

PROJET DE FIN D'ETUDES

En vue de l'obtention du diplôme

D'INGENIEUR D'ETAT

En Ingénierie Informatique et Technologies Emergentes (2ITE)

Création d'une application web pour la gestion des remboursements des avoirs complexes

Réalisé à : Intelcia IT Solutions

Réalisé par :

Ibrahim CHAHBOUNE

Encadré par :

Prof. Mohamed HANINE

Mme. Fadwa ZARHOURY

Soutenu le 15/06/2023 devant le jury :

Prof. Rahhal ERRATTAHI

Prof. Chafik BAIDADA

Année Universitaire 2022/2023

Dédicaces

Je dédie ce modeste travail à :

Mes parents, pour leur amour, leur soutien et leurs sacrifices. Sans eux, je n'aurais pas pu arriver jusqu'ici et je suis reconnaissant de tout ce qu'ils ont fait pour moi.

Mes enseignants et mes encadrants, pour leurs précieux conseils et leur accompagnement tout au long de ce projet. Leurs encouragements et leur expertise m'ont permis d'aller au-delà de mes limites et de réaliser ce projet avec succès.

Ma famille et mes amis, pour leur soutien moral et leur compréhension pendant cette période de travail intense. Votre présence et votre soutien ont été une source de motivation pour moi.

Toutes personnes qui croient en moi et en mes capacités. Vos encouragements et votre confiance ont été un véritable moteur pour moi tout au long de ce projet, et je suis reconnaissant pour votre soutien indéfectible.

Remerciements

C'est un devoir agréable d'exprimer en quelques lignes la reconnaissance qu'on doit à tous ceux dont on a sollicité l'aide et la collaboration.

Je souhaite exprimer ma profonde gratitude envers le corps professoral de la filière Ingénierie Informatique et Technologies Emergentes (2ITE) de l'École Nationale des Sciences Appliquées d'El Jadida. Leur enseignement de qualité, leur dévouement et leur passion pour leur domaine ont été une source d'inspiration tout au long de mes années de formation. Grâce à eux, j'ai acquis des compétences solides et une base de connaissances précieuse qui me seront bénéfiques tout au long de ma carrière. Je tiens à remercier chaque professeur pour leur investissement dans notre réussite académique, leurs encouragements et leurs conseils qui ont été inestimables pour surmonter les défis. Je suis honoré d'avoir été formé par une équipe professorale aussi talentueuse.

J'exprime aussi mes vifs remerciements aux membres du jury, Professeurs M. Rahhal ERRATTAHI et M. Chafik BAIDADA pour l'intérêt qu'ils ont porté à mon sujet de fin d'études en acceptant d'examiner mon travail et de l'enrichir par leurs précieuses remarques.

Je tiens également à exprimer ma profonde gratitude envers mon encadrant pédagogique Prof. Mohamed HANINE pour son soutien inconditionnel tout au long de mon projet de fin d'études. Je suis reconnaissant pour la qualité de l'accompagnement qu'il m'a offert, et je ne saurais jamais assez le remercier pour sa patience, sa disponibilité et sa bienveillance.

Ainsi, je tiens à remercier l'encadrante Mme. Fadwa ZARHOURY pour ses précieux conseils et les orientations qui sont un grand secours tout au long de la période de réalisation et de conception du projet de stage de fin d'études. Elle a toujours été disponible pour moi et m'a bien accueillie malgré son emploi du temps chargé.

Je profite de cette tribune pour remercier Mr. Soufiane EL AMMARI, Mme Gwladys DEMBELLE et toutes les personnes qui de passage, ont pu m'apporter leur contribution, que ce soit au niveau des idées qu'à celui des conceptions. Qu'ils trouvent ici l'expression de mes sincères reconnaissances.

Je remercie Dieu tout-Puissant de m'avoir permis de mener à terme ce travail qui est pour moi le point de départ d'une merveilleuse aventure, celle de la recherche, source de remise en cause permanente et de perfectionnement perpétuel.

Liste des abréviations

JWT	JSON Web Token
S3	Simple Storage Service
SASS	Syntactically awesome stylesheets
UML	Unified Modeling Language
SCRUM	Systematic Customer Resolution Unraveling Meeting
MCD	Modèle conceptuel des données

Tableau 1: Liste des abréviations

Résumé

Dans le cadre de ce projet de stage de fin d'études, j'ai eu l'opportunité de travailler sur une application web de gestion des remboursements des avoirs pour l'entreprise SFR, un fournisseur de services de télécommunications. L'application vise à simplifier le processus de remboursement en gérant les avoirs accordés aux clients lorsqu'ils effectuent des retours de produits ou décident de convertir ces avoirs en remboursement. Cette application web a pour objectif de centraliser la gestion du remboursement des avoirs, de suivre leur historique de remboursements et de simplifier le processus de remboursement. Elle offre également à l'entreprise la possibilité de suivre les remboursements en attente.

Dans le contexte de cette application web, un avoir est considéré comme une entité contenant des informations telles que le montant, le solde, la date de création et le client associé. Les clients ont la possibilité d'utiliser ces avoirs lors de leurs achats futurs.

La réalisation de ce projet revêt une importance stratégique pour l'entreprise, car il vise à optimiser la gestion de remboursements des avoirs, offrant ainsi un meilleur service client. Grâce à cette application web, l'entreprise pourra répondre aux attentes de ses clients de manière plus efficace et améliorer leur expérience globale.

Afin de garantir une conception efficace et une bonne gestion du projet, nous avons opté pour la méthode de gestion de projet Agile. Cette méthode permet de travailler de manière collaborative et d'adapter le projet en fonction des retours des utilisateurs.

Le présent rapport représente le travail accompli et se compose de cinq chapitres qui retracent les différentes étapes du projet, de l'analyse des besoins à la réalisation finale du projet.

Mots clé: SCRUM, JWT, UML, Angular, Spring Boot, Amazon S3, NgRx

Abstract

As part of this final year internship project, I had the opportunity to work on a web application for managing refunds and credits for SFR, a telecommunications service provider. The application aims to streamline the refund process by managing the credits granted to customers when they return products or choose to convert them into refunds. This web application aims to centralize the management of refunds and track their refund history, simplifying the entire process. It also provides the company with the ability to track pending refunds.

In the context of this web application, a credit is considered as an entity containing information such as the amount, balance, creation date, and associated customer. Customers have the option to use these credits for future purchases.

The completion of this project holds strategic importance for the company as it aims to optimize the management of refunds and provide enhanced customer service. Through this web application, the company will be able to effectively meet customer expectations and improve their overall experience.

To ensure efficient design and project management, we adopted the Agile project management methodology. This methodology allows for collaborative work and enables project adaptation based on user feedback.

This report represents the completed work and consists of five chapters that trace the various stages of the project, from needs analysis to the final implementation.

Keywords: SCRUM, JWT, UML, Angular, Spring Boot, Amazon S3, NgRx

Table of Contents

Dédicaces	2
Remerciements	3
Liste des abréviations	4
Résumé	5
Abstract	6
Table of Contents	7
Liste des figures	10
Liste des tableaux	12
Introduction	13
Chapitre 1: Contexte du projet	14
1.1 Introduction	14
1.2 Présentation de l'organisme d'accueil	14
1.3 Services d'Intelcia	15
1.4 Le pôle IT Solutions	15
1.4.1 Services d'Intelcia IT Solutions	16
1.4.2 Clients Intelcia IT Solutions.....	18
1.4.3 Structure organisationnelle.....	19
1.5 Présentation du projet	19
1.5.1 La Présentation de SFR	20
1.5.2 Objectif du projet	20
1.6 Description des services externes en interaction avec le projet.....	20
1.6.1 Service d'authentification	20
1.6.2 Service d'avoir	21
1.7 Conclusion	21
Chapitre 2: Etude préalable	22
2.1 Introduction	22
2.2 Etude préalable	22
2.3 Critique de l'existant	22
2.4 But de la solution proposée	23
2.4.1 Apport sur le plan technique	23

2.4.2	Apport sur le plan fonctionnel.....	23
2.5	L'identification des acteurs	24
2.6	Besoins fonctionnels.....	24
2.7	Spécification non fonctionnelle	26
	Conclusion.....	27
Chapitre 3:	Méthodologie	28
3.1	Introduction	28
3.2	Choix de la méthode	28
3.3	La méthode SCRUM	28
3.4	Le processus SCRUM	29
3.5	Définition de période Sprint	29
3.6	Identification des sprints.....	30
3.7	Le backlog du produit.....	30
3.7.1	Histoire utilisateurs	30
3.7.2	Sprint Backlog.....	32
3.8	Planification du projet	33
3.9	Conclusion	34
Chapitre 4:	Conception	35
4.1	Introduction	35
4.2	Architecture applicative.....	35
4.3	Diagramme de classe	36
4.3.1	Diagramme de classe de l'application.....	36
4.3.2	Dictionnaire de données	38
4.4	Diagrammes de cas d'utilisation.....	39
4.4.1	Diagramme de cas d'utilisation pour un responsable ou agent de point de vente ...	40
4.4.2	Diagramme de cas d'utilisation pour un agent comptable	41
4.5	Diagramme de séquence.....	41
4.5.1	Diagramme de séquence d'authentification	42
4.5.2	Diagramme de séquence du changement du statut.....	43
4.6	Diagramme d'activité	43
4.6.1	Diagramme d'activité du cas de remboursement sur facture	44
4.6.2	Diagramme d'activité du cas de la transformation d'avoir en remboursement	46

4.7 Conclusion	47
Chapitre 5: Réalisation	48
5.1 Introduction	48
5.2 Outils et technologie	48
5.3 Approche de développement	50
5.4 Présentation de l'application web	50
5.5 Conclusion	60
Conclusion et perspectives	61
Bibliographie	62

Liste des figures

Figure 1.1: Présence d'Intelcia à l'internationale	14
Figure 1.2: Solution offertes par Intelcia	15
Figure 1.3: Service d'infogérance.....	16
Figure 1.4: Service des solutions applicatives	16
Figure 1.5: Services de Business Intelligence	17
Figure 1.6: Service de Consulting	17
Figure 1.7: Clients d'Intelcia It Solutions.....	18
Figure 1.8: Structure organisationnelle	19
Figure 3.1: Principe de la méthode SCRUM.....	29
Figure 3.2 Identification des sprints	30
Figure 3.3: Diagramme de Gantt.....	34
Figure 4.1: Architecture générale de l'application	36
Figure 4.2 : Diagramme de classe de l'application	37
Figure 4.3 : Diagramme de cas d'utilisation de l'agent et le responsable de point de vente	40
Figure 4.4 : Diagramme de cas d'utilisation pour un agent comptable.....	41
Figure 4.5 : Diagramme de séquence d'authentification.....	42
Figure 4.6:Diagramme de séquence du changement du statut	43
Figure 4.7: Diagramme d'activité du cas de remboursement sur facture	45
Figure 4.8: Diagramme d'activité du cas de la transformation de l'avoir en remboursement	46
Figure 5.1: Page d'accueil.....	51
Figure 5.2: Page d'accueil - Changement du point de vente	51
Figure 5.3 : Page liste des demandes de remboursements	52
Figure 5.4 : Page liste des demandes de remboursements – filtrage.....	52
Figure 5.5 : Page liste des demandes de remboursements – changement de statues en masse	53
Figure 5.6 : Page liste des demandes de remboursements – confirmation du changement du statue en masse	53
Figure 5.7: Page liste des demandes de remboursements - création de la demande	54
Figure 5.8: Page détail.....	54
Figure 5.9: Page détail - Client modifié	55
Figure 5.10: Modification - Informations de la demande de remboursement.....	55
Figure 5.11: Modification - Informations client.....	56
Figure 5.12: Modification - Ajout des pièces justificatives	56
Figure 5.13: Modification - Ajout d'un commentaire	57
Figure 5.14: Modification – Récapitulatif.....	57
Figure 5.15: Liste des pièces justificatives.....	58
Figure 5.16: Visualiser le fichier - en cours de chargement.....	58
Figure 5.17: Visualiser un fichier.....	59
Figure 5.18: Invalider un fichier	59
Figure 5.19: Fichier Invalide	60

Figure 5.20: Historique des actions60

Figure 5.21: Liste des commentaires.....60

Liste des tableaux

Tableau 1: Liste des abréviations	4
Tableau 3.1: Backlog du produit pour l'agent et le responsable de point de vente	31
Tableau 3.2: Backlog du produit pour l'agent comptable.....	32
Tableau 3.3 : Sprint backlog	33
Tableau 4.1: Dictionnaire de la table Refund.....	38
Tableau 4.2: Dictionnaire de la table Event	38
Tableau 4.3: Dictionnaire de la table Comment.....	38
Tableau 4.4 : Dictionnaire de la table File	38
Tableau 4.5: Dictionnaire de la table Customer	39
Tableau 4.6: Dictionnaire de la table AuditingEntity	39
Tableau 5.1 : Technologies utilisées coté frontend	48
Tableau 5.2: Technologies utilisées coté backend	49
Tableau 5.3: Liste des outils de l'environnement de travail	49

Introduction

L'industrie des services financiers est un secteur en constante évolution qui nécessite des solutions technologiques avancées pour répondre aux besoins croissants des clients. Dans ce contexte, la gestion des remboursements des avoirs est une préoccupation majeure pour les entreprises qui cherchent à offrir une expérience client optimale tout en respectant les réglementations en vigueur.

La création d'une application web dédiée à la gestion des remboursements des avoirs représente donc un enjeu majeur pour les entreprises souhaitant se démarquer sur le marché en offrant des solutions innovantes et efficaces à leurs clients. Cette application doit être capable de traiter efficacement un grand volume de données complexes tout en offrant une interface utilisateur conviviale et facile à utiliser.

Dans ce contexte, ce projet de création d'une application web pour la gestion des remboursements des avoirs vise à répondre à cette problématique en offrant une solution efficace et fonctionnelle pour la gestion des remboursements des avoirs. Ce projet s'inscrit dans une démarche d'optimisation des processus métiers en offrant une solution innovante et adaptée aux besoins de l'entreprise dans le secteur des services financiers.

Ce projet a permis d'explorer différentes questions relatives à la problématique de la construction de systèmes d'informations, notamment les étapes du cycle de vie d'un projet informatique, les différents outils et langages à utiliser selon le type d'application et les domaines envisagés, ainsi que les démarches à suivre pour réussir ce genre de travail.

Le présent document est organisé en chapitre comme suite : Le premier chapitre consiste à présenter l'organisme d'accueil et aborder le contexte général du projet en mentionnant tous les points nécessaires concernant l'application, Dans le deuxième nous abordons l'étude préalable du projet dans laquelle on identifie les acteurs et on précise les fonctionnalités de l'application suite à une spécification des besoins fonctionnels et non fonctionnels, autrement dit c'est une précision des exigences des différents utilisateurs de notre application. Cependant, le troisième chapitre se concentre sur la méthodologie de travail adoptée, mettant en évidence l'approche agile SCRUM. Il fournit une description détaillée des différentes itérations du projet, ainsi que la planification des sprints et de définition du backlog. D'autre part, le quatrième chapitre est consacré à la phase de la conception où on présente un modèle conceptuel de l'application et on identifie les différents acteurs du système ainsi qu'on présente les différents diagrammes décrivant l'architecture interne du système à réaliser.

Finalement, le dernier chapitre intitulé réalisation est dédié à la présentation des technologies utilisées lors de la réalisation de l'application, ainsi les différentes interfaces de cette application, c'est un chapitre de la mise en valeur de tout le processus du développement de l'application.

Chapitre 1: Contexte du projet

1.1 Introduction

Actuellement, les services financiers constituent un secteur en constante évolution qui requiert des solutions technologiques avancées pour répondre aux besoins croissants des clients. La gestion des remboursements des avoirs est une préoccupation majeure pour les entreprises cherchant à offrir une expérience client optimale tout en se conformant aux réglementations en vigueur. C'est dans ce contexte que s'inscrit notre projet de création d'une application web dédiée à la gestion des remboursements des avoirs. Cette première partie du rapport donne une vision globale du projet, son cadre général ainsi qu'une description détaillée des services externes en interaction avec l'application.

1.2 Présentation de l'organisme d'accueil

INTELCIA Group est le premier groupe marocain spécialisé dans les métiers de l'externalisation des services avec une démarche de partenaire à valeur ajoutée. Il intervient dans les secteurs télécoms, médias, assurances, e-commerce et d'autres secteurs. Créé il y a 20 ans au Maroc et porté par une ambition forte, le Groupe a rapidement étendu sa présence géographique et ses activités pour accompagner ses clients sur l'ensemble de leurs opérations d'externalisation et leur permettre de mieux se concentrer sur leurs enjeux et cœur de métier.

En 2020, INTELCIA poursuit son expansion internationale avec une implantation en Amérique dans les caraïbes et de nouvelles destinations en Europe. Avec plus de 26 580 collaborateurs, le groupe est présent aujourd'hui sur 56 sites à travers 17 pays dont 1 site client en France et 11 sites clients au Portugal et ambitionne d'atteindre 30 000 collaborateurs vers la fin de 2021 avec une présence plus importante en Europe et de nouvelles implantations aux Etats-Unis.



Figure 1.1: Présence d'Intelcia à l'internationale

1.3 Services d'Intelcia

Les services proposés par Intelcia sont vastes et variés, permettant à ses clients de se concentrer sur leur cœur de métier tout en bénéficiant d'une assistance efficace et personnalisée. Les pôles de solutions offrent une gamme complète de services qui couvrent différents domaines, allant de la gestion de la relation client à la gestion des processus métiers, en passant par la comptabilité, le recouvrement, les ressources humaines, la conformité et l'IT.

Grâce à une approche innovante et centrée sur le client, Intelcia offre des solutions sur mesure pour chaque entreprise, basées sur les dernières technologies et les meilleures pratiques de l'industrie. Les équipes d'experts d'Intelcia travaillent en étroite collaboration avec les clients pour comprendre leurs besoins spécifiques et leur offrir des solutions adaptées à leurs besoins, visant à améliorer l'efficacité, la qualité et l'expérience client. Les solutions d'Intelcia sont conçues pour aider les clients à atteindre leurs objectifs tout en optimisant leurs coûts et en améliorant leur compétitivité sur le marché.



Figure 1.2: Solution offertes par Intelcia

1.4 Le pôle IT Solutions

Intelcia It Solutions est une filiale d'Intelcia, spécialisée dans les services de technologies de l'information. L'entreprise offre une gamme complète de services IT pour aider ses clients à atteindre leurs objectifs commerciaux en utilisant les dernières technologies et les meilleures pratiques de l'industrie.

Les services proposés par Intelcia It Solutions comprennent la consultation, l'intégration, le développement d'applications, l'exploitation d'infrastructures, la régie et bien plus encore. Les experts d'Intelcia It Solutions travaillent en étroite collaboration avec les clients pour comprendre leurs besoins spécifiques et leur offrir des solutions personnalisées pour améliorer leur efficacité et leur productivité.

Grâce à son expertise en matière de technologie et à sa solide expérience dans l'externalisation de services, Intelcia It Solutions offre des solutions efficaces et rentables.

pour les entreprises de toutes tailles. Que ce soit pour la mise en œuvre de nouveaux systèmes, la gestion de projets informatiques complexes ou la maintenance de l'infrastructure existante, Intelcia It Solutions est un partenaire de confiance pour aider les clients à atteindre leurs objectifs commerciaux.

1.4.1 Services d'Intelcia IT Solutions

Intelcia IT Solutions est le pôle d'expertise IT du groupe Intelcia, qui offre une gamme complète de services pour aider les entreprises à améliorer leur efficacité et leur productivité grâce aux technologies de l'information. Les services offerts par Intelcia IT Solutions sont regroupés en 4 piliers principaux :

1. **Services managés** : Ce pilier de service fournit des solutions pour la gestion et la maintenance des infrastructures informatiques des entreprises. Cela inclut des services de surveillance, de maintenance préventive, de gestion des incidents, de sécurité et de sauvegarde pour assurer une disponibilité et une performance optimales des systèmes informatiques.

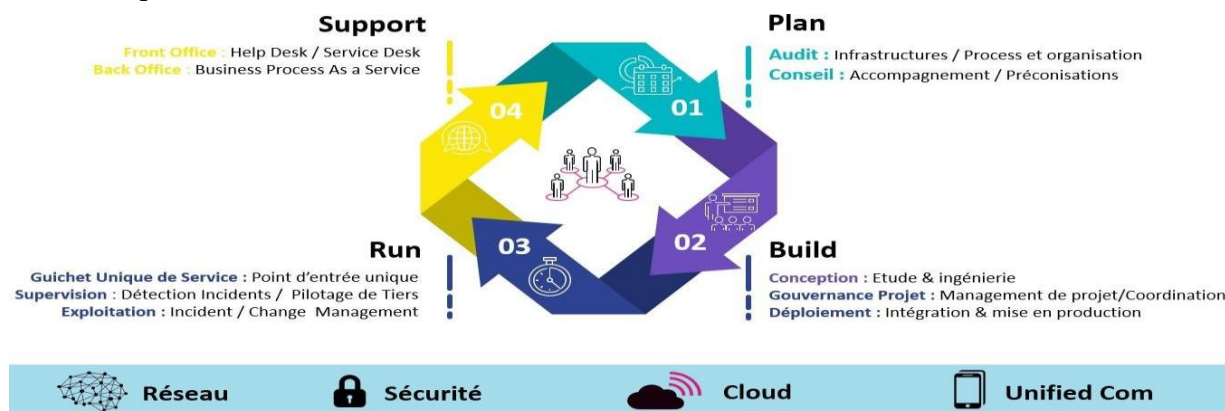


Figure 1.3: Service d'infogérance

2. **Solutions applicatives** : Ce pilier de service offre des solutions pour la conception, le développement et la mise en œuvre d'applications sur mesure pour les entreprises. Les experts d'Intelcia IT Solutions travaillent en étroite collaboration avec les clients pour comprendre leurs besoins spécifiques et concevoir des applications personnalisées pour répondre à ces besoins, tout en offrant un support continu.



