



DÉDICACE

À ma famille, merci pour votre soutien inébranlable tout au long de mon parcours. Votre présence constante, vos conseils avisés et votre confiance en moi ont été essentiels dans la réussite de ce projet.

À mes chers amis, vous avez été des piliers tout au long de cette aventure académique. Votre soutien indéfectible et vos encouragements m'ont permis de surmonter les moments difficiles et ont été un véritable moteur dans la réalisation de ce projet.

À mon binôme, pour ton travail acharné, ton dévouement et ton esprit d'équipe. Ce projet n'aurait pas été le même sans toi, et ton soutien a été inestimable tout au long de ce travail commun.

À vous tous,

Je dédie ce travail.

Fida SASSI

Islem TURKI



REMERCIEMENT

Nous tenons à exprimer notre profonde gratitude et notre reconnaissance envers toutes les personnes qui nous ont soutenus et aidés tout au long de la réalisation de ce travail.

Nous remercions tout particulièrement M. Blél Dhyaddine, ingénieur chez ArabSoft, pour ses précieux conseils, son expertise et sa disponibilité, ainsi que pour sa bienveillance et son écoute attentive. Nous souhaitons également remercier chaleureusement tous les membres de l'entreprise ArabSoft pour leur accueil et leur soutien.

Nous tenons à adresser nos sincères remerciements à M. Waheb Aymen, notre encadrant, pour son accompagnement constant et ses conseils judicieux. Sa générosité et son dévouement ont été essentiels à l'aboutissement de ce projet.

Nous souhaitons également exprimer notre reconnaissance à l'ISETN (Institut Supérieur des Études Technologiques de Nabeul) et à ses professeurs, pour la qualité de l'enseignement dispensé et pour avoir contribué à notre formation. Un grand merci à tous les enseignants qui nous ont guidés et soutenus tout au long de nos études, en particulier dans le cadre de ce projet.

Enfin, nous remercions les membres du jury d'avoir accepté d'évaluer notre travail et pour leurs remarques constructives qui nous aideront à progresser.



TABLE DES MATIÈRES

LISTE DES FIGURES	x
LISTE DES TABLEAUX	xii
LISTE DES ABRÉVIATIONS	xiii
INTRODUCTION GÉNÉRALE	1
Chapitre 1 : Etude de Projet	2
1.1 Présentation de l'organisme d'accueil	3
1.1.1 Présentation générale	3
1.1.2 Organigramme de Arabsoft	3
1.1.3 Positionnement du stage dans le domaine AJIR	4
1.2 Etude du projet	5
1.2.1 Cadre du projet	5
1.2.2 Analyse de l'existant	5
1.2.3 Solution proposée	6
1.3 Méthodologie de gestion de projet	7
1.3.1 Choix de la méthodologie	8
1.3.1.1 Définition de la méthode : Scrum	8
1.3.1.2 Fondements théoriques	9
1.3.1.3 Mise en œuvre pratique	9
Chapitre 2 : Analyse et Spécification des Besoins	11
2.1 Sprint 0	12
2.2 Spécification des besoins	12
2.3 Identification des acteurs	12
2.4 Besoins fonctionnels et non fonctionnels	13
2.4.1 Besoins fonctionnels	13

2.4.2	Besoins non fonctionnels	15
2.5	Pilotage du projet avec Scrum	16
2.6	L'équipe Scrum	16
2.7	Backlog du produit	17
2.8	Diagramme de cas d'utilisation global	19
2.9	Planification des Sprints	20
2.10	Environnement de travail	20
2.11	Architecture de l'application	26
2.11.1	Architecture logique	26
2.11.2	Architecture du projet Front-End	27
2.11.3	Flux de données et traitement	28
Chapitre 3 : Conception et réalisation du sprint 1		29
3.1	Sprint 1	30
3.2	User Stories	30
3.3	Diagrammes	31
3.3.1	Diagramme de cas d'utilisation du sprint 1	31
3.3.2	Diagramme de classes du sprint 1	32
3.4	Description textuelle du cas "Login Admin"	32
3.4.1	Diagramme du cas "Login Admin"	33
3.5	Description textuelle du cas "Ajouter succursale sociétés"	34
3.5.1	Diagramme du cas "Ajouter succursale Société"	35
3.6	Description textuelle du cas "Modifier succursale société"	35
3.6.1	Diagramme du cas "Modifier succursale Société"	36
3.7	Description textuelle du cas "Supprimer succursale société"	37
3.7.1	Diagramme du cas "Supprimer succursale Société"	37
3.8	Description textuelle du cas "Ajouter un service par nom"	38
3.8.1	Diagramme du cas "Ajouter un service par nom"	39
3.9	Description textuelle du cas "Ajouter un service par nom et chefs de service"	39
3.9.1	Diagramme du cas "Ajouter un service par nom et chefs de service"	40
3.10	Description textuelle du cas "Ajouter un Personnel"	40
3.10.1	Diagramme du cas "Ajouter un Personnel"	41
3.11	Description textuelle du cas "Inscription Personnel"	42
3.11.1	Diagramme du cas "Inscription Personnel"	43
3.12	Description textuelle du cas "Activer un Personnel"	43

3.12.1 Diagramme du cas "Activer un Personnel"	44
3.13 Description textuelle du cas "Modifier un Personnel"	45
3.13.1 Diagramme du cas "Modifier un Personnel"	45
3.14 Description textuelle du cas "Login Personnel"	46
3.14.1 Diagramme associés au cas "Login Personnel"	47
3.15 Description textuelle du cas "Modification du mot de passe"	48
3.15.1 Diagramme de séquence du cas "Modification du mot de passe"	49
3.16 Résultats obtenus	49
3.16.1 Authentification	49
3.16.1.1 Interfaces Web	50
3.16.1.2 Interfaces Mobile	51
3.16.2 Admin - Gestion des personnels	52
3.16.3 Personnels - Gestion du Profil	53
3.16.3.1 Interfaces Web	53
3.16.3.2 Interfaces Mobile	53
3.16.4 Gerer les succursales sociétés	54
3.16.5 Gerer les services	55
Chapitre 4 : Conception et réalisation du sprint 2	57
4.1 Sprint 2	58
4.2 Backlog du sprint	59
4.3 Diagrammes du cas "Sprint 2"	62
4.3.1 Diagramme de cas d'utilisation	62
4.3.2 Diagramme de classes	63
4.4 La gestion des demandes	64
4.5 Description textuelle du cas : Ajouter une demande	64
4.5.1 Diagramme du cas "Ajouter une demande"	64
4.6 Description textuelle du cas : Modifier une demande	65
4.6.1 Diagramme du cas "Modifier une demande"	65
4.7 Description textuelle du cas : Supprimer une demande	66
4.7.1 Diagramme du cas "Supprimer une demande"	67
4.8 Description textuelle du cas : Examiner (rejeter/ approuver) une demande	67
4.8.1 Diagramme du cas "Examiner (rejeter/ approuver) une demande"	68
4.9 Description textuelle du cas : Traiter une demande	69
4.9.1 Diagramme du cas "Traiter une demande"	69

4.10	La gestion des notifications	70
4.10.1	Technologies utilisées et justification des choix :	70
4.10.1.1	WebSocket : Comment ça fonctionne ?	70
4.11	Description Textuelle des Cas d'Utilisation	71
4.11.1	Description textuelle du cas "Recevoir une notification en temps réel"	71
4.11.1.1	Diagramme de Séquence du cas "Recevoir une notification en temps réel"	72
4.11.2	Description textuelle du cas "Consulter les notifications internes"	72
4.11.2.1	Diagramme de Séquence du cas "Consulter les notifications internes"	73
4.11.3	Flux de l'envoi d'un e-mail	74
4.11.3.1	Détails techniques	74
4.11.4	Description textuelle du cas "Envoi d'e-mails"	75
4.11.4.1	Diagramme de séquence du cas "Envoi d'e-mails"	75
4.12	Résultats obtenus	77
4.12.1	Demandes	77
4.12.1.1	Interfaces Web	77
4.12.1.2	Interfaces Mobile	79
4.12.2	Notifications	81
4.12.2.1	Interfaces Web	81
4.12.2.2	Interfaces Mobile	81
Chapitre 5 : Intégration du Module d'analyse de CV par AI		82
5.1	Sprint 3 :	83
5.2	Technologies Utilisées et Justification des Choix	83
5.3	Backlog du Sprint	85
5.4	Diagrammes du Sprint 3	85
5.4.1	Diagramme de cas d'utilisation	85
5.5	Diagramme de classes global du Sprint 3	86
5.6	Description Textuelle des Cas d'Utilisation	88
5.6.1	Description textuelle du cas " Ajout d'une offre par l'Admin"	88
5.6.1.1	Diagramme de Séquence du cas "Ajout d'une offre par l'Admin"	89
5.6.2	Description textuelle du cas " Postuler a une offre"	89
5.6.2.1	Diagramme de Séquence du cas "Postuler a une offre"	91
5.6.3	Description textuelle du cas "Traitement du CV par l'IA"	91

5.6.3.1	Diagramme de séquence du cas "Traitement du CV par l'IA"	93
5.7	Prompt d'analyse pour l'intelligence artificielle	93
5.8	Hébergement des applications Web et Mobile	94
5.8.1	Application Web - Render	94
5.8.2	Application Mobile - Expo	95
5.8.2.1	Génération et Hébergement de l'APK	95
5.9	Résultats Obtenus	95
5.9.1	Interfaces Web	95
5.9.1.1	Partie Admin : CRUD Offres, consulter les offres, consulter les candidats avec scores	95
5.9.1.2	Partie Candidat : Postuler à une offre	96
ANNEXES		98
A.1	Annexe 1 : Prompt utilisé pour l'analyse IA	98
A.2	Annexe 2 : Résultat réel de l'analyse IA d'un candidat	100
BIBLIOGRAPHIE		100
CONCLUSION GÉNÉRALE		103



LISTE DES FIGURES

1.1	Logo de la société ArabSoft	3
1.2	Organigramme de la société	4
1.3	Cycle de vie Scrum appliqué à notre projet	9
1.4	Cycle de vie Scrum appliqué à notre projet	10
2.1	Diagramme de cas d'utilisation global.	19
2.2	Logo VS Code	21
2.3	Logo IntelliJ IDEA	21
2.4	Logo Spring Boot	21
2.5	Logo commun à React et React Native	22
2.6	Logo Expo	22
2.7	Logo CSS3	22
2.8	Logo MongoDB	22
2.9	Logo Postman	23
2.10	Logo Python	23
2.11	Logo LLaMA	24
2.12	Logo LangChain	24
2.13	Logo Groq	24
2.14	Logo Render	24
2.15	Logo Overleaf	25
2.16	Logo PyMuPDF	25
2.17	Logo GitHub	25
2.18	Logo Draw.io	26
2.19	Logo PlantUML	26
2.20	Architecture de l'application	28
3.1	Diagramme de cas d'utilisation du sprint 1	31
3.2	Diagramme de classes du sprint 1	32
3.3	Diagramme de sequence du sprint 1 Login admin	33
3.4	Diagramme de sequence du sprint 1 du cas "Ajouter succursale Société"	35

3.5	Diagramme de sequence du sprint 1 du cas "Modifier une succursale société"	36
3.6	Diagramme de sequence du sprint 1 du cas "Supprimer une succursale société"	37
3.7	Diagramme de sequence du sprint 1 du cas "Ajouter un service par nom"	39
3.8	Diagramme de sequence du sprint 1 du cas "Ajouter un service par nom et chefs de service"	40
3.9	Diagramme de séquence du sprint 1 ajouter personnel	41
3.10	Diagramme de séquence du sprint 1 Inscription personnel	43
3.11	Diagramme de séquence du sprint 1 activer Personnel	44
3.12	Diagramme de séquence du sprint 1 Modifier Personnel	45
3.13	Diagramme de séquence du sprint 1 — Login Personnel	47
3.14	Diagramme de séquence du sprint 1 Modification du mot de passe	49
3.15	Interface Web d'inscription - Thème clair	50
3.16	Interface Web de connexion - Thème clair	50
3.17	Interfaces de profil personnel	50
3.18	Interfaces Mobiles d'authentification - Thème clair	51
3.19	Interfaces Mobiles d'authentification - Thème sombre	51
3.20	Interface Web de la liste des personnels - Thème clair	52
3.21	Interface Web d'ajout personnel - Thème clair	52
3.22	Interfaces de gestion personnel	52
3.23	Profile (Thème clair)	53
3.24	Interfaces de profil personnel	53
3.25	Interface Web d'ajout succursale société - Thème clair	54
3.26	Interface Web de la liste des succursales - Thème clair	54
3.27	Interface Web modification d'un succursale - Thème clair	54
3.28	Interface Web d'ajout service avec nom - Thème clair	55
3.29	Interface Web d'ajout service avec nom et chefs - Thème clair	55
3.30	Interface Web de la liste des services - Thème clair	55
3.31	Interface Web modification d'un service - Thème clair	56
3.32	Interfaces de gestion personnel	56
3.33	Interfaces de gestion personnel	56
4.1	Diagramme de cas d'utilisation du cas "Sprint 2"	62
4.2	Diagramme de classes du cas "Sprint 2"	63
4.3	Diagramme de séquence du cas "Ajouter une demande"	64
4.4	Diagramme de séquence du cas "Modifier une demande"	65

4.5	Diagramme de séquence du cas "Supprimer une demande"	67
4.6	Diagramme de séquence du cas "Examiner (rejeter/ approuver) une demande" .	68
4.7	Diagramme de séquence du cas "Traiter une demande"	69
4.8	Connection WebSocket	70
4.9	Diagramme de séquence "Recevoir une notification en temps réel"	72
4.10	Diagramme de séquence "Consulter les notifications internes"	73
4.11	Flux de l'envoi d'un e-mail	74
4.12	Diagramme de séquence : Processus d'envoi d'e-mail	76
4.14	Interfaces Web des demandes	78
4.15	Interfaces web pour la gestion des demandes	79
4.16	Interfaces mobiles pour la gestion des demandes	80
4.17	Icône de notification dans l'interface utilisateur	81
4.18	Interface des notifications côté Web	81
4.19	Notifications côté mobile	81
5.1	Comparaison des vitesses de génération (tokens/seconde) pour Llama 3.1 70B selon différentes plateformes. Groq est nettement le plus rapide.	84
5.2	Exemple de création d'une clé API sur la plateforme Groq.	84
5.3	Diagramme de cas d'utilisation du Sprint 3	86
5.4	Diagramme de classe global du Sprint 3	87
5.5	Diagramme de séquence du sprint 3 – Cas "Ajouter une offre"	89
5.6	Diagramme de séquence du sprint 3 – Cas ""Postuler a une offre"	91
5.7	Diagramme de séquence du sprint 3 – Cas "Traitement du CV par l'IA"	93
5.8	Prompt envoyé au modèle LLaMA 3 pour l'analyse des CVs	94
5.9	Capture d'écran de l'hébergement de l'application web sur Render	94
5.10	Interfaces admin : consultation des offres	95
5.11	Interfaces admin : Consultation des candidats qui ont postuler a une Offre . . .	96
5.12	Interfaces admin : ajout d'une offre	96
5.13	Interfaces candidat : consultation des offres	96
5.14	Interfaces candidat : postuler à une offre	97
A.1	Python Prompt	98
A.2	Exemple d'une offre pour postuler	99
A.3	Exemple d'un CV soumis à l'analyse IA	99
A.4	Exemple de résultat IA stocké en base de données pour un candidat	100



LISTE DES TABLEAUX

1.1	Équipe Scrum du projet	8
1.2	Principaux éléments de la méthodologie Scrum dans notre projet	9
2.1	Tableau des acteurs du système	12
2.2	User Stories par rôle et priorité	18
2.3	Planification des sprints	20
2.4	Comparaison des environnements techniques utilisés	20
2.5	Avantages de MongoDB pour le projet Portal RH	23
3.1	Liste des User Stories du sprint 1	31
3.2	Description du cas « Login admin »	33
3.3	Cas d'utilisation – Modifier une succursale société	36
3.4	Cas d'utilisation – Supprimer une succursale société	37
3.5	Cas d'utilisation – Ajouter un service par nom	38
3.6	Cas d'utilisation : Ajouter un service par nom et chefs de service	40
3.7	Description du cas d'utilisation "Ajouter Personnel"	41
3.8	Description du cas d'utilisation "Inscription Personnel"	42
3.9	Scénarios du cas d'utilisation "Activer Personnel"	44
3.10	Description du cas d'utilisation "Modifier un Personnel"	45
3.11	Description du cas d'utilisation "Login Personnel"	46
3.12	Description textuelle du cas d'utilisation "Modification du mot de passe"	48
4.1	Backlog Sprint 2	61
4.2	Description textuelle du cas – Ajouter une demande	64
4.3	Description textuelle du cas – Modifier une demande	65
4.4	Description textuelle du cas – Supprimer une demande	66
4.5	Scénario nominal du cas d'utilisation : Examiner une demande	68
4.6	Description textuelle du cas – Traiter une demande	69
4.7	Description du cas d'utilisation : Gestion des notifications en temps réel	71
4.8	Description du cas d'utilisation : Consultation des notifications	73
4.9	Description du cas d'utilisation "Envoi d'e-mail via EmailService"	75

5.1	Backlog du sprint 3	85
5.2	Contexte – Ajout d’une offre par l’Admin	88
5.3	Déroulement – Cas Postuler a une offre	90
5.4	Description – Cas "Traitement du CV par l’IA"	92



LISTE DES ABRÉVIATIONS

Oracle ADF Oracle Application Development Framework

RH Ressources Humaines

UML Unified Modeling Language (Langage de modélisation unifié)

AJIR Applications de Gestion Intégrée des Ressources

IA Intelligence Artificielle

LLM Large Language Model

JSON JavaScript Object Notation

LLM Large Language Model

API Application Programming Interface

NLP Natural Language Processing

JWT JSON Web Token

SMTP Simple Mail Transfert Protocol

STARTTLS Start Transport Layer Security

MIME Multipurpose Internet Mail Extensions



INTRODUCTION GÉNÉRALE

Internet et les technologies web jouent aujourd'hui un rôle clé dans la transformation des processus métier, notamment dans le domaine des ressources humaines.

Dans le cadre de notre projet de fin d'études à l'Institut Supérieur des Études Technologiques de Nabeul (ISETN), nous avons conçu l'« **Application RH pour la Gestion des Demandes des Employés et des Recrutements** », une solution web visant à simplifier l'administration du personnel et à automatiser le recrutement.

Pour en améliorer l'accessibilité, nous avons développé une version mobile permettant aux employés de soumettre leurs demandes via smartphone.

Un module d'intelligence artificielle a été intégré pour analyser les CV et optimiser la présélection des candidats, tandis qu'un système de notifications en temps réel assure une communication fluide et réactive.

Le rapport se structure ainsi :

Chapitre 1 : présentation de l'entreprise, problématique, état de l'art et méthodologie.

Chapitre 2 : analyse des besoins, modélisation UML, choix techniques et planification.

Chapitres 3, 4 et 5 : déroulement du projet à travers trois sprints, de la conception au déploiement.

Ce projet, réalisé en collaboration avec **ARABSOFT**, s'inscrit dans l'obtention de notre licence en Informatique de Gestion et illustre une démarche concrète pour moderniser la gestion RH grâce à des outils numériques innovants.

Chapitre 1 : Etude de Projet

1.1 Présentation de l'organisme d'accueil

1.1.1 Présentation générale

Arabsoft, fondée par M. Mohamed TRIKI en 1985 avec une équipe de 8 employés, emploie aujourd'hui plus de 150 collaborateurs en plus de 115 spécialistes en conception, développement et formation.

Aujourd'hui, ArabSoft est le principal éditeur tunisien de solutions de gestion : c'est donc dans l'industrie du secteur public que la société a su renforcer sa notoriété

La société ArabSoft est également reconnue à travers le monde par son esprit d'entrepreneuriat et son espérance d'innovation : ceci a entre autre permis au société de construire ses relations stables avec les clients, fortifier la base du marché détenue dans la vie collaborative.

En 2008, ArabSoft reçoit un label d'excellence : LE (ISO 9001).

ArabSoft, à plus de 30 ans d'expérience en gestion de élevée, développe et distribue des solutions d'assistance consultatives pour différents secteurs opérationnels mobilisant en majorité des grands clients.



FIGURE 1.1 – Logo de la société ArabSoft

1.1.2 Organigramme de Arabsoft

L'organigramme d'Arabsoft montre comment l'entreprise est organisée, en mettant en lumière les divers départements qui la composent.

La FIGURE suivante illustre l'organigramme de « ArabSoft ».

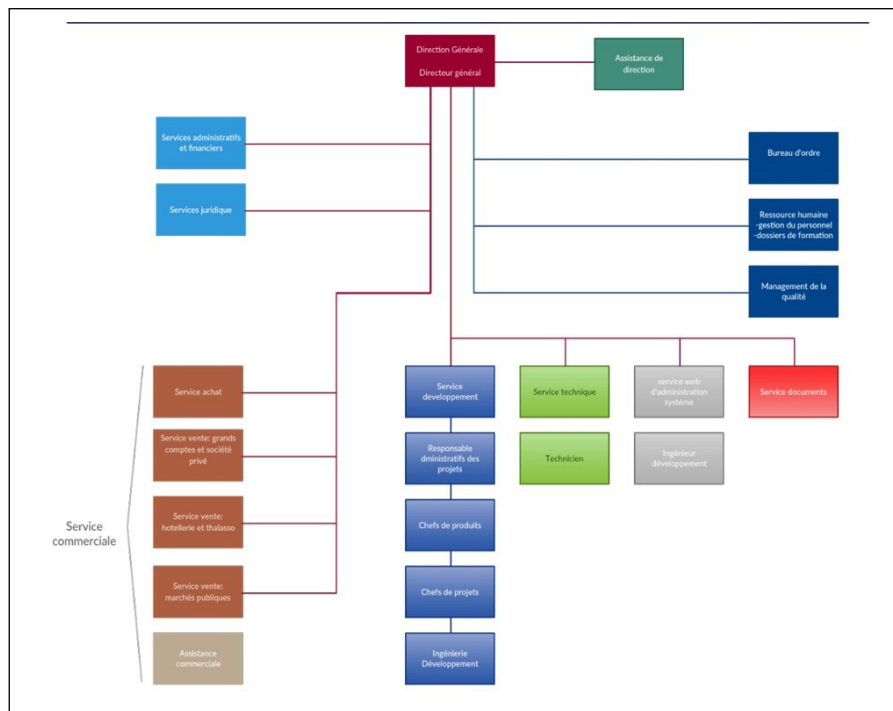


FIGURE 1.2 – Organigramme de la société

1.1.3 Positionnement du stage dans le domaine AJIR

Le stage que nous avons effectué s'inscrit dans le domaine **AJIR (Applications de Gestion Intégrée des Ressources)**, un pilier stratégique dans l'écosystème des solutions proposées par ArabSoft.

Le domaine AJIR regroupe l'ensemble des outils informatiques conçus pour optimiser la gestion des ressources humaines et administratives. Il couvre notamment :

- la gestion des dossiers du personnel,
- le suivi des congés,
- l'automatisation des processus RH (recrutement, validation, affectation),
- la centralisation et la traçabilité des données administratives.

Notre projet, centré sur le développement d'un **Portail RH intelligent et interactif**, s'intègre naturellement dans ce domaine. Il vise à moderniser les processus RH de l'entreprise en y intégrant :

- des interfaces ergonomiques accessibles sur web et mobile,
- un système d'analyse de CV basé sur l'intelligence artificielle,
- un moteur de notification temps réel basé sur WebSocket,
- et une architecture technique souple et maintenable.

Ce positionnement dans le domaine AJIR permet ainsi de répondre aux défis contemporains des entreprises en matière de digitalisation, d'automatisation et de gestion intelligente des ressources humaines.

1.2 Etude du projet

Dans cette section, nous allons examiner en détail l'état actuel du projet, dans le but d'identifier les éventuelles problématiques et d'y apporter des solutions adaptées.

1.2.1 Cadre du projet

Ce stage s'inscrit dans le cadre de la validation de notre projet de fin d'études en vue de l'obtention de notre licence en Technologie de l'Informatique, délivrée par l'Institut Supérieur des Études Technologiques de Nabeul (ISET Nabeul).

Dans ce cadre, il nous est demandé de réaliser un projet concret mettant en valeur les connaissances théoriques ainsi que les compétences techniques et pratiques acquises tout au long de notre formation universitaire.

Nous avons ainsi intégré la société ArabSoft pour y effectuer un stage d'une durée de quatre mois, au cours duquel nous serons amenés à concevoir et développer une solution répondant à des problématiques liées à la Technologie de l'Informatique.

1.2.2 Analyse de l'existant

L'analyse de l'existant permet d'identifier les limites du système actuel afin de guider la conception d'une solution plus adaptée.

Dans notre contexte, le portail Oracle ADF, dédié à la gestion RH, offre robustesse et modularité, mais présente aujourd'hui plusieurs faiblesses : une interface utilisateur vieillissante, une intégration limitée avec les autres systèmes, une incapacité à répondre aux nouveaux besoins métiers comme le télétravail, les workflows agiles, l'intelligence artificielle (par exemple, l'analyse automatique des CV) ainsi que l'absence de fonctionnalités modernes telles que la notification en temps réel.

Ces limites freinent l'innovation, alourdissent les processus RH et réduisent l'efficacité opérationnelle. Il devient donc essentiel de repenser l'architecture pour proposer une solution moderne, flexible et évolutive, capable d'accompagner l'entreprise face aux défis actuels et futurs.

1.2.3 Solution proposée

Pour pallier les inconvénients liés à Oracle ADF, nous proposons une solution moderne et flexible basée sur une architecture **React/React Native** pour le frontend et **Spring Boot** pour le backend, avec **MongoDB** comme base de données. Cette approche constitue une alternative performante, évolutive et indépendante des technologies propriétaires.

Avantages de la solution

- **Performance optimisée** : Grâce à React (pour le web) et React Native (pour le mobile), l'interface utilisateur est réactive, fluide et légère. Spring Boot permet un backend rapide avec une faible latence, évitant les lenteurs rencontrées avec ADF.
- **Interface dynamique et moderne** : L'interface utilisateur est entièrement dynamique, avec une navigation fluide, des composants interactifs et une expérience utilisateur améliorée sur tous les appareils.
- **Notifications en temps réel** : Grâce à l'utilisation de la technologie **WebSocket**, les utilisateurs bénéficient d'un système de communication bidirectionnelle en temps réel avec le serveur. Cela permet de recevoir instantanément des notifications lors d'événements importants tels que la validation d'une demande, la mise à jour de données, etc., sans avoir besoin de recharger la page.
- **Maintenance simplifiée** : Les frameworks utilisés (React, Spring Boot) sont maintenus activement par de larges communautés. La structure modulaire du code facilite les mises à jour et l'ajout de nouvelles fonctionnalités sans perturber le système existant.
- **Indépendance technologique** : Contrairement à Oracle ADF, la solution proposée est libre de toute dépendance à des outils propriétaires. Spring Boot embarque son propre serveur, ce qui permet un déploiement simple et autonome. L'application peut ainsi être facilement déployée sur des plateformes cloud modernes telles que **Render**, sans configuration serveur complexe. De plus, MongoDB, en tant que base NoSQL, offre une bonne performance en termes de temps d'accès et d'optimisation mémoire, ce qui la rend adaptée à des applications nécessitant flexibilité et efficacité dans la gestion des données.
- **Accessibilité pour les développeurs** : Les technologies choisies sont largement adoptées dans l'industrie, ce qui facilite le recrutement, la formation et l'intégration rapide de nouveaux développeurs.

Fonctionnalités Clés du Portail

La solution développée propose un ensemble de fonctionnalités modernes et adaptées aux besoins des services RH et des employés. Parmi les principales :

- **Soumission et suivi des demandes** : Les employés peuvent soumettre différentes demandes (congrés, autorisations, attestations, etc.) et suivre leur état d'avancement.
- **Gestion des employés** : Une interface dédiée permet à la DRH de gérer les informations des employés (ajout, modification, activation/désactivation), assurant une centralisation des données RH.
- **Automatisation des workflows** : Les processus de validation (par niveaux hiérarchiques) sont automatisés, accompagnés de notifications en temps réel pour chaque étape du traitement des demandes.
- **Rapports analytiques** : Des tableaux de bord dynamiques offrent des statistiques détaillées (nombre de personnels, types et nombres de demandes, etc.) afin d'aider à la prise de décision stratégique.
- **Assistance par intelligence artificielle** : Un module d'IA permet d'analyser les CV des candidats pour en extraire automatiquement les compétences clés. Cela aide les administrateurs RH à accélérer le processus de présélection et à prendre des décisions éclairées.

1.3 Méthodologie de gestion de projet

Une méthodologie de gestion de projet consiste en un ensemble de principes, techniques et procédures que les professionnels d'une discipline particulière appliquent pour gérer leurs projets. Les méthodologies varient non seulement selon la structure des organisations qui les adoptent, mais également en termes de livrables, de processus et d'outils utilisés (tels que les diagrammes de Gantt), voire du logiciel de gestion de projet choisi.

Les méthodologies de gestion de projet peuvent être classées en deux grandes catégories :

- **Méthodologies traditionnelles** : Ces méthodologies suivent un processus linéaire et séquentiel, souvent appelé "cycle en V" ou "waterfall". Les phases du projet (planification, conception, développement, tests, déploiement) sont clairement définies et suivent un ordre spécifique. Un exemple classique est la méthode **Waterfall**, où chaque phase doit être terminée avant de passer à la suivante.
- **Méthodologies agiles** : Contrairement aux méthodes traditionnelles, les méthodologies agiles sont itératives et incrémentales. Elles mettent l'accent sur la flexibilité, la collaboration et la réponse rapide aux changements. Le travail est organisé en petites itérations appelées **sprints**, et les équipes ajustent continuellement le plan en fonction des retours et des besoins du client. Un exemple populaire est la méthode **Scrum**.

1.3.1 Choix de la méthodologie

Dans un contexte RH aux besoins évolutifs, qu'ils soient réglementaires, liés aux compétences ou à la personnalisation, une méthodologie agile s'impose pour garantir réactivité, collaboration et livraison progressive.

Le cadre **Scrum** a été retenu comme méthodologie optimale, car il :

- Offre une flexibilité face aux changements grâce à des sprints itératifs de 2 à 4 semaines.
- Assure un alignement constant avec les parties prenantes grâce aux cérémonies (*daily meetings*, revues, rétrospectives).
- Réduit les risques en validant rapidement les fonctionnalités clés (par exemple : modules RH , interface candidat) avant un déploiement complet.
- S'intègre parfaitement aux technologies choisies (Spring Boot, MongoDB), facilitant l'intégration continue et les évolutions.

Contrairement aux méthodes classiques (comme le cycle en V), Scrum combine structure et agilité, accélérant la création de valeur et répondant efficacement aux besoins RH dynamiques.

1.3.1.1 Définition de la méthode : Scrum

Scrum est un cadre de travail agile utilisé pour la gestion de projets complexes, notamment dans le développement logiciel, mais également dans d'autres domaines. Il favorise la collaboration, l'adaptabilité et l'implantation incrémentielle de produits à forte valeur ajoutée.

Rôle	Membre
Product Owner (PO)	BVMT
Scrum Master	Blel Dhia
Équipe de développement	Sassi Fida et Turki Islem

TABLE 1.1 – Équipe Scrum du projet

Présentation du client

BVMT (Bourse des Valeurs Mobilières de Tunis) est l'institution chargée de l'organisation et du contrôle du marché boursier tunisien. Elle a pour mission de garantir la transparence des échanges financiers et d'accompagner les entreprises dans leur financement. Dans le cadre de sa digitalisation, elle est le commanditaire du projet *Application RH pour la Gestion des Demandes des employés et des Recrutements*.

