuxième sprint. Dans le prochain chapitre, nous présenterons le dernier sprint qui comporte la partie communication.

# Chapitre 5 : Sprint 3 – Module communication

## Introduction

Tout au long de ce chapitre nous aurons un sprint à la même vélocité que les sprints précédents nous commençons par le backlog du sprint suivi de l'étude fonctionnelle et comportementale, pour finir avec le diagramme de classes et la réalisation.

### 1. Backlog du sprint 3

Tableau 14: Backlog du sprint 3

Fonctionnalité	ID	User story	Taches	Story point
Gestion des réclamations	1	En tant que admin je veux consulter la liste des réclamations, et pouvoir répondre et changer leurs états pour les archiver.	- Créer la classe réclamation. - Donner le droit à l'admin de consulter et répondre et archiver	4
	2	En tant que client je veux pouvoir envoyer des réclamations au service clientèle et les consulter.	- Donner le droit au client de consulter les réclamations et de les envoyer.	3
Gestion des messages	3	En tant que client je veux discuter avec l'admin en envoyant et lisant des messages.	- Créer la classe message. - Ajouter les attributs et méthodes nécessaires. - Donner le droit au client d'envoyer et consulter ses messages.	3

	4	En tant que admin je veux discuter avec le client en envoyant et lisant des messages.	- Donner le droit à l'admin d'envoyer et de consulter ses messages.	2
Gestion des rendez-vous	5	En tant que clients je veux planifier un rendez-vous le annuler et consulter la liste des mes rendez-vous.	- Créer la classe rendez-vous - Donner le droit au client de consulter la liste des rendez-vous , planifier un rendez-vous et pouvoir le modifier ou l'annuler .	3
	6	En tant que admin je veux consulter la liste des rendez-vous, accepter un rendez-vous ou l'annuler ou suggérer une autre date.	- Donner le droit à l'administrateur de consulter tous les rendez-vous, les accepter ou les annuler, et donner la possibilité de changer la date	3
Gestion des fichiers	7	En tant que client je veux envoyer des fichiers à l'administrateur, les consulter , les modifier ou les supprimer	- Créer la classe fichier. - Ajouter les attributs et les méthodes nécessaire. - Donner le droit au client d'envoyer des fichiers , les consulter , les modifier et supprimer	1
	8	En tant que admin je veux consulter la liste des fichiers, les télécharger et les supprimer.	- Donner le droit à l'administrateur de consulter les fichiers les télécharger ou les supprimer .	1
Gestion des notifications	9	En tant que client je veux recevoir des notifications dans l'application, par SMS et par email, les consulter et les supprimer.	- Créer la classe notification. - Ajouter les méthodes nécessaires. - Donner le droit au client de recevoir et de consulter les notification et de les supprimer	5

## 2. Description fonctionnelle des histoires du sprint 3

Nous décrirons les histoires de la partie développement du sprint 3 en présentant les diagrammes de cas d'utilisation et leurs descriptions textuelles.

### 2.1. Diagrammes de cas d'utilisation

Dans cette section, nous commençons par présenter le diagramme de cas d'utilisation global correspondant au sprint 3, par la suite, nous raffinerons par acteur.

#### 2.2.1. Diagramme de cas d'utilisation global du sprint 3

Comme les précédents sprints, nous présentons le diagramme de cas d'utilisation global qui illustre les fonctionnalités offertes par le module communication

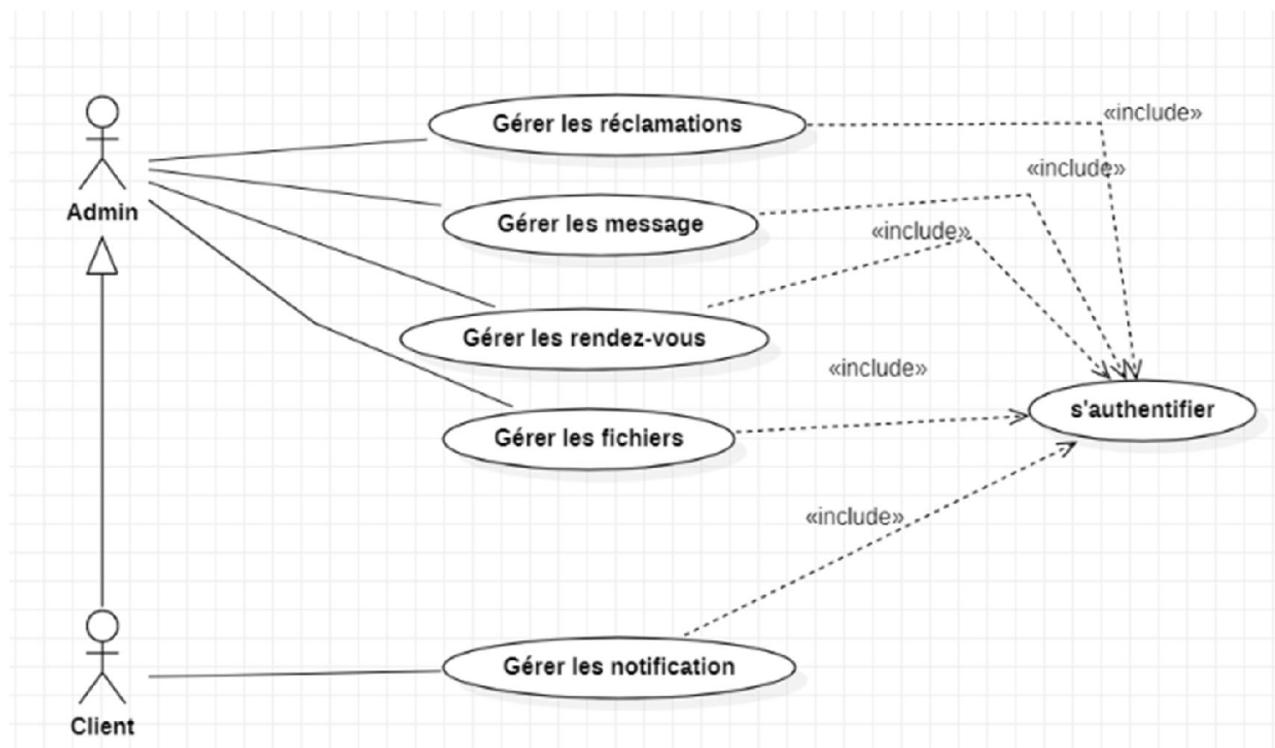


Figure 42: Diagramme de cas d'utilisation globale du sprint 3

## 2.2.2. Diagramme de cas d'utilisation détaillés du sprint 2

On passe au raffinement des diagrammes par acteur.

