

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A/Mira de Béjaïa
Faculté des Sciences Exactes
Département d'Informatique

Mémoire de fin de cycle

En vue de l'obtention du diplôme de Master Professionnel en Informatique

Option : Génie Logiciel

Thème

Conception et réalisation d'une application web de calcul des charges des distributeurs Cas d'études : Cevital- Béjaïa

Réalisé par :

M^{lle} DJERROUD Sabrina

M^{lle} HAMITOUCHE Lynda

Soutenu le 01 juillet 2025 devant le jury composé de :

Encadrant : M^r ZERARGA Loutfi

Président : M^r BEDJOU Khaled

Examinateur : M^{me} GASMI Badrina

Examinateur : M^{me} KHOULALENE Nadjjet

Examinateur : M^r SIDER Abderrahmane

Année Universitaire : 2024/2025

Dédicace

****Au nom de Dieu, le Clément, le Miséricordieux.****

Avec tous mes sentiments de respect et la reconnaissance que je vous dois, je dédie ce travail à mon **paradis**, à la *prunelle de mes yeux*, à la *source de ma joie et de mon bonheur*, à celle qui illumine mon chemin, ma **MAMAN**.

À celui qui m'a permis de devenir la personne que je suis aujourd'hui, ma *source de vie, d'amour et de soutien*, **PAPA**, qui a toujours été à mes côtés pour me guider et m'encourager.

À mes frères et sœurs : **Katia**, **Manel**, et **Amine**, pour leur amour, leurs rires et leurs encouragements. Vous avez été une véritable source de force et de motivation. Vous avez toujours été là à chaque étape. Merci d'avoir cru en moi et de m'avoir donné la force de continuer.

À **Amine**, pour sa bienveillance, son soutien inconditionnel dans les moments les plus difficiles. Merci d'avoir été là, de m'avoir écoutée et de m'avoir encouragée. Ta présence m'a donné la force d'avancer. Je ne pourrai jamais assez te remercier pour tout ce que tu as fait pour moi.

Merci à vous tous pour votre amour, votre soutien, vos prières et vos encouragements. Sans vous, ce parcours n'aurait pas eu la même saveur. C'est grâce à vous que je suis parvenue à ce moment, et je n'aurai jamais assez de mots pour exprimer ma gratitude.

Lynda.

Dédicace

Au nom de Dieu, le Clément, le Miséricordieux.

À celle qui incarne l'amour pur et le dévouement sans limites, à celle dont les prières m'ont protégée et les sacrifices ont pavé mon chemin, ma précieuse **MAMAN**, ma lumière et mon refuge. Je te dois ce parcours et bien plus encore.

À mon **PAPA**, source de ma stabilité, modèle de courage et de sagesse, dont les paroles ont éclairé ma route et dont la patience m'a portée dans les moments d'épreuve. Merci pour ton amour constant et ton soutien indéfectible.

À mes frères bien-aimés : *Zahir, Mohand Zine, Dahmane, Ahmed, Fouad, Halim* et *Messipsa*, pour votre générosité, vos paroles rassurantes et votre soutien constant. Vous m'avez toujours rappelé que je n'étais jamais seule. Votre présence et vos encouragements m'ont portée tout au long de ce parcours.

À mes sœurs chéries : *Kahina, Nadira, Wardia* et *Nadia*, pour votre écoute, vos conseils précieux et votre sagesse partagée à chaque étape de ma vie. Votre amour, votre présence apaisante et vos encouragements m'ont toujours guidée et soutenue. Que Dieu vous protège.

À mes précieuses amies : *Koko, Tinhinane* et ma chère binôme *Lynda*, merci pour votre complicité, votre soutien et votre bienveillance tout au long de ce parcours.

Ce projet est le fruit de vos sacrifices, de vos conseils et de votre foi en moi. Qu'Allah vous préserve et vous comble de Ses bienfaits. Sans vous, ce travail n'aurait pu voir le jour.

Sabrina.

Remerciements

Remerciements

Tout d'abord, nous remercions Allah, le Tout-Puissant, de nous avoir donné la force, le courage et la patience nécessaires pour mener ce travail à son terme. Sans Lui, rien de tout cela n'aurait été possible.

Ce jour marque l'aboutissement d'un long parcours académique à l'Université Abderrahmane Mira de Béjaïa « Targa Ouzemmour », qui a été pour nous un lieu de savoir, de rencontres et de défis enrichissants.

*Nous exprimons nos sincères remerciements à notre encadrant, **Dr. ZERARGA Loufti**, pour nous avoir honorées par son encadrement, sa disponibilité et ses conseils précieux.*

Nous remercions également les membres du jury pour avoir accepté d'évaluer notre mémoire, ainsi que pour la pertinence de leurs remarques et suggestions.

*Nous exprimons aussi notre gratitude à M. Rachid Rahmani, directeur du service commercial, ainsi qu'à M. Aït Ikhlef de la direction des systèmes d'information de l'entreprise **CEVITAL Agro-Industrie**, pour leur accueil chaleureux et leur aide précieuse.*

Notre reconnaissance va également à nos familles pour leur soutien moral et financier, et pour nous avoir encouragées et accompagnées tout au long de notre parcours universitaire.

Enfin, nous remercions chaleureusement toutes les personnes qui, de près ou de loin, ont contribué à l'élaboration de ce mémoire.

Lynda & Sabrina

Table des matières

Table des figures	7
Liste des tableaux	9
Liste des abréviations	10
Introduction Générale	11
1 Contexte d'étude et recueil des besoins	13
Introduction	13
1.1 Présentation du Groupe Cevital	13
1.2 Historique du Cevital	13
1.3 Missions et Objectifs de Cevital	14
1.4 Valeurs du groupe Cevital	15
1.5 Organigramme du Groupe Cevital	15
1.5.1 Direction des systèmes d'informations	16
1.5.2 Direction commerciale	17
1.6 Etude de l'existant	20
1.7 Problématique	20
1.8 Solutions proposées	21
Conclusion	21
2 Méthodologie de conception et spécification des besoins	23
Introduction	23
2.1 Méthodologie de conception	23
2.1.1 Méthodes agiles	23
2.1.2 Méthode Scrum	24
2.1.3 Langage de modélisation UML	26
2.1.4 Le design pattern Modèle-Vue-Contrôleur (MVC)	27
2.2 Pilotage du projet avec Scrum	28
2.2.1 Sprint 0	28
Conclusion	36

3	Release 1	37
	Introduction	37
	3.1 Sprint 1 : Espace Administrateur et Chargé du Reporting	37
	3.1.1 Diagramme du cas d'utilisation du sprint 1	38
	3.1.2 Cas d'utilisation "S'authentifier"	39
	3.1.3 Cas d'utilisation "Ajouter un distributeur"	40
	3.1.4 Diagramme de classe du Sprint 1	42
	3.1.5 Interface du sprint 1	43
	3.2 Sprint 2 : Espace du Chargé du Reporting	45
	3.2.1 Diagramme du cas d'utilisation du deuxième sprint	46
	3.2.2 Cas d'utilisation "Modifier le salaire d'un Front-Office"	47
	3.2.3 Diagramme de classe du Sprint 2	49
	3.2.4 Interface du sprint 2	50
	Conclusion	51
4	Release 2	52
	Introduction	52
	4.1 Sprint 3 : Espace du Chargé du Reporting	52
	4.1.1 Diagramme du cas d'utilisation du sprint 3	53
	4.1.2 Cas d'utilisation "Gestion des réalisations"	54
	4.1.3 Cas d'utilisation "Consulter la liste des jours ouvrables pour chaque mois"	55
	4.1.4 Diagramme de classe du Sprint 3	57
	4.2 Sprint 4 : Espace du Superviseur et Validation	59
	4.2.1 Diagramme du cas d'utilisation du sprint 4	59
	4.2.2 Cas d'utilisation "Approuver / Rejeter une présence"	60
	4.2.3 Diagramme de classe du Sprint 4	63
	4.2.4 Diagramme d'activité pour le cas d'utilisation " Gestion du circuit de validation des présences"	64
	Conclusion	67
5	Release 3	68
	Introduction	68
	5.1 Sprint 5 : Fonction de calcul des charges	68
	5.1.1 Diagramme du cas d'utilisation du cinquième sprint	68
	5.1.2 Cas d'utilisation "Gestion des calculs des charges du Front-office et du Back-office"	69
	5.1.3 Cas d'utilisation "Modifier le montant des charges"	71
	5.1.4 Diagramme de classe du Sprint 5	73
	5.2 Une démonstration de la responsivité ¹	76
	5.2.1 Téléphone	77
	5.2.2 Tablette	77

5.3	Conclusion	77
6	Conception de la base de données	79
	Introduction	79
6.1	Règles de gestion	79
6.2	Diagramme de classe	80
6.3	Le modèle relationnel	81
6.4	Dictionnaire des données	82
6.5	Conclusion	91
	Conclusion Générale et Perspectives	92
	Bibliographie	94
A	Technologies, outils et protocoles utilisés	96
A.1	Frontend – Développement des interfaces utilisateur	96
	A.1.1 React JS (Web)	96
	A.1.2 React Native (Mobile)	97
	A.1.3 CSS3	97
	A.1.4 JavaScript	97
A.2	Backend – Serveur, API, logique métier	97
	A.2.1 Node.js et Express.js	97
A.3	Base de données – Stockage structuré	98
	A.3.1 MySQL	98
	A.3.2 Sequelize (ORM)	98
A.4	Authentification et sécurité	98
	A.4.1 JSON Web Tokens (JWT)	98
	A.4.2 Rôles et permissions	98
	A.4.3 HTTPS	98
A.5	Outils de développement	99

Table des figures

1.5.1 Organigramme de Cevital [3].	15
1.5.2 Organigramme de la DSI.	16
1.5.3 Organigramme du service commercial.	17
1.5.4 Hiérarchie du fonctionnement de la direction commerciale.	18
2.1.1 Méthode Scrum [5].	25
2.1.2 Les diagrammes UML	27
2.1.3 Architecture MVC	28
2.2.1 Diagramme du cas d'utilisation global.	30
3.1.1 Diagramme du cas d'utilisation du sprint 1.	38
3.1.2 Diagramme de séquence du cas d'utilisation "S'authentifier".	40
3.1.3 Diagramme de séquence du cas d'utilisation "Ajouter un distributeur".	42
3.1.4 Diagramme de classe du sprint 1.	43
3.1.5 Interface d'Authentification	44
3.1.6 Interface Distributeurs	44
3.1.7 Interface Dépôts.	45
3.1.8 Interface Dépôts.	45
3.2.1 Diagramme du cas d'utilisation du sprint 2.	47
3.2.2 Diagramme de séquence du cas d'utilisation "Modifier le salaire d'un Front-Office".	49
3.2.3 Diagramme de classe du sprint 2.	50
3.2.4 Interface Effectif Front-Office.	51
3.2.5 Interface Charges Fixes du Back-Office.	51
4.1.1 Diagramme du cas d'utilisation du sprint 3.	53
4.1.2 Diagramme de séquence du cas d'utilisation "Rechercher une réalisation".	55
4.1.3 Diagramme de séquence du cas d'utilisation "Consulter la liste des jours ouvrables".	57
4.1.4 Diagramme de classe du sprint 3	58
4.1.5 Interface Objectifs.	58
4.1.6 Interface des jours ouvrables.	59
4.2.1 Diagramme du cas d'utilisation du sprint 4.	60
4.2.2 Diagramme de séquence du cas d'utilisation "Approuver ou rejeter une présence".	63
4.2.3 Diagramme de classe du quatrième sprint.	64

4.2.4 Diagramme d'activité du cas d'utilisation "Gestion du circuit de validation des présences".	65
4.2.5 Interface Primes.	65
4.2.6 Interface de saisie des présences -partie 1-.	66
4.2.7 Interface de saisie des présences -partie 2-.	66
4.2.8 Interface de validation des Présences.	67
5.1.1 Diagramme du cas d'utilisation du sprint 5.	69
5.1.2 Diagramme de séquence du cas d'utilisation "Calcul des charges du Front-Office".	71
5.1.3 Diagramme de séquence du cas d'utilisation "Modifier un montant des charges du Front-Office".	73
5.1.4 Diagramme du cas d'utilisation du sprint 5.	74
5.1.5 Interface de calcul des charges du Front-Office -partie 1-.	74
5.1.6 Interface de calcul des charges du Front-Office -partie 2-.	75
5.1.7 Interface de calcul des charges du Front-Office -partie 3-.	75
5.1.8 Interface de calcul des charges du Front-Office -partie 4-.	76
5.2.1 Vue de l'application sur téléphone	77
5.2.2 Vue de l'application sur tablette	77
6.2.1 Diagramme de classe global.	80

Liste des tableaux

2.1	La répartition des rôles pour la réalisation du projet.	28
2.2	Liste des User Stories.	32
2.3	Product Backlog.	34
2.4	Sprint Backlog selon la méthode Scrum, organisé par Release.	35
3.1	Tableau descriptif du cas d'utilisation "S'authentifier".	39
3.2	Tableau descriptif du cas d'utilisation "Ajouter un distributeur".	41
3.3	Tableau descriptif du cas d'utilisation "Modifier le salaire d'un Front-Office".	48
4.1	Tableau descriptif du cas d'utilisation "Rechercher une réalisation".	54
4.2	Tableau descriptif du cas d'utilisation "Consulter la liste des jours ouvrables".	56
4.3	Tableau descriptif du cas d'utilisation "Approuver ou rejeter une présence".	61
5.1	Tableau descriptif du cas d'utilisation "Calcul des charges du Front-Office".	70
5.2	Tableau descriptif du cas d'utilisation "Modifier un montant des charges Front-Office".	72
6.1	Dictionnaire des données.	91

Liste des abréviations

CNAS Caisse Nationale des Assurances Sociales.

CSS Cascading Style Sheet.

DD Distribution Directe.

DI Distribution Indirecte.

DSI Direction des Systèmes d'Information.

HTML HyperText Markup Language.

HORECA Hôtellerie, Restauration, Cafétéria.

JS JavaScript.

MVC Model-View-Controller, ou Modèle-Vue-Contrôleur.

MySQL My Structured Query Language.

PC Personal Computer (Ordinateur personnel).

SCRUM S : Self-organization (Auto-organisation), C : Collaboration, R : Respect, U : Unity (Unité d'équipe), M : Motivation.

SPA Société par Actions.

UML Unified Modeling Language (Langage de Modélisation Unifié).

VSCo Visual Studio Code.

WampServer W : Windows, A : Apache, M : MySQL, P : PHP, Server.

CO₂ Dioxyde de Carbone.

Introduction Générale

Dans un contexte économique où la performance opérationnelle et la rigueur de gestion sont des facteurs déterminants de compétitivité, la digitalisation s'impose comme un levier incontournable pour optimiser les processus internes des entreprises. Elle transforme en profondeur les méthodes de travail en offrant des outils innovants permettant une centralisation des données et une prise de décision plus rapide et plus efficace. C'est dans cette dynamique que s'inscrit Cevital, leader de l'agro-industrie en Algérie, qui fait face à divers défis notamment au sein de son service commercial, en ce qui concerne le calcul des charges des distributeurs, un enjeu crucial pour assurer un suivi rigoureux et une répartition optimale des coûts logistiques.

Bien que l'entreprise utilise actuellement une application desktop pour le calcul des charges de ses distributeurs, cette dernière présente des limites importantes, notamment un mode d'accès restreint au poste local et une absence de mobilité. Ces contraintes rendent l'application inaccessible en dehors du réseau interne et limitent son usage sur d'autres supports, ce qui affecte l'efficacité des opérations. D'où la nécessité de concevoir une solution web plus performante, et adaptée aux besoins de mobilité, tout en assurant la centralisation et la sécurisation des données.

Ce projet a pour objectif de concevoir et de mettre en œuvre une application web destinée à optimiser le calcul des charges des distributeurs au sein du service commercial de Cevital. Il s'agit de proposer un outil accessible, fiable et mieux intégré à l'environnement de travail, afin de faciliter le suivi des opérations et d'améliorer la qualité des données exploitées.

Ce projet est structuré en six chapitres. Tout d'abord, nous commencerons par le premier chapitre qui est un chapitre introductif dans lequel nous expliquons et décrivons le contexte et les objectifs de notre travail. Nous y présentons tout d'abord l'organisme d'accueil, la SPA CEVITAL de Béjaïa, puis nous décrivons la direction des systèmes d'information ainsi que la direction commerciale, en incluant leurs organigrammes respectifs. Nous poursuivons avec une étude de l'existant. Enfin, nous exposons la problématique et proposons des solutions envisageables pour y répondre.

Le deuxième chapitre est consacré aux approches méthodologiques et techniques adoptées dans notre projet. Nous mettons en avant la méthode Scrum et ses éléments clés, puis nous

détaillons la modélisation UML des besoins et l'architecture logicielle choisie, notamment l'approche MVC. Nous illustrons également l'utilisation concrète de Scrum dans notre projet à travers l'identification des acteurs, le diagramme de cas d'utilisation global, la rédaction des user stories, l'élaboration du Product Backlog, ainsi que la planification des releases.

Le troisième chapitre présente le travail réalisé lors du premier release, comprenant les sprints 1 et 2. Pour chacun, nous fournissons le diagramme de cas d'utilisation accompagné des descriptions textuelles, les diagrammes de séquence, un diagramme de classes et les captures d'écran illustrant les interfaces développées.

Le quatrième chapitre est dédié au deuxième release, qui couvre les sprints 3 et 4. Nous y détaillons les fonctionnalités implémentées dans ces sprints en suivant la même structure : diagramme de cas d'utilisation, descriptions textuelles, diagrammes de séquence, diagramme de classes et des captures d'écran.

Le cinquième chapitre présente le troisième release, correspondant au sprint 5. Ce dernier sprint est axé principalement sur la finalisation des fonctionnalités, notamment celles relatives au calcul des charges. Comme pour les phases précédentes, nous y exposons les éléments de modélisation et les interfaces développées.

Le sixième et dernier chapitre conclut la modélisation des données de notre application en présentant les éléments clés structurant le système. Il détaille les règles de gestion, le diagramme de classes global, le modèle relationnel, ainsi que le dictionnaire des données.

Enfin, nous terminerons ce projet par une conclusion générale accompagnée de perspectives que nous souhaitons concrétiser dans le futur.

Chapitre 1

Contexte d'étude et recueil des besoins

Introduction

Ce chapitre introductif vise à expliquer et à décrire le contexte et les objectifs de notre projet. En présentant dans un premier lieu l'organisme d'accueil (SPA CEVITAL de Béjaia). En deuxième lieu, nous décrivons la direction des systèmes d'informations, et la direction commerciale, en incluant leurs organigrammes respectifs. Par la suite, nous procédons à une étude de l'existant afin d'analyser l'outil actuellement utilisé et d'identifier ses principales limites. Enfin, nous présentons la problématique ainsi que les solutions envisagées pour y répondre.

1.1 Présentation du Groupe Cevital

Le Groupe Cevital est un conglomérat algérien actif dans plusieurs secteurs, notamment l'industrie agroalimentaire, la grande distribution, l'industrie métallurgique, l'électronique et l'électroménager, la sidérurgie, l'industrie du verre plat, la construction industrielle, l'automobile, les services, les médias. . . Fondé en 1998 à Béjaïa par Issad Rebrab, il est également présent à l'international. Cevital est la troisième entreprise algérienne en termes de chiffre d'affaires, ce qui lui permet d'être le leader de l'agroalimentaire en Afrique. Le groupe renferme 26 filiales et emploie 18 000 personnes réparties sur trois continents [1].

Cevital Agro-industrie est composée de plusieurs unités de production telles que : une unité de conditionnement d'huile, une raffinerie d'huile, une unité de conditionnement de sucre, une raffinerie de sucre, une margarinerie, une unité de plastique, et trois unités externe telles que : une unité de conditionnement d'eau minérale qui se situe à Tizi Ouzou (Lala Khedidja), une unité de fabrication et de conditionnement de boisson rafraîchissante et de sauces (site EL-KSEUR) et une unité de four à chaux et CO₂ située à El Khroub (Constantine).

1.2 Historique du Cevital

Cevital est un Groupe familial qui s'est bâti sur une histoire, un parcours et des valeurs qui ont fait sa réussite et sa renommée. Première entreprise privée algérienne à avoir investi

dans des secteurs d'activités diversifiés, elle a traversé d'importantes étapes historiques pour atteindre sa taille et sa notoriété actuelle. Ces étapes sont les suivantes :

- 1971 : Lancement de la construction métallique.
- 1988 : Création de METAL SIDER (Sidérurgie).
- 1991 : Reprise des activités I.B.M en Algérie / Création du quotidien liberté.
- 1997 : Création de Hyundai Motors Algérie.
- 1998 : Création de Cevital SPA industries agroalimentaires.
- 2006 : Création de Numidis et Immobis ; Acquisition de COJEK.
- 2007 : SAMHA –Production & distribution SAMSUNG / Création MFG (VERRE PLAT).
- 2008 : Nolis –Transport maritime / commercialisation du verre plat en Europe / Création de NUMILOG.
- 2009 : Augmentation de la production du sucre de 1 M T/AN.
- 2013 : OXXO(France) / ALAS (Espagne).
- 2014 : BRANDT (France) / AFFERPI (Italie) ex LUCCHINI PIOMBINO[2].

1.3 Missions et Objectifs de Cevital

Cevital est un conglomérat industriel qui s'est donné pour mission de fournir des produits et des services de qualité afin de répondre aux besoins des consommateurs en Algérie et à l'international. En tant que première entreprise privée algérienne investissant dans divers secteurs, elle vise à contribuer au développement économique en créant des opportunités d'emploi et en soutenant la production locale. Cevital cherche également à promouvoir l'innovation et la modernisation industrielle en intégrant des technologies avancées. Son ambition est de renforcer sa position sur les marchés internationaux grâce à l'exportation et à des partenariats stratégiques. Afin de concrétiser cette mission, Cevital s'est fixé plusieurs objectifs :

- Étendre la distribution de ses produits sur l'ensemble du territoire national et à l'international.
- Moderniser ses installations industrielles en adoptant de nouvelles technologies pour accroître la production.
- Encourager la production locale des graines oléagineuses en apportant un soutien financier aux agriculteurs.
- Optimiser les offres d'emploi en créant de nouvelles opportunités sur le marché du travail.
- Renforcer sa présence sur les marchés étrangers en développant ses exportations et en positionnant ses produits à l'international.

1.4 Valeurs du groupe Cevital

Le groupe Cevital fonde son succès sur quatre valeurs essentielles, qui forment le pilier de sa culture d'entreprise et sont regroupées sous l'acronyme I.R.I.S

- **Intégrité** : Il s'agit d'adopter une éthique professionnelle irréprochable, de rejeter toute forme de corruption et d'agir avec une haute rigueur intellectuelle et morale.
- **Respect** : C'est le fait d'accorder et témoigner régulièrement et attentivement à l'ensemble des collaborateurs, des acteurs de la vie économique et sociale et de l'environnement interne et externe.
- **Initiative** : Il s'agit d'anticiper les problèmes potentiels et d'imaginer et proposer des solutions innovantes.
- **Solidarité** : Il s'agit de s'entraider mutuellement, de se rendre disponible pour l'entreprise et les membres de l'équipe ainsi de partager spontanément notre savoir et nos expériences.

1.5 Organigramme du Groupe Cevital

Afin de garantir une gestion efficace et une coordination optimale entre ses différentes branches, Cevital est structuré en plusieurs directions, chacune ayant un rôle bien défini. La figure 1.5.1 illustre l'organigramme global de l'entreprise qui regroupe les principales directions.

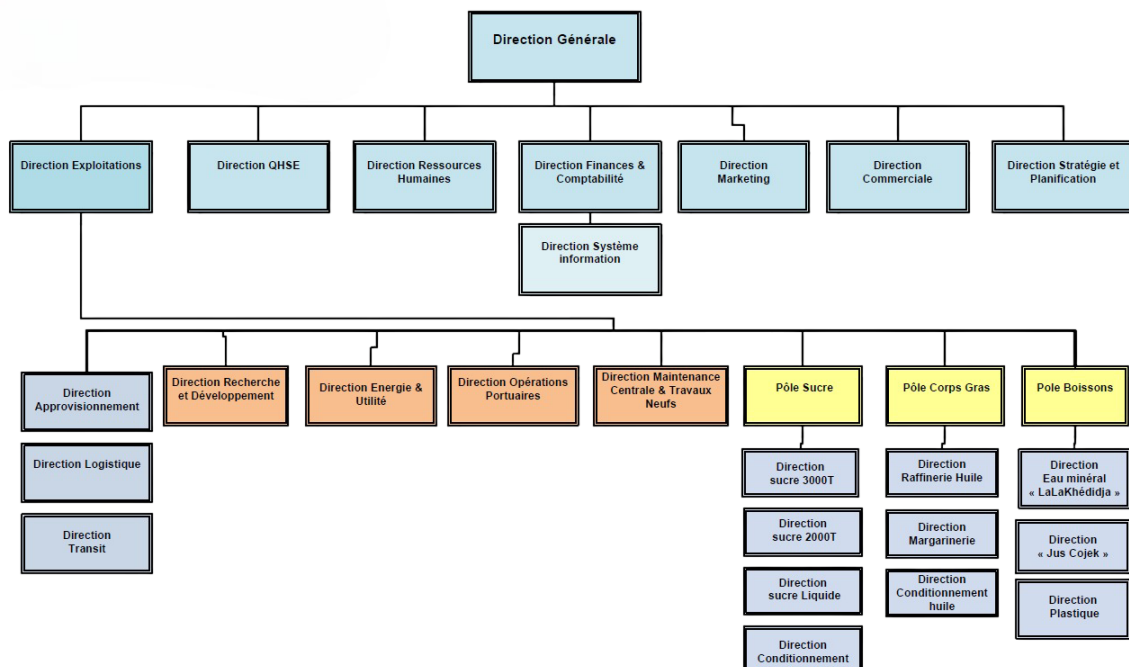


FIGURE 1.5.1 – Organigramme de Cevital [3].

Parmi ces directions, notre travail s'inscrit dans le cadre de la direction des systèmes d'informations, bien que la solution que nous développons soit destinée à la direction commerciale.

1.5.1 Direction des systèmes d'informations

La Direction des Systèmes d'Information (DSI) de Cevital Agroalimentaire Béjaïa joue un rôle essentiel dans la gestion et l'évolution des technologies de l'information au sein de l'entreprise. Elle veille à la mise en place d'un système d'information structuré, garantissant la cohérence, la sécurité et la performance des outils numériques.

Notre stage s'est déroulé au sein de cette direction, plus précisément dans le département des applications métier, chargé de :

- Développer et maintenir les solutions informatiques adaptées aux besoins de l'entreprise.
- Assurer la disponibilité et la mise à niveau des systèmes.
- Planifier leur évolution en fonction des objectifs stratégiques et des avancées technologiques.

La figure 1.5.2 illustre l'organigramme détaillé de la DSI, mettant en évidence ses principaux départements et leurs missions respectives.

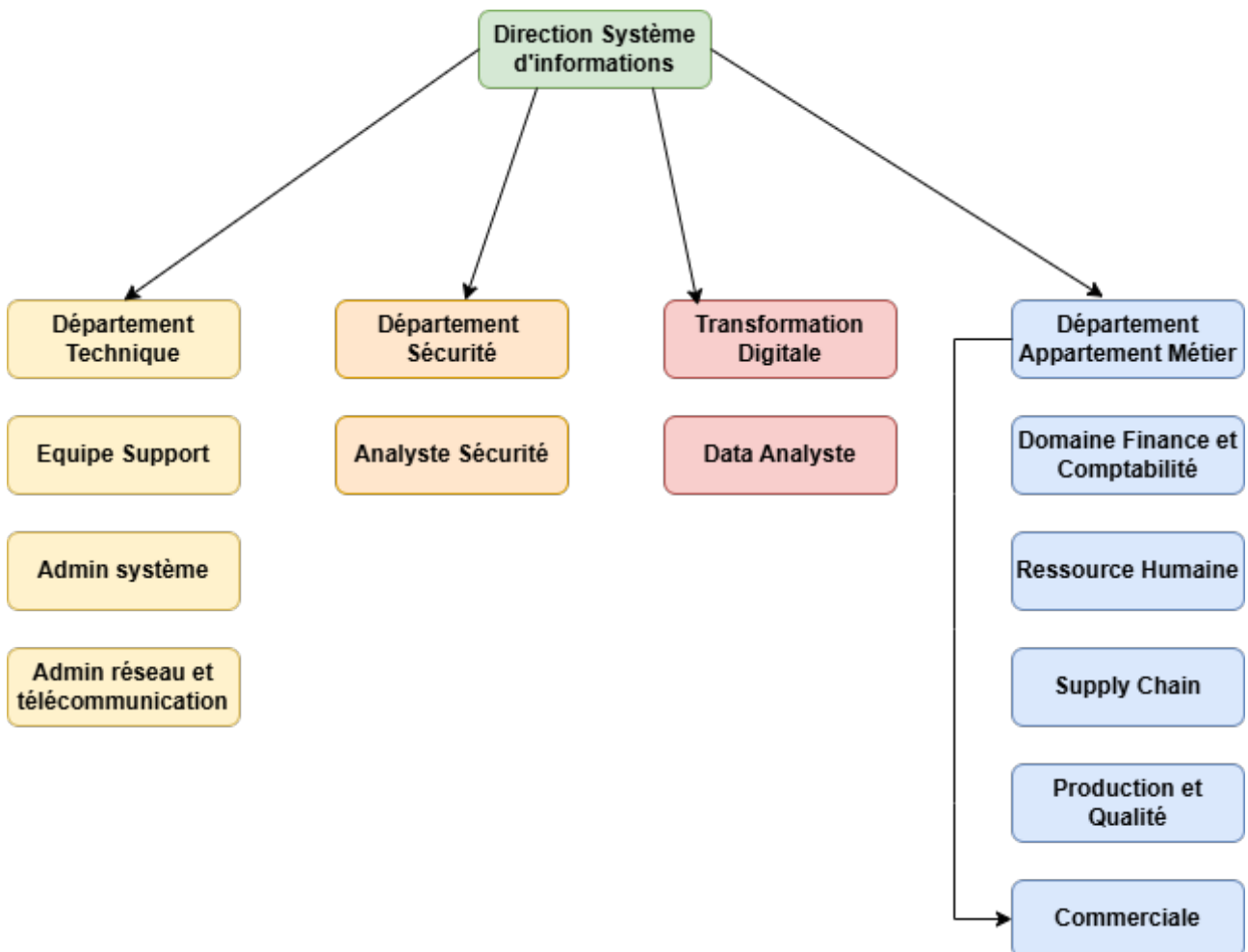


FIGURE 1.5.2 – Organigramme de la DSI.

Bien que nous travaillions sous la supervision de la DSI, la solution que nous proposons répond aux besoins spécifiques du service commerciale, ce qui nous amène à détailler son fonctionnement et les problèmes qu'il rencontre actuellement.

1.5.2 Direction commerciale

La direction commerciale de Cevital est chargée de la gestion des activités de vente et de distribution des produits du groupe à travers le territoire national et international. Afin d'assurer un suivi rigoureux et une gestion optimisée, elle repose sur une structure hiérarchique bien définie, permettant une meilleure coordination entre les différents acteurs impliqués dans la chaîne de distribution.

Cette organisation permet :

- La gestion des distributeurs et des points de vente.
- Le suivi des ventes et des performances commerciales.
- La gestion des flux de distribution et des stocks.
- L'élaboration de stratégies commerciales pour améliorer les performances.
- Assurance d'une présence optimale des produits sur le marché.