1.3.1.2 Fondements théoriques

Les principes fondamentaux qui guident notre approche reposent sur les valeurs suivantes :

- **Les individus et leurs interactions** avant les processus et les outils
- **Un logiciel fonctionnel** avant une documentation exhaustive
- **La collaboration avec le client** avant la négociation de contrats
- **La réactivité au changement** avant la rigidité d'un plan préétabli

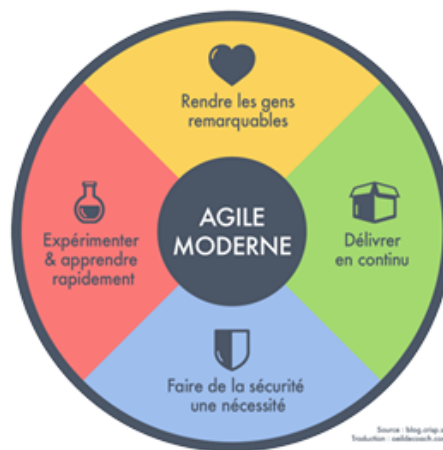


FIGURE 1.3 – Cycle de vie Scrum appliqué à notre projet

1.3.1.3 Mise en œuvre pratique

L'application concrète de Scrum dans notre projet repose sur les éléments suivants :

Élément	Description
Sprints	Itérations de 2 à 4 semaines, chacune ciblant un ensemble de fonctionnalités spécifiques
Daily Stand-up	Réunion quotidienne de 15 minutes à 9h pour synchroniser l'équipe et identifier les obstacles
Sprint Review	Présentation des fonctionnalités complétées devant les parties prenantes à la fin de chaque sprint
Rétrospective	Réflexion en équipe sur le déroulement du sprint pour identifier des axes d'amélioration

TABLE 1.2 – Principaux éléments de la méthodologie Scrum dans notre projet

Cette méthodologie nous permet de :

- Maintenir une **vélocité** constante tout au long du projet
- Assurer une **transparence** totale dans le suivi de l'avancement
- Faciliter une **adaptation** rapide face aux changements
- Garantir une **qualité** optimale du produit final

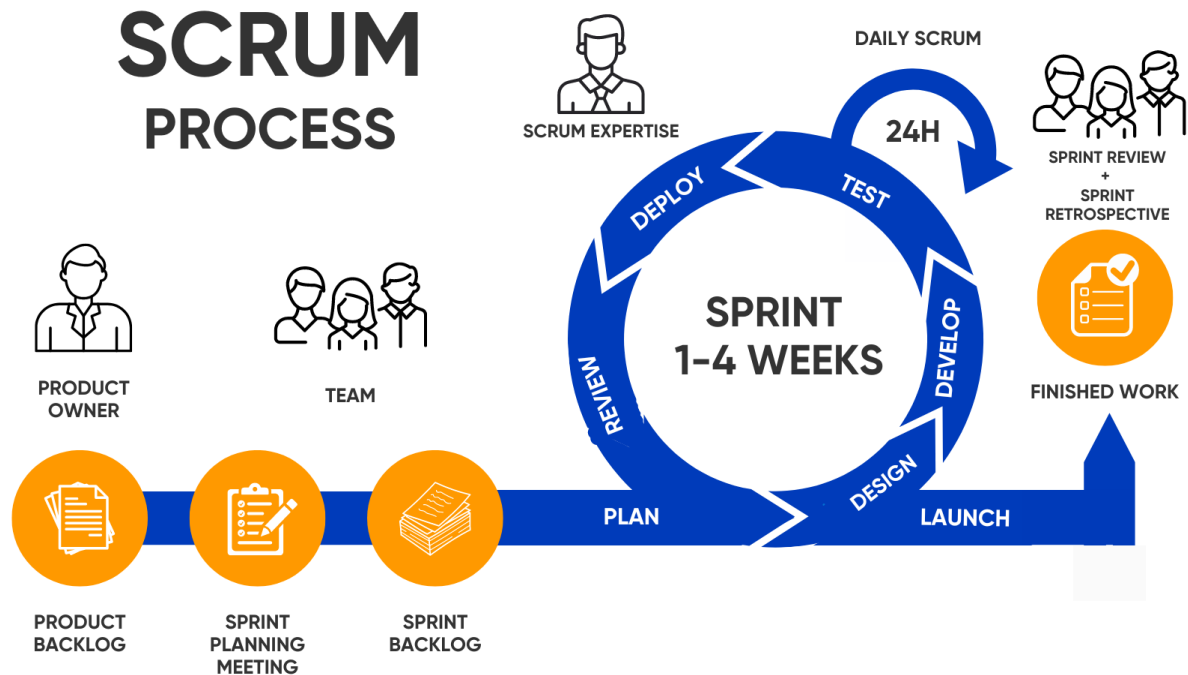


FIGURE 1.4 – Cycle de vie Scrum appliqué à notre projet

Avantages observés :

- **Réduction des risques** Identification rapide des problèmes et obstacles
- **Amélioration de la satisfaction client** Collecte continue de retours utilisateurs
- **Optimisation des coûts** Diminution des efforts inutiles et meilleure planification

Chapitre 2 : Analyse et Spécification des Besoins

2.1 Sprint 0

Le sprint 0, aussi appelé itération 0, est une étape préparatoire essentielle avant le début du développement réel. Contrairement aux autres sprints, il n'a pas de durée fixe et ne nécessite pas forcément de livraison à la fin. Son but principal est de poser les bases du projet.

Durant cette phase, nous avons travaillé sur plusieurs points importants :

- **Préparation de l'environnement de développement** et mise en place des outils nécessaires.
- **Clarification des objectifs du projet** et de la vision globale partagée par l'équipe.
- **Définition d'une première version du backlog produit**, qui regroupe les fonctionnalités à développer.
- **Réflexion sur l'architecture technique** adaptée aux besoins du projet, en tenant compte des technologies choisies (React, React Native, Spring Boot, MongoDB).

2.2 Spécification des besoins

Cette étape consiste à bien comprendre et cadrer le contexte du système. Dans ce qui suit, nous listons les acteurs, les besoins fonctionnels et les besoins non fonctionnels du projet.

2.3 Identification des acteurs

Le tableau suivant récapitule les différents acteurs impliqués dans le système.

Acteur	Description
Administrateur	Il contrôle la plateforme, et gère les comptes du personnel et gère les nouveaux candidats .
Administrateur RH	Il gère Ses personnel et Gère les différents types des demandes .
Chef hiérarchique	Il gère les différents types de demandes et consulte les historiques relatifs à son ou ses services concernés.
Collaborateur	Il gère ses propres demandes et consulte son Compte.

TABLE 2.1 – Tableau des acteurs du système

2.4 Besoins fonctionnels et non fonctionnels

2.4.1 Besoins fonctionnels

Les besoins fonctionnels décrivent les fonctionnalités principales que le système doit offrir. Parmi eux :

(A) Les fonctionnalités de l'administrateur

1. S'authentifier.
2. Gérer les personnels :
 - Ajouter des personnels.
 - Activer/Désactiver des personnels.
 - Assigner des rôles.
 - Assigner des services.
 - Modifier les informations du personnel.
3. Gérer les services :
 - Ajouter un service.
 - Modifier un service.
 - Supprimer un service.
 - Assigner jusqu'à trois chefs de service avec des poids (1, 2, 3).
 - Consulter les services.
4. Gérer les candidatures :
 - Ajouter une candidature.
 - Visualiser les candidats ayant postulé.
 - Bénéficier d'un module d'intelligence artificielle pour l'analyse automatique des CV.
5. Gérer les succursales société :
 - Ajouter une succursale société.
 - Modifier une succursale société.
 - Supprimer une succursale société.
6. Consulter son profil.
7. Recevoir des notifications en temps réel lors des changements d'état.

(B) Les fonctionnalités de l'administrateur RH

1. S'authentifier.
2. Visualiser les membres de son équipe (tous les personnels activés).

3. Traiter les demandes acceptées par les chefs :
 - Demande d'autorisations.
 - Demande de congé.
 - Demande de formation.
4. Traiter ou refuser les demandes suivantes :
 - Demande de préavance.
 - Demande de document.
5. Consulter le calendrier des congés des personnels.
6. Disposer de toutes les fonctionnalités du collaborateur.
7. Recevoir des notifications en temps réel lors des changements d'état.

(C) Les fonctionnalités du chef hiérarchique

1. S'authentifier.
2. Visualiser les personnels actifs des services sous sa direction (avec mention de son poids).
3. Analyser les demandes des personnels de ses services :
 - Approuver ou refuser :
 - Demande d'autorisations.
 - Demande de congé.
 - Demande de formation.
 - Les décisions des chefs de poids 2 et 3 sont des avis consultatifs.
 - Le chef de poids 1 prend la décision finale.
4. Consulter le calendrier des congés acceptés des personnels de son service concerné.
5. Disposer de toutes les fonctionnalités du collaborateur.
6. Recevoir des notifications en temps réel lors des changements d'état.

Remarques :

- Un chef peut être responsable de plusieurs services, avec des poids différents dans chacun.
- Pour chaque service, il y a trois chefs.

(D) Les fonctionnalités du collaborateur

1. S'authentifier.
2. Gérer des demandes :
 - Ajouter des demandes.
 - Modifier des demandes.
 - Supprimer des demandes.
 - Consulter l'historique de ses demandes.

3. Recevoir des notifications (changements d'état des demandes, rappels, etc.).
4. Consulter son profil personnel.
5. Consulter le calendrier de ses congés acceptés par son chef de service (chef de poids 1).
6. Télécharger ses documents administratifs.

Remarque : Chaque collaborateur est rattaché à un seul service.

(F) Les fonctionnalités de nouveau candidat

1. Consulter les offres d'emploi disponibles.
2. Postuler à une ou plusieurs offres.
3. Suivre le statut de sa candidature :
 - Notification par email au changement de statut.
4. Bénéficier d'un module d'intelligence artificielle pour l'analyse automatique du CV et l'optimisation de la candidature.

2.4.2 Besoins non fonctionnels

Les besoins non fonctionnels décrivent les contraintes de qualité et de performance du système :

1. Sécurité des données

- Protection des informations personnelles et confidentielles.
- Mots de passe cryptés.

2. Ergonomie adaptée au contexte RH

- L'interface est conçue pour minimiser les clics nécessaires à la gestion du personnel et des demandes, avec des tableaux filtrables, une barre de recherche intelligente et des boutons d'action rapides.
- Une version mobile est disponible pour permettre aux personnels de soumettre leurs demandes depuis leur smartphone.

3. Performance optimisée

- Temps de réponse inférieur à 60 secondes pour 80 % des requêtes.
- Capacité à gérer plusieurs utilisateurs simultanés sans dégradation des performances.

4. Disponibilité et accessibilité

- Le portail est hébergé sur un serveur cloud (Render) avec un taux de disponibilité visé de 99,9 %, permettant un accès permanent pour les candidats et les personnels.

5. Performance IA et affichage

- L'évaluation automatique des CV via l'IA (LLaMA 3.2 + LangChain) doit fournir une réponse en moins de 20 secondes pour assurer une fluidité de traitement.

- Le chargement des tableaux de candidatures doit rester instantané (moins de 20 s), même avec plusieurs centaines d'entrées.

6. Scalabilité et maintenance

- L'architecture backend (Spring Boot) et frontend (React) est conçue pour supporter une montée en charge progressive, avec séparation des couches et appels API optimisés.
- Le code est structuré de manière modulaire pour faciliter l'ajout futur de nouvelles fonctionnalités (vidéo entretien, chat RH/candidat, scoring avancé).
- L'intégration de fonctionnalités basées sur une IA plus avancée côté candidat (par exemple : simulation d'entretien, analyse des offres) est prévue dans l'évolution de la plateforme.

2.5 Pilotage du projet avec Scrum

Pour développer notre application de gestion RH, nous avons choisi la méthode agile Scrum. Cette méthode repose sur :

- Une collaboration continue entre les membres de l'équipe.
- Des réunions régulières (daily meetings, sprint planning, revues de sprint).
- Un travail divisé en sprints (courtes périodes de développement avec des objectifs clairs).
- Un suivi constant de l'avancement du projet pour livrer un produit de qualité, étape par étape.

2.6 L'équipe Scrum

Dans notre projet, nous utilisons la méthode agile Scrum pour organiser le travail en équipe. Cette méthode nous aide à bien définir les rôles de chacun et à mieux collaborer pour atteindre nos objectifs.

Les rôles dans l'équipe Scrum :

1. Product Owner (PO)

- C'est la personne qui définit les besoins du projet.
- Il conçoit le produit (la plateforme RH) et s'assure qu'il répond aux attentes des utilisateurs et respecte les délais.
- Il travaille en collaboration avec les parties prenantes pour bien comprendre leurs besoins.
- Dans notre projet : BVMT.

2. Scrum Master (SM)

- Il veille au bon déroulement du projet en suivant les principes de Scrum.
- Il facilite la communication entre les membres de l'équipe.
- Il aide à résoudre les blocages et à coordonner les travaux.
- Dans notre projet : Mr. Dhyaddine BLEL.

3. Équipe de développement (Dev Team)

- Ce sont les personnes qui réalisent concrètement le projet.
- Elles peuvent avoir différents rôles : développeur, designer, rédacteur, analyste de données, etc.
- Elles participent activement à chaque sprint pour livrer des fonctionnalités.
- Dans notre projet : Sassi Fida, Turki Islem.

2.7 Backlog du produit

Le backlog produit est une liste organisée de toutes les fonctionnalités que l'on souhaite intégrer dans l'application. Il sert de guide pour l'équipe de développement et permet de prioriser les tâches les plus importantes.

Ce backlog est géré par le Product Owner, qui peut ajouter, modifier ou supprimer des éléments selon les retours des utilisateurs ou les besoins du projet.

NB : 1=Haute , 2= Moyenne , 3= Basse

ID	Rôle	User Story	Priorité
1	Administrateur	En tant qu'admin, je peux gérer les personnels (ajouter, activer, assigner rôle/service, modifier).	1
2	Administrateur	En tant qu'admin, je peux gérer les services (CRUD + assignation chef) et attribuer plusieurs chefs à un service en définissant un poids de priorité (1, 2, 3).	1
3	Administrateur	En tant qu'admin, je peux gérer les candidatures.	2
4	Administrateur	En tant qu'admin, je peux gérer les sociétés.	2
5	Administrateur	En tant qu'admin, je peux consulter mon profil.	3
6	Personnel	En tant que personnel, je peux créer un compte.	1
7	Personnel	En tant que personnel, je peux m'authentifier pour accéder à mon interface.	1
8	Personnel	En tant que personnel, je peux consulter mon profil.	2

9	Personnel	En tant que personnel, je peux gérer mes demandes (ajouter, modifier, supprimer, consulter).	1
10	Personnel	En tant que personnel, je reçois des notifications.	2
11	Personnel	En tant que personnel, je peux consulter le calendrier de mes congés acceptés.	2
12	Personnel	En tant que personnel, je peux télécharger mes documents administratifs.	2
13	Chef Hiérarchique	En tant que chef, je peux visualiser les personnels actifs de mes services, avec l'indication de mon poids dans chaque.	2
14	Chef Hiérarchique	En tant que chef, je peux approuver ou refuser les demandes, avec avis ou décision selon mon poids.	1
15	Chef Hiérarchique	En tant que chef, je peux consulter le calendrier des congés de mon équipe.	2
16	Chef Hiérarchique	En tant que chef, je peux télécharger des documents administratifs.	2
17	Chef Hiérarchique	En tant que chef, je peux gérer mes propres demandes(ajouter, modifier, supprimer, consulter).	2
18	RH	En tant que RH, je peux visualiser les personnels activés.	1
19	RH	En tant que RH, je peux traiter les demandes approuvées par le chef de poid 1 (Autorisation , Conge ,Formation).	1
20	RH	En tant que RH, je peux traiter ou refuser les demandes (avance, documents).	1
21	RH	En tant que RH, je peux consulter le calendrier des congés du personnel.	2
22	RH	En tant que RH, je peux télécharger des documents administratifs.	2
23	RH	En tant que RH, je peux gérer mes propres demandes(ajouter, modifier, supprimer, consulter).	2
24	Candidat	En tant que candidat, je peux consulter les offres d'emploi.	2
25	Candidat	En tant que candidat, je peux postuler à des offres par mail.	1
26	Candidat	En tant que candidat, je peux déposer mon CV.	2
27	Candidat	En tant que candidat, je peux suivre le statut de ma candidature.	2

TABLE 2.2 – User Stories par rôle et priorité

2.8 Diagramme de cas d'utilisation global

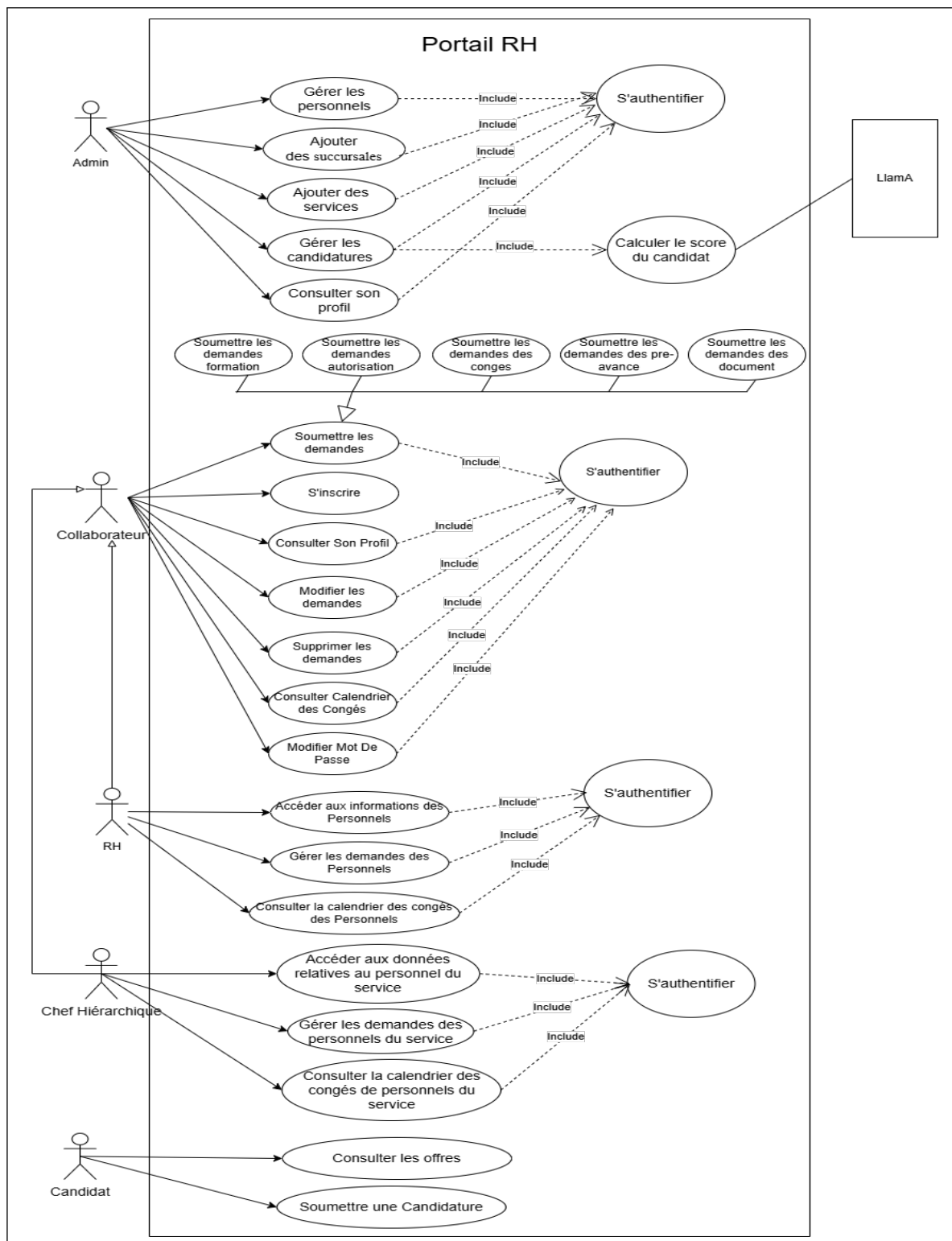


FIGURE 2.1 – Diagramme de cas d'utilisation global.

2.9 Planification des Sprints

Notre aventure de développement s’est articulée en trois sprints bien distincts :

Sprint	Fonctionnalités développées
Sprint 1	- Mise en place de l’authentification et de la gestion des utilisateurs - Gestion des succursales sociétés et services
Sprint 2	- Développement du système de gestion des demandes. - Système de notifications interactives : • Notifications internes (WebSocket) : alertes en temps réel • Notifications externes (Email) : création de compte, réinitialisation de mot de passe, blocage de compte
Sprint 3	- Intégration d’un module intelligent d’analyse de CV - Extraction automatique des compétences et évaluation des profils via IA - Implémentation d’un pipeline automatisé avec Python, LangChain et LLaMA - Mise en production et hébergement de la plateforme

TABLE 2.3 – Planification des sprints

Ce découpage nous a permis de progresser de manière agile, en enrichissant le projet à chaque étape, tout en assurant stabilité, interactivité et intelligence.

2.10 Environnement de travail

Dans cette section, nous allons mentionner tous les équipements, logiciels et outils utilisés pour concevoir notre application.

Caractéristique	MSI Cyborg 15 A12V	Acer Aspire A315-55G
Processeur (CPU)	12 ^e génération Intel® Core™ i5-12450H	Intel® Core™ i7-10510U (1.80 GHz – 2.30 GHz)
Mémoire vive (RAM)	16 Go DDR5	8 Go DDR4
Carte graphique (GPU)	NVIDIA® GeForce® RTX 2050	NVIDIA® GeForce® MX230
Stockage (SSD)	512 Go SSD NVMe	256 Go SSD NVMe
Système d’exploitation (SE)	Windows 11	Windows 11

TABLE 2.4 – Comparaison des environnements techniques utilisés

Environnement de développement et de modélisation

L'environnement logiciel regroupe l'ensemble des outils utilisés aux différentes étapes du travail : le développement, le déploiement et l'exécution de l'application. Il permet aux développeurs de créer, tester et mettre en œuvre des solutions logicielles.

Outils de développement :

- **VS Code** : Visual Studio Code est un éditeur de code source développé par Microsoft, léger et extensible, utilisé pour le développement web et logiciel.



FIGURE 2.2 – Logo VS Code

- **IntelliJ IDEA** : Environnement de développement intégré (IDE) pour Java, développé par JetBrains, offrant une assistance intelligente au codage.



FIGURE 2.3 – Logo IntelliJ IDEA

- **Spring Boot** : Framework Java qui simplifie la création d'applications autonomes prêtes à l'emploi avec une configuration minimale.



FIGURE 2.4 – Logo Spring Boot

- **React** : Bibliothèque JavaScript développée par Meta pour la création d'interfaces utilisateur dynamiques et réactives, principalement pour le web.
- **React Native** : Framework basé sur React, permettant de créer des applications mobiles natives pour Android et iOS.

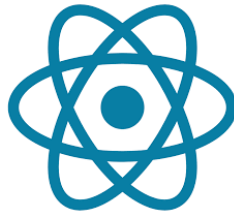


FIGURE 2.5 – Logo commun à React et React Native

- **Expo** : Outil simplifiant le développement d'applications mobiles avec React Native, en permettant de coder, tester et déployer rapidement sans configuration native complexe.



FIGURE 2.6 – Logo Expo

- **CSS (Cascading Style Sheets)** : Langage de feuille de style utilisé pour décrire la présentation des documents HTML : couleurs, polices, marges, disposition, animations, etc.



FIGURE 2.7 – Logo CSS3

- **MongoDB** : Base de données NoSQL orientée document, permettant de stocker les données au format JSON avec flexibilité.



FIGURE 2.8 – Logo MongoDB

Critère	Avantage de MongoDB
Flexibilité	Modèle évolutif sans schéma rigide, idéal pour des données RH en constante évolution.
Développement rapide	Manipulation simple des données, gain de temps avec Mongoose.
Adaptabilité	Bien adapté aux applications web, mobiles et cloud.
Performance	Bon traitement des grandes quantités de données (CV, candidatures, etc.).
Scalabilité	Extension horizontale facilitée pour gérer la montée en charge.
Optimisation	Réduction de la redondance via références, indexation et validation avec Mongoose.

TABLE 2.5 – Avantages de MongoDB pour le projet Portal RH

- **Postman** : Plateforme permettant de concevoir, tester et documenter des APIs REST de manière intuitive.

**FIGURE 2.9 – Logo Postman**

- **Python** : Langage de programmation interprété, multiparadigme et multiplateforme, favorisant la programmation impérative, fonctionnelle et orientée objet.

**FIGURE 2.10 – Logo Python**

- **LLaMA** : Grand modèle de langage développé par Meta pour les usages de traitement du langage naturel.



FIGURE 2.11 – Logo LLaMA

- **LangChain** : Framework pour développer des applications alimentées par des modèles linguistiques, facilitant l'intégration avec des sources de données, API et outils externes.

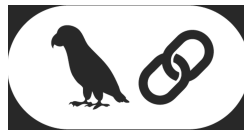


FIGURE 2.12 – Logo LangChain

- **Groq** : Entreprise spécialisée dans les architectures matérielles AI, optimisées pour accélérer les tâches d'apprentissage automatique.



FIGURE 2.13 – Logo Groq

- **Render** : Plateforme cloud permettant de déployer et héberger des applications web, APIs, sites statiques et bases de données.



FIGURE 2.14 – Logo Render

- **Overleaf** : Éditeur collaboratif en ligne pour rédiger des documents scientifiques en LaTeX.



FIGURE 2.15 – Logo Overleaf

- **PyMuPDF** : Bibliothèque Python haute performance pour l'extraction, l'analyse, la conversion et la manipulation de documents PDF (et autres formats).



FIGURE 2.16 – Logo PyMuPDF

- **GitHub** : Plateforme de gestion de code source basée sur Git, permettant de collaborer, versionner et partager des projets en équipe.



FIGURE 2.17 – Logo GitHub

Outils de conception :

- **Draw.io** : Outil en ligne gratuit pour créer des organigrammes, diagrammes de processus, modèles UML, schémas de bases de données et réseaux.



FIGURE 2.18 – Logo Draw.io

- **PlantUML** : Outil open-source permettant de créer des diagrammes à partir de texte brut, idéal pour les diagrammes de séquence et les descriptions précises d'interactions.

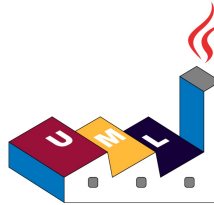


FIGURE 2.19 – Logo PlantUML

2.11 Architecture de l'application

2.11.1 Architecture logique

Le développement de notre application **RH pour la Gestion des Demandes des employés et des Recrutements** repose sur une architecture en couches (*Layered Architecture*), adaptée au contexte des applications Web modernes avec Spring Boot. Bien que certains ajustements aient été réalisés pour simplifier la gestion de la logique métier, cette approche garantit une séparation claire des responsabilités, facilite la maintenance du code et renforce la testabilité.

a) Définition :

L'architecture en couches consiste à organiser le code source en différents niveaux, où chaque couche est responsable d'un aspect précis de l'application. Cette organisation favorise la clarté, la modularité et la réutilisabilité des composants.

b) Couches principales de l'application :

- **Couche Présentation (Controller)** : Cette couche est responsable de la réception des requêtes HTTP provenant de l'interface utilisateur ou d'autres services clients (ex : application mobile). Elle délègue le traitement à la couche métier et retourne les réponses appropriées.

- **Couche Métier (Service)** : Elle contient la logique métier propre à Portail RH. Par exemple, c'est dans cette couche que sont gérés le traitement des pièces jointes ou l'évaluation automatique des CV via l'IA.
- **Couche Modèle (Model)** : Elle définit les entités métiers (comme `Personnel`, `demandes`) et les objets de transfert de données utilisés dans toute l'application.
- **Couche Accès aux données (Repository)** : Assure la communication avec la base de données en utilisant Spring Data JPA. Elle gère les opérations de création, lecture, mise à jour et suppression (CRUD) des entités.
- **Couche Intelligence Artificielle (IA Service)** : Une extension spécialisée de la couche Service, dédiée aux interactions avec les scripts Python et les modèles d'IA (comme LLaMA). Cette couche prend en charge l'analyse des CV, l'extraction des compétences.
- **Couche Utilitaires et Configuration et Exceptions** : Elle comprend des classes transversales comme la configuration de sécurité, notifications, gestion de fichiers, ou la gestion des exceptions personnalisées.

c) Avantages de cette architecture :

- **Séparation des préoccupations** : Chaque couche a un rôle bien défini, ce qui améliore la lisibilité et la maintenance du code.
- **Modularité** : Les composants peuvent être modifiés ou remplacés sans impacter les autres couches.
- **Testabilité accrue** : Les tests unitaires ou d'intégration peuvent cibler spécifiquement une couche.
- **Évolutivité** : Facilité d'intégration de nouvelles fonctionnalités (ex : ajout de chatbot RH ou d'un système AI plus avancé pour les candidats).
- **Sécurité renforcée** : L'accès direct à la base de données est limité à des interfaces précises, et la logique métier ne dépend pas de l'interface utilisateur.

Cette structure a permis de bien organiser le code, d'accélérer le développement en équipe, et de préparer l'application à de futures évolutions sans remettre en cause l'existant.

2.11.2 Architecture du projet Front-End

L'application front-end a été développée avec React.js en adoptant une architecture modulaire basée sur les fonctionnalités (*feature-based architecture*). Cette structure consiste à organiser les composants, styles et logiques métier selon les modules fonctionnels de l'application, tels que *admin*, *RH*, *Chef_hierarchique*, ou encore *collaborateurs*.

Chaque dossier de fonctionnalité contient les composants React spécifiques à son domaine, ce qui favorise une meilleure maintenabilité du code, une évolutivité facilitée et une répartition claire des responsabilités. D'autres dossiers comme *services*, *hooks*, *contexts* et *assets* contiennent des ressources partagées entre les différentes fonctionnalités.

Cette approche permet également une collaboration efficace entre les membres de l'équipe en limitant les conflits liés au développement parallèle.

2.11.3 Flux de données et traitement

Le flux de données se fait par des appels API entre le frontend et le backend. Lorsque l'utilisateur interagit avec l'application, le frontend envoie des requêtes au backend via des API. Le backend, à travers ses contrôleurs, gère la logique métier et retourne les données au frontend sous forme de JSON. Le frontend utilise ensuite ces données pour mettre à jour l'interface utilisateur, garantissant une expérience interactive et dynamique.

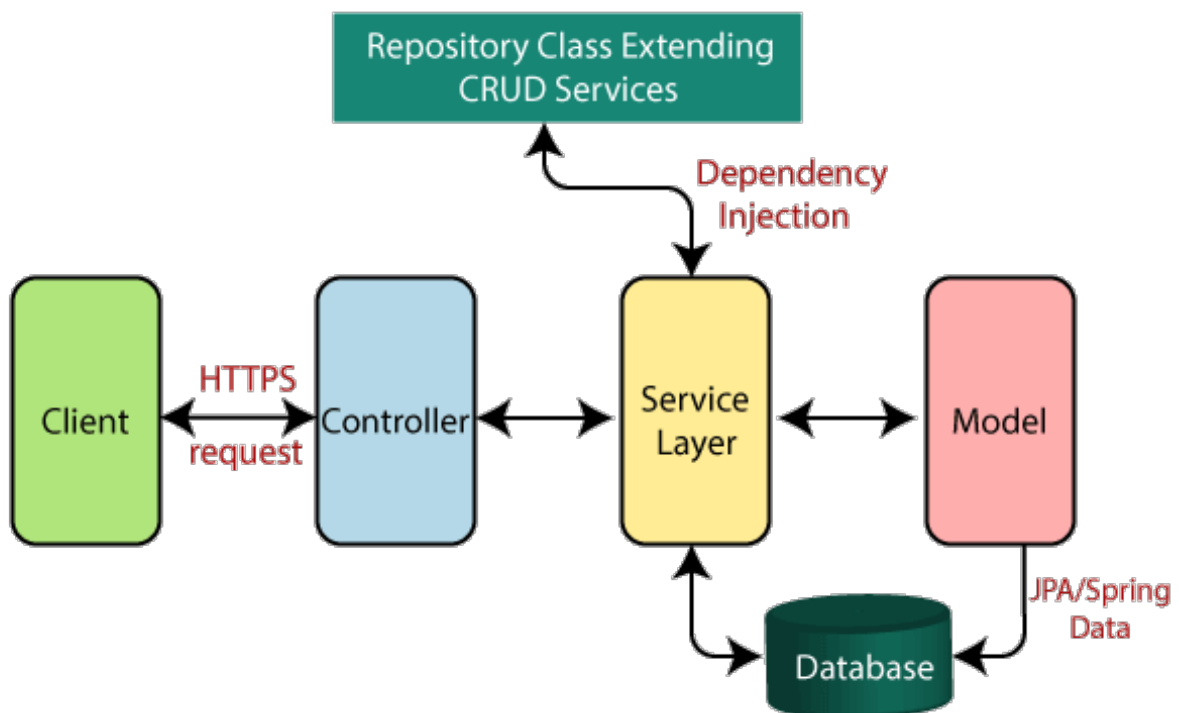


FIGURE 2.20 – Architecture de l'application

Chapitre 3 : Conception et réalisation du sprint 1

3.1 Sprint 1

Ce sprint d'une durée d'un mois, consacré au développement des composants essentiels : authentification, gestion des sociétés, gestion des services et gestion des personnels. Cette phase a permis de consolider l'architecture technique tout en maintenant un échange régulier avec les parties prenantes.