- Diagramme de cas d'utilisation détaillé de l'administrateur

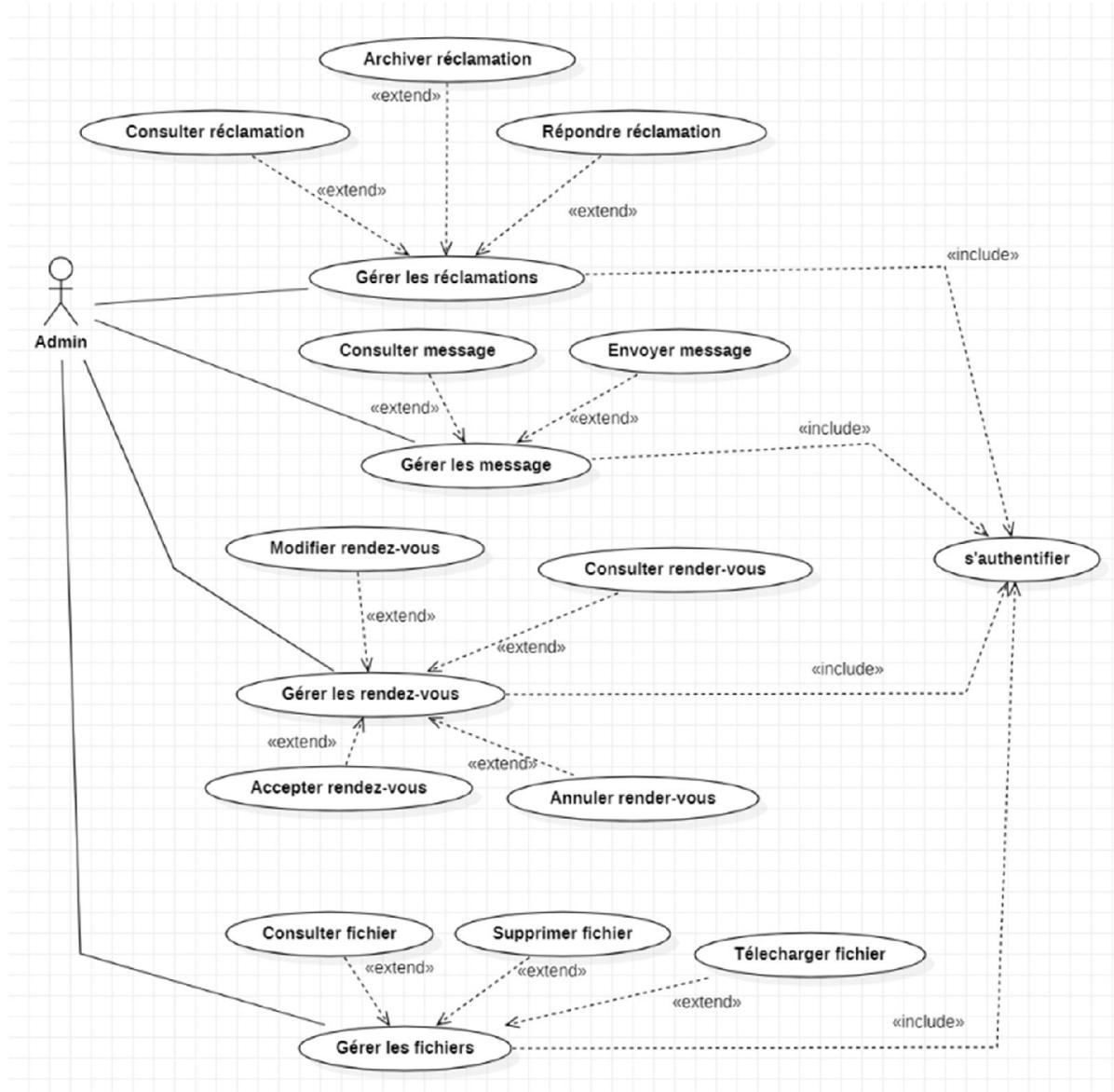


Figure 43: Diagramme de cas d'utilisation détaillé de l'administrateur

- Diagramme de cas d'utilisation détaillé du client

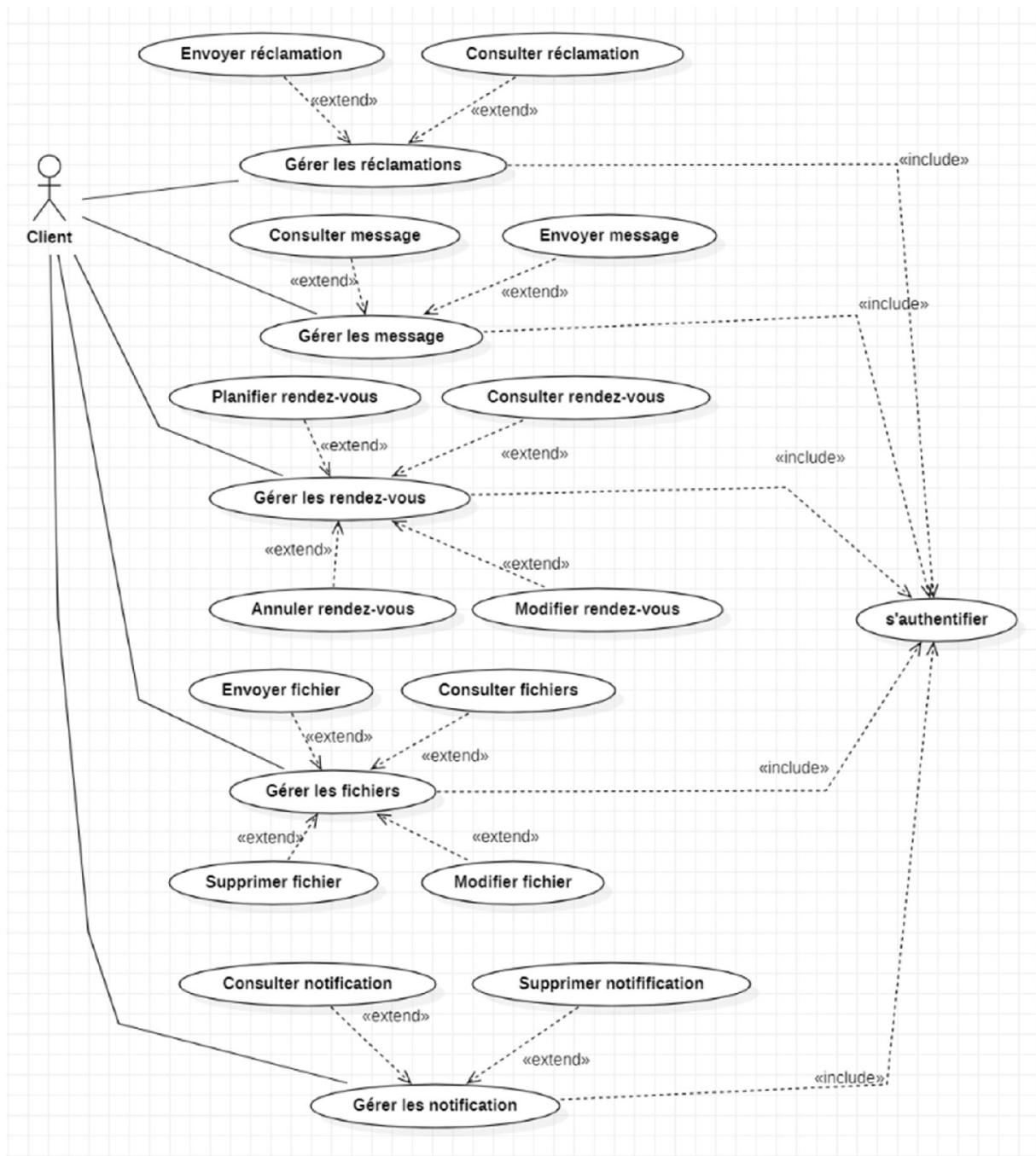


Figure 44: Diagramme de cas d'utilisation détaillé du client

## 2.2. Description textuelle des cas d'utilisation

Nous enchainons à présent avec la description textuelle des cas d'utilisation.

Tableau 15:Description textuelle du cas d'utilisation « Consulter un rendez-vous »

<b>Cas d'utilisation :</b> Consulter un rendez-vous
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Consulter un rendez-vous
<b>Précondition :</b> Le client est authentifié, le rendez-vous existe déjà dans la base.
<b>Description du scénario :</b> <i>Scénario principale (succès) :</i> 1- Le client demande la liste des rendez-vous. 2- Le système affiche un calendrier. 3- Le client choisit une date du calendrier. 4- Le système affiche le détail du rendez-vous. <i>Scénario alternatif :</i> 4.1- Il n'y a aucun rendez-vous.

Tableau 16:Description textuelle du cas d'utilisation « Envoyer un fichier »

<b>Cas d'utilisation :</b> Envoyer un fichier
<b>Acteur initiateur :</b> Client
<b>But du cas :</b> Envoyer un fichier
<b>Précondition :</b> Le client est authentifié, format du fichier valide .
<b>Description du scénario :</b> <i>Scénario principale (succès) :</i> 1- Le client demande le formulaire. 2- Le système affiche le formulaire d'envoi de fichier. 3- Le client choisit le fichier à envoyer. 4- Le système affiche un message du succès. <i>Scénario alternatif :</i> 3.1- Le format du fichier est invalide ou sa taille est grande. 4.1- Le système affiche un message d'erreur

