



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Direction Générale des Etudes Technologiques Institut Supérieur des
Etudes Technologiques du Kef Département de Technologie de
l'Informatique

RAPPORT DE PROJET DE FIN D'ETUDES

PRESENTE POUR L'OBTENTION DU DIPLOME NATIONAL DE LICENCE
EN TECHNOLOGIE DE L'INFORMATIQUE

SPECIALITE : Développement Système d'information

REALISE PAR :

Ahmed BelKahla et Fraj Cherni

Développement d'un site web contenant la
gestion de ressources humaines et la gestion de
stock pour la société

Encadrant académique : Mr. Mohamed Hamdi

Encadrant professionnel : Mme. Oumayma Suissi

Année Universitaire : 2022-2023



RAPPORT DE FIN D'ÉTUDE

Présenté en vue de l'obtention de
DIPLOME EN LICENCE TECHNOLOGIE DE L'INFORMATIQUE

Mention : Informatique
Spécialité : Développement des systèmes d'informations

Par
AHMED BELKAHLA ET FRAJ CHERNI

Développement d'un site web contenant la
gestion de ressources humaines et la gestion de
stock pour la société



Encadrant académique : Mr. Mohamed Hamdi
Encadrant professionnel : Mme. Oumayma Suissi

Année Universitaire : 2022-2023

Développement d'un site web contenant la gestion de ressources humaines et la gestion de stock pour la société

Dédicaces

Prévue de respect, de gratitude, et de reconnaissance
Je dédie ce travail

À mes chers parents

Je vous dois ce que je suis aujourd'hui grâce à votre amour, à votre patience vos innombrables sacrifices. Que ce modeste travail soit pour vous une petite compensation et reconnaissance envers ce que vous avez fait d'incroyable pour moi. Que dieu, le tout puissant, vous préserve et vous procure santé et longue vie afin que je puisse, à mon tour, vous combler.

À l'âme de mes grands parents

À ma famille

Pour leur soutien moral et leur encouragement.

À mes professeurs

Pour leurs aides et soutien pendant toute la période de ma formation.

À mes chers amis

Pour leur témoignage de la fraternité qui nous associe.

Merci de tout mon cœur

Ahmed Belkahla et Fraj Cherni

Remerciements

je tiens tout d'abord à remercier **Mr.Mohamed hamdi**, mon encadrant académique pour sa disponibilité, sa patience, son suivi pour faire évoluer le processus du projet . Qu'il trouve ici l'expression de toute ma gratitude.

Je tiens tout d'abord à remercier **Mme Suissi oumaïma**, pour sa disponibilité, sa patience, et sa contribution pour faire évoluer les choses et réussir mon projet.Ainsi pour sa bienveillance et sympathie et son suivi tout au long du stage.

Je suis très reconnaissante à ma famille, ma mère et mon père, pour leur support et encouragements continus.

Table des figures

Liste des tableaux

1	Cadre général du projet	3
1.1	Organisme d'accueil	3
1.1.1	Les Activités de comelit	4
1.1.2	Les objectifs de comelit :	4
1.2	Étude de l'existant	4
1.2.1	Description de l'existant	4
1.2.2	Critique de l'existant	4
1.2.3	Solution envisagées	5
1.2.4	Méthodologie adopté	5
1.2.4.1	La méthodologie SCRUM d'agile	5
1.2.4.2	Les rôles de la méthode SCRUM	6
2	Spécification des besoins	8
2.1	Spécification des besoins :	8
2.1.1	Identification des acteurs	8
2.1.2	Besoins fonctionnels	9
2.1.3	Besoins non fonctionnels	10
2.2	Pilotage du projet avec SCRUM	11
2.2.1	User Stories	11
2.2.2	Equipe SCRUM	11
2.2.3	Product Backlog	12
2.2.4	Planification Des Sprints	13
2.3	Architecture	14
2.4	Environnement de travail	14
2.4.1	Environnement matériel	14
2.4.2	Environnement Technique	15
2.5	Diagramme de cas d'utilisation global	17
2.6	Diagramme de Classes	18
3	Release 1	20
3.1	Sprint 1 : Authentication est Gestion des employées	20
3.1.1	Backlog du sprint	21