1.5.2.1 Structure organisationnelle de la direction commerciale

La figure 1.5.3 illustre le schéma organisationnel de la direction commerciale de Cevital.

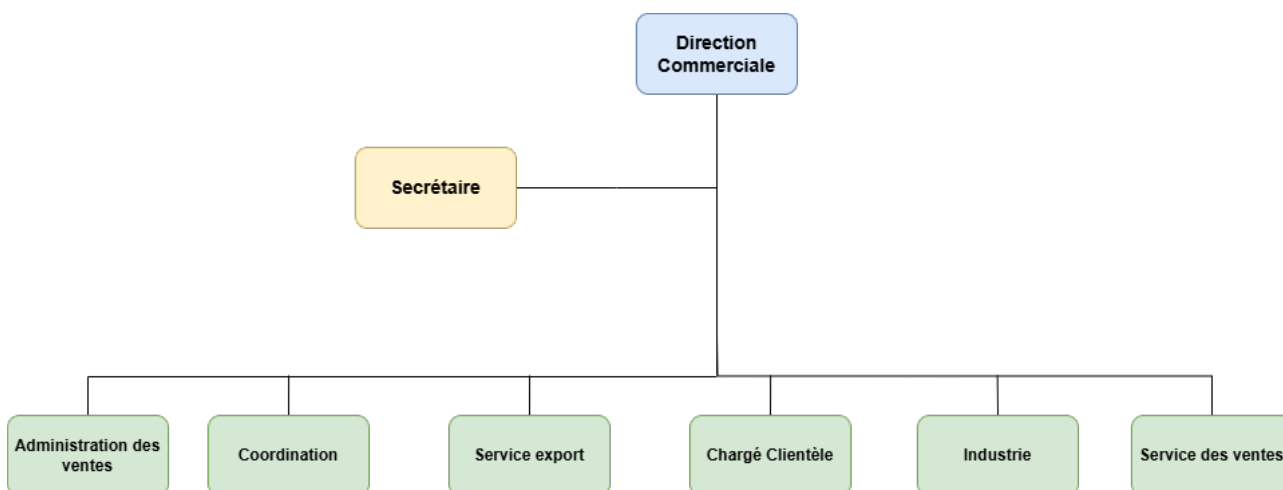


FIGURE 1.5.3 – Organigramme du service commercial.

1.5.2.2 Fonctionnement de la direction commerciale

L'organisation commerciale de Cevital est structurée de manière hiérarchique pour optimiser la gestion des distributeurs. La figure 1.5.4 illustre cette hiérarchie.

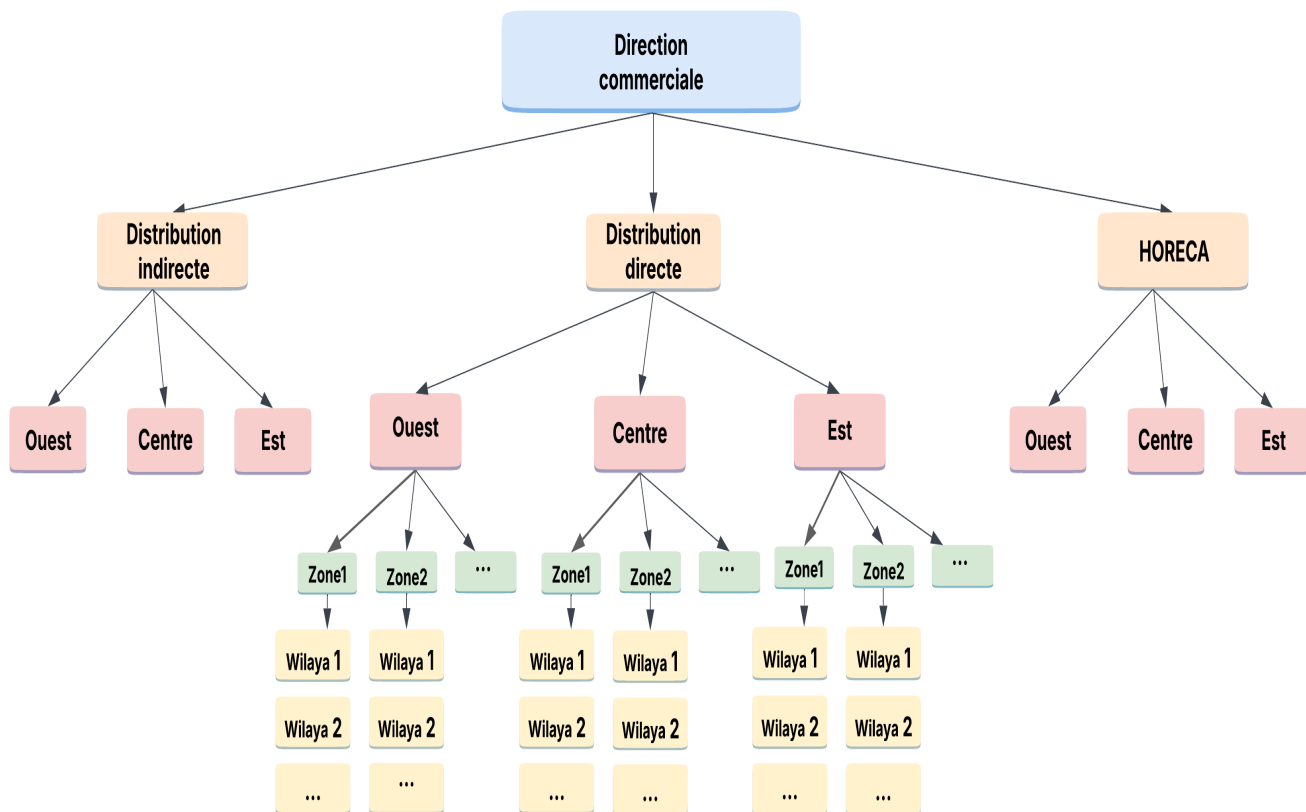


FIGURE 1.5.4 – Hiérarchie du fonctionnement de la direction commerciale.

***Remarque** : Par souci de lisibilité, seul un canal (Distribution directe) détaillé est représenté ici. Les autres canaux suivent la même structure et le même principe de conception.

Cette structure repose sur trois niveaux hiérarchiques principaux, ces niveaux sont les suivants :

- **Canal** : Un canal représente une grande division de la direction commerciale, regroupant des activités spécifiques liées à la vente et à la distribution. Chez Cevital, il y a trois canaux principaux :
 - **DI (Distribution Indirecte)** : Regroupe les ventes effectuées via des intermédiaires (grossistes).
 - **DD (Distribution Directe)** : Concerne les ventes réalisées directement aux clients sans intermédiaires (livraisons directes aux magasins, etc).
 - **HORECA (Hôtellerie, Restauration, Cafétéria)** : Gère la distribution des produits destinés aux hôtels, restaurants et cafétérias.

Chaque canal est dirigé par un national des ventes, qui supervise l'ensemble des opérations de son canal à l'échelle nationale.

- **Région** : Un canal est divisé en trois régions correspondant aux grandes zones géographiques de l'Algérie :
 - **Région Est**

- **Région Centre**
- **Région Ouest**

Chaque région est dirigée par un Chef de Région, qui coordonne les activités commerciales et veille au bon fonctionnement des zones sous sa responsabilité.

- **Zone** : Chaque région est ensuite subdivisée en zones, qui sont des territoires plus petits englobant plusieurs wilayas.

Chaque zone est dirigée par un Chef de Zone, qui gère les opérations locales et fait le lien entre les superviseurs et le Chef de Région.

Une zone regroupe plusieurs wilayas, et chaque wilaya est supervisée par un ou plusieurs superviseurs, chargés de contrôler les présences. Ce dernier occupe une position stratégique dans la chaîne opérationnelle, assurant la liaison entre les équipes terrain et les niveaux supérieurs de la hiérarchie (Chef de Zone, Chef de Région, National des ventes, ...).

Chaque wilaya dispose d'un ou plusieurs dépôts, pouvant relever d'un, deux ou trois canaux de distribution (DI, DD, HORECA), en fonction des spécificités de la zone. Ces dépôts sont divisés en deux unités fonctionnelles :

- **Back-Office** est dédié aux fonctions administratives, à la comptabilité ainsi qu'au suivi des activités internes du dépôt. Il englobe notamment :
 - **Les effectifs** : Magasinier, caissier, administrateur des ventes, gérant, etc.
 - **Charges sociales** : Cotisations CNAS, assurances, etc.
 - **Dépenses fixes** : Loyer, amortissement des chariots, etc.
- **Front-Office** est chargé des opérations sur le terrain. Il s'occupe de toutes les activités en dehors du dépôt, comme la préparation, le contrôle et la livraison des commandes, ainsi que toute autre tâche liée à l'exécution sur le terrain, il regroupe :
 - **Les effectifs opérationnels** : vendeurs, prévendeurs, chauffeurs (légers et lourds), livreurs, etc.
 - **Les flottes de distribution** : camions, véhicules utilitaires, scooters, etc.

De plus, le **superviseur** qui a pour missions principales :

- Les présences quotidiennes des ressources (personnelles et matérielles) pour chaque canal d'un dépôt.
- Le suivi en temps réel des opérations.

À la fin de chaque mois, les données recueillies sont analysées afin d'estimer les dépenses supportées par les distributeurs. Cette analyse permet de calculer les montants à rembourser et d'apporter les ajustements nécessaires en fonction des frais engagés et des opérations réalisées.

Pour assurer le suivi quotidien des opérations, tous les acteurs, notamment les superviseurs, utilisent une application desktop développée en interne par le département Système d'Information de Cevital.

1.6 Etude de l'existant

Dans le cadre de notre stage, nous avons réalisé une analyse approfondie de l'application, actuellement utilisée par l'équipe commerciale. Cet outil occupe une place centrale dans le pilotage de leurs activités.

Ses principales fonctionnalités sont les suivantes :

- La gestion des distributeurs et de leurs dépôts.
- La gestion des équipes Front-Office et Back-Office, ainsi que le suivi de leurs salaires et prestations.
- L'enregistrement des présences quotidiennes du personnel et des flottes de distribution.
- Le suivi du chiffre d'affaires, des objectifs fixés et des réalisations mensuelles des distributeurs, ainsi que le nombre de jours ouvrables pour chaque mois.
- Processus hiérarchique de validation des présences.
- La consolidation des données en vue du calcul mensuel des charges et des remboursements.

Remarque : En raison du caractère confidentiel des données traitées (informations personnelles, financières et logistiques sensibles), aucune capture d'écran de cette application interne ne peut être présentée dans ce rapport.

Bien que cette application réponde aux besoins opérationnels de base, notre analyse a mis en évidence plusieurs limites, tant sur le plan technique qu'organisationnel, à savoir l'utilisation exclusive sur desktop, l'absence d'automatisation pour certaines tâches, ainsi que la dépendance à des traitements manuels, entraînent une perte de temps, un risque d'erreurs et un manque de réactivité dans le suivi des activités.

C'est dans ce contexte que s'inscrit notre mission de stage.

1.7 Problématique

Durant notre stage au sein de l'entreprise Cevital, nous avons eu l'opportunité de contribuer au calcul des charges des distributeurs au sein du service commercial. Cependant, le système existant, qui repose sur une application desktop et des processus manuels, présente plusieurs lacunes et limitations qui entravent l'efficacité des opérations et la prise de décision.

Parmi les problèmes rencontrés, nous pouvons citer :

- **Manque de mobilité :** Les superviseurs sont contraints d'utiliser un PC portable lors de leurs déplacements sur le terrain, ce qui est peu pratique et complique la saisie des informations ainsi que le suivi des présences des employés et des flottes.
- **Problèmes de performance :** L'application souffre de lenteurs importantes, aussi bien lors du chargement de l'interface que lors du traitement des données, ce qui impacte négativement l'efficacité des opérations quotidiennes.

- **Manque d'ergonomie** : L'application souffre d'une ergonomie insuffisante, entraînant une navigation peu fluide et peu intuitive. L'accès aux informations essentielles nécessite plusieurs manipulations manuelles, ce qui alourdit les tâches quotidiennes des utilisateurs, ralentit leur activité et réduit leur efficacité.
- **Absence de flexibilité dans la gestion des données** : L'application ne permet pas la modification de certaines données essentielles, notamment l'ajustement des montants des charges, limitant ainsi la flexibilité et l'efficacité du traitement des informations.

Face à ces constats, une question centrale se pose :

Comment concevoir une solution numérique et moderne, capable d'offrir une interface plus ergonomique et un accès fluide depuis tout type d'appareil (ordinateur, tablette, smartphone), afin de répondre efficacement aux besoins du service commercial de Cevital ?

1.8 Solutions proposées

Dans le cadre de notre mission, nous avons proposé une solution visant à remédier aux différentes problématiques identifiées, notamment en automatisant et optimisant le calcul des charges des distributeurs. Notre solution repose sur le développement d'un système logiciel sous forme d'une application web, qui vise les objectifs suivants :

- **Offrir une accessibilité multiplateforme** : Permettre aux acteurs, notamment les superviseurs d'accéder au système depuis un mobile ou un ordinateur, éliminant ainsi la contrainte d'un usage exclusif sur PC.
- **Optimiser les performances du système** : Assurer une navigation fluide et réactive avec un chargement rapide des interfaces et des interactions instantanées.
- **Améliorer l'ergonomie et l'intuitivité** : Concevoir une interface moderne et facile à utiliser, réduisant le nombre d'actions nécessaires pour accéder aux informations essentielles.
- **Renforcer la gestion des données** : Ajouter des fonctionnalités permettant la modification et la mise à jour des montants des charges, offrant ainsi plus de flexibilité aux utilisateurs.
- **Garantir la sécurité des informations** : Mettre en place des protocoles d'authentification et un accès sécurisé pour protéger les données.

Conclusion

Dans ce premier chapitre, nous avons présenté le groupe Cevital, son historique, ses missions, ses objectifs, ses valeurs et son organigramme général, ainsi que ceux de la Direction des Systèmes d'Information et de la Direction Commerciale. Nous avons également détaillé le fonctionnement de la Direction Commerciale, en mettant en évidence sa structure hiérarchique.

Enfin, nous avons exploré l'existant, identifié la problématique liée au calcul des charges des distributeurs et proposé des solutions pour y remédier.

Chapitre 2

Méthodologie de conception et spécification des besoins

Introduction

Ce chapitre présente les approches méthodologiques et techniques adoptées, en mettant l'accent sur la méthode Scrum et ses éléments clés, ainsi que le formalisme UML utilisé pour modéliser les besoins, et l'architecture MVC adoptée pour structurer le code.

Nous illustrons l'utilisation de la méthode Scrum dans la gestion de notre projet à travers plusieurs éléments essentiels : la répartition des rôles pour la réalisation du projet, l'identification des acteurs, la modélisation d'un diagramme du cas d'utilisation global, ainsi que la rédaction des User Stories.

Nous abordons ensuite la constitution du Product Backlog, suivie de la planification des releases sous forme de sprints successifs.

2.1 Méthodologie de conception

Dans le cadre de la conception de notre projet, nous avons appliqué les méthodes de conception agiles et avons utilisé le Langage de Modélisation Unifié (UML) pour la modélisation.

2.1.1 Méthodes agiles

La méthode agile est une approche du développement logiciel dont l'objectif est de distribuer en continu des logiciels opérationnels créés sur la base d'itérations rapides.

L'expression « méthode agile » est une façon d'envisager la collaboration et les workflows, avec un ensemble de valeurs destiné à orienter nos choix concernant ce que nous faisons et comment nous le faisons. Concrètement, les méthodes de développement de logiciels agile sont axées sur la distribution rapide de petites parties de logiciels opérationnels pour améliorer la satisfaction client. Elles se basent sur une approche qui encourage l'adaptation et le travail d'équipe pour favoriser l'amélioration continue[4]. Ces méthodes sont beaucoup plus avantageuses par rapport aux autres méthodes de conceptions traditionnelles, et offrant plus de réussite avec :

- Flexibilité.
- Communication transparente.
- Test continu.
- Client au cœur du processus.

Et pour la performance de notre système de gestion des distributeurs et calcul de leurs charges et pour satisfaire les exigences du client, nous avons utilisé la méthode agile SCRUM pour la conception et le développement de notre système, en effet le processus Scrum s'adapte parfaitement à la décomposition de notre projet.

2.1.2 Méthode Scrum

Scrum est un framework de gestion de projet agile, basé sur une approche itérative et incrémentale. Il permet aux individus, aux équipes et aux organisations de résoudre des problèmes complexes par la mise en place des solutions dites adaptatives. C'est dans cette optique que nous avons choisi cette méthodologie pour notre projet.

Scrum met l'accent sur la planification des sprints, les réunions quotidiennes (Daily Scrum) où l'équipe partage l'état d'avancement, les réussites et les obstacles rencontrés, ainsi que la rétrospective de sprint visant à améliorer continuellement le processus de développement. Après plusieurs itérations, une version (release) regroupant toutes les fonctionnalités développées et testées est livrée au client.

Ce cadre s'articule autour de trois principes fondamentaux :

- **La transparence** : tous les membres de l'équipe doivent avoir connaissance des informations relatives au produit à développer ;
- **L'inspection** : des évaluations régulières sont indispensables pour réadapter le projet si nécessaire ;
- **L'adaptation** : la mise en œuvre de nouvelles mesures est nécessaire lorsqu'une inspection démontre des écarts sur les résultats mesurés [5].

La figure 2.1.1 illustre les principaux composants de la méthode scrum.

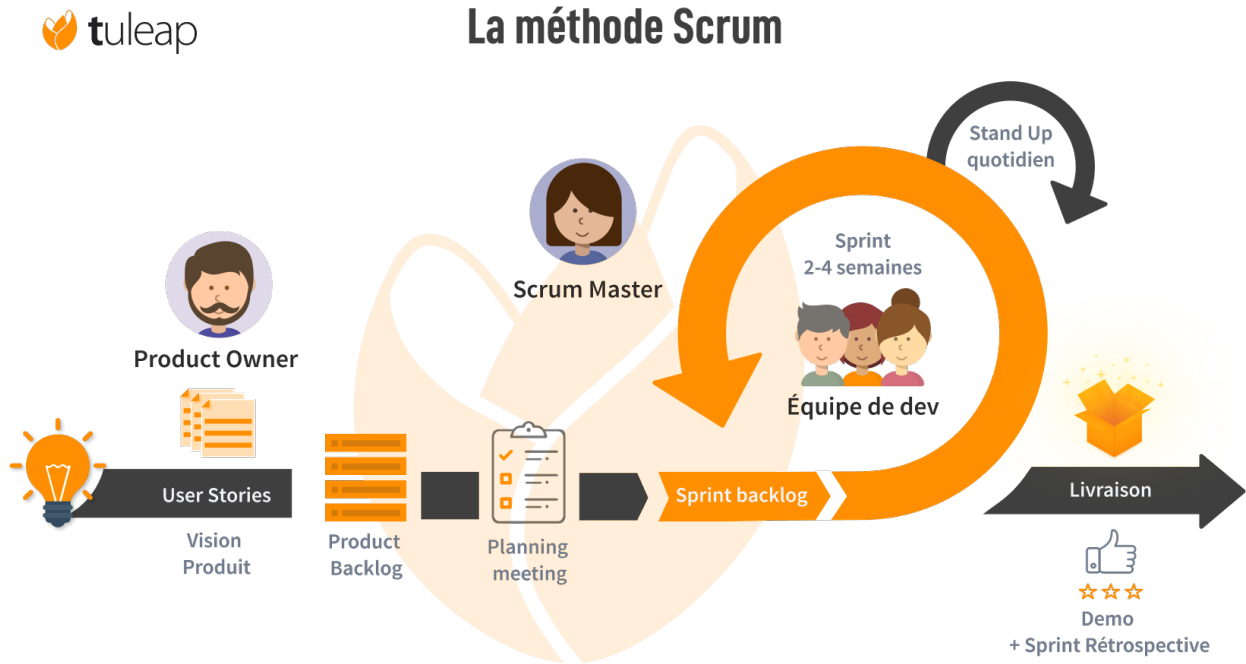


FIGURE 2.1.1 – Méthode Scrum [5].

2.1.2.1 Rôles Scrum

Scrum définit trois rôles clés :

- **Product Owner (Responsable du produit)** : est responsable de maximiser la valeur du produit développé par l'équipe Scrum. Il gère le Product Backlog, en priorisant les éléments selon la valeur métier et les besoins des parties prenantes.
- **Scrum Master (Coach de l'équipe)** : est un facilitateur qui veille à la bonne compréhension et à la bonne application de Scrum. Il aide l'équipe à surmonter les obstacles, anime les événements Scrum et protège l'équipe des interférences extérieures.
- **Équipe de développement** : les membres de l'équipe qui réalisent les tâches techniques nécessaires pour créer un incrément utilisable à chaque sprint. Ils s'auto-organisent et sont collectivement responsables de la qualité du travail livré [6].

2.1.2.2 Étapes de la méthode Scrum (Artefacts)

Scrum utilise trois artefacts principaux pour faciliter la transparence et la compréhension du travail à réaliser :

- **Product Backlog** : une liste ordonnée et évolutive de tout ce qui est nécessaire pour améliorer le produit. Il est géré par le Product Owner et reflète les besoins métier et les exigences du produit.
- **Sprint Backlog** : contient les éléments du Product Backlog sélectionnés pour un sprint, accompagnés d'un plan pour les réaliser. Il est créé et mis à jour par les développeurs tout au long du sprint.

- **Increment** : le résultat tangible et utilisable produit à la fin de chaque sprint. Il s'agit de la somme de tous les éléments complétés qui respectent la Définition de Terminé (« Definition of Done ») [6].

2.1.2.3 Événements Scrum

Scrum repose sur plusieurs événements pour organiser et synchroniser le travail :

- **Sprint** : un cycle de développement itératif et temporellement limité (généralement 1 à 4 semaines) durant lequel un incrément est créé. Tous les événements Scrum se déroulent pendant ce cadre temporel.
- **Sprint Planning (planification du Sprint)** : est une réunion en début de sprint où l'équipe définit ce qu'elle va livrer et comment elle va s'y prendre.
- **Daily Scrum** : une réunion quotidienne de 15 minutes où les développeurs synchronisent leur travail et planifient la journée à venir. Elle améliore la communication, l'inspection et l'adaptation.
- **Sprint Review** : a lieu à la fin du sprint et permet à l'équipe de présenter l'incrément aux parties prenantes, d'en recueillir des retours et de réviser le Product Backlog si nécessaire.
- **Sprint Rétrospective** : est la dernière réunion du sprint, où l'équipe réfléchit à la façon dont le travail s'est déroulé et propose des améliorations pour le prochain sprint [6].

Pour appliquer efficacement Scrum dans le développement de notre application de gestion des charges des distributeurs, il est crucial de structurer et de visualiser clairement les différentes composantes du système, pour cela ,nous avons opté pour le langage de modélisation UML (Unified Modeling Language).

2.1.3 Langage de modélisation UML

Le langage UML (Unified Modeling Language) est constitué de diagrammes intégrés utilisés par les développeurs informatiques pour la représentation visuelle des objets, des états et des processus dans un logiciel ou un système. Le langage de modélisation peut servir de modèle pour un projet et garantir une architecture d'information structurée ; il peut également aider les développeurs à présenter leur description d'un système d'une manière compréhensible pour les spécialistes externes.

UML est principalement utilisé dans le développement de logiciels orientés objet. Les améliorations apportées à la norme dans la version 2.0 la rendent également adaptée à la représentation des processus de gestion [7].

Il existe 14 types de diagrammes permettant de illustrer à la fois la structure et le comportement d'un système. Ces diagrammes sont répartis en trois grandes catégories illustrés par la figure 2.1.2.

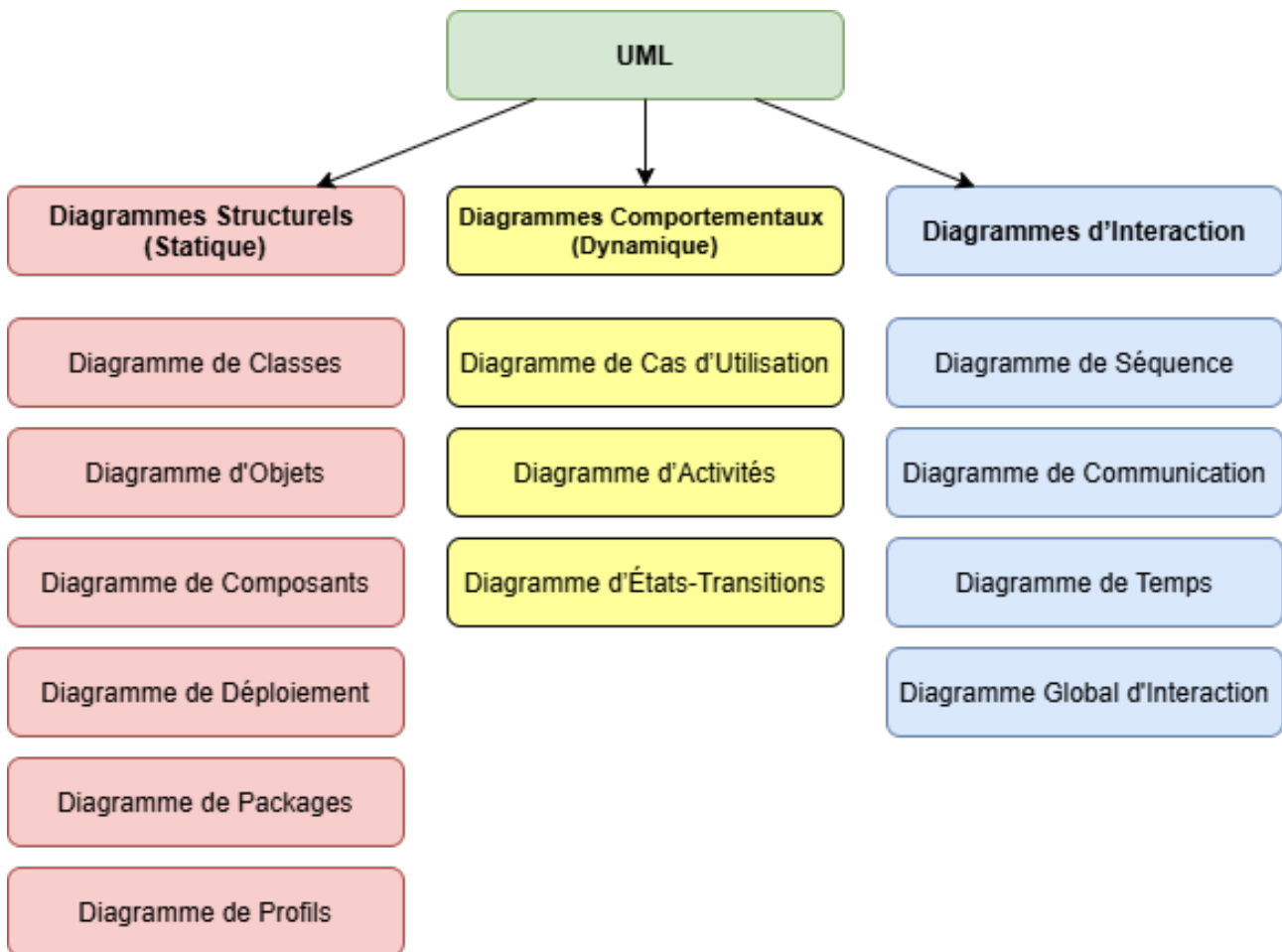


FIGURE 2.1.2 – Les diagrammes UML

2.1.4 Le design pattern Modèle-Vue-Contrôleur (MVC)

Le Modèle-Vue-Contrôleur (MVC) est un design pattern (patron de conception) utilisé pour structurer les applications logicielles. Il consiste à séparer le modèle de données, l'interface utilisateur et la logique de contrôle.

Ce modèle d'architecture impose une séparation claire entre les données, les traitements et la présentation, ce qui aboutit à trois composants fondamentaux dans l'application finale : le Modèle, la Vue et le Contrôleur.

- **Le modèle** : illustre le comportement de l'application ,le traitement des données, interactions avec la base de données, etc. Il décrit les données manipulées par l'application et définit les méthodes d'accès.
- **La vue** : correspond à l'interface avec laquelle l'utilisateur interagit. Les résultats renvoyés par le modèle sont dénués de toute présentation mais sont présentés par les vues.
- **Le contrôleur** : prend en charge la gestion des évènements de synchronisation pour mettre à jour la vue ou le modèle. Il n'effectue aucun traitement, ne modifie aucune donnée, il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondante à la demande [8].

La figure 2.1.3 illustre cette architecture et les interactions entre ses composants :

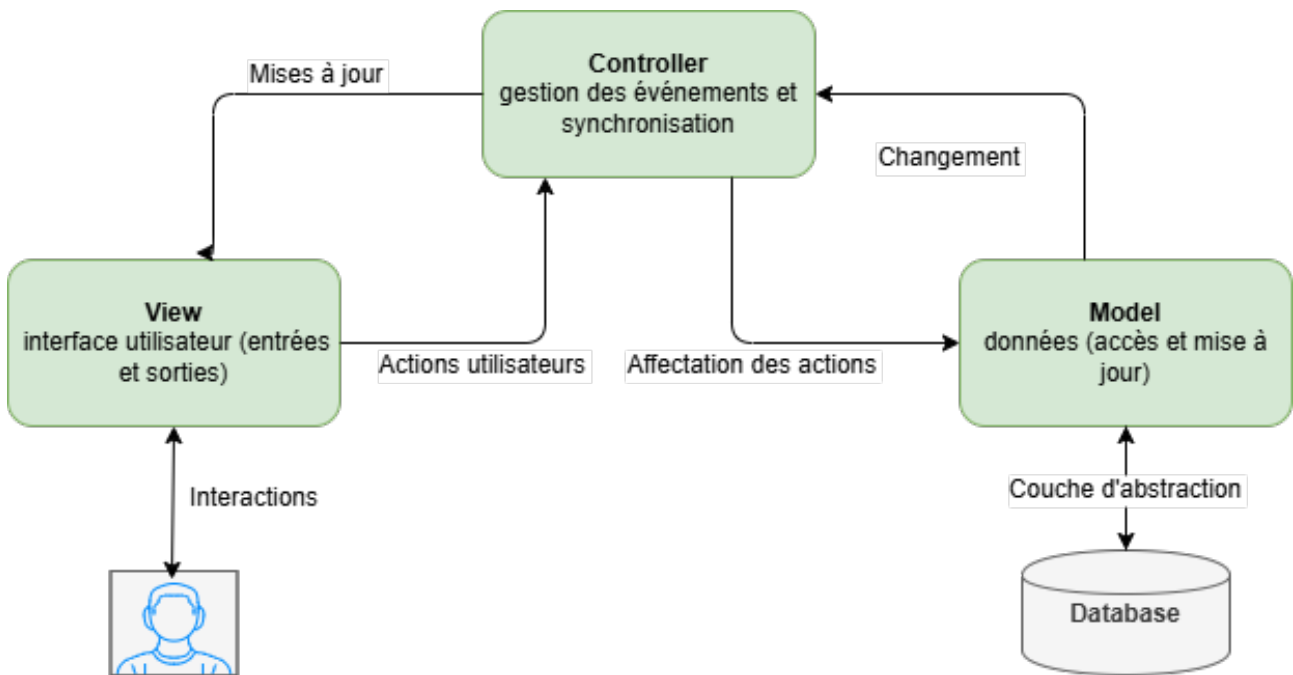


FIGURE 2.1.3 – Architecture MVC

2.2 Pilotage du projet avec Scrum

2.2.1 Sprint 0

2.2.1.1 Rôles

Pour notre application, les rôles sont répartis comme suite :

Rôle Scrum	Personnes affectées
Product Owner	Service commercial de Cevital Béjaia
Scrum Master	Mr. ZERARGA Loufi
Équipe de développement	Mlle HAMITOUCHE Lynda et Mlle DJERROUD Sabrina

TABLE 2.1 – La répartition des rôles pour la réalisation du projet.