Tableau 17:Description textuelle du cas d'utilisation « Télécharger un fichier »

<b>Cas d'utilisation :</b> Télécharger un fichier
<b>Acteur initiateur :</b> Admin
<b>But du cas :</b> Consulter un fichier
<b>Précondition :</b> L'administrateur est authentifié, le fichier doit être dans la base.
<b>Description du scénario :</b> <i>Scénario principale (succès) :</i> 1- L'admin demande la liste des fichiers. 2- Le système affiche la liste des fichiers. 3- Le client choisit un fichier à télécharger 4- Le système télécharge le fichier. <i>Scénario alternatif :</i> 3.1- La liste des fichiers est vide

Tableau 18:Description textuelle du cas d'utilisation « Consulter une réclamation »

<b>Cas d'utilisation :</b> Consulter une réclamation
<b>Acteur initiateur :</b> Client, Administrateur
<b>But du cas :</b> Consulter les notifications
<b>Précondition :</b> L'utilisateur est authentifié, la réclamation doit être dans la base.
<b>Description du scénario :</b> <i>Scénario principale (succès) :</i> 1- L'utilisateur demande la liste des réclamations. 2- Le système affiche la liste des réclamations. 3- L'utilisateur choisit une réclamation à consulter 4- Le système affiche les détails de la réclamation <i>Scénario alternatif :</i> 3.1- La liste des réclamations est vide.

### 3. Description comportementale des histoires du sprint 3

Ensuite, nous présentons les diagrammes de séquences correspondant aux principaux cas d'utilisation de notre sprint 3.