- **Système d'authentification :**

- Mécanisme de connexion/déconnexion sécurisé
- Gestion des rôles et permissions
- Procédure de récupération de mot de passe

- **Gestion des succursale sociétés :**

- ajout, modification, consultation et suppression.

- **Gestion des services :**

- ajout, modification, consultation et suppression.

- **Profil utilisateur :**

- Conception de l'interface du profil utilisateur
- Mise en place de la structure de données des profils
- Implémentation de la fonctionnalité de modification du mot de passe

Ces fonctionnalités constituent la fondation sur laquelle s'appuieront les développements ultérieurs.

3.2 User Stories

Id	Titre	User Story	Priorité
US1	Authentification	En tant que personnel, je veux m'authentifier sur la plateforme afin d'accéder à mon espace personnel de manière sécurisée.	Haute
US2	Gérer les succursales	En tant qu'administrateur, je veux pouvoir gérer les succursales des sociétés (ajouter, consulter, modifier, supprimer).	Haute
US3	Gérer les services	En tant qu'administrateur, je veux pouvoir gérer les services (ajouter, consulter, modifier, supprimer).	Haute
US4	Gestion des comptes	En tant qu'administrateur, je veux gérer les comptes du personnel afin d'assurer une administration efficace et sécurisée.	Haute
US5	Filtrage du personnel	En tant qu'administrateur, je veux filtrer les membres du personnel par nom, matricule, statut, service et rôle afin de faciliter la gestion.	Moyenne

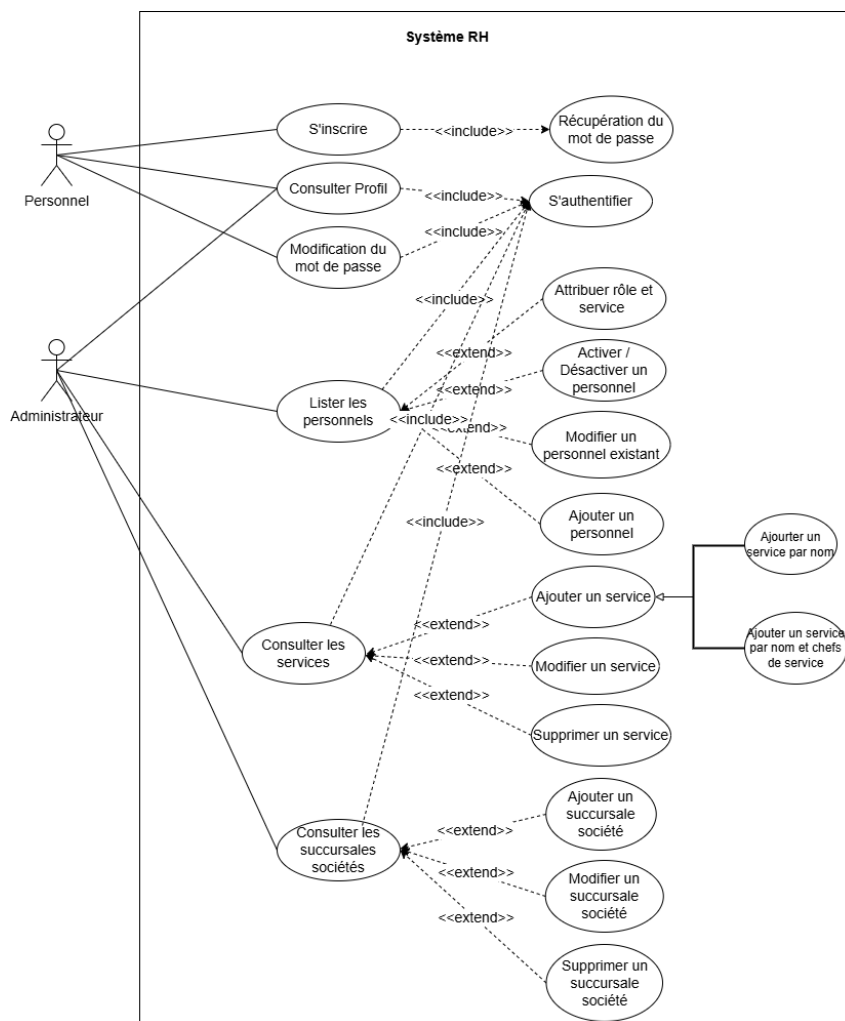
Id	Titre	User Story	Priorité
US6	Consultation du profil	En tant que personnel, je veux consulter mon profil et mes informations personnelles, mon rôle, mon service et mes chefs de service, et changer mon mot de passe.	Moyenne
US7	Thème visuel	En tant qu'utilisateur, je veux choisir un thème (clair/sombre) pour améliorer mon confort d'utilisation.	Moyenne

TABLE 3.1 – Liste des User Stories du sprint 1

3.3 Diagrammes

3.3.1 Diagramme de cas d'utilisation du sprint 1

La Figure 3.1 illustre les interactions principales couvertes lors de ce premier sprint.

**FIGURE 3.1 – Diagramme de cas d'utilisation du sprint 1**

3.3.2 Diagramme de classes du sprint 1

La Figure 3.2 présente la structure des classes définies pour le sprint 1, mettant en avant les relations entre les entités principales du système.

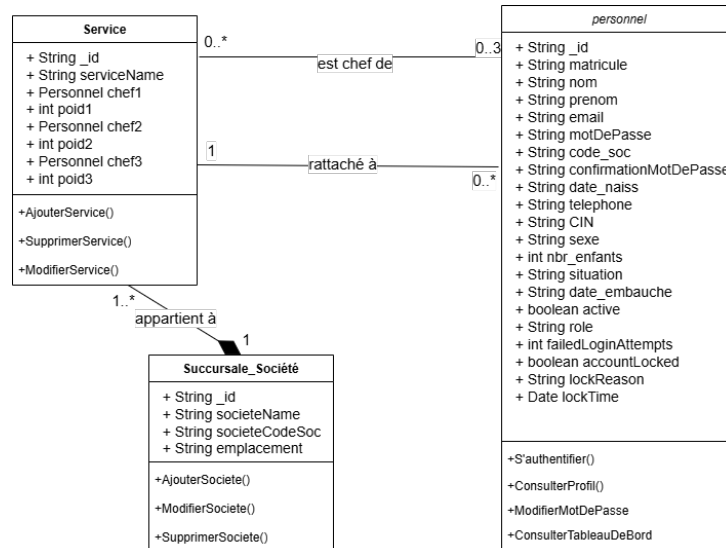


FIGURE 3.2 – Diagramme de classes du sprint 1

3.4 Description textuelle du cas "Login Admin"

Élément	Authentification Admin
Acteurs concernés	— Administrateur
Préconditions	— Les informations d'admin sont disponibles (nom, prénom, email, matricule, etc.).
Postconditions	— Admin authentifié
Scénario Nominal	1. L'administrateur remplit un formulaire login avec ses informations. 2. Un message success s'affiche.

Élément	Authentification Admin (suite)
Scénario Alternatif	1. a) Si matricule est invalide, un message d'erreur est affiché. b) Si mot de passe est invalide, un message d'erreur est affiché. c) Après trois tentatives de connexion incorrectes, un message d'erreur est affiché et un e-mail de sécurité est envoyé à l'utilisateur et son compte se bloque.

TABLE 3.2 – Description du cas « Login admin »

3.4.1 Diagramme du cas "Login Admin"

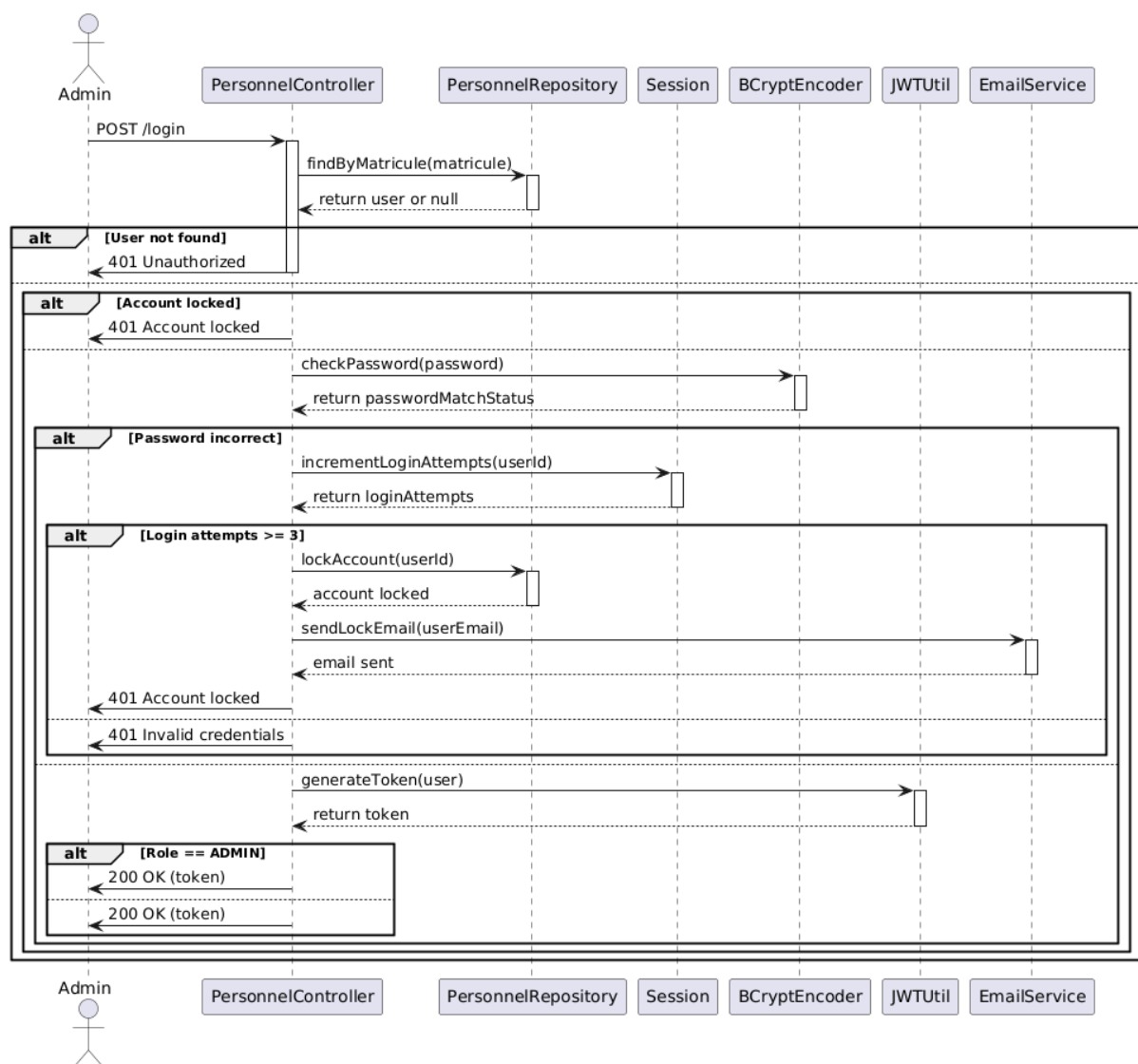


FIGURE 3.3 – Diagramme de sequence du sprint 1 Login admin

3.5 Description textuelle du cas "Ajouter succursale sociétés"

Élément	Description
Acteurs concernés	• Administrateur
Préconditions	• Administrateur authentifié
Post conditions	• Succursale Société Ajoutée
Scénario nominal	1. L'administrateur accède à l'interface de gestion des sociétés. 2. Il clique sur "Ajouter une société". 3. Il saisit les informations de la société (code, adresse, etc.). 4. Il soumet le formulaire. 5. Le système enregistre les données et affiche un message de succès.
Scénarios d'erreur	3.a) Champs manquants ou invalides : le système affiche un message d'erreur.

Tableau 3.5 : Cas d'utilisation – Ajouter une succursale société

3.5.1 Diagramme du cas "Ajouter succursale Société"

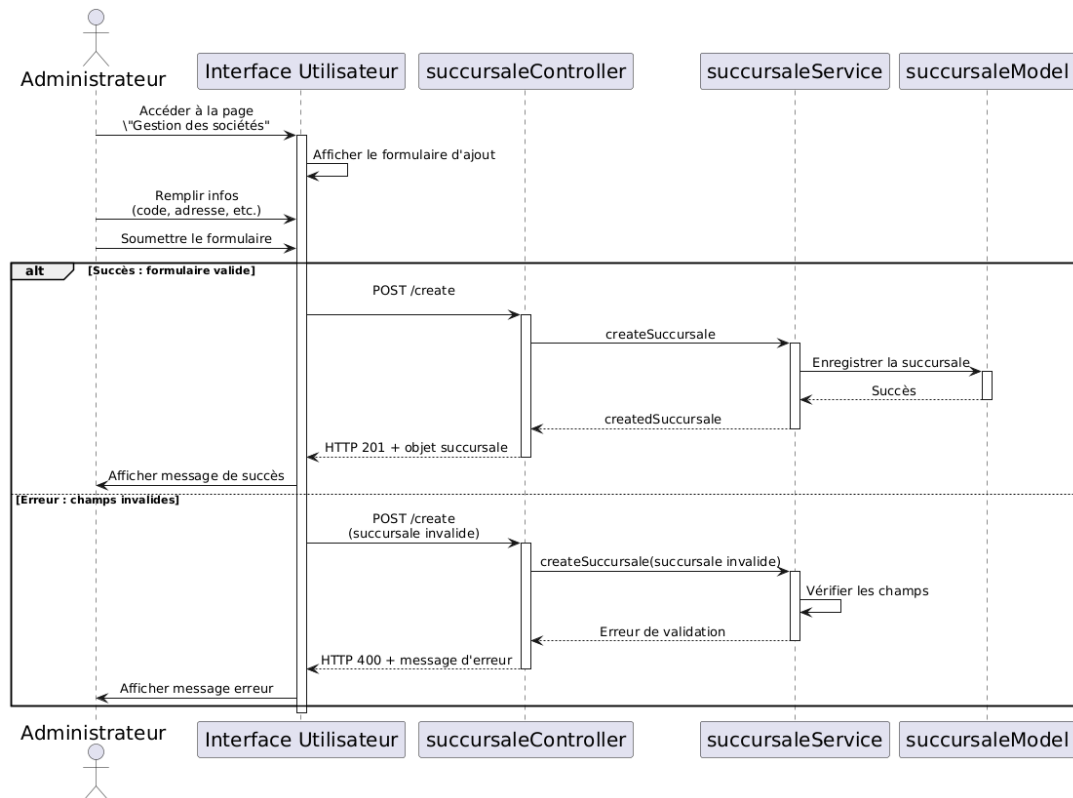


FIGURE 3.4 – Diagramme de sequence du sprint 1 du cas "Ajouter succursale Société"

3.6 Description textuelle du cas "Modifier succursale société"

Élément	Description
Acteurs concernés	Administrateur
Préconditions	Administrateur authentifié
Postconditions	Succursale société modifiée

Élément	Description
Scénario nominal	1. L'administrateur accède à la liste des sociétés. 2. Il sélectionne la société à modifier. 3. Il modifie les champs souhaités (ex : nom, emplacement). 4. Il valide les modifications. 5. Le système met à jour les informations et affiche un message de succès.
Scénarios d'erreur	3.a) Champs invalides : le système affiche un message d'erreur.

TABLE 3.3 – Cas d'utilisation – Modifier une succursale société

3.6.1 Diagramme du cas "Modifier succursale Société"

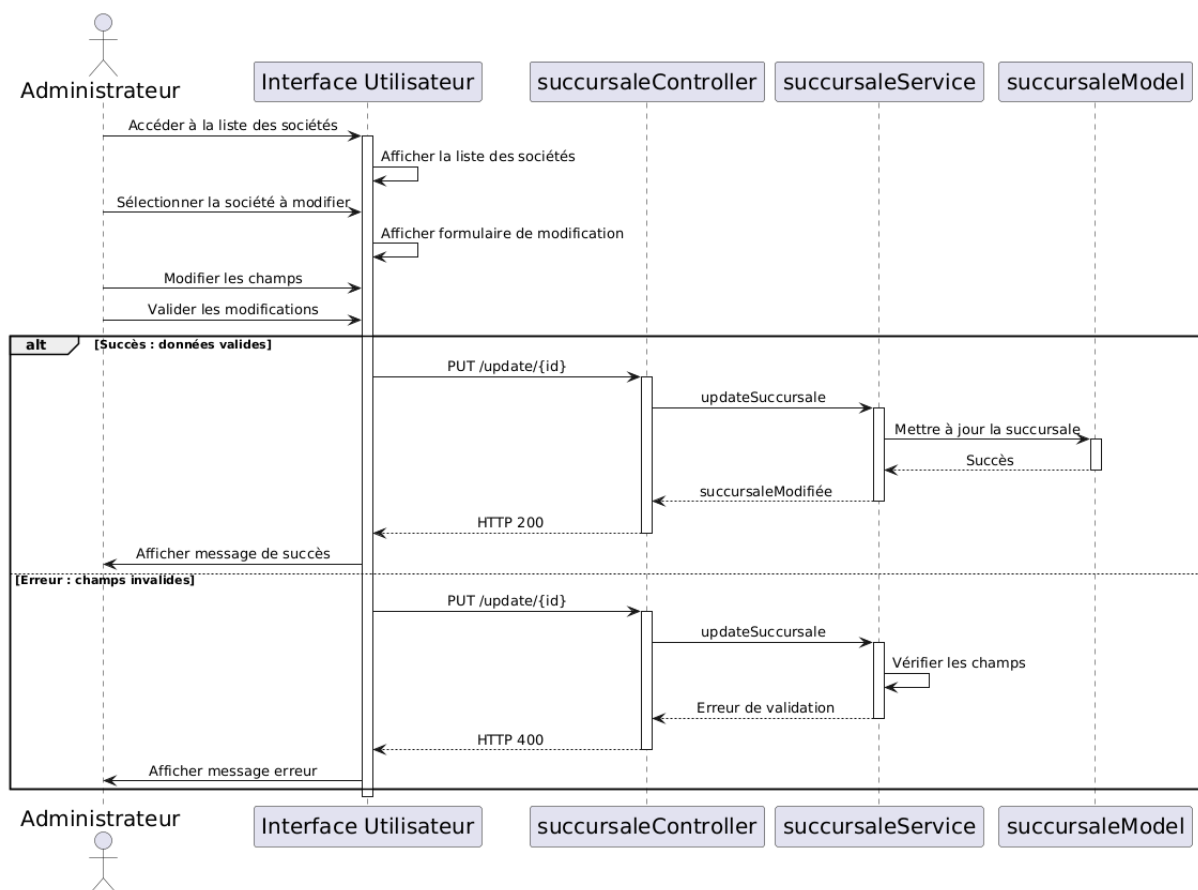


FIGURE 3.5 – Diagramme de séquence du sprint 1 du cas "Modifier une succursale société"

3.7 Description textuelle du cas "Supprimer succursale société"

Élément	Description
Acteurs concernés	Administrateur
Préconditions	Administrateur authentifié
Post conditions	Succursale Société Supprimée
Scénario nominal	- L'administrateur accède à la liste des sociétés. - Il sélectionne une société à supprimer. - Il confirme l'action de suppression. - Le système supprime la société et affiche un message de succès.
Scénarios d'erreur	L'administrateur annule la confirmation : la suppression est interrompue.

TABLE 3.4 – Cas d'utilisation – Supprimer une succursale société

3.7.1 Diagramme du cas "Supprimer succursale Société"

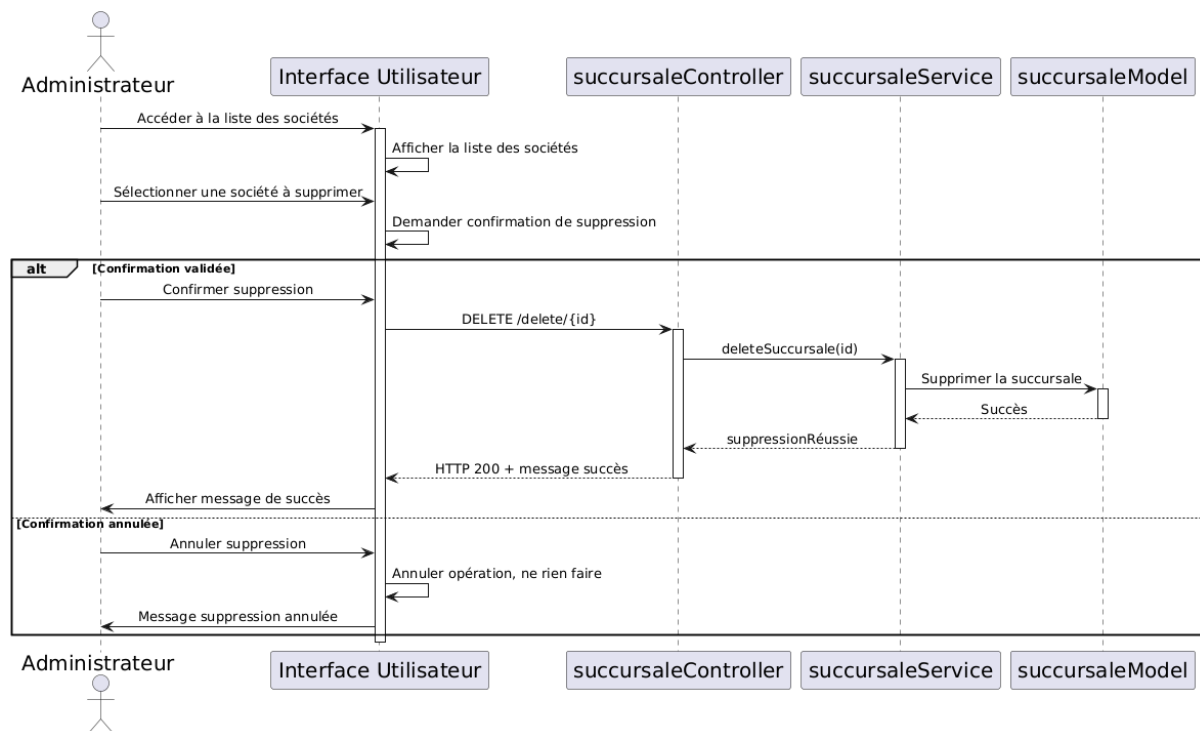


FIGURE 3.6 – Diagramme de sequence du sprint 1 du cas "Supprimer une succursale société"

3.8 Description textuelle du cas "Ajouter un service par nom"

Élément	Description
Acteurs concernés	• Administrateur RH
Préconditions	• Administrateur authentifié
Post conditions	• Service Ajoutée
Scénario nominal	1. L'administrateur accède à l'interface de gestion des services. 2. Il clique sur "Ajouter un service par nom". 3. Il saisit les informations du service (nom du service). 4. Il soumet le formulaire. 5. Le système enregistre les données et affiche un message de confirmation.
Scénarios d'erreur	3.a) Champs manquants ou invalides : le système affiche un message d'erreur.

TABLE 3.5 – Cas d'utilisation – Ajouter un service par nom

3.8.1 Diagramme du cas "Ajouter un service par nom"

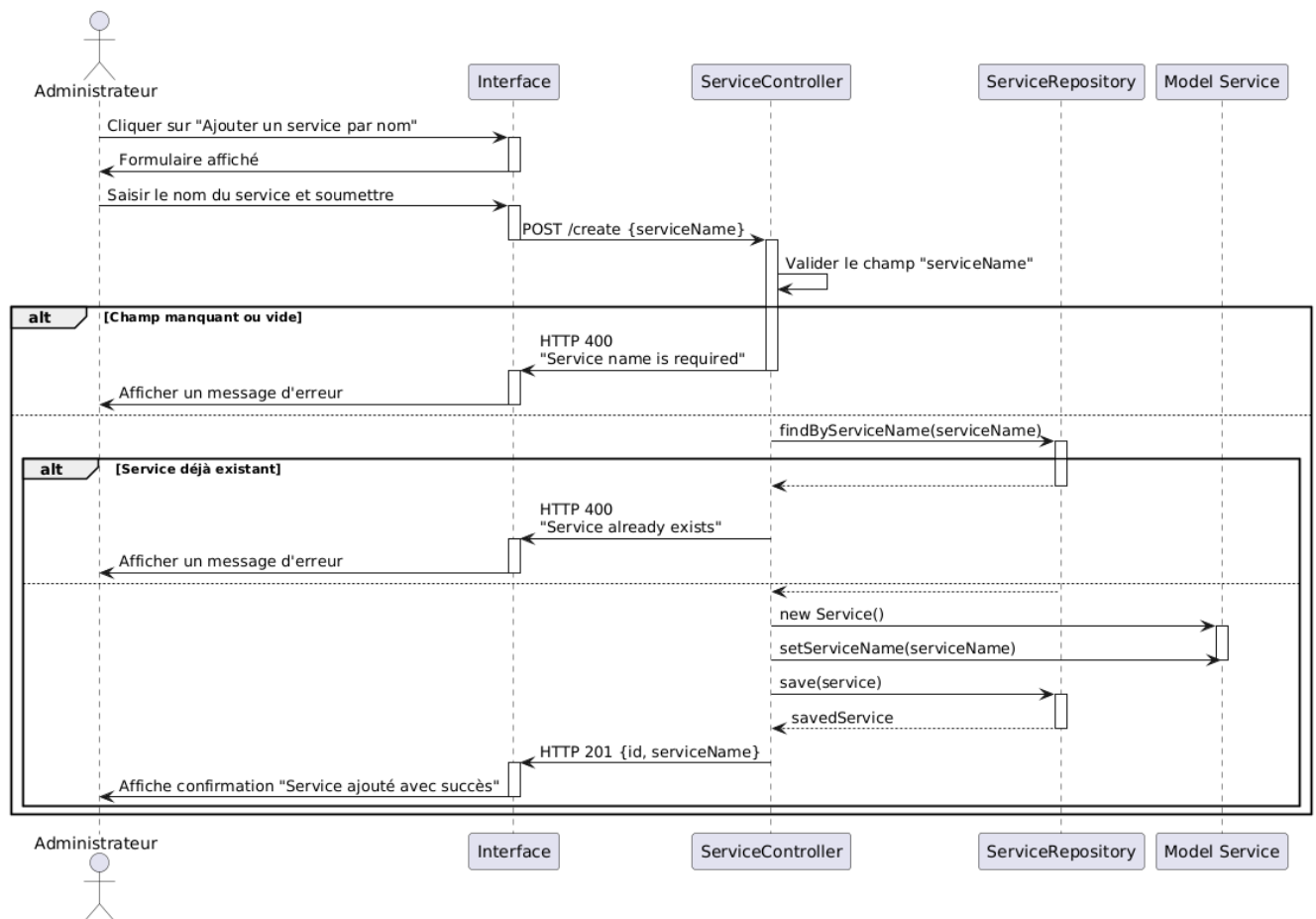


FIGURE 3.7 – Diagramme de sequence du sprint 1 du cas "Ajouter un service par nom"

3.9 Description textuelle du cas "Ajouter un service par nom et chefs de service"

Élément	Description
Acteurs concernés	Administrateur RH
Préconditions	Administrateur authentifié
Scénario nominal	1. L'administrateur accède à l'interface de gestion des services. 2. Il clique sur "Ajouter un service par nom et chefs de service". 3. Il saisit les informations du service (nom, chefs avec poids). 4. Il soumet le formulaire. 5. Le système enregistre les données et affiche un message de confirmation.

Élément	Description
Scénarios d'erreur	3.a) Champs manquants ou invalides : le système affiche un message d'erreur.

TABLE 3.6 – Cas d'utilisation : Ajouter un service par nom et chefs de service

3.9.1 Diagramme du cas "Ajouter un service par nom et chefs de service"

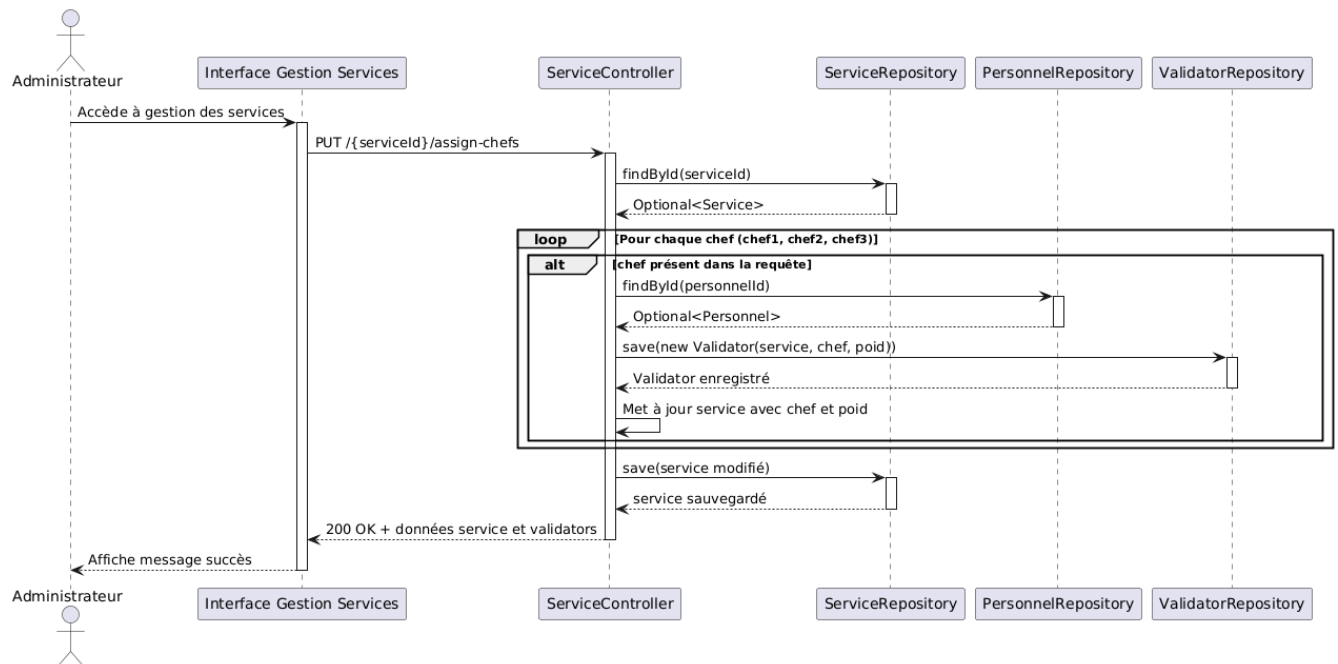


FIGURE 3.8 – Diagramme de sequence du sprint 1 du cas "Ajouter un service par nom et chefs de service"

3.10 Description textuelle du cas "Ajouter un Personnel"

Élément	Ajouter Personnel
Acteurs concernés	— Administrateur — Personnel
Préconditions	— L'administrateur doit être authentifié. — Les informations du personnel sont disponibles (nom, prénom, email, etc.).