Figure 1.4: Service des solutions applicatives

- 3. Business Intelligence :** Ce pilier de service propose des solutions pour la collecte, l'analyse et la présentation de données commerciales clés, afin d'aider les entreprises à prendre des décisions éclairées et à améliorer leur efficacité. Les experts d'Intelcia IT Solutions utilisent des technologies avancées d'analyse de données pour identifier les tendances et les opportunités de croissance, et pour aider les entreprises à prendre des décisions basées sur des données.

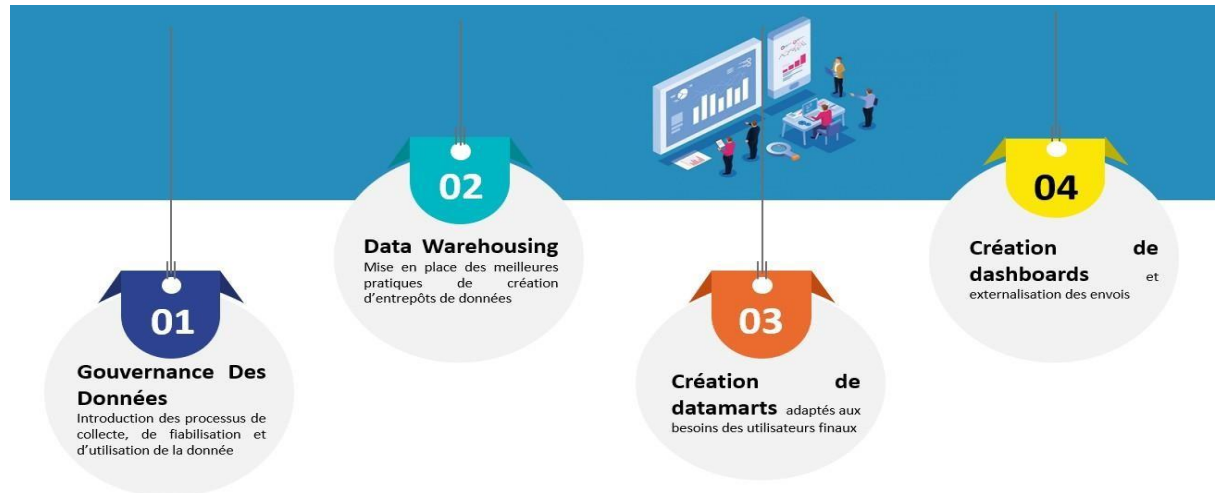


Figure 1.5: Services de Business Intelligence

- 4. Consulting :** Ce pilier de service offre des services de consultation pour aider les entreprises à élaborer une stratégie IT efficace pour leur entreprise, en évaluant leurs besoins et en proposant des solutions adaptées à leurs besoins spécifiques. Les experts d'Intelcia IT Solutions travaillent en étroite collaboration avec les clients pour comprendre leurs défis commerciaux et identifier les opportunités de croissance, tout en proposant des solutions rentables et efficaces pour y répondre.

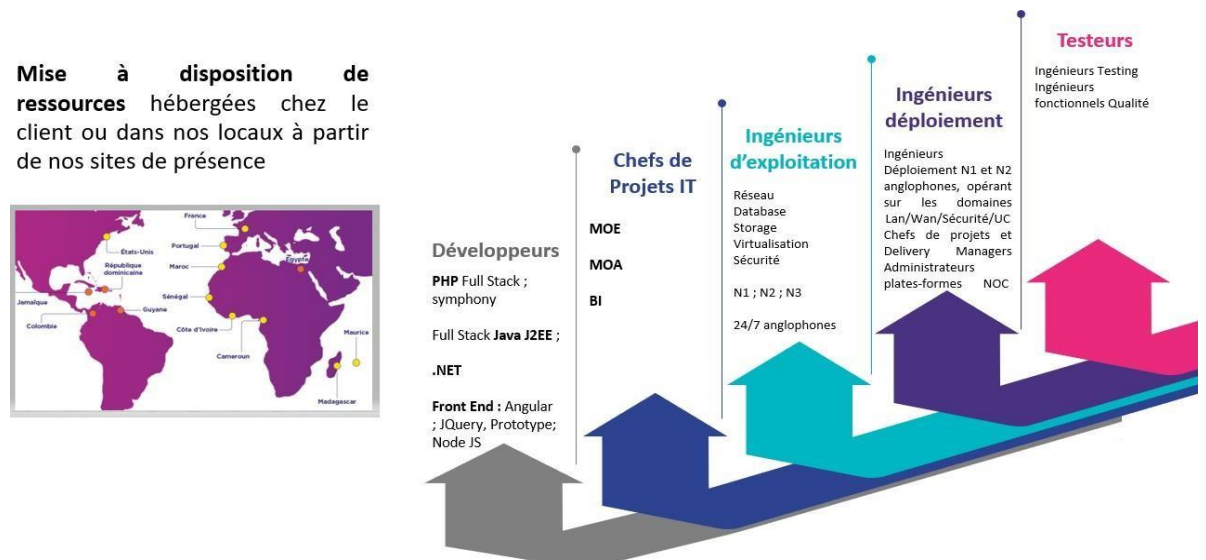


Figure 1.6: Service de Consulting

1.4.2 Clients Intelcia IT Solutions

Intelcia IT Solutions est un partenaire technologique de confiance pour une grande variété de clients, allant des petites et moyennes entreprises aux grandes entreprises. Les clients de l'entreprise proviennent de divers secteurs, tels que la finance, la santé, les télécommunications et la vente au détail. Intelcia IT Solutions comprend les besoins uniques de chaque client et fournit des solutions sur mesure pour répondre à leurs exigences spécifiques. Les clients de l'entreprise comptent sur son expertise dans des domaines tels que le cloud computing, l'analyse de données, la cybersécurité et le développement de logiciels pour améliorer leur efficacité opérationnelle, améliorer l'expérience client et stimuler la croissance. En mettant l'accent sur la construction de partenariats à long terme, Intelcia IT Solutions s'engage à offrir à ses clients un service exceptionnel et à fournir une valeur qui dépasse leurs attentes.



Figure 1.7: Clients d'Intelcia It Solutions

1.4.3 Structure organisationnelle

Le pôle IT Solutions est composé de plusieurs direction représentant chacune des business unit. La figure ci-dessous présente la structure organisationnelle du pôle.

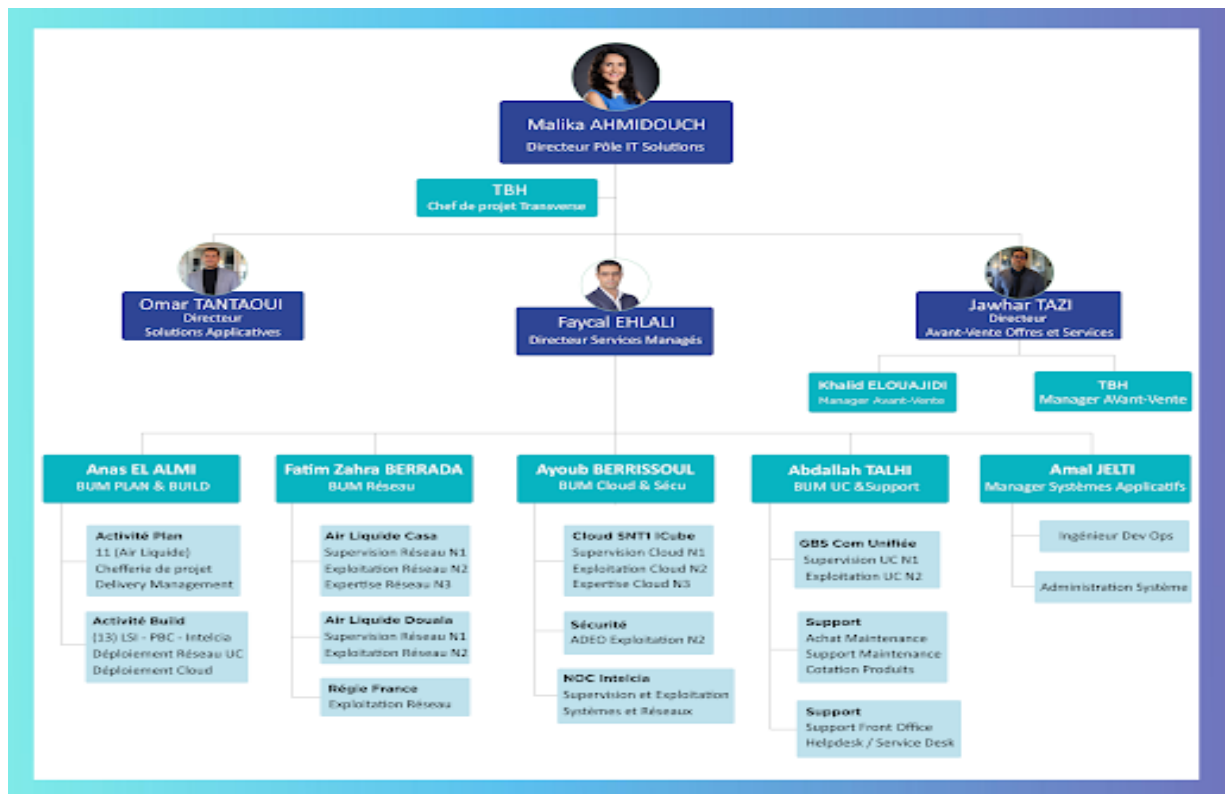


Figure 1.8: Structure organisationnelle

1.5 Présentation du projet

Dans un monde de plus en plus compétitif, les entreprises cherchent à offrir un service de qualité à leurs clients. Cela comprend notamment la gestion efficace et rapide des remboursements d'avoir. Cependant, de nombreuses entreprises rencontrent des difficultés avec des processus peu optimisés et manquent de visibilité sur les demandes de remboursement.

Afin de remédier à ces problèmes, il est essentiel pour les entreprises d'adopter des solutions informatiques permettant de gérer de manière automatisée et centralisée les demandes de remboursement. En optimisant ces processus, les entreprises peuvent réduire les retards, les erreurs et les coûts associés à la gestion manuelle des remboursements.

Dans le cadre de ce projet, l'objectif est de développer une application qui permettra à l'entreprise SFR de gérer de manière plus efficace et rentable les demandes de remboursement d'avoir. Cette application offrira une interface conviviale pour l'enregistrement, le traitement et le suivi des demandes, fournissant ainsi une meilleure visibilité sur le processus de remboursement.

1.5.1 La Présentation de SFR

SFR (Société française du radiotéléphone) est une entreprise de télécommunications majeure en France, offrant une large gamme de services de téléphonie mobile, d'accès à Internet et de télévision. Forte de nombreuses années d'expérience et d'expertise dans le secteur des communications, SFR s'est engagée à offrir des solutions innovantes et de qualité à ses clients.

Avec des millions d'abonnés à travers le pays, SFR se positionne comme un acteur clé dans le domaine de la connectivité et des services numériques. L'entreprise s'efforce constamment d'améliorer l'expérience client en proposant des offres adaptées, des technologies de pointe et un service clientèle de premier ordre.

La gestion des demandes de remboursement d'avoir est un aspect important de l'activité de SFR, nécessitant une approche efficace et bien structurée. Afin de répondre aux attentes élevées de ses clients et d'optimiser ses processus internes, SFR a décidé de développer une application dédiée à la gestion des remboursements d'avoir. Cette initiative vise à simplifier et à automatiser les flux de travail liés aux demandes de remboursement, offrant ainsi une meilleure expérience tant pour les clients que pour l'entreprise elle-même.

1.5.2 Objectif du projet

L'objectif de ce projet est de développer une application informatique efficace pour la gestion des remboursements des avoirs. Cette application permettra de faciliter le suivi et le traitement des demandes de remboursement, ainsi que la gestion des informations relatives aux clients. L'objectif est également de mettre en place une solution moderne et efficiente pour remplacer le système actuel, qui n'est pas suffisamment performant et ne permet pas une traçabilité adéquate des demandes de remboursement. Cette application devra donc répondre aux besoins des différents acteurs impliqués dans le processus de remboursement, tant au niveau du point de vente que de la direction. L'objectif final est d'améliorer la satisfaction des clients en offrant un service plus rapide et plus fiable, ainsi que de permettre aux équipes de gérer plus efficacement les demandes de remboursement et d'optimiser leur temps et leurs ressources.

1.6 Description des services externes en interaction avec le projet

1.6.1 Service d'authentification

Ce projet utilise un service d'authentification externe intégré à l'Intranet de SFR. Ce service gère de manière centralisée l'authentification des utilisateurs en utilisant un nom d'utilisateur et un mot de passe. Il prend également en charge la gestion des rôles et des droits, ce qui rend le système plus dynamique. Chaque utilisateur dispose d'autorisations spécifiques qui lui sont attribuées en fonction de son rôle. Dans notre projet, nous avons identifié trois acteurs principaux : l'agent de point de vente, le responsable de point de vente

et l'agent comptable, chacun ayant des autorisations spécifiques pour accéder aux fonctionnalités qui leur sont associées.

Le processus d'authentification suit une séquence d'étapes bien définies. Lorsqu'un utilisateur souhaite se connecter, notre application utilise le service d'authentification externe pour vérifier son identité. Une fois que l'identité de l'utilisateur est validée avec succès, notre application récupère les informations relatives à cet utilisateur, telles que son nom, son rôle et ses autorisations. Ensuite, un jeton d'accès sécurisé est généré pour cet utilisateur. Ce jeton d'accès sera utilisé dans notre application pour authentifier toutes les requêtes ultérieures de l'utilisateur, garantissant ainsi un accès sécurisé et contrôlé aux différentes fonctionnalités correspondantes.

1.6.2 Service d'avoir

Le service d'avoir externe joue un rôle essentiel lors de la création d'une demande de remboursement dans notre application. En effet, ce service est intégré à notre système pour gérer les avoirs associés à chaque demande de remboursement. Lorsque l'agent de point de vente crée une demande de remboursement, notre application établit une liaison avec l'identifiant de l'avoir correspondant. Ainsi, toutes les informations pertinentes liées à cet avoir, telles que les détails du client, sont automatiquement récupérées à partir du service d'avoir externe.

De plus, ce service d'avoir externe offre une fonctionnalité supplémentaire importante. Dans certaines situations spécifiques, il peut être nécessaire de griller l'avoir afin qu'il ne puisse plus être utilisé ultérieurement. Cette fonctionnalité est utilisée lorsque on aura une demande d'accord validée. En grillant l'avoir, nous nous assurons qu'il ne sera plus utilisé, évitant ainsi toute confusion ou utilisation incorrecte de cet avoir.

Ainsi, grâce à l'intégration de ce service d'avoir externe dans notre processus de création de demande de remboursement, nous pouvons garantir la cohérence et l'exactitude des informations du client liées à l'avoir, tout en offrant une fonctionnalité de sécurité permettant de griller l'avoir en cas de besoin.

1.7 Conclusion

Ce chapitre a fait l'objet d'une présentation général du contexte du projet en élaborant les différentes fonctionnalités que cette application doit englober et en soulignant également le cadre des outils principaux. Dans le chapitre suivant, une étude préalable du projet sera présentée.

Chapitre 2: Etude préalable

2.1 Introduction

Comme dans tout projet, avant de commencer la réalisation et avant même la conception, on se doit d'analyser l'existant pour souligner les fonctionnalités à améliorer, ainsi pour critiquer celles manquantes en dégagant les limites et les inconvénients du processus actuel dans le but d'aboutir à une meilleure réalisation. Ainsi, nous devons identifier les acteurs et les besoins fonctionnels et non fonctionnels du système. Ceci fera l'objectif de cette partie, et comme deuxième étape nous présenterons une spécification des besoins pour s'adapter aux attentes des utilisateurs de l'application afin de réussir la conception détaillée de l'application.

2.2 Etude préalable

L'étude de l'existant constitue le cœur de la phase d'analyse d'un projet. Cette étape est primordiale pour la mise en route de tout projet informatique ou autre, et qui permet de définir le contexte de fonctionnement, ou bien le processus métier, et de dégager les différentes imperfections dans le système actuel afin de les corriger.

En effet, Le processus de remboursement des avoirs est un élément clé dans la gestion des transactions commerciales. Les entreprises doivent s'assurer que leurs clients sont satisfaits de leurs produits ou services, et le remboursement des avoirs peut être une solution pour résoudre les problèmes de satisfaction. Cependant, le système actuel de remboursement peut présenter des défis en matière de communication, de traçabilité et de sécurité des données. Dans ce contexte, il est important d'examiner de près le fonctionnement actuel du système de remboursement des avoirs pour identifier les opportunités d'amélioration.

Le processus de remboursement des avoirs implique plusieurs étapes différentes qui sont toutes gérées par des services distincts. Tout d'abord, les agents des points de vente soumettent les demandes de remboursement qui sont ensuite examinées par le service impliqué. Les agents chargés de traiter les demandes de remboursement vérifient l'admissibilité des demandes et, si nécessaire, contactent les agents de points de vente pour obtenir des informations supplémentaires. Les pièces jointes, telles que le CIN et le RIB, sont également envoyées par e-mail pour être incluses dans le dossier de remboursement. Donc, le processus de remboursement actuel est fortement tributaire de l'utilisation des e-mails comme principal moyen de communication entre les différentes parties prenantes.

2.3 Critique de l'existant

Le système actuel de remboursement des avoirs comporte plusieurs lacunes qui peuvent entraver le processus de manière significative. Tout d'abord, l'utilisation des e-mails pour toute forme de communication entre les parties prenantes peut entraîner des retards et une confusion quant à la personne ou au service responsable du suivi de l'état de la demande de remboursement. De plus, il manque de traçabilité et de suivi clair pour les différentes étapes

du processus de remboursement, ce qui peut rendre difficile la résolution des problèmes et le suivi des retards dans le processus de remboursement. Enfin, la transmission de pièces jointes par e-mail peut poser des problèmes de confidentialité et de sécurité des données, surtout si les pièces jointes contiennent des informations sensibles.

En somme, bien que le système de remboursement des avoirs en place puisse fonctionner, il présente plusieurs limites qui peuvent être améliorées par une optimisation du processus et de la communication, ainsi que la sécurité.

2.4 But de la solution proposée

Le but de la solution proposée est de mettre en place un système de remboursement des avoirs plus efficace, rapide et sécurisé. Cela implique la création d'un service centralisé qui gère l'ensemble du processus de remboursement, depuis la soumission de la demande jusqu'à son traitement final. Les demandes de remboursement seront traitées de manière électronique, avec une traçabilité claire et un suivi précis de l'état de la demande. En outre, la solution proposée inclura des mesures de sécurité supplémentaires pour protéger les données des clients, telles que l'utilisation de protocoles de cryptage et de stockage sécurisé des informations personnelles. L'objectif ultime de cette solution est d'améliorer l'expérience client en fournissant un processus de remboursement plus efficace et plus sécurisé, tout en réduisant les coûts opérationnels pour l'entreprise.

2.4.1 Apport sur le plan technique

Sur le plan technique, la solution propose une approche moderne pour la gestion du processus de remboursement des avoirs. Tout d'abord, on utilise Amazon S3 pour stocker et gérer les fichiers relatifs au processus de remboursement. Cette solution permet une gestion efficace des fichiers tout en garantissant leur sécurité grâce au cryptage et au décryptage des données. Ensuite, on utilise Angular comme Framework pour le développement de l'interface utilisateur, qui offre une expérience utilisateur intuitive et conviviale. Côté backend, on a choisi Spring Boot, un Framework Java de renommée mondiale pour le développement rapide et efficace d'applications web. Cette combinaison de technologies permet de créer une application web robuste et sécurisée pour gérer le processus de remboursement des avoirs de manière efficace et transparente.

2.4.2 Apport sur le plan fonctionnel

Sur le plan fonctionnel, la solution proposée permet d'améliorer considérablement le processus de remboursement des avoirs en simplifiant les tâches des différents acteurs impliqués. Grâce à l'interface utilisateur intuitive et conviviale de l'application, les agents des points de vente pourront facilement soumettre les demandes de remboursement sans risque d'erreurs ou de retard dans le traitement. Les responsables chargés de la vérification des demandes de remboursement pourront accéder aux dossiers des clients en un seul clic et traiter les demandes rapidement. En outre, l'utilisation de la technologie Amazon S3 pour le stockage et la gestion des fichiers garantit la sécurité et la confidentialité des données.