2.2.1.2 Identification des acteurs

Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système [9]. Dans notre application, on a six acteurs qui sont :

- **Administrateur** : Responsable de la gestion et de l'administration de l'application, incluant la gestion des comptes des utilisateurs . Il veille aussi au bon fonctionnement du serveur de données et sa sécurité et il fait partie du département des systèmes d'informations.
- **Superviseur** : Il contrôle les présences des employés ainsi que les flottes de distribution au sein des dépôts d'un canal donné, en plus de gérer les primes mensuelles des em-

ployés du Front-office d'un dépôt. Il s'assure que les opérations quotidiennes se déroulent efficacement et conformément aux directives établies.

- **Chargé du reporting** : Responsable de la gestion des distributeurs et leurs dépôts, la gestion du front-office et du back-office avec leur salaires, prestations ainsi que les charges fixes. Il a aussi pour rôle la gestion des objectifs et les réalisations des distributeurs ainsi leurs chiffres d'affaires avec le nombre de jours ouvrables pour chaque mois, Il veille aussi à la validation des présences des employés et des flottes du Front-office et procède à l'enregistrement ainsi qu'au calcul des charges des distributeurs pour chaque canal d'un dépôt, que ce soit pour les charges du Front-office ou du Back-office .
- **Chef de région** : Il est responsable des superviseurs relevant de sa région et de son canal. Son rôle consiste à coordonner les informations transmises par les superviseurs concernant les présences du personnel et des flottes au sein de sa zone géographique attribuée (Centre, Est ou Ouest).
- **National des ventes** : Il est responsable des chefs de région partageant le même canal que lui. Son rôle consiste à valider les informations remontées par les chefs de région et assure la cohérence des données.
- **Directeur commercial** : Il a pour rôle d'ajustement des montants des charges et à procéder à leur clôture.

2.2.1.3 Diagramme du cas d'utilisation global

Un diagramme du cas d'utilisation illustre le comportement fonctionnel d'un système logiciel, décomposé en plusieurs sprints formant le backlog produit. La figure 2.2.1 présente notre diagramme du cas d'utilisation global.

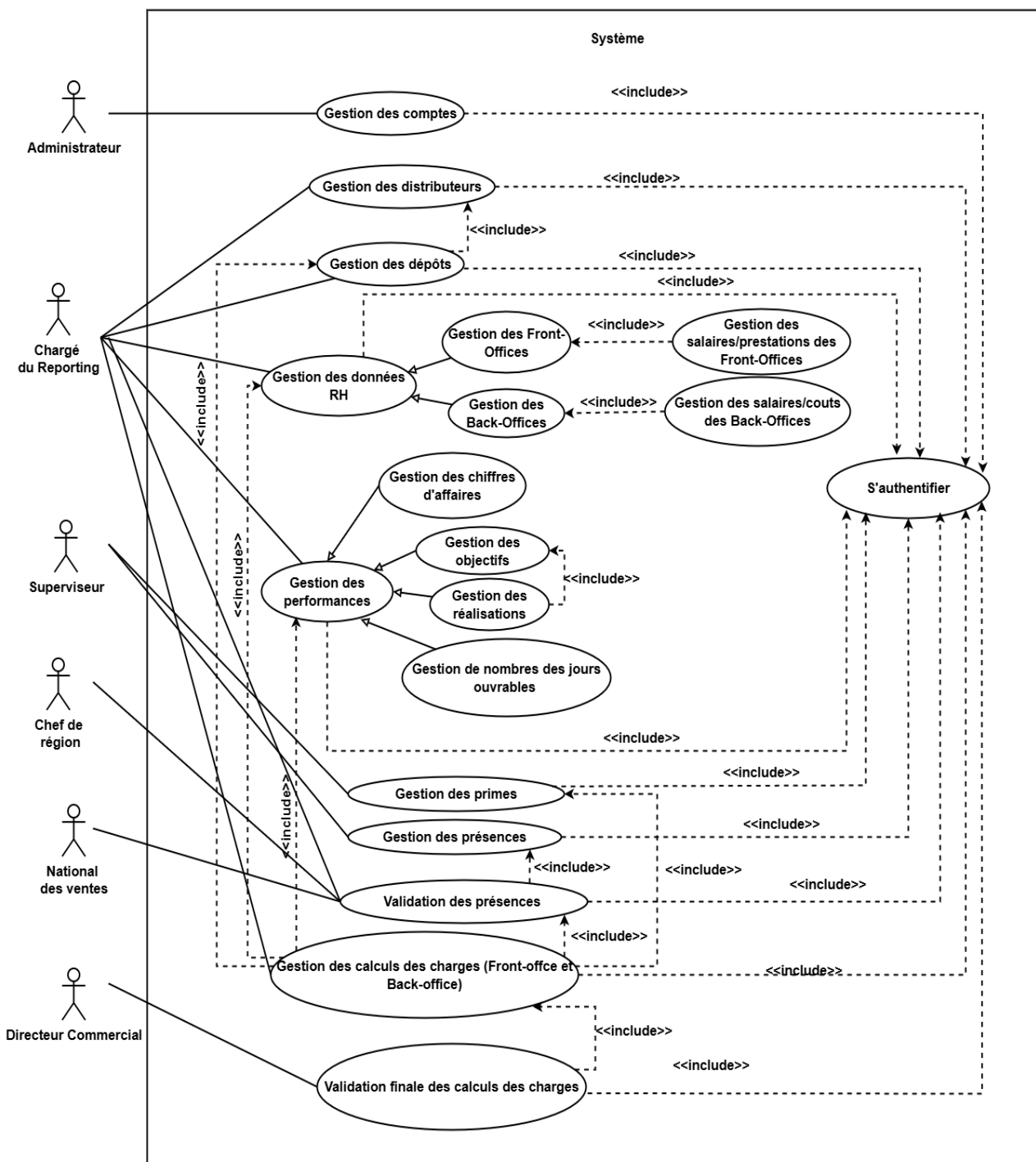


FIGURE 2.2.1 – Diagramme du cas d'utilisation global.

2.2.1.4 User Stories

Une user story est une explication informelle et générale d'une fonctionnalité logicielle rédigée du point de vue de l'utilisateur final. Son objectif est d'expliquer comment une fonctionnalité logicielle apportera de la valeur au client [10].

Dans le cadre de notre projet, nous avons identifié six acteurs qui interagissent directement avec le système étudié. Le tableau 2.2 présente les user-stories.

ID	User Story
US01	En tant qu'utilisateur (Administrateur, Superviseur, Chef de Région, National des ventes, Chargé du Reporting, Directeur Commercial), je souhaite m'authentifier afin d'accéder aux fonctionnalités de l'application selon mon rôle.
US02	En tant qu'Administrateur, je souhaite gérer les comptes des utilisateurs afin de pouvoir les ajouter, modifier, rechercher et consulter la liste des comptes existants.
US03	En tant que Chargé du Reporting, je souhaite gérer les distributeurs afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des distributeurs existants, ainsi que les importer et les exporter.
US04	En tant que Chargé du Reporting, je souhaite gérer les dépôts afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des dépôts existants, ainsi que les importer et les exporter.
US05	En tant que Chargé du Reporting, je souhaite gérer les effectifs du FrontOffice et du BackOffice afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des effectifs existants, ainsi que les importer et les exporter.
US06	En tant que Chargé du Reporting, je souhaite gérer les salaires du personnel du FrontOffice et du BackOffice afin de pouvoir les ajouter, les modifier, les rechercher, les consulter et exporter la liste des salaires existants.
US07	En tant que Chargé du Reporting, je souhaite gérer les prestations du FrontOffice et les charges fixes du BackOffice afin de pouvoir les ajouter, les modifier, les rechercher, les consulter et exporter la liste des prestations et des charges fixes existantes.
US08	En tant que Chargé du Reporting, je souhaite gérer les objectifs afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des objectifs existants, ainsi que les importer et les exporter.
US09	En tant que Chargé du Reporting, je souhaite gérer les réalisations afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des réalisations existantes, ainsi que les importer et les exporter.
US10	En tant que Chargé du Reporting, je souhaite gérer les chiffres d'affaires afin de pouvoir les ajouter, les modifier, les rechercher, consulter la liste des chiffres d'affaires existants, ainsi que les importer et les exporter.
US11	En tant que Chargé du Reporting, je souhaite gérer les jours ouvrables afin de pouvoir les ajouter, les modifier, les rechercher, ainsi que consulter la liste des jours ouvrables existants.

ID	User Story
US12	En tant que Chargé du Reporting, je souhaite calculer ou mettre à jour le total des charges du FrontOffice et du BackOffice, individuellement ou groupé, consulter et exporter l'historique des charges.
US13	En tant que Superviseur, je souhaite gérer les présences des employés et le suivi des flottes d'un dépôt afin de pouvoir les ajouter, modifier les présences rejetées et consulter l'historique des présences.
US14	En tant que Superviseur, je souhaite gérer les primes afin de pouvoir les ajouter, les modifier, les rechercher, consulter et exporter la liste des primes existantes.
US15	En tant que Chef de Région, National des ventes ou Chargé du Reporting, je souhaite valider les présences et le suivi des flottes en les acceptant ou en les rejetant.
US16	En tant que Directeur Commercial, je souhaite ajuster les montants des charges à libérer, afin de les valider et de procéder à leur clôture finale, et exporter l'historique des charges.

TABLE 2.2 – Liste des User Stories.

2.2.1.5 Product backlog

Dans ce tableau, nous avons détaillé le Product Backlog, qui présente les différentes fonctionnalités définies à partir des User Stories mentionnées précédemment.

Légende des priorités :

- **Haute** : fonctionnalité essentielle au bon fonctionnement de l'application.
- **Moyenne** : fonctionnalité importante, mais non bloquante.
- **Faible** : fonctionnalité optionnelle ou amélioration envisagée pour une version future.

Fonctionnalités	User Stories	Priorité
Authentification	US01 — En tant qu'utilisateur (Administrateur, Superviseur, Chargé du Reporting, Chef de Région, National des ventes, Directeur Commercial), je souhaite m'authentifier pour accéder à l'application.	Haute
Gestion des comptes	US02 — En tant qu'Administrateur, je souhaite gérer les comptes des utilisateurs (ajouter, modifier, rechercher, consulter la liste des comptes).	Haute

Fonctionnalités	User Stories	Priorité
Gestion des distributeurs	US03 — En tant que Chargé du Reporting, je souhaite gérer les distributeurs (ajouter, modifier, rechercher, consulter, importer et exporter la liste des distributeurs).	Moyenne
Gestion des dépôts	US04 — En tant que Chargé du Reporting, je souhaite gérer les dépôts (ajouter, modifier, rechercher, consulter, exporter et importer la liste des dépôts).	Moyenne
Gestion des effectifs (FrontOffice et BackOffice)	US05 — En tant que Chargé du Reporting, je souhaite gérer les effectifs du FrontOffice et du BackOffice (ajouter, modifier, rechercher, consulter, exporter et importer).	Moyenne
Gestion des salaires	US06 — En tant que Chargé du Reporting, je souhaite gérer les salaires du personnel du FrontOffice et du BackOffice (ajouter, modifier, rechercher, consulter, exporter).	Moyenne
Gestion des prestations et charges fixes	US07 — En tant que Chargé du Reporting, je souhaite gérer les prestations du FrontOffice et les charges fixes du BackOffice (ajouter, modifier, rechercher, consulter, exporter).	Moyenne
Gestion des objectifs	US08 — En tant que Chargé du Reporting, je souhaite gérer les objectifs (ajouter, modifier, rechercher, consulter, importer et exporter).	Moyenne
Gestion des réalisations	US09 — En tant que Chargé du Reporting, je souhaite gérer les réalisations (ajouter, modifier, rechercher, consulter, importer et exporter).	Moyenne
Gestion des chiffres d'affaires	US10 — En tant que Chargé du Reporting, je souhaite gérer les chiffres d'affaires (ajouter, modifier, consulter, importer et exporter).	Moyenne
Gestion des jours ouvrables	US11 — En tant que Chargé du Reporting, je souhaite gérer les jours ouvrables (ajouter, modifier, rechercher, consulter).	Faible

Fonctionnalités	User Stories	Priorité
Calcul des charges FrontOffice et BackOffice	US12 — En tant que Chargé du Reporting, je souhaite calculer les charges totales (individuelles ou groupées), consulter et exporter l'historique.	Haute
Gestion des présences et flottes	US13 — En tant que Superviseur, je souhaite gérer les présences du personnel et des flottes (ajouter, modifier, consulter l'historique).	Haute
Gestion des primes	US14 — En tant que Superviseur, je souhaite gérer les primes (ajouter, modifier, rechercher, consulter et exporter).	Faible
Validation des présences	US15 — En tant que Chef de Région, National des ventes ou Chargé du Reporting, je souhaite valider les présences et le suivi des flottes (accepter, rejeter).	Haute
Validation finale des charges	US16 — En tant que Directeur Commercial, je souhaite ajuster les montants à libérer, les valider et exporter l'historique des charges.	Haute

TABLE 2.3 – Product Backlog.

2.2.1.6 Planification des releases

La planification des releases permet d'organiser les sprints en fonction de leur priorité. Le tableau 2.4 présente cette organisation.

Release	Sprint	Fonctionnalités
Release 1	Sprint 1 (<i>Espace Administrateur</i>)	- Authentification - Gestion des comptes
	Sprint 1 (<i>Espace Chargé Reporting</i>)	- Gestion des distributeurs - Gestion des dépôts
	Sprint 2 (<i>Espace Chargé Reporting</i>)	- Gestion du Front-office - Gestion du Back-office - Gestion des salaires, prestations, coûts du Front-office et du Back-office

Release	Sprint	Fonctionnalités
Release 2	Sprint 3 <i>(Espace Chargé Reporting)</i>	- Gestion des chiffres d'affaires - Gestion des objectifs mensuels - Gestion des réalisations - Gestion des jours ouvrables pour chaque mois
	Sprint 4 <i>(Espace Superviseur et Validation)</i>	- Gestion des présences des employés et des flottes - Gestion des primes - Circuit de validation des présences
Release 3	Sprint 5 <i>(Fonction de calcul des charges)</i>	- Calcul des charges du Front-office et du Back-office - Modification et validation finale des charges

TABLE 2.4 – Sprint Backlog selon la méthode Scrum, organisé par Release.

Dans le cadre de la gestion agile du projet selon la méthode Scrum, la répartition des fonctionnalités entre les différents sprints et releases repose sur une logique de priorisation progressive. Elle consiste à livrer d'abord une base fonctionnelle stable, puis à intégrer progressivement les modules métiers essentiels, avant de terminer par les traitements complexes liés au calcul et à la validation des charges.

La répartition suivante présente la priorisation des fonctionnalités développées au cours des différents sprints et releases.

- Release 1 : Sprints 1 & 2 – Fondations de l'application.

Les fonctionnalités choisies posent les bases du système :

- Sprint 1 : assure la sécurité avec l'authentification et la gestion des comptes et introduit les entités principales du métier (les distributeurs et les dépôts).
- Sprint 2 : englobe les fonctionnalités de gestion des Front-Office, Back-office, salaires, prestations et coûts des Front-offices et Back-offices, ce sont des fonctionnalités essentielles pour structurer les données.

- Release 2 : Sprints 3 & 4 – Suivi et performance

Les fonctionnalités choisies posent les bases du système :

- Sprint 3 : couvre le suivi commercial : Jours ouvrables, chiffres d'affaires, objectifs et réalisations.
- Sprint 4 : introduit le suivi opérationnel : présences des personnels et des flottes, validations hiérarchiques, primes. Ces fonctionnalités apportent une vision complète des performances et du terrain.

- Release 3 : Sprint 5 – Calcul et clôture des charges

Ce dernier sprint est dédié au calcul des charges, à leur validation finale et à la clôture, basée sur les données précédemment intégrées. Il représente l'objectif final du système.

Conclusion

Ce chapitre nous a permis de présenter les différentes approches méthodologiques et techniques adoptées pour la réalisation du projet. Nous avons mis en avant la méthode Scrum, ses principes et son rôle dans le pilotage du projet, ainsi que l'architecture MVC et la modélisation UML pour assurer une structuration claire du développement.

De plus, nous avons détaillé les éléments essentiels à la gestion du projet, notamment la répartition des rôles, l'identification des acteurs, la modélisation du diagramme du cas d'utilisation global, la rédaction des User Stories, la constitution du Product Backlog, ainsi que la planification des releases sous forme de sprints successifs, assurant ainsi une organisation efficace et cohérente du projet.

Chapitre 3

Release 1

Introduction

Dans ce chapitre, nous allons présenter le travail effectué lors du premier release, qui comprend les sprints 1 et 2 portant sur les différentes parties de l'application.

Le sprint 1 a permis de mettre en place les bases du système, notamment l'authentification sécurisée des utilisateurs et la gestion des comptes, garantissant ainsi un accès contrôlé et sécurisé avec la gestion des distributeurs et de leurs dépôts.

Le sprint 2 a ensuite permis de développer les fonctionnalités de gestion des effectifs (Front-Office et Back-Office), ainsi que des salaires, prestations et coûts des Front-Offices et Back-offices, assurant ainsi un suivi complet des opérations et des finances.

Pour chaque sprint, nous exposerons les diagrammes du cas d'utilisation, accompagnés des descriptions textuelles et des diagrammes de séquence correspondant à chaque cas. Un diagramme de classes sera également présenté pour chaque sprint, suivi des captures d'écran illustrant les interfaces développées.

3.1 Sprint 1 : Espace Administrateur et Chargé du Reporting

Ce sprint s'est étalé sur 2 semaine, il comprend 4 items qui sont :

- **L'authentification** : Est le processus permettant de vérifier l'identité d'un utilisateur, afin de garantir que seuls les acteurs légitimes puissent accéder aux ressources protégées du système d'information.
- **Gestion des comptes** : Regroupe l'ensemble des fonctionnalités destinées à administrer les comptes utilisateurs, notamment la création de comptes pour les superviseurs, chefs de région, nationaux des ventes, chargé du reporting et le directeur commercial, ainsi que leur modification, leur recherche et la consultation de l'ensemble des profils existants.
- **Gestion des distributeurs** : Cet item permet de gérer les distributeurs en offrant des fonctionnalités d'ajout, de modification, de recherche et de consultation de la liste des

distributeurs.

- **Gestion des dépôts** : Cette fonctionnalité permet de gérer les dépôts de chaque distributeur, avec la possibilité d'ajouter, de modifier, de rechercher et de consulter les dépôts enregistrés, ainsi que l'affectation des superviseurs pour chaque canal d'un dépôt.

3.1.1 Diagramme du cas d'utilisation du sprint 1

Le diagramme du cas d'utilisation de la figure 3.1.1 illustre les différents cas d'utilisation correspondant aux éléments du sprint 1.

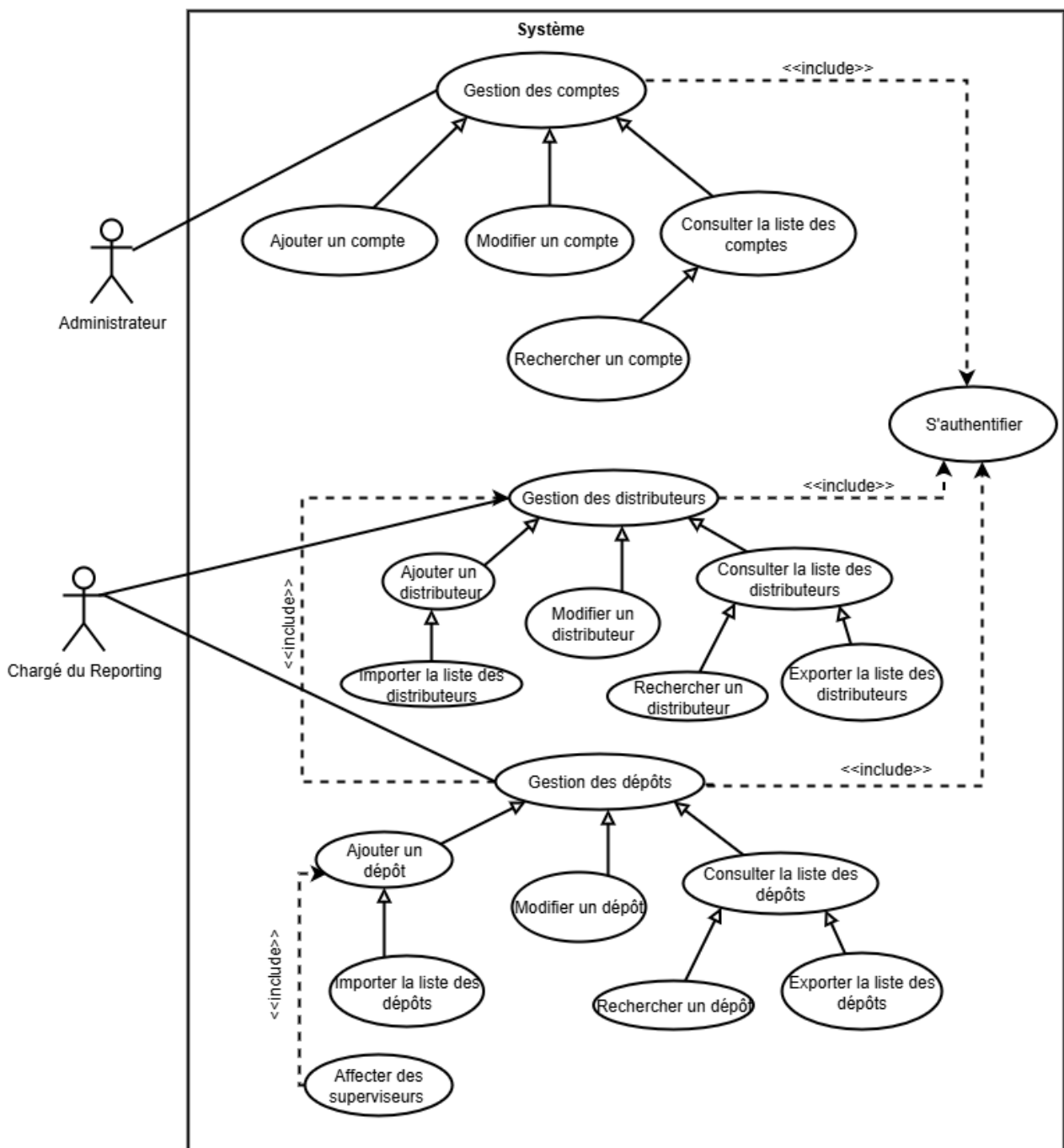


FIGURE 3.1.1 – Diagramme du cas d'utilisation du sprint 1.

3.1.2 Cas d'utilisation "S'authentifier"

3.1.2.1 Description textuelle du cas d'utilisation "S'authentifier"

La description textuelle du cas d'utilisation "S'authentifier" est détaillée dans le tableau 3.1.

Élément	Description
Cas d'utilisation	S'authentifier.
Acteurs	Administrateur, Chargé du reporting, Chef de région, National des ventes, Superviseur, Directeur commercial.
Description	Lorsqu'un utilisateur du système veut accéder à l'application, il doit saisir son nom d'utilisateur et son mot de passe. Ensuite, le système vérifie s'ils sont corrects ou non afin d'autoriser ou refuser l'accès.
Précondition	L'utilisateur doit avoir un compte.
Scénario nominal	1. L'utilisateur introduit son nom d'utilisateur et le mot de passe de son compte ; 2. Si l'utilisateur est identifié, le système le dirige vers l'interface d' « Accueil ».
Scénario alternatif	À l'étape 1 du scénario nominal : Nom d'utilisateur ou mot de passe non valide : Le système affiche un message d'erreur : « Votre nom d'utilisateur ou votre mot de passe est incorrect ».
Post-condition	Authentification et accès autorisé.

TABLE 3.1 – Tableau descriptif du cas d'utilisation "S'authentifier".

3.1.2.2 Diagramme de séquence du cas d'utilisation "S'authentifier"

La figure 3.1.2 illustre le diagramme de séquence du cas d'utilisation "S'authentifier".

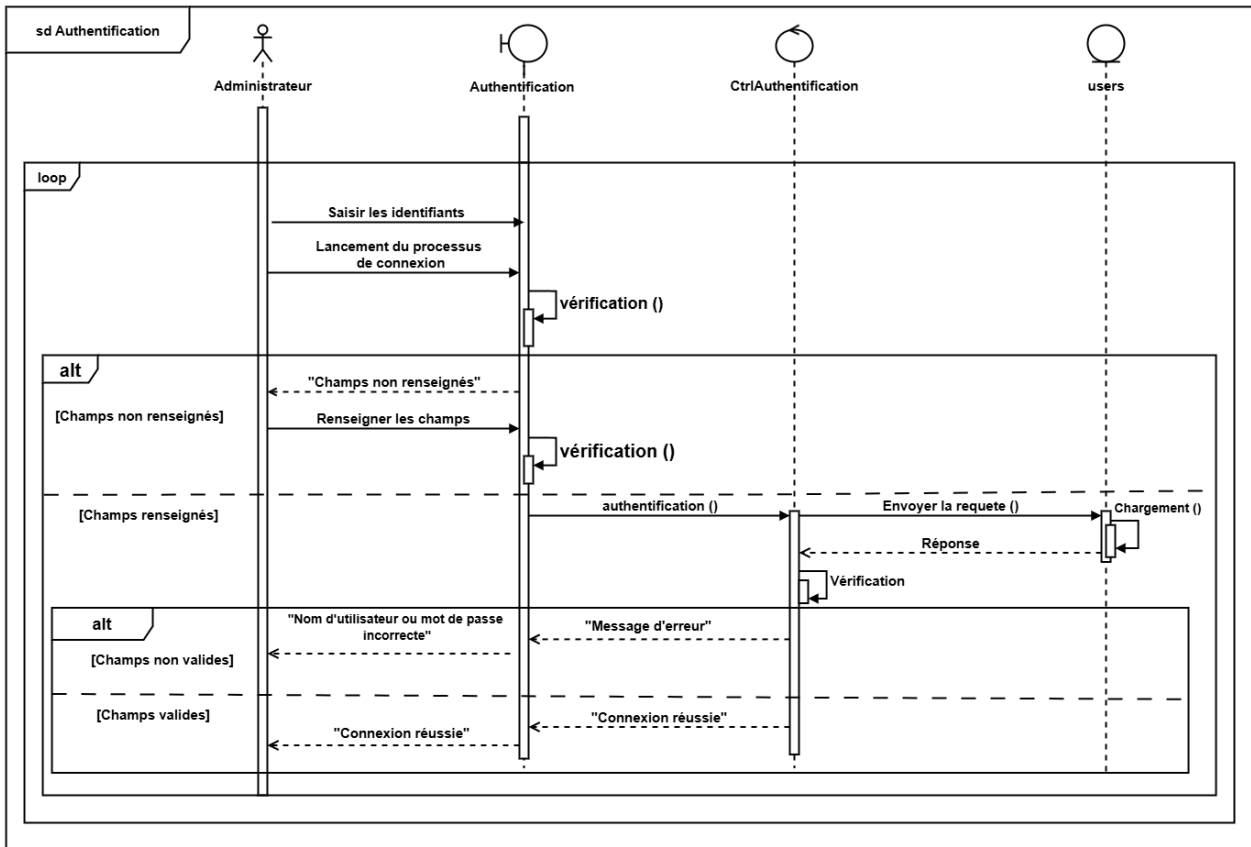


FIGURE 3.1.2 – Diagramme de séquence du cas d’utilisation ”S’authentifier”.

3.1.3 Cas d’utilisation ”Ajouter un distributeur”

Dans ce qui suit, nous présentons le scénario détaillé de ce cas d’utilisation globale, à travers des descriptions textuelles mettant en évidence les interactions entre l’acteur principal et le système, ainsi que des diagrammes de séquence illustrant le déroulement des échanges de manière visuelle et structurée.

3.1.3.1 Description textuelle du cas d’utilisation ”Ajouter un distributeur”

La description textuelle du cas d’utilisation ”Ajouter un distributeur” est détaillée dans le tableau 3.2.

Élément	Description
Cas d'utilisation	Ajouter un distributeur.
Acteurs	Chargé du Reporting.
Description	Ce cas permet au chargé du reporting d'ajouter un nouveau distributeur.
Précondition	Le chargé du reporting doit posséder un compte valide. Le chargé du reporting doit être authentifié.
Scénario nominal	<ol style="list-style-type: none"> 1. Le chargé du reporting clique sur l'élément 'Distributeurs' du menu pour accéder aux distributeurs ; 2. Le système le redirige vers l'interface des distributeurs ; 3. Le chargé du reporting clique sur le bouton 'Ajouter un distributeur' ; 4. Le système affiche le formulaire d'ajout ; 5. Le chargé du reporting remplit le formulaire d'ajout puis il enregistre ; 6. Le système vérifie le formulaire d'ajout puis crée le nouveau distributeur.
Scénario alternatif	À l'étape 6 du scénario nominal : Si le formulaire est mal rempli, le système affiche un message d'erreur.
Post condition	Le distributeur est ajouté avec succès.

TABLE 3.2 – Tableau descriptif du cas d'utilisation "Ajouter un distributeur".

3.1.3.2 Diagramme de séquence du cas d'utilisation "Ajouter un distributeur"

La figure 3.1.3 illustre le diagramme de séquence du cas d'utilisation "Ajouter un distributeur".