3.1.2	Analyse	21
3.1.2.1	Diagramme de cas d'utilisation Sprint 1	21
3.1.2.2	Description textuelle	22
3.1.3	Conception	26
3.1.3.1	Organigramme de conception	26
3.1.4	Réalisation	31
3.1.5	Test	33
3.2	Sprint 2 « Gestion des Congés »	34
3.2.1	Backlog du Sprint 2	35
3.2.2	Analyse	36
3.2.2.1	Diagramme du cas d'utilisation « Sprint 2 »	36
3.2.2.2	Description textuelle :	37
3.2.3	Conception	38
3.2.3.1	Diagramme de classes	38
3.2.4	Diagramme de séquence	39
3.2.5	Réalisation	42
3.2.6	Test	43
3.2.7	Conclusion	44
4	Release 2	45
4.1	Sprint 1 : Gestion des Missions Et Gestion de Stock	46
4.1.1	Backlog du sprint	46
4.1.2	Analyse	46
4.1.3	Description textuelle	48
4.1.4	Conception	53
4.1.5	Réalisation	60
4.1.6	Test	61
4.2	Sprint 2 « Gestion de site web »	62
4.2.1	Backlog du sprint	62
4.2.2	Analyse	62
4.2.3	Conception	67
4.2.4	Réalisation	71
4.2.5	Conclusion	71
	Bibliographie	73

TABLE DES FIGURES

1.1	Logo entreprise d'accueil	3
1.2	Cycle de vie d'un projet SCRUM	6
1.3	Rôles de méthode SCRUM	6
2.1	découpage des User Stories	11
2.2	découpage en Release	13
2.3	Architecture MERN Stack	14
2.4	La pile technologique MERN	15
2.5	Visual studio code	16
2.6	Draw.io	16
2.7	PostMan	16
2.8	Bootstrap	17
2.9	Canva	17
2.10	Diagramme de cas d'utilisation globale	18
2.11	Diagramme de classes	19
3.1	Diagramme de cas d'utilisation du Sprint 1	22
3.2	Organigramme de conception	26
3.3	Diagramme de classe du Sprint 1	27
3.4	Diagramme de Séquence "Authentification"	27
3.5	Diagramme Séquence "Ajouter un employée"	28
3.6	Diagramme Séquence "Controller les accès"	28
3.7	Diagramme Séquence "Déconnexion "	29
3.8	Diagramme d'activité "Authentification "	29
3.9	Diagramme d'activité "Ajouter un employée "	30
3.10	Interface " Authentification"	31
3.11	Interface d'accueil d'administrateurs	32
3.12	Interface de gestion des employés	32
3.13	Iterface de gestion des Salaires	33
3.14	Iterface de gestion des accès	33
3.15	Diagramme du cas d'utilisation « Sprint 2 »	36
3.16	diagramme de classe sprint 2	38
3.17	Diagramme de séquence " demander un congé "	39
3.18	Diagramme de séquence " traiter un congé "	39
3.19	Diagramme d'activité " demander un congé "	40
3.20	Diagramme d'activité " Accepter un congé "	41