Tout cela contribue à améliorer l'efficacité globale du processus de remboursement des avoirs, ce qui se traduit par un gain de temps et une meilleure satisfaction pour les clients.

2.5 L'identification des acteurs

Le rôle d'un acteur représente une entité externe qui interagit directement avec le système étudié, en échangeant de l'information (en entrée et en sortie). On trouve les acteurs en observant les utilisateurs directs du système, les responsables de la maintenance, ainsi que les autres systèmes avec lesquels il interagisse.

Dans ce projet, on peut distinguer les acteurs principaux : les agents point de ventes, les responsables de point de ventes, les agents commerciaux :

- **Activités d'agent point de vente :** Les agents de point de vente jouent un rôle clé dans le processus de remboursement des avoirs. Grâce à cette application, ils peuvent consulter les demandes de remboursement qui doivent être traitées dans leur point de vente, ainsi que créer des demandes de remboursement dans le cas de la transformation d'un avoir en remboursement. De plus, ils peuvent ajouter les fichiers nécessaires au dossier de remboursement, tels que le CIN ou le RIB, et si le dossier est incomplet, ils ont la possibilité d'ajouter les fichiers manquants.
- **Activités du responsable de point de vente :** Le responsable de point de vente est en charge de la gestion de plusieurs points de vente à travers l'application. Il a les mêmes fonctionnalités qu'un agent de point de vente, à savoir la consultation et la création de demandes de remboursement pour son point de vente. De plus, il peut ajouter des fichiers nécessaires au dossier de remboursement, tels que le CIN ou le RIB, et les intégrer dans la demande de remboursement. En outre, le responsable de point de vente peut également visualiser les demandes de remboursement de tous les points de vente qu'il gère et suivre l'état d'avancement de chaque demande.
- **Activités d'agent comptable :** Les agents comptables ont un rôle important dans le processus de remboursement des avoirs. Ils sont responsables de la vérification des documents inclus dans les dossiers de remboursements pour s'assurer qu'ils sont complets et conformes aux exigences légales. Ils peuvent également changer le statut du dossier à "remboursé" lorsqu'ils ont validé tous les documents et que le remboursement peut être effectué. De plus, les agents comptables ont la possibilité de consulter et de lister la totalité des dossiers de remboursements pour tous les points de vente. Ils peuvent ainsi facilement identifier les dossiers qui nécessitent leur attention.

2.6 Besoins fonctionnels

La gestion des remboursements des avoirs est cruciale, et elle nécessite la mise en place de plusieurs fonctionnalités clés pour garantir un traitement efficace des dossiers. Deux cas de remboursement comptable ont été identifiés, qui nécessitent un dossier de suivi pour chaque demande de remboursement.

Le premier cas concerne le remboursement sur facture. Lorsqu'une demande d'accord a été validée pour un remboursement, un dossier de remboursement doit être créé automatiquement en exposant notre service web de création de dossier. Il est important de noter que la création d'un dossier de remboursement est obligatoire et qu'une demande de remboursement ne sera jamais refusée ou annulée à ce stade. De plus, l'avoir doit être grillé dès la création du dossier de remboursement pour éviter toute confusion.

Le deuxième cas concerne la transformation de l'avoir en remboursement. Dans ce cas, la création du dossier de demande de remboursement doit être effectuée manuellement par les agents de point de vente. Il est important de noter que l'avoir doit pouvoir être utilisé, mais seulement après la validation de la demande de remboursement. Une fois la demande de remboursement validée, l'avoir doit être grillé. Il est également essentiel d'avoir un statut sur le dossier de remboursement qui déclenchera un remboursement comptable.

Dans le cadre de ce système, tous les acteurs ont des privilèges spéciaux selon leur rôle. Alors, dans cette partie on va éclaircir les tâches accordées à chaque utilisateur en détails afin de bien construire une claire image sur le fonctionnement du système.

- **Généralités :**

Chaque utilisateur qui souhaite accéder au système doit passer par les étapes d'authentification pour assurer la sécurité et la confidentialité des données. Le processus d'authentification vérifie la validité et l'exactitude de l'identifiant fourni par l'utilisateur.

Lorsqu'un utilisateur tente d'accéder à une fonctionnalité de l'application, le système récupère le jeton JWT et extrait l'identifiant d'utilisateur. Ensuite, le système fait appel au service d'authentification externe pour vérifier l'identifiant de l'utilisateur. Si l'identifiant est valide, le service d'authentification renvoie les informations pertinentes sur l'utilisateur, ainsi que le(s) point(s) de vente associé(s).

Ce service d'authentification externe authentifie l'utilisateur en utilisant les mécanismes appropriés, tels que la comparaison des informations d'identification avec la base de données. Si les informations sont valides, l'utilisateur est autorisé à accéder à la fonctionnalité demandée.

- **Agent de point de vente :**

Un agent de point de vente est en mesure de consulter les demandes de remboursement qui doivent être traitées dans son point de vente, de créer des demandes de remboursement dans le cas de la transformation d'un avoir en remboursement, d'ajouter les fichiers nécessaires au dossier de remboursement (comme le CIN, le RIB, etc.) et de pouvoir invalider ou rétablir ces fichiers selon les besoins. L'agent de point de vente doit également être en mesure de changer le statut de la demande de remboursement en fonction de l'avancement de son traitement et d'ajouter un commentaire significatif, si nécessaire, pour expliquer la raison de ce changement. De plus, il doit avoir la possibilité de consulter la liste des actions à effectuer sur le dossier de remboursement, afin de savoir quelles tâches doivent encore être réalisées, ainsi que d'accéder à l'historique des actions déjà effectuées, pour obtenir une vue complète des actions prises sur le dossier de remboursement.

- **Responsable de point de vente :**

Un responsable de point de vente partage les mêmes responsabilités que les agents de point de vente en ce qui concerne la gestion des demandes de remboursement. En plus des tâches mentionnées précédemment, le responsable de point de vente a la capacité supplémentaire de naviguer entre les différents points de vente dont il est responsable. Cela lui permet de consulter les demandes de remboursement de chaque point de vente spécifique et d'effectuer les actions nécessaires en conséquence.

Le responsable de point de vente peut donc accéder aux demandes de remboursement spécifiques à chaque point de vente sous sa responsabilité, les examiner, apporter des modifications si nécessaires, ajouter des fichiers pertinents au dossier de remboursement et gérer le statut des demandes en fonction de leur traitement. De plus, il peut ajouter des commentaires significatifs pour expliquer les changements de statut et suivre les actions à effectuer sur chaque dossier de remboursement.

- **Agent comptable :**

Pour un agent comptable, il dispose l'accès aux dossiers de remboursement pour effectuer les remboursements comptables nécessaires. Il doit être en mesure de valider les remboursements approuvés par les agents de point de vente, ainsi que de consulter les dossiers pour vérifier la conformité des documents. Le comptable doit également être en mesure de modifier les statuts des dossiers en fonction de l'avancement du traitement.

2.7 Spécification non fonctionnelle

Toute solution informatique doit en plus de répondre aux exigences fonctionnelles respecter un ensemble de critères dits non fonctionnels pour garantir la qualité logicielle. Nous détaillons dans cette partie les aspects transversaux de ce projet. D'où, à part les besoins fondamentaux, notre futur système doit répondre aux critères suivants :

- **La performance :**

Le système doit être avant tout performant c'est-à-dire à travers ses fonctionnalités, il doit répondre à toutes les exigences des usagers d'une manière optimale.

- **Rapidité du traitement :**

En effet, vu le nombre important des transactions quotidiennes, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.

- **La convivialité :**

Le futur système doit être facile à utiliser. En effet, les interfaces utilisateurs doivent être conviviales c'est-à-dire simples, ergonomiques et adaptées à l'utilisateur.

- **La Sécurité :**

Vu que les données manipulées par notre application sont critiques, il est impératif que le système soit hautement sécurisé. Des mesures de sécurité robustes doivent être mises en place pour protéger les données sensibles des utilisateurs, tels que les informations de compte et les données financières.

- **La confidentialité :**

Elle est assurée par un système d'accès qui gère les autorisations des utilisateurs de l'application. Ce système permet de contrôler l'accès des utilisateurs aux différentes fonctionnalités de l'application en fonction de leurs rôles et responsabilités. Cela garantit que les informations confidentielles sont accessibles uniquement aux personnes autorisées.

- **La réutilisabilité :**

Nous avons veillé à ce que la granularité des composants de ce projet soit la plus fine possible pour pouvoir les réutiliser.

- **Extensibilité :**

Les composants sont conçus de façon qu'ils peuvent être extensibles sans pour autant être modifiés.

Conclusion

Ce chapitre a été consacré à l'étude de l'existant et la spécification des besoins qui représentent la première phase du cycle de développement de tout projet informatique. Cette étude m'a permis de lister toutes les attentes de cette application web ainsi que les contraintes de l'exécution, ce qui nous entraînera à une nouvelle phase, celle de la méthodologie de travail qui sera l'objet du prochain chapitre.

Chapitre 3: Méthodologie

3.1 Introduction

Dans le cadre de la méthodologie de travail suivie pour la réalisation de ce dernier, on a opté pour la méthode SCRUM étant une méthode agile de gestion de projets informatiques privilégiant la communication, et facilitant les réorientations opportunes. Dans ce chapitre nous définirons les sprints, la conduite et le backlog du projet à réaliser.

3.2 Choix de la méthode

Le choix entre une méthode et une autre, dépend de la nature du projet et de sa taille. Pour des projets de petite taille et dont le domaine est maîtrisé, par exemple, un cycle de vie en cascade s'avère largement suffisant. Lorsqu'il s'agit d'un projet où les données ne sont pas réunies dès le départ, où les besoins sont incomplets voire flous, il faut s'orienter vers une méthode itérative ou orientées prototypes.

Parmi les méthodes itératives, nous pouvons distinguer les méthodes AGILE largement utilisées de nos jours à travers le monde. Une méthode AGILE est menée dans un esprit collaboratif et s'adapte aux approches incrémentales. Elle engendre des produits de haute qualité tout en tenant compte de l'évolution des besoins du client. Une méthode AGILE assure une meilleure communication avec le client et une meilleure visibilité du produit livrable. Elle permet aussi de gérer la qualité en continu et de détecter des problèmes le plus tôt au fur et à mesure, permettant ainsi d'entreprendre des actions correctrices sans trop de pénalités dans les coûts et les délais. Il y a moult méthodes AGILE et il ne s'agit pas de choisir la meilleure méthode parmi celles existantes. Il s'agit plutôt de sélectionner la méthode la plus adaptée à notre projet.

La nature de projet qui doit être évolutif et dont tous les besoins n'ont pas encore été totalement identifiés, nous a orientées vers une méthode de type AGILE et plus particulièrement SCRUM.

3.3 La méthode SCRUM

La méthode SCRUM est un Framework ou cadre de travail de gestion de projets informatiques privilégiant la communication, et facilitant les réorientations opportunes. Ce cadre définit trois rôles principaux :

- **Le Product Owner** qui porte la vision du produit à réaliser (représentant généralement le client) ;
- **Le Scrum Master** garant de l'application de la méthodologie Scrum ;
- **L'équipe de développement** qui réalise le produit.

3.4 Le processus SCRUM

La vie d'un projet SCRUM est rythmée par un ensemble de réunions clairement définies et strictement limitées dans le temps (time boxing) :

- **Planification du Sprint (Sprint = itération) :** au cours de cette réunion, l'équipe de développement sélectionne les éléments prioritaires du « Product Backlog » (liste ordonnée des exigences fonctionnelles et non fonctionnelles du projet) qu'elle pense pouvoir réaliser au cours du sprint (en accord avec le « Product Owner »)
- **Revue de Sprint :** au cours de cette réunion qui a lieu à la fin du sprint, l'équipe de développement présente les fonctionnalités terminées au cours du sprint et recueille les feedbacks du Product Owner et des utilisateurs finaux. C'est également le moment d'anticiper le périmètre des prochains sprints et d'ajuster au besoin la planification de release (nombre de sprints restants) ;
- **Rétrospective de Sprint :** la rétrospective qui a généralement lieu après la revue de sprint est l'occasion d'améliorer (productivité, qualité, efficacité, conditions de travail, etc.) à la lueur du "vécu" sur le sprint écoulé (principe d'amélioration continue)
- **Mêlée quotidienne :** il s'agit d'une réunion de synchronisation de l'équipe de développement qui se fait debout (elle est aussi appelée "stand up meeting") en 15 minutes maximum au cours de laquelle chacun répond principalement à 3 questions : « Qu'est-ce que j'ai terminé depuis la dernière mêlée ? Qu'est-ce que j'aurai terminé d'ici la prochaine mêlée ? Quels obstacles me retardent ? ».

3.5 Définition de période Sprint

Le sprint est une période d'un mois au maximum, au bout de laquelle l'équipe délivre un incrément du produit, potentiellement livrable. Une fois la durée choisie, elle reste constante pendant toute la durée du développement. Un nouveau Sprint démarre dès la fin du précédent.

Chaque Sprint possède un but et on lui associe une liste d'éléments du carnet du produit (fonctionnalité) à réaliser.

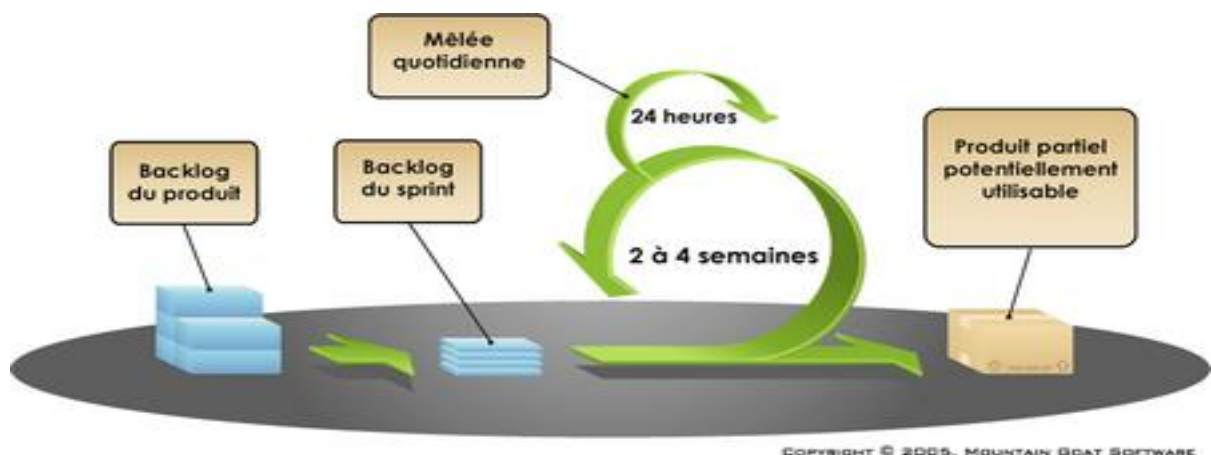


Figure 3.1: Principe de la méthode SCRUM

3.6 Identification des sprints

Après une réunion de planification du projet nous sommes arrivés à identifier ses sprints et à prévoir ce qui sera réalisé dans chacun d'eux. Les sprints identifiés sont les suivants :

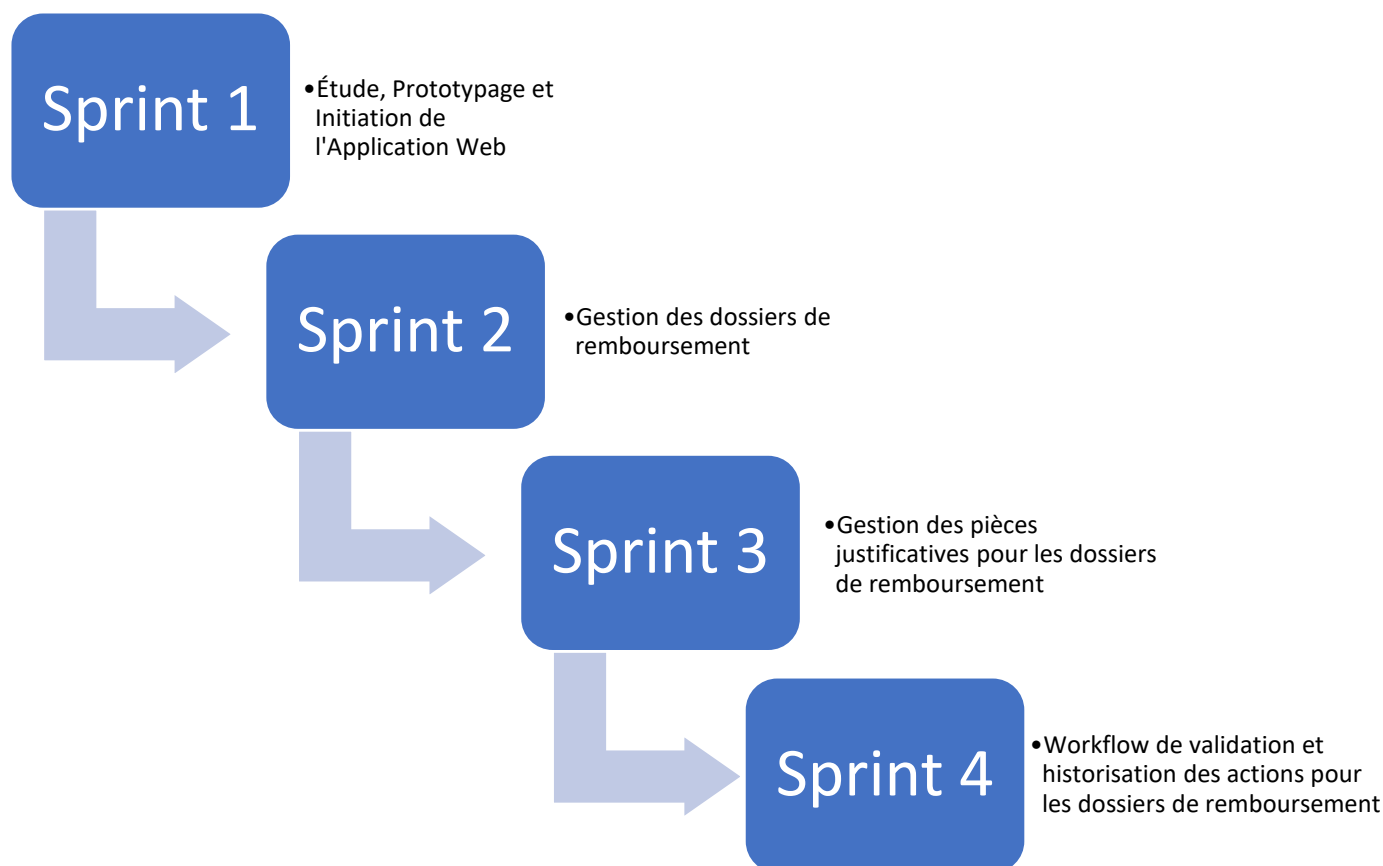


Figure 3.2 Identification des sprints

3.7 Le backlog du produit

Le backlog du produit est un élément essentiel de la méthodologie Scrum, qui représente l'ensemble des fonctionnalités techniques ou fonctionnelles nécessaires à la réalisation d'un produit. Ces fonctionnalités sont ensuite divisées en deux types : les histoires utilisateurs (user story) et les histoires techniques (technical story). Les histoires utilisateurs décrivent les besoins fonctionnels des utilisateurs finaux et les histoires techniques se concentrent sur les aspects techniques de l'application.

3.7.1 Histoire utilisateurs

Dans ce projet, on a identifié trois acteurs principaux : l'agent de point de vente, le responsable du point de vente et l'agent comptable, et on a créé des backlog produit pour chacun d'entre eux. Les tableaux de backlog produit (Tableau 3.1 pour l'agent et le responsable du point de vente, et

Tableau 3.2 pour l'agent comptable) résument les fonctionnalités et les besoins spécifiques de chaque acteur. Chaque histoire utilisateur est caractérisée par ses risques et sa priorité afin de faciliter la planification et la mise en œuvre de chaque fonctionnalité.

Il est important de noter que ces tableaux de backlog produit se concentrent uniquement sur les besoins fonctionnels des utilisateurs et excluent les histoires techniques telles que la préparation de la maquette graphique, les travaux de conception et les jeux de test. Cette approche nous permet de nous concentrer sur la planification et la mise en œuvre des fonctionnalités les plus importantes pour chaque acteur, en garantissant une utilisation optimale des ressources et une progression efficace du projet. On a également attribué des priorités et évalué les risques associés à chaque histoire utilisateur, ce qui facilite la prise de décisions éclairées lors de la planification et de la gestion des sprints de développement.