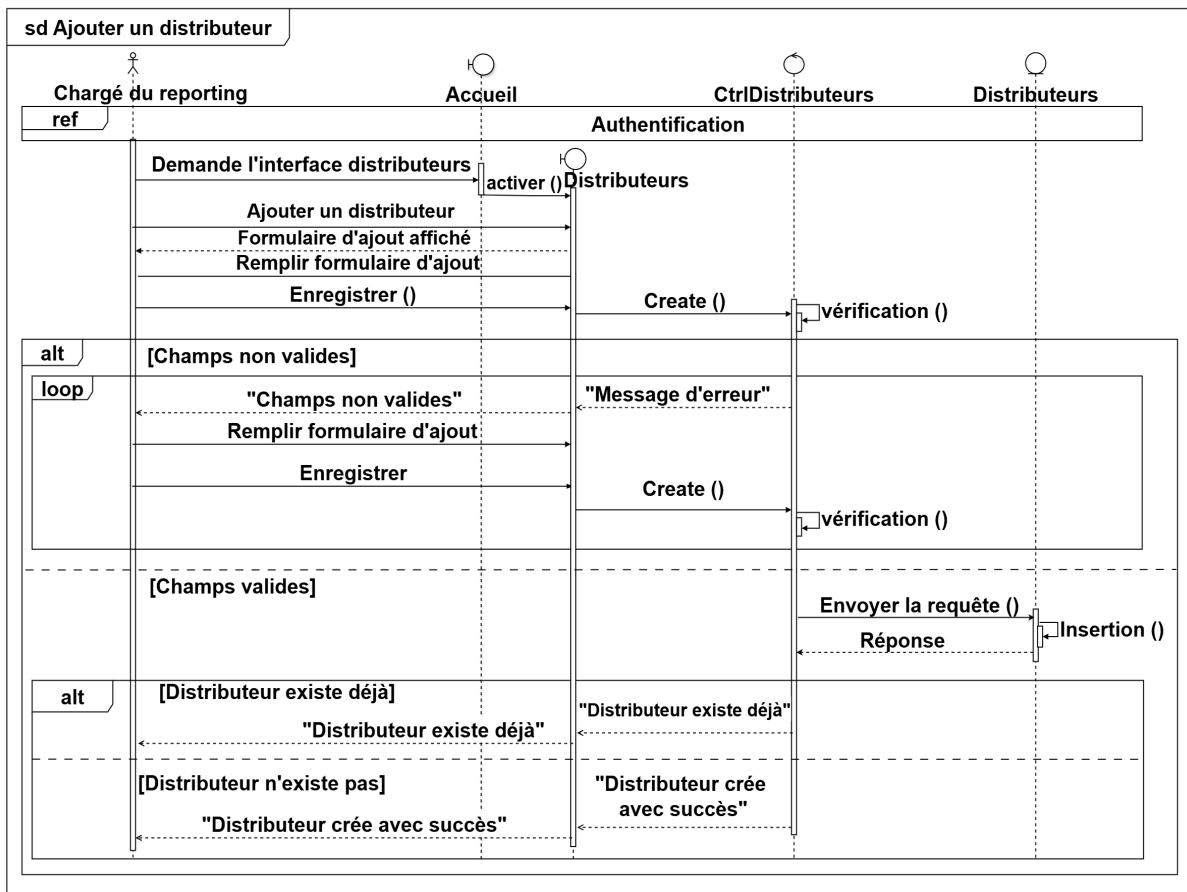


FIGURE 3.1.3 – Diagramme de séquence du cas d'utilisation "Ajouter un distributeur".

3.1.4 Diagramme de classe du Sprint 1

La figure 3.1.4 illustre le diagramme de classe du sprint 1.

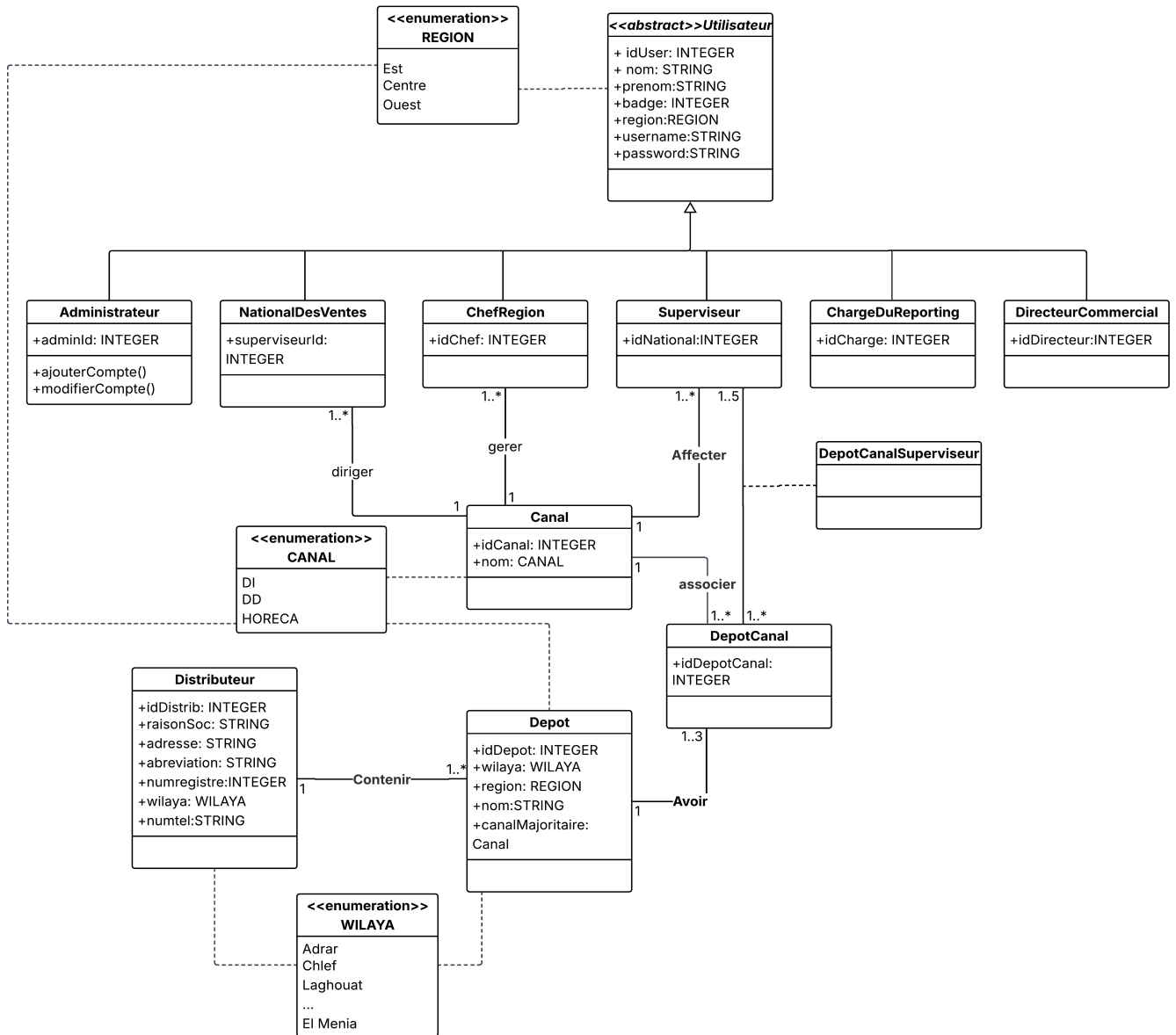


FIGURE 3.1.4 – Diagramme de classe du sprint 1.

3.1.5 Interface du sprint 1

Dans ce qui suit, nous allons présenter les interfaces réalisées lors du sprint 1.

3.1.5.1 Interface Authentification

Grâce à cette interface présentée dans la figure 3.1.5, l'utilisateur peut s'authentifier et accéder à l'espace qui correspond à son profil.

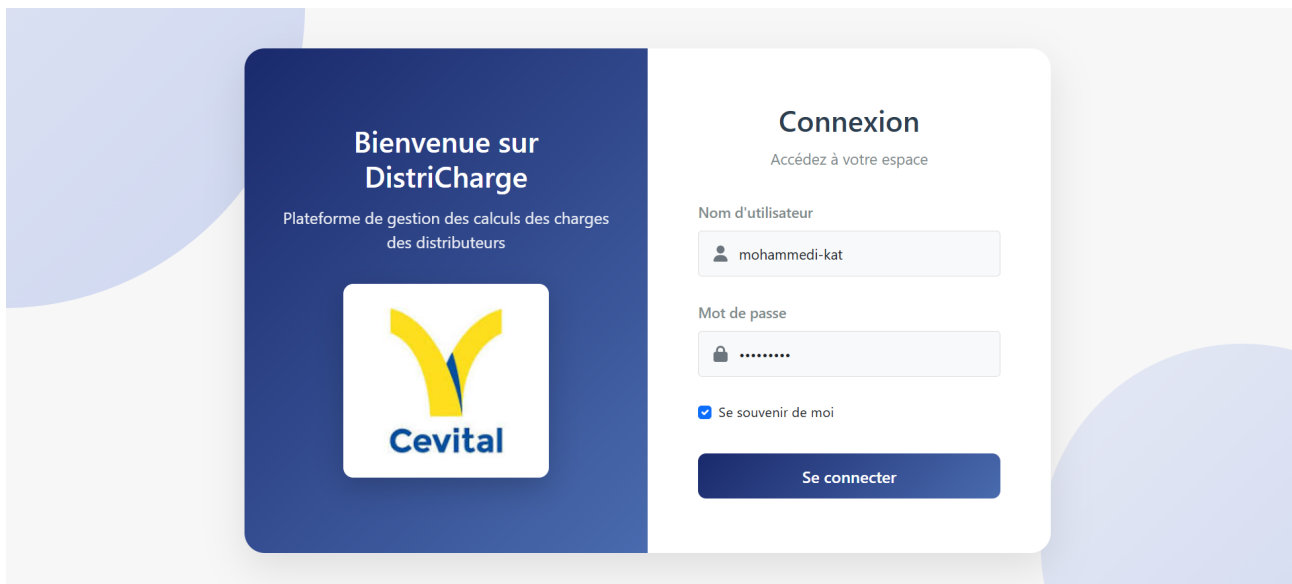


FIGURE 3.1.5 – Interface d'Authentification

3.1.5.2 Interface Distributeurs

L'interface présentée dans la figure 3.1.6 permet au chargé du reporting d'ajouter un distributeur.

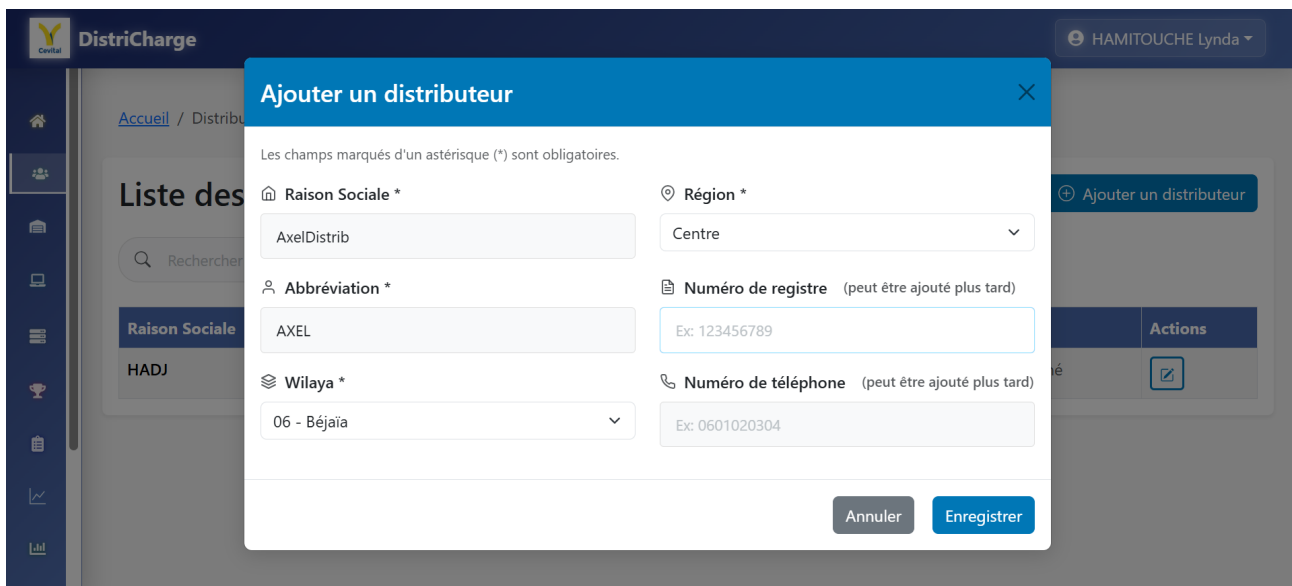


FIGURE 3.1.6 – Interface Distributeurs

3.1.5.3 Interface Dépôts

L'interface dans la figure 3.1.7 présente la liste des dépôts.

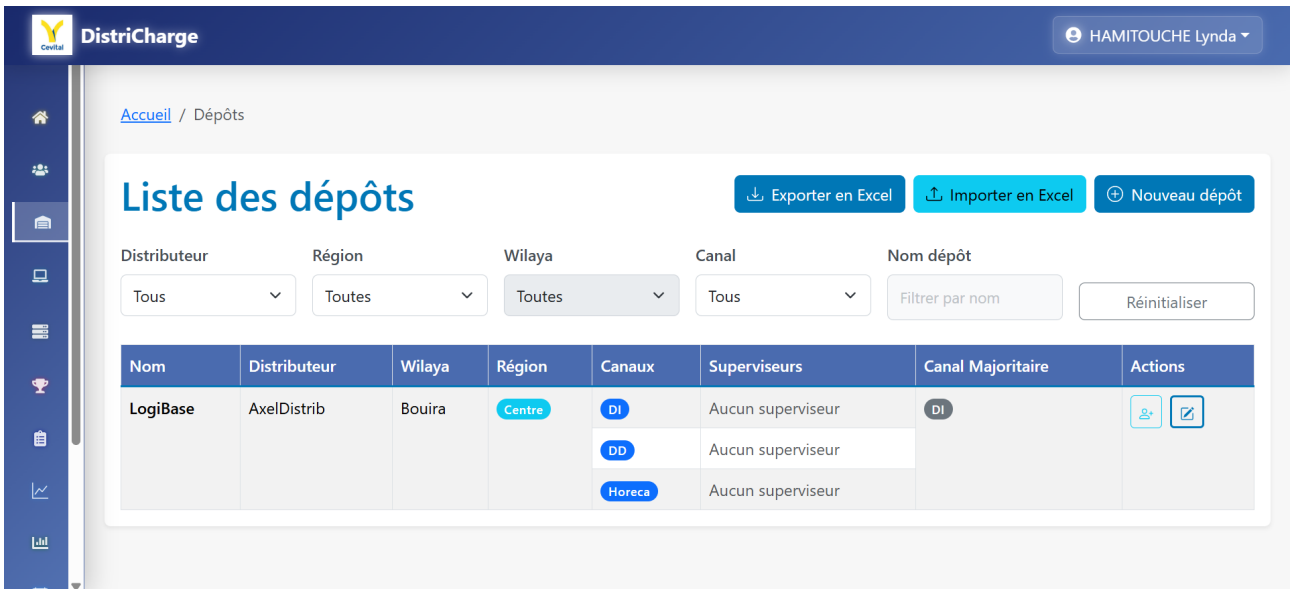


FIGURE 3.1.7 – Interface Dépôts.

Cette interface présentée dans la figure 3.1.8 permet au chargé du reporting d’affecter 1 à 5 superviseurs pour un dépôt d’un canal donné.

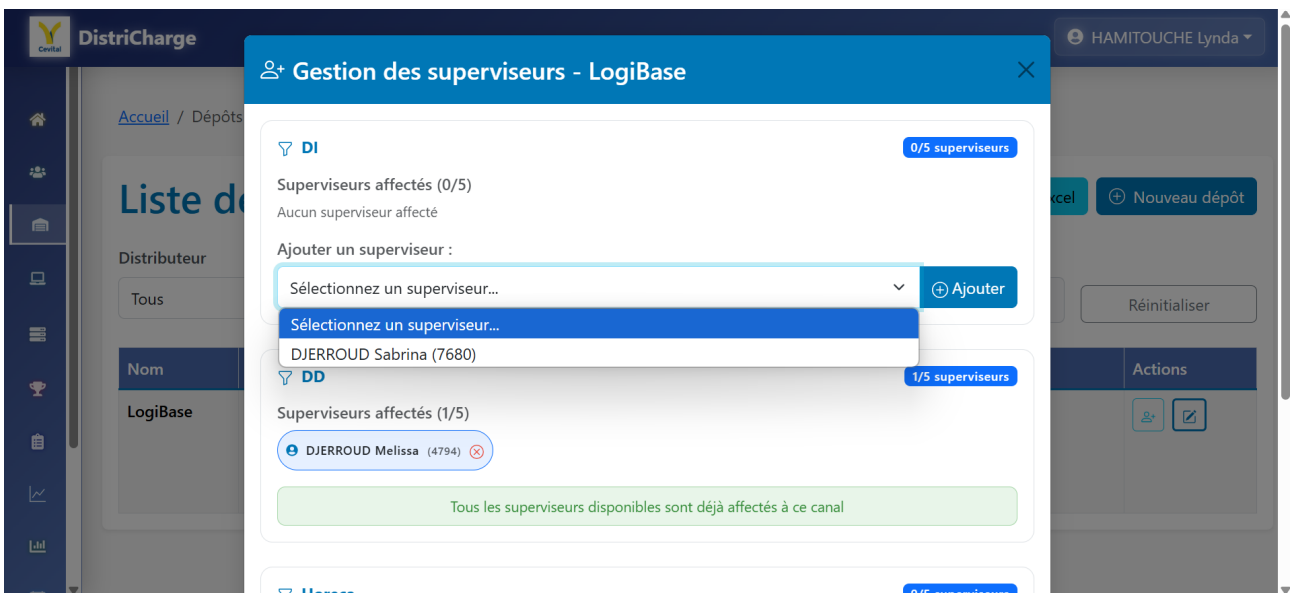


FIGURE 3.1.8 – Interface Dépôts.

3.2 Sprint 2 : Espace du Chargé du Reporting

Ce sprint s’est étalé sur 3 semaines, il comprend 5 items qui sont :

- **Gestion des Front-Offices** : Cette fonctionnalité permet de gérer les effectifs du Front-Office avec les flottes de distribution, en offrant la possibilité d’ajouter le nombre maximal des effectifs et des flottes pour chaque canal d’un dépôt, de modifier leurs valeurs, de les rechercher dans le système et de consulter la liste complète des effectifs du Front-Office associés aux différents dépôts.

- **Gestion des Back-Offices** : Similaire à celle des Front-Offices, cette fonctionnalité concerne les effectifs et les charges de type Back-Office . Elle permet l'ajout, la modification, la recherche et la consultation des employés occupant des fonctions de back-office au sein de l'organisation.
- **Gestion des salaires, prestations et coûts des Front-Offices et des Back-Offices** : Cette fonctionnalité permet d'enregistrer et de gérer les informations relatives aux salaires, prestations et coûts des Front-Offices et des Back-Offices. L'utilisateur peut saisir les montants, effectuer des ajustements, et consulter les salaires par employé ou par charge pour le Back-Office, ainsi que par flotte pour le Front-Office.

3.2.1 Diagramme du cas d'utilisation du deuxième sprint

La figure 3.2.1 présente le diagramme du cas d'utilisation décrivant les fonctionnalités prévues dans le cadre du deuxième sprint.

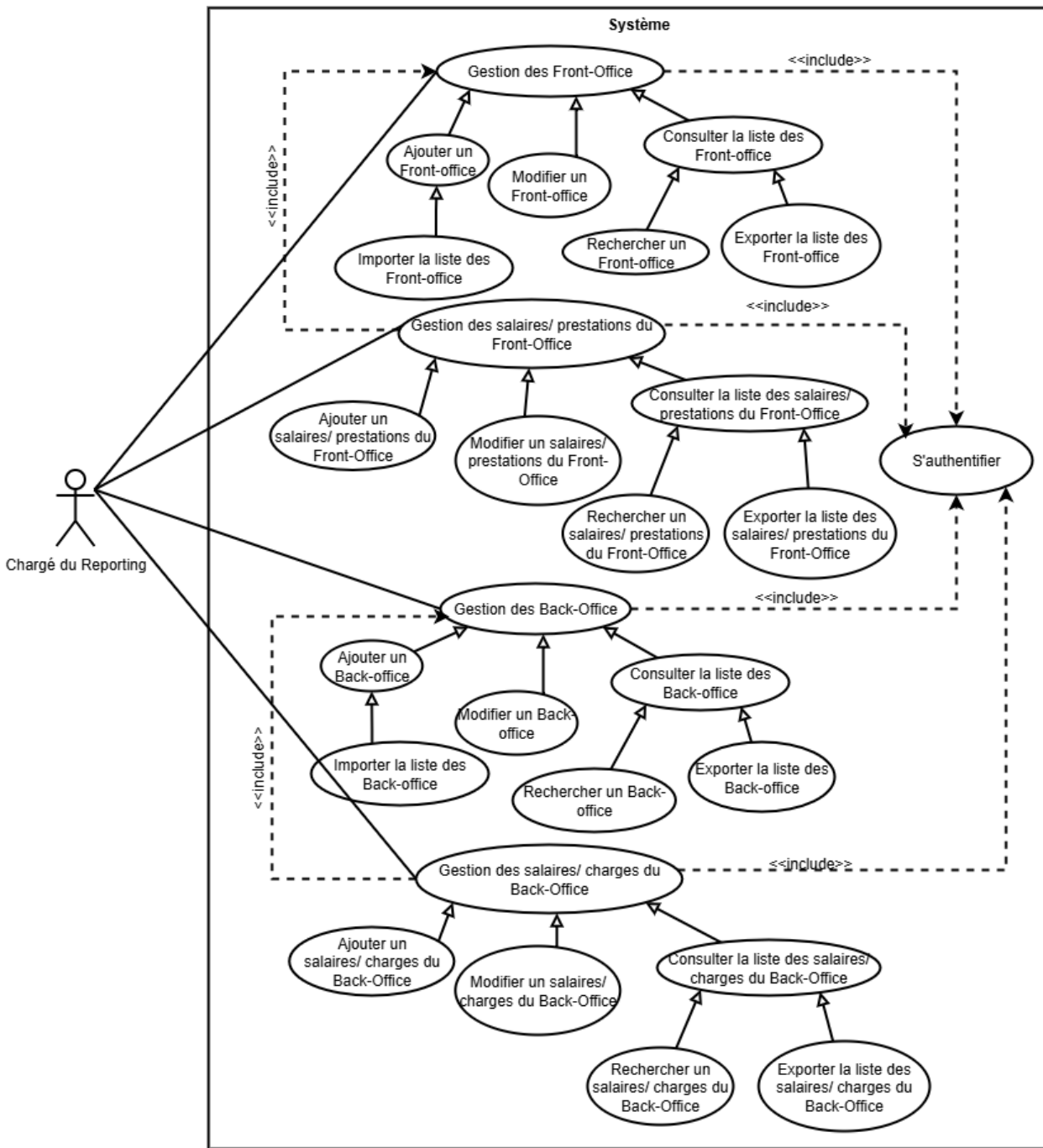


FIGURE 3.2.1 – Diagramme du cas d'utilisation du sprint 2.

3.2.2 Cas d'utilisation "Modifier le salaire d'un Front-Office"

3.2.2.1 Description textuelle du cas d'utilisation "Modifier le salaire d'un Front-Office"

La description textuelle du cas d'utilisation "Modifier le salaire d'un Front-Office" est détaillée dans le tableau 3.3.

Élément	Description
Cas d'utilisation	Modifier le salaire d'un Front-Office.
Acteurs	Chargé du Reporting.
Description	Ce cas permet au chargé du reporting de modifier les informations du salaire d'un Front-Office.
Précondition	Le chargé du reporting doit posséder un compte valide. Le chargé du reporting doit être authentifié.
Scénario nominal	<ol style="list-style-type: none"> 1. Le chargé du reporting clique sur l'élément "Salaire" du 'Front-Office' du menu pour accéder à la liste des salaires ; 2. Le système le redirige vers l'interface des salaires ; 3. Le chargé du reporting recherche le salaire par distributeur ou par dépôt ; 4. Le chargé du reporting clique sur le bouton modifier ; 5. Le système affiche le formulaire de modification ; 6. Le chargé du reporting effectue les modifications nécessaires puis il valide ; 7. Le système vérifie le formulaire ; 8. Le système vérifie et met à jour les données.
Scénario alternatif	À l'étape 6 du scénario nominal : Si le formulaire est mal rempli, le système affiche un message d'erreur.
Post-condition	Le salaire est modifié avec succès.

TABLE 3.3 – Tableau descriptif du cas d'utilisation "Modifier le salaire d'un Front-Office".

3.2.2.2 Diagramme de séquence du cas d'utilisation "Modifier le salaire d'un Front-Office"

La figure 3.2.2 illustre le diagramme de séquence du cas d'utilisation "Modifier le salaire d'un Front-Office".

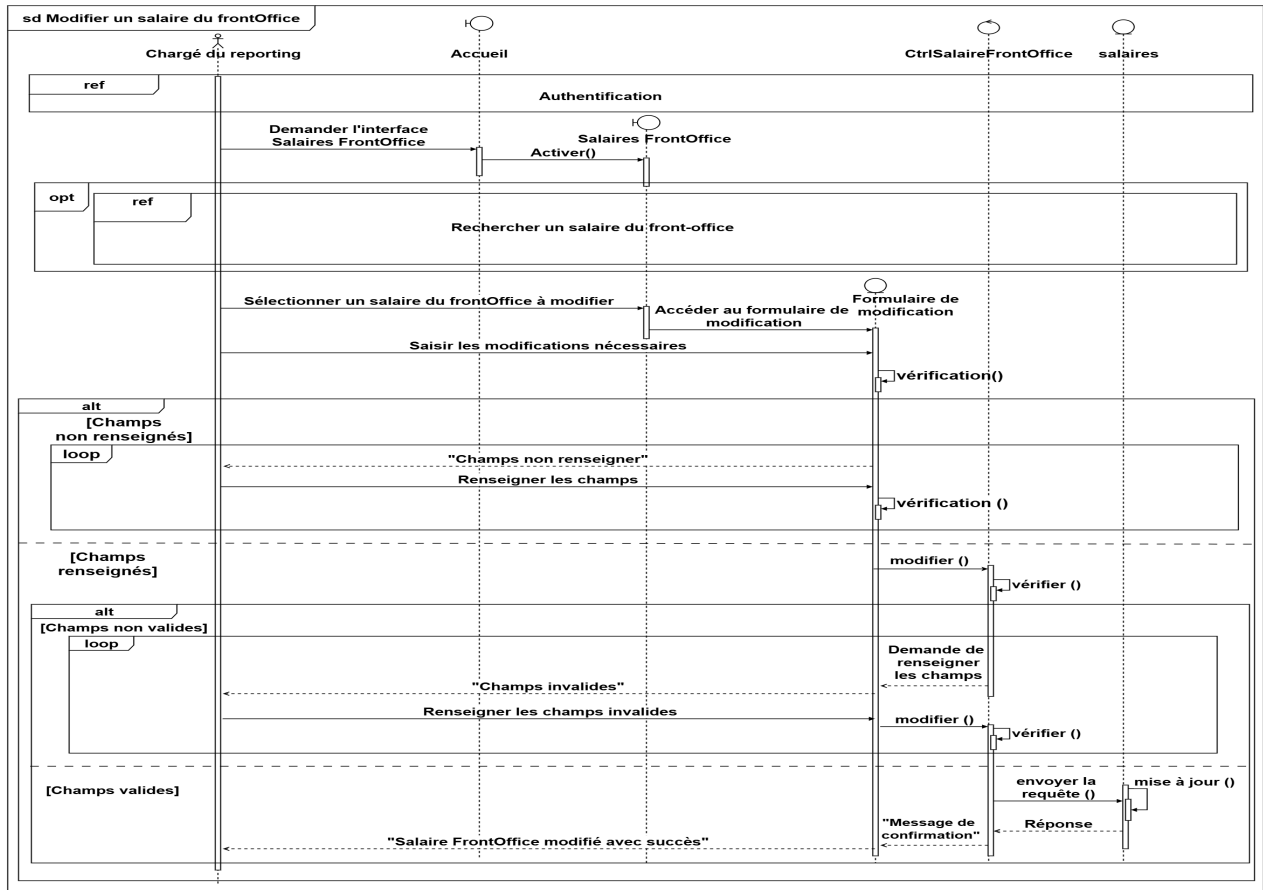


FIGURE 3.2.2 – Diagramme de séquence du cas d’utilisation ”Modifier le salaire d’un Front-Office”.

3.2.3 Diagramme de classe du Sprint 2

La figure 3.2.3 illustre le diagramme de classe du sprint 2.

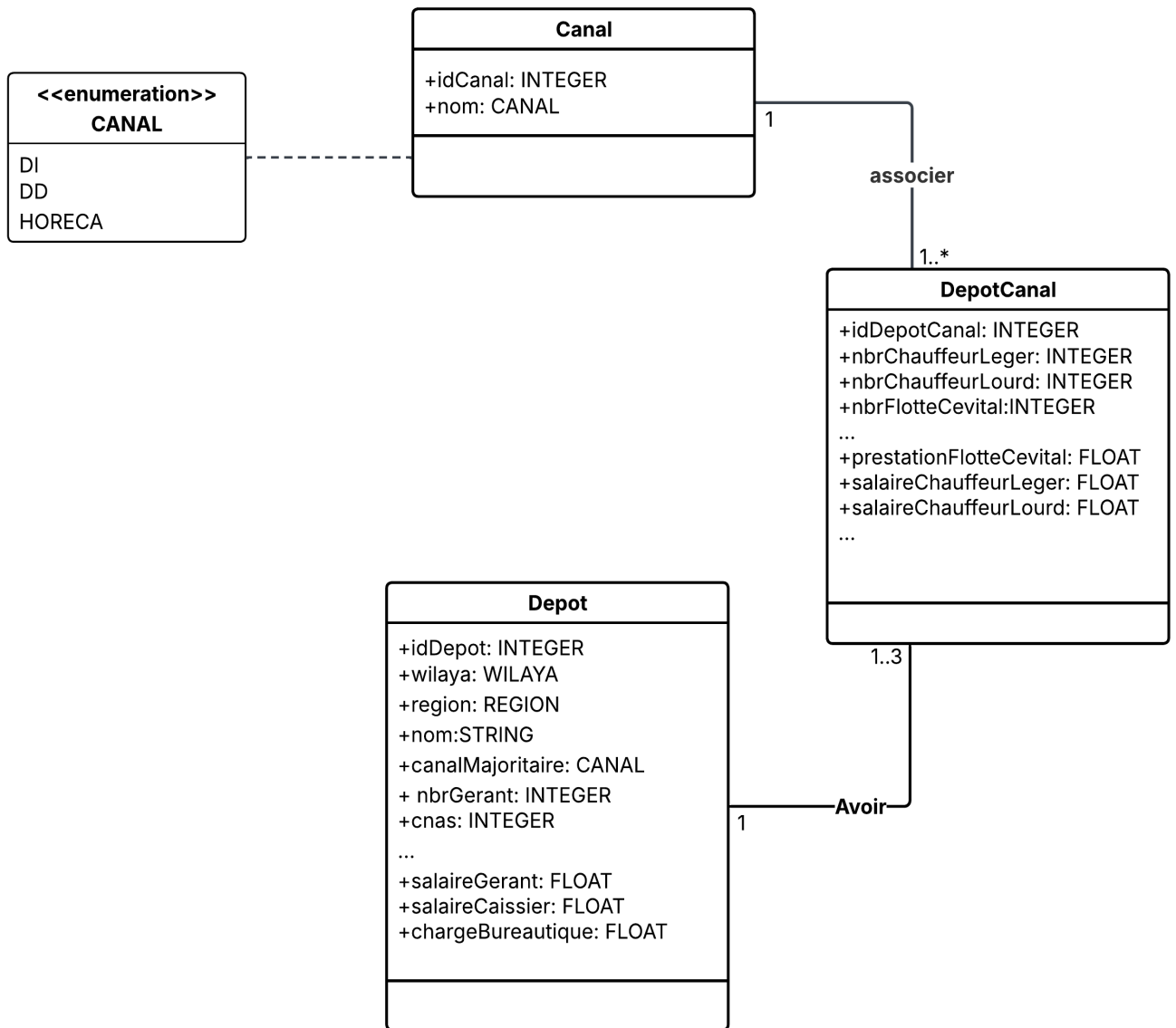


FIGURE 3.2.3 – Diagramme de classe du sprint 2.

3.2.4 Interface du sprint 2

Dans ce qui suit, nous allons présenter les interfaces réalisées lors du sprint 2.

3.2.4.1 Interface Effectif Front-Office

Cette interface présentée dans la figure 3.2.4 permet au chargé du reporting d’ajouter le nombre maximum des personnels et des flottes pour chaque canal d’un dépôt.