3.21	Interface demander un conge	42
3.22	Interface liste de congés	42
3.23	Interface de Traitement de congés	43
4.1	Release 2	45
4.2	Diagramme de cas d'utilisation "Gestion des missions "	46
4.3	Diagramme de cas d'utilisation "Gestion des missions "	47
4.4	Diagramme de cas d'utilisation "Gestion les Stocks "	47
4.5	Diagramme de classe sprint 1	53
4.6	Diagramme de séquence"Ajouter une mission "	54
4.7	Diagramme de séquence "supprimer une mission "	54
4.8	Diagramme de séquence"impression d'une mission "	55
4.9	Diagramme de séquence «Ajouter un Bon d' Entrée »	55
4.10	Diagramme de séquence «Ajouter un Bon de Transfert »	56
4.11	Diagramme de séquence «Modifier un bon de sortie »	56
4.12	Diagramme d'activité «Modifier une mission »	57
4.13	Diagramme d'activité «Modifier un Bon d'entrée »	58
4.14	Diagramme d'activité «Ajouter un Bon de transfert »	59
4.15	Interface de Gestion des missions	60
4.16	Interface demander un conge	61
4.17	Diagramme de cas d'utilisation "Gestion de site web "	63
4.18	Diagramme de séquence «Disposer une offre d'emploi »	67
4.19	Diagramme de classe « Gérer site web »	67
4.20	Diagramme de séquence «Traiter une offre d'emploi »	68
4.21	Diagramme de séquence «Supprimer un Evenement »	68
4.22	Diagramme de séquence «Traiter une demande de stage »	69
4.23	Diagramme d'activité «Ajouter une offre d'emploi »	69
4.24	Diagramme d'activité «Postuler une domande d'emplo »	70
4.25	Diagramme d'activité «Ajouter un événement »	70
4.26	Interface d'offre de Comelit	71

LISTE DES TABLEAUX

2.1	Identification des acteurs	9
2.2	Les besoins fonctionnels	10
2.3	Equipe SCRUM	11
2.4	Planification des Sprints	14
3.1	Backlog du Sprint 1	21
3.2	Cas d'utilisation : Authentification	23
3.3	Cas d'utilisation : Gérer les Employeer	23
3.4	Cas d'utilisation : Ajouter un Employée	24
3.5	Cas d'utilisation : modifier un Employée	24
3.6	Cas d'utilisation : Gestion des accès	25
3.7	Cas d'utilisation : Gérer les accès	25
3.8	Cas d'utilisation : Gestion des salaires	26
3.9	Test du première Sprint	34
3.10	Backlog du Sprint 2	35
3.11	Cas d'utilisation : Gérer les congé	37
3.12	Cas d'utilisation : Demander un congé	37
3.13	Cas d'utilisation : Demander un congé	38
3.14	Test du deuxième Sprint	43
4.1	Backlog du Sprint 1 de release 2	46
4.2	Cas d'utilisation : Gérer les missions	48
4.3	Cas d'utilisation : Ajouter une mission	48
4.4	Cas d'utilisation : modifier une mission	49
4.5	Cas d'utilisation : supprimer une mission	49
4.6	Cas d'utilisation : Gérer le stock	50
4.7	Cas d'utilisation : Ajouter un article	50
4.8	Cas d'utilisation : Ajouter un bon de transfert	51
4.9	Cas d'utilisation : supprimer un bon de transfert	52
4.10	Cas d'utilisation : Ajouter un bon d'entrée	52
4.11	Test du deuxième Sprint	62
4.12	Backlog du Sprint 1 de release 2	63
4.13	Cas d'utilisation : Gérer le site web	64
4.14	Cas d'utilisation : Disposer une offre d'emploi	64
4.15	Cas d'utilisation : Traiter une demande de stage	65
4.16	Cas d'utilisation : Traiter une demande d'emploi	65

4.17 Cas d'utilisation : Supprimer un événement	66
4.18 Cas d'utilisation : modifier un offre de stage	66

Acronyme

UML Unified Modeling Language

SCRUM Systematic Customer Resolution Unraveling Meeting

API Application Programming Interface

INTRODUCTION GÉNÉRALE

Dans nos jours, l'avancement technologique ne cesse d'augmenter ce qui a créé une concurrence au niveau des entreprises qui ne s'arrêtent donc pas de développer des solutions et des nouvelles méthodes pour faire face aux défis technologiques et automatiser leurs différents services. .

La vie moderne est très chargée grâce aux développements technologiques de l'informatique ce qui a renforcé la concurrence entre plusieurs fournisseurs de service.