Élément	Ajouter Personnel (suite)
Postconditions	— Un compte utilisateur est créé, et un matricule est généré. — Un email de bienvenue et un lien d'activation sont envoyés au personnel.
Scénario Nominal	1. L'administrateur remplit un formulaire avec les informations du personnel. 2. Le système génère un matricule unique et valide l'email de l'utilisateur. 3. Un email est envoyé avec un lien d'activation.
Scénario Alternatif	1.a) Si l'email est invalide, un message d'erreur est affiché.

TABLE 3.7 – Description du cas d'utilisation "Ajouter Personnel"

3.10.1 Diagramme du cas "Ajouter un Personnel"

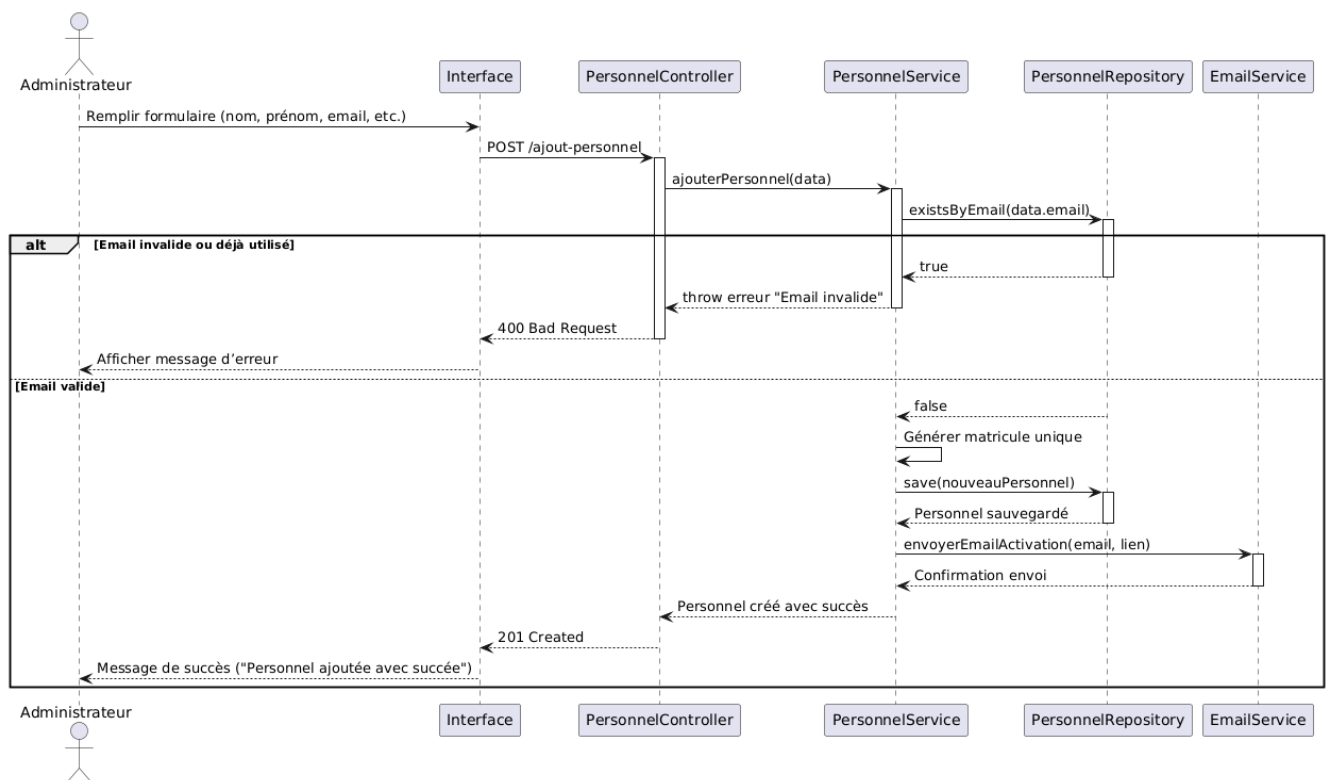


FIGURE 3.9 – Diagramme de séquence du sprint 1 ajouter personnel

3.11 Description textuelle du cas "Inscription Personnel"

Élément	Inscription Personnel
Acteurs concernés	— Personnel
Préconditions	— Les informations d'un personnel sont disponibles (matricule, email).
Postconditions	— Personnel enregistré
Scénario Nominal	1. Le personnel remplit un formulaire Inscription avec ses informations. 2. Un message success s'affiche. 3. Un mail de confirmation est envoyée.
Scénario Alternatif	1.a) Si mot de passe est faible, un message d'erreur est affiché. 1.b) Si matricule est invalide, un message d'erreur est affiché. 1.c) Si email est invalide, un message d'erreur est affiché. 1.d) Après trois tentatives de connexion incorrectes, un message d'erreur est affiché et un e-mail de sécurité est envoyé à l'utilisateur et son compte se bloque.

TABLE 3.8 – Description du cas d'utilisation "Inscription Personnel"

3.11.1 Diagramme du cas "Inscription Personnel"

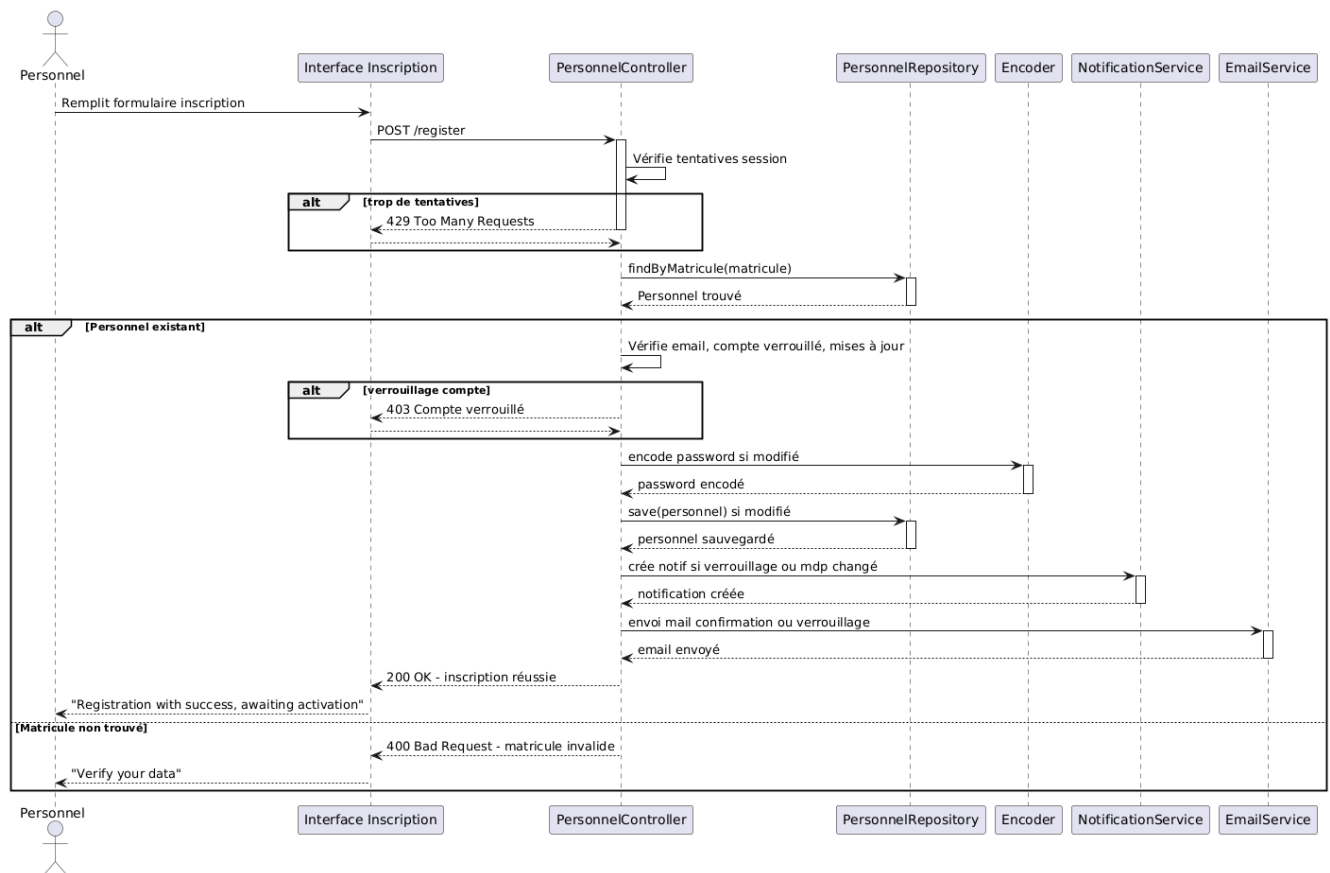


FIGURE 3.10 – Diagramme de séquence du sprint 1 Inscription personnel

3.12 Description textuelle du cas "Activer un Personnel"

Élément	Activer Personnel
Acteurs concernés	— Administrateur — Personnel
Préconditions	— Le personnel doit avoir complété l'enregistrement et soumis ses informations. — L'administrateur doit être authentifié pour activer un compte.

Élément	Activer Personnel (suite)
Scénario Nominal	1. L'administrateur attribue un rôle et un service au personnel. 2. L'administrateur vérifie les informations et approuve l'activation du personnel. 3. L'administrateur modifie les informations du personnel. 4. L'activation est confirmée et un email est envoyé pour informer le personnel.
Scénario Alternatif	1.a) Si les informations sont incorrectes, l'activation est refusée. 1.b) L'administrateur doit corriger les erreurs avant de pouvoir activer le personnel.
Postconditions	— Le personnel a accès à son compte et peut se connecter. — Le rôle et le service de l'utilisateur sont définis dans le système.

TABLE 3.9 – Scénarios du cas d'utilisation "Activer Personnel"

3.12.1 Diagramme du cas "Activer un Personnel"

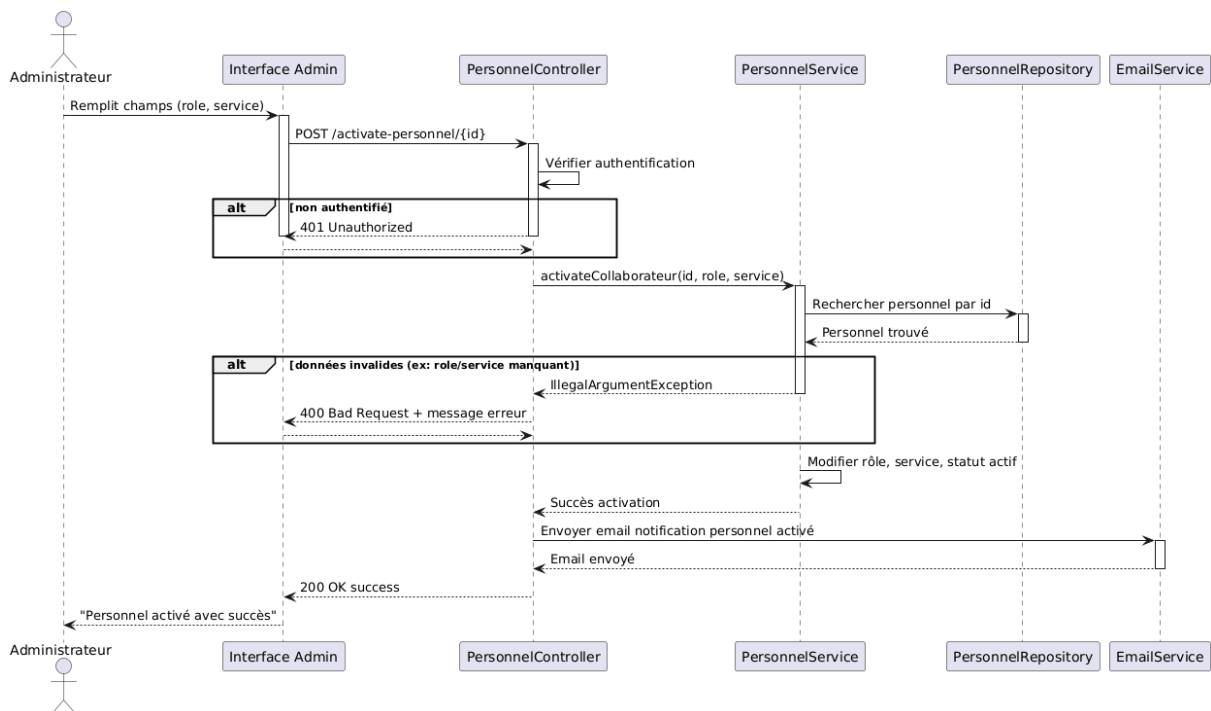


FIGURE 3.11 – Diagramme de séquence du sprint 1 activer Personnel

3.13 Description textuelle du cas "Modifier un Personnel"

Élément	Modifier un Personnel
Acteurs concernés	— Admin
Préconditions	— Personnel enregistré.
Postconditions	— Personnel modifié
Scénario Nominal	1. L'admin remplit un formulaire de modification avec les informations d'un personnel. 2. Un message de succès s'affiche.
Scénario Alternatif	1.a) Si un personnel n'est pas encore enregistré, le bouton est désactivé.

TABLE 3.10 – Description du cas d'utilisation "Modifier un Personnel"

3.13.1 Diagramme du cas "Modifier un Personnel"

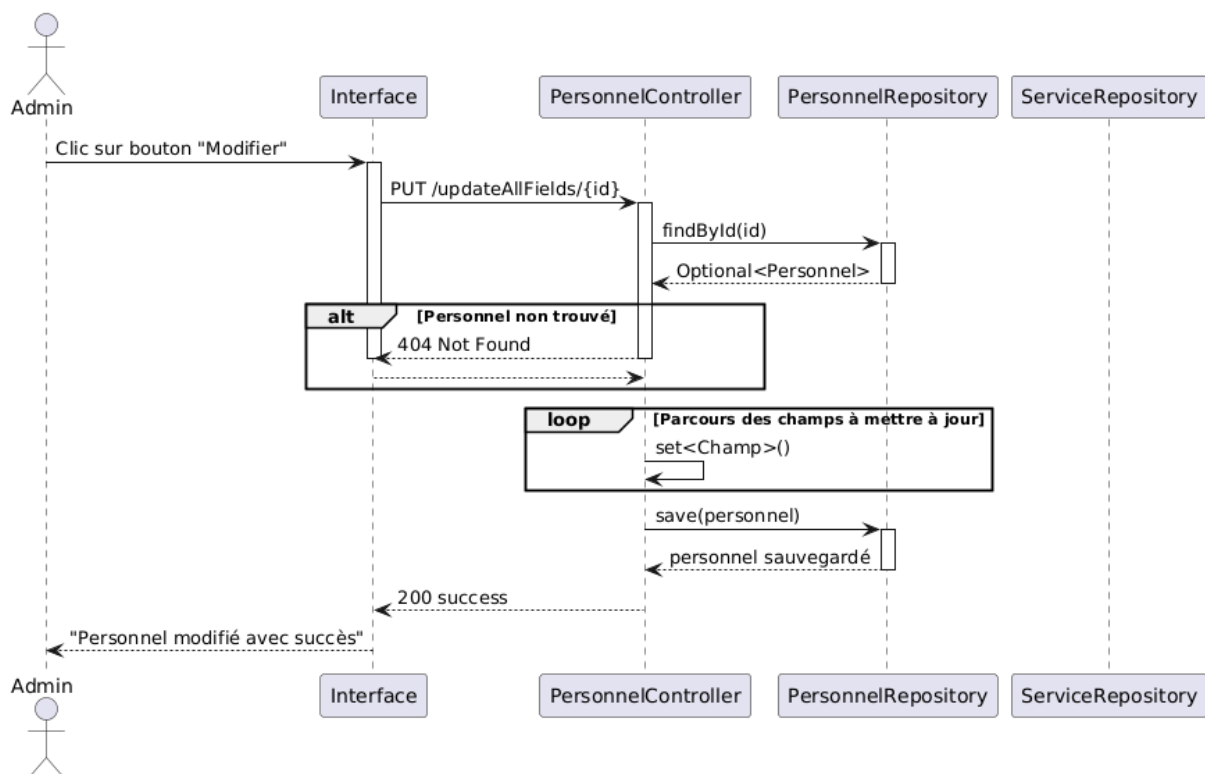


FIGURE 3.12 – Diagramme de séquence du sprint 1 Modifier Personnel

3.14 Description textuelle du cas "Login Personnel"

Élément	Authentification Personnel
Acteurs concernés	— Personnel
Préconditions	— Les informations d'un Personnel sont disponibles (nom, prénom, email, matricule, etc.).
Postconditions	— Personnel authentifié
Scénario Nominal	1. Le Personnel remplit un formulaire login avec ses informations. 2. Un message de succès s'affiche.
Scénario Alternatif	1.a) Si matricule est invalide, un message d'erreur est affiché. 1.b) Si mot de passe est invalide, un message d'erreur est affiché. 1.c) Après trois tentatives de connexion incorrectes, un message d'erreur est affiché, un e-mail de sécurité est envoyé à l'utilisateur et son compte se bloque.

TABLE 3.11 – Description du cas d'utilisation "Login Personnel"

3.14.1 Diagramme associés au cas "Login Personnel"

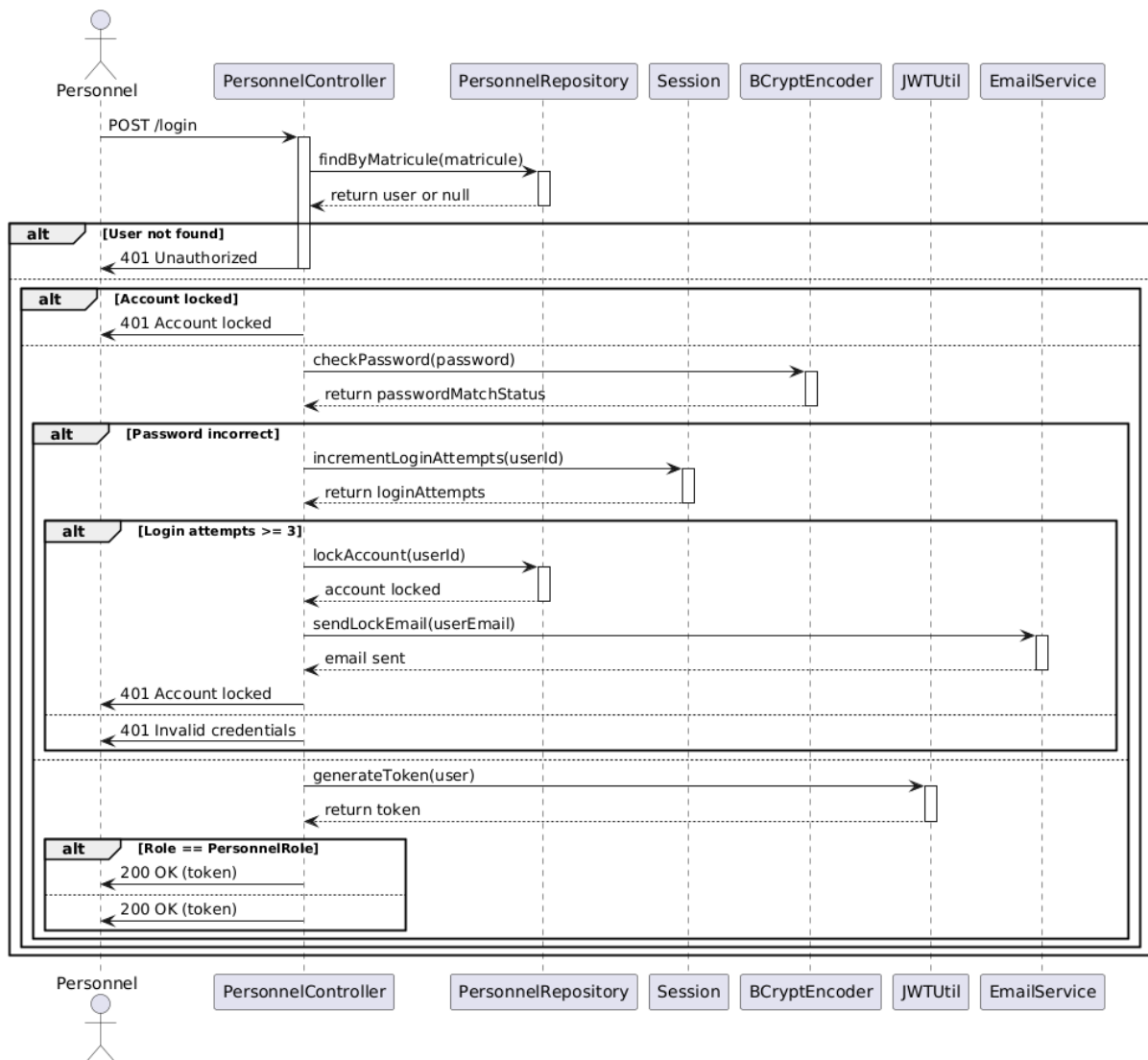


FIGURE 3.13 – Diagramme de séquence du sprint 1 — Login Personnel

3.15 Description textuelle du cas "Modification du mot de passe"

Cas d'utilisation	Modification du mot de passe
Acteurs	Personnel
Précondition	L'utilisateur doit être authentifié
Post-condition	L'interface de profil est affichée
Scénario nominal	1. L'utilisateur clique sur le bouton "Profil". 2. Le système affiche l'interface de profil. 3. L'utilisateur clique sur le bouton "Modifier le mot de passe". 4. Le système affiche un formulaire de modification de mot de passe. 5. L'utilisateur saisit l'ancien mot de passe et le nouveau mot de passe. 6. Le système vérifie les informations et confirme la modification.
Scénario alternatif	1.a) Si le mot de passe actuel est incorrect, un message d'erreur est affiché. 1.b) Si le nouveau mot de passe est trop faible, un message d'erreur est affiché. 1.c) Si la confirmation du mot de passe ne correspond pas, un message d'erreur est affiché.

TABLE 3.12 – Description textuelle du cas d'utilisation "Modification du mot de passe"

3.15.1 Diagramme de séquence du cas "Modification du mot de passe"

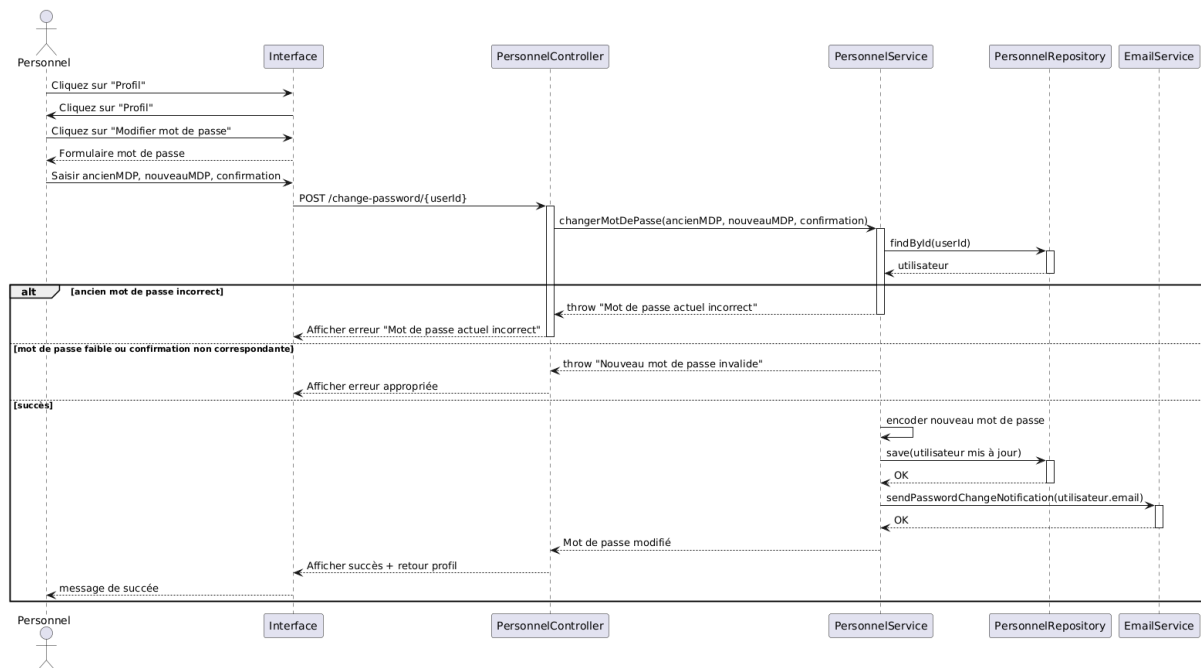


FIGURE 3.14 – Diagramme de séquence du sprint 1 Modification du mot de passe

3.16 Résultats obtenus

3.16.1 Authentification

Les figures 3.16 et 3.18 présentent les interfaces d'authentification développées pour les versions Web et Mobile en thème clair.

3.16.1.1 Interfaces Web

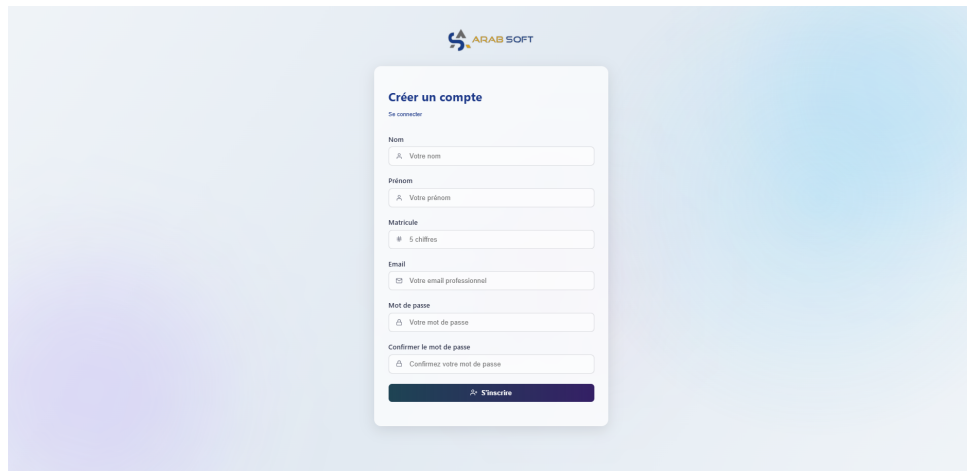


FIGURE 3.15 – Interface Web d'inscription - Thème clair

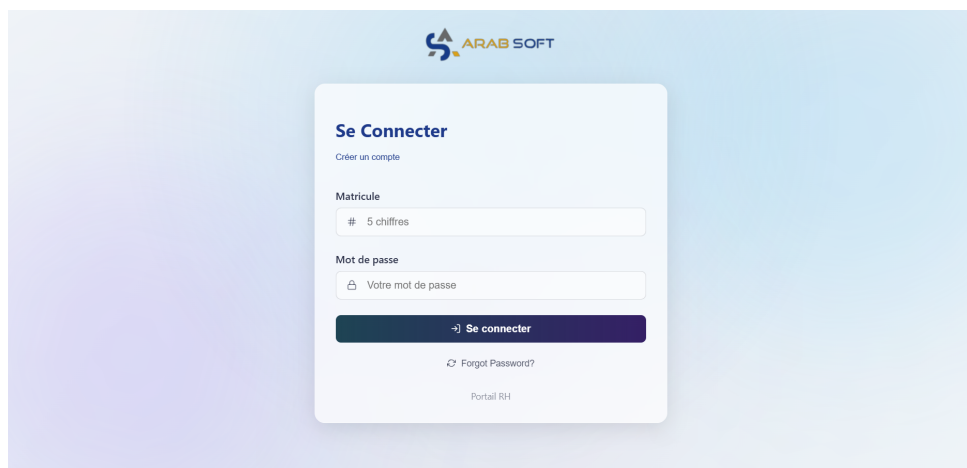
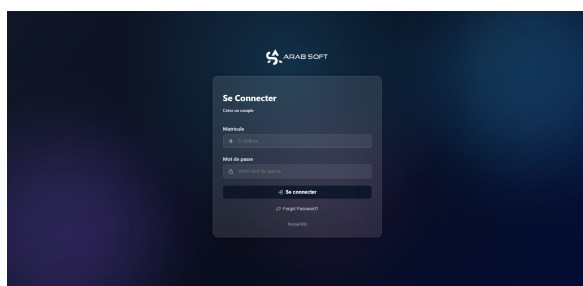
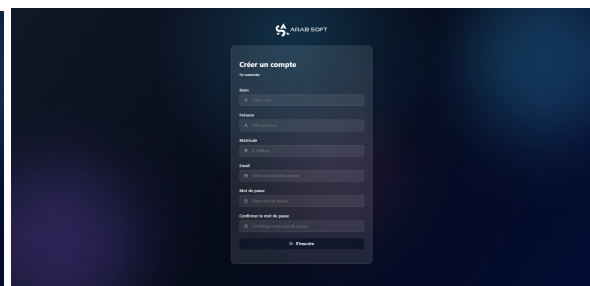


FIGURE 3.16 – Interface Web de connexion - Thème clair



(a) Interface Web de connexion -Thème sombre



(b) Interface Web d'inscription -Thème sombre

FIGURE 3.17 – Interfaces de profil personnel

3.16.1.2 Interfaces Mobile

4:51 PM

ARAB SOFT

Créer un compte

Remplissez le formulaire pour créer votre compte

[Se connecter](#)

Nom

Votre nom

Prénom

Votre prénom

Matricule

5 chiffres

Email

Votre email professionnel

Mot de passe

Votre mot de passe

Confirmer le mot de passe

Confirmez votre mot de passe

(a) Inscription - Thème clair

4:42 PM

ARAB SOFT

Connexion au portail RH

Entrez vos identifiants pour accéder à votre espace personnel

[Créer un compte](#)

Matricule

5 chiffres

Mot de passe

Votre mot de passe

[Se connecter](#)

Portail RH

(b) Connexion - Thème clair

FIGURE 3.18 – Interfaces Mobiles d'authentification - Thème clair

4:41 PM

ARAB SOFT

Connexion au portail RH

Entrez vos identifiants pour accéder à votre espace personnel

[Créer un compte](#)

Matricule

5 chiffres

Mot de passe

Votre mot de passe

[Se connecter](#)

Portail RH

(a) Thème sombre

4:51 PM

ARAB SOFT

Créer un compte

Remplissez le formulaire pour créer votre compte

[Se connecter](#)