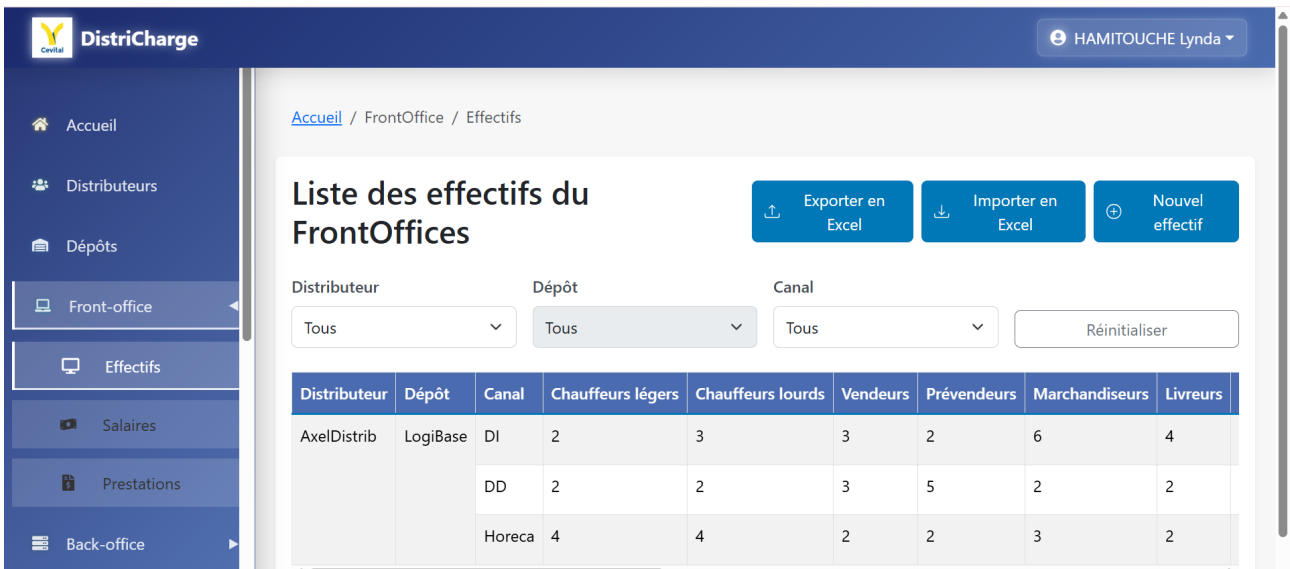


FIGURE 3.2.4 – Interface Effectif Front-Office.

3.2.4.2 Interface Charges Fixes du Back-Office

Cette interface illustrée dans la figure 3.2.5 permet au chargé du reporting d’ajouter les charges fixes d’un Back-office pour un dépôt donné.

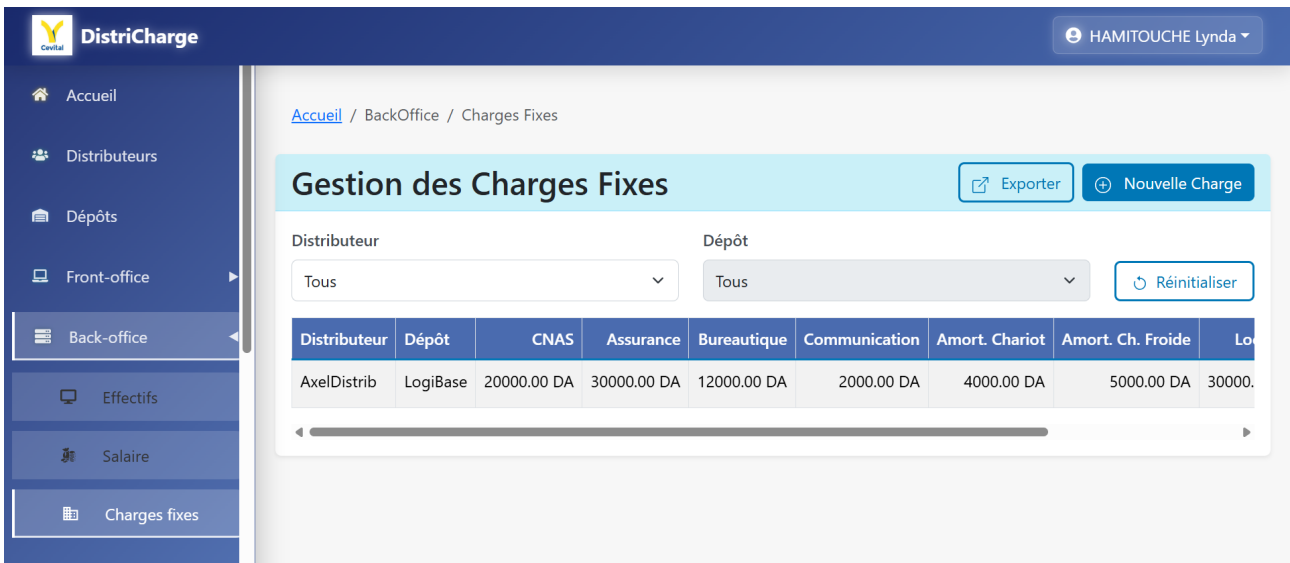


FIGURE 3.2.5 – Interface Charges Fixes du Back-Office.

Conclusion

Dans ce chapitre consacré à l’analyse des sprints 1 et 2 du premier release, nous avons réalisé une analyse approfondie des fonctionnalités développées lors de ces deux phases initiales du projet. Les diagrammes UML (cas d’utilisation, séquences et classes) et les captures d’écran des interfaces ont permis d’illustrer clairement la logique métier et l’expérience utilisateur. Ces travaux fournissent une base solide et structurée pour les prochaines étapes de développement.

Chapitre 4

Release 2

Introduction

Dans ce chapitre, nous présentons le deuxième release, couvrant les sprints 3 et 4.

Le Sprint 3 a porté sur le suivi des performances commerciales à travers la gestion des chiffres d'affaires, des objectifs, des réalisations ainsi que le nombre de jours ouvrables pour chaque mois.

Le Sprint 4 s'est axé sur la gestion des ressources humaines, incluant la gestion des primes, des présences des employés et des flottes, ainsi que le circuit de validation hiérarchique associé.

Pour chaque sprint, nous détaillerons les cas d'utilisation à travers des diagrammes accompagnés de descriptions textuelles, ainsi que les diagrammes de séquence correspondants. Un diagramme de classes sera également fourni pour chaque sprint, suivi des captures d'écran illustrant les interfaces développées durant cette phase.

4.1 Sprint 3 : Espace du Chargé du Reporting

Ce sprint s'est étalé sur 2 semaines, il comprend 4 items qui sont :

- **Gestion du nombre de jours ouvrables du mois** : Cette fonctionnalité permet au chargé du reporting d'ajouter, modifier, rechercher et consulter le nombre de jours ouvrables pour chaque mois.
- **Gestion des objectifs** : Cette fonctionnalité permet au chargé du reporting de suivre les objectifs fixés pour chaque distributeur selon les canaux concernés. Elle comprend les actions de création, de mise à jour, d'importation, d'exportation, de recherche ainsi que la consultation de l'ensemble des objectifs enregistrés.
- **Gestion des réalisations** : Cette fonctionnalité permet au chargé du reporting de gérer les réalisations enregistrées pour chaque distributeur en fonction de ses canaux. Elle inclut l'ajout, la modification, l'importation, l'exportation, la recherche ainsi que la consultation de la liste des réalisations disponibles.
- **Gestion des chiffres d'affaires** : Cet item permet au Chargé du reporting de gérer les chiffres d'affaires associés à chaque distributeur selon ses canaux. Il offre les fonctionnalités

d'ajout, de modification, d'importation, d'exportation, de recherche et de consultation de la liste des chiffres d'affaires.

4.1.1 Diagramme du cas d'utilisation du sprint 3

Le diagramme du cas d'utilisation dans la figure 4.1.1 illustre les différents cas d'utilisation correspondant aux éléments du troisième sprint.

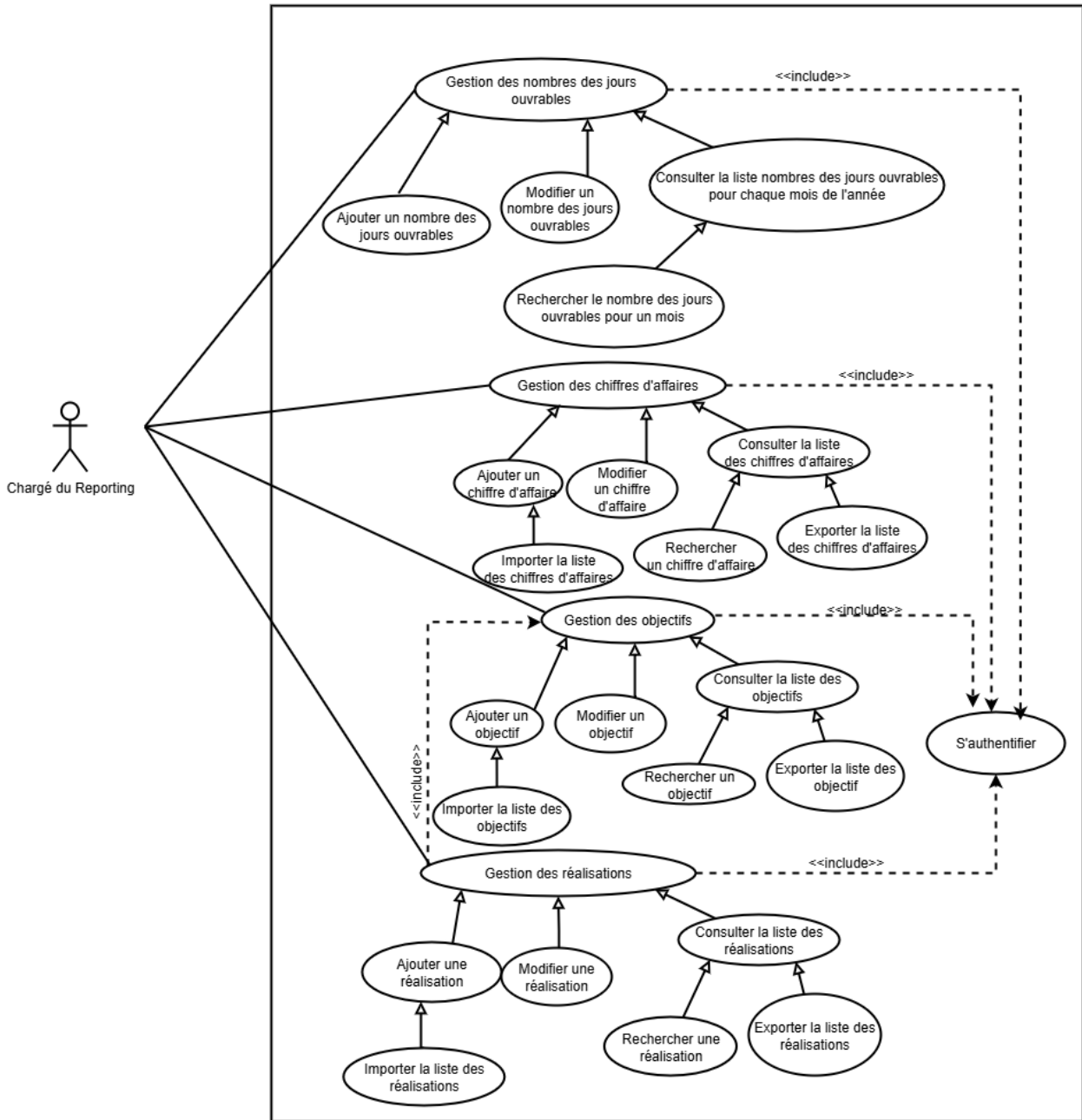


FIGURE 4.1.1 – Diagramme du cas d'utilisation du sprint 3.

4.1.2 Cas d'utilisation "Gestion des réalisations"

4.1.2.1 Description textuelle du cas d'utilisation "Rechercher une réalisation"

La description textuelle du cas d'utilisation "Rechercher une réalisation" est détaillée dans le tableau 4.1.

Élément	Description
Cas d'utilisation	Rechercher une réalisation.
Acteurs	Chargé du Reporting.
Description	Permet au chargé du reporting de rechercher une réalisation spécifique à un distributeur dans la liste des réalisations.
Précondition	Le chargé du reporting doit être authentifié. La liste des réalisations doit être accessible.
Scénario nominal	<ol style="list-style-type: none"> 1. Le chargé du reporting clique sur l'élément "Réalisations" dans le menu ; 2. Le système redirige vers l'interface des réalisations ; 3. Le chargé du reporting saisit un critère de recherche (distributeur, dépôt, canal, mois, année) dans le champ dédié ; 4. Le système filtre les éléments en fonction du critère saisi ; 5. Le système affiche les résultats de la recherche.
Scénario alternatif	<p>À l'étape 5 du scénario nominal :</p> <p>"Aucun résultat trouvé" : Le système affiche un message "Aucun résultat trouvé".</p> <p>Recherche vide : Le système affiche toutes les réalisations disponibles.</p>
Post condition	Le chargé du reporting peut sélectionner une réalisation pour la modifier (si besoin). La réalisation recherchée est affichée avec succès.

TABLE 4.1 – Tableau descriptif du cas d'utilisation "Rechercher une réalisation".

4.1.2.2 Diagramme de séquence du cas d'utilisation "Rechercher une réalisation"

La figure 4.1.2 illustre le diagramme de séquence du cas d'utilisation "Rechercher une réalisation"

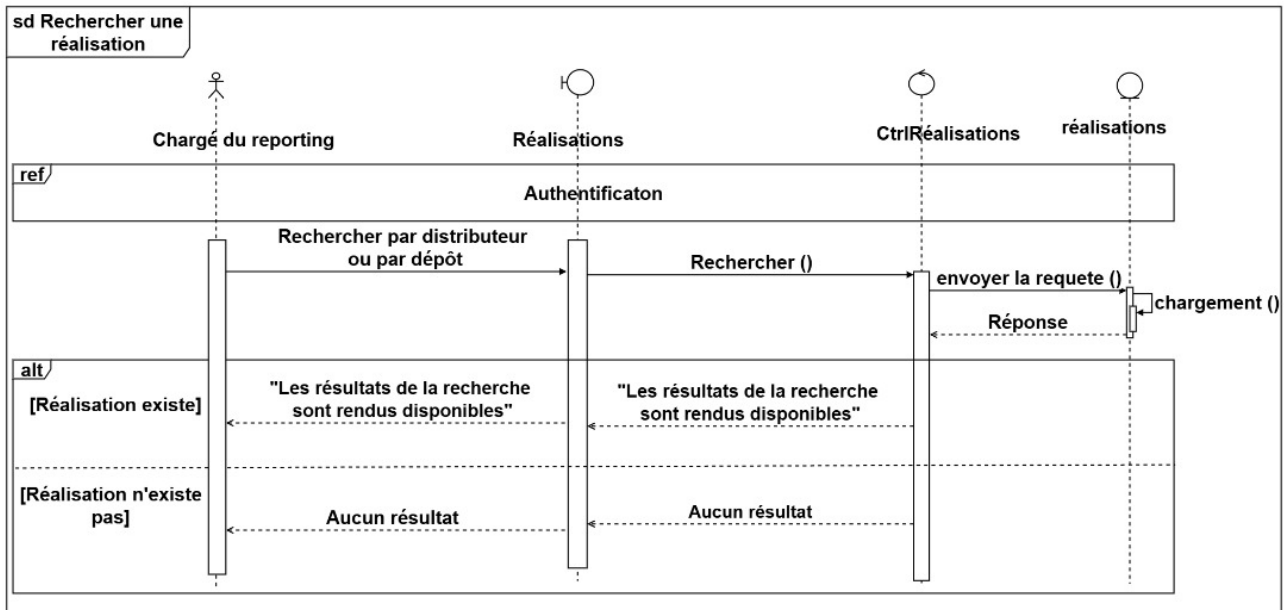


FIGURE 4.1.2 – Diagramme de séquence du cas d’utilisation "Rechercher une réalisation".

4.1.3 Cas d’utilisation "Consulter la liste des jours ouvrables pour chaque mois"

4.1.3.1 Description textuelle du cas d’utilisation "Consulter la liste des jours ouvrables pour chaque mois"

La description textuelle du cas d’utilisation "Consulter la liste des jours ouvrables" est détaillée dans le tableau 4.2.

Élément	Description
Cas d'utilisation	Consulter la liste des jours ouvrables pour chaque mois.
Acteurs	Chargé du Reporting.
Description	Permet au chargé du reporting d'afficher la liste des jours ouvrables existants pour chaque mois.
Préconditions	1. Le chargé du reporting doit posséder un compte valide. 2. Le chargé du reporting doit être authentifié.
Scénario nominal	1. Le chargé du reporting clique sur l'élément "Jours Ouvrables" dans le menu. 2. Le système redirige vers l'interface des jours ouvrables. 3. Le système charge et affiche la liste des jours ouvrables (triée par défaut).
Scénarios alternatifs	À l'étape 3 du scénario nominal : "Aucune donnée trouvée" : Le système affiche un message "Aucune donnée disponible". Erreur de chargement : Le système affiche un message d'erreur "Erreur lors du chargement des données".
Post-conditions	La liste des jours ouvrables est affichée correctement.

TABLE 4.2 – Tableau descriptif du cas d'utilisation "Consulter la liste des jours ouvrables".

4.1.3.2 Diagramme de séquence du cas d'utilisation "Consulter la liste des jours ouvrables"

La figure 4.1.3 illustre le diagramme de séquence du cas d'utilisation "Consulter la liste des jours ouvrables".

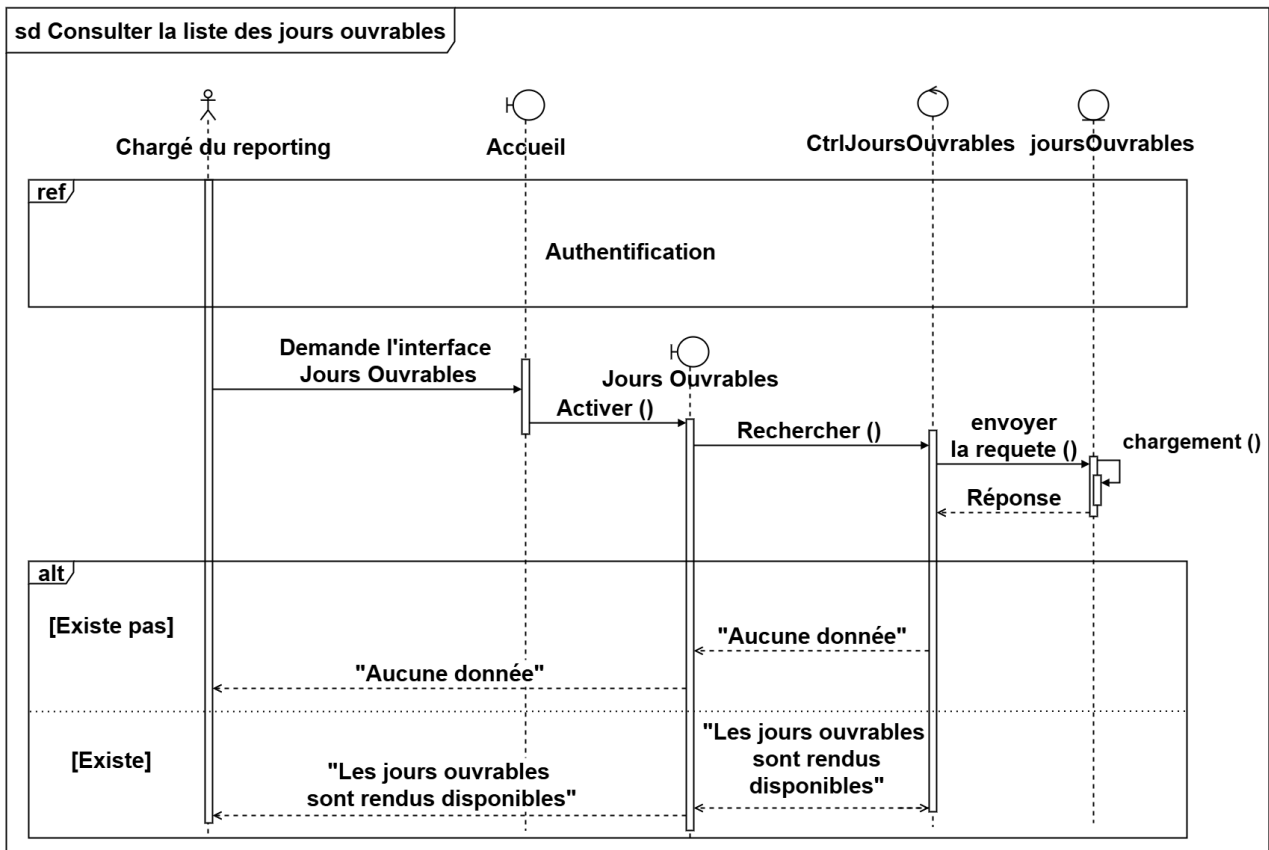


FIGURE 4.1.3 – Diagramme de séquence du cas d’utilisation ”Consulter la liste des jours ouvrables”.

4.1.4 Diagramme de classe du Sprint 3

La figure 4.1.4 illustre le diagramme de classe du sprint 3.

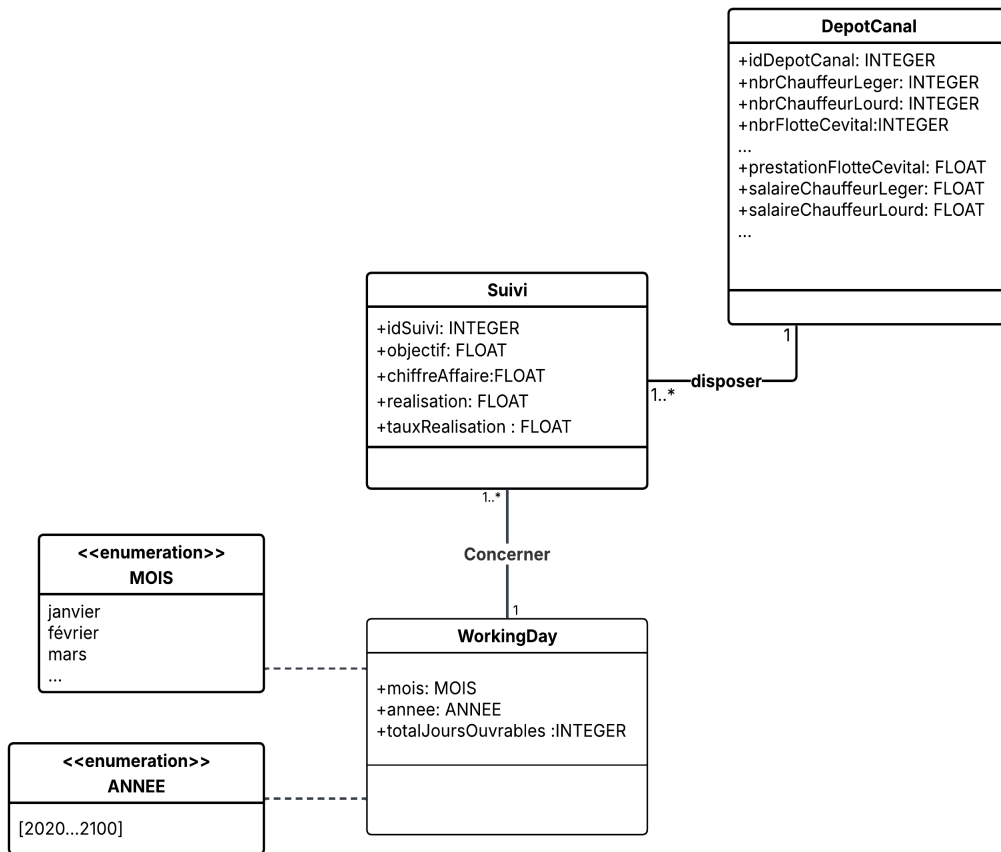


FIGURE 4.1.4 – Diagramme de classe du sprint 3

4.1.4.1 Interface Objectifs

L'interface illustrée dans la figure 4.1.5 permet au chargé du reporting d'ajouter les objectifs pour chaque canal d'un dépôt donné pour un distributeur donné.

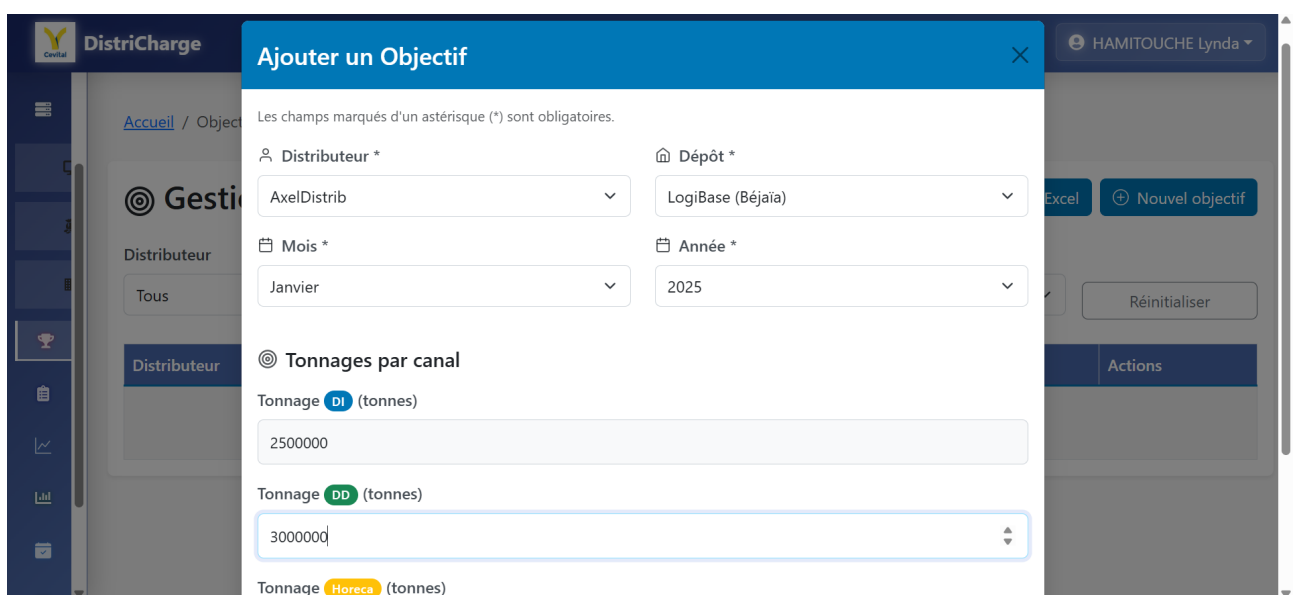


FIGURE 4.1.5 – Interface Objectifs.

4.1.4.2 Interface des jours ouvrables

Cette interface présentée dans la figure 4.1.6 permet au chargé du reporting de saisir le nombre de jours ouvrables pour chaque mois de l'année.

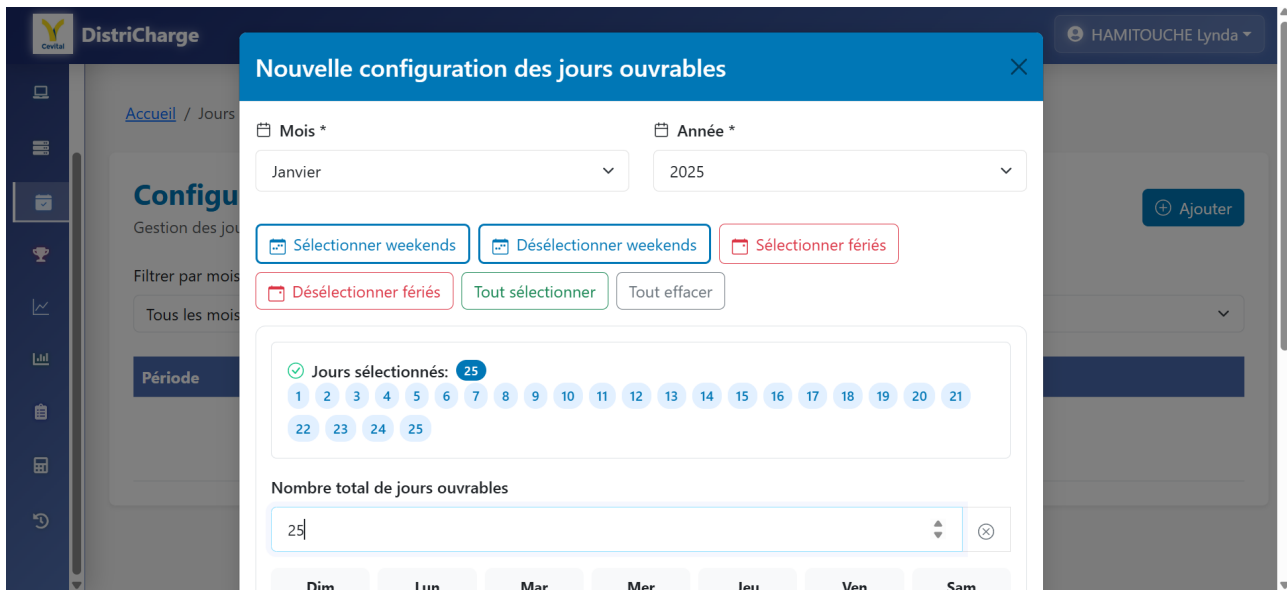


FIGURE 4.1.6 – Interface des jours ouvrables.

4.2 Sprint 4 : Espace du Superviseur et Validation

Ce sprint s'est étalé sur 2 semaines, il comprend 3 items qui sont :

- **Gestion des primes** : Cette fonctionnalité permet au chargé du reporting d'ajouter, modifier, rechercher et consulter les primes accordées aux distributeurs.
- **Gestion des présences des employés et des flottes** : Cette fonctionnalité permet au superviseur de saisir les présences quotidiennes des employés et des flottes pour chaque dépôt auquel il est affecté. Il peut également modifier les informations de la présence en cas de refus et consulter l'historique des saisies.
- **Gestion du circuit de validation des présences** : Cette fonctionnalité permet de gérer le circuit de validation des présences saisies par le superviseur. Les validateurs, tels que les chefs de région, les nationaux de vente,... peuvent approuver ou rejeter une présence, avec la possibilité d'ajouter un motif en cas de rejet.

4.2.1 Diagramme du cas d'utilisation du sprint 4

Le diagramme du cas d'utilisation dans la figure 4.2.1 illustre les différents cas d'utilisation correspondant aux éléments du quatrième sprint.