- **Diagramme de séquence « Consulter un rendez-vous »**

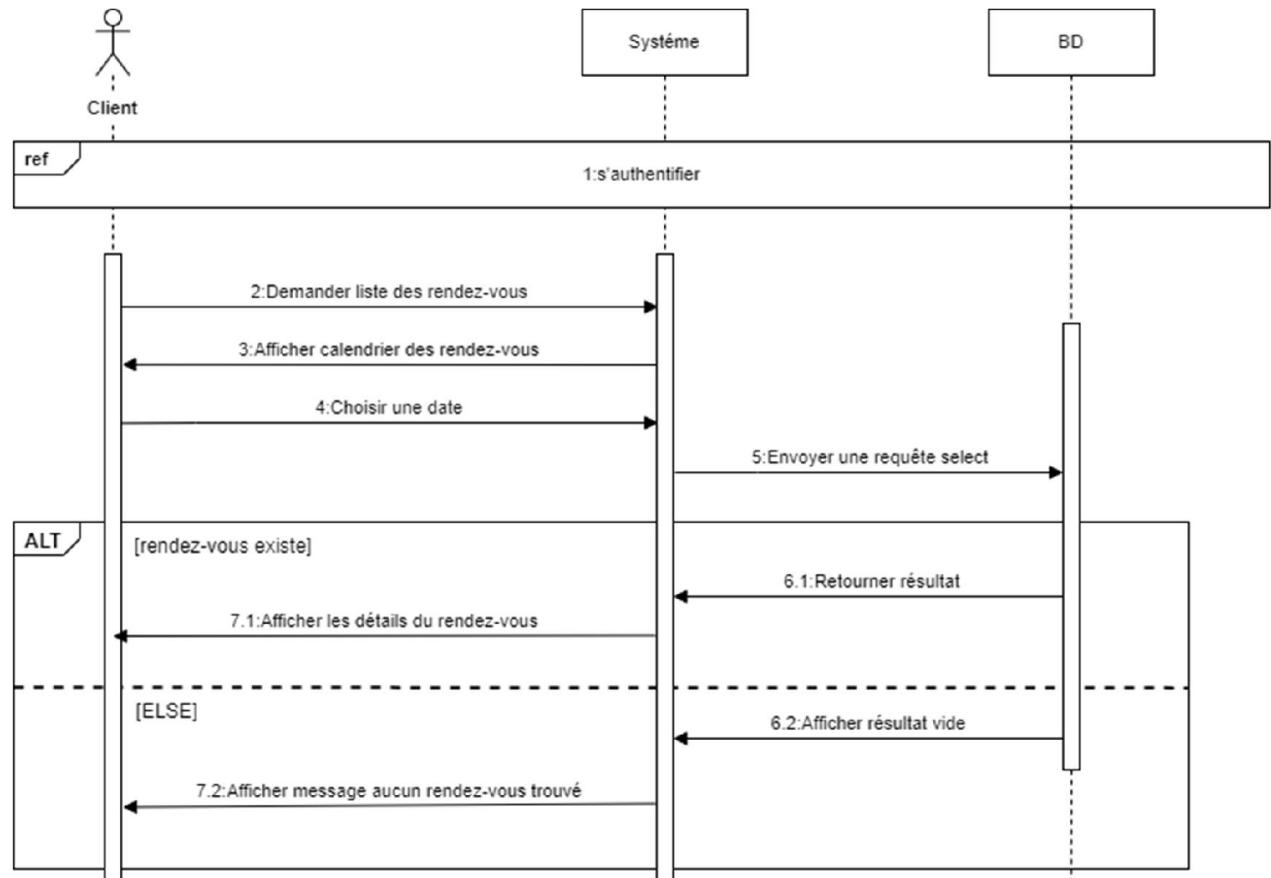


Figure 45 : Diagramme de séquence « Consulter un rendez-vous »

- **Diagramme de séquence « Envoyer un fichier »**

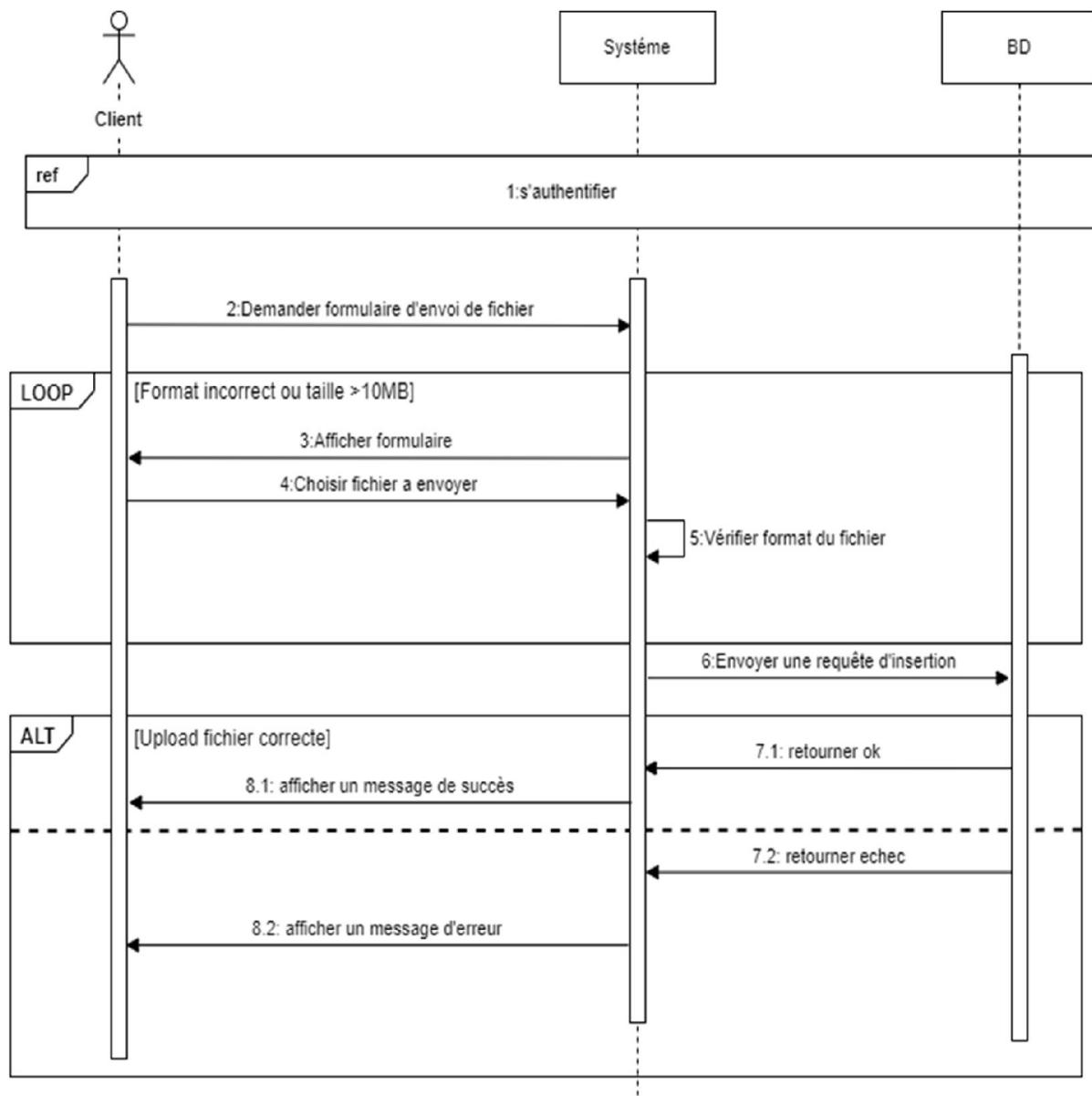


Figure 46 : Diagramme de séquence « Envoyer un fichier »