Nom

Votre nom

Prénom

Votre prénom

Matricule

5 chiffres

Email

Votre email professionnel

Mot de passe

Votre mot de passe

Confirmer le mot de passe

Confirmez votre mot de passe

(b) Thème sombre

FIGURE 3.19 – Interfaces Mobiles d'authentification - Thème sombre

3.16.2 Admin - Gestion des personnels

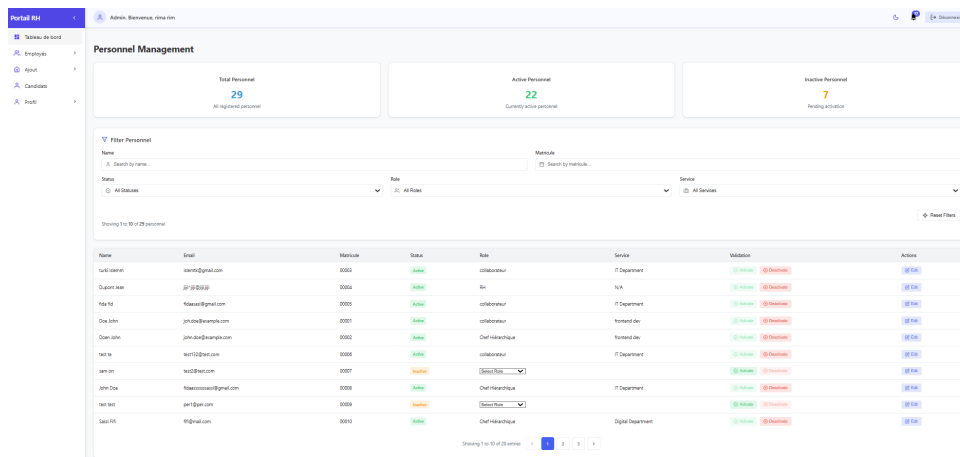


FIGURE 3.20 – Interface Web de la liste des personnels - Thème clair

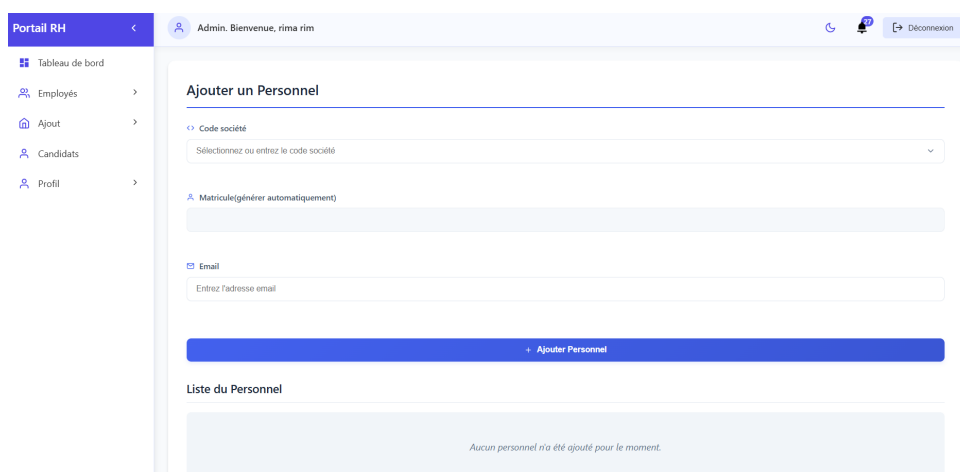
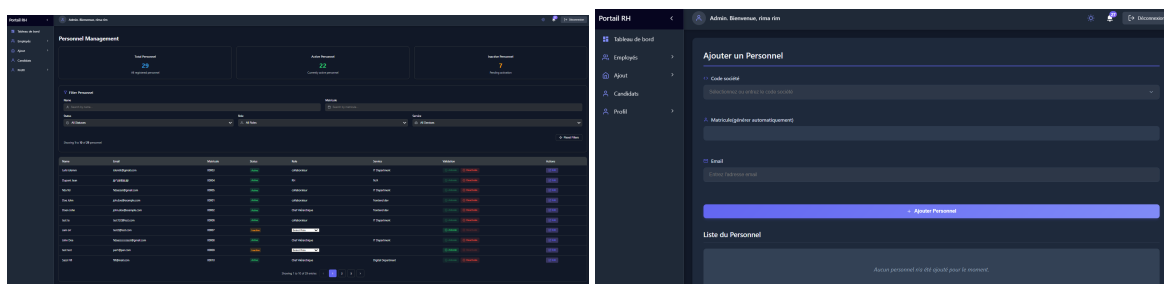


FIGURE 3.21 – Interface Web d'ajout personnel - Thème clair



(a) Interface Web de la liste des personnels -Thème sombre

(b) Interface Web d'ajout personnels -Thème sombre

FIGURE 3.22 – Interfaces de gestion personnel

3.16.3 Personnels - Gestion du Profil

3.16.3.1 Interfaces Web

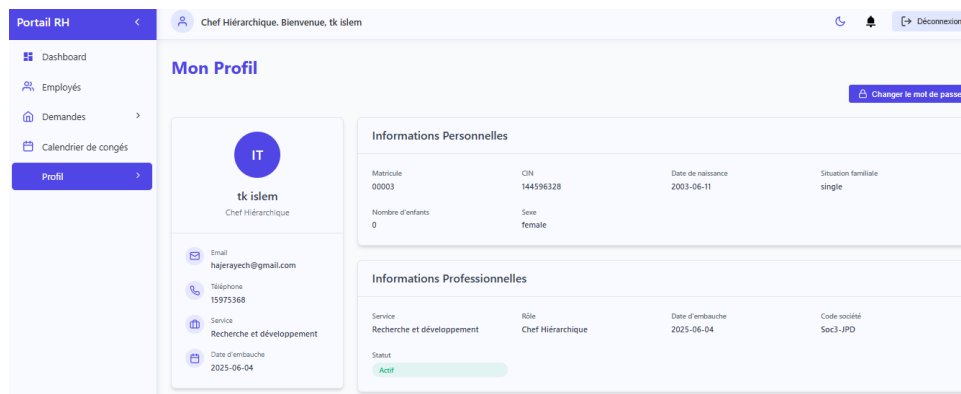
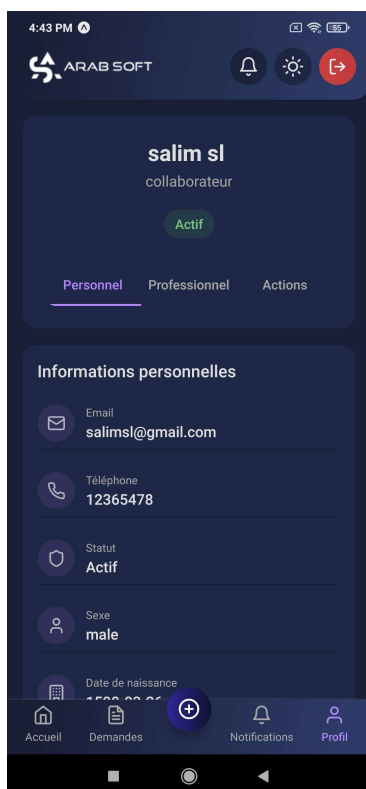
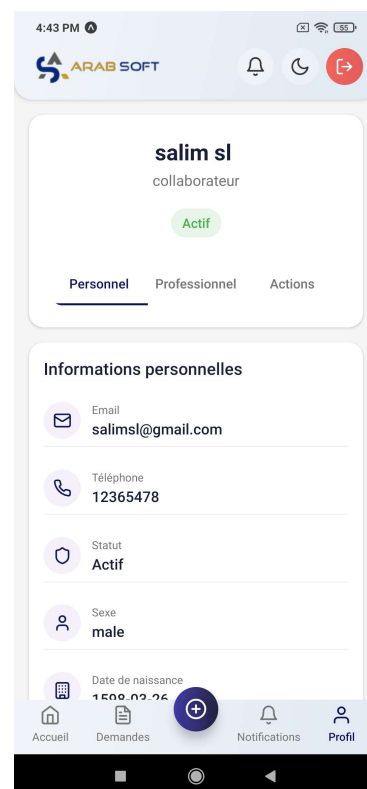


FIGURE 3.23 – Profile (Thème clair)

3.16.3.2 Interfaces Mobile



(a) Profile (Thème sombre)



(b) Profile (Thème clair)

FIGURE 3.24 – Interfaces de profil personnel

3.16.4 Gerer les succursales sociétés

FIGURE 3.25 – Interface Web d’ajout succursale société - Thème clair

Nom	Code	Emplacement	Actions
Societe 1	Soc1-JPH	Montplaisir	Modifier Supprimer
Societe 2	Soc2-JPH	Lac1	Modifier Supprimer
Societe 3	Soc3-JPD	Ariana	Modifier Supprimer

FIGURE 3.26 – Interface Web de la liste des succursales - Thème clair

FIGURE 3.27 – Interface Web modification d’un succursale - Thème clair

3.16.5 Gerer les services

FIGURE 3.28 – Interface Web d’ajout service avec nom - Thème clair

FIGURE 3.29 – Interface Web d’ajout service avec nom et chefs - Thème clair

Service Name	Chief 1 (Weight 1)	Chief 2 (Weight 2)	Chief 3 (Weight 3)	Actions
Développement logiciel	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
Recherche et développement	00005 - test test2	00006 - test test3	00007 - test	Edit Delete
Direction financière	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
Direction financière	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
Direction financière	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
Direction financière	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
IT	00002 - sassi fida	00003 - tk islem	00004 - test	Edit Delete
service achat	00002 - sassi fida	00004 - test test1	00007 - test	Edit Delete
Departement logistique	00005 - test test2	00004 - test test1	00003 - tk	Edit Delete

FIGURE 3.30 – Interface Web de la liste des services - Thème clair

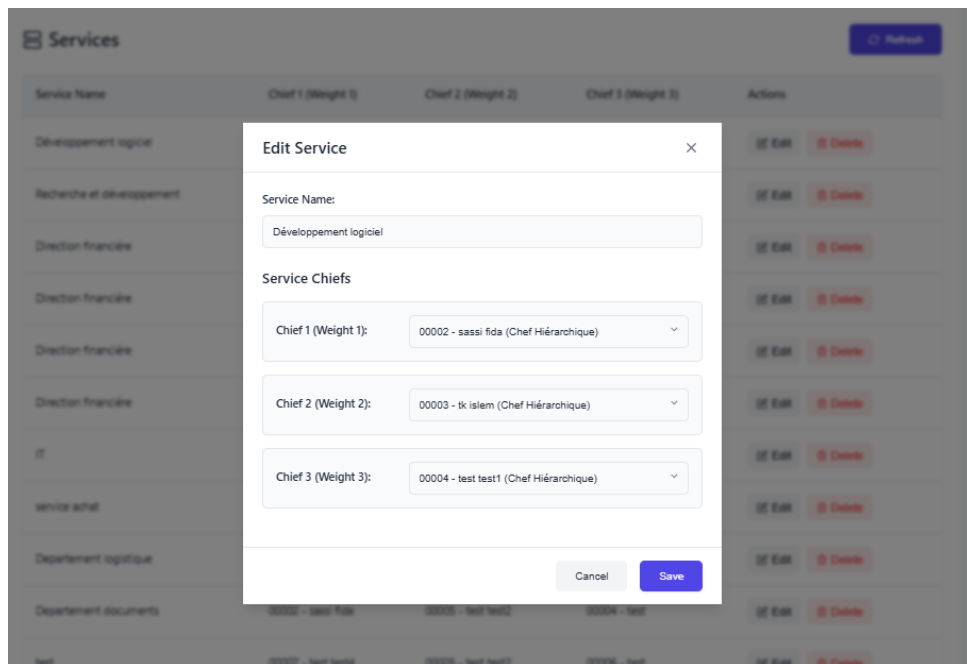
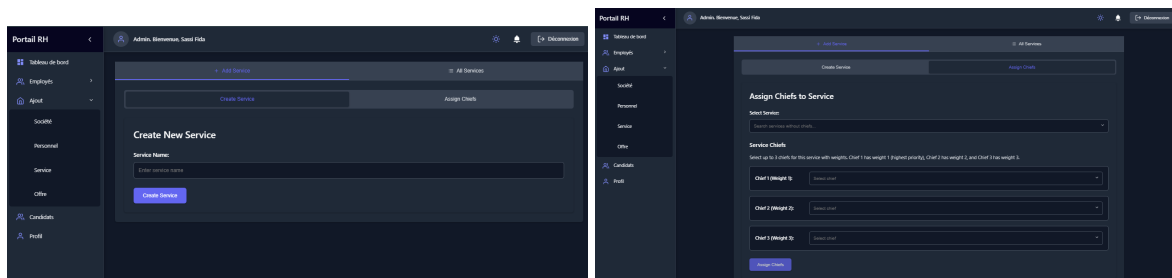
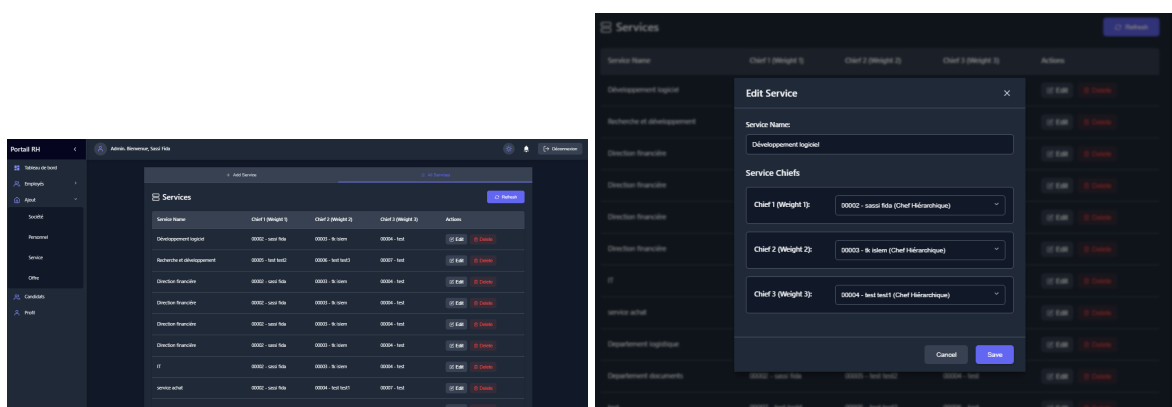


FIGURE 3.31 – Interface Web modification d'un service - Thème clair



(a) Interface Web d'ajout service avec nom -Thème sombre **(b) Interface Web d'ajout service avec nom et chefs -Thème sombre**

FIGURE 3.32 – Interfaces de gestion personnel



(a) Interface Web de la liste des services -Thème sombre **(b) Interface Web modification d'un service -Thème sombre**

FIGURE 3.33 – Interfaces de gestion personnel

Chapitre 4 : Conception et réalisation du sprint 2

4.1 Sprint 2

Ce sprint d'un mois a été consacré à l'extension du projet par l'intégration de fonctionnalités clés pour la gestion des demandes : ajout, modification et traitement des demandes de congé, d'autorisation, de formation, etc., ainsi que la mise en place d'un système complet de notifications internes (en temps réel) et externes (par email).

- Mise en place d'un système de notifications internes en temps réel
- Intégration d'un système de notifications par email
- Assurer la traçabilité des événements sensibles
- Amélioration de l'expérience utilisateur grâce à une interface réactive et informative

— **Système de demandes :**

- Ajout de différentes demandes (congé, autorisation, formation, document, prêt/avance)
- Consultation, modification et suppression des demandes tant que leur statut est *Instantané*
- Consultation et enregistrement des documents dans un espace dédié
- Consultation du calendrier de congés
- Gestion des demandes par les chefs hiérarchiques et l'administrateur RH (approbation, rejet, traitement)

— **Notifications internes (temps réel) :**

- Affichage d'une icône cloche dans la barre de navigation avec compteur dynamique
- Ouverture d'une fenêtre modale listant les notifications récentes
- Marquage automatique des notifications comme *lues* à l'ouverture
- Historique consultable des notifications
- Notifications aux administrateurs lors de l'enregistrement d'un nouveau personnel, du blocage d'un compte ou d'un changement de mot de passe par un personnel
- Notifications aux chefs de service et au RH lors de la soumission d'une nouvelle demande
- Notification au personnel concerné lors de la validation ou du rejet d'une demande

— **Notifications par email :**

- Envoi automatique d'email lors de l'ajout d'un nouveau personnel (transmission du matricule)
- Envoi d'email en cas d'inscription, d'activation, de désactivation, de verrouillage de compte ou d'échec de connexion (après 3 tentatives)
- Notification de réinitialisation de mot de passe

— **Cas d'usage couverts par les notifications :**

- Ajout de personnel
- Connexion et sécurité des comptes
- Gestion des demandes (congé, formation, autorisation, document, prêt/avance)
- Actions d’approbation ou de rejet par les supérieurs

Ces fonctionnalités constituent la fondation sur laquelle s’appuieront les développements ultérieurs.

4.2 Backlog du sprint

Cet tableau présente les user stories prioritaires sélectionnées pour ce sprint.

Id	Titre	User Story	Priorité
D1	Gérer mes demandes de congé	En tant que personnel, je veux pouvoir gérer mes demandes de congé (ajouter, consulter) et aussi (modifier, supprimer) tant qu’elles ne sont pas encore traitées.	Haute
D2	Gérer mes demandes d’autorisation	En tant que personnel, je veux pouvoir gérer mes demandes d’autorisation (ajouter, consulter) et aussi (modifier, supprimer) tant qu’elles ne sont pas encore traitées.	Haute
D3	Gérer mes demandes de formation	En tant que personnel, je veux pouvoir gérer mes demandes de formation (ajouter, consulter) et aussi (modifier, supprimer) tant qu’elles ne sont pas encore traitées.	Haute
D4	Gérer mes demandes de documents	En tant que personnel, je veux pouvoir gérer mes demandes de documents administratifs (ajouter, consulter) et aussi (modifier, supprimer) tant qu’elles ne sont pas encore traitées.	Haute
D5	Gérer mes demandes de prêt/avance	En tant que personnel, je veux pouvoir gérer mes demandes de prêt ou avance (ajouter, consulter) et aussi (modifier, supprimer) tant qu’elles ne sont pas encore traitées.	Haute
D6	Traitement des congés	En tant que chef de service, je veux pouvoir accepter ou rejeter les demandes de congé des membres de mon service.	Haute

Id	Titre	User Story	Priorité
D7	Traitement des autorisations	En tant que chef de service, je veux pouvoir accepter ou rejeter les demandes d'autorisation des membres de mon service.	Haute
D8	Traitement des formations	En tant que chef de service, je veux pouvoir accepter ou rejeter les demandes de formation des membres de mon service.	Haute
D9	Traitement des documents	En tant qu'administrateur RH, je veux pouvoir consulter et traiter ou rejeter les demandes de documents de tous les employés.	Haute
D10	Traitement des prêts/avances	En tant qu'administrateur RH, je veux pouvoir consulter et traiter ou rejeter les demandes de prêts ou avances de tous les employés.	Haute
D11	Traitement des demandes des formations	En tant qu'administrateur RH, je veux pouvoir consulter et traiter l'ensemble des demandes de formation.	Haute
D12	Traitement des demandes des autorisations	En tant qu'administrateur RH, je veux pouvoir consulter et traiter l'ensemble des demandes d'autorisation.	Haute
D13	Traitement des demandes des congés	En tant qu'administrateur RH, je veux pouvoir consulter et traiter l'ensemble des demandes de congé.	Haute
D14	Consulter calendrier des congés (admin/chef)	En tant qu'administrateur RH ou chef hiérarchique, je veux pouvoir consulter les demandes de congé approuvées dans un calendrier.	Moyenne
D15	Consulter calendrier des congés (personnel)	En tant que personnel, je veux pouvoir consulter les demandes de mes congés approuvés dans un calendrier.	Moyenne
D16	Consulter et télécharger les documents	En tant que personnel, je veux pouvoir consulter mes documents envoyés par l'RH lors du traitement de mes demandes de document.	Moyenne

Id	Titre	User Story	Priorité
N1	Notification nouvel enregistrement personnel	En tant qu'administrateur, je souhaite être notifié en temps réel lors de l'enregistrement d'un nouveau personnel pour pouvoir intervenir rapidement.	Haute
N2	Email identifiant après enregistrement	En tant que personnel, je veux recevoir un email avec mon identifiant de connexion après mon enregistrement.	Haute
N3	Icône notifications	En tant qu'utilisateur, je veux voir une icône de notifications avec un compteur de nouveaux messages.	Haute
N4	Consulter notifications modale	En tant qu'utilisateur, je veux pouvoir consulter mes notifications dans une fenêtre modale.	Moyenne
N5	Email activation / désactivation compte	En tant que personnel, je souhaite recevoir un email lors de l'activation, la désactivation ou le verrouillage de mon compte.	Haute
N6	Notification changement mot de passe/blocage	En tant qu'administrateur, je veux être notifié lorsqu'un personnel change son mot de passe ou blocage de compte.	Basse
N7	Notification soumission demande chef	En tant que chef hiérarchique, je veux être notifié lorsqu'un employé soumet une demande (congé, formation, autorisation).	Haute
N8	Notification soumission RH	En tant que RH, je veux être notifié lorsqu'un employé soumet une demande (Prêt, Avance, Document) et demande (congé, formation, autorisation) acceptée par chef.	Haute
N9	Notification résultat demande	En tant que personnel, je veux recevoir une notification lorsque ma demande est acceptée ou rejetée.	Moyenne
N10	Historique notifications	En tant qu'utilisateur, je souhaite accéder à l'historique de mes notifications.	Basse

TABLE 4.1 – Backlog Sprint 2

4.3 Diagrammes du cas "Sprint 2"

4.3.1 Diagramme de cas d'utilisation

La Figure 4.1 illustre les interactions principales couvertes lors de ce deuxième sprint.

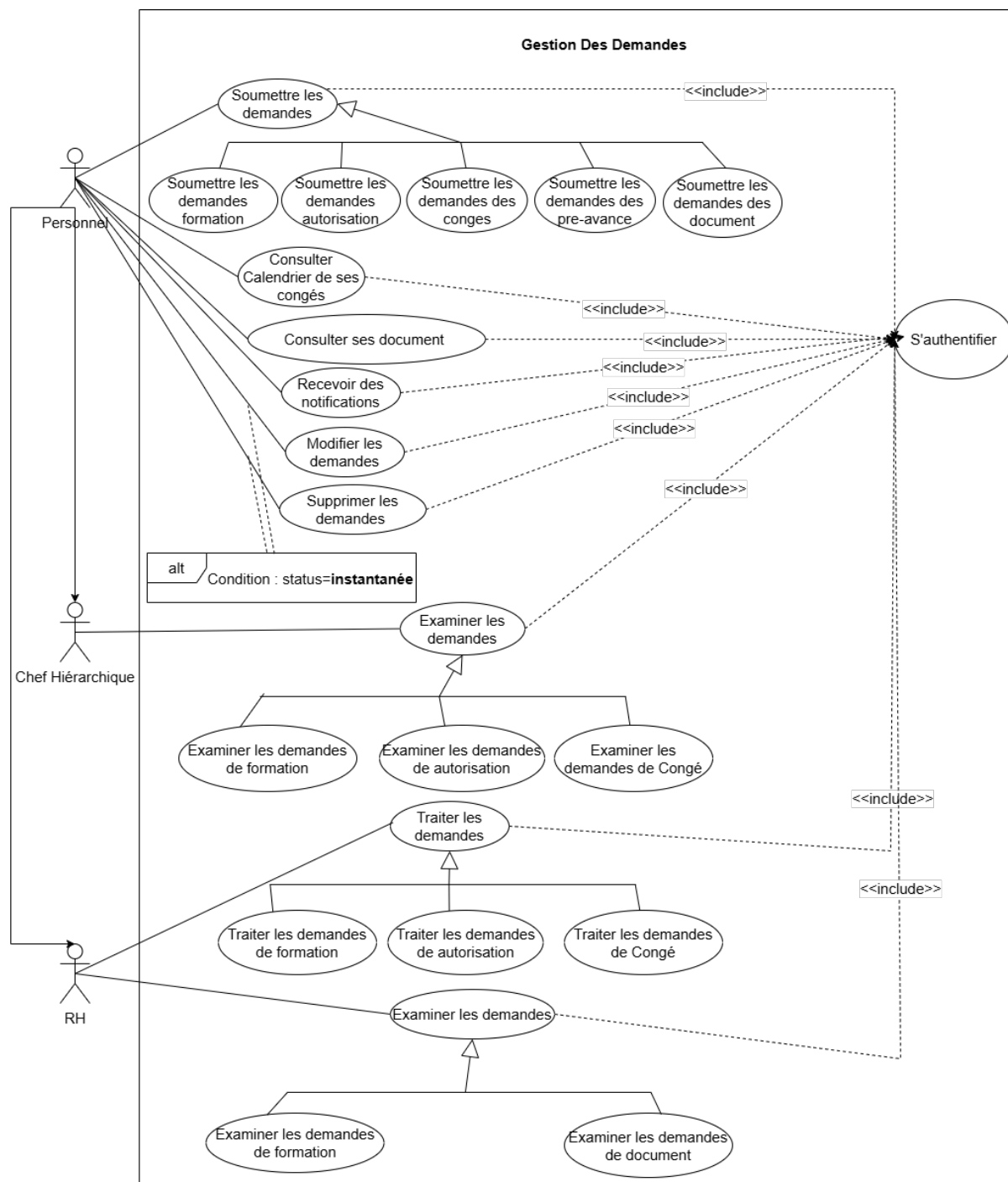


FIGURE 4.1 – Diagramme de cas d'utilisation du cas "Sprint 2"

4.3.2 Diagramme de classes

La Figure 4.2 présente la structure des classes principales mises en œuvre pour ce sprint.

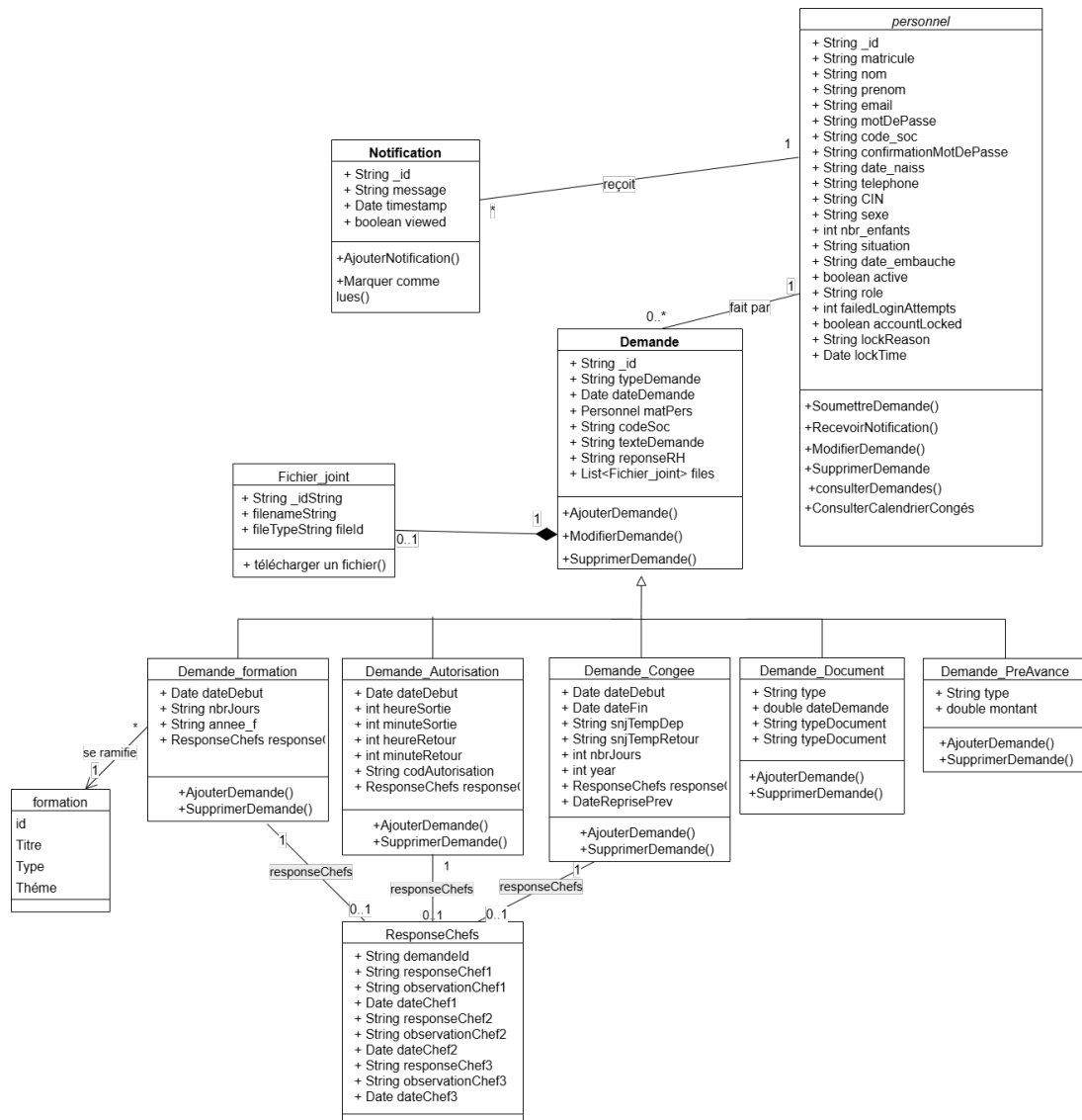


FIGURE 4.2 – Diagramme de classes du cas "Sprint 2"

4.4 La gestion des demandes

4.5 Description textuelle du cas : Ajouter une demande

Élément	Description
Acteurs concernés	Personnel
Préconditions	Personnel authentifié
Post conditions	Demande ajoutée
Scénario nominal	- Le personnel accède à l'interface de création de demande. - Il saisit les informations nécessaires. - Il valide l'envoi de la demande. - La demande est enregistrée avec le statut <i>"instantané"</i>.
Scénarios d'erreur	2.a) Champs obligatoires manquants ou invalides.

TABLE 4.2 – Description textuelle du cas – Ajouter une demande

4.5.1 Diagramme du cas "Ajouter une demande"

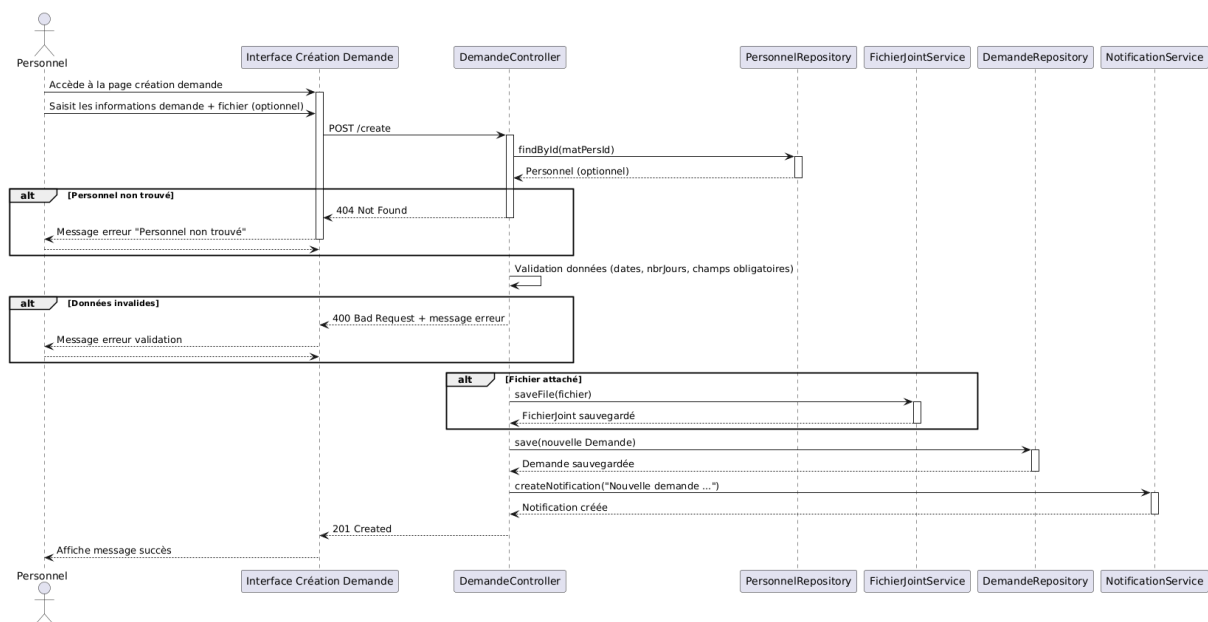


FIGURE 4.3 – Diagramme de séquence du cas "Ajouter une demande"

4.6 Description textuelle du cas : Modifier une demande

Élément	Description
Acteurs concernés	Personnel
Préconditions	- Personnel authentifié - La demande doit être au statut <i>"instantané"</i>
Post conditions	Demande modifiée
Scénario nominal	- Le personnel accède à ses demandes. - Il sélectionne une demande à modifier. - Il modifie les champs souhaités. - Il sauvegarde les modifications.
Scénarios d'erreur	Demande déjà traitée (modification interdite).