Nom	Description	Thème	Risque	Priorité
Création d'un dossier de remboursement	En tant qu'agent point de vente ou responsable de point de vente, je veux pouvoir créer un dossier de remboursement pour un client.	Gestion de dossier de remboursement	Faible	Elevé
Gestion des pièces jointes d'un dossier de remboursement	En tant qu'agent point de vente ou responsable de point de vente, je veux pouvoir gérer les pièces jointes liées à un dossier de remboursement afin de faciliter la soumission et le traitement du dossier.	Gestion de dossier de remboursement	Moyen	Elevé
Changer le statut d'un dossier de remboursement	En tant qu'agent point de vente ou responsable de point de vente, je veux pouvoir changer le statut d'un dossier de remboursement (par exemple de "Annulé" à "Complet") afin qu'il soit prêt à être traité par un agent comptable.	Gestion des dossiers de remboursement	Moyen	Elevé
Consultation des demandes de remboursement des points de vente sous responsabilité	En tant qu'agent de point de vente ou responsable de point de vente, je veux pouvoir consulter la liste des demandes de remboursement soumises par mon point de vente ou les agents de point de vente sous ma responsabilité, afin de suivre l'état d'avancement des demandes, de vérifier leur validité, leur pertinence et de fournir des informations aux clients en cas de besoin	Gestion des dossiers de remboursement	Faible	Elevé
Ajouter des commentaires pour une demande de remboursement	En tant qu'agent de point de vente ou responsable de point de vente, la tâche consiste à ajouter des commentaires liés à une demande de remboursement. Cette fonctionnalité permet de fournir des informations complémentaires, de signaler des problèmes ou de suivre l'état de la demande.	Gestion des dossiers de remboursement	Faible	Elevé

Tableau 3.1: Backlog du produit pour l'agent et le responsable de point de vente

Nom	Description	Thème	Risque	Priorité
-----	-------------	-------	--------	----------

Consultation des dossiers de remboursement	En tant qu'agent comptable, je veux pouvoir consulter un dossier de remboursement validé pour vérifier la conformité des documents et autoriser le traitement comptable du remboursement.	Gestion de dossier de remboursement	Faible	Elevé
Vérification des pièces jointes d'un dossier de remboursement	En tant qu'agent comptable, je souhaite pouvoir vérifier les pièces jointes d'un dossier de remboursement soumis par un agent de point de vente. L'objectif est de garantir la conformité des documents et leur validité. Les fonctionnalités requises pour cette tâche comprennent la possibilité de télécharger les pièces jointes, de les visualiser et de les valider ou invalider selon leur conformité.	Gestion de dossier de remboursement	Moyen	Elevé
Changement de statut d'un dossier de remboursement	En tant qu'agent comptable, je veux pouvoir changer le statut d'un dossier de remboursement, afin de refléter son avancement dans le processus de traitement. Les statuts possibles incluent "Incomplet", "Valide", "Remboursé", "Mauvaise adresse" Cette tâche est importante pour garantir que les demandes de remboursement sont traitées de manière efficace.	Gestion des dossiers de remboursement	Moyen	Elevé
Ajouter des commentaires pour une demande de remboursement	En tant qu'agent comptable, je veux pouvoir ajouter des commentaires à un dossier de remboursement lorsque je change son statut, afin d'expliquer les raisons du changement de statut et de fournir des informations supplémentaires à l'agent de point de vente et au client concerné.	Gestion des dossiers de remboursement	Faible	Elevé

Tableau 3.2: Backlog du produit pour l'agent comptable

3.7.2 Sprint Backlog

Le Sprint Backlog récapitule toutes les tâches qui ont été identifiées et planifiées pour le sprint en cours. Il permet à l'équipe de développement de visualiser clairement les activités à réaliser et de suivre leur progression tout au long du sprint. Chaque tâche est décrite de manière détaillée, comprenant une brève description, des critères d'acceptation, une estimation de la charge de travail et une assignation à un membre de l'équipe.

Ce tableau (Tableau 3.3) offre une vue d'ensemble des tâches à réaliser. Il permet également de suivre l'avancement des travaux, de détecter rapidement les éventuels retards ou obstacles, et d'ajuster les priorités si nécessaires.

Sprint	Tâches
Étude, Prototypage et Initiation de l'Application Web	Etude de l'existant
	Maquettes du frontend
	Backend CI/CD
	Frontend CI/CD
	Authentification
Gestion des dossiers de remboursement	Modèle de données
	Création d'un dossier de remboursement
	Récupération des dossiers de remboursement par critères
	Lister des dossiers de remboursement
	Liste des dossiers en attente d'action en fonction de rôle de l'utilisateur
	Adapter la liste des dossiers de remboursement en fonction du rôle
	Afficher le PDV de rattachement
	Afficher le détail d'un dossier de remboursement
	Modification de la demande de remboursement
	Création d'un commentaire
	Modification d'Informations client
Gestion des pièces justificatives pour les dossiers de remboursement	Paramétrage S3
	Sécurisation des fichiers sur le S3
	Création des fichiers
	Invalidation et rétablissement d'un fichier
	Liste des fichiers d'un dossier de remboursement
Workflow de validation et historisation des actions pour les dossiers de remboursement	Griller un avoir selon le type du dossier de remboursement (sur facture, transformation de l'avoir en remboursement)
	Conditionner le changement du statut
	Afficher Historique des actions du dossier de remboursement

Tableau 3.3 : Sprint backlog

3.8 Planification du projet

La planification du projet revêt une importance capitale pour assurer sa réussite et sa bonne exécution. Dans cette section, nous allons aborder les différentes étapes de la planification que nous avons suivies pour notre projet. Tout d'abord, nous avons analysé en détail les besoins et les objectifs du projet, en tenant compte des exigences des utilisateurs finaux. Cette analyse approfondie nous a permis de définir clairement les fonctionnalités et les livrables attendus.

Ensuite, nous avons utilisé la méthode de développement Agile pour organiser notre travail en sprints itératifs. Chaque sprint correspond à une période définie, généralement de deux à quatre semaines, pendant laquelle nous nous concentrons sur un ensemble spécifique de fonctionnalités à développer. Nous avons créé un backlog du produit qui répertorie toutes les histoires utilisateurs identifiées lors de l'analyse des besoins. Ces histoires ont été hiérarchisées en fonction de leur priorité et de leur complexité, ce qui nous a permis de les planifier et de les répartir sur les sprints.

Pendant le déroulement des sprints, nous avons régulièrement suivi l'avancement du projet et ajusté notre planification en fonction des obstacles rencontrés ou des changements de priorités. Cette flexibilité nous a permis de rester réactifs et d'adapter notre approche en fonction des besoins émergents.

En outre, nous avons également préparé un diagramme de Gantt détaillant la planification du projet.

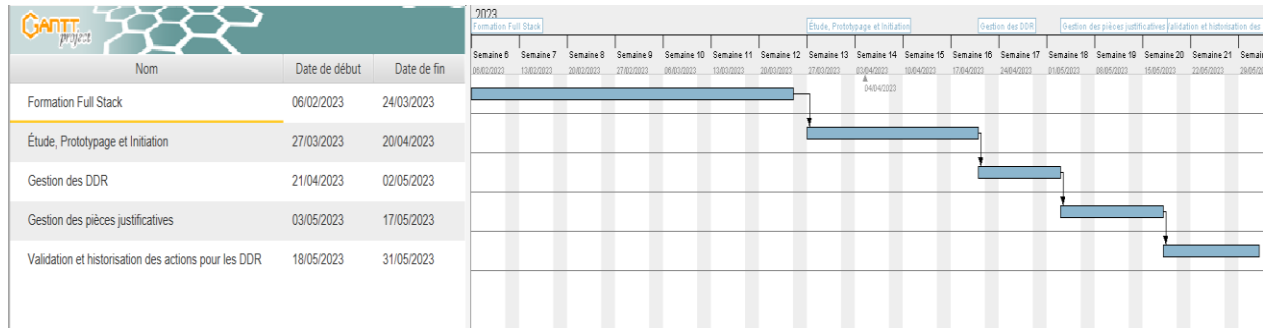


Figure 3.3: Diagramme de Gantt

3.9 Conclusion

Ce chapitre a fait l'objet de l'identification de la méthodologie de travail à suivre tout au long du projet, à savoir la méthode SCRUM, tout en élaborant les spécifications de chaque acteur dans le backlog du produit, ce qui nous mènera à réaliser la conception qui fera l'objet du chapitre suivant.

Chapitre 4: Conception

4.1 Introduction

La conception est une étape primordiale dans le développement de tout projet informatique. Il s'agit de concevoir les différents acteurs du système, leurs données, leurs activités ainsi que la relation entre eux, ceci se fait à travers des diagrammes qui représentent des modèles conceptuels de données. Dans ce chapitre, on présente les différents diagrammes qui modélisent le système.

4.2 Architecture applicative

Pour le développement de cette application, nous avons opté pour une architecture monolithique, avec des interactions avec deux services externes : le service d'authentification et le service de gestion des avoirs. Bien que notre architecture soit monolithique dans le contexte de ce projet, elle est considérée comme un service dédié au sein de l'écosystème de SFR.

Dans notre approche, nous avons utilisé les éléments suivants :

- **Service d'authentification externe** : Nous avons intégré un service d'authentification externe pour gérer les fonctionnalités de connexion et d'authentification des utilisateurs. Ce service nous permet de récupérer les informations d'identification des utilisateurs lorsqu'ils se connectent à notre application. En utilisant ces informations, nous validons l'identité de l'utilisateur et générons un JSON Web Token (JWT) pour sécuriser leur accès à l'application. L'intégration de ce service externe renforce la sécurité du système d'authentification et garantit une expérience utilisateur fluide et fiable.
- **Service de gestion des avoirs externe** : On a intégré un service externe de gestion des avoirs pour faciliter la récupération des informations liées aux avoirs financiers. Ce service nous permet d'obtenir les détails des avoirs ainsi que les informations des clients associés à chaque avoir. En utilisant ces informations, nous sommes en mesure d'afficher les avoirs correspondants dans notre application et de les traiter selon les besoins.
- **Base de données PostgreSQL** : Nous avons choisi PostgreSQL comme système de gestion de base de données pour stocker et gérer les données de notre application. PostgreSQL offre des fonctionnalités avancées, une stabilité et une robustesse reconnues, ce qui en fait un choix approprié pour notre projet.
- **Stockage des fichiers avec Amazon S3** : Pour gérer les fichiers associés à notre application (par exemple, les pièces jointes aux demandes de remboursement), nous avons utilisé Amazon S3. Cette solution de stockage cloud nous offre une capacité de stockage évolutive, une haute disponibilité et une sécurité des données avancée.

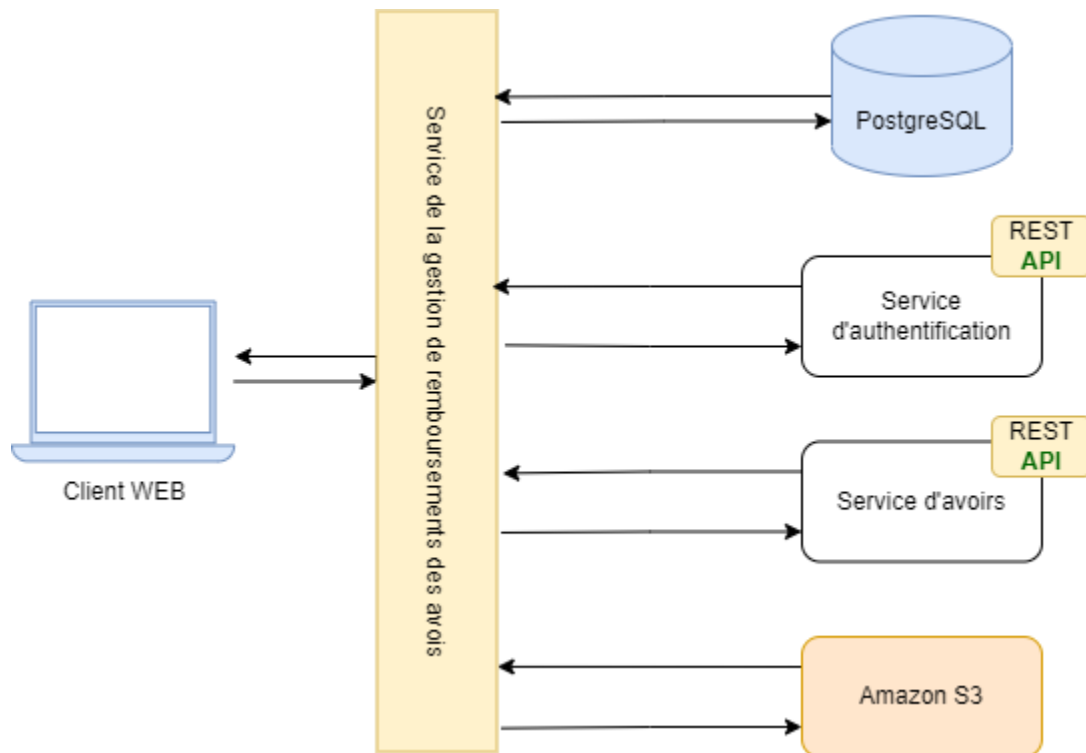


Figure 4.1: Architecture générale de l'application

4.3 Diagramme de classe

Le modèle conceptuel des données (MCD) présenté par la figure a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités. Il donne une vision globale de la base de données de l'application. Ce premier niveau de conception consiste en une phase d'analyse du problème réel.

Ainsi, le diagramme de classe montre la structure statique du modèle d'information, particulièrement les choses qui existent, leur structure interne, et leurs relations aux autres choses. Un diagramme de classe ne doit présenter aucune information de nature temporelle. Contenu : classes, sous-classes, attributs et valeurs, méthodes, liens (multiplicité, généralisation, composition), catégories et dépendance

Donc, un diagramme de classes décrit clairement la structure d'un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets, les figures ci-dessous représentent les diagrammes de classe utilisés dans le développement du projet selon chaque service.

4.3.1 Diagramme de classe de l'application

Le diagramme de classe de notre application représente les différentes entités clés impliquées dans la gestion des remboursements. Parmi ces entités, nous avons la classe

"Refund" qui incarne un dossier de remboursement spécifique. La classe "Event" est utilisée pour enregistrer les actions et les événements qui se produisent dans un dossier de remboursement, fournissant ainsi un historique des activités liées à ce dossier. La classe "Comment" permet aux utilisateurs d'ajouter des commentaires significatifs sur les remboursements, favorisant ainsi la communication et la collaboration. La classe "File" gère les différents fichiers associés aux remboursements, offrant un moyen de stocker et de référencer les pièces jointes pertinentes.

En plus de ces classes, nous avons également la classe "Customer" qui représente les clients utilisant notre application de gestion des remboursements. Cette classe contient les informations relatives aux clients, telles que leur nom, leur adresse, etc.

Une autre classe importante est "AuditingEntity" (ou les autres classes qui en héritent) qui est utilisée pour capturer des informations supplémentaires sur les entités, telles que la personne qui a créé ou modifié une entité donnée, ainsi que la date et l'heure de la modification ou de la création. Cette classe facilite la traçabilité et l'audit des actions effectuées sur les entités.

Le diagramme de classe fournit une vue claire des relations et des attributs de chaque classe, ce qui permet une meilleure compréhension de l'architecture globale de notre application de gestion des remboursements.

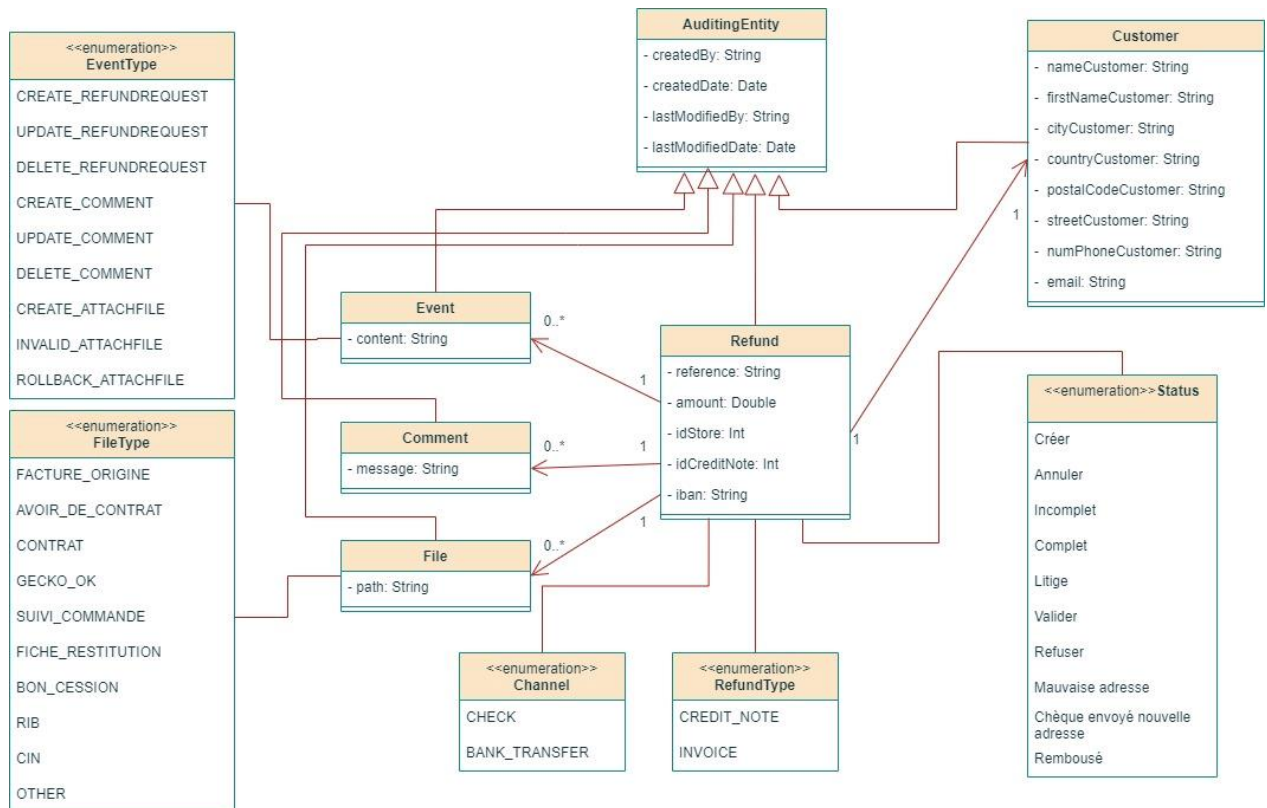


Figure 4.2 : Diagramme de classe de l'application

4.3.2 Dictionnaire de données

Le dictionnaire de données est un élément crucial de la conception d'une application. Il décrit l'ensemble des données qui seront stockées dans la base de données de l'application ainsi que les attributs qui leur sont associés. Dans cette section, nous présentons le dictionnaire de données de notre application de gestion des remboursements. Ce dictionnaire donne une vue détaillée des différentes entités de l'application et de leurs attributs, ce qui facilite la compréhension de l'architecture globale de l'application et de son fonctionnement.