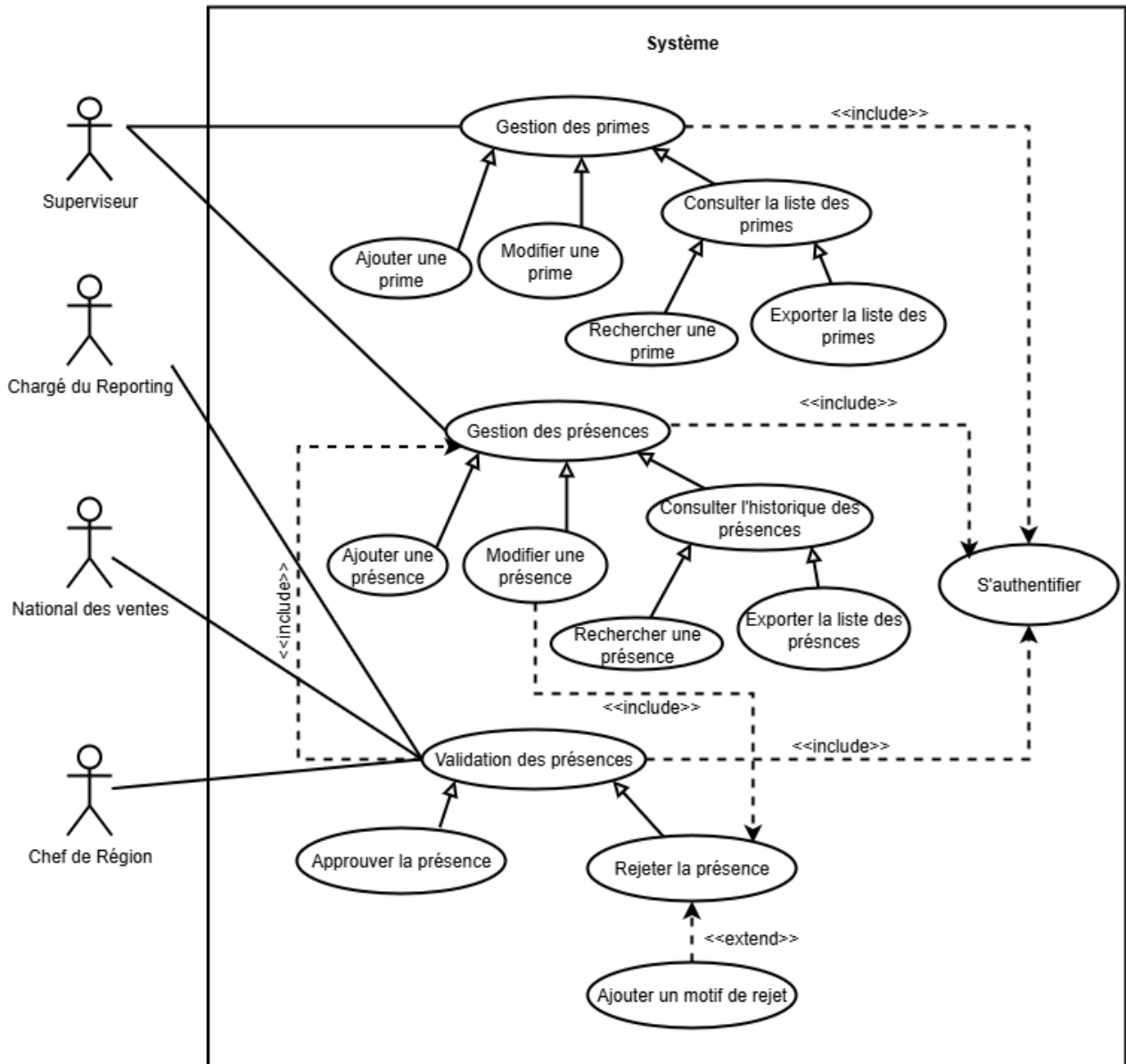


FIGURE 4.2.1 – Diagramme du cas d’utilisation du sprint 4.

4.2.2 Cas d’utilisation ”Approuver / Rejeter une présence”

4.2.2.1 Description textuelle du cas d’utilisation ”Approuver / Rejeter une présence”

La description textuelle du cas d’utilisation ”Approuver / Rejeter une présence ” est détaillée par le tableau 4.3.

Élément	Description
Cas d'utilisation	Approuver / Rejeter une présence.
Acteurs	Chef de région, National des ventes, Chargé du Reporting.
Préconditions	Les acteurs doivent posséder des comptes valides. Les acteurs doivent être authentifiés. La présence doit avoir été préalablement saisie par un superviseur.
Description	Ce cas permet aux acteurs de valider ou rejeter successivement les présences des employés et des flottes d'un canal de chaque dépôt, saisies par le superviseur. Chaque présence suit un circuit de validation défini par un ordre hiérarchique des acteurs définis précédemment. À chaque étape, l'acteur peut soit approuver la présence pour la transmettre au niveau supérieur, soit la rejeter avec justification. En cas de rejet, la présence est renvoyée au superviseur pour une nouvelle correction, avant de reprendre le circuit de validation.
Scénario nominal	<ol style="list-style-type: none"> 1. Le premier acteur (Chef de région) reçoit une notification concernant une présence à valider ; 2. En cliquant sur l'élément "Gérer les validations" de la page d'accueil ou sur "Présences" du menu, il accède à la liste des présences en attente ; 3. L'acteur sélectionne une présence dans la liste des présences en attente ; 4. L'acteur consulte les détails de la présence puis la valide ; 5. La présence est transmise à l'acteur suivant dans le circuit de validation ; 6. Si la présence est conforme, elle est enregistrée, puis transférée au niveau supérieur ; 7. Le processus est répété pour le national de vente jusqu'au chargé du reporting.
Scénario Alternatif	<ol style="list-style-type: none"> 1. Si la présence est non-conforme : 2. L'acteur sélectionne la présence. 3. Il clique sur "Rejeter". 4. La demande est enregistrée comme "rejetée". 4. À chaque étape de validation (chef de région, national des ventes, chargé du reporting), le scénario principal ou alternatif peut-être appliqué.
Post-conditions	À l'étape 5 du scénario nominal : Si toutes les validations sont effectuées avec succès, la présence est définitivement approuvée. Si une présence est rejetée à un niveau, elle n'est pas transmise aux niveaux supérieurs.

TABLE 4.3 – Tableau descriptif du cas d'utilisation "Approuver ou rejeter une présence".

4.2.2.2 Diagramme de séquence du cas d'utilisation "Approuver ou rejeter une présence"

La figure 4.2.2 illustre le diagramme de séquence du cas d'utilisation "Approuver ou rejeter une présence".

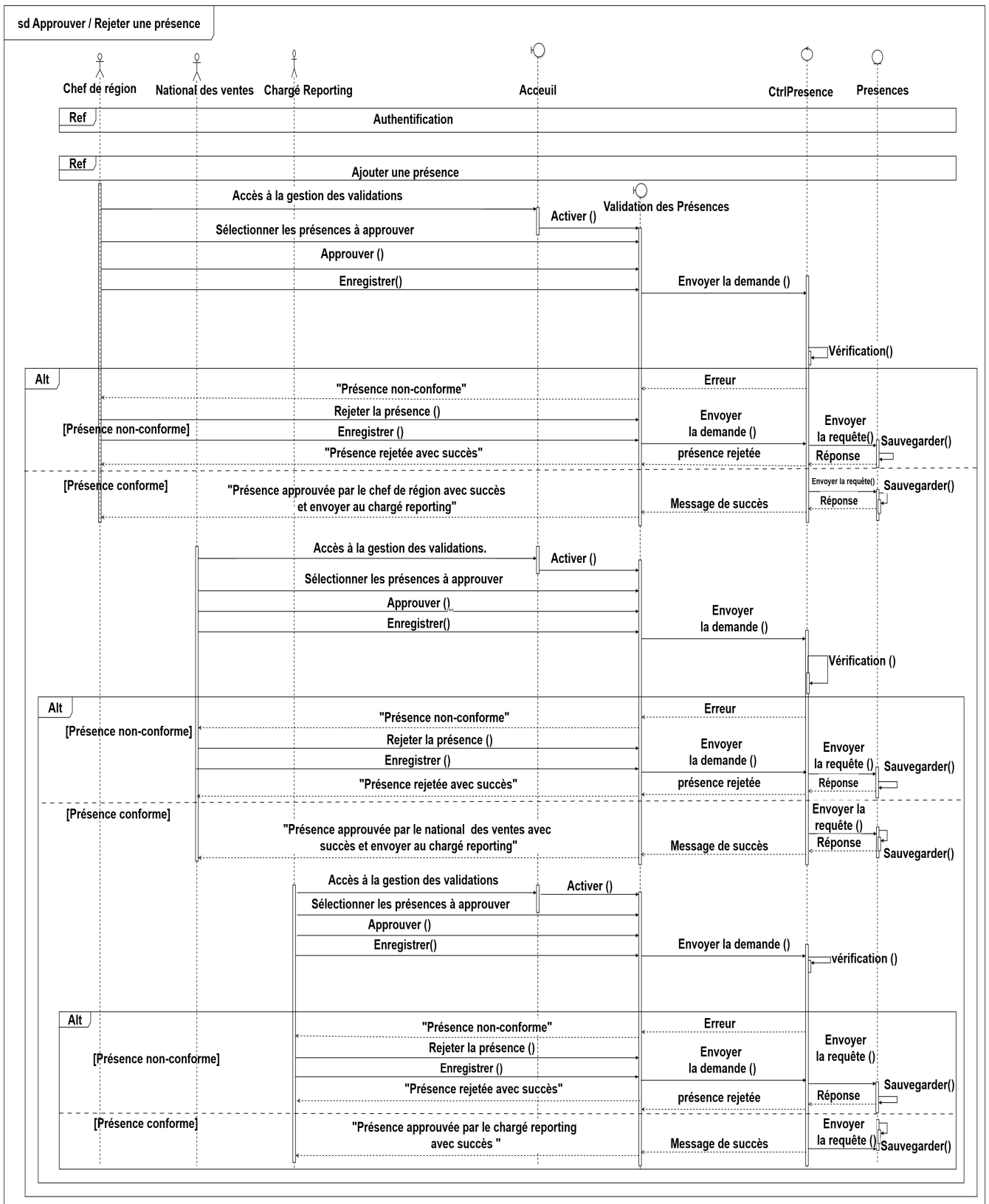


FIGURE 4.2.2 – Diagramme de séquence du cas d’utilisation ”Approuver ou rejeter une présence”.

4.2.3 Diagramme de classe du Sprint 4

La figure 4.2.3 illustre le diagramme de classe du sprint 4.

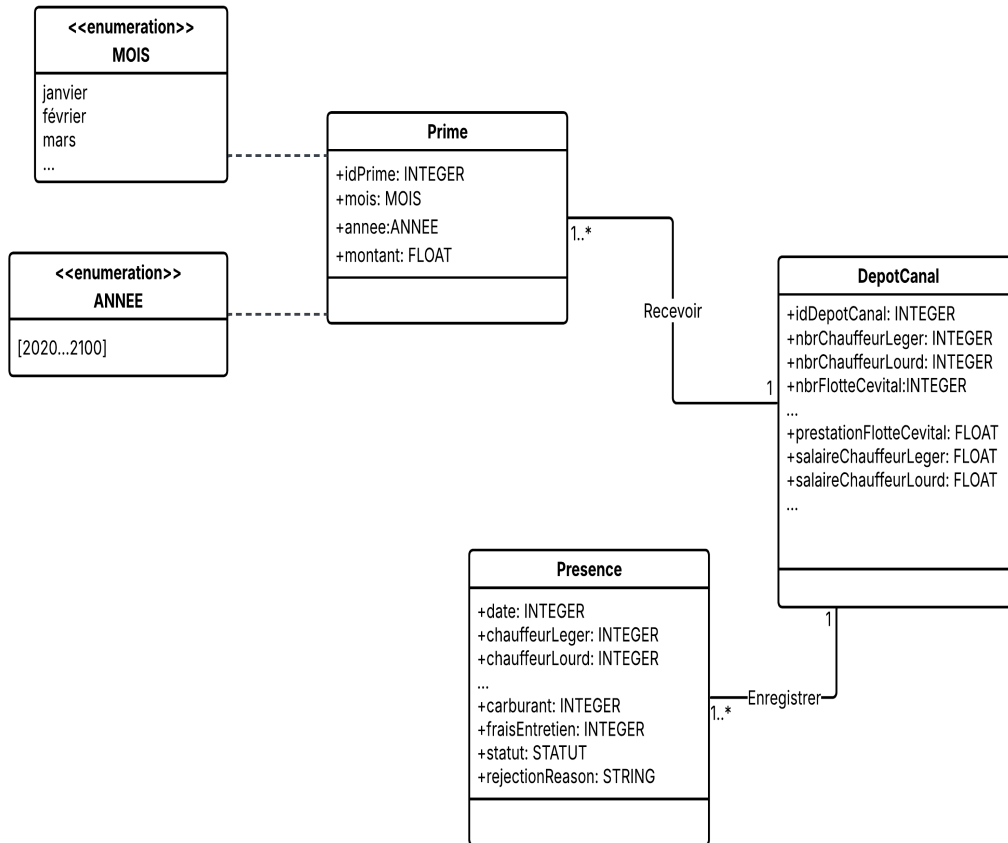


FIGURE 4.2.3 – Diagramme de classe du quatrième sprint.

4.2.4 Diagramme d’activité pour le cas d’utilisation " Gestion du circuit de validation des présences "

Le diagramme d’activité illustré dans la figure 4.2.4 modélise le circuit de validation des présences tel qu’il est implémenté dans notre application. Il traduit, sous forme graphique, le processus métier dans lequel une présence saisie par un superviseur circule à travers plusieurs niveaux de validation hiérarchique : Chef de Région, National des Ventes, Chargé du Reporting.

À chaque étape, l’acteur concerné a la possibilité d’approuver la présence — ce qui permet sa transmission à l’étape suivante — ou de la rejeter, ce qui met immédiatement fin au processus de validation. Dans ce cas, le superviseur est notifié du rejet accompagné de ses motifs.

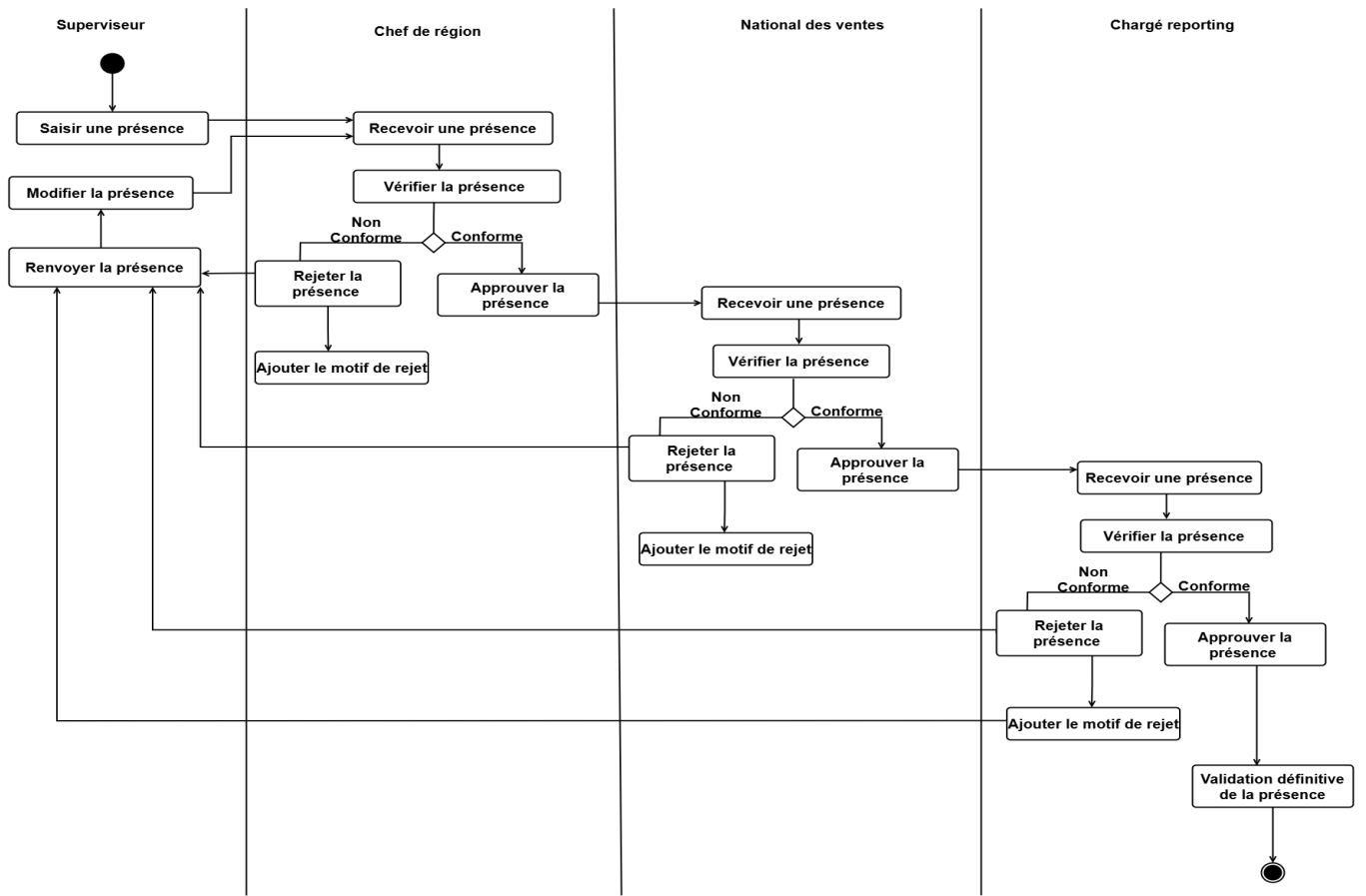


FIGURE 4.2.4 – Diagramme d’activité du cas d’utilisation ”Gestion du circuit de validation des présences”.

4.2.4.1 Espace Superviseur : Interface Primes

Cette interface présentée dans la figure 4.2.5 permet au superviseur d’ajouter les primes pour chaque canal d’un dépôt donné.

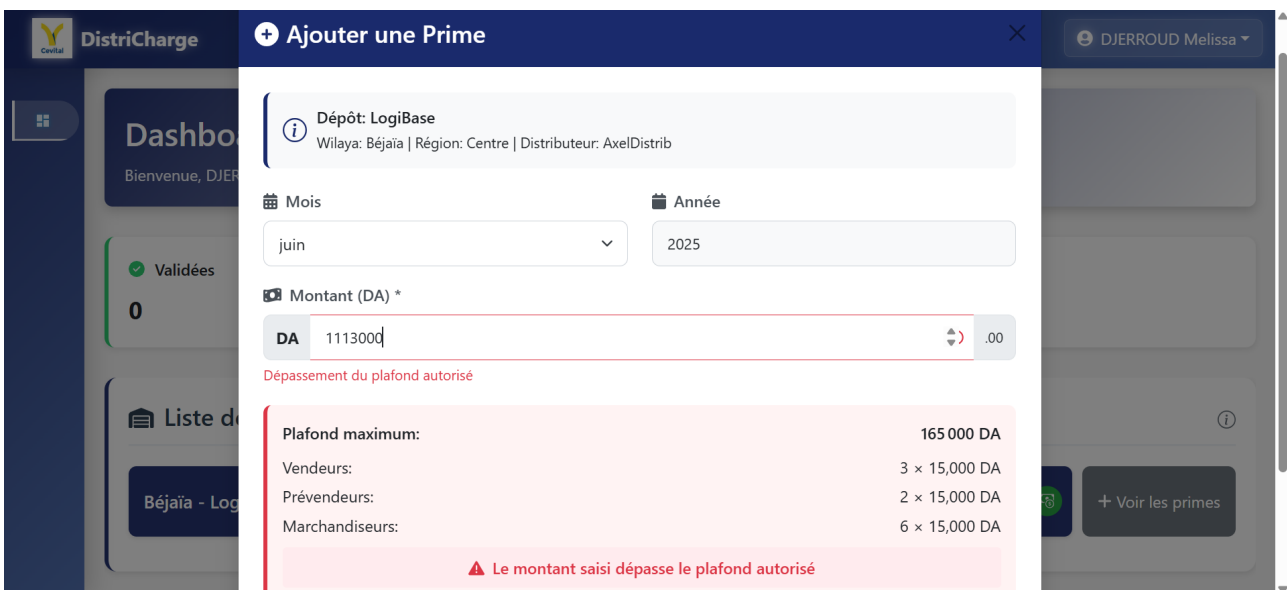


FIGURE 4.2.5 – Interface Primes.

4.2.4.2 Interface de saisie des présences

Cette interface permet au superviseur d'ajouter des présences du personnel et des flottes pour chaque canal d'un dépôt donné.

FIGURE 4.2.6 – Interface de saisie des présences -partie 1-.

FIGURE 4.2.7 – Interface de saisie des présences -partie 2-.

4.2.4.3 Interface de validation des Présences

Cette interface présentée dans la figure 4.2.8 permet aux acteurs (Chef de région , national des ventes et chargé du reporting) d'approuver ou de refuser les présences des personnels et des flottes pour chaque canal d'un dépôt donné que ce soit individuellement ou par sélection.

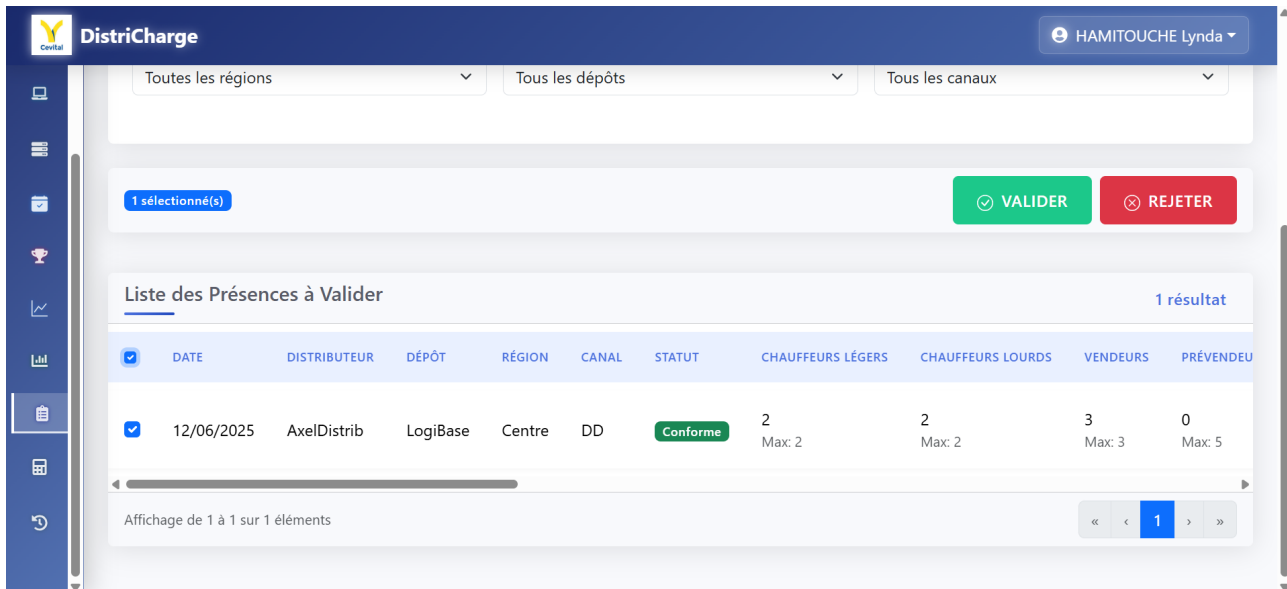


FIGURE 4.2.8 – Interface de validation des Présences.

Conclusion

Dans ce chapitre, nous avons présenté les fonctionnalités développées durant les sprints 3 et 4 du deuxième release. Les diagrammes UML ont permis d'illustrer clairement la logique métier et les processus implémentés, constituant ainsi une base solide pour la suite du projet.

Chapitre 5

Release 3

Introduction

Ce chapitre présente la dernière étape du développement de notre application, qui correspond au troisième release. Elle comprend le sprint 5, durant lequel nous avons terminé les fonctionnalités prévues, en particulier celles liées au calcul des charges ainsi que leur validation finale. Comme dans les chapitres précédents, nous allons montrer pour ce sprint les cas d'utilisation avec leurs descriptions textuelles, les échanges entre les différentes parties de l'application à travers des diagrammes de séquence, ainsi qu'un diagramme de classe. Enfin, des captures d'écran viendront illustrer les interfaces créées pendant cette phase finale.

5.1 Sprint 5 : Fonction de calcul des charges

Ce sprint s'est étalé sur 2 semaines, il comprend 2 items qui sont :

- **Gestion des calculs des charges du Front-Office et du Back-Office** : Cette fonctionnalité permet au Chargé du reporting de calculer et d'enregistrer automatiquement les charges du Front-office et du Back-office pour chaque mois de l'année. Elle offre également la possibilité de consulter l'historique des charges déjà enregistrées.
- **Modification et validation finale des charges** : Cette fonctionnalité permet au Directeur Commercial de modifier le montant final des charges et de clôturer les charges des distributeurs, que ce soit au niveau du front-office ou du back-office, avec la possibilité d'exporter l'historique.

5.1.1 Diagramme du cas d'utilisation du cinquième sprint

Le diagramme du cas d'utilisation dans la figure 5.1.1 illustre les différents cas d'utilisation correspondant aux éléments du cinquième sprint.

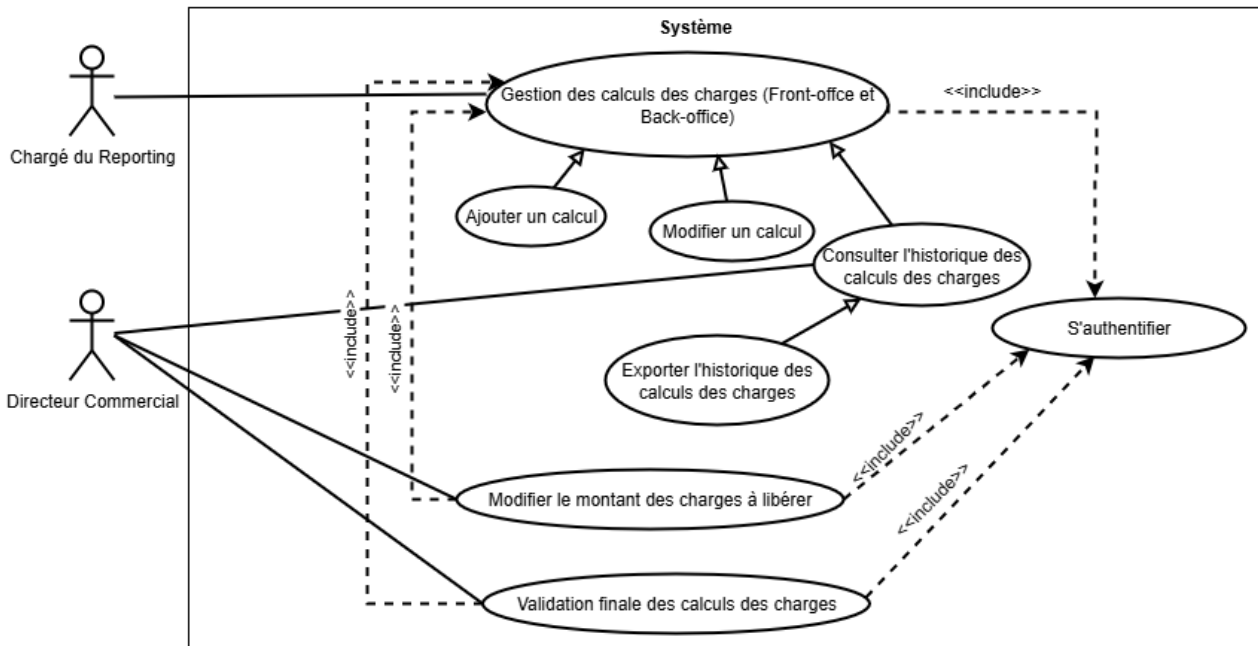


FIGURE 5.1.1 – Diagramme du cas d’utilisation du sprint 5.

5.1.2 Cas d’utilisation ”Gestion des calculs des charges du Front-office et du Back-office”

5.1.2.1 Description textuelle du cas d’utilisation ”Calcul des charges du Front-Office”

La description textuelle du cas d’utilisation ”Calcul des charges du Front-Office” est détaillée par le tableau 5.1.

Élément	Description
Cas d'utilisation	Calcul des charges du Front-Office.
Acteurs	Chargé du reporting.
Description	Ce cas d'utilisation permet au Chargé du reporting de calculer les charges d'un Front-Office pour un canal donné d'un dépôt.
Préconditions	Le Chargé du reporting doit disposer d'un compte valide. Il doit être authentifié. Les informations concernant le Front-Office (les effectifs, les salaires, les prestations, les présences des personnels et des flottes, les primes, les objectifs, les réalisations, les chiffres d'affaires et les jours ouvrables) doivent être déjà renseignées pour le canal du dépôt concerné.
Scénario nominal	<ol style="list-style-type: none"> 1. Le Chargé du reporting clique sur l'option "Calcul des charges du Front-Office" dans le menu de son espace ; 2. Le système le redirige vers l'interface de calcul des charges ; 3. Le Chargé du reporting sélectionne le distributeur, le dépôt, le canal, le mois et l'année ; 4. Le système affiche automatiquement les données du personnel, des flottes et des coûts additionnels avec les montants calculés ; 5. Le Chargé du reporting valide l'enregistrement du calcul ; 6. Le système vérifie les données du formulaire, puis enregistre le nouveau calcul.
Scénario alternatif	À l'étape 5 du scénario nominal : Si un calcul existe déjà pour le mois et l'année sélectionnés, le système affiche un message "Des données existent déjà pour ce front office pour cette période". L'enregistrement écrasera les données existantes.
Postcondition	Le calcul est enregistré avec succès.

TABLE 5.1 – Tableau descriptif du cas d'utilisation "Calcul des charges du Front-Office".

5.1.2.2 Diagramme de séquence du cas d'utilisation "Calcul des charges du Front-Office"

La figure 5.1.2 illustre le diagramme de séquence du cas d'utilisation "Calcul des charges du Front-Office".

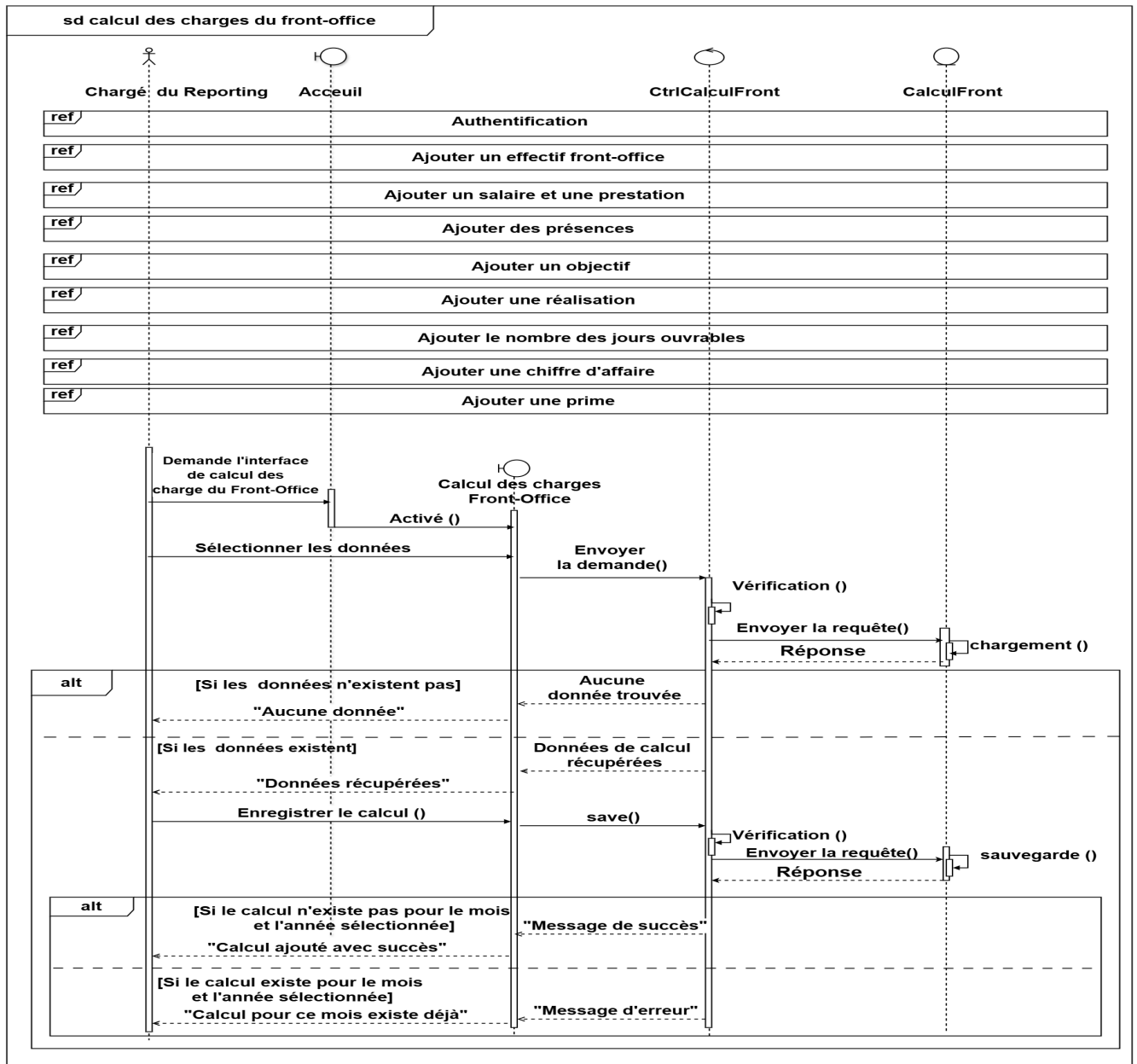


FIGURE 5.1.2 – Diagramme de séquence du cas d’utilisation ”Calcul des charges du Front-Office”.