TABLE 4.3 – Description textuelle du cas – Modifier une demande

4.6.1 Diagramme du cas "Modifier une demande"

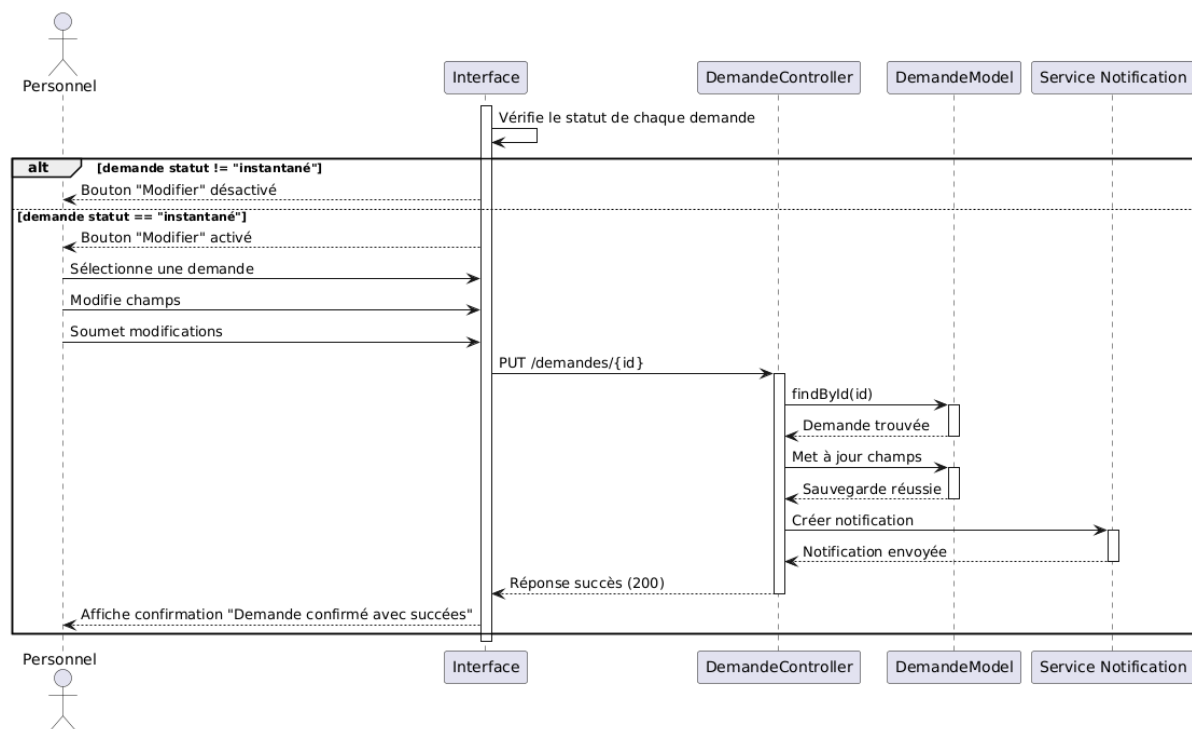


FIGURE 4.4 – Diagramme de séquence du cas "Modifier une demande"

4.7 Description textuelle du cas : Supprimer une demande

Élément	Description
Acteurs concernés	• Personnel
Préconditions	• Personnel authentifié • La demande doit être au statut "<i>instantané</i>"
Post conditions	• Demande supprimée
Scénario nominal	1. Le personnel consulte ses demandes. 2. Il sélectionne une demande à supprimer. 3. Il confirme la suppression. 4. La demande est supprimée du système.
Scénarios d'erreur	1.a) Tentative de suppression d'une demande déjà traitée.

TABLE 4.4 – Description textuelle du cas – Supprimer une demande

4.7.1 Diagramme du cas "Supprimer une demande"

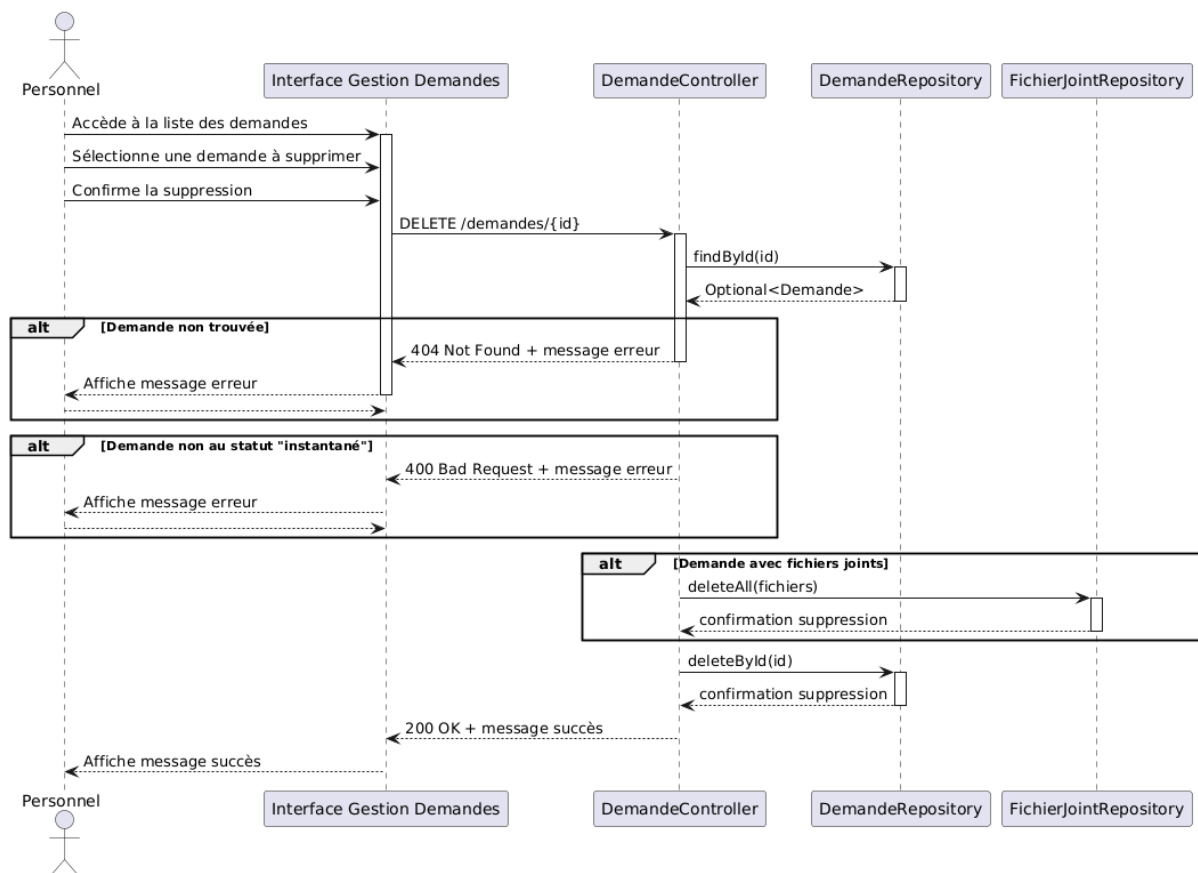


FIGURE 4.5 – Diagramme de séquence du cas "Supprimer une demande"

4.8 Description textuelle du cas : Examiner (rejeter/ approuver) une demande

Élément	Description
Acteurs concernés	• Ressources Humaines (RH) • Chef du service
Préconditions	• RH authentifié • Chef du service authentifié • Demande préalablement approuvée par le chef de service

Élément	Description
Post conditions	• Demande approuvée ou rejetée avec notification envoyée au personnel concerné
Scénario nominal	1. Le RH ou le chef du service consulte les demandes (suivant le type de demande). 2. Il sélectionne une demande à examiner. 3. Il décide d'approuver ou de rejeter la demande. 4. La décision est enregistrée. 5. Le personnel est notifié de la décision.

TABLE 4.5 – Scénario nominal du cas d'utilisation : Examiner une demande

4.8.1 Diagramme du cas "Examiner (rejeter/ approuver) une demande"

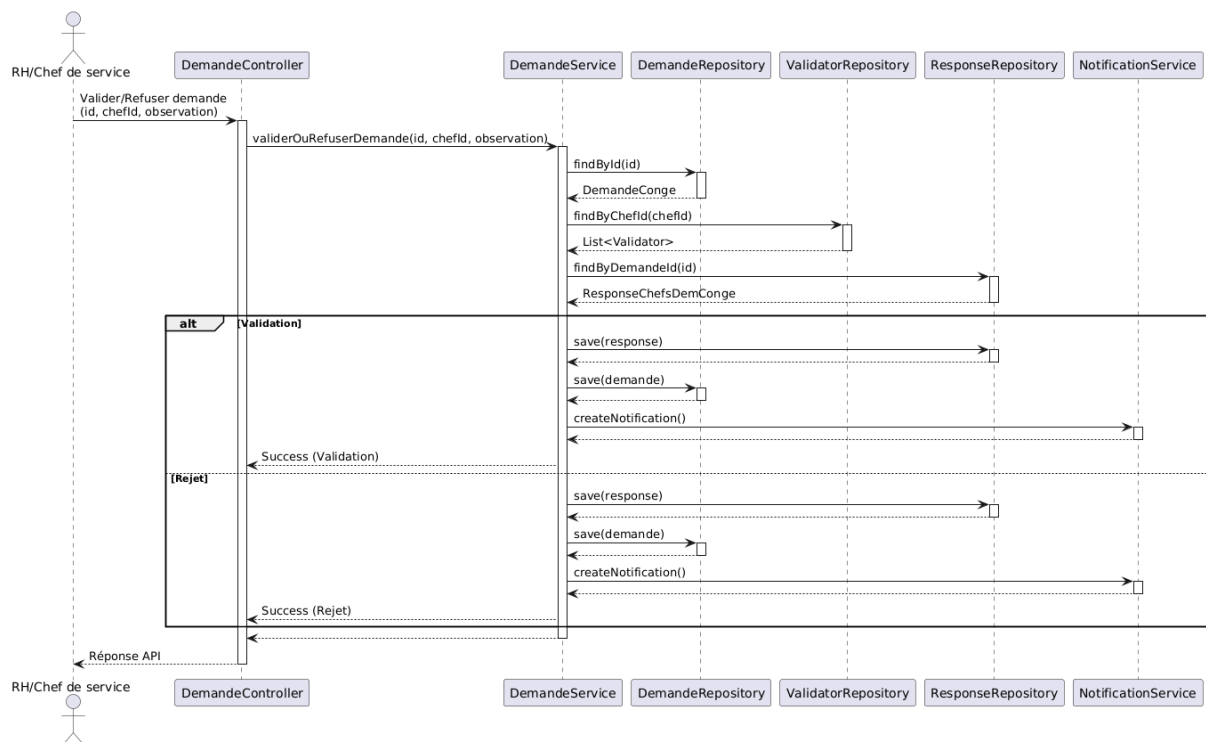


FIGURE 4.6 – Diagramme de séquence du cas "Examiner (rejeter/ approuver) une demande"

4.9 Description textuelle du cas : Traiter une demande

Élément	Description
Acteurs concernés	• RH
Préconditions	• RH authentifié • Demande existante et en attente de traitement
Post conditions	• Demande marquée comme traitée
Scénario nominal	1. Le RH consulte les demandes . 2. Il sélectionne une demande à traiter. 3. Il effectue le traitement nécessaire. 4. Le statut de la demande passe à "traitée".

TABLE 4.6 – Description textuelle du cas – Traiter une demande

4.9.1 Diagramme du cas "Traiter une demande"

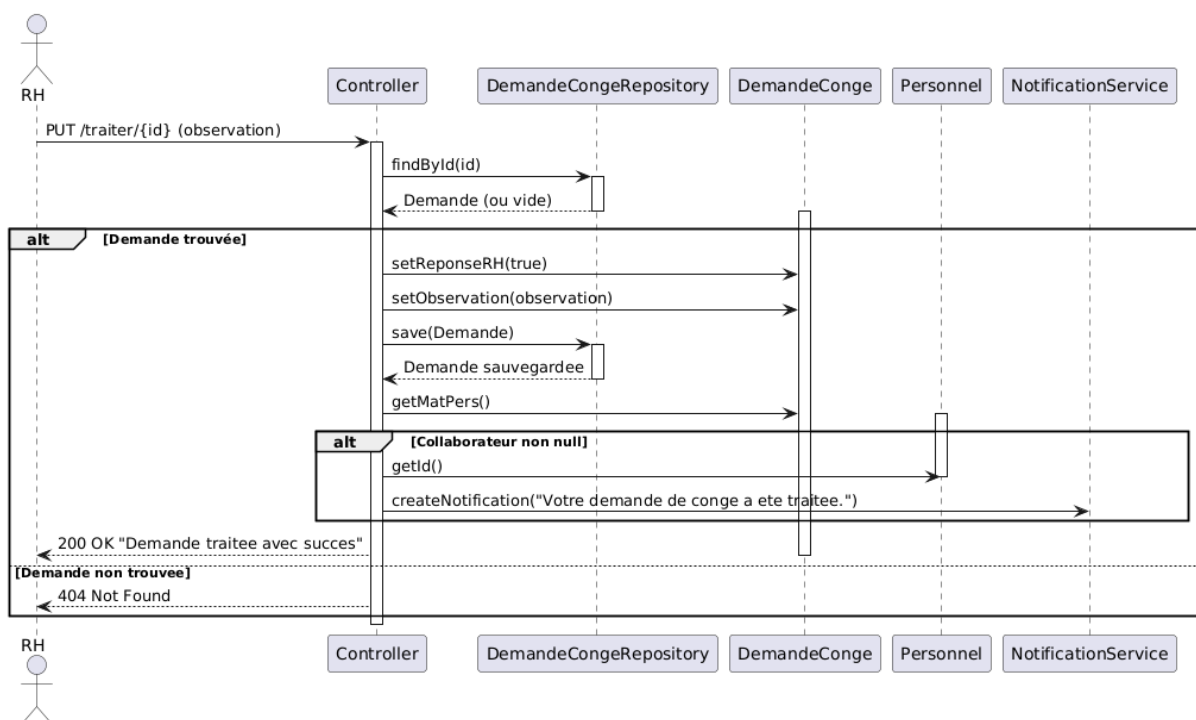


FIGURE 4.7 – Diagramme de séquence du cas "Traiter une demande"

4.10 La gestion des notifications

4.10.1 Technologies utilisées et justification des choix :

Pour assurer la réactivité du système, WebSocket a été utilisé pour la gestion des notifications internes en temps réel. Cette technologie a été choisie pour sa capacité à maintenir une connexion persistante entre le client et le serveur, permettant ainsi une communication bidirectionnelle instantanée, idéale pour ce type de fonctionnalités. Quant aux notifications par email, elles ont été implémentées via un système de messagerie serveur intégré, assurant la fiabilité des envois d'emails pour les cas sensibles nécessitant une trace formelle.

Ce choix technologique permet de combiner efficacité, scalabilité et meilleure expérience utilisateur, tout en répondant aux exigences de sécurité et de traçabilité du système.

4.10.1.1 WebSocket : Comment ça fonctionne ?

1. **Ouverture** : Le navigateur ouvre une connexion WebSocket avec le serveur via une requête spéciale appelée handshake. Cette requête est initiée par le protocole HTTP, puis basculée en WebSocket si le serveur l'accepte.
2. **Échange** : Une fois la connexion établie, les deux parties (client et serveur) peuvent échanger des messages de manière bidirectionnelle et asynchrone, sans avoir besoin de rouvrir une connexion à chaque message.
3. **Fermeture** : Une fois la communication terminée, la connexion peut être fermée proprement par l'une ou l'autre des parties.

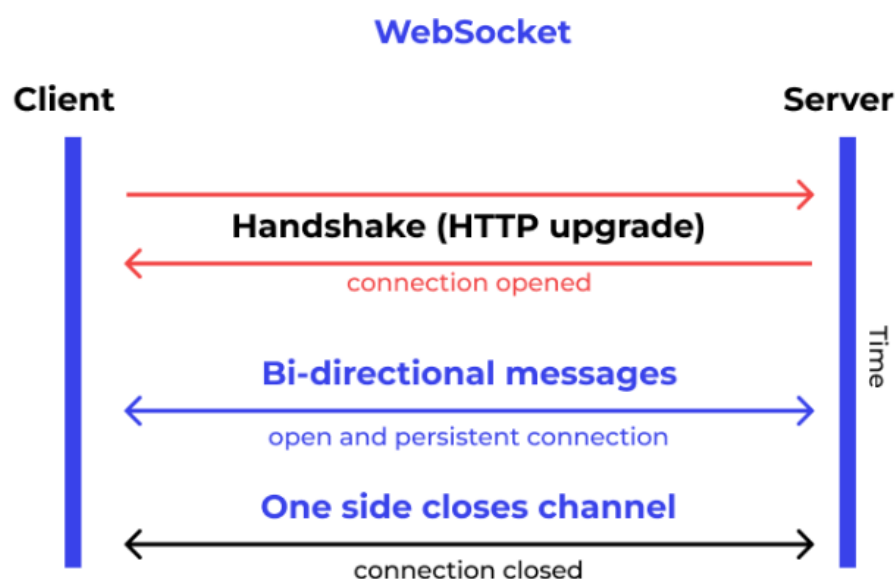


FIGURE 4.8 – Connection WebSocket

4.11 Description Textuelle des Cas d'Utilisation

Cette section présente la description des cas d'utilisation principaux réalisés lors du sprint, orientée autour du système de notifications temps réel et notifications par e-mail.

4.11.1 Description textuelle du cas "Recevoir une notification en temps réel"

Ce scénario explique comment le système envoie des notifications en temps réel aux utilisateurs concernés, dès qu'un événement important se produit.

Élément	Description
Titre	Gestion des notifications en temps réel
Acteurs Concernés	— Personnel (incluant Chef hiérarchique / RH) — Administrateur
Préconditions	— L'utilisateur est connecté au système. — Il dispose d'un rôle autorisé à recevoir les notifications. — Un événement système a été déclenché (ex. : demande soumise, nouvel utilisateur ajouté. . .).
Déclencheur	— Un employé soumet une demande (congé, formation, autorisation. . .). — Un nouvel employé est ajouté dans le système. — Tout autre événement métier générateur de notification.
Scénario nominal	1. Le système détecte un événement spécifique. 2. Le système identifie automatiquement les destinataires. 3. Le système déclenche deux types de notifications. 4. L'utilisateur concerné reçoit la notification instantanément. 5. Le système enregistre l'état de la notification (non lue/lue).
Postconditions	— Le destinataire est informé immédiatement de l'action à traiter. — L'interface affiche la notification et permet d'agir en conséquence. — Les notifications sont tracées et archivées dans l'historique utilisateur.

TABLE 4.7 – Description du cas d'utilisation : Gestion des notifications en temps réel

4.11.1.1 Diagramme de Séquence du cas "Recevoir une notification en temps réel"

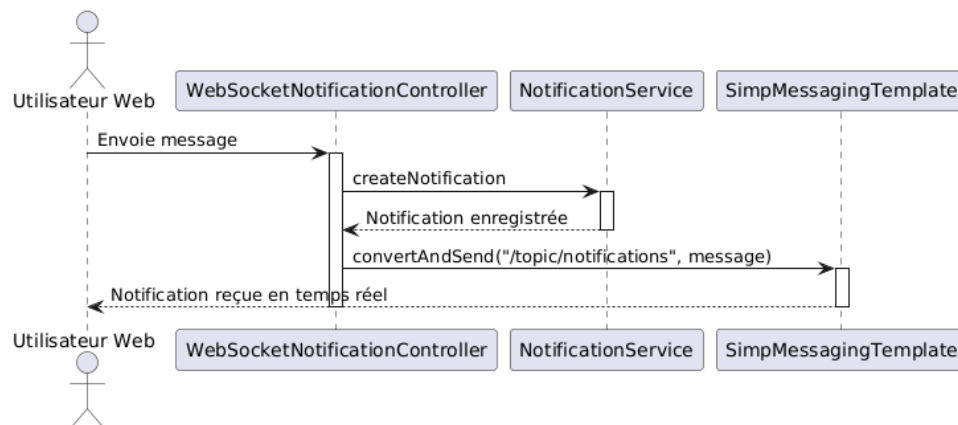


FIGURE 4.9 – Diagramme de séquence "Recevoir une notification en temps réel"

4.11.2 Description textuelle du cas "Consulter les notifications internes"

Ce scénario illustre le processus complet permettant à un utilisateur connecté de consulter ses notifications récentes, puis d'accéder à l'historique complet des alertes reçues.

Élément	Description
Titre	Gestion de la consultation des notifications
Acteurs Concernés	— Personnel (incluant Chef hiérarchique / RH) — Administrateur
Préconditions	— L'utilisateur est connecté au système. — Il a reçu au moins une notification. — Il dispose d'un historique de notifications.
Déclencheur	— L'utilisateur clique sur l'icône de notification dans la barre de navigation. — L'utilisateur clique sur "Voir toutes les notifications".

Élément	Description (suite)
Scénario nominal	1. L'utilisateur clique sur l'icône de notification. 2. Les notifications ouvertes sont marquées comme lues. 3. Le système affiche un aperçu des notifications récentes. 4. L'utilisateur clique sur "Voir toutes les notifications". 5. Le système affiche une page dédiée avec la liste complète des notifications (ordonnées par date).
Postconditions	— L'utilisateur a pu consulter ses notifications récentes et anciennes. — Les notifications ouvertes sont marquées comme lues.

TABLE 4.8 – Description du cas d'utilisation : Consultation des notifications

4.11.2.1 Diagramme de Séquence du cas "Consulter les notifications internes"

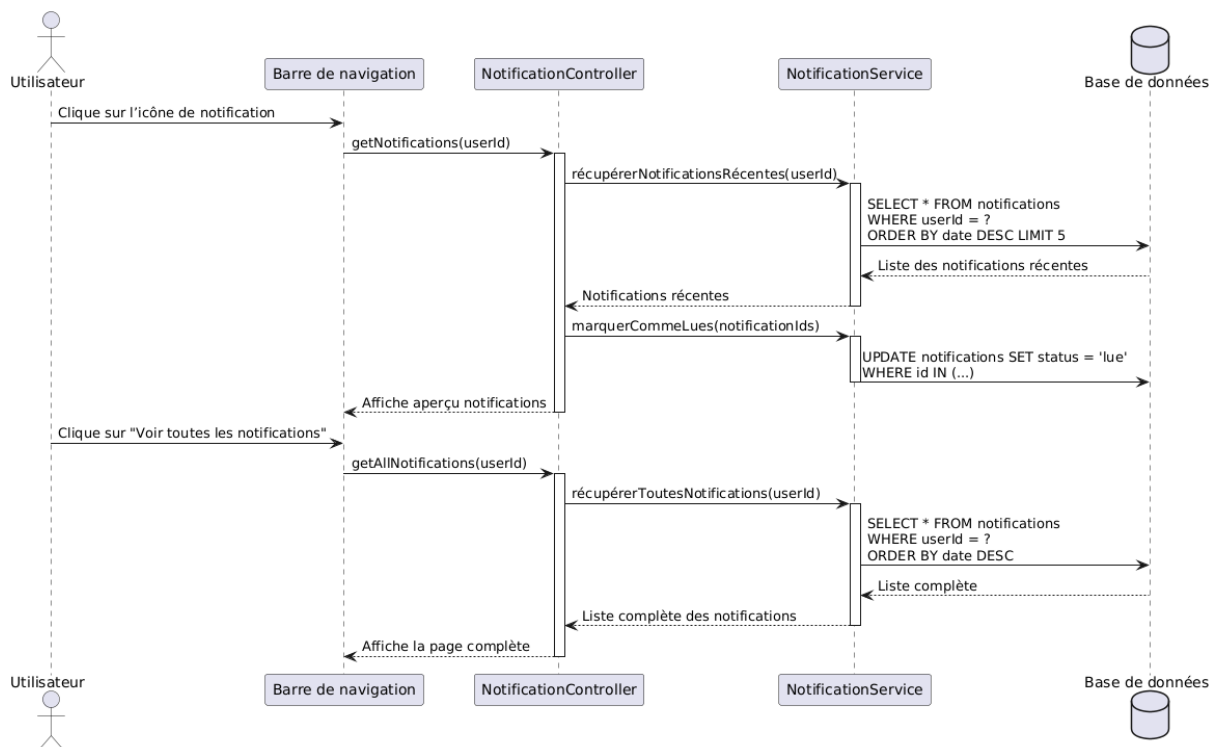


FIGURE 4.10 – Diagramme de séquence "Consulter les notifications internes"

4.11.3 Flux de l'envoi d'un e-mail

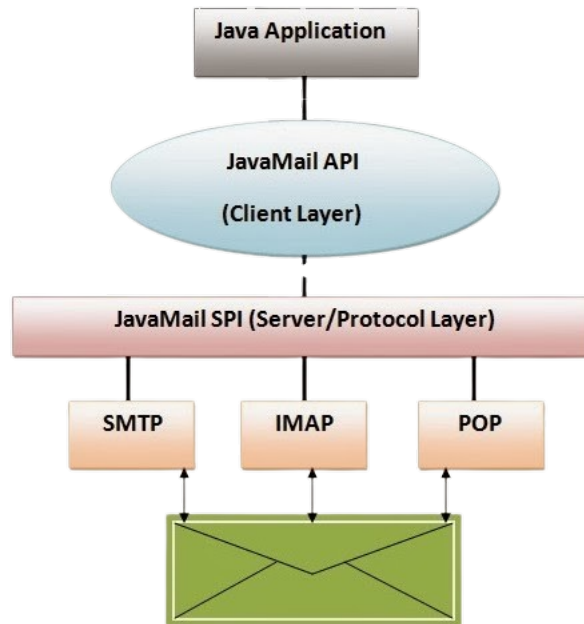


FIGURE 4.11 – Flux de l'envoi d'un e-mail

4.11.3.1 Détails techniques

- Serveur SMTP : smtp.gmail.com, port 587 (avec STARTTLS)
- API JavaMailSender pour l'envoi des e-mails
- Messages MIME pour le contenu HTML
- Annotation `@Async` de Spring pour les opérations non bloquantes

4.11.4 Description textuelle du cas "Envoi d'e-mails"

Élément	Description
Acteurs Concernés	— Utilisateur backend (par exemple, l'administrateur qui ajoute un nouveau personnel) — Système backend (Spring Boot) — Service Gmail SMTP
Préconditions	— Les identifiants Gmail sont correctement configurés dans l'application — Le serveur SMTP (smtp.gmail.com) est accessible — Le module EmailService est opérationnel
Scénario Nominal	1. Le backend reçoit un déclencheur (par exemple, l'ajout d'un nouveau personnel) 2. Le EmailService prépare l'adresse e-mail du destinataire, l'objet et le contenu HTML 3. Le EmailService utilise le bean JavaMailSender pour créer un message MIME 4. L'authentification SMTP est réalisée avec les identifiants Gmail 5. L'e-mail est transmis via une connexion sécurisée (STARTTLS) au serveur SMTP Gmail 6. Le service consigne la réussite de l'envoi ou l'échec dans les logs
Postconditions	— Un e-mail HTML de confirmation est envoyé au destinataire — En cas d'échec, une trace d'erreur est enregistrée dans les logs pour analyse
Scénarios Alternatifs	4.a SMTP injoignable : Le serveur SMTP Gmail est indisponible, l'e-mail échoue et l'erreur est loggée. 5.a Timeout réseau : La connexion au serveur SMTP expire, l'e-mail n'est pas envoyé et l'incident est loggé.

TABLE 4.9 – Description du cas d'utilisation "Envoi d'e-mail via EmailService"

4.11.4.1 Diagramme de séquence du cas "Envoi d'e-mails"

Le diagramme de séquence suivant illustre l'interaction entre les composants du backend pour l'envoi d'un e-mail de confirmation.

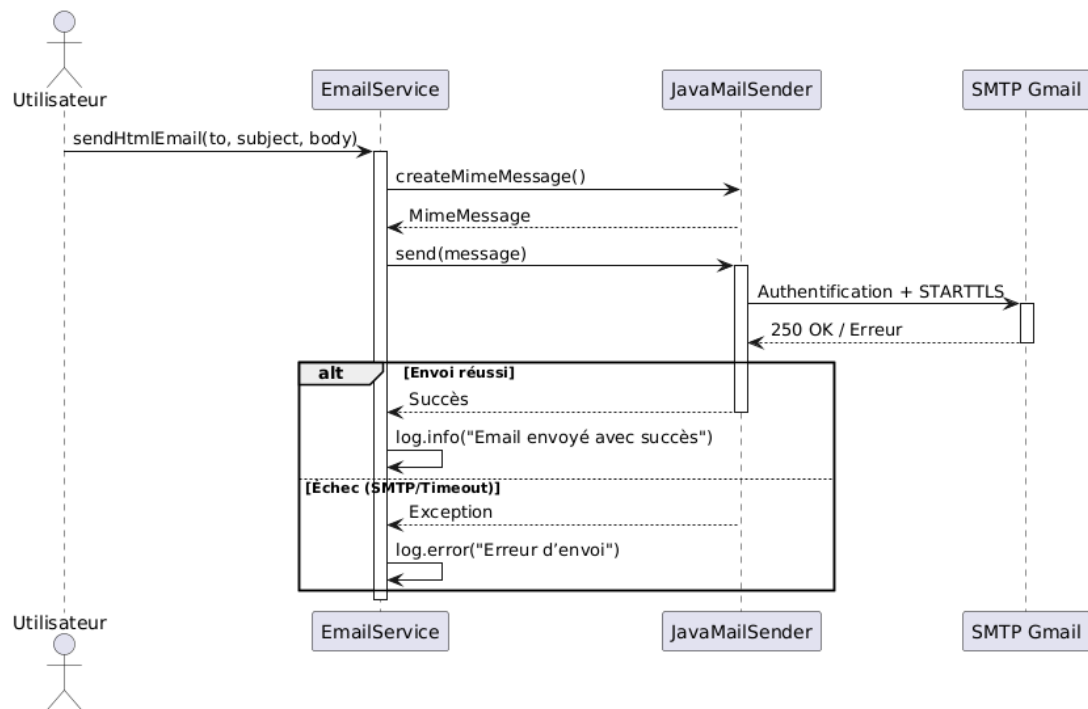
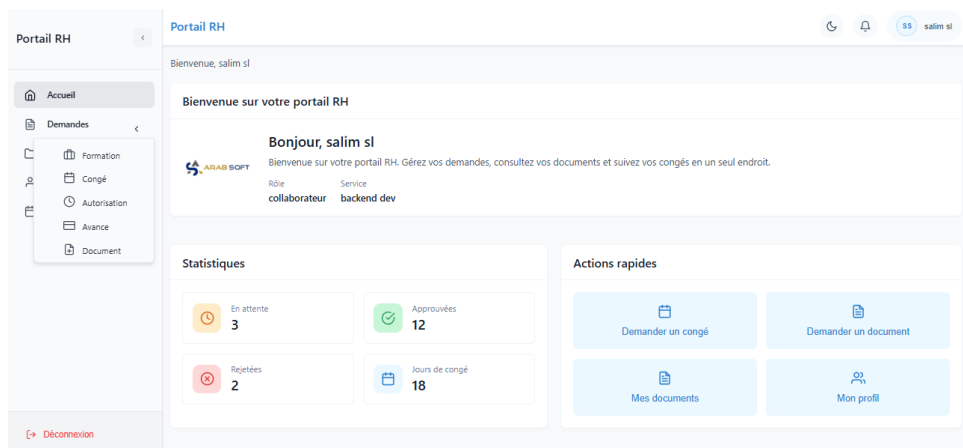


FIGURE 4.12 – Diagramme de séquence : Processus d'envoi d'e-mail

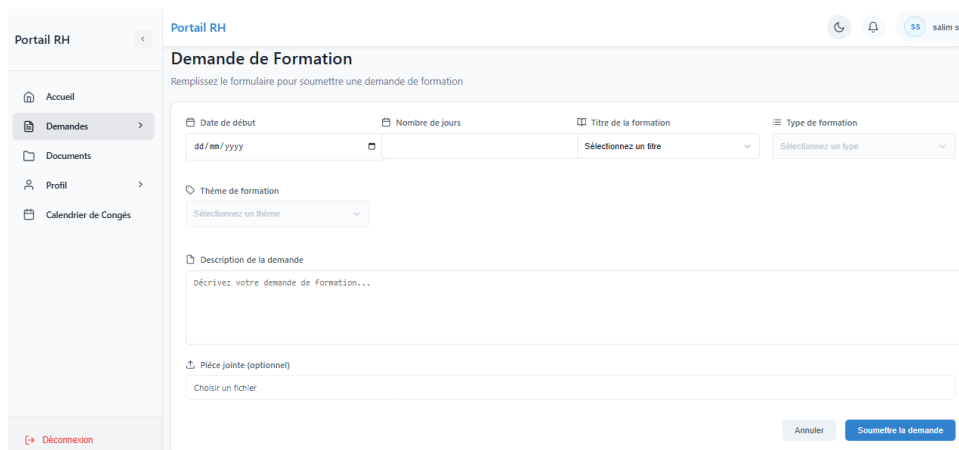
4.12 Résultats obtenus

4.12.1 Demandes

4.12.1.1 Interfaces Web



(a) Accueil - Thème clair



(b) Demande Formation - Thème clair

Portail RH

Demande de Congé

Répondez le formulaire pour soumettre une demande de congé

Date de début: 01/04/2025 Date de fin: 01/04/2025 Nombre de jours: 1 Motif de la demande: Sédentarité en hospitalisation

Description de la demande: Décrivez votre demande de congé...