- **Diagramme de séquence « Télécharger un fichier »**

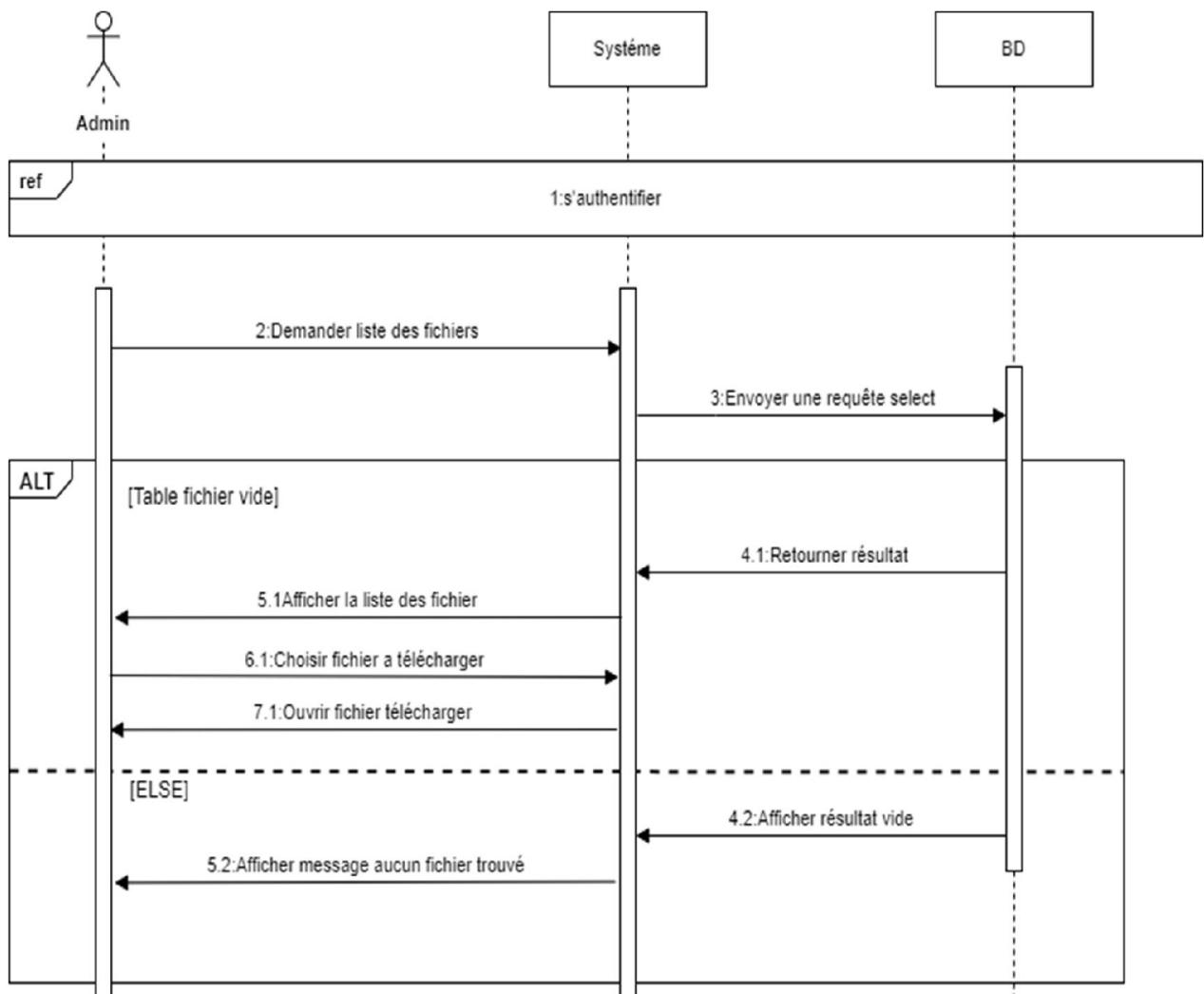


Figure 47: Diagramme de séquence « Télécharger un fichier »

## 4. Diagramme de classes du sprint 3

La figure illustre le diagramme de classes pour le troisième sprint de notre projet. Il comporte 7 classes.