5.1.3 Cas d’utilisation ”Modifier le montant des charges”

5.1.3.1 Description textuelle du cas d’utilisation ”Modifier un montant des charges Front-Office”

La description textuelle du cas d’utilisation ”Modifier un montant des charges du Front-Office” est détaillée par le tableau 5.2.

Élément	Description
Cas d'utilisation	Modifier un montant des charges du Front-Office.
Acteurs	Directeur Commercial.
Description	Ce cas d'utilisation permet au directeur commercial de modifier le montant des charges du Front-Office à libérer.
Précondition	Le directeur commercial doit posséder un compte valide. Le directeur commercial doit être authentifié. Le montant des charges du Front-Office était déjà calculé pour les données sélectionnées.
Scénario nominal	<ol style="list-style-type: none"> 1. Le directeur commercial clique sur l'élément 'Clôture des charges' du menu pour accéder à la liste des charges ; 2. Le système le redirige vers l'interface des charges ; 3. Le directeur commercial sélectionne une charge dans le tableau ; 4. Le système récupère les informations de la ligne sélectionnée (distributeur, dépôt, canal, montant, etc.) ; 5. Le directeur commercial modifie le montant dans le champ de saisie correspondant ; 6. Le directeur commercial clique sur le bouton modifier ; 7. Le système effectue une vérification puis une mise à jour des données.
Scénario alternatif	À l'étape 6 du scénario nominal : Si le montant saisi n'est pas valide, le système affiche un message d'erreur.
Post condition	Le montant est modifié avec succès.

TABLE 5.2 – Tableau descriptif du cas d'utilisation "Modifier un montant des charges Front-Office".

5.1.3.2 Diagramme de séquence du cas d'utilisation "Modifier un montant des charges du Front-Office"

La figure 5.1.3 illustre le diagramme de séquence du cas d'utilisation "Modifier un montant des charges du Front-Office".

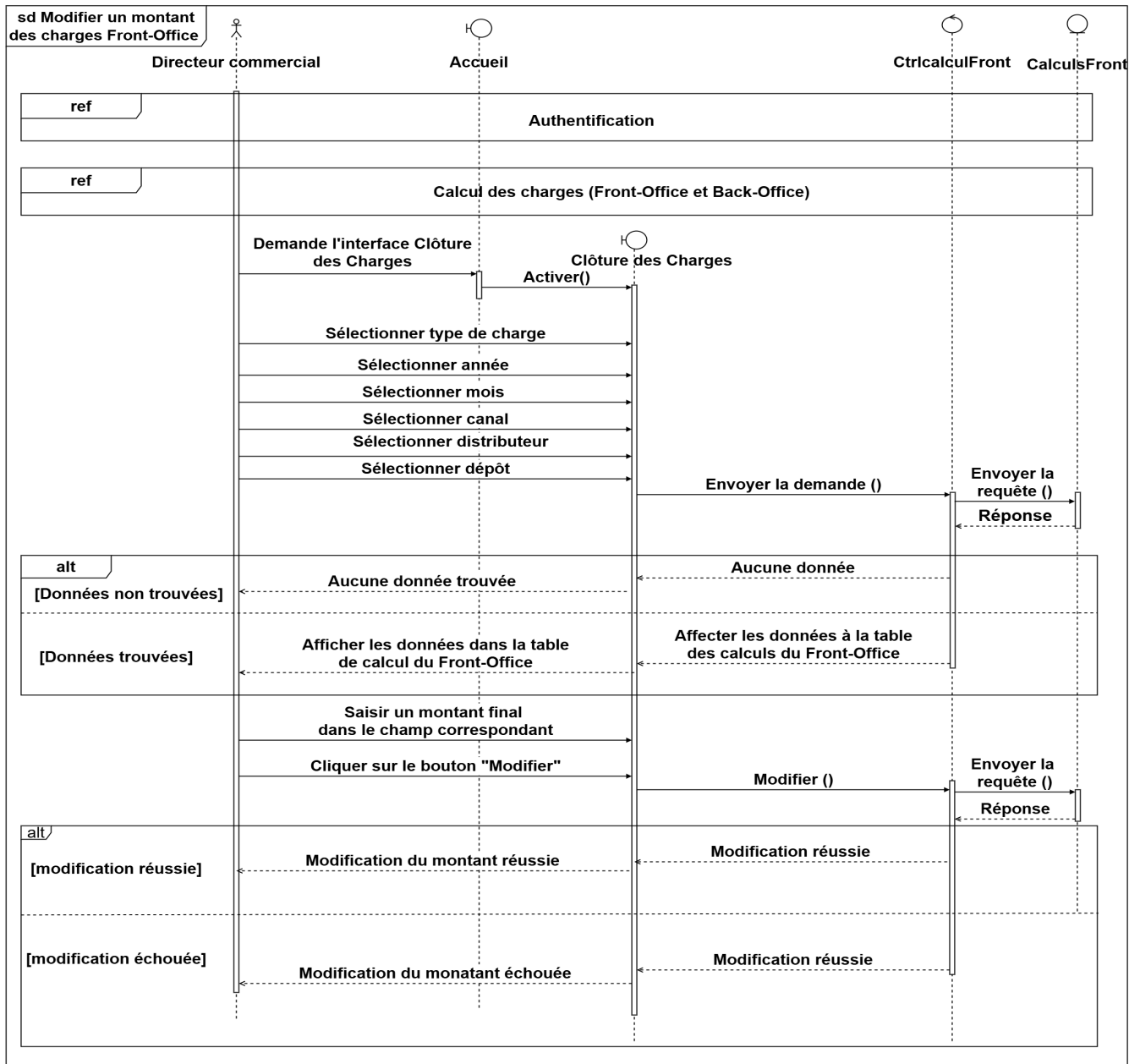


FIGURE 5.1.3 – Diagramme de séquence du cas d’utilisation ”Modifier un montant des charges du Front-Office”.

5.1.4 Diagramme de classe du Sprint 5

La figure 5.1.4 illustre le diagramme de classe du sprint 5.

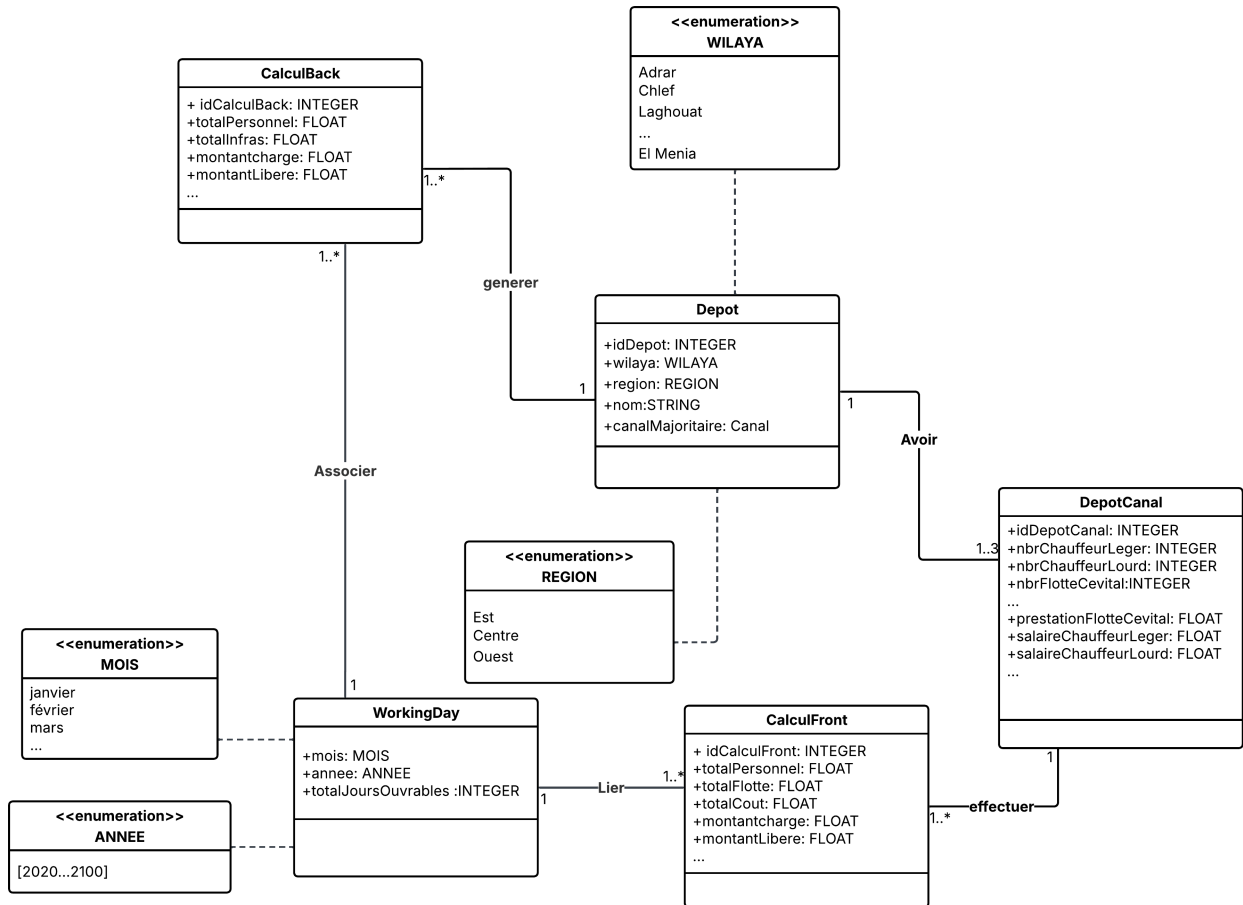


FIGURE 5.1.4 – Diagramme du cas d’utilisation du sprint 5.

5.1.4.1 Interface de calcul des charges du Front-Office

Cette interface permet au chargé du reporting d’ajouter le calcul des charges du Front-office pour chaque mois de l’année par un ajout individuel ou groupé.

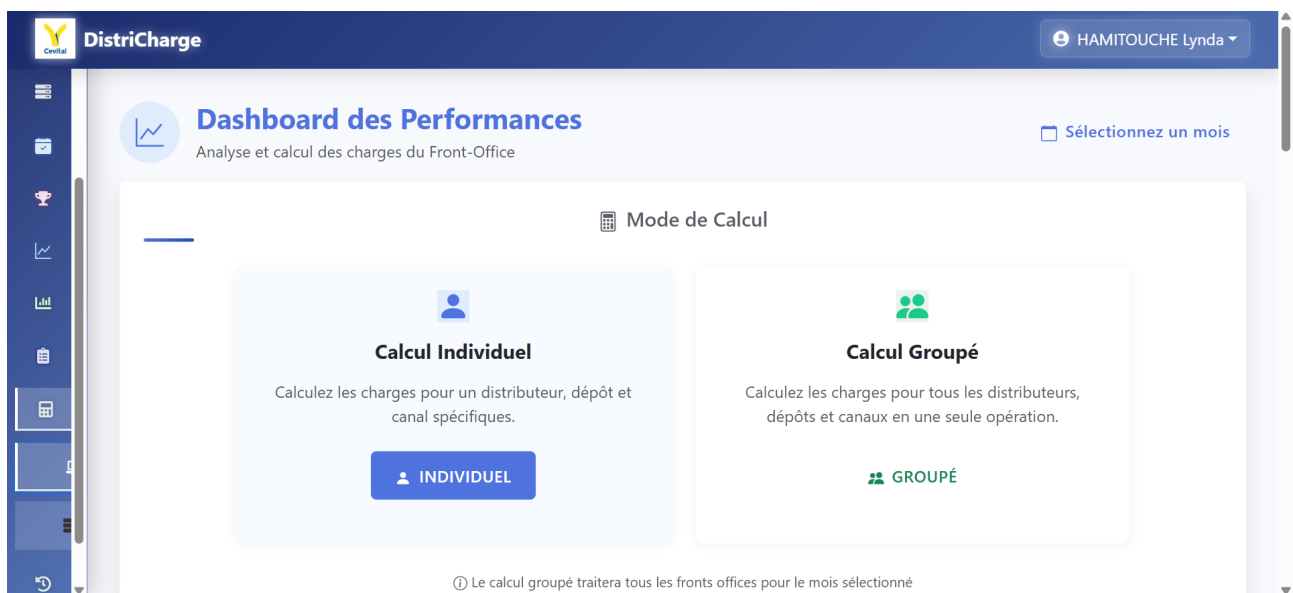


FIGURE 5.1.5 – Interface de calcul des charges du Front-Office -partie 1-.

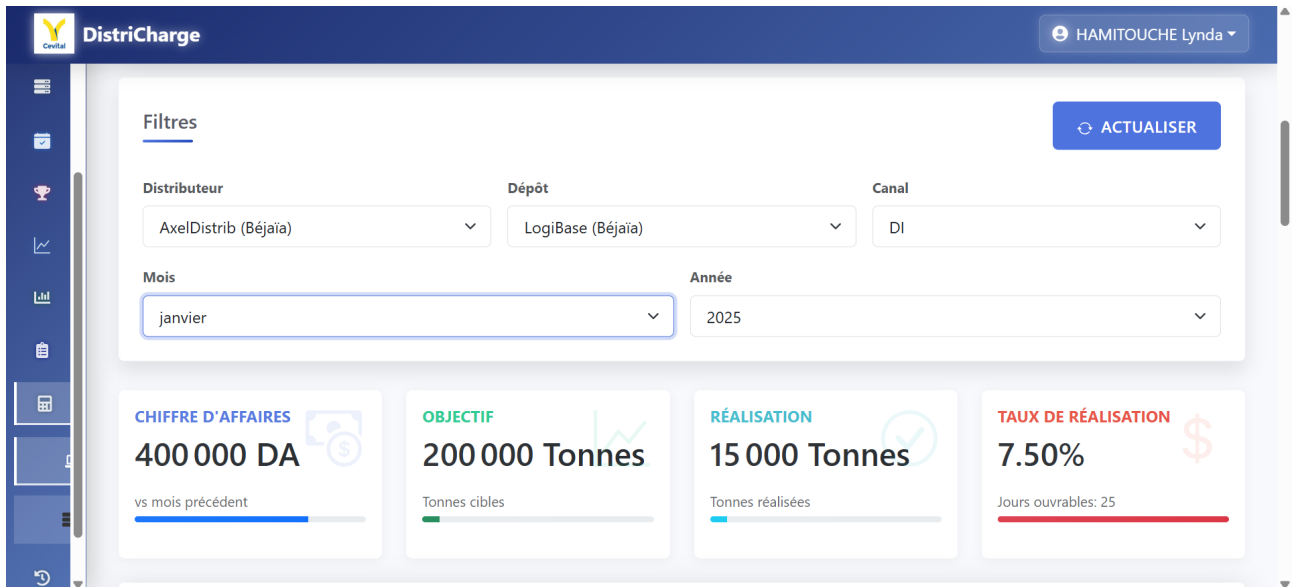


FIGURE 5.1.6 – Interface de calcul des charges du Front-Office -partie 2-.

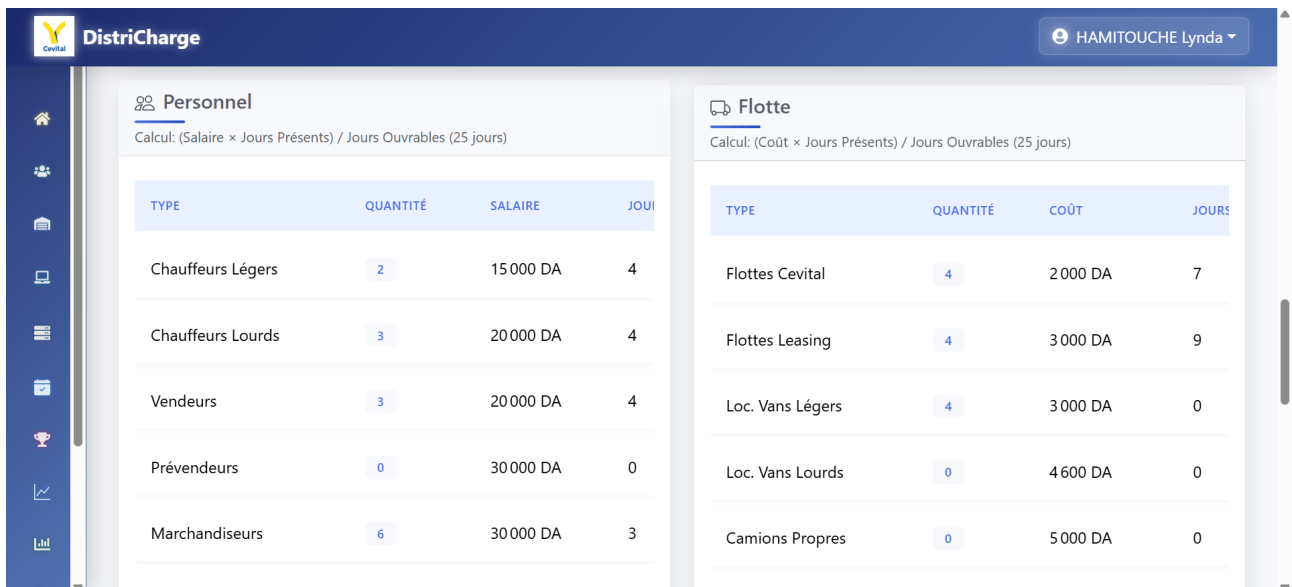


FIGURE 5.1.7 – Interface de calcul des charges du Front-Office -partie 3-.

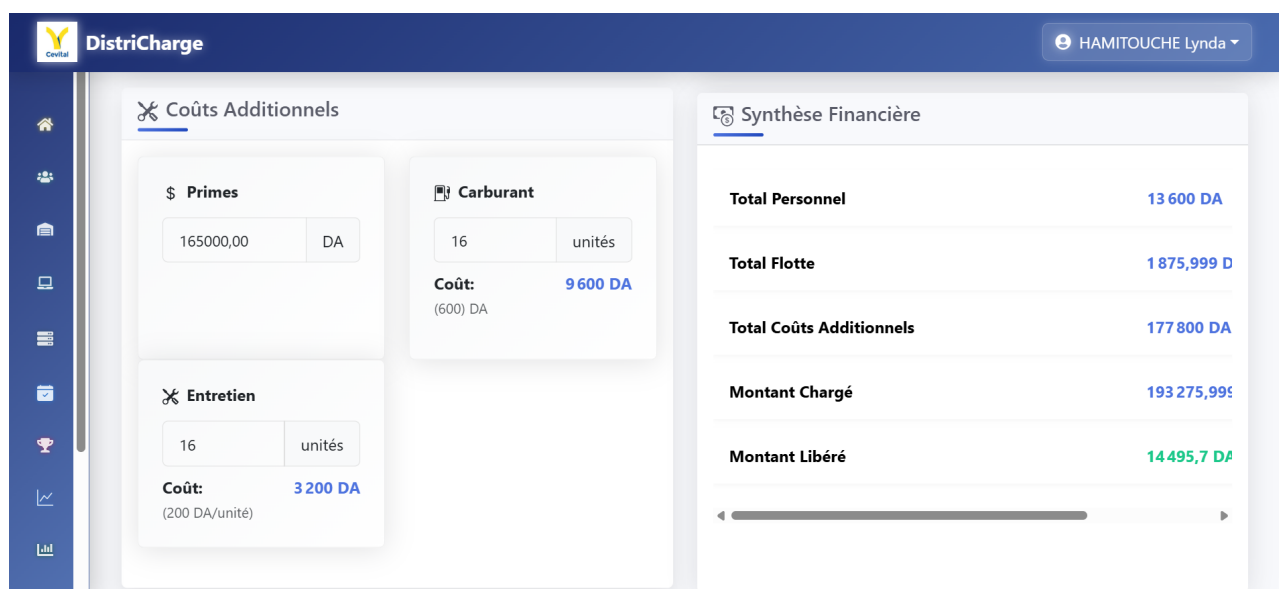


FIGURE 5.1.8 – Interface de calcul des charges du Front-Office -partie 4-.

5.2 Une démonstration de la responsivité ¹

L'une des problématiques majeures identifiées lors de l'analyse du besoin concerne le manque de mobilité pour les superviseurs sur le terrain. En effet, ces derniers sont souvent contraints d'utiliser un ordinateur portable pour accéder au système, ce qui s'avère peu pratique et rend la saisie des informations, notamment le suivi des présences des employés et des flottes, complexe et peu efficace.

Pour répondre à cette contrainte, nous avons conçu une application accessible sur plusieurs types d'appareils : ordinateurs, tablettes et smartphones. Grâce à une interface responsive, l'application s'adapte automatiquement à la taille de l'écran utilisé, offrant ainsi une expérience utilisateur fluide et intuitive, quel que soit le terminal. Cela permet aux superviseurs d'interagir avec le système directement depuis leur smartphone lors de leurs déplacements, sans dépendre d'un poste fixe.

Nous présentons ci-dessous plusieurs captures d'écran illustrant cette adaptabilité, qui démontrent concrètement comment l'interface se réorganise selon le type d'appareil, tout en maintenant une cohérence visuelle et une accessibilité optimale des fonctionnalités.

1. La responsivité (ou design adaptatif) désigne la capacité d'une interface à s'adapter automatiquement aux différentes tailles d'écran.

5.2.1 Téléphone

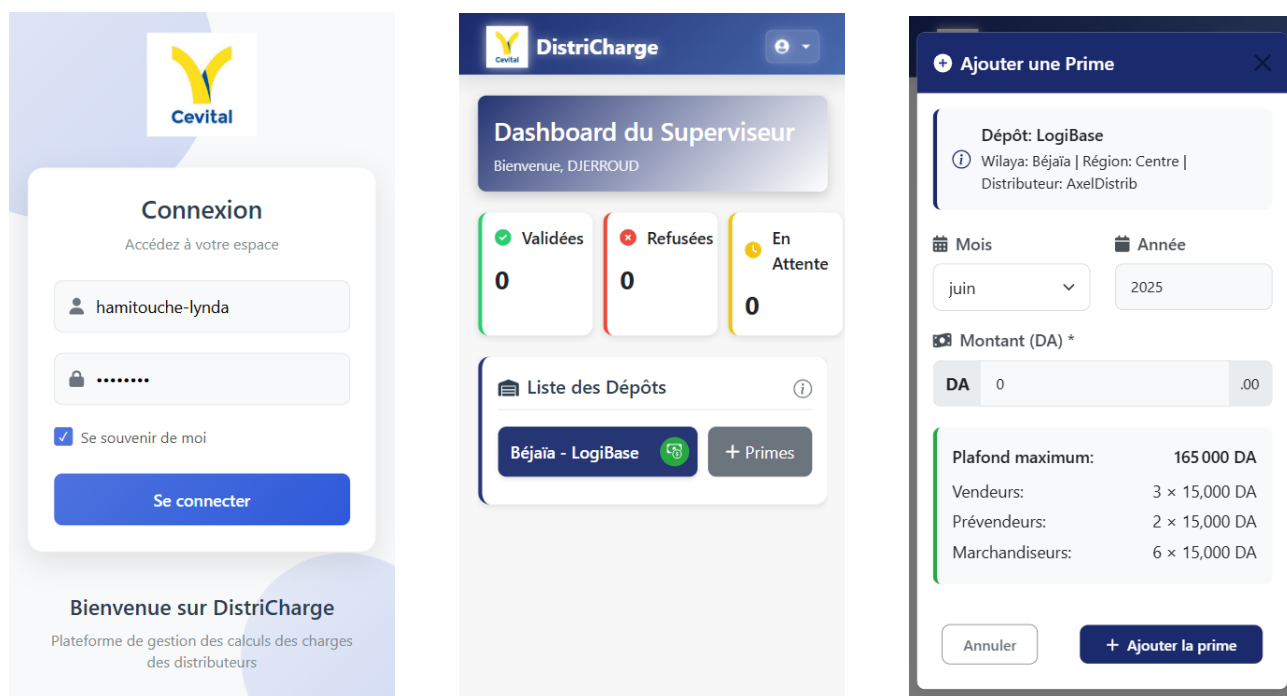


FIGURE 5.2.1 – Vue de l’application sur téléphone

5.2.2 Tablette

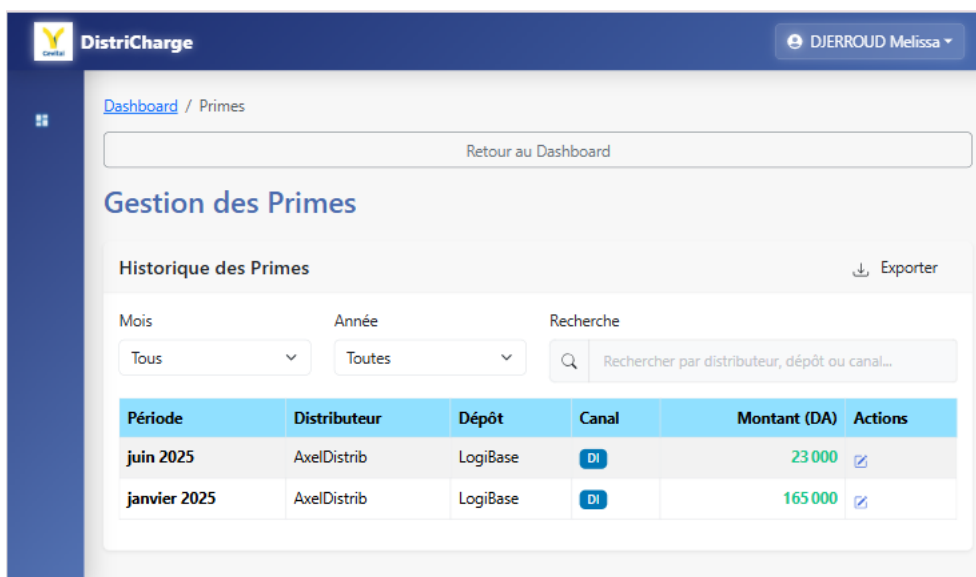


FIGURE 5.2.2 – Vue de l’application sur tablette

5.3 Conclusion

Dans ce chapitre, nous avons présenté la dernière étape du développement de notre application. Il s’agit du troisième release, qui inclut le sprint 5. Durant cette phase, nous avons

finalisé toutes les fonctionnalités prévues, notamment celles liées au calcul des charges. Nous avons montré les cas d'utilisation, les interactions entre les différentes parties de l'application, ainsi que le diagramme de classe. Nous avons aussi ajouté des captures d'écran pour montrer les interfaces réalisées, avec une démonstration de la responsivité des interfaces de l'application. Cette étape marque la fin du développement et confirme que notre application est complète et prête à être utilisée.

Chapitre 6

Conception de la base de données

Introduction

Ce chapitre présente les éléments clés de la modélisation des données de l'application *Dis-triCharge*. Il comprend le diagramme de classe global, représentant les entités et leurs relations, ainsi que les règles de gestion définissant les contraintes métier. Le modèle relationnel traduit cette modélisation en structure de base de données, et le dictionnaire des données détaille les attributs utilisés. Ces éléments assurent la cohérence et la fiabilité du système d'information.

6.1 Règles de gestion

Ci-dessous, les règles de gestion que nous avons suivies pour concevoir le système. Elles définissent les contraintes métier et orientent la structuration des données ainsi que le comportement attendu de notre application.

- Un utilisateur peut-être administrateur, superviseur, chef de région, chargé du reporting, national des ventes ou directeur commercial.
- On a trois canaux : DI, DD, HORECA.
- Un distributeur peut avoir un ou plusieurs dépôts.
- Un dépôt peut-être associé à un ou trois canaux.
- Un dépôt peut avoir un Back-office optionnel.
- Un canal peut-être affecté à un ou plusieurs dépôts.
- Un dépôt peut avoir un seul canal comme canal majoritaire parmi ceux associés au dépôt.
- Un dépôt canal relie un dépôt à un canal spécifique.
- Les utilisateurs (superviseur, chef de région et national des ventes) peut-être affecter à un seul canal.
- Un dépôt canal peut avoir un seul Front-office.
- Un dépôt canal peut avoir un objectif, une réalisation et un chiffre d'affaire pour chaque période (mois et année).

- Un dépôt canal peut-être supervisé par un ou cinq superviseurs.
- Un superviseur est affecté à un ou plusieurs dépôt canal.
- Un dépôt canal peut avoir une ou plusieurs présences.
- Un dépôt canal peut recevoir une ou plusieurs primes mensuelles.
- Un dépôt canal peut avoir un ou plusieurs calculs de charge mensuel du Front-Office.
- Un dépôt canal peut avoir un ou plusieurs calculs des charges Back-office.

6.2 Diagramme de classe

Le diagramme de classes est un élément fondamental de la modélisation en génie logiciel. Il décrit les classes du système, leurs attributs, leurs méthodes et les relations entre elles. Ci-dessous, le diagramme de classes global de notre application qui offre une vue d'ensemble de sa structure.