En ce qui concerne les méthodologies de travail employées dans notre projet, nous avons opté pour les méthodologies agiles, plus précisément pour la méthode SCRUM. .
Ce rapport résume notre travail. Il est organisé en quatre chapitres :

- **Le premier chapitre** est consacré à la présentation générale de l'organisme d'accueil ainsi le cadre de projet, qui va placer le projet dans son contexte général.
- **Le deuxième chapitre** intitulé « Spécification des besoins » sera consacré pour une analyse des besoins, une présentation des tâches à faire au cours de ce projet, et aussi pour la conception générale suivi par la présentation de l'architecture générale du projet et l'environnement du travail. .
- **Les deux derniers chapitres** « Release 1 » et « Release 2 » constituent le corps de notre rapport. Ils comportent la première version de notre projet défini en se basant sur la méthodologie Scrum. Nous présentons tout au long de ces chapitres la spécification, la conception et la réalisation des sprints. .
- **Conclusion générale** se focalise sur l'enseignement retiré ,les principales observations et quelques perspectives intéressantes concernant ce stage.

CHAPITRE 1

CADRE GÉNÉRAL DU PROJET

Introduction

Dans ce chapitre, nous mettrons notre projet dans son contexte général. Nous commençons par présenter l'organisme d'accueil, son organigramme et sa structure. Ensuite nous avons tenté d'analyser l'existant pour mieux comprendre les problèmes afin de les corriger. Enfin, nous passons à une présentation de la méthodologie adoptée ainsi que la planification de notre projet.

1.1 Organisme d'accueil

Dans cette section, nous présentons l'organisation hôte **Comelit** et ses domaines d'activité



FIGURE 1.1 – Logo entreprise d'accueil

La société Comelit (CRED) est une société italienne renommée mondialement avec plus de 50 ans d'expérience. La technologie et les systèmes de sécurité sont des composants essentiels de l'expérience de Comelit.

Au fil du temps, Comelit a acquis un savoir-faire de haut niveau dans les systèmes de vidéo-surveillance, d'alarme anti-intrusion, de domotique et de contrôle des accès.

La branche de la société Comelit est gérée par Mr Walid Barreh disposant d'un centre de développement situé au Pôle El Ghazala des Technologies de Communications en Ariana. Comelit a opéré sur des perspectives ambitieuses, une étude de marché cohérente, et une équipe de travail expérimentée et douée des dernières innovations en High Tech, afin de surmonter les challenges d'une période économique délicate.

1.1.1 Les Activités de comelit

Les activités principales de Comelit sont :

- La vidéoprotection** : une gamme complète de produits pour la vidéoprotection. Analogique, IP et HD sur câble coaxial.
- L'anti-intrusion** : un système complet en mesure de répondre à toutes les exigences en termes de sécurité via des systèmes câblés ou via radio
- La domotique** : un système qui assure un contrôle total de votre espace, pour un maximum de confort et de sécurité
- Le contrôle des accès** : un système bénéficiant de la technologie Mifare.
- Évacuation en cas d'incendie** : un système de détection d'incendie avec technologie classique et adressable.

1.1.2 Les objectifs de comelit :

COMELIT possède comme deux objectifs comme suit :

Mettre l'accent sur l'humanité et ses valeurs : Comelit a toujours placé l'humanité au centre de sa vision. Cette démarche humaniste se manifeste par la focalisation de l'entreprise sur ses collaborateurs, l'organisation de son environnement de travail, l'éthique de ses relations avec les fournisseurs et les distributeurs et le partenariat qu'elle entretient avec les installateurs. Chaque produit Comelit exprime cette vision, construite sur la base de valeurs durables.

Une innovation constante et une technologie conviviale : cherche constamment à l'avenir de sa mission. l'innovation grâce à l'investissement continu dans la main-d'œuvre et la technologie trouve un débouché dans la conception de nouvelles solutions dédiées à améliorer la qualité de la vie quotidienne en maximisant la simplicité et la fonctionnalité.