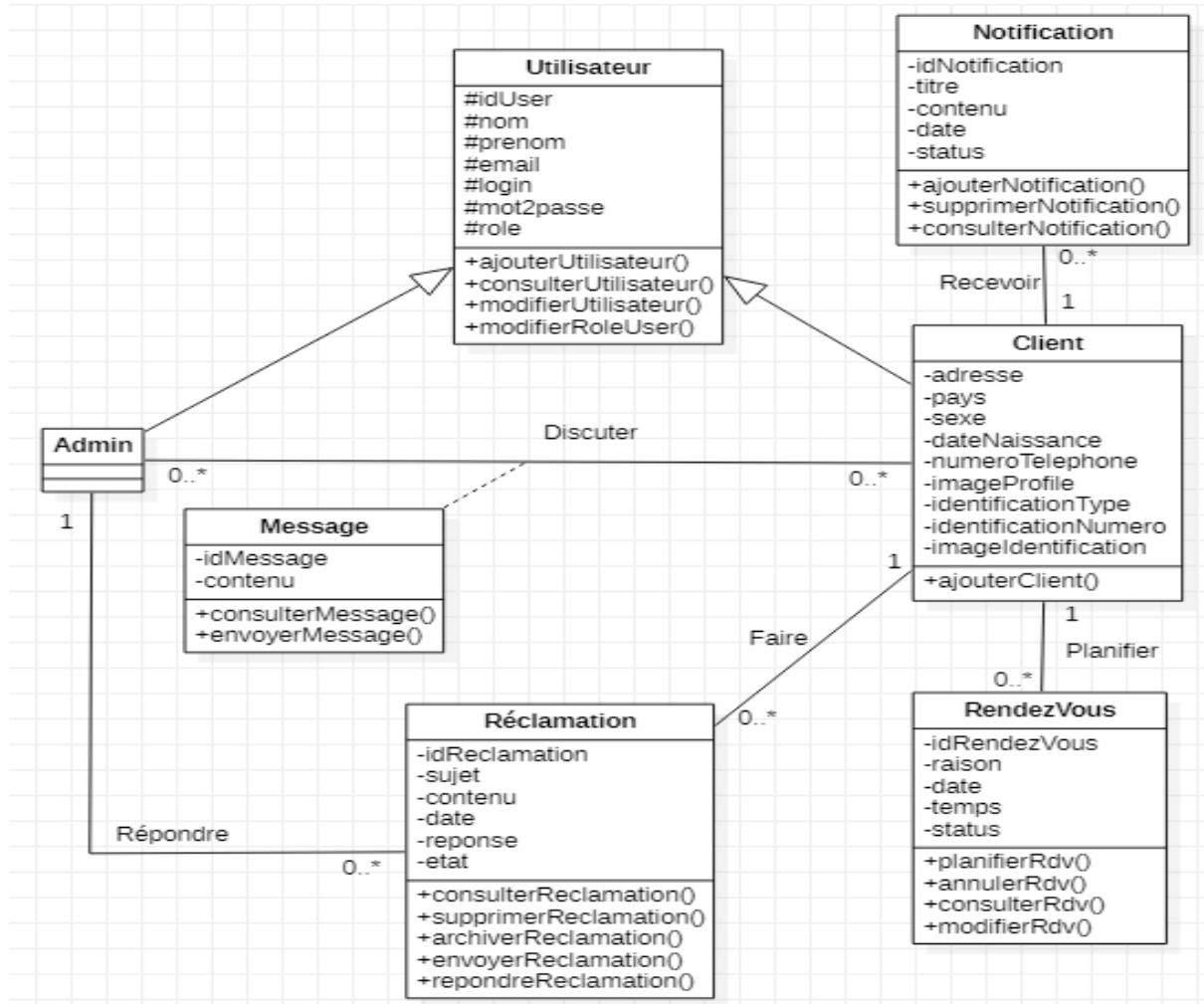


Figure 48: Diagramme de classe sprint 3

## 5. Review du sprint 3

### 5.1. Espace administrateur

Nous commençons par présenter les interfaces de l'espace consacré à l'administrateur

- **Interface réclamation**

A partir de cette interface l'administrateur peut consulter les réclamations envoyées par les clients, il peut les filtrer, y répondre et aussi les archiver.

Figure 49 : Interface réclamation - admin

### • Interface fichiers

En cliquant sur « Files » dans le menu à gauche, une liste de fichiers s'affiche. A partir de cette liste, l'administrateur peut consulter les fichiers envoyés par les clients, il peut les télécharger en cliquant sur l'icône de download, il peut aussi les supprimer.

ID	File name	Client	Download	Delete
1	Bank_ Its Origin, Meaning, Objectives & Function.pdf	Rami Chargui		
2	Bank_ Its Origin, Meaning, Objectives & Function.pdf	Rami Chargui		

Figure 50 : Interface liste des fichiers - admin

- **Interface changer un rendez-vous**

L’option « appointments » dans le menu à gauche permet d’afficher la liste des rendez-vous planifiés par les clients. L’admin peut les accepter ou les annuler. En plus, il a la possibilité de suggérer une autre date comme le montre la figure 50.

ID	Date	Time	Reason	Client	Status	Validate	Change	Refuse
22	06/05/2022	15:00	Meeting	Rami Chargui	waiting			
23	6/5/2022	11:14	Close account	Rami Chargui	waiting			

Figure 51 : Interface rendez-vous - admin

- **Interface messagerie**

L’application permet à l’administrateur de discuter avec ses clients en lisant et envoyant des messages grâce à l’interface de messagerie.

Figure 52 : Interface messagerie - admin

## 5.2.Espace client

Maintenant nous présentons les interfaces du client qui ont été réalisées lors du sprints 3.

- **Menu client**

Après avoir terminé toutes les fonctionnalités de l'application, la figure 52 décrit le menu offert au client.

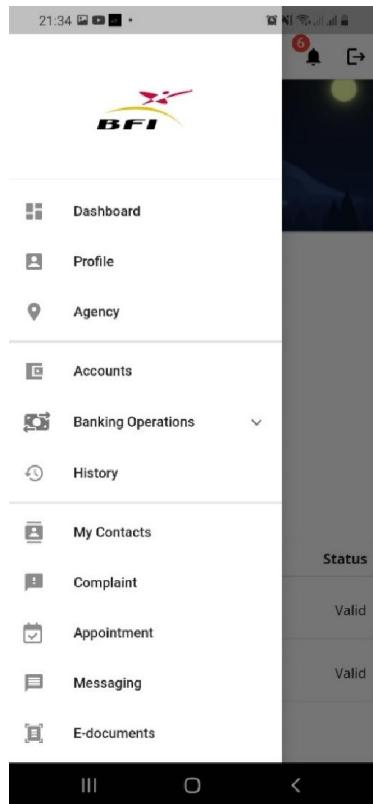


Figure 53: Interface menu - client

- **Interface réclamation**

Le client peut envoyer des réclamations et consulter les réponses de la banque grâce aux interfaces décrites par la figure 53 et obtenue en cliquant sur « Complaints » dans le menu.

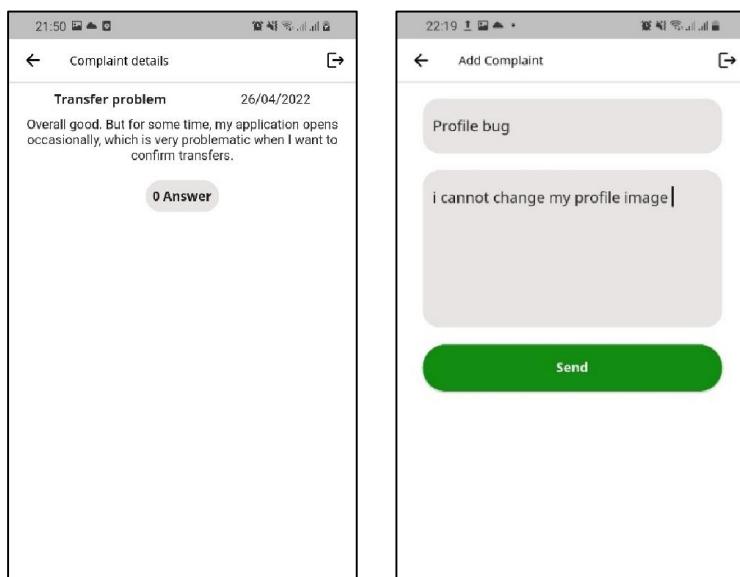


Figure 54: Interface réclamation – client

- **Interface gestion des fichiers**

Lorsqu'un client a besoin d'envoyer des documents numériques à sa banque, il utilise le formulaire de la figure 54. Il lui suffit d'indiquer l'objet et la description du document et de télécharger le fichier en format PDF ou Word. Ensuite, il lui est possible de consulter les fichiers envoyés, de les modifier (les détails ou bien le fichier en question) ou de les supprimer.