- Classe « Refund »

Attribut	Désignation	Type
reference	La référence unique du dossier de remboursement.	Texte
amount	Le montant du remboursement.	Décimal
idStore	L'identifiant du point de vente associé au remboursement.	Entier
idCreditNote	L'identifiant de l'avoir associée au remboursement.	Entier
iban	Le numéro IBAN utilisé pour le remboursement.	Texte
status	Le statut du remboursement (créé, annulé, incomplet, etc.).	Texte
type	Le type de remboursement (facture, avoir).	Texte
channel	Le canal utilisé pour le remboursement (chèque, virement bancaire).	Texte

Tableau 4.1: Dictionnaire de la table Refund

- Classe « Event »

Attribut	Désignation	Type
content	Une colonne qui stocke l'ancienne version avant l'évènement sous forme de chaîne de caractères JSON.	Texte
status	Cet attribut représente le statut de l'événement. Il permet de catégoriser les différents types d'événements dans l'application.	Texte

Tableau 4.2: Dictionnaire de la table Event

- Classe « Comment »

Attribut	Désignation	Type
message	Cet attribut représente le contenu du commentaire laissé par l'utilisateur. Il s'agit d'une chaîne de caractères qui contient le message ou la description associée au commentaire.	Texte

Tableau 4.3: Dictionnaire de la table Comment

- Classe « File »

Attribut	Désignation	Type
path	Cet attribut représente le chemin du fichier dans le système de stockage Amazon S3.	Texte
type	Cet attribut indique le type de fichier associé. Ce champ permet de classer les fichiers selon leur nature ou leur utilisation.	Texte

Tableau 4.4 : Dictionnaire de la table File

- Classe « Customer »

Attribut	Désignation	Type
nameCustomer	Cet attribut représente le nom du client.	Texte
firstNameCustomer	Cet attribut représente le prénom du client.	Texte
cityCustomer	Cet attribut représente la ville où réside le client.	Texte
countryCustomer	Cet attribut représente le pays de résidence du client.	Texte
postalCodeCustomer	Cet attribut représente le code postal de la localité où réside le client.	Texte
streetCustomer	Cet attribut représente le nom de la rue ou l'adresse postale du client.	Texte
numPhoneCustomer	Cet attribut représente le numéro de téléphone du client.	Texte
email	Cet attribut représente l'adresse e-mail du client.	Texte

Tableau 4.5: Dictionnaire de la table Customer

- Classe « AuditingEntity »

Attribut	Désignation	Type
createdBy	Cet attribut représente l'identifiant de l'utilisateur qui a créé l'entité. Il stocke l'information sur la personne qui a effectué la création.	Texte
createdDate	Cet attribut représente la date et l'heure de création de l'entité. Il enregistre le moment précis où l'entité a été créée.	Date
lastModifiedBy	Cet attribut représente l'identifiant de l'utilisateur qui a apporté la dernière modification à l'entité. Il stocke l'information sur la personne qui a effectué la dernière modification.	Texte
lastModifiedDate	Cet attribut représente la date et l'heure de la dernière modification de l'entité. Il enregistre le moment précis où l'entité a été modifiée pour la dernière fois.	Date

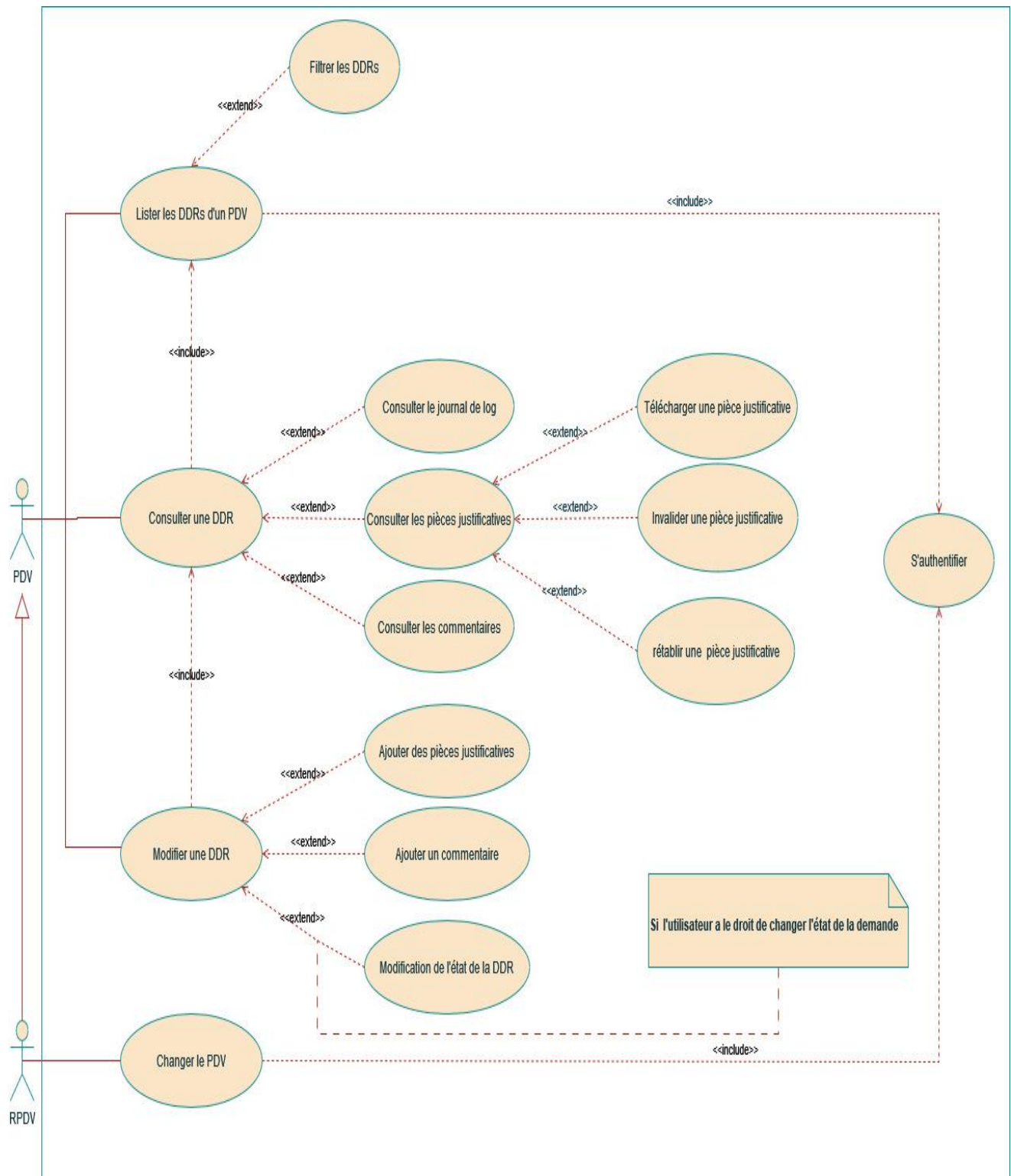
Tableau 4.6: Dictionnaire de la table AuditingEntity

4.4 Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation modélise le service qui sera rendu par le système. Ce type de diagramme décrit les besoins du client et représente les interactions des différents utilisateurs avec l'application.

- **Les acteurs** : Les acteurs sont les utilisateurs qui interagissent avec le système et ils sont décrit par leurs besoins vis-à-vis ce dernier. Dans ce projet, on différencie entre plusieurs acteurs à savoir : les étudiants, les professeurs, et les staffes administratives.
- **Un cas d'utilisation** représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier, la figure suivante représente les différents cas d'utilisations offerts par l'application rassemblés dans des packages qui seront bien détaillés dans les figures qui suivent.

4.4.1 Diagramme de cas d'utilisation pour un responsable ou agent de point de vente



4.4.2 Diagramme de cas d'utilisation pour un agent comptable

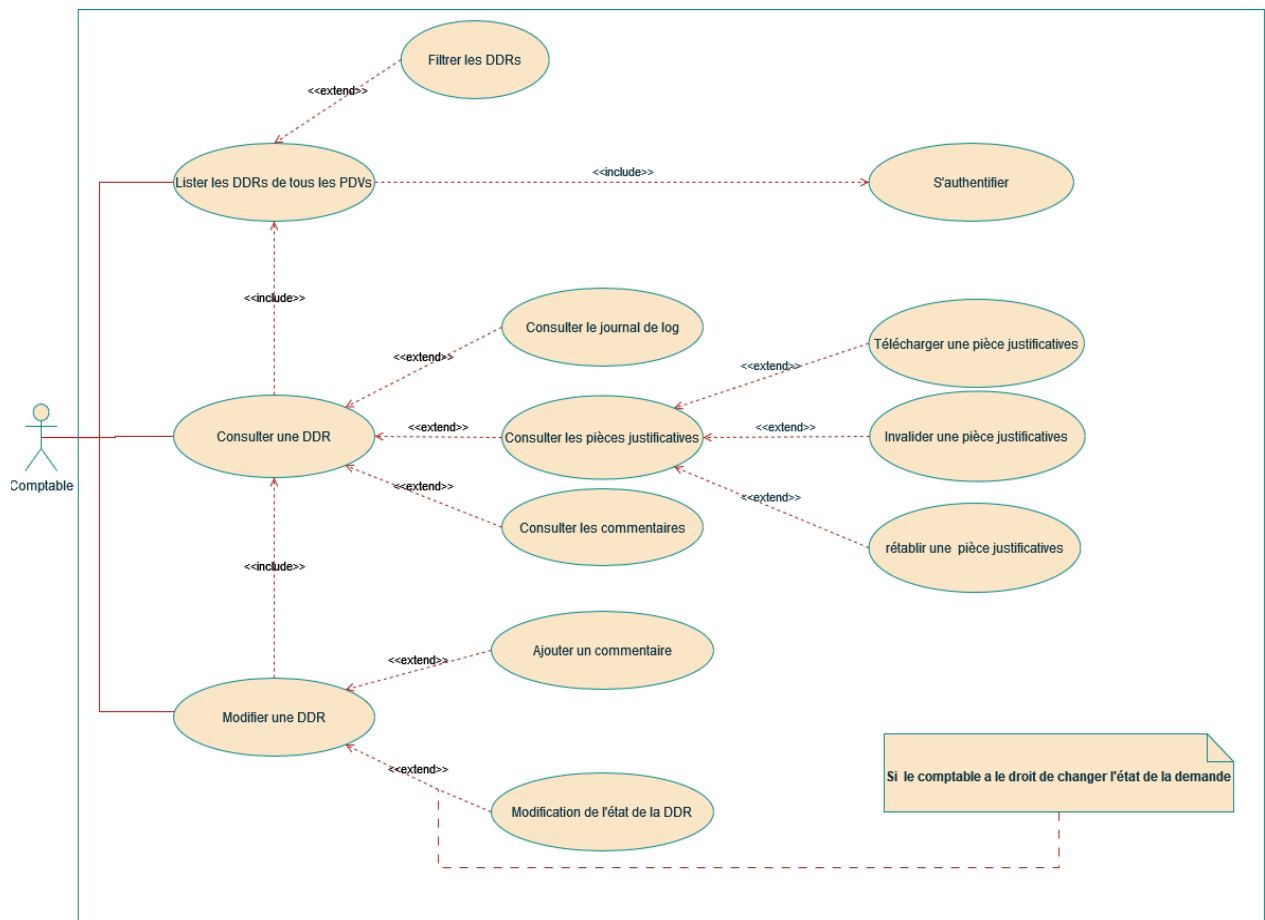


Figure 4.4 : Diagramme de cas d'utilisation pour un agent comptable

4.5 Diagramme de séquence

Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un Diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent à l'aide de messages.

➤ Lignes de vie dans les diagrammes de séquence :

Dans les diagrammes UML tels que les diagrammes de séquence ou de communication, les lignes de vie représentent les objets entrant dans une interaction. Dans un scénario d'une banque, par exemple, les lignes de vie peuvent représenter des objets comme le système bancaire ou un client. Chaque instance d'une interaction est représentée par une ligne de vie.

➤ Messages dans les diagrammes UML

Un message est un élément de diagramme Unified Modeling Language (UML) qui définit un type particulier de communication entre les instances au cours d'une interaction. Un message fait circuler des informations d'une instance, représentée par une ligne de vie, à une autre instance au cours d'une interaction.

4.5.1 Diagramme de séquence d'authentification

Dans le diagramme de séquence ci-dessous, nous présentons le mécanisme d'authentification qui commence par une authentification régulière auprès du portail de services sur l'Intranet de SFR. Ensuite, le choix du service de remboursement nous redirige vers notre application où nous récupérerons en détail les données des utilisateurs et construisons le JWT (JSON Web Token) afin qu'ils puissent accéder aux fonctionnalités de l'application.

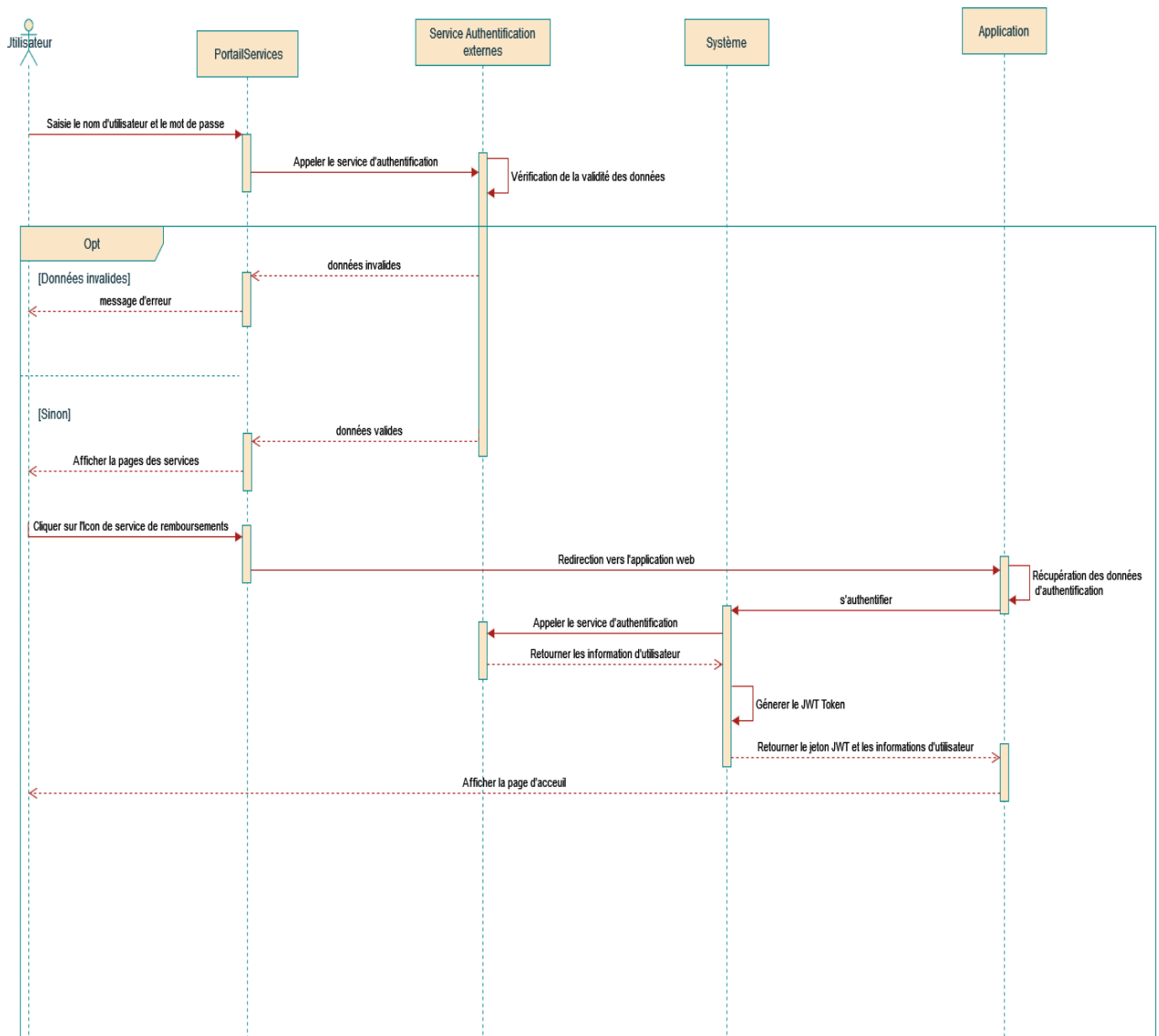


Figure 4.5 : Diagramme de séquence d'authentification

4.5.2 Diagramme de séquence du changement de statut

Le diagramme de séquence ci-dessous illustre le flux de changement de statut. Tout d'abord, une vérification est effectuée pour valider les droits de l'utilisateur. Ensuite, une seconde vérification est réalisée pour garantir que le nouveau statut choisi est parmi les transitions autorisées. (Les figures : « Figure 4.7 et Figure 4.8 » montrent le flux de changement de statut en fonction des deux états existants)

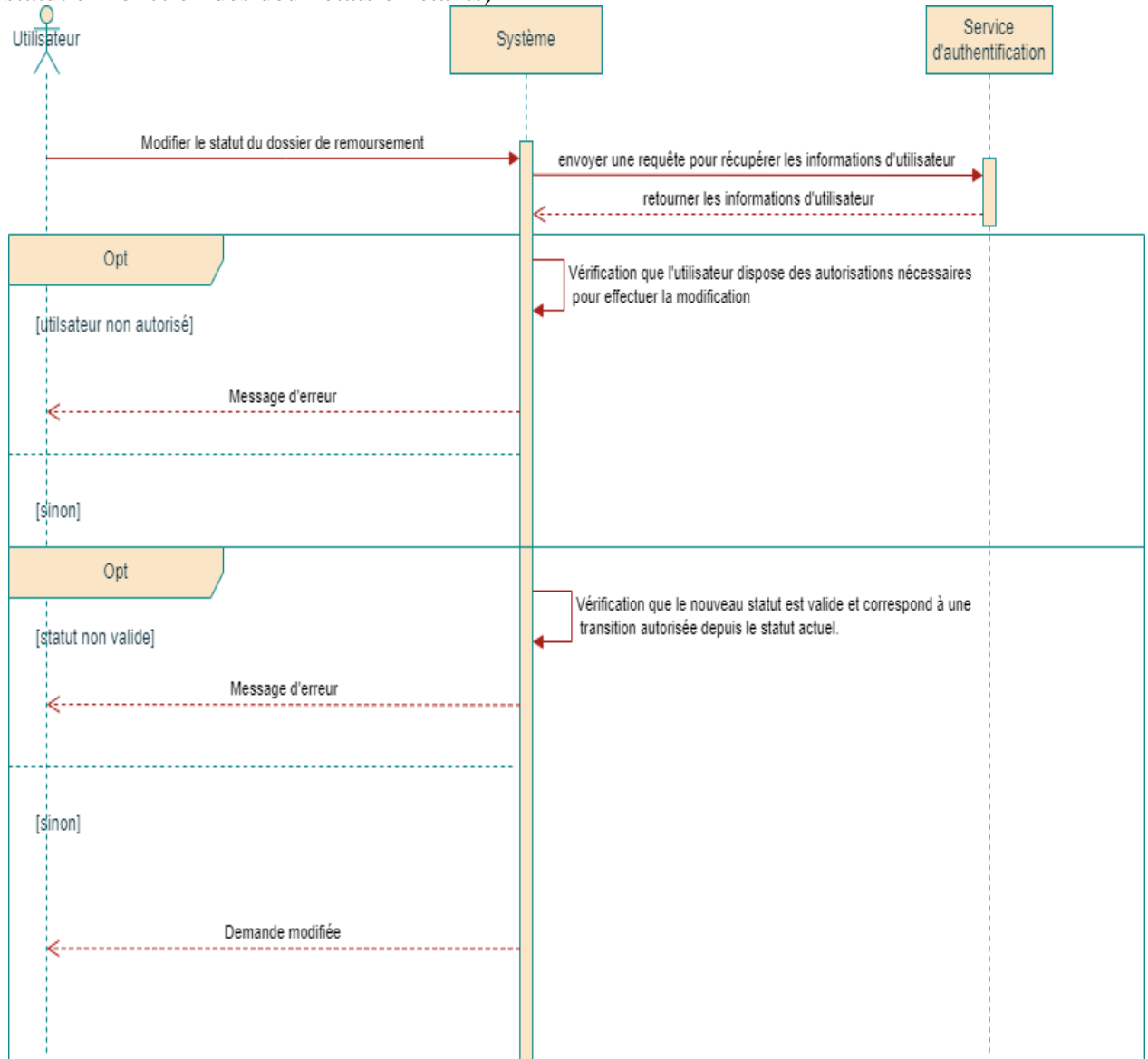


Figure 4.6: Diagramme de séquence du changement de statut

4.6 Diagramme d'activité

Le diagramme d'activité est un outil puissant pour représenter visuellement les flux de travail et les processus d'une application. Il offre une vue globale des différentes étapes, actions et décisions qui se déroulent lors de l'exécution d'une tâche ou d'un scénario. Ce diagramme met en évidence la séquence des actions, les conditions et les boucles, permettant ainsi de comprendre le déroulement logique de chaque activité.

Les rubriques suivantes décrivent les éléments de modèle dans les diagrammes d'activité :

- **Activités :**

Dans la modélisation UML, les activités sont des éléments de conteneur qui décrivent le niveau de comportement plus haut dans un diagramme d'activité. Les activités contiennent différents noeuds d'activité et bords d'activité qui représentent la séquence de tâches dans un flux de travaux débouchant sur un comportement.

- **Actions :**

Dans la modélisation UML, une action représente une unité de fonctionnalité distincte dans une activité.

- **Noeuds de commande :**

Dans les diagrammes d'activité, un nœud de commande est un nœud d'activité abstrait qui coordonne le flux de commande dans une activité.

- **Noeuds d'objet :**

Dans les diagrammes d'activité, un nœud d'objet est un nœud d'activité abstrait qui permet de définir le flux d'objet dans une activité. Un nœud d'objet indique qu'une instance d'un discriminant peut être disponible à un point particulier de l'activité.

- **Bords d'activité :**

Dans un diagramme d'activité, un arc d'activité est une connexion directe entre deux noeuds d'activité. Lorsqu'une action donnée est terminée dans une activité, le bord d'activité continue le flux jusqu'à la prochaine action dans la séquence.

Dans notre projet, le diagramme d'activité joue un rôle essentiel dans la modélisation et la compréhension des flux de travail liés aux demandes de remboursement. Il nous permet d'analyser et d'optimiser les processus, d'identifier les points de décision et de coordination, et d'assurer une exécution fluide et efficace des tâches. En utilisant des symboles et des notations spécifiques, le diagramme d'activité offre une représentation claire et concise des actions et des interactions entre les différents acteurs et les systèmes impliqués. Il constitue un outil précieux pour la communication, la documentation et l'amélioration continue de notre application de gestion des remboursements.

4.6.1 Diagramme d'activité du cas de remboursement sur facture

Dans ce scénario particulier, la demande de remboursement n'est pas directement initiée par notre service, mais par un service externe qui utilise notre service pour créer la demande, à condition qu'un accord ait été obtenu. Pendant ce processus, le service externe effectue également une opération de "griller" l'avoir associé, le rendant ainsi indisponible et inutilisable pour d'autres transactions. Cette étape de "grillage de l'avoir" vise à éviter toute utilisation ultérieure de l'avoir déjà utilisé pour la demande de remboursement en question, ce qui garantit la cohérence et l'intégrité du processus de remboursement. Une fois la

demande de remboursement créée, toutes les actions ultérieures sont effectuées au sein de notre application, y compris le suivi et le traitement de la demande.