1.2 Étude de l'existant

1.2.1 Description de l'existant

Ce projet, intitulé « Développement d'un site web ayant deux rôles : public (voir les offres d'emploi et de stage, postuler une candidature spontanée) et intranet (la gestion des ressources humaines et de stock), pour la société », rentre dans le cadre de notre projet de fin d'études. Il vient conclure notre formation en licence nationale en technologie de l'informatique à Iset Kef. Il a été proposé par le directeur de CRED, M. Walid Barreh, qui nous a intégrés au sein de son équipe pour nous familiariser avec l'environnement professionnel et mettre en pratique ce que nous avons acquis pendant les années de notre formation

1.2.2 Critique de l'existant

La problématique principale de l'entreprise réside dans la gestion inefficace de ses ressources humaines et de son stock. Actuellement, l'entreprise ne dispose pas d'un système centralisé pour la communication des offres d'emploi et de stage, ni pour la gestion des candidatures spontanées. De plus, la gestion du stock est réalisée de manière manuelle, ce qui engendre des erreurs, des retards et une perte de productivité.

1.2.3 Solution envisagées

Le projet consiste à développer un site web doté de deux rôles distincts : un rôle public et un rôle intranet.

Le rôle public permettra aux visiteurs de consulter les offres d'emploi et de stage, ainsi que de postuler une candidature spontanée. Le rôle intranet sera dédié à la gestion des ressources humaines et du stock de l'entreprise.

La solution proposée offrira les avantages suivants :

Amélioration de la communication : Le site web permettra à l'entreprise de diffuser efficacement les offres d'emploi et de stage, ce qui attirera un plus grand nombre de candidats qualifiés. De plus, les candidatures spontanées pourront être facilement collectées et gérées.

Optimisation de la gestion des ressources humaines : Le rôle intranet du site web offrira des fonctionnalités de gestion complètes, telles que la création et la gestion des profils des employés, le suivi des congés et des absences, ainsi que la gestion des évaluations de performance. Cela permettra à l'entreprise de mieux organiser ses ressources humaines et de favoriser une gestion plus efficace.

Automatisation de la gestion du stock : Le site web proposera des fonctionnalités de gestion des stocks, telles que l'enregistrement des entrées et des sorties, le suivi des mouvements de stock, et la génération de rapports. L'automatisation de ces processus réduira les erreurs, les pertes de temps et améliorera la précision des données.

Centralisation des informations :

Le site web permettra d'avoir un point centralisé pour accéder aux informations relatives aux offres d'emploi, aux candidatures, aux ressources humaines et au stock. Cela facilitera la recherche d'informations, la prise de décision et la collaboration entre les différents départements de l'entreprise.

1.2.4 Méthodologie adopté

1.2.4.1 La méthodologie SCRUM d'agile

Le choix d'un processus et d'une méthodologie de travail est une étape essentielle afin de faciliter et d'accélérer la transformation des besoins des utilisateurs en un système logiciel. La méthode **Scrum** est une méthode agile, créée en 2002, dont le nom est un terme emprunté au rugby qui signifie « **la mêlée** ». Elle s'appuie sur le découpage des projets en itérations encore nommées « **sprints** ». Un sprint peut avoir une durée qui varie généralement entre deux semaines et un mois.

Avant chaque sprint, les tâches sont estimées en temps et en complexité à l'aide de certaines pratiques comme le « **planning poker** », une manière ludique de chiffrer la complexité des tâches ou évolutions à l'aide de cartes à l'instar du célèbre jeu dont le nom est repris. Ces estimations permettent à la fois de planifier les livraisons, mais aussi d'estimer le coût de ces tâches auprès du client. Les fonctionnalités (encore appelées « **user stories** ») qui font l'objet d'un sprint constituent ce que l'on appelle un « **sprint backlog** » du produit éventuellement livrable à la fin du sprint. Il est nécessaire de distinguer le sprint backlog du « Product backlog » qui lui correspond à l'ensemble des fonctionnalités attendues pour le produit sur l'ensemble des sprints