Choisir un fichier

Annuler Soumettre la demande

(a) Demande Congé - Thème clair

Portail RH

Demande d'Autorisation

Répondez le formulaire pour soumettre une demande d'autorisation

Date: 01/04/2025 Heure de Sortie: 14:00 Heure de Retour: 18:00

Texte de la Demande: Décrivez votre demande d'autorisation...

Choisir un fichier

Annuler Soumettre la demande

(b) Demande Autorisation - Thème clair

Portail RH

Calendrier des Congés Acceptés

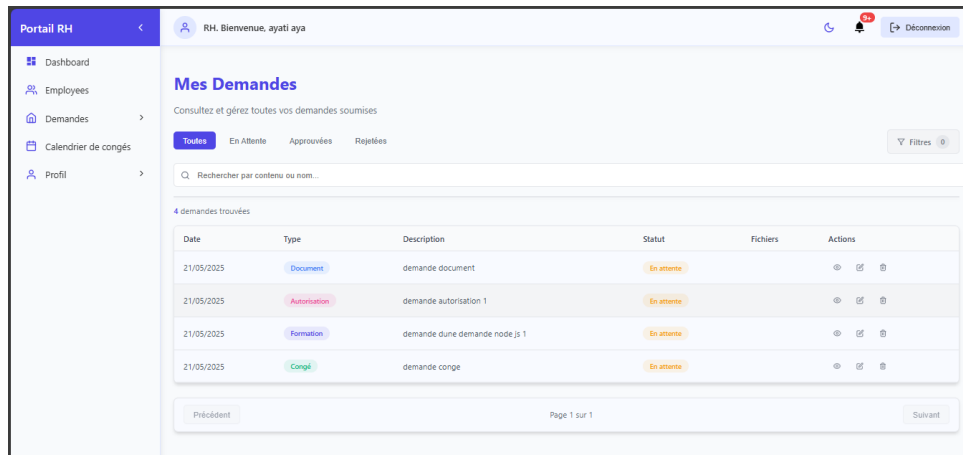
Avril 2025

Mes Congés Acceptés

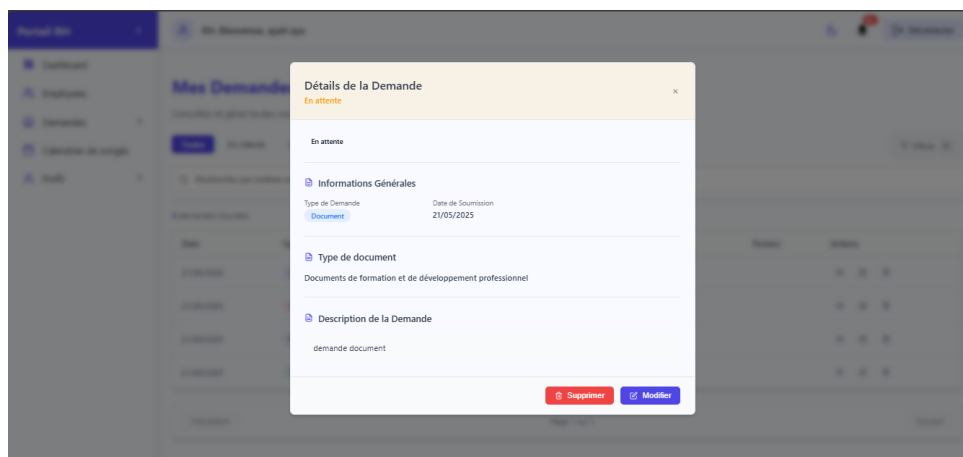
- salim sl ✓ Accepté
29/04/2025 - 06/05/2025
10 jours
- salim sl ✓ Accepté
27/04/2025 - 28/04/2025
2 jours
- salim sl ✓ Accepté
24/04/2025 - 30/04/2025
7 jours
- salim sl ✓ Accepté
15/04/2025 - 17/04/2025

(c) Calendrier Congé - Thème clair

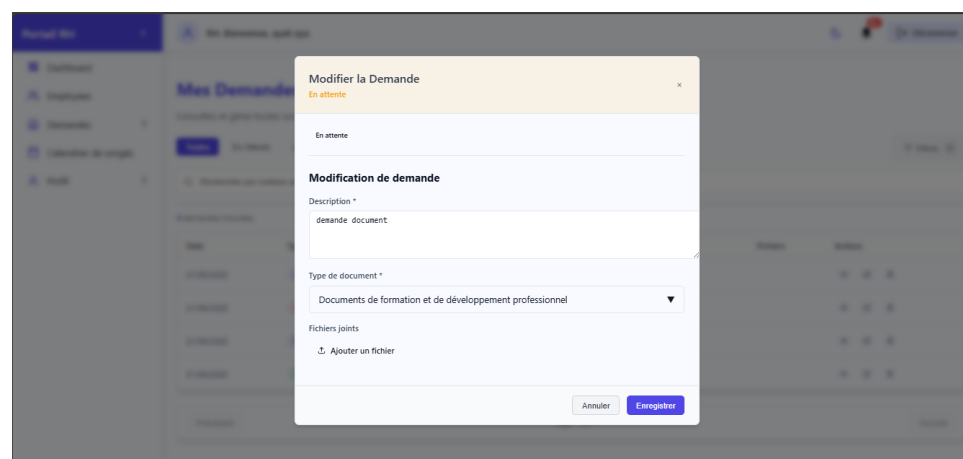
FIGURE 4.14 – Interfaces Web des demandes



(a) Historique Demandes - Thème clair



(b) Détails d'une demande - Thème clair



(c) Modification de demande - Thème clair

FIGURE 4.15 – Interfaces web pour la gestion des demandes

4.12.1.2 Interfaces Mobile

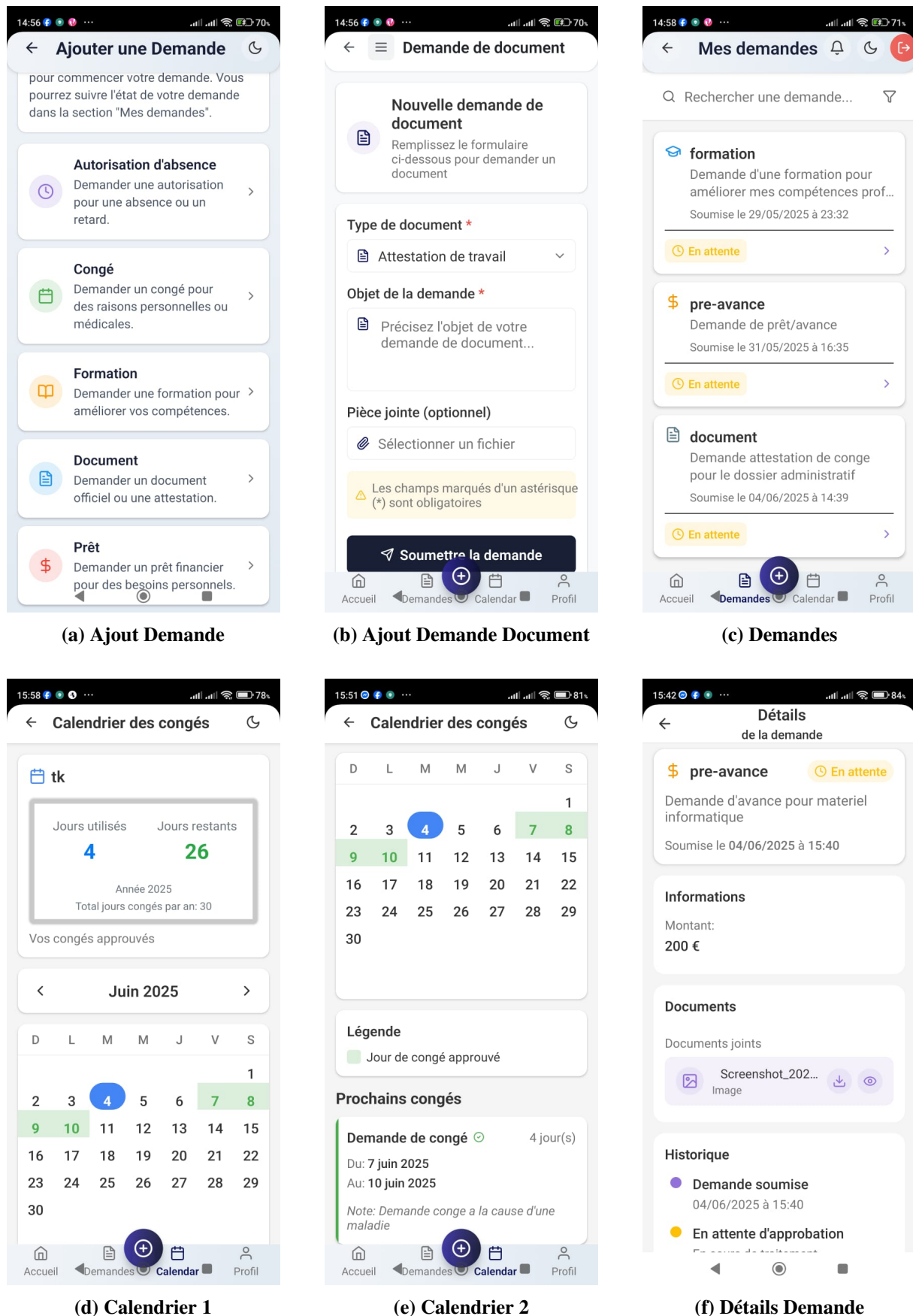


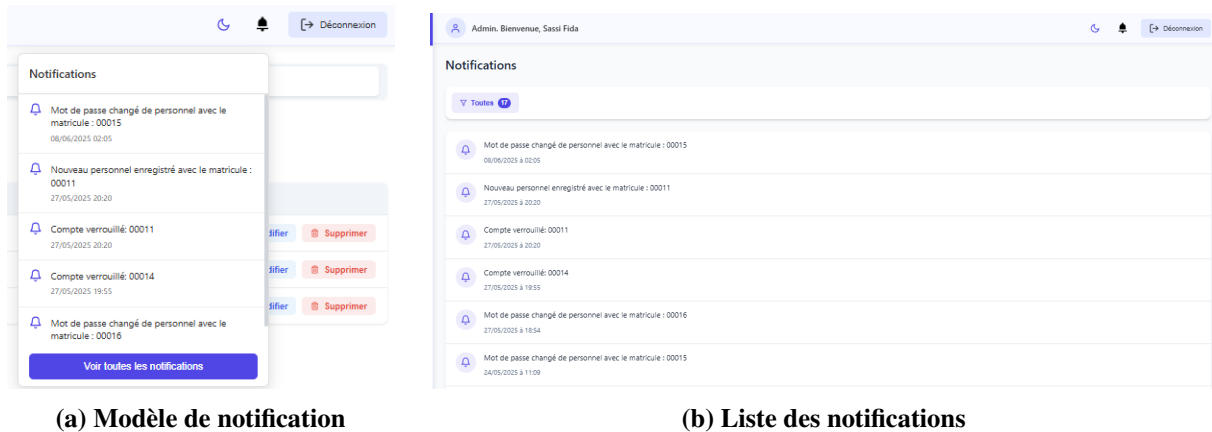
FIGURE 4.16 – Interfaces mobiles pour la gestion des demandes

4.12.2 Notifications

4.12.2.1 Interfaces Web



FIGURE 4.17 – Icône de notification dans l'interface utilisateur

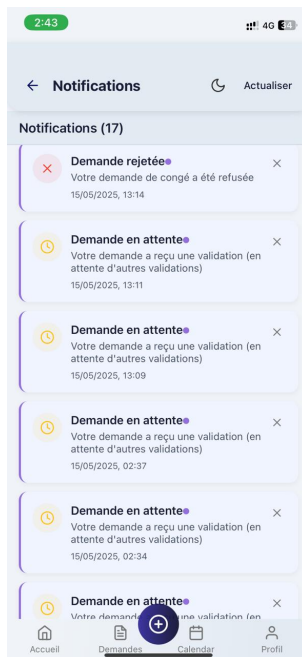


(a) Modèle de notification

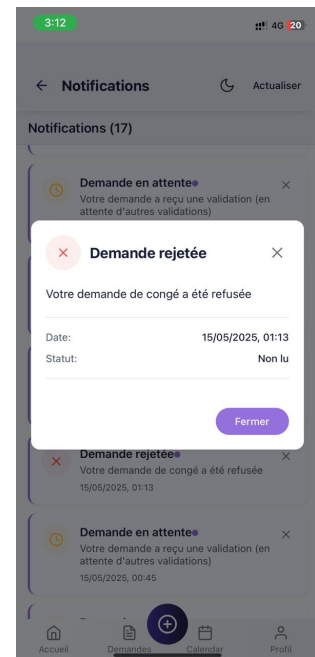
(b) Liste des notifications

FIGURE 4.18 – Interface des notifications côté Web

4.12.2.2 Interfaces Mobile



(a) Liste notifications - Clair



(b) Detail de notification mobile

FIGURE 4.19 – Notifications côté mobile

Chapitre 5 : Intégration du Module d'analyse de CV par AI

5.1 Sprint 3 :

Ce sprint d'une durée de 3 semaines m'a permis de développer les composants essentiels suivants

- Extraire le texte brut des CV.
- Identifier les compétences mentionnées dans le CV correspondant à une liste prédéfinie pour une offre.
- Utiliser un modèle de langage (LLM) via une API externe pour réaliser cette analyse et générer des informations structurées.
- Intégrer ces informations (compétences trouvées/manquantes, score) dans les informations d'un candidat .

Ce module vise à accélérer le processus de présélection et à améliorer la pertinence des candidatures examinées par les recruteurs.

5.2 Technologies Utilisées et Justification des Choix

Plusieurs technologies ont été sélectionnées et combinées pour réaliser cette fonctionnalité :

- **Python (Script d'Analyse) :**
 - *Utilisation :* Langage choisi pour le script externe (`ai_script.py`) réalisant l'extraction de texte et l'appel à l'API d'IA.
 - *Avantages :* Écosystème de référence pour le traitement du langage naturel (NLP) et l'intelligence artificielle, vaste choix de bibliothèques spécialisées, facilité de prototypage et d'intégration avec des services d'IA externes.
- **Langchain (Framework Python) :**
 - *Utilisation :* Utilisé dans le script Python pour simplifier et standardiser l'interaction avec l'API Groq et le modèle Llama 3.
 - *Avantages :* Abstrait la complexité des appels API directs, facilite la gestion des prompts, la structuration des entrées/sorties pour le LLM, et l'intégration de différents composants (modèles, parseurs, etc.). Permet de se concentrer sur la logique métier de l'analyse plutôt que sur les détails de l'API.
- **Groq API & Llama 3 70B (Service IA) :**
 - *Utilisation :* Le script Python interroge le modèle Llama 3 70B hébergé sur la plateforme Groq via une requête API sécurisée à l'aide d'une clé d'authentification (API Key).
 - *Pourquoi utiliser l'API Key Groq ?*

- Une clé API est indispensable pour authentifier les requêtes envoyées au service Groq. Elle permet de sécuriser l'accès, suivre l'utilisation et garantir que seuls les utilisateurs autorisés exploitent la puissance du modèle.
- Elle est générée depuis le tableau de bord utilisateur Groq, puis intégrée dans le script Python pour permettre les appels au modèle Llama 3 70B.
- *Avantages :*
 - **Performance (Groq) :** La plateforme Groq est spécialisée dans l'inférence ultra-rapide des grands modèles de langage grâce à sa puce dédiée LPU (Language Processing Unit). Elle permet une génération de texte jusqu'à **249 tokens par seconde**, nettement supérieure à d'autres fournisseurs (voir figure 5.1).
 - **Qualité (Llama 3 70B) :** Le modèle open-source Llama 3, développé par Meta, se distingue par sa capacité à comprendre et analyser le langage naturel, ce qui le rend idéal pour des tâches telles que l'analyse de CV et la génération de données structurées (ex : en format JSON).

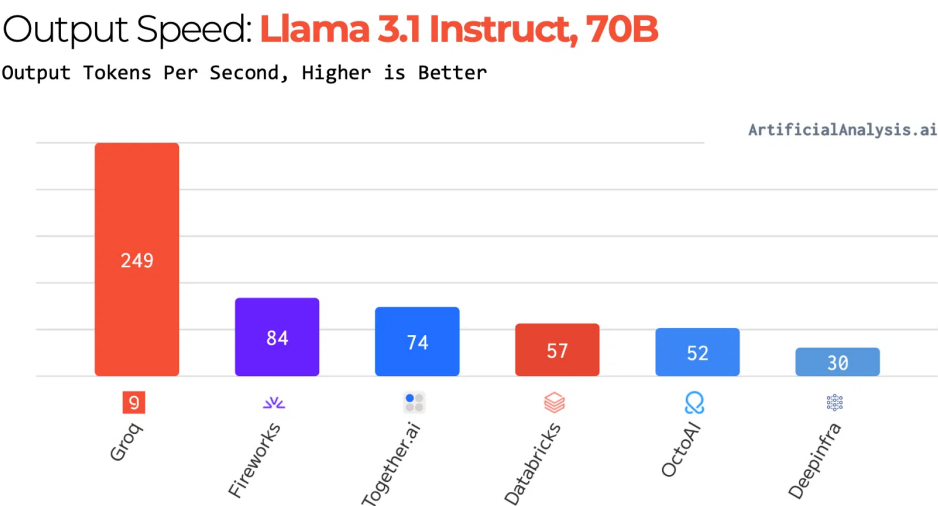


FIGURE 5.1 – Comparaison des vitesses de génération (tokens/seconde) pour Llama 3.1 70B selon différentes plateformes. Groq est nettement le plus rapide.

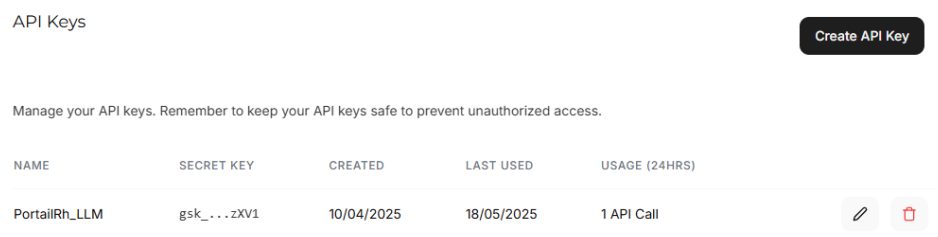


FIGURE 5.2 – Exemple de création d'une clé API sur la plateforme Groq.

— **PyMuPDF (Extraction PDF en Python) :**

- *Utilisation* : Bibliothèque Python pour lire et extraire le contenu textuel des fichiers PDF.
- *Avantages* : Très performante, précise, gère une grande variété de formats PDF complexes, et est généralement plus rapide et moins gourmande en ressources que d'autres alternatives comme `pdfminer.six`

5.3 Backlog du Sprint

La réalisation de ce module d'analyse IA a été planifiée au travers du backlog de sprint suivant :

ID	User Story	Priorité
PB1	En tant qu'administrateur, je veux m'authentifier afin d'accéder à l'espace de gestion des candidatures.	Haute
PB2	En tant qu'administrateur, je veux pouvoir ajouter manuellement une candidature.	Haute
PB3	En tant que système, je veux analyser les CV à l'aide d'une IA afin de faciliter l'évaluation des compétences.	Critique
PB4	En tant que candidat, je veux soumettre mon CV sans authentification afin de postuler rapidement à une offre.	Moyenne
PB5	En tant qu'IA, je veux extraire et structurer les compétences depuis un CV afin de produire un résumé exploitable.	Haute
PB6	En tant que système, je veux envoyer un email automatique après l'analyse du CV afin d'informer le candidat du status.	Basse

TABLE 5.1 – Backlog du sprint 3

5.4 Diagrammes du Sprint 3

5.4.1 Diagramme de cas d'utilisation

La Figure 5.3 illustre les interactions principales couvertes lors de ce sprint.

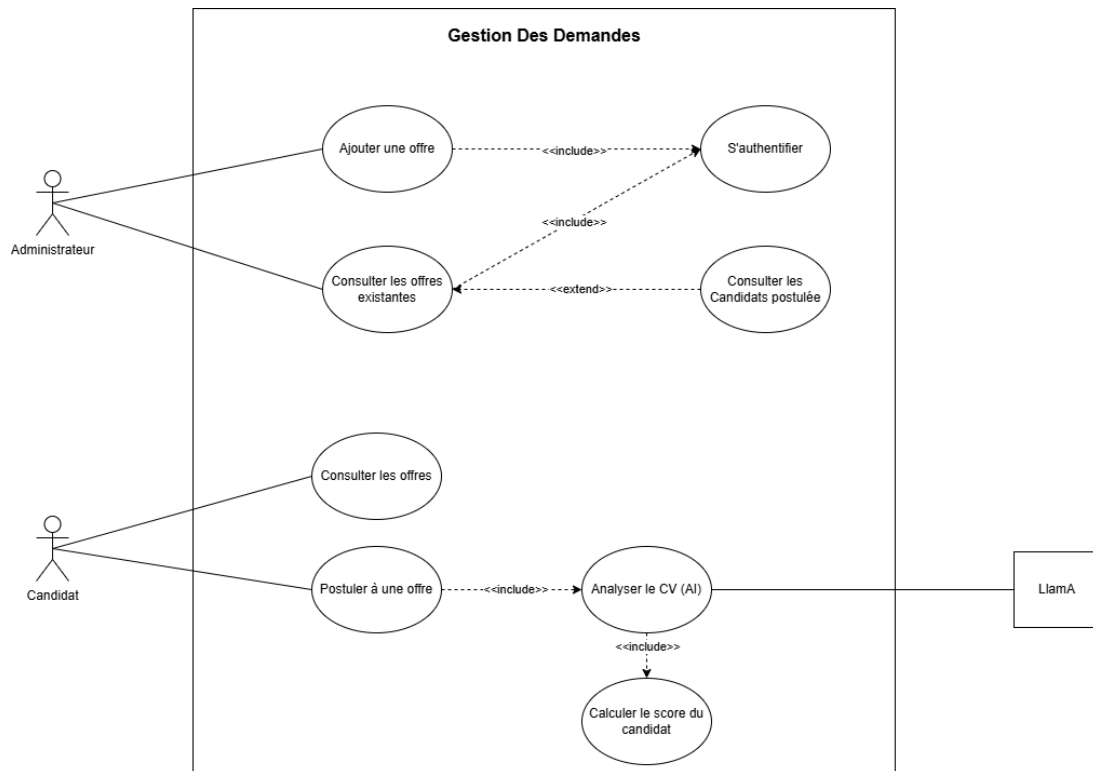


FIGURE 5.3 – Diagramme de cas d'utilisation du Sprint 3

5.5 Diagramme de classes global du Sprint 3

La Figure 5.5 présente la structure des classes définies pour le sprint 3, mettant en avant les relations entre les entités principales du système

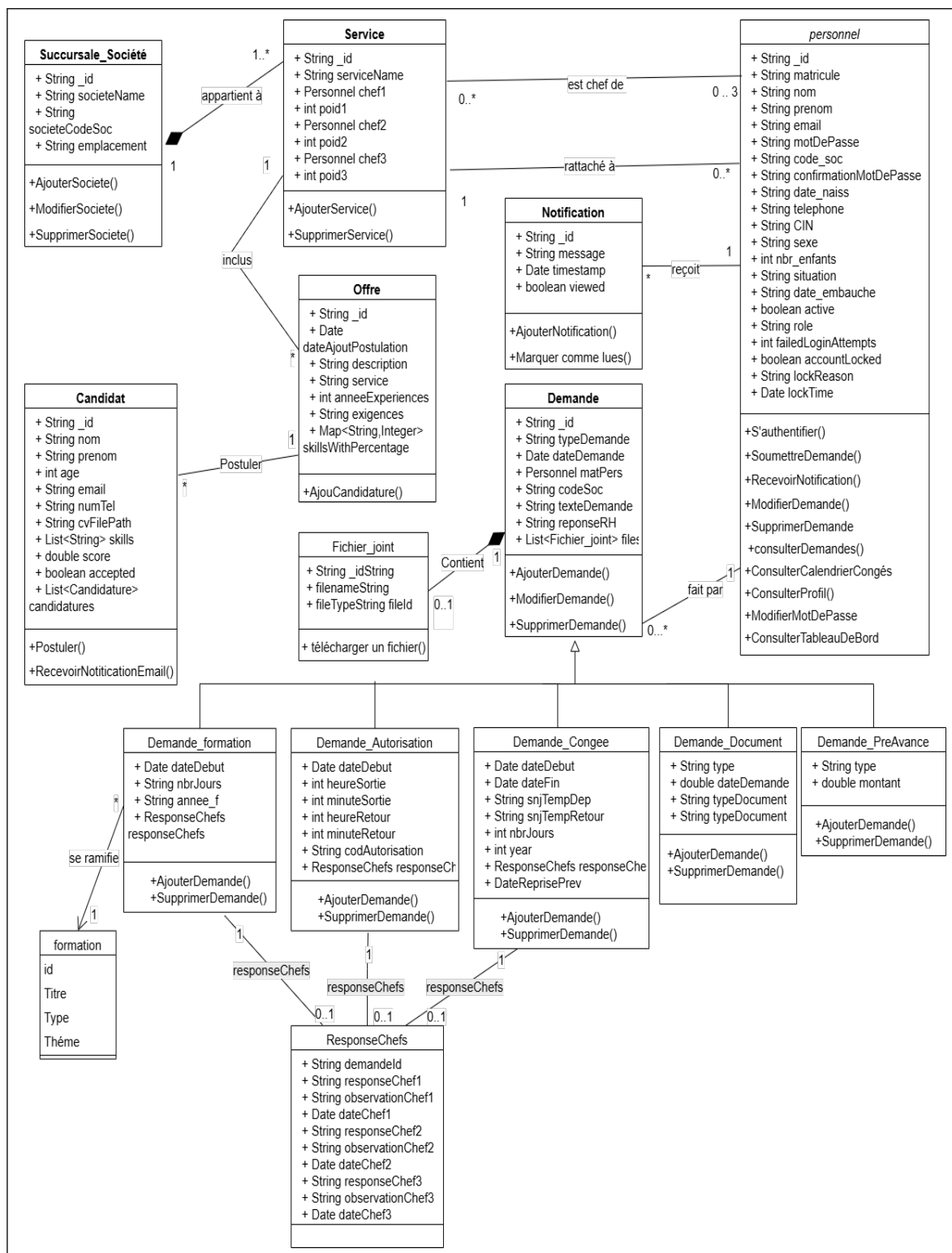


FIGURE 5.4 – Diagramme de classe global du Sprint 3

5.6 Description Textuelle des Cas d'Utilisation

Cette section présente la description des cas d'utilisation principaux réalisés lors du sprint, suivie des diagrammes de séquence correspondants. L'objectif est d'illustrer les interactions entre les acteurs (utilisateurs ou systèmes) et les composants applicatifs du système lors de l'exécution de fonctionnalités clés.

5.6.1 Description textuelle du cas " Ajout d'une offre par l'Admin"

Ce scénario illustre le processus complet permettant à un administrateur RH authentifié d'ajouter manuellement une offre dans le système.

Élément	Description
Acteurs Concernés	— Admin (authentifié) — Système
Scénario Nominal	1. Authentification admin avec JWT valide 2. Affichage du formulaire d'ajout 3. Soumission des détails (nom, poste, etc.) 4. Validation et sauvegarde en base 5. Affichage de la confirmation
Scénarios Alternatifs	1. 1.a Token invalide : Redirection vers la page de login 2. 4.a Validation échouée : Affichage d'un message d'erreur spécifique
Postconditions	— Une nouvelle candidature est enregistrée

TABLE 5.2 – Contexte – Ajout d'une offre par l'Admin

5.6.1.1 Diagramme de Séquence du cas "Ajout d'une offre par l'Admin"

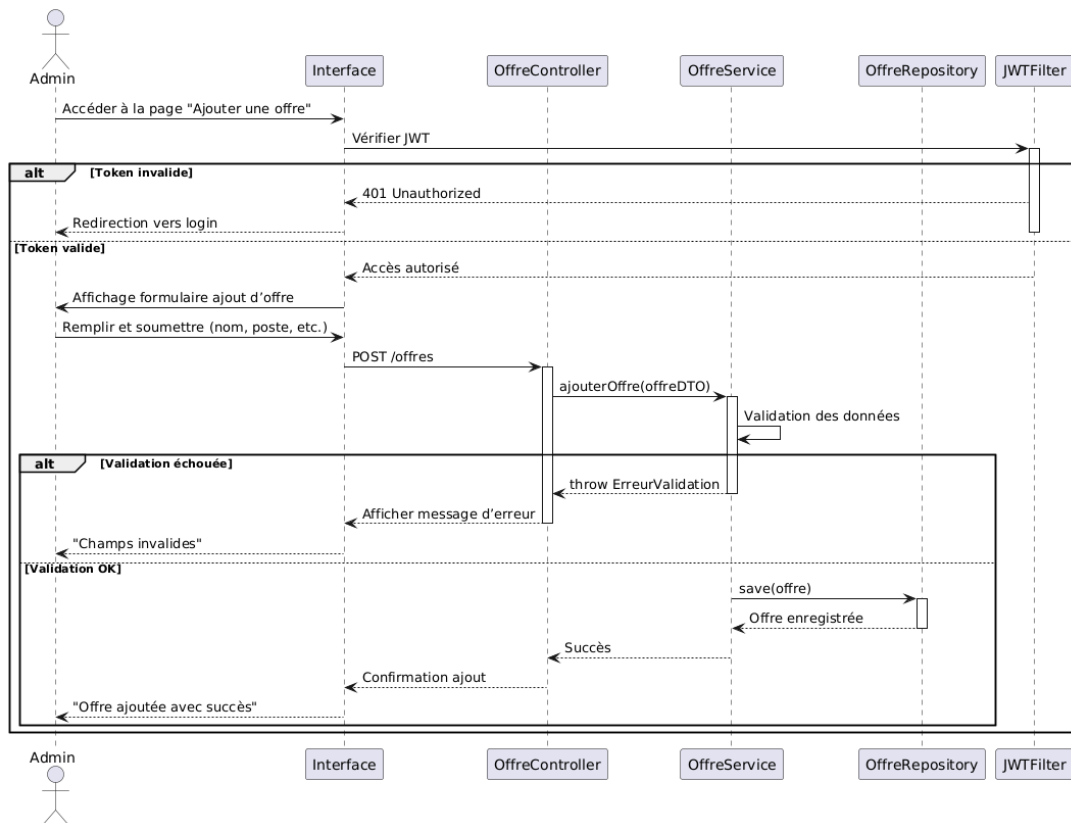


FIGURE 5.5 – Diagramme de séquence du sprint 3 – Cas "Ajouter une offre"

5.6.2 Description textuelle du cas " Postuler a une offre"

Ce scénario décrit le parcours d'un candidat non authentifié souhaitant postuler à une offre. Il peut consulter les offres disponibles et soumettre directement son CV. Ce processus est simplifié afin de maximiser le taux de candidatures reçues.