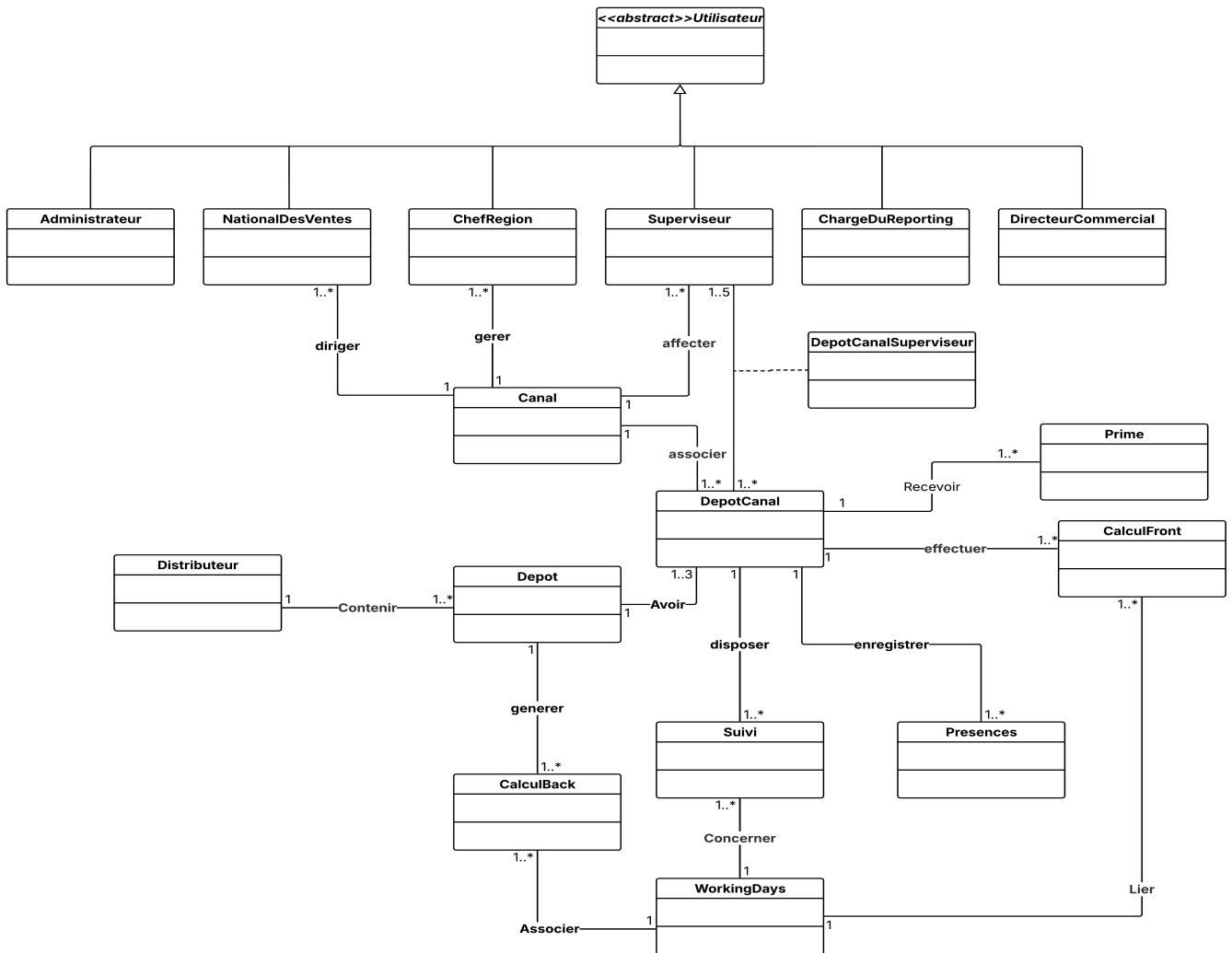


FIGURE 6.2.1 – Diagramme de classe global.

6.3 Le modèle relationnel

Pour mettre en place la base de données, nous avons transformé le modèle conceptuel (représenté sous forme de diagrammes de classes UML) en un modèle logique. Cette transformation suit des règles précises pour adapter les classes et leurs relations en tables relationnelles. En appliquant ces règles pour chaque sprint, nous avons obtenu le schéma relationnel final qui servira de base à l'implémentation de notre base de données.

- **Utilisateur** (idUser, nom, prenom, badge, region, username, password, typeUser)
- **Administrateur** (#idUser)
- **Superviseur** (#idUser, #idCanal)
- **ChefRegion**(#idUser, #idCanal)
- **NationalDesVentes**(#idUser, #idCanal)
- **ChargeReporting**(#idUser)
- **DirecteurCommercial** (#idUser)
- **Distributeur** (idDistrib, raisonSoc, adresse, abreviation, numregistre, wilaya, numtel).
- **Canal** (idCanal, nom).
- **Depot** (idDepot, wilaya, region, nom, canalMajoritaire, nbrGerant, nbrAdv, nbrMagasinier, nbrCaissier, nbrComptable, nbrManutentionnaire, nbrGardien, nbrFemmeMenage, cnas, assurance, bureautique, communication, amortissementChariot, amortissementChambreFroide, location, salGerant, salAdv, salMagasinier, salCaissier, salComptable, salManutentionnaire, salGardien, salFemmeMenage, chargeCnas, chargeAssurance, chargeBureautique, chargeCommunication, chargeAmortissementChariot, chargeAmortissementChambreFroide, chargeLocation).
- **DepotCanal** (idDepotCanal, nbrChauffeursLegers, nbrChauffeursLourds, nbrVendeurs, nbrPrévendeurs, nbrMarchandiseurs, nbrLivres, nbrTéléopérateurs, nbrFlottesCevitale, nbrFlottesLeasing, nbrLocationsVansLegers, nbrLocationsVansLourds, nbrCamionsPropres, nbrLocationsCamionsFrigorifiques, nbrLocationsVehicule, nbrAmortissementScooters, salChauffeursLegers, salChauffeursLourds, salVendeurs, salPrevendeurs, salMarchandiseurs, salLivres, presFlottesCevitale, presFlottesLeasing, presLocationsVansLegers, presLocationsVansLourds, presCamionsPropres, presLocationsCamionsFrigorifiques, presLocationsVehicule, presAmortissementScooters, primes, carburant, entretien teleopérateurs, #idDepot, #idCanal).
- **DepotCanalSuperviseur** (#idDepotCanal, #idSuperviseur).
- **Suivi** (idSuivi, objectif, chiffreAffaire, realisation, tauxRealisation, #idDepotCanal, #mois, #annee).
- **Workingday** (mois, annee, totalJoursOuvrables).
- **Prime** (idPrime, mois, annee, montant, #idDepotCanal).

- **Presence** (date, chauffeursLegers, chauffeursLourds, vendeurs, prévendeurs, marchandiseurs, livreurs, téléopérateurs, flottesCevitale, flottesLeasing, locationsVansLegers, locationsVansLourds, camionsPropres, locationsCamionsFrigorifiques, locationsVehicule, amortissementScooters, status, rejectionReason, #idDepotCanal).
- **CalculFront** (idCalculFront, totalPersonnel, totalFlotte, totalCoûts, chargeTotale, montantCharge, montantLibere, montantFinal, chiffreAffaire, objectif, realisation, tauxRealisation, chauffeursLegers, chauffeursLourds, vendeurs, prevendeurs, marchandiseurs, livreurs, teleoperateurs, flottesCevitale, flottesLeasing, locationsVansLegers, locationsVansLourds, camionsPropres, locationsCamionsFrigorifiques, locationsVehicule, amortissementScooters, primes, carburant, entretien, statut, #mois, #annee, #idDepotCanal).
- **CalculBack** (idCalculBack, gerant, adv, magasinier, caissier, comptable, manutentionnaire, gardien, femmeMenage, cnas, assurance, bureautique, communication, amortissementChariot, amortissementChambreFroide, location, totalPersonnel, totalInfrastructure, chargeTotale, montantCharge, montantLibere, montantFinal, tauxRealisation, statut, #mois, #annee, #idDepot).

6.4 Dictionnaire des données

Le tableau 6.1 ci-dessous présente le dictionnaire de données correspondant au diagramme de classes illustré précédemment.

Classe	Attribut	Description	Type
Utilisateur	idUser	Identifiant de l'utilisateur	Integer
	nom	Nom de l'utilisateur	String
	prenom	Prénom de l'utilisateur	String
	badge	Numéro de badge	Integer
	region	Région assignée	Enum [Est, Centre, Ouest]
	username	Nom d'utilisateur	String
	password	Mot de passe	String
	typeUser	Type utilisateur	String
Administrateur	idAdmin	Identifiant d'administrateur	Integer
Superviseur	idSuperviseur	Identifiant du superviseur	Integer

Classe	Attribut	Description	Type
ChefRegion	idChef	Identifiant du chef de région	Integer
NationalDesVentes	idNational	Identifiant du national des ventes	Integer
ChargeReporting	idCharge	Identifiant du chargé du reporting	Integer
DirecteurCommercial	idDirecteur	Identifiant du directeur commercial	Integer
Distributeur	idDistrib	Identifiant du distributeur	Integer
	raisonSoc	Raison sociale	String
	adresse	Adresse	String
	abreviation	Abréviation	String
	numregistre	Numéro de registre	Integer
	wilaya	Wilaya	Enum [Adrar,...]
	numtel	numéro de téléphone	String
Canal	idCanal	Identifiant du canal	Integer
	nom	Nom du canal	Enum [DI, DD, HORECA]
Depot	idDepot	Identifiant du dépôt	Integer
	wilaya	Wilaya	Enum [Adrar,...]
	region	Région	Enum [Est, Centre, Ouest]
	nom	Nom du dépôt	String
	canalMajoritaire	Canal majoritaire	Enum[DI, DD, HORECA]
	nbrGerant	Nombre de gérants	Integer
	nbrAdv	Nombre d'administrateur des ventes	Integer
	nbrMagasinier	Nombre de magasiniers	Integer
	nbrCaissier	Nombre de caissiers	Integer

Classe	Attribut	Description	Type
	nbrComptable	Nombre de comptables	Integer
	nbrManutentionnaire	Nombre de manutentionnaires	Integer
	nbrGardien	Nombre de gardiens	Integer
	nbrFemmeMenage	Nombre de femmes de ménage	Integer
	cnas	Cotisation CNAS	Integer
	assurance	Assurance	Integer
	bureautique	bureautique	Integer
	communication	communication	Integer
	amortissementChariot	Amortissement chariot	Integer
	amortissementChambreFroide	Amortissement de la chambre froide	Integer
	location	location	Integer
	salGerant	Salaire du gérant	Float
	salAdv	Salaire du ADV	Float
	salMagasinier	Salaire du magasinier	Float
	salCaissier	Salaire du caissier	Float
	salComptable	Salaire du comptable	Float
	salManutentionnaire	Salaire du manutentionnaire	Float
	salGardien	Salaire du gardien	Float
	salFemmeMenage	Salaire de la femme de ménage	Float
	chargeCnas	Charges de la CNAS	Float
	chargeAssurance	Charges de l'assurance	Float

Classe	Attribut	Description	Type
	chargeBureautique	Charges du bureau-tique	Float
	chargeCommunication	Charges de commu-nication	Float
	chargeAmortissement Chariot	Charges d'amortissement des chariot	Float
	chargeAmortissement ChambreFroide	Charges d'amor-tissement de la chambre froide	Float
	chargeLocation	Charges de location	Float
DepotCanal	idDepotCanal	Identifiant du depot-canal	Integer
	nbrChauffeursLegers	Nombre de chauffeurs légers	Integer
	nbrChauffeursLourds	Nombre de chauffeurs lourds	Integer
	nbrVendeurs	Nombre de vendeurs	Integer
	nbrPrévendeurs	Nombre de préven-deurs	Integer
	nbrTeleopérateurs	Nombre des télé-opérateurs	Integer
	nbrMarchandiseurs	Nombre de mar-chandiseurs	Integer
	nbrLivreurs	Nombre de livreurs	Integer
	nbrTéléopérateurs	Nombre de téléopé-rateurs	Integer
	nbrFlottesCevitale	Nombre de flottes Cevitale	Integer
	nbrFlottesLeasing	Nombre de flottes leasing	Integer
	nbrLocationsVansLegers	Nombre de vans lé-gers loués	Integer

Classe	Attribut	Description	Type
	nbrLocationsVansLourds	Nombre de vans lourds loués	Integer
	nbrCamionsPropres	Nombre de camions propres	Integer
	nbrLocationsCamionsFrigorifiques	Nombre de camions frigorifiques loués	Integer
	nbrLocationsVehicule	Nombre de véhicules loués	Integer
	nbrAmortissementScooter	Nombre d'amortissement des scooters	Integer
	salChauffeursLegers	Salaire des chauffeurs légers	Float
	salChauffeursLourds	Salaire des chauffeurs lourds	Float
	salVendeurs	Salaire des vendeurs	Float
	salPrevendeurs	Salaire des prévendeurs	Float
	salMarchandiseurs	Salaire des marchandiseurs	Float
	salLivreurs	Salaire des livreurs	Float
	presFlottesCevitale	Prestation des flottes Cevitale	Float
	presFlottesLeasing	Prestation des flottes leasing	Float
	presLocationsVansLegers	Prestation des vans légers loués	Float
	presLocationsVansLourds	Prestation des vans lourds loués	Float
	presCamionsPropres	Prestation des camions propres	Float
	presLocations-CamionsFrigorifiques	Prestation des camions frigorifiques loués	Float

Classe	Attribut	Description	Type
	presLocationsVehicule	Prestation des véhicules loués	Float
	presAmortissementScooter	Prestation des amortissement scooters	Float
	primes	Montant des primes	Float
	carburant	Coût du carburant	Float
	entretien	Frais d'entretien	Float
Suivi	idSuivi	Identifiant	Integer
	objectif	Objectif fixé	Float
	chiffreAffaire	Chiffre d'affaires	Float
	realisation	Réalisation	Float
	tauxRealisation	Taux de réalisation	Float
Workingday	mois	Mois	Enum [janvier, février...]
	annee	Année	Enum[2020...2100]
	totalJoursOuvrables	Nombre de jours ouvrables	Integer
Prime	idPrime	Identifiant	Integer
	mois	Mois	Enum [janvier, février...]
	annee	Année	Enum[2020...2100]
	montant	Montant de la prime	Float
Presence	date	Date	Date
	chauffeursLegers	Présence des chauffeurs légers	Integer
	chauffeursLourds	Présence des chauffeurs lourds	Integer
	vendeurs	Présence des vendeurs	Integer
	prévendeurs	Présence des pré-vendeurs	Integer

Classe	Attribut	Description	Type
	marchandiseurs	Présence des marchandiseurs	Integer
	livreurs	Présence des livreurs	Integer
	téléopérateurs	Présence des téléopérateurs	Integer
	flottesCevitale	Présence des flottes Cevitale	Integer
	flottesLeasing	Présence des flottes leasing	Integer
	locationsVansLegers	Présence des vans légers loués	Integer
	locationsVansLourds	Présence des vans lourds loués	Integer
	camionsPropres	Présence des camions propres	Integer
	locationsCamionsFrigorifiques	Présence des camions frigorifiques loués	Integer
	locationsVehicule	Présence des véhicules loués	Integer
	amortissementScooters	Présence des amortissement scooters	Integer
	status	Statut	Enum[pending, rejected,...]
	rejectionReason	Raison du rejet	String
CalculFront	idCalculFront	Identifiant du calcul Front	Integer
	totalPersonnel	Total des charges du personnel	Float
	totalFlotte	Total des charges des flottes	Float
	totalCouts	Total des coûts	Float
	chargeTotale	Charge totale	Float

Classe	Attribut	Description	Type
	montantCharge	Montant des charges	Float
	montantLibere	Montant libéré	Float
	montantFinal	Montant final	Float
	chiffreAffaire	Chiffre d'affaire	Float
	objectif	Objectif	Float
	realisation	Réalisation	Float
	tauxRealisation	Taux de réalisation	Float
	chauffeursLegers	charge des chauffeurs légers	Float
	chauffeursLourds	charge des chauffeurs lourds	Float
	vendeurs	charge des vendeurs	Float
	prevendeurs	charge des prévendeurs	Float
	marchandiseurs	charge des marchandiseurs	Float
	livreurs	charge des livreurs	Float
	teleopérateurs	charge des téléopérateurs	Float
	flottesCevitale	charge des flottes Cevitale	Float
	flottesLeasing	charge des flottes leasing	Float
	locationsVansLegers	charge de location vans légers	Float
	locationsVansLourds	charges de location de vans lourds	Float
	camionsPropres	charges des camions propres	Float

Classe	Attribut	Description	Type
	locationsCamions Frigorifiques	charges de locations camions frigorifiques	Float
	locationsVehicule	charges de locations véhicules	Float
	amortissementScooters	charges des amortissement scooters	Float
	primes	Primes	Float
	carburant	frais totaux du carburant	Float
	entretien	frais totaux d'entretien	Float
	statut	Statut du calcul	Enum [pending, cloture...]
CalculBack	idCalculBack	Identifiant	Integer
	gerant	Charges des gérants	Float
	adv	charges d'ADV	Float
	magasinier	Charges des magasiniers	Float
	caissier	Charges des caissiers	Float
	comptable	Charges des comptables	Float
	manutentionnaire	Charges des manutentionnaires	Float
	gardien	Charges des gardiens	Float
	femmeMenage	Charges des femmes de ménage	Float
	cnas	Coût de CNAS	Float
	assurance	Coût d'assurance	Float

Classe	Attribut	Description	Type
	bureautique	Coût de la bureautique	Float
	communication	Coût de communication	Float
	amortissementChariot	Coût d'amortissement des chariots	Float
	amortissementChambreFroide	Coût d'amortissement de la chambre froide	Float
	location	Coût de location	Float
	totalPersonnel	Total du personnel	Float
	totalInfrastructure	Total d'infrastructure	Float
	chargeTotale	Charge totale	Float
	montantCharge	Montant des charges	Float
	montantLibere	Montant libéré	Float
	montantFinal	Montant final	Float
	tauxRealisation	Taux de réalisation	Float
	statut	Statut du calcul	Enum [pending, cloture...]

TABLE 6.1 – Dictionnaire des données.

6.5 Conclusion

L'élaboration des règles de gestion, du diagramme de classes, du dictionnaire de données et du modèle relationnel a constitué une étape clé dans la conception de notre application. Ces éléments ont permis d'assurer une modélisation rigoureuse du système, en alignant parfaitement les aspects fonctionnels et techniques avec les exigences exprimées, tout en garantissant la cohérence et la fiabilité du système.

Conclusion Générale et Perspectives

CE projet de fin d'études, réalisé au sein de l'entreprise Cevital, nous a permis de mettre en pratique les connaissances théoriques acquises au cours de notre parcours universitaire, tout en développant de nouvelles compétences à la fois techniques et méthodologiques.

Le travail accompli durant notre stage s'inscrit dans le cadre de la conception et du développement d'une application web destinée au calcul des charges des distributeurs, dont le cas d'étude est le service commercial de Cevital. Cette application intègre plusieurs fonctionnalités essentielles, telles que la gestion des comptes utilisateurs selon les rôles définis, la gestion des distributeurs et de leurs dépôts, l'enregistrement des effectifs (front-office et back-office), le suivi des présences, ainsi que la gestion des chiffres d'affaires, des objectifs, des réalisations et des charges.

L'aspect central de l'application repose sur le calcul automatisé des charges, basé sur les données mensuelles recueillies (effectifs, salaires, prestations, jours ouvrables, etc.). Le système permet de générer, pour chaque distributeur, un bilan détaillé des coûts du front-office et du back-office, avec la possibilité de consulter l'historique des calculs et d'effectuer des modifications validées par les responsables hiérarchiques. Ce processus vise à offrir une vision claire et précise des dépenses engagées, tout en assurant la fiabilité et la traçabilité des résultats. Il constitue ainsi un outil d'aide à la décision pour le service commercial.

Nous avons réussi à concevoir cette application en suivant la méthodologie Agile Scrum, largement reconnue dans le domaine du développement logiciel, ce qui nous a permis de bien organiser notre travail et de faire avancer le projet de manière itérative. De plus, nous avons utilisé le langage UML pour modéliser les différents aspects et fonctionnalités de l'application.

Nous avons également eu recours à divers outils et langages de développement pour mener à bien ce projet. Pour la partie frontend, nous avons utilisé React, une bibliothèque JavaScript moderne, afin de concevoir une interface utilisateur dynamique et réactive. Pour le backend, nous avons opté pour Node.js avec le framework Express.js, afin de gérer les différentes fonctionnalités côté serveur. Enfin, la base de données relationnelle a été assurée par MySQL.

Le développement de l'application DistriCharge a été une expérience profondément enrichissante, tant sur le plan technique que personnel. Ce projet nous a permis de consolider nos compétences en conception, en programmation, ainsi qu'en gestion de projet.

Par ailleurs, plusieurs perspectives d'amélioration peuvent être envisagées pour faire évoluer l'application DistriCharge et mieux répondre aux besoins du service commercial. Parmi celles-ci, on peut envisager :

- L'ajout d'un système de notifications pour informer les superviseurs des nouvelles affectations ou modifications ;
- L'intégration d'une messagerie interne pour faciliter la communication entre les différents acteurs (superviseurs, chefs de zone, etc.) ;
- L'archivage et l'historique des affectations et des changements effectués dans la plateforme.
- L'intégration avec d'autres systèmes internes (exemple : gestion des stocks ou des ventes).

Ces perspectives ouvrent la voie à une évolution continue de l'application, en renforçant son utilité au sein du service commercial de Cevital, DistriCharge vise à améliorer l'efficacité des processus et la coordination entre les différents acteurs impliqués.

Bibliographie

- [1] Wikipedia contributors. Cevital — wikipédia, 2024.
- [2] Cevital. L'histoire du groupe, 2025.
- [3] Roma. Organigramme cevital. <https://fr.scribd.com/document/730676325/organigramme-cevital>, 2025. Consulté le 5 mai 2025.
- [4] Red Hat. Qu'est-ce que la méthodologie agile ?, 2025.
- [5] Tuleap. Les grands principes scrum, 2025.
- [6] Ken Schwaber and Jeff Sutherland. *Le Guide Scrum – La Règle du jeu de Scrum*. Scrum.org, 2020.
- [7] IONOS Digital Guide. Uml : Un langage de modélisation pour la programmation orientée objet, 2025.
- [8] Techno-Science.net. Modèle-vue-contrôleur - définition, 2025.
- [9] Laurent Audibert. Cours uml - diagramme de cas d'utilisation, 2025. Consulté le 2 mars 2025, section 2-2-1.
- [10] Atlassian. User stories, 2025.
- [11] React – a javascript library for building user interfaces. <https://reactjs.org>, 2024. Consulté en 2025.
- [12] React native – create native apps for android and ios using react. <https://reactnative.dev>, 2024. Consulté en 2025.
- [13] Tailwind css – rapidly build modern websites without ever leaving your html. <https://tailwindcss.com>, 2024. Consulté en 2025.
- [14] Mozilla Developer Network. Javascript guide – mdn web docs. <https://developer.mozilla.org/en-US/docs/Web/JavaScript>, 2024. Consulté en 2025.
- [15] Node.js – javascript runtime built on chrome's v8 javascript engine. <https://nodejs.org>, 2024. Consulté en 2025.

- [16] Express – fast, unopinionated, minimalist web framework for node.js. <https://expressjs.com>, 2024. Consulté en 2025.
- [17] Mysql : : The world’s most popular open source database. <https://www.mysql.com>, 2024. Consulté en 2025.
- [18] Sequelize – node.js orm for postgres, mysql, mariadb, sqlite and microsoft sql server. <https://sequelize.org>, 2024. Consulté en 2025.
- [19] Jwt.io – json web tokens. <https://jwt.io>, 2024. Consulté en 2025.
- [20] Visual studio code – code editing. redefined. <https://code.visualstudio.com>, 2024. Consulté en 2025.
- [21] Postman api platform. <https://www.postman.com>, 2024. Consulté en 2025.
- [22] Github – where the world builds software. <https://github.com>, 2024. Consulté en 2025.
- [23] Lucidchart – intelligent diagramming for teams. <https://www.lucidchart.com>, 2024. Consulté en 2025.
- [24] draw.io – online diagram software. <https://www.drawio.com>, 2024. Consulté en 2025.
- [25] Wampserver – apache, php and mysql on windows. <https://www.wampserver.com>, 2024. Consulté en 2025.

Annexe A

Technologies, outils et protocoles utilisés

Dans le cadre de la conception et du développement de l'application **DistriCharge**, nous avons adopté une architecture full-stack JavaScript, en privilégiant les technologies modernes, maintenables et bien adaptées aux projets web et mobiles. Ce choix a été guidé par des critères de simplicité, d'évolutivité, de performance et de disponibilité des ressources. L'application est structurée selon une architecture client-serveur, avec une séparation claire entre le frontend (interfaces utilisateur), le backend (logique métier et API) et la base de données (stockage structuré).

Afin d'assurer une **accessibilité optimale sur différents supports** (ordinateur, smartphone), nous avons fait le choix d'adopter une **approche multiplateforme**. Pour cela, deux interfaces ont été développées :

- Une interface **web**, avec **React JS** [11], optimisée pour les écrans larges (PC, laptop).
- Une interface **mobile**, avec **React Native** [12], adaptée aux écrans de petite taille, utilisée principalement par les superviseurs sur le terrain.

Bien que React JS et React Native soient basés sur la même bibliothèque JavaScript, ce sont deux projets distincts, chacun ciblant une plateforme spécifique. Toutefois, une partie de la logique métier a été mutualisée afin de garantir une cohérence fonctionnelle.

A.1 Frontend – Développement des interfaces utilisateur

A.1.1 React JS (Web)

Utilisé pour développer l'interface web de l'application [11]. React nous a permis de :

- Créer des composants modulaires et réutilisables (ex : composants pour l'affichage des formulaires, tableaux dynamiques, cartes de statistiques, etc.)
- Gérer l'état de l'application localement (via `useState`) et globalement (via `Redux Toolkit`).
- Implémenter un système de navigation fluide entre les pages avec `React Router DOM`.

- Afficher des interfaces dynamiques et réactives selon les rôles utilisateurs (superviseur, chef de région, etc.).

A.1.2 React Native (Mobile)

Développement de l'application mobile [12], pour les utilisateurs en déplacement :

- Réutilisation des fonctions de traitement et des appels API.
- Interface fluide adaptée aux écrans mobiles.
- Offrir une expérience utilisateur fluide sur mobile.

A.1.3 CSS3

La stylisation de l'interface repose sur **CSS3** avec le framework **Tailwind CSS** [13], assurant :

- Design responsive compatible PC, tablette, mobile.
- Thèmes adaptables selon les préférences utilisateur.
- Intégration de composants visuels comme les modales, tableaux, notifications...

A.1.4 JavaScript

Tout le code côté client (React, logique des composants, appels API, gestion des événements) est écrit en **JavaScript** [14], ce qui garantit la cohérence entre le frontend et le backend.

A.2 Backend – Serveur, API, logique métier

Le backend de DistriCharge a été conçu comme une API RESTful. Il centralise toutes les logiques métier de l'application : authentification, création et gestion des dépôts, présences, affectations, validations, calculs, etc.

A.2.1 Node.js et Express.js

Le backend est basé sur Node.js [15], avec le framework Express [16] :

- API REST structurée avec routes, middlewares et contrôleurs.
- Architecture MVC.
- Gérer les erreurs centralisées et les statuts HTTP.

Nous avons suivi une organisation claire :

- **/routes** : routes de l'API.
- **/controllers** : logique métier de chaque fonctionnalité.
- **/middlewares** : sécurité (JWT), validation d'entrée, contrôle d'accès.
- **/models** : modèles Sequelize pour la BDD.

A.3 Base de données – Stockage structuré

A.3.1 MySQL

SGBDR relationnel robuste utilisé pour stocker l'ensemble des données métier. [17]. Nous avons défini plusieurs entités clés :

- Utilisateurs (avec rôles : superviseur, chef de région, national. . .).
- Distributeurs , dépôts, canal, ...
- Présences, suivi, charges, ...

A.3.2 Sequelize (ORM)

Permet de définir les modèles, les relations, et gérer les migrations [18] :

- Définir les modèles JavaScript correspondant aux tables MySQL.
- Gérer les migrations, associations et synchronisation des modèles
- Les relations définies dans Sequelize reflètent les règles métiers : associations hasMany, belongsToMany, belongsTo, etc.

A.4 Authentification et sécurité

A.4.1 JSON Web Tokens (JWT)

Authentification basée sur JSON Web Tokens [19] :

- Lorsqu'un utilisateur se connecte, le serveur lui génère un token encodé.
- Ce token est ensuite utilisé pour authentifier les requêtes côté client.
- Il contient des informations sécurisées (ID, rôle, durée d'expiration).
- Chaque route protégée du backend vérifie la validité du token avec un middleware, et bloque l'accès en cas de token expiré ou invalide.

A.4.2 Rôles et permissions

Le système de contrôle d'accès basé sur les rôles (superviseur, chef de région, national des ventes, chargé du reporting, directeur commercial). Chaque utilisateur est authentifié et identifié par son rôle dès la connexion, ce qui détermine automatiquement les données auxquelles il peut accéder et les actions qu'il peut effectuer. Ce système garantit non seulement la confidentialité des informations sensibles, mais aussi la fluidité des processus métiers.

A.4.3 HTTPS

Bien que le développement ait été fait en local via HTTP, le déploiement est prévu en HTTPS pour sécuriser les échanges, notamment ceux liés à l'authentification.

A.5 Outils de développement

- **Visual Studio Code (VSCode)** : éditeur de code avec extensions (Prettier, ESLint, GitLens) [20]
- **Postman** : Postman nous a permis de tester manuellement toutes les routes backend, notamment l'authentification, la création de ressources, les erreurs et les droits d'accès. [21]
- **Git & GitHub** :
 - Git a servi pour le versionnement, le travail en branches, les fusions et la traçabilité.
 - GitHub a été utilisé pour héberger le dépôt, partager le code, gérer les issues, les commits et la documentation technique [22].
- **Lucidchart & Draw.io** : Création des diagrammes UML (cas d'utilisation, classes, séquences). [23, 24]
- **WampServer** : Hébergement local de la base MySQL pendant le développement [25].

Résumé

Ce mémoire s'inscrit dans le cadre d'un stage de fin de cycle réalisé au sein du département Système d'Information de l'entreprise Cevital, en vue de l'obtention du diplôme de **Master 2 en Informatique, option Génie Logiciel**.

L'objectif principal de ce projet est de concevoir et de développer une application web **DistriCharge**, visant à digitaliser plusieurs processus internes du service commercial, notamment la gestion des distributeurs, des dépôts, des effectifs, des primes et des charges... L'application permet de centraliser l'ensemble des opérations et d'en assurer l'accès selon les rôles métier : superviseurs, chefs de région, nationaux de vente, chargé du reporting et directeur commercial. Cette centralisation permet non seulement de remplacer les procédures manuelles, mais également de fiabiliser les données et de garantir une meilleure traçabilité.

Pour la mise en œuvre de cette solution, nous avons adopté une démarche agile basée sur la méthodologie **Scrum**, et avons utilisé des technologies modernes telles que **React** pour le développement du front-end, **Node.js** et **Express** pour le back-end, et **MySQL** pour le stockage des données.

Mots-clés : DistriCharge, Cevital, service commercial, digitalisation, application web, Scrum, React, Node.js.

Abstract

This thesis is part of a final-year internship carried out within the Information Systems Department of Cevital, in fulfillment of the requirements for the **Master's degree in Computer Science, Software Engineering track**.

The main objective of this project is to design and develop a web application called **DistriCharge**, aimed at digitizing several internal processes of the commercial department, notably the management of distributors, depots, personnel, bonuses, and expenses. The application centralizes all operations and ensures access based on business roles : supervisors, regional managers, national sales managers, reporting officers, and the commercial director. This centralization not only replaces manual procedures but also improves data reliability and ensures better traceability.

To implement this solution, we adopted an agile approach based on the **Scrum** methodology, and used modern technologies such as **React** for front-end development, **Node.js** and **Express** for the back-end, and **MySQL** for data storage.

Keywords : DistriCharge, Cevital, commercial department, digitization, web application, expense management, Scrum, React, Node.js.