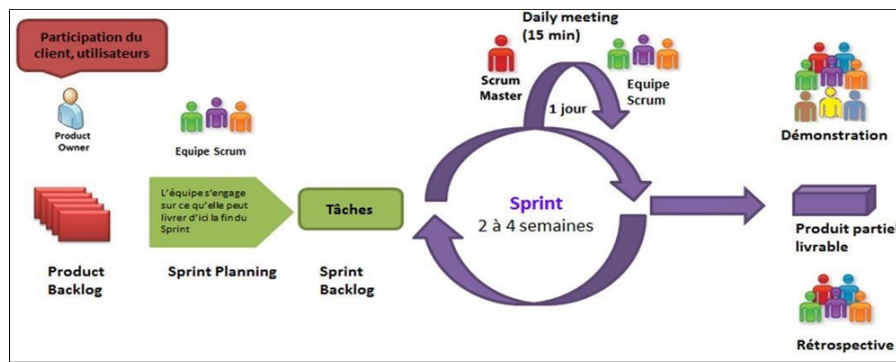


FIGURE 1.2 – Cycle de vie d'un projet SCRUM

1.2.4.2 Les rôles de la méthode SCRUM

- **Le Product Owner (le propriétaire du produit)** : il s'agit du représentant officiel du client au sein d'un projet SCRUM. Il est l'interlocuteur principal du SCRUM Master et des membres de l'équipe. Il définit les besoins du produit et rédige les spécifications. Il peut se faire aider de responsables fonctionnels pour la rédaction des spécifications. Il est également chargé de définir et prioriser les user stories pour chaque sprint.
- **Le SCRUM Master (le directeur du produit)** : il s'agit d'une personne chargée de veiller à la mise en application de la méthode et au respect de ses objectifs. Il ne s'agit pas d'un chef de projet, mais d'une personne chargée de lever les obstacles éventuels qui empêcheraient l'avancement de l'équipe et du projet pendant les différents sprints.
- **Le SCRUM Team (l'équipe de SCRUM)** : ce sont les personnes chargées de la réalisation du sprint et d'un produit utilisable en fin de sprint. Il peut s'agir de développeurs, architectes, personnes chargées de faire des tests fonctionnels...



FIGURE 1.3 – Rôles de méthode SCRUM

Conclusion

Ce chapitre constitue une partie introductive dans laquelle une présentation de l'organisme d'accueil a été élaborée en premier lieu, avec la présentation du cadre de projet et la méthodologie utilisée. L'objectif du deuxième chapitre est la conception du projet.

CHAPITRE 2

SPÉCIFICATION DES BESOINS

Introduction

Dans ce chapitre nous allons présenter les acteurs principaux de notre application. Ensuite, nous allons identifier les besoins fonctionnels et non fonctionnels du projet. Après, nous allons présenter les User stories, l'équipe SCRUM, le Product Backlog ainsi que la planification des sprints. Par la suite, nous enchaînons avec la modélisation de diagramme de cas d'utilisation global du projet. Enfin, nous allons clôturer le chapitre par l'identification de l'architecture de l'application et l'environnement matériel et logiciel.

2.1 Spécification des besoins

2.1.1 Identification des acteurs

Un acteur est une entité extérieure au système modélisé, et qui interagit directement avec lui. Notre projet fait intervenir trois acteurs principaux, comme le montre le tableau ci-après :

Acteur	Fonction
Administrateur	Gérer les employées Gérer les missions Gérer les congés (Annuel, spéciale, maladie, compensation, autorisation) Gérer les salaires Gérer les stocks Gérer le Site web

Visiteur	Consulter le site web de l'entreprise Consulter les offres d'emplois et les offres des stages Postuler une candidature
Employé	Demander des congés (Annuel, Spéciale, Maladie, Compensation, Autorisation) Consulter la liste de demande de congé Consulter sans profil

TABLE 2.1 – Identification des acteurs

2.1.2 Besoins fonctionnels

Les spécifications fonctionnelles permettent de définir les différents besoins que notre application doit satisfaire pour répondre aux attentes des utilisateurs. Autrement dit, dans ce contexte, il nous faut bien décrire les détails des besoins afin de tracer un diagramme de cas d'utilisation global. L'application doit assurer les fonctionnalités suivantes :