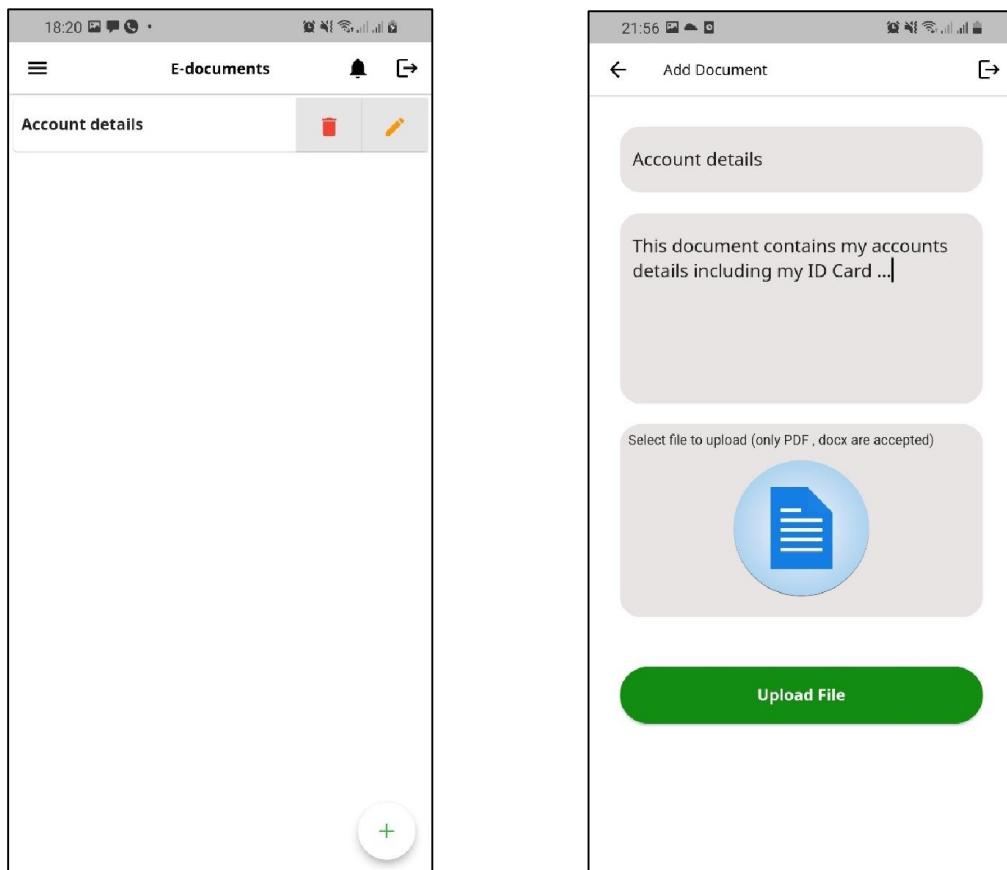


Figure 55 : Interface fichier - client

- **Interface rendez-vous**

Grace à cette interface le client peut consulter la liste des rendez-vous. En effet, il lui suffit de cliquer sur une date marquée en rouge pour afficher le rendez-vous planifié et son état (en attente, validé). Il peut également demander un rendez-vous en appuyant longtemps sur le jour désiré afin d'afficher le formulaire d'ajout de rendez-vous. Dans ce formulaire le client indique l'heure et la raison pour le rendez-vous.

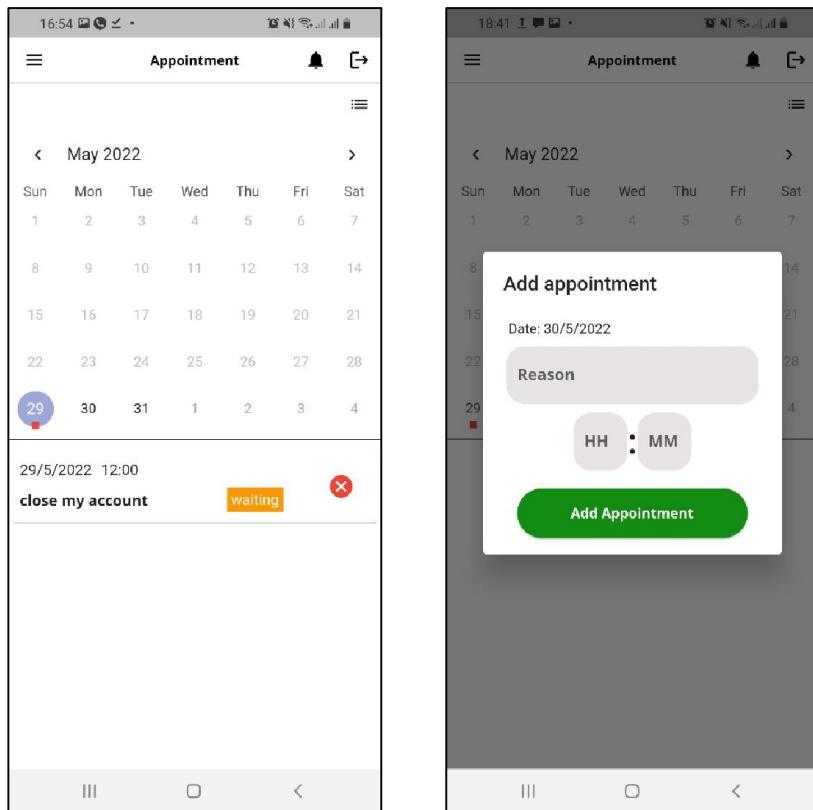


Figure 56: Interface rendez-vous - client

- **Interface messagerie**

Le client peut aussi discuter avec un représentant de la banque (un administrateur) à travers l'échange de messages. La figure 56 affiche un exemple de discussion.

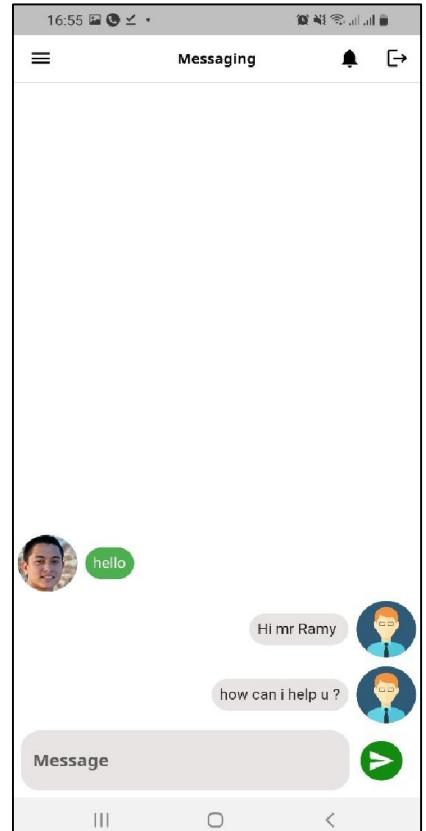


Figure 57: Interface messagerie - client

- **Interface notification**

L’application envoie automatiquement des notifications aux clients lors de l’activation du compte bancaire, la validation d’une transaction bancaire, ou de l’acceptation d’une demande de rendez-vous. La figure 57 montre la liste de notification reçues.