Élément	Description
Acteurs Concernés	— Candidat — Système — Service IA (analyse de CV)
Préconditions	— Le candidat accède à la plateforme sans authentification — Des offres d'emploi sont disponibles dans le système
Postconditions	— La candidature est sauvegardée avec succès — Le fichier CV est stocké — Les compétences extraites sont associées à la fiche candidat — Le candidat reçoit un email de confirmation
Scénario Nominal	1. Le candidat consulte les offres disponibles 2. Il sélectionne une offre à laquelle il souhaite postuler 3. Il remplit le formulaire avec ses informations de base (nom, prénom, email, etc.) 4. Il téléverse son CV 5. Le système sauvegarde le fichier et les données en base 6. Le système appelle le service IA pour extraire les informations du CV 7. Les compétences extraites sont enregistrées et liées à la candidature 8. Le système envoie un email de confirmation au candidat
Erreurs de Saisie	— 3.a Champs requis non remplis (nom, email...) : message d'erreur s'affiche — 3.b Adresse email invalide : indication immédiate sous le champ concerné

TABLE 5.3 – Déroulement – Cas Postuler a une offre

5.6.2.1 Diagramme de Séquence du cas "Postuler a une offre"

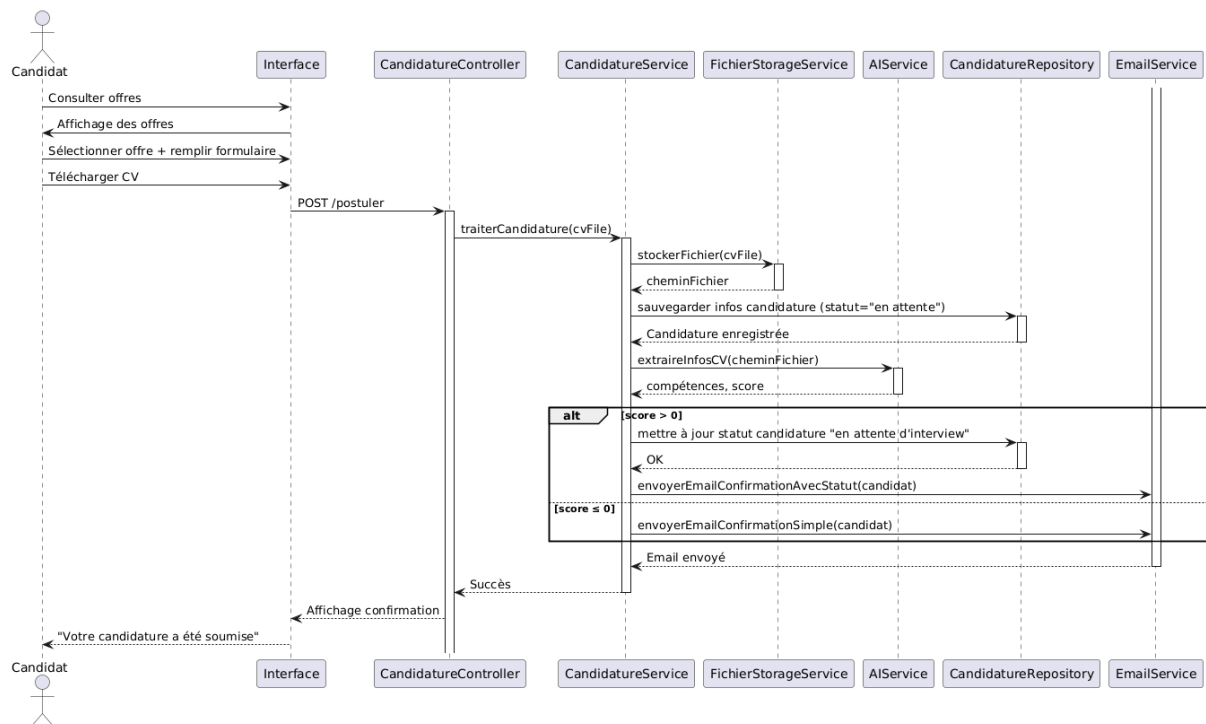


FIGURE 5.6 – Diagramme de séquence du sprint 3 – Cas "Postuler a une offre"

5.6.3 Description textuelle du cas "Traitement du CV par l'IA"

Une fois qu'un CV est soumis, le système déclenche automatiquement un traitement basé sur l'intelligence artificielle. Ce processus permet d'extraire les compétences techniques et d'évaluer sa pertinence par rapport au poste ciblé. Un email de confirmation est envoyée au candidat.

Élément	Description
Acteurs Concernés	— Candidat (non authentifié) — Système — Service IA (analyse du contenu du CV) — Service de stockage de fichiers
Préconditions	— Un CV valide est soumis via le formulaire de candidature — Le service IA est disponible et opérationnel
Scénario Nominal	1. Le candidat téléverse son CV 2. Le système enregistre le fichier sur le serveur 3. Le système transmet le CV au service IA 4. Le service IA extrait les compétences 5. Les informations extraites sont sauvegardées dans la base de données 6. Un email est envoyé au candidat
Postconditions	— Le fichier CV est stocké de manière sécurisée — Les métadonnées (compétences, résumé IA) sont associées au candidat — Un email est envoyé avec les résultats de l'analyse
Scénarios Alternatifs	1. Service IA non disponible : Le système notifie le candidat de l'indisponibilité et enregistre la candidature sans analyse
Erreurs de Traitement	1.a. Fichier non valide (format ou taille) : rejet du fichier avec message d'erreur

TABLE 5.4 – Description – Cas "Traitement du CV par l'IA"

5.6.3.1 Diagramme de séquence du cas "Traitement du CV par l'IA"

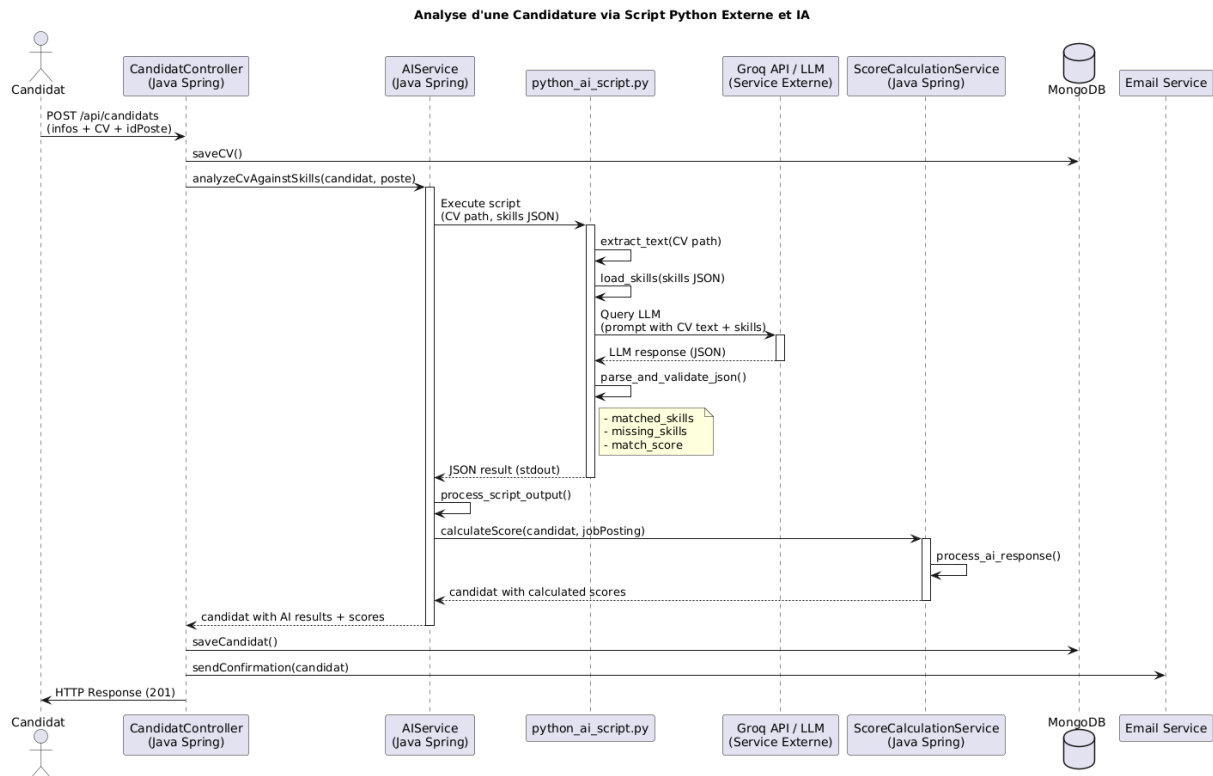


FIGURE 5.7 – Diagramme de séquence du sprint 3 – Cas "Traitement du CV par l'IA"

5.7 Prompt d'analyse pour l'intelligence artificielle

Dans le cadre de l'évaluation automatique des CVs, un prompt a été conçu et intégré dans le système afin de permettre au modèle LLaMA 3 - 70B (via Groq) d'analyser le contenu d'un CV en fonction des compétences requises pour un poste donné.

Ce prompt joue un rôle essentiel dans le processus de présélection : il permet d'extraire de manière automatique et standardisée les correspondances entre les compétences d'un candidat et celles attendues par l'entreprise. Grâce à ce mécanisme, nous obtenons une évaluation objective comprenant :

- Les compétences présentes dans le CV et correspondant à celles demandées,
- Les compétences manquantes,
- Un score de correspondance (match score)

La figure ci-dessous illustre le prompt tel qu'il est envoyé au modèle de langage :

```

prompt = f"""
Analyze the provided CV text based ONLY on the "Required Skills" list below.

**CV Text:**
--- START CV ---
{cv_text[:15000]} ... [CV text potentially truncated]
--- END CV ---

**Required Skills:**
{json.dumps(skills_list, indent=2)}

**Instructions:**
1. Identify ONLY skills from the **Required Skills** list mentioned or strongly implied in the CV. List these in 'matched_skills'.
2. List ONLY skills from the **Required Skills** list NOT found in the CV in 'missing_skills'.
3. Calculate 'match_score' (integer 0-100): (Number of Matched Skills / Total Required Skills) * 100. Round to nearest integer.

**Output Format:**
Return ONLY* a single, valid JSON object containing ONLY the following 5 keys. NO extra text, comments, or markdown.
{{
  "matched_skills": ["Skill A", "Skill B", ...],
  "missing_skills": ["Skill C", "Skill D", ...],
  "match_score": <integer_percentage>,
}}
"""

```

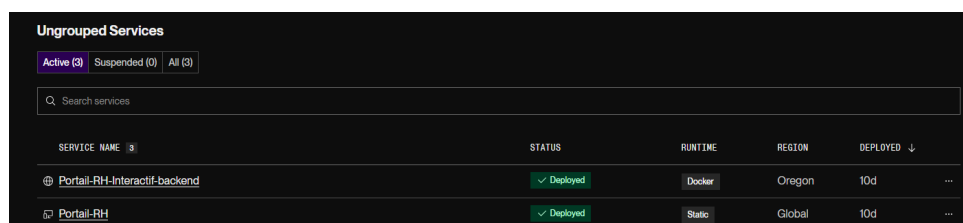
FIGURE 5.8 – Prompt envoyé au modèle LLaMA 3 pour l'analyse des CVs

5.8 Hébergement des applications Web et Mobile

5.8.1 Application Web - Render

Pour l'hébergement de l'application web, nous avons utilisé la plateforme **Render**, un service de cloud moderne qui permet de déployer facilement des applications web, des APIs, et des services statiques. Render offre une intégration continue (CI/CD), des déploiements automatiques depuis GitHub, une gestion simplifiée des bases de données, et une interface utilisateur intuitive.

L'application frontend développée avec *React* a été déployée comme un site statique sur Render, tandis que le backend Spring Boot a été hébergé comme un service web connecté à une base de données MongoDB également sur Render.



Ungrouped Services				
Active (3) Suspended (0) All (3)				
Search services				
SERVICE NAME	STATUS	RUNTIME	REGION	DEPLOYED
Portail-RH-Interactif-backend	✓ Deployed	Docker	Oregon	10d
Portail-RH	✓ Deployed	Static	Global	10d

FIGURE 5.9 – Capture d'écran de l'hébergement de l'application web sur Render

5.8.2 Application Mobile - Expo

5.8.2.1 Génération et Hébergement de l'APK

Pour générer le fichier APK de l'application Android, nous avons utilisé la commande suivante dans le dossier android du projet :

```
./gradlew assembleRelease
```

Cette commande produit un fichier `app-release.apk`. Ce fichier APK est signé et prêt à être installé sur un appareil Android en mode production.

Pour distribuer l'application, le fichier APK a été hébergé sur un service de stockage en ligne (par exemple Google Drive) afin de permettre un accès facile au téléchargement. Pour une distribution plus large et officielle, l'application pourra être publiée sur le Google Play Store en respectant les procédures de soumission et de validation imposées par Google.

5.9 Résultats Obtenus

5.9.1 Interfaces Web

5.9.1.1 Partie Admin : CRUD Offres, consulter les offres, consulter les candidats avec scores

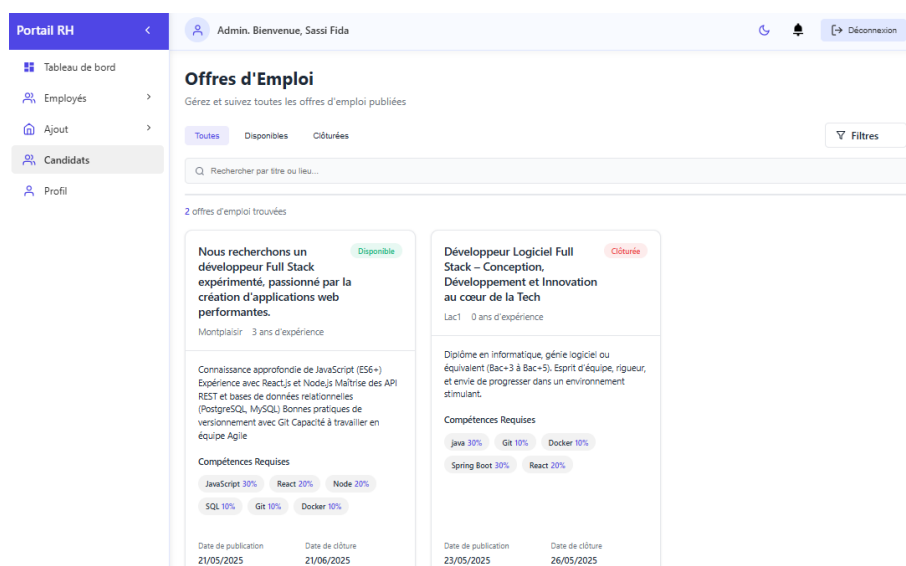


FIGURE 5.10 – Interfaces admin : consultation des offres

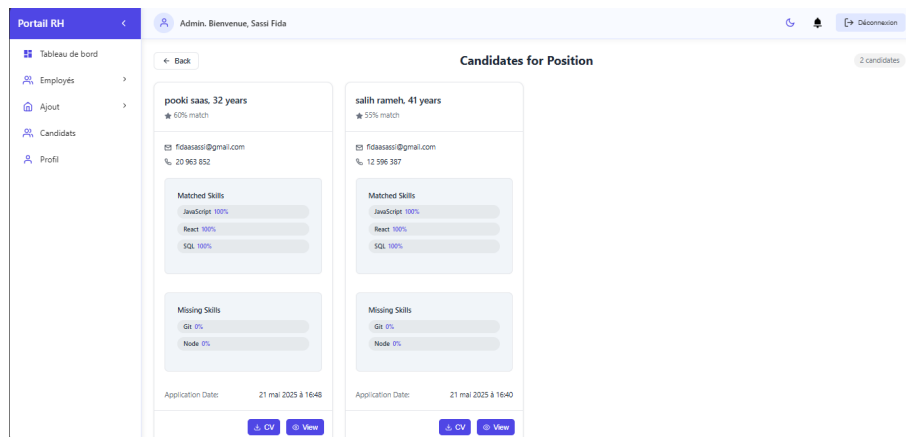


FIGURE 5.11 – Interfaces admin : Consultation des candidats qui ont postuler a une Offre

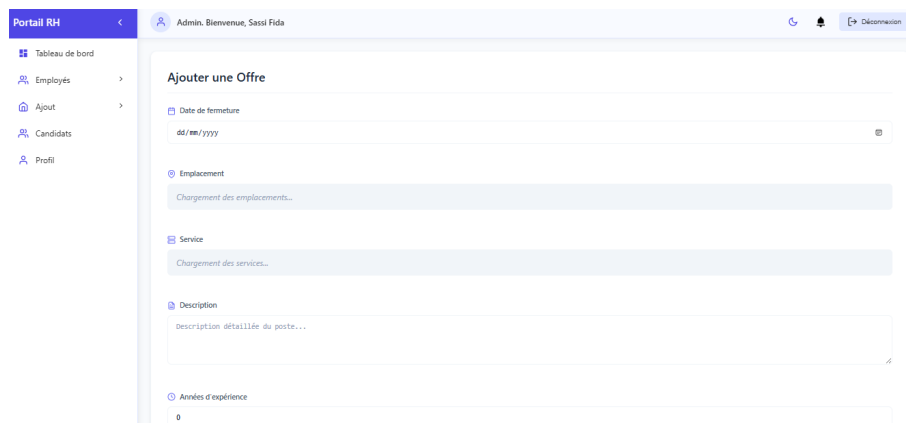


FIGURE 5.12 – Interfaces admin : ajout d'une offre

5.9.1.2 Partie Candidat : Postuler à une offre

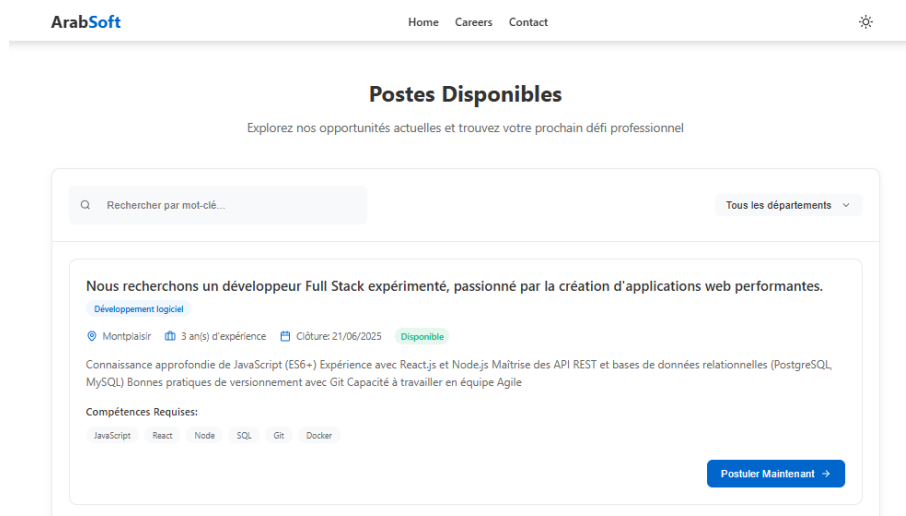


FIGURE 5.13 – Interfaces candidat : consultation des offres

The screenshot shows a web application interface for 'ArabSoft' with a navigation bar (Home, Careers, Contact) and a settings icon. A modal titled 'Postuler pour ce poste' is open, displaying a progress bar with two steps: '1 Informations Personnelles' (active) and '2 Contact & CV'. The form fields for Step 1 are: 'Nom *' (text input), 'Prénom *' (text input), and 'Âge *' (text input). A 'Suivant' button is at the bottom. The background shows a job listing for 'Nous recherchons un développeur' with details about location, experience, and required skills.

(a) Formulaire de postulation - étape 1

The screenshot shows the same modal, now at Step 2: 'Contact & CV'. The progress bar shows Step 1 as completed with a checkmark. A green success message 'CV téléchargé avec succès!' is displayed. The form fields for Step 2 are: 'Email *' (text input with value 'islemtk@gmail.com'), 'Téléphone * (8 chiffres)' (text input with value '56 165 126'), and 'CV (PDF, DOCX, TXT) *' (file upload area with a 'Parcourir' button). A 'Précédent' button is on the left, and an 'Envoyer ma candidature' button is on the right. The background job listing is still visible.

(b) Formulaire de postulation - étape 2

FIGURE 5.14 – Interfaces candidat : postuler à une offre

ANNEXES

A.1 Annexe 1 : Prompt utilisé pour l'analyse IA

Dans cette annexe, nous présentons le prompt utilisé pour interroger le modèle d'intelligence artificielle (LLaMA 3 via Groq API) dans le cadre de l'analyse des CV. Ce prompt permet d'extraire les compétences d'un candidat à partir de son CV.

Prompt complet

```
prompt = """
Analyze the provided CV text based ONLY on the "Required Skills" list below.

**CV Text:**
--- START CV ---
{cv_text[:15000]} ... [CV text potentially truncated]
--- END CV ---

**Required Skills:**
{json.dumps(skills_list, indent=2)}

**Instructions:**
1. Identify ONLY skills from the **Required Skills** list mentioned or strongly implied in the CV. List these in 'matched_skills'.
2. List ONLY skills from the **Required Skills** list NOT found in the CV in 'missing_skills'.
3. Calculate 'match_score' (integer 0-100): (Number of Matched Skills / Total Required Skills) * 100. Round to nearest integer.

**Output Format:**
Return *ONLY* a single, valid JSON object containing ONLY the following 5 keys. NO extra text, comments, or markdown.
{{
  "matched_skills": ["Skill A", "Skill B", ...],
  "missing_skills": ["Skill C", "Skill D", ...],
  "match_score": <integer_percentage>,
}}
"""
```

FIGURE A.1 – Python Prompt

Pour cette Offre :



FIGURE A.2 – Exemple d'une offre pour postuler

Analyse ce CV et identifie les compétences clés du candidat.

Présente les résultats sous forme de liste avec des catégories claires.

Le CV à analyser est le suivant :

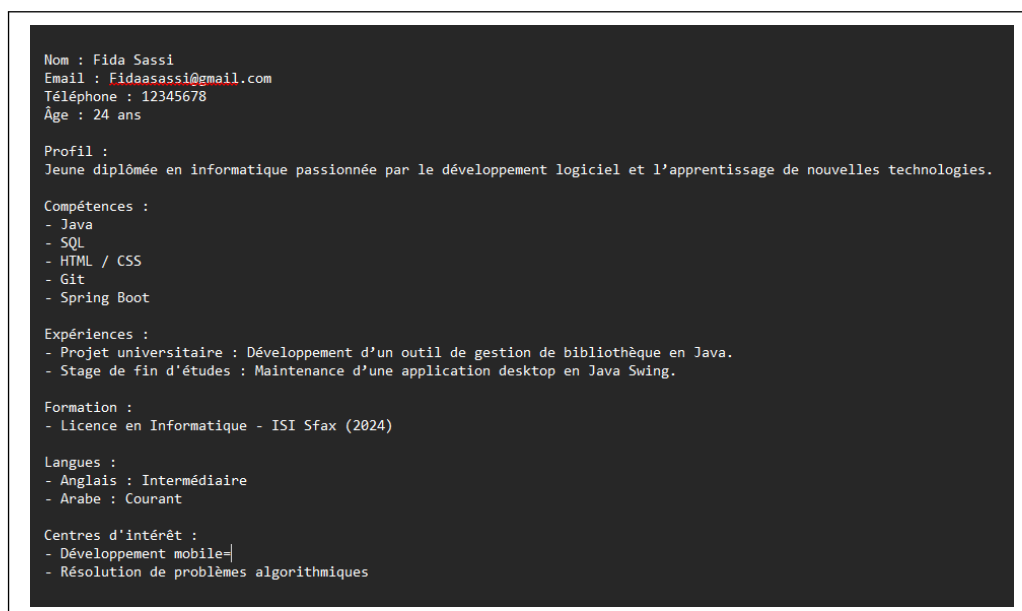


FIGURE A.3 – Exemple d'un CV soumis à l'analyse IA

Explications

Ce prompt a été conçu pour guider le modèle afin d'extraire les informations les plus pertinentes du CV. Il fournit un cadre clair à l'IA pour éviter les réponses trop vagues ou imprécises. L'objectif est d'obtenir une structure cohérente facilitant le traitement automatique des résultats.

Résumé de l'analyse générée

- **Compétences détectées (matched skills) :** Java , Spring Boot , Git
- **Compétences manquantes (missing skills) :** React , Docker

— Pourcentage d'appariement : 70

A.2 Annexe 2 : Résultat réel de l'analyse IA d'un candidat

L'image ci-dessous montre un exemple réel d'une réponse générée par le module d'analyse IA. Cette réponse a été enregistrée dans la base de données après traitement du CV du candidat. Le score de correspondance, les compétences détectées et manquantes sont extraits automatiquement à partir du retour JSON de l'IA.

```
_id: ObjectId('683078bea19b8f1469ec7698')
nom: "Trabelsi"
prenom: "Sami"
age: 25
email: "fidaasassi@gmail.com"
numTel: "12 596 387"
cvFilePath: "C:\Users\MSI\Desktop\Portail-RH-Interactif\backend\PortailRH\uploads\c_"
dateCandidature: 2025-05-23T13:31:42.659+00:00
matchedSkills: Array (3)
  0: "Git"
  1: "Spring Boot"
  2: "java"
missingSkills: Array (2)
  0: "Docker"
  1: "React"
score: 7
matchPercentage: 70
accepted: true
candidatures: Array (1)
_class: "com.example.PortailRH.Model.candidat"
```

FIGURE A.4 – Exemple de résultat IA stocké en base de données pour un candidat

Notes techniques

- Le prompt est envoyé via une requête HTTP POST à l'API Groq, qui utilise le modèle LLaMA 3.
- Le fichier du CV est traité côté serveur en Python pour extraire le texte brut.
- Le texte est ensuite inséré dans le prompt et envoyé au modèle.
- La réponse IA (au format JSON) est analysée et persistée dans MongoDB avec les informations suivantes :
 - Compétences reconnues
 - Compétences manquantes
 - Pourcentage de compatibilité



BIBLIOGRAPHIE

- [1] ARABSOFT, Inc. *arabsoft identification* **[en ligne]**. Disponible sur :
<https://www.arabsoft.com.tn/>
- [2] Expo, Inc. *Expo Documentation* **[en ligne]**. Disponible sur :
<https://expo.dev/>
- [3] W3C, Inc. *CSS, Cascading Style Sheets Documentation* **[en ligne]**. Disponible sur :
<https://www.w3.org/Style/CSS/>
- [4] Render, Inc. *Render, Cloud Application Hosting Documentation* **[en ligne]**. Disponible sur :
<https://www.w3.org/Style/CSS/>
- [5] Overleaf, Inc. *Collaborative LaTeX Editor, Documentation* **[en ligne]**. Disponible sur :
<https://www.overleaf.com/>
- [6] Postman, Inc. *Postman API Platform Documentation* **[en ligne]**. Disponible sur :
<https://www.postman.com/>
- [7] GitHub, Inc. *GitHub – Where the world builds software. Documentation* **[en ligne]**.
Disponible sur :
<https://github.com/>
- [8] Pivotal Software, Inc. *IntelliJ IDEA Documentation* **[en ligne]**. Disponible sur :
<https://www.jetbrains.com/idea/>
- [9] React, Inc. *React Documentation* **[en ligne]**. Disponible sur :
<https://react.dev/>
- [10] React Native, Inc. *React Native Documentation* **[en ligne]**. Disponible sur :
<https://reactnative.dev/>
- [11] MongoDB, Inc. *MongoDB Documentation* **[en ligne]**. Disponible sur :
<https://www.mongodb.com/resources/products/fundamentals/why-use-mongodb>
- [12] Groq, Inc. *Groq API Documentation* **[en ligne]**. Disponible sur :
<https://artificialanalysis.ai/providers/groq>
- [13] PyMuPDF, Inc. *PyMuPDF Documentation* **[en ligne]**. Disponible sur :
<https://pymupdf.readthedocs.io/en/latest/>
- [14] LLAMA, Inc. *LLAMA Documentation* **[en ligne]**. Disponible sur :
https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3/

- [15] Langchain, Inc. *Langchain Documentation* [en ligne]. Disponible sur :
<https://python.langchain.com/docs/introduction/>



CONCLUSION GÉNÉRALE

Au terme de notre projet de fin d'études, nous avons développé l'**Application RH pour la Gestion des Demandes des Employés et des Recrutements**, une solution web et mobile visant à simplifier et optimiser les processus RH.

Grâce aux compétences acquises à l'**ISETN** et à l'expérience pratique du stage, nous avons atteint nos objectifs : créer une application fonctionnelle, moderne (avec **React.js**, **Spring Boot**, **MongoDB**), accompagnée d'une version mobile en **React Native**. Nous avons également utilisé la plateforme **Render** pour héberger les services et garantir un déploiement fluide.

Ce projet a été extrêmement enrichissant, nous permettant d'approfondir nos compétences techniques, de surmonter des défis liés à l'intégration technologique et d'appliquer des méthodologies agiles comme **SCRUM**.

Pour l'avenir, plusieurs améliorations restent envisageables :

- Développement d'un module d'intelligence artificielle plus approfondi, capable d'analyser en profondeur les candidatures et de s'intégrer dans un processus global d'optimisation RH.
- Mise en place d'outils analytiques avancés (dashboards, statistiques) pour améliorer la prise de décision RH.
- Optimisation continue des performances et de l'expérience utilisateur sur les plateformes web et mobile.

En conclusion, cette expérience a consolidé nos acquis, renforcé notre capacité à travailler en équipe et ouvert la voie à de futures évolutions ambitieuses pour **PortailRH**. Nous remercions chaleureusement notre encadrant académique, notre tuteur en entreprise et toutes les personnes qui nous ont soutenus.

Application RH pour la Gestion des Demandes des Employés et des Recrutements

Islem Turki
Sassi Fida

Résumé :

Ce projet s'inscrit dans le cadre de notre projet de fin d'études réalisé au sein de la société **Arab Soft**. Il porte sur la conception et le développement d'un **Portail RH** visant à digitaliser les processus de gestion des ressources humaines, tels que la gestion des congés, des documents administratifs et des demandes internes.

L'objectif principal est de mettre en place une plateforme **intuitive, performante et sécurisée**, offrant aux collaborateurs et responsables une interface centralisée pour suivre et traiter les différentes démarches RH. Le projet repose sur une **architecture moderne**, en utilisant les technologies **Java / Spring Boot** pour le backend et **React** pour le frontend.

Le développement du portail a été mené en suivant la **methodologie agile Scrum**, permettant une gestion itérative et collaborative du projet. Des **sprints** réguliers ont permis d'assurer une livraison continue de fonctionnalités, tout en intégrant les retours du client afin d'adapter le produit aux besoins réels du service RH.

Ce portail vise à améliorer la **productivité**, la **traçabilité** et la **fluidité** des échanges au sein du département des ressources humaines.

Mots clés : Java, Spring Boot, React, Git, Scrum, Portail RH, digitalisation.

Abstract :

This project was carried out as part of our final year internship at **Arab Soft**. It focuses on the design and development of an **HR Portal** aimed at digitizing human resources management processes, such as leave requests, administrative document handling, and internal request tracking.

The main objective is to implement an **intuitive, efficient, and secure** platform, providing employees and managers with a centralized interface to monitor and manage various HR-related processes. The project is built on a **modern architecture**, using **Java / Spring Boot** for the backend and **React** for the frontend.

The development was conducted following the **agile Scrum methodology**, allowing for an iterative and collaborative project management approach. Regular **sprints** ensured the continuous delivery of new features while incorporating client feedback to better align the product with actual HR department needs.

The portal aims to enhance **productivity, traceability, and the efficiency** of communication within the HR department.

Keywords : Java, Spring Boot, React, Git, Scrum, HR Portal, digitization.