Objectif	Description
Module connexion	Pour l'utilisateur (administrateur ou employé) : <ul style="list-style-type: none"> — L'utilisateur peut s'authentifier via un formulaire d'authentification par son adresse mail et son mot de passe
Dashboard des employés	Le Dashboard des employés comporte les fonctionnalités suivantes : <ul style="list-style-type: none"> — Demandes de congés : les employés peuvent faire des demandes de congé avec différents types tels que maladie, spéciale ... — Consultation des congés : les employés peuvent accéder à leurs demandes de congé. — Consultation du profil : les employés peuvent consulter leur profil.
Dashboard des administrateurs	Le Dashboard admin comporte la : <ul style="list-style-type: none"> — Gestion des Accès : L'administrateur peut contrôler les accès des utilisateurs en autorisant ou refusant les permissions. — Gestion des Employées : Ajout, modification et suppression des employées. — Gestion des Missions : création, modification et suppression des missions. — Gestion des Congés : Accepter ou refuser les demandes des congés. — Gestion des Stocks : gestion des Suivies, gestion des bon d'entrée ,de Bon de sortie et de Bon de transfert. — Gestion des Salaires : L'administrateur a la possibilité seulement de consulter la liste complète des salaires des employés de l'entreprise, aussi il est le seul qui fait des augmentations de salaire aux employés.

Dashboard des administrateurs	<p>Le Dashboard admin comporte la :</p> <ul style="list-style-type: none"> — Gestion des fonctionnalités de Site web : création, modification et suppression des offres d'emplois ou des stages. — Consulter les candidatures spontanées et les demandes de stage et de emploi reçues. Accepter ou refuser les candidatures/demandes en fonction des critères de sélection préétablis (compétences, expériences).
-------------------------------	---

TABLE 2.2 – Les besoins fonctionnels

2.1.3 Besoins non fonctionnels

Ces besoins sont les contraintes techniques exigées et les fonctionnalités nécessaires pour rendre le logiciel plus performant et qui se résument généralement en :

- **Sécurité** : Les données sensibles telles que les informations des employés, les salaires, les congés et les missions doivent être stockées en toute sécurité et protégées contre les attaques.
- **Fiabilité** : L'application doit être fiable et résistante aux erreurs. Elle doit être en mesure de gérer les erreurs et de récupérer rapidement en cas de panne par exemple dans le calcul des congés.
- **Performance** : L'application doit être rapide et réactive pour assurer une expérience utilisateur fluide. Les temps de réponse doivent être optimisés.
- **Ergonomie** : L'interface utilisateur doit être ergonomique, intuitive et facile à utiliser. Les éléments de l'interface doivent être clairement identifiés et organisés de manière logique.
- **Maintenabilité** : L'application doit être facilement maintenable et extensible. Le code doit être propre, modulaire et documenté pour permettre une évolution future de l'application.

2.2 Pilotage du projet avec SCRUM

2.2.1 User Stories

Les histoires d'utilisateurs sont des brèves descriptions simples d'une fonctionnalité présentée du point de vue de la personne qui souhaite utiliser la nouvelle fonctionnalité, généralement un utilisateur ou un client du système. Ils suivent généralement un modèle simple

En tant que [type d'utilisateur], je souhaite [souhait] afin de [but]

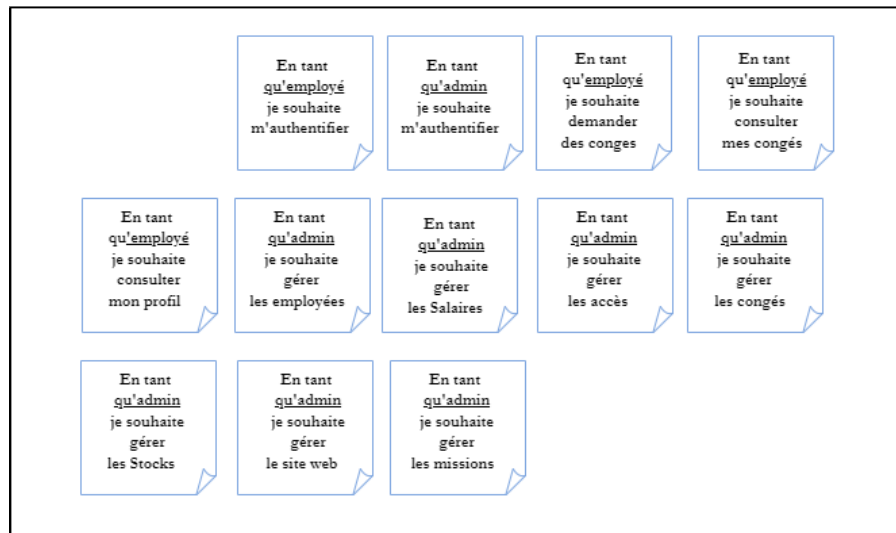


FIGURE 2.1 – découpage des User Stories

2.2.2 Equipe SCRUM

Pour notre application les rôles sont repartis comme suit :