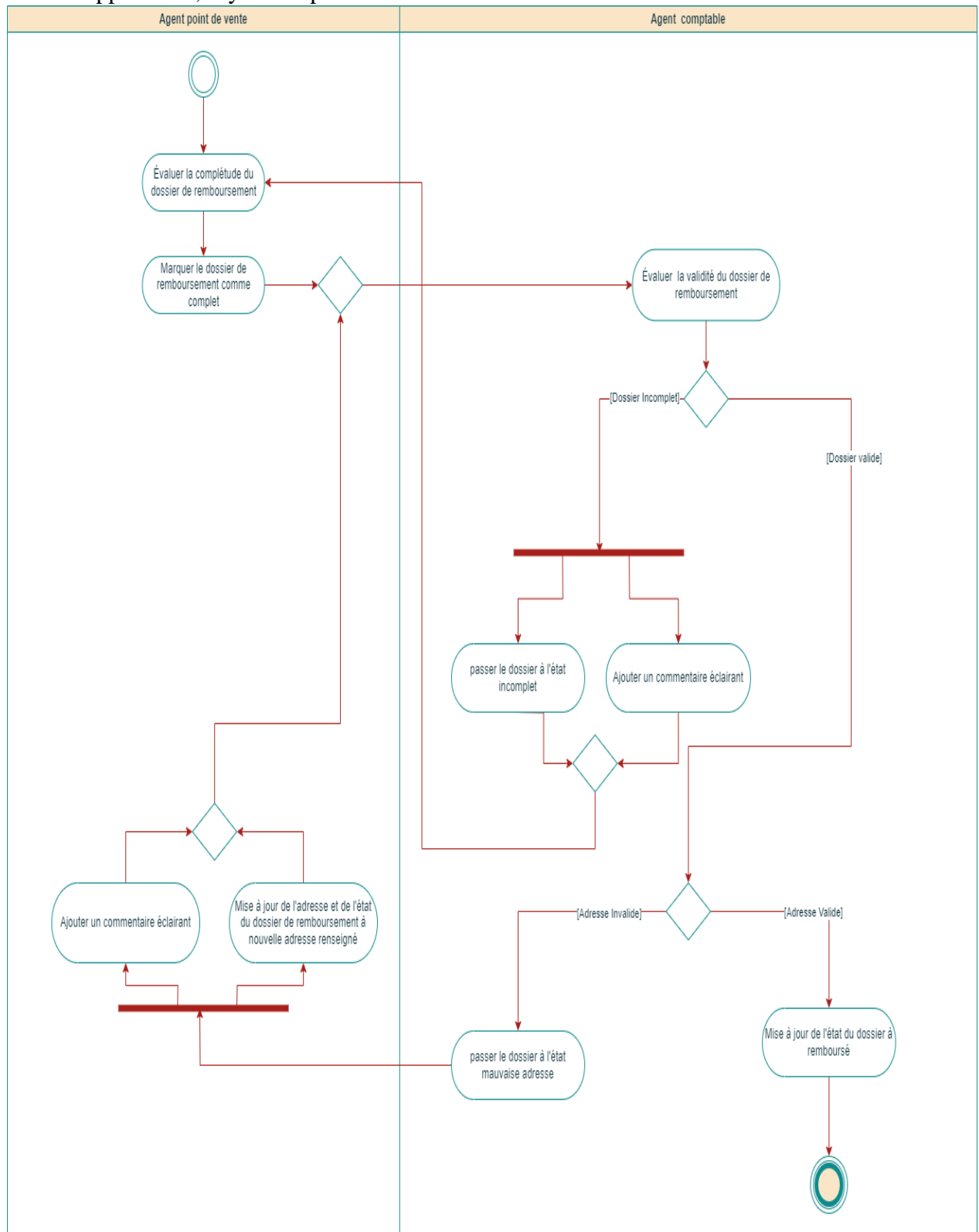


Figure 4.7: Diagramme d'activité du cas de remboursement sur facture

4.6.2 Diagramme d'activité du cas de la transformation d'avoir en remboursement

Dans ce scénario, la demande de remboursement est créée depuis notre application par l'agent de point de vente. Toutes les opérations liées à la gestion du remboursement sont ensuite effectuées automatiquement dans notre application. Au moment de la création de la demande, aucun accord n'est obtenu immédiatement. Cependant, lors de la phase de validation de la demande de remboursement, nous procédons à griller l'avoir. Cela signifie que suite à l'obtention d'un accord, nous marquons l'avoir associé comme étant indisponible et inutilisable.

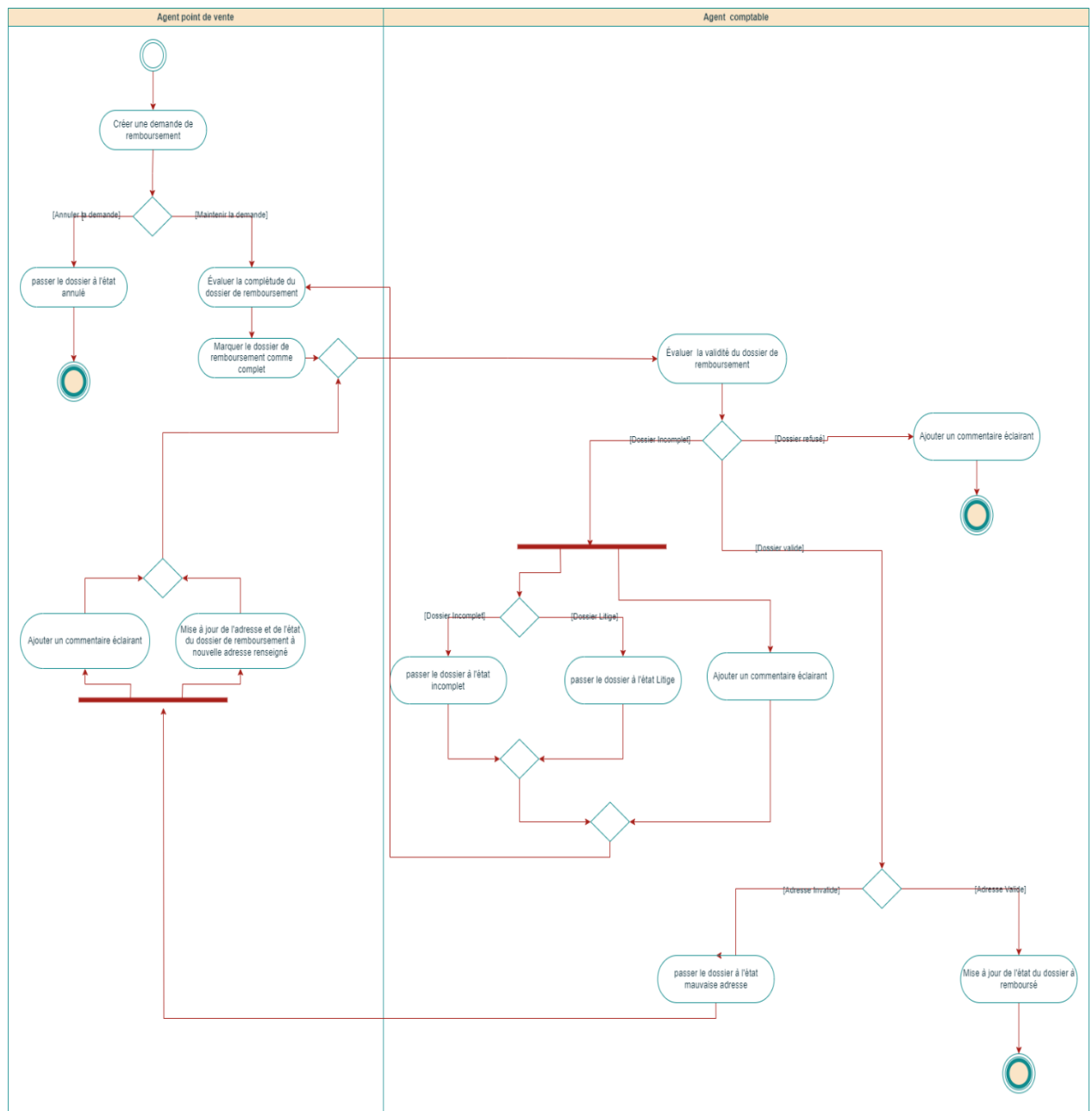


Figure 4.8: Diagramme d'activité du cas de la transformation de l'avoir en remboursement

4.7 Conclusion

Dans ce chapitre, nous avons réalisé la conception détaillée de l'application de gestion des remboursements d'avoir. Nous avons utilisé divers diagrammes pour modéliser les acteurs, les activités et les interactions clés. La prochaine phase de réalisation comprendra l'exploration de l'environnement de travail, y compris les outils et technologies utilisés. Nous aborderons également l'approche de développement et présenterons les différentes interfaces de l'application.

Chapitre 5: Réalisation

5.1 Introduction

Après une analyse approfondie du contexte du projet et des besoins fonctionnels et non fonctionnels de l'application web, nous entamons maintenant la phase de réalisation. Cette étape consiste à traduire la conception du système en code informatique, permettant ainsi l'automatisation des fonctionnalités définies dans la spécification. La première partie de ce chapitre se concentre sur l'identification de l'environnement de travail, y compris les outils et technologies utilisés. Ensuite, nous abordons le processus de développement. Enfin, nous présentons l'application elle-même à travers des captures d'écran de ses différentes interfaces, mettant en évidence son fonctionnement.

5.2 Outils et technologie

Dans le cadre du développement de notre application de gestion de remboursement des avoirs complexes, nous avons utilisé une combinaison d'outils et de technologies modernes pour assurer une mise en œuvre efficace et robuste. Pour la partie front-end de l'application, nous avons utilisé le Framework Angular, associé à Material UI, Sass et NgRx, pour créer des interfaces utilisateur réactives, esthétiques et conviviales.



Angular est un Framework web open source développé par Google. Il facilite la création d'applications web dynamiques en utilisant TypeScript comme langage de programmation principal. Angular offre des fonctionnalités avancées telles que la liaison de données bidirectionnelle et la gestion des composants, ce qui permet de construire des applications réactives et modulaires.



TypeScript est un langage de programmation libre et open source développée par Microsoft qui a pour but d'améliorer et de sécuriser la production de code JavaScript. Il s'agit d'un sur-ensemble syntaxique strict de JavaScript (c'est-à-dire que tout code JavaScript correct peut être utilisé avec TypeScript). Le code TypeScript est transcompilé en JavaScript, et peut ainsi être interprété n'importe quel navigateur web ou moteur JavaScript.



Sass (Syntactically awesome stylesheets) est un langage de script préprocesseur qui est compilé ou interprété en CSS (Cascading Style Sheets). SassScript est le langage de script en lui-même. Sass est disponible en deux syntaxes. La syntaxe originale, appelée "syntaxe indentée" qui utilise l'indentation pour séparer les blocs de code et les sauts de ligne pour les séparer les directives.



Material-UI est une bibliothèque d'interface utilisateur (UI) pour le Framework Angular, qui permet de créer des applications web modernes et esthétiquement attrayantes. Basée sur le concept de Material Design développé par Google, Material-UI fournit des composants réutilisables prêts à l'emploi, des styles préconfigurés et des outils de personnalisation flexibles pour faciliter le développement d'interfaces cohérentes et réactives.



NgRx est une bibliothèque de gestion d'état pour Angular qui s'appuie sur le modèle Redux. Elle facilite la gestion des données et des actions de l'application en proposant des fonctionnalités telles que les reducers, les actions et les sélecteurs.

Tableau 5.1 : Technologies utilisées côté frontend

Concernant la partie backend, nous avons opté pour le langage de programmation Java et le Framework Spring Boot, qui nous ont permis de développer rapidement des services web RESTful. Nous avons utilisé une base de données PostgreSQL pour la gestion de la persistance des données. De plus, nous avons intégré Amazon S3 pour le stockage sécurisé et évolutif des fichiers associés à notre application.



Spring Boot est un Framework open-source basé sur Java qui facilite le développement d'applications web et d'applications autonomes (standalone). Il fournit une approche simplifiée pour créer rapidement des applications Java en fournissant une configuration par défaut et des conventions de développement intelligentes.



PostgreSQL est un système de gestion de base de données relationnelle open-source. Il est connu pour être robuste, extensible et offrir de puissantes fonctionnalités. PostgreSQL est compatible avec les normes SQL et est largement utilisé dans le développement d'applications web, les systèmes d'entreprise et les projets de données volumineuses.



Amazon S3 (Simple Storage Service) est un service de stockage d'objets proposé par Amazon Web Services (AWS). Il permet aux entreprises et aux développeurs de stocker et de récupérer de grandes quantités de données de manière sécurisée, fiable et scalable sur le cloud.



Spring Security est un Framework de sécurité pour les applications Java. Il fournit des fonctionnalités et des bibliothèques qui permettent de gérer l'authentification, l'autorisation, la protection contre les attaques et d'autres aspects liés à la sécurité dans les applications Java.

Tableau 5.2: Technologies utilisées coté backend

En ce qui concerne l'environnement de développement, nous avons utilisé des IDE tels que IntelliJ IDEA et Visual Studio Code, qui offrent des fonctionnalités avancées de développement, de débogage et de productivité. De plus, Nous avons également utilisé GitLab, une plateforme de gestion du cycle de vie des applications, pour héberger notre dépôt Git et faciliter le suivi des modifications, la gestion des problèmes et la planification des versions.



IntelliJ IDEA est un environnement de développement intégré (IDE) utilisé principalement pour la programmation Java. Il offre une gamme complète d'outils et de fonctionnalités pour faciliter le développement logiciel, notamment l'édition de code, la gestion de projets, le débogage, la refactoring, les tests unitaires et bien plus encore.



Visual Studio Code (VS Code) est un éditeur de code source développé par Microsoft. C'est une application légère, gratuite et multiplateforme qui offre un large éventail de fonctionnalités pour le développement de logiciels. VS Code prend en charge de nombreux langages de programmation et propose des fonctionnalités telles que la coloration syntaxique, l'autocomplétion, le débogage, la gestion de versions, les extensions et bien plus encore.



GitLab est une plateforme de gestion de code source basée sur Git, un système de contrôle de version populaire. Il s'agit d'un outil de développement logiciel collaboratif qui permet aux équipes de travailler ensemble sur des projets, de versionner leur code, de suivre les modifications et de gérer les différentes branches de développement.

Tableau 5.3: Liste des outils de l'environnement de travail

5.3 Approche de développement

Dans notre projet, nous avons adopté une approche de développement basée sur l'intégration continue (CI) et les tests automatisés. Nous avons mis en place un flux de travail automatisé où chaque fois que des modifications sont poussées vers une branche spécifique, elles sont automatiquement intégrées, testées à l'aide de tests unitaires et d'intégration, puis déployées dans un environnement séparé.

Ce processus d'intégration continue et de tests automatisés nous permet de détecter rapidement les erreurs de code, de garantir une qualité élevée et de réduire les conflits lors de la fusion des modifications. Les tests unitaires nous permettent de vérifier le bon fonctionnement des différentes parties de notre code de manière isolée, tandis que les tests d'intégration s'assurent que les différentes composantes de notre application fonctionnent correctement ensemble.

L'environnement séparé dans lequel les modifications sont déployées permet au client et aux parties prenantes de consulter l'application en cours de développement, d'interagir avec celle-ci et de fournir des commentaires et des remarques précieuses. Cette approche favorise une collaboration transparente avec le client, permettant un ajustement continu en fonction des besoins et des retours d'expérience.

En combinant l'intégration continue, les tests automatisés et la collaboration étroite avec le client, nous sommes en mesure d'assurer une livraison régulière et itérative des fonctionnalités. Cela nous permet de garantir la qualité des fonctionnalités développées et leur conformité aux attentes du client, tout en facilitant les ajustements et les améliorations au fur et à mesure de l'avancement du projet.

5.4 Présentation de l'application web

Dans cette section, nous offrons un aperçu des différentes interfaces qui constituent notre application. Cependant, il est important de noter que certaines parties spécifiques, comme le mécanisme d'authentification, ne seront pas présentées ici. Cette fonctionnalité est intégrée à une autre application qui regroupe l'ensemble des services, y compris notre application. À partir de cette application principale, les utilisateurs peuvent accéder automatiquement à notre application sans avoir à passer par une authentification séparée.

- **Page d'accueil**

Lorsque les utilisateurs accèdent à notre application, ils sont immédiatement dirigés vers une liste de notifications. Ces notifications sont essentielles pour signaler les différentes actions ou tâches qui exigent l'attention de l'utilisateur. Elles servent à communiquer les demandes spécifiques qui nécessitent une action ou une prise de décision.

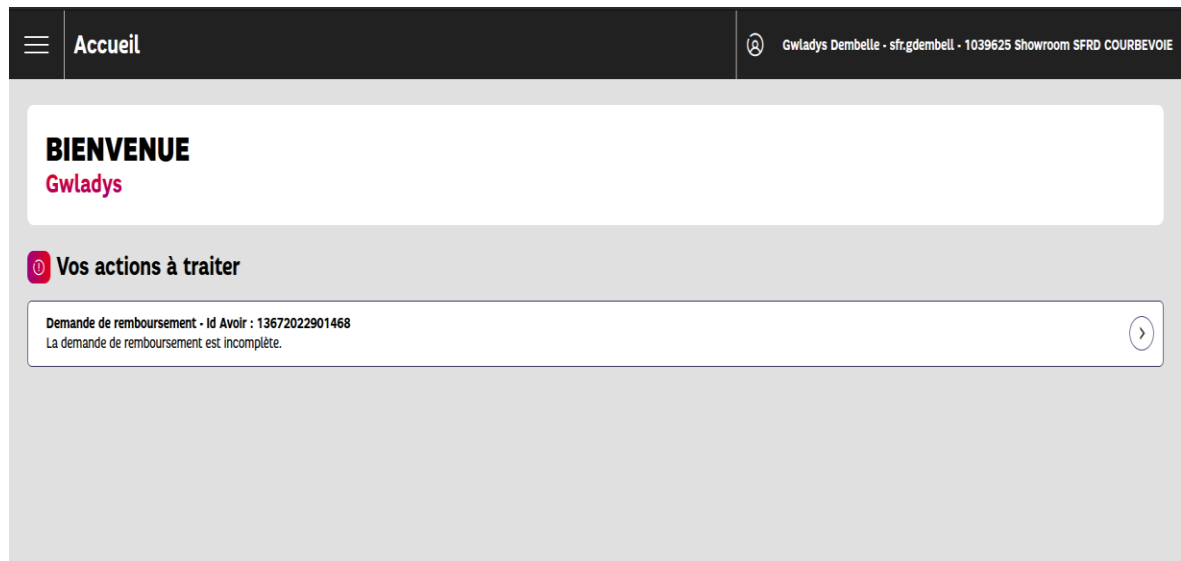


Figure 5.1: Page d'accueil

Il est important de souligner que les notifications sont personnalisées en fonction des rôles de chaque utilisateur. Par conséquent, les notifications affichées peuvent varier en fonction du rôle de l'utilisateur, de ses autorisations. Par exemple, un responsable de point de vente aura uniquement accès aux notifications liées à son magasin spécifique qu'il a sélectionné. De plus, il a la possibilité de changer le point de vente sélectionné afin de consulter les notifications d'autres magasins.