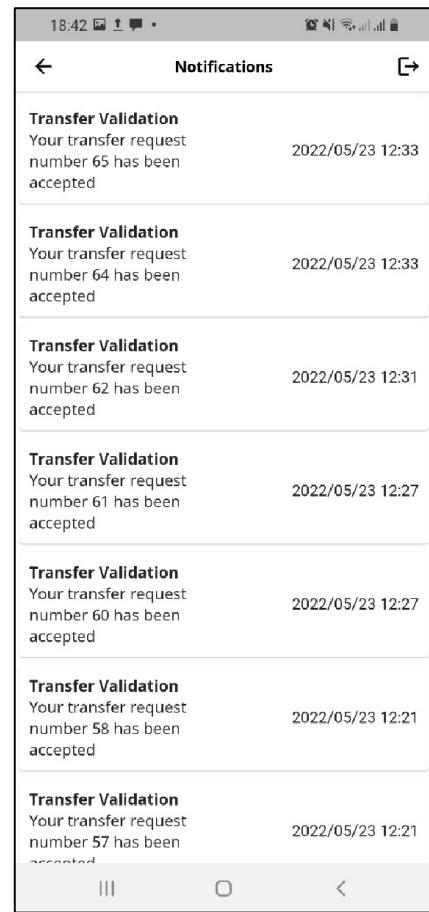


Figure 58 : Interface notification

## Conclusion

Dans ce dernier chapitre, nous nous sommes concentrés sur la conception et le développement de notre troisième et dernier sprint. Nous avons donc abouti à un produit fonctionnel, intégral et livrable.

## Conclusion générale

Dans le cadre de notre projet de fin d'étude, nous avons conçu et développé une application d'une banque digitale. Le présent manuscrit a détaillé toutes les étapes par lesquelles nous sommes passées pour arriver au résultat attendu. Nous avons essayé tout au long de notre travail de construire notre application incrément par incrément en utilisant la méthodologie scrum.

Nous avons commencé dans un premier lieu par comprendre le contexte général de notre application et identifier les différentes exigences de notre futur système. Nous avons préparé par la suite notre planning de travail en respectant les priorités. Puis, nous avons entamé la conception et la réalisation de chaque sprint en organisant des réunions avec le scrum master notamment pour valider chaque release et faire la rétrospective de chaque sprint.

Malgré toutes les difficultés rencontrées au niveau du processus métier et les contraintes de temps, nous avons réussi à réaliser la totalité de notre application tout en respectant l'aspect sécuritaire et en préparant la documentation nécessaire.

Ce travail a été très enrichissant puisqu'il nous a permis de découvrir un nouveau domaine de travail et de s'éloigner des projets académiques traditionnels. Il nous a aussi permis d'approfondir nos connaissances en programmation Web avec React et mobile avec Flutter.

Finalement, bien que notre application offre déjà plusieurs fonctionnalités, il est possible de l'enrichir encore. En effet divers modules peuvent être intégrés notamment un système de paiement de facture ou bien un module chatbot doté d'une Intelligence Artificielle pour remplacer la messagerie avec le service clientèle et répondre automatiquement aux questions des clients.

# Webographie

[1] Définition et principe des méthodes agiles [En ligne] <https://www.piloter.org/projet/methode/methode-agile.htm> [Consulté le 16/02/2022].

[2] Visual Studio Code [En ligne] <https://blog.webnet.fr/visual-studio-code/> [Consulté le 28/02/2022].

[3] MySQL [En ligne]

<https://kinsta.com/fr/base-de-connaissances/qu-est-ce-que-mysql/> [Consulté le 28/02/2022].

[4] Postman c'est quoi ? [En ligne] <https://www.eewee.fr/postman-cest-quoi/> [Consulté le 28/02/2022].

[5] Adobe XD [En ligne] <https://www.usabilis.com/adobe-xd-prototypage/> [Consulté le 28/02/2022].

[6] Qu'est ce que JIRA ? [En ligne] <https://www.cadremploi.fr/editorial/conseils/conseils-carriere/detail/article/jose-pas-demander-mais-quest-ce-que-jira.html> [Consulté le 28/02/2022].

[7] Qu'est ce que Firebase Cloud Messaging ? [En ligne] <https://blog.back4app.com/fr/quest-ce-que-firebase-cloud-messaging/> [Consulté le 16/02/2022].

[8] Dart [En ligne] <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/le-langage-de-programmation-dart/> [Consulté le 28/02/2022].

[9] Qu'est-ce que la technologie Java et pourquoi en ai-je besoin ? [En ligne] [https://www.java.com/fr/download/help/whatis\\_java.html](https://www.java.com/fr/download/help/whatis_java.html) [Consulté le 28/02/2022].

[10] Qu'est-ce que le JavaScript ? [En ligne] [https://developer.mozilla.org/fr/docs/Learn/JavaScript/First\\_steps/](https://developer.mozilla.org/fr/docs/Learn/JavaScript/First_steps/) [Consulté le 28/02/2022].

[11] Flutter [En ligne] <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/flutter-cest-quoi/> [Consulté le 28/02/2022].

[12] Qu'est-ce que spring boot ? [En ligne] <https://www.axopen.com/spring-boot-lyon/> [Consulté le 28/02/2022].

[13] React [En ligne] <https://www.50a.fr/0/react> [Consulté le 28/02/2022].

[14] Preuve d'authentification avec JWT [En ligne] <https://blog.ippon.fr/2017/10/12/preuve-dauthentification-avec-jwt/> [Consulté le 28/02/2022].

[15] WebSockets [En ligne] [https://developer.mozilla.org/fr/docs/Web/API/WebSockets\\_API](https://developer.mozilla.org/fr/docs/Web/API/WebSockets_API) [Consulté le 28/02/2022].

[16] Qu'est qu'un backlog ? [En ligne] <https://hubvisory.com/blog/qu-est-qu-un-backlog-comment-le-construire-et-le-gerer/> [Consulté le 24/03/2022].

[17] Diagramme de Use case [En ligne] <https://www.uml-sysml.org/diagrammes-uml-et-sysml/diagramme-uml/use-case-diagramme/> [Consulté le 18/03/2022].

[18] Diagramme de séquence [En ligne] [Création de diagrammes de séquence UML - IONOS](#) [Consulté le 30/03/2022].

[19] Diagramme de classes [En ligne] <https://www.lucidchart.com/pages/fr/diagramme-de-classes-uml> [Consulté le 30/03/2022].