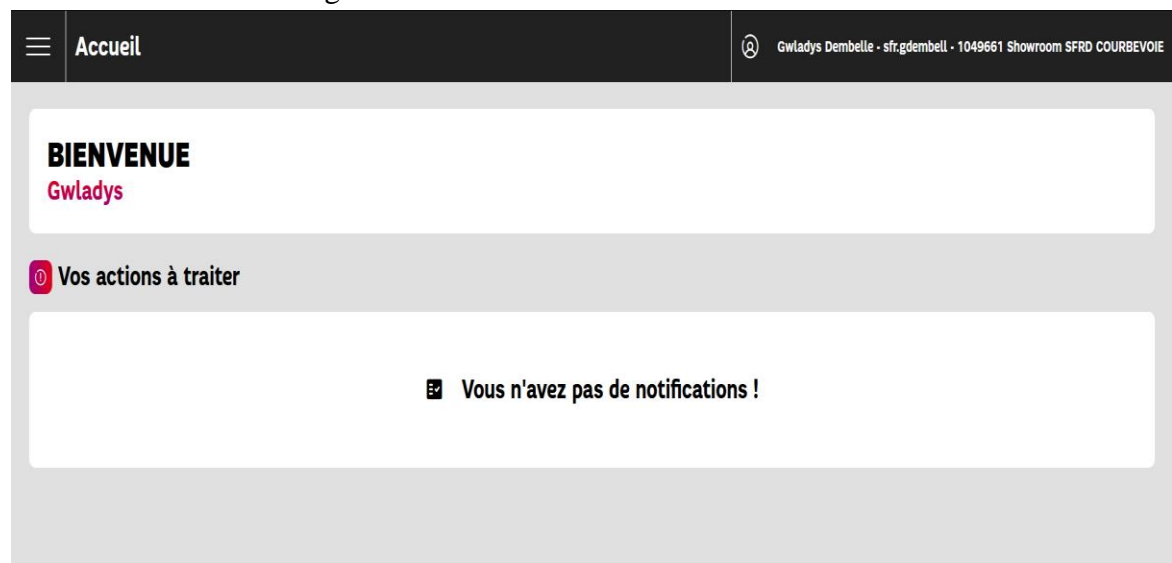


Figure 5.2: Page d'accueil - Changement du point de vente

- **Page liste des demandes de remboursements**

Cette interface comprend également une liste des demandes de remboursement. Cette liste offre aux utilisateurs une vue d'ensemble de toutes les demandes en attente de traitement. Chaque demande est présentée avec des informations clés telles que le numéro de l'avoir, le montant, le nom du client, et l'état actuel de la demande. Ainsi, en cliquant sur la flèche située dans le tableau, nous sommes automatiquement redirigés vers la page de détail de la demande de remboursement correspondante

Suivi demande de remboursement						Gwladys Dembelle - sfr.gdembell - 1039625 Showroom SFRD COURBEVOIE	
Demandes de remboursement						Recherche avancée	+ Ajouter
Id Avoir	Statut	Id Client	Nom Client	Email Client	Solde		
13672022901243	CREE	770654611	LOLA SMITH	amal.benouiran.prestataire@sfr.com	15,51 €	>	
13672022901236	CREE	770654560	MICKAEL JACKSON	fabien.couly@sfr.com	0,00 €	>	
13672022901250	CREE	770654622	HECTOR LOZANO	fabien.couly@sfr.com	0,00 €	>	
13672022901259	CREE	770654645	SALMA HAYEK	amal.benouiran.prestataire@sfr.com	0,00 €	>	
13672022901263	CREE	770654679	ANNIE CROCHE	marieclaire.jasko.prestataire@sfr.com	0,00 €	>	
13672022901265	CREE	770654685	BELLA CLOONEY	amal.benouiran.prestataire@sfr.com	0,00 €	>	
13672022901268	CREE	CLI_1039625	ANONYME ANONYME	test@test.com	12,99 €	>	
13672022901468	INCOMPLET	770654724	ONCE AGAIN	amal.benouiran.prestataire@sfr.com	1 129,00 €	>	
						< 1 - 8 of 8 >	

Figure 5.3 : Page liste des demandes de remboursements

Les utilisateurs peuvent effectuer des recherches en utilisant des critères tels que l'identifiant d'avoir, le statut de la demande, ainsi que des informations liées au client comme le nom et le prénom. Ces fonctionnalités de tri et de filtrage permettent aux utilisateurs de trouver rapidement les demandes spécifiques qu'ils souhaitent consulter ou traiter

Suivi demande de remboursement						Gwladys Dembelle - sfr.gdembell - 1039625 Showroom SFRD COURBEVOIE	
Demandes de remboursement						Recherche avancée	+ Ajouter
Filtrer par dossier			Filtrer par client				
Statut INCOMPLET			N° d'avoir				
			Rechercher Réinitialiser				
Id Avoir	Statut	Id Client	Nom Client	Email Client	Solde		
13672022901468	INCOMPLET	770654724	ONCE AGAIN	amal.benouiran.prestataire@sfr.com	1 129,00 €	>	
						< 1 - 1 of 1 >	

Figure 5.4 : Page liste des demandes de remboursements – filtrage

Si nous appliquons un filtre sur les statuts en sélectionnant le statut "Valide" et que l'acteur effectuant l'action est un agent comptable, il est possible de modifier en masse le statut des demandes de remboursement sélectionnées. Cette fonctionnalité permet d'optimiser le processus de gestion en évitant de modifier manuellement chaque demande individuellement. En un seul clic, l'agent comptable peut mettre à jour les statuts des

demandes de remboursement correspondantes, ce qui permet de gagner du temps et d'assurer une cohérence dans le traitement des demandes.

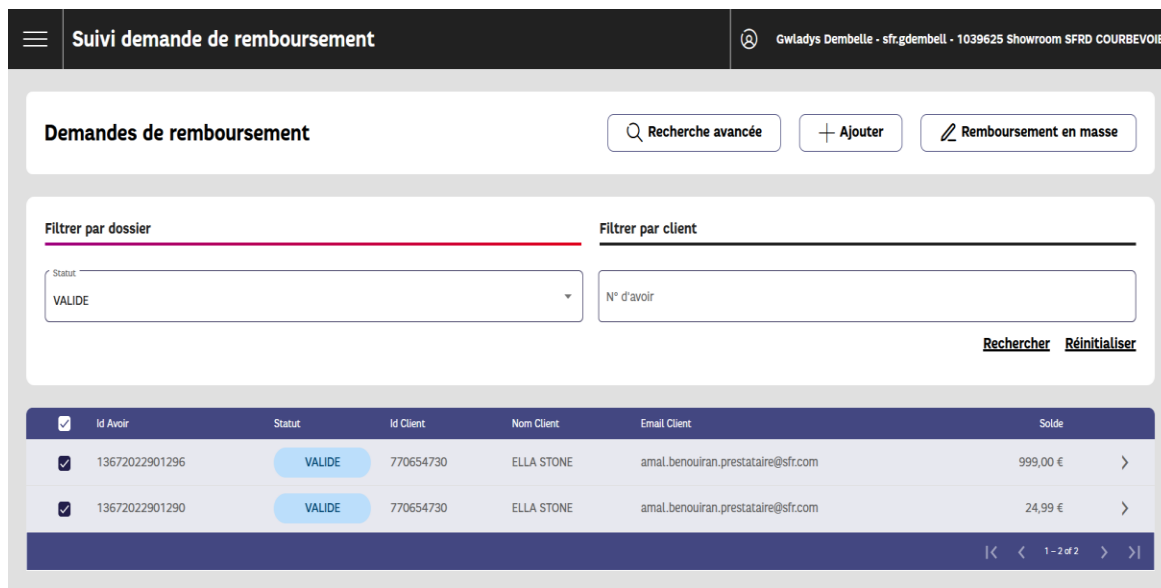


Figure 5.5 : Page liste des demandes de remboursements – changement de statues en masse
Et si vous cliquez sur le bouton "Remboursement en masse", un pop-up s'affichera pour confirmer l'action.

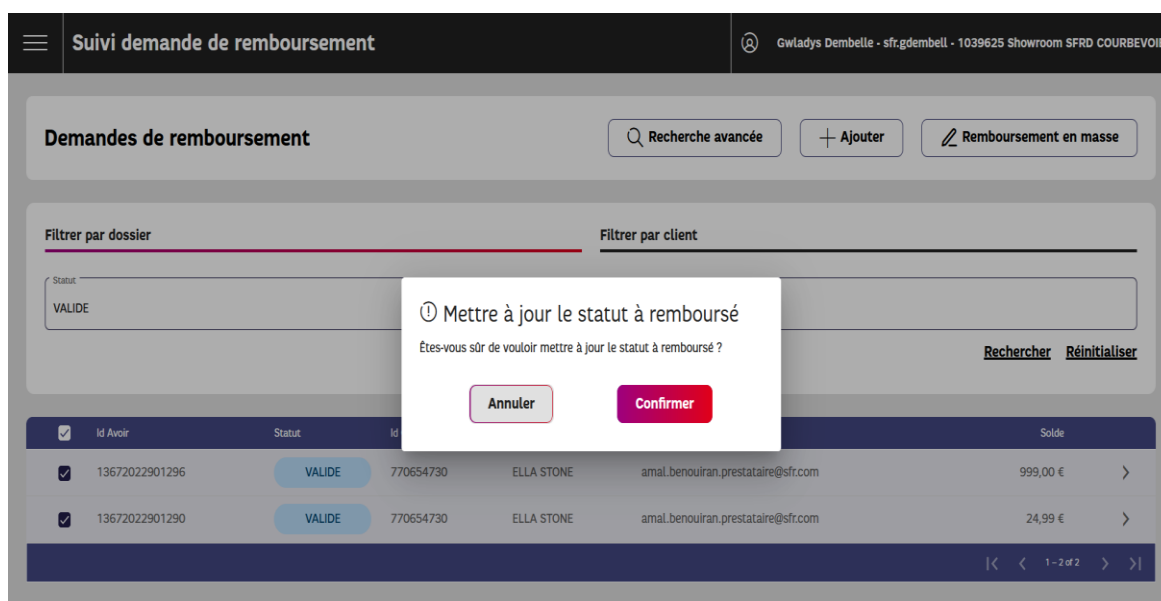


Figure 5.6 : Page liste des demandes de remboursements – confirmation du changement du statue en masse

Lorsqu'on clique sur le bouton « Ajouter » (Figure 5.3), l'interface de création de la demande se présente sous la forme d'une fenêtre contextuelle (popup) qui s'affiche lorsque les utilisateurs souhaitent créer une nouvelle demande. Cette interface offre un formulaire structuré où les utilisateurs peuvent saisir toutes les informations nécessaires pour la demande de remboursement. Le formulaire comprend des champs

tels que l'identifiant de l'avoir, le canal et le IBAN si le canal du remboursement est à travers le virement.

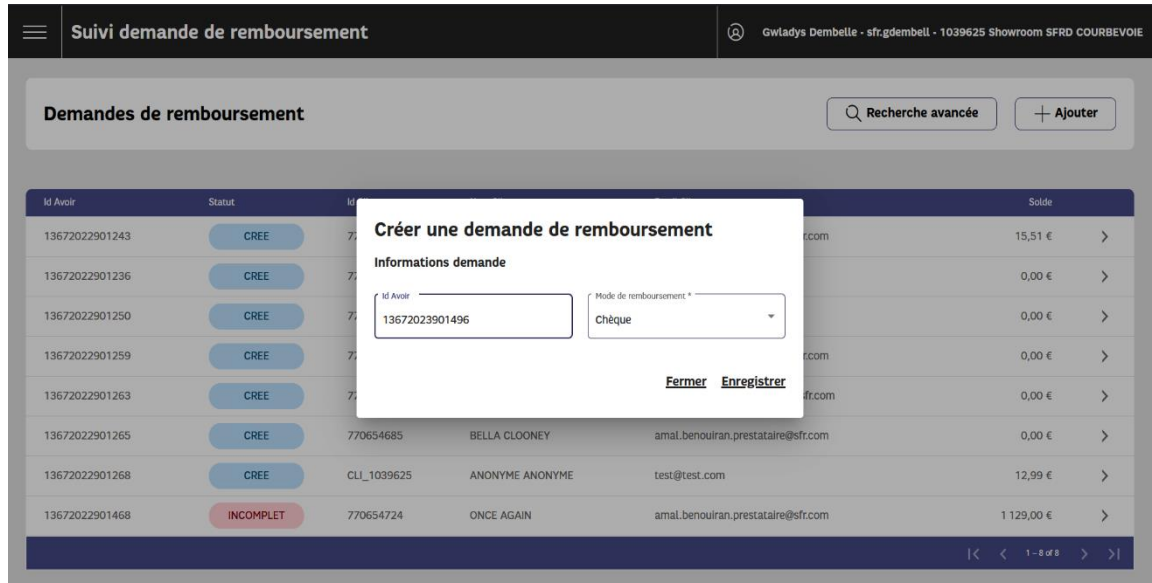


Figure 5.7: Page liste des demandes de remboursements - création de la demande

- **Page détail de la demande de remboursement**

Une fois que la demande de remboursement a été créée (Figure 5.7 **Error! Reference source not found.**), on va être automatiquement redirigé vers la page de détail correspondante. Cette page offre une vue complète de toutes les informations essentielles liées à la demande, notamment les détails spécifiques de la demande, les informations du client, la liste des pièces justificatives associées, les commentaires échangés et l'historique des actions effectuées.

Dans la figure suivante, vous pouvez voir le message de confirmation indiquant que la demande de remboursement a été créée avec succès. Ce message s'affiche dans la page de détail de la demande, confirmant ainsi que toutes les informations ont été enregistrées correctement.

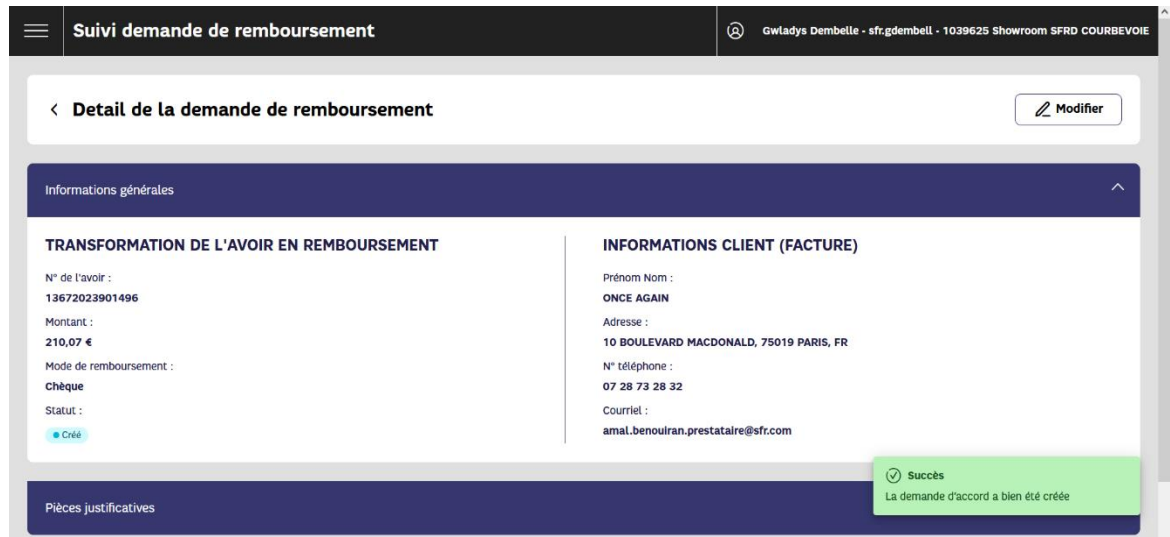


Figure 5.8: Page détail

Dans le cas où les informations de la demande ont été modifiées, en plus des informations de la demande et du client, une section distincte est affichée pour mettre en évidence les éléments modifiés.

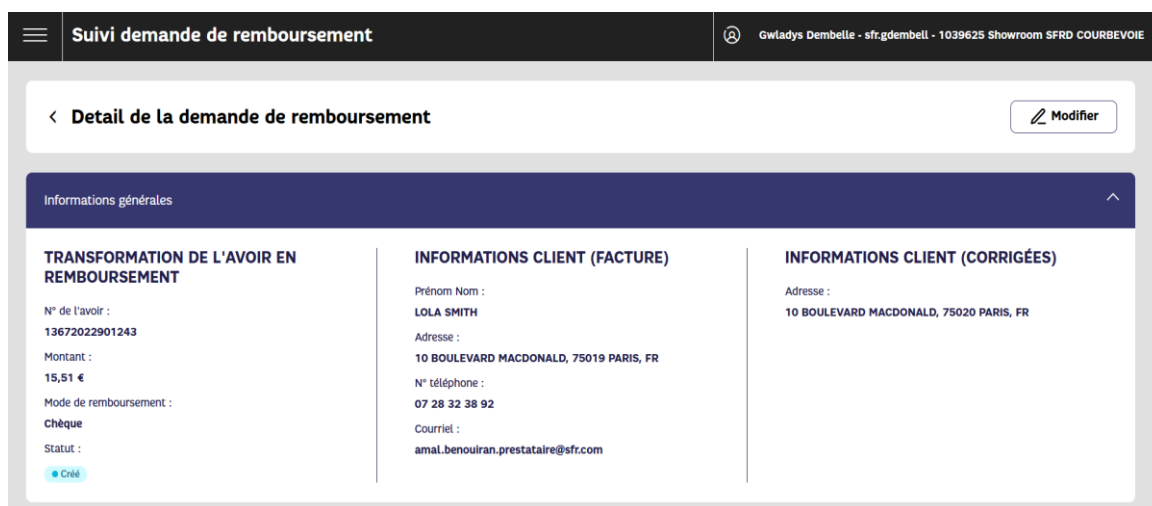


Figure 5.9: Page détail - Client modifié

Lorsqu'on clique sur le bouton « Modifier » (Figure 5.8Figure 5.8), un stepper s'affiche, présentant les différentes étapes permettant de modifier une demande de remboursement. Ce stepper vous guide tout au long du processus de modification en vous présentant chaque étape de manière claire et structurée.

La première étape de ce processus concerne la demande de remboursement elle-même. Vous pouvez apporter les modifications nécessaires aux détails de la demande, tels que le statut, et le canal de remboursement.

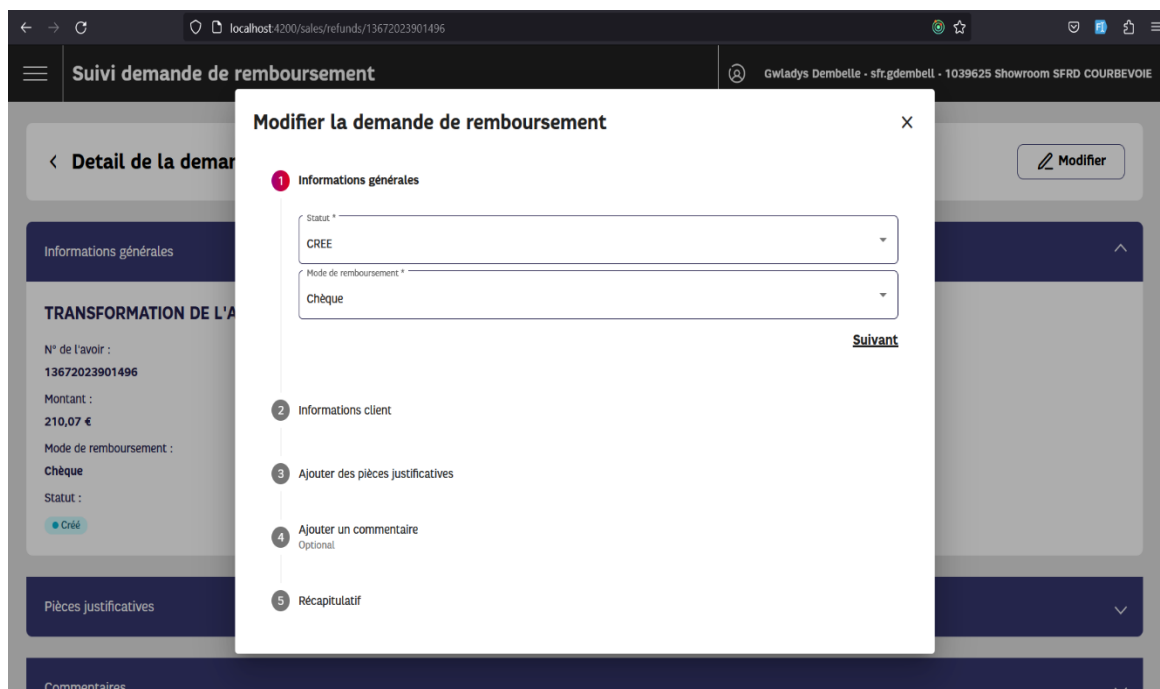


Figure 5.10: Modification - Informations de la demande de remboursement

La deuxième étape du processus de modification concerne les informations du client. Vous pouvez mettre à jour les détails du client associé à la demande de remboursement, tels que son nom, son prénom, ses coordonnées, etc.

The screenshot shows a web application interface for managing a refund request. A modal titled "Modifier la demande de remboursement" is open, displaying step 2 of a 5-step process: "Informations client". The background shows a sidebar with "Suivi demande de remboursement" and "Detail de la demar", and a main area with "TRANSFORMATION DE L'A" and various fields like "N° de l'avoir : 13672023901496", "Montant : 210,07 €", "Mode de remboursement : chèque", and "Statut : Créé".

The modal form includes the following fields:

- 1 Informations générales
- 2 Informations client (active step)
- 3 Ajouter des pièces justificatives
- 4 Ajouter un commentaire
- 5 Récapitulatif

The "Informations client" section contains the following input fields:

- Ville client * : PARIS
- Pays client * : FR
- Code postal client * : 75019
- Adresse client * : 10 BOULEVARD MACDONALD
- Email client * : amal.benouiran.prestataire@sfr.com
- Téléphone client * : 0728732832

Navigation links "Précédent" and "Suivant" are visible at the bottom right of the modal.

Figure 5.11: Modification - Informations client

La troisième étape du processus de modification concerne les pièces justificatives. On a la possibilité de sélectionner plusieurs fichiers justificatifs pertinents pour la demande de remboursement. Lors d'ajout des fichiers, on doit spécifier le type de chaque document, tels que le RIB, la carte d'identité, etc.

The screenshot shows the same web application interface, but the modal is now on step 3: "Ajouter des pièces justificatives". The sidebar and background content remain the same.

The modal form shows the following steps:

- 1 Informations générales
- 2 Informations client
- 3 Ajouter des pièces justificatives (active step)
- 4 Ajouter un commentaire
- 5 Récapitulatif

The "Ajouter des pièces justificatives" section includes:

- A "Parcourir..." button and the text "Aucun fichier sélectionné."
- A "Type de fichier *" dropdown menu with "RIB client" selected.

Navigation links "Précédent" and "Suivant" are visible at the bottom right of the modal.

Figure 5.12: Modification - Ajout des pièces justificatives

La quatrième étape du processus de modification concerne l'ajout de commentaires. Cette étape est facultative et intervient uniquement lorsque le changement de statut de la demande nécessite l'ajout d'un commentaire.

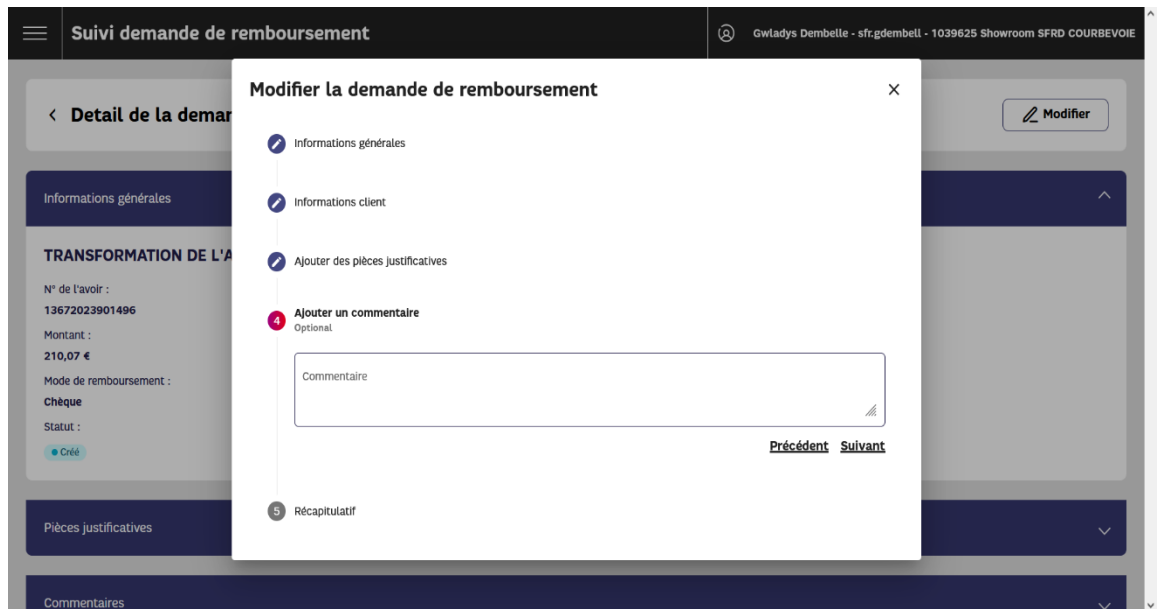


Figure 5.13: Modification - Ajout d'un commentaire

La dernière étape du processus de modification est un récapitulatif où on a la possibilité d'enregistrer les modifications effectuées, ce qui confirme les nouvelles informations associées à la demande de remboursement. Si on décide d'annuler les modifications, on peut également le faire à ce stade.

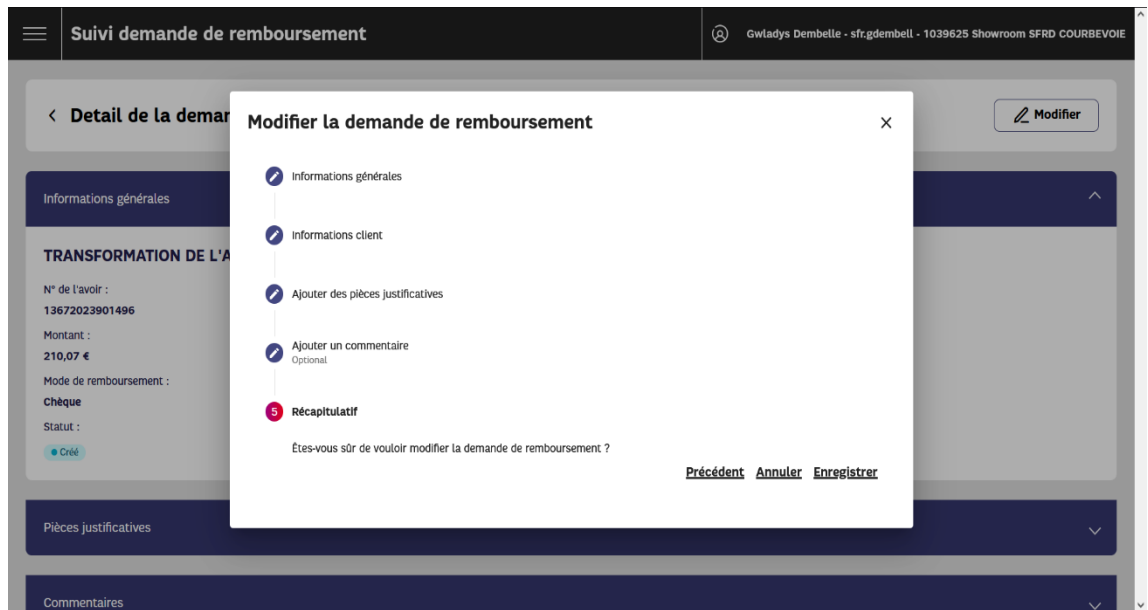


Figure 5.14: Modification – Récapitulatif

Dans la page de détail de la demande de remboursement, nous avons également un panneau d'extension qui affiche la liste des fichiers associés à cette demande. Pour chaque fichier valide, vous disposez de trois actions disponibles :

- ✓ **Télécharger:** On peut télécharger le fichier sur votre appareil afin de le conserver localement.

- ✓ **Visualiser:** On peut avoir la possibilité de visualiser le contenu du fichier directement dans l'application. Cela nous permet d'examiner le contenu du fichier sans avoir à le télécharger.
- ✓ **Invalider:** Si nécessaire, on peut invalider un fichier associé à la demande de remboursement. Cette action est utile si on constate qu'un fichier est incorrect, obsolète ou ne correspond pas aux critères requis.

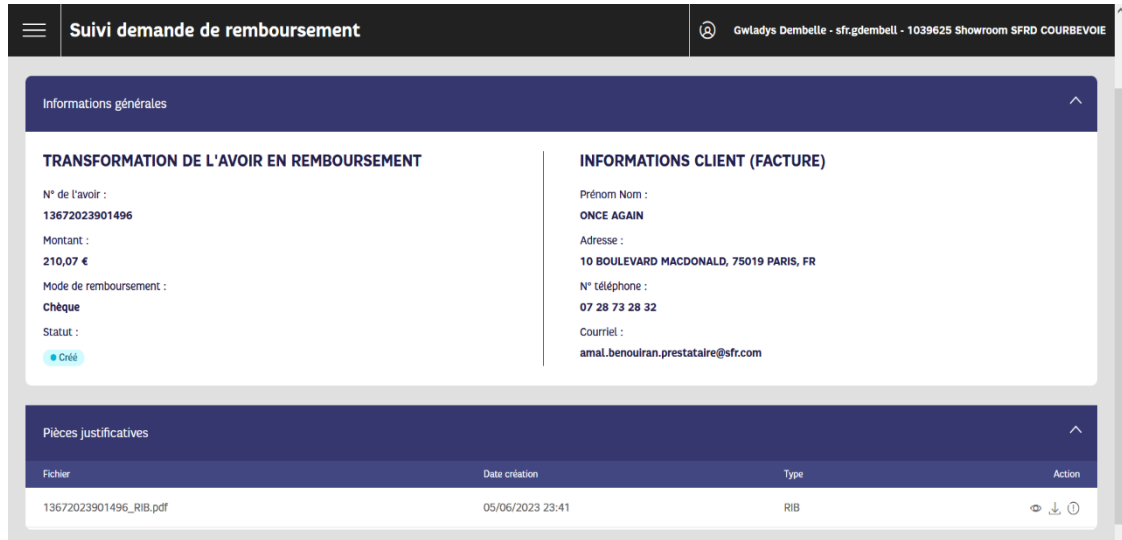


Figure 5.15: Liste des pièces justificatives

Lorsque vous cliquez sur l'icône de l'œil pour visualiser un fichier, une fenêtre contextuelle s'ouvre. Dans un premier temps, on verra que le fichier est en cours de chargement.

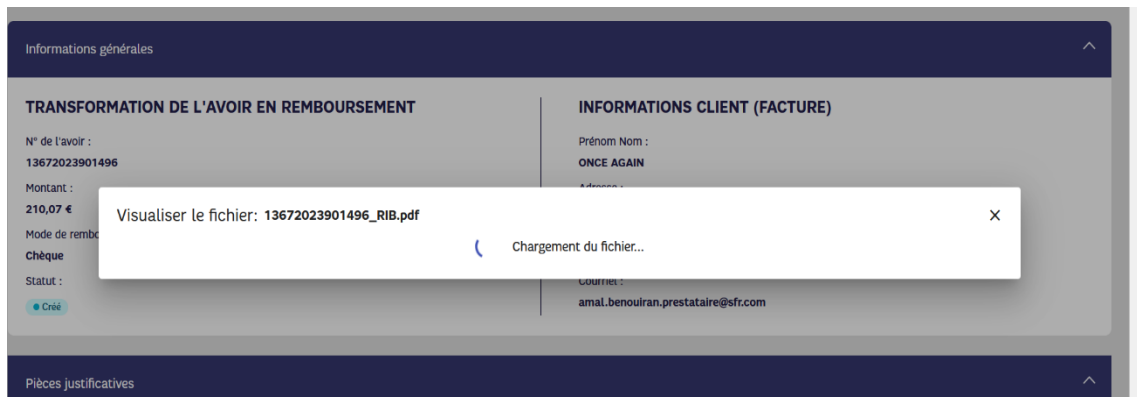


Figure 5.16: Visualiser le fichier - en cours de chargement

Une fois que le chargement est terminé, on pourra visualiser le contenu du fichier avec plusieurs options supplémentaires. On aura la possibilité de zoomer pour examiner les détails, de télécharger le fichier et même de rechercher des mots spécifiques dans le fichier.

Ces fonctionnalités avancées sont particulièrement utiles lorsque on travaille avec des fichiers PDF. On pourra naviguer à travers les pages, effectuer un zoom avant ou arrière pour une lecture plus précise et utiliser la fonction de recherche pour trouver rapidement des termes spécifiques dans le contenu du fichier.

Cependant, si le fichier associé est une image, on pourra simplement visualiser l'image sans les fonctionnalités spécifiques aux fichiers PDF.

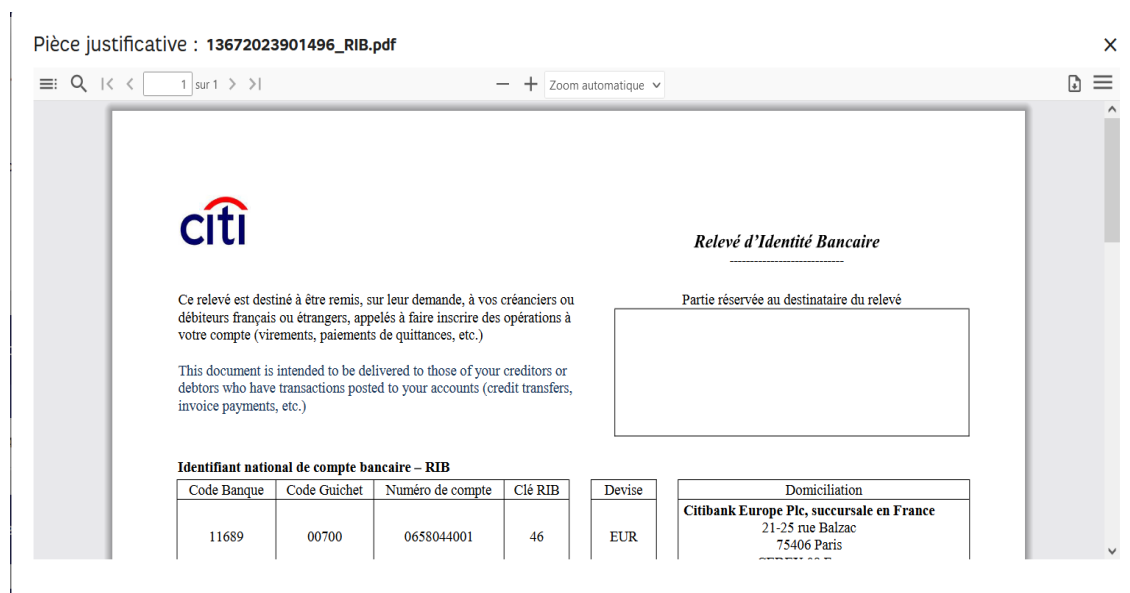


Figure 5.17: Visualiser un fichier

Si on souhaite invalider un fichier, on peut le faire en cliquant sur l'icône correspondante du fichier sélectionné. Une fois cela fait, une fenêtre contextuelle de confirmation s'affiche pour confirmer l'action d'invalidation. Cette étape supplémentaire vise à éviter toute action accidentelle ou indésirable sur les fichiers. En procédant ainsi, on aura la possibilité de confirmer ou d'annuler votre demande d'invalidation.

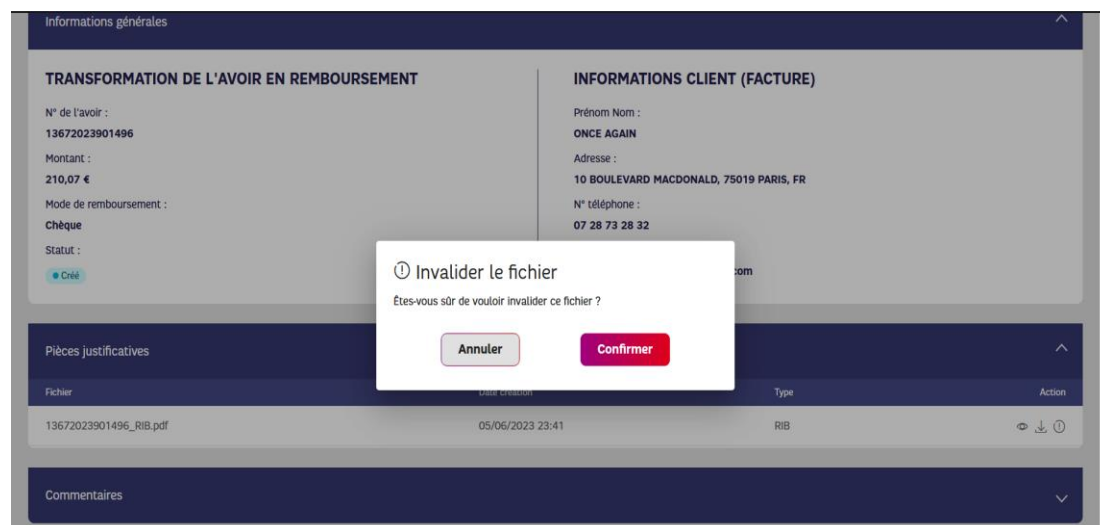


Figure 5.18: Invalider un fichier

Après avoir invalidé un fichier, on remarquera que son nom est modifié en ajoutant un préfixe "_INVALIDE". Cela permet de visualiser facilement les fichiers invalidés dans

la liste des fichiers associés à la demande de remboursement. Malgré l'invalidation, les icônes d'action de téléchargement et de visualisation restent disponibles, en permettant de télécharger ou de visualiser le fichier si nécessaire. Toutefois, l'icône d'invalidation est remplacée par l'icône de rétablissement. En cliquant sur cette nouvelle icône, on aura la possibilité de rétablir le fichier de son statut invalide à son statut valide, en cas de besoin ultérieur. Cela nous offre une flexibilité supplémentaire dans la gestion des fichiers associés à la demande de remboursement.

Pièces justificatives			
Fichier	Date création	Type	Action
13672023901496_RIB_INVALID.pdf	05/06/2023 23:41	RIB	  

Figure 5.19: Fichier Invalide

Dans la page de détail, vous avez la possibilité de consulter l'historique des événements liés à la demande de remboursement. Cet historique vous permet de suivre toutes les actions effectuées sur la demande, ainsi que l'utilisateur qui a effectué chaque action.

Historique des actions		
Action	Date création	Acteur
● Création de la demande de remboursement	05/06/2023 23:39	sfr.gdembell
● Ajout d'une pièce justificative	05/06/2023 23:41	sfr.gdembell
● Modification de la demande de remboursement	05/06/2023 23:41	sfr.gdembell
● Invalidation d'une pièce justificative	05/06/2023 23:49	sfr.gdembell
● Ajout d'un commentaire	05/06/2023 23:54	sfr.gdembell

Figure 5.20: Historique des actions

Dans la page de détail, vous avez la possibilité de consulter les commentaires associés à la demande de remboursement. Chaque commentaire affiche le nom de l'utilisateur qui l'a créé, ainsi que la date et l'heure de sa création.

Commentaires		
Message	Date création	Créé par
Commentaire	05/06/2023 23:54	sfr.gdembell

Figure 5.21: Liste des commentaires

5.5 Conclusion

Ce chapitre avait pour but l'implémentation de l'application tout en décrivant le prototype de sa réalisation et l'environnement de travail ainsi qu'une représentation de quelques interfaces que nous avons jugé les plus importantes et en expliquant brièvement comment le projet a été planifié.

Conclusion et perspectives

Ce rapport de projet présente en détail toutes les étapes et phases de la réalisation du projet intitulé "Conception et réalisation d'une application de gestion de remboursement des avoirs complexes". L'objectif de ce projet était d'automatiser et de simplifier le processus de remboursement des avoirs complexes, en fournissant une application web conviviale et fonctionnelle.

Tout au long de la préparation de ce projet, j'ai cherché à mettre en pratique les connaissances acquises durant mes études dans le but de développer une application performante et accessible à différents utilisateurs, tout en assurant une gestion efficace du remboursement des avoirs. L'aboutissement de ce travail s'est révélé extrêmement enrichissant, car il m'a permis de développer mes compétences en programmation. De plus, ce projet a été d'une importance considérable, car il m'a permis de suivre une méthodologie de travail bien étudiée.

D'ailleurs, la réalisation de ce projet s'est déroulée en suivant les différentes étapes du cycle de vie d'un projet informatique, notamment l'étude du contexte, des besoins fonctionnels et non fonctionnels, la conception et la réalisation. Pour cette dernière étape, j'ai utilisé des outils intéressants tels que NgRx, qui constitue une nouvelle acquisition dans ma formation en développement informatique, ainsi que Spring Boot, Angular, Material UI, Sass, Git et GitLab.

En perspectives, ce projet pourrait être amélioré en ajoutant des fonctionnalités avancées, telles que la possibilité de visualiser les éléments qui ont été modifiés ou ajoutés lors de chaque action d'enregistrement de demande. Cela permettrait d'avoir une meilleure visibilité sur les changements effectués et d'offrir une traçabilité plus précise des modifications dans l'application. De plus, la mise en place d'une stratégie de rétention des fichiers au niveau d'Amazon S3, limitant la conservation des fichiers à une période d'un an, permettrait d'optimiser l'utilisation de l'espace de stockage et de garantir une meilleure gestion des données de l'application.

En conclusion, ce projet a été une opportunité précieuse pour mettre en pratique mes compétences techniques et développer une application de gestion de remboursement des avoirs complexes qui a été déployée avec succès en production. Il a non seulement renforcé ma capacité à travailler de manière autonome et efficace, mais il a également ouvert de nouvelles perspectives d'amélioration et d'expansion. J'ai pu relever les défis inhérents à la conception et à la réalisation d'un projet informatique d'envergure, et je suis extrêmement fier du travail accompli. La concrétisation de ce projet, maintenant utilisé réellement, témoigne de son utilité et de son impact dans le domaine de la gestion des remboursements des avoirs.

Bibliographie

- [1]: Angular Documentation, <https://angular.io/docs> , consulté le 12/04/2023
- [2]: Spring Boot Documentation, <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> , consulté le 25/04/2023
- [3]: NgRx Documentation, <https://ngrx.io/docs> , consulté le 17/04/2023
- [4]: GitLab Documentation, <https://docs.gitlab.com/> , consulté le 11/04/2023
- [5]: Exemple cryptage Amazon s3, <https://codetinkering.com/aws-s3-encryption-java-api/> , consulté le 03/05/2023
- [6]: Angular Material Documentation, <https://material.angular.io/>, consulté le 20/04/2023
- [7]: Module de gestion d'affichage des fichiers PDF, <https://pdfviewer.net/extended-pdf-viewer/simple> , consulté le 10/05/2023

Le présent rapport est une synthèse du travail effectuée dans le cadre de mon stage de fin d'études au sein d'Intelcia It Solutions. L'objectif de ce projet est la mise en place d'une application web pour la gestion de remboursement d'avoir complexes.



Mon rôle réside dans l'analyse, la conception et la réalisation du projet. En commençant par l'étude de l'existant, la détection des besoins fonctionnelles et non fonctionnelles, la détermination de la méthodologie du projet, et la conception pour passer à l'étape de la réalisation de l'application.

On a adopté le processus SCRUM pour réaliser ce travail qui se compose de quatre étapes nécessaires :

La première étape reflète la phase d'étude fonctionnelle du projet. Elle comprend l'analyse du cahier de charge, l'identification des acteurs qui interagissent avec le système, la rédaction des spécifications fonctionnelles et non fonctionnelles.

La deuxième étape est l'étude technique. Elle décrit les contraintes logicielles et les technologies adoptées pour réaliser ce projet.

La troisième étape symbolise le fruit de la conception en représentant la solution aboutie en se basant sur le langage UML.

La quatrième consistait à implémenter, tester et déployer l'application.

Mots clé: SCRUM, JWT, UML, Angular, Spring Boot, Amazon S3, NgRx

Ibrahim Chahboune

Ingénierie Informatique et Technologies Emergentes

Option : Ingénierie des données massives (Big Data)