Rôles SCRUM	Affecté à	Rôle
Product Owner	Walid Barreh	Le représentant des Administrateurs et des utilisateurs, définit les besoins, les priorités et les fonctionnalités et dirige l'activité de l'équipe de développement.
SCRUM Master	Oumayma Suissi	Elle assure globalement la Supervision de l'avancement du projet et des activités de l'équipe. Il assure également l'organisation des réunions et la bonne application de la méthode AGILE de par ce biais.
développement Team	Fraj cherni et Ahmed belkahla	Chargent de la réalisation de projet de la conception au développement .

TABLE 2.3 – Equipe SCRUM

2.2.3 Product Backlog

Le Product Backlog est l'artefact le plus important de SCRUM. En effet, c'est l'ensemble des caractéristiques fonctionnelles qui compose le produit. Ces caractéristiques sont appelées des histoires utilisateurs (user story). Chaque histoire utilisateur est caractérisée par une priorité définie par le Product Owner.

Les éléments du Product Backlog, appelé aussi les histoires utilisateurs, précisent la fonctionnalité désirée par le client, généralement, écrit sous la forme suivante « En tant que X, je veux Y ».

Le Backlog de produit présenté dans le tableau comprend les champs suivants :

— **ID** : C'est un nombre unique et auto-incrémenté pour chaque histoire utilisateur.

— **Thème** : Pour mieux ordonner les user stories.

— « **User story** » : Comporte la description d'une fonctionnalité désirée par le client suivant la forme « En tant que . . . Je veux . . . » o **Priorité** : Par rapport au client (Entreprise) représentée suivant la méthode "MoSCoW" qui est une technique possédant un objectif qui s'articule autour d'un accord entre le maître d'œuvre (MOE) et le maître d'ouvrage (MOA) sur l'importance des tâches que l'on va réaliser par rapport aux délais prévus. MoSCoW a pour signification

(le o ne représente rien c'est juste pour rendre le mot prononçable) :

— **M (Must have)** : doit être fait (vital).

— **S (Should have)** : devrait être fait dans la mesure du possible (essentiel).

— **C (Could have)** : pourrait être fait dans la mesure où cela n'a pas d'impact sur les autres tâches (confort).

— **W (Won't have)** : ne sera pas fait cette fois mais sera fait plus tard (luxue, c'est la zone d'optimisation budgétaire).

— **Complexité** : La complexité est évaluée entre 3 niveaux faible, moyenne et élevée.

Le Backlog produit de notre projet est le suivant :

Catégorie	User stories	Priorités	Complexité
S'authentifier	En tant que Utilisateur je peux accéder à mon compte grâce à une adresse mail et un mot de passe	M	Faible
Demande des congés	En tant que Utilisateur je peux demander des congés	C	Moyenne
Consulter le profil	En tant que utilisateur je peux consulter mon profil .	C	Faible
Gestion des accès	En tant que Admin je peux gérer les accès.	M	Moyenne
Gestion des congés	En tant que Admin je peux gérer les Conges .	M	Moyenne
Gestion des employés	En tant que Admin je peux gérer les employés.	M	Moyenne
Gérer les missions	En tant que Admin je Peux gérer les Missions.	M	Moyenne
Gestion des salaires	En tant que Admin je peux gérer les Salaires .	M	Moyenne
Gérer les Stocks	En tant que Admin je peux gérer les Stocks .	M	Moyenne
Gérer le Site web	En tant que Admin je peux gérer le site web.	M	Moyenne

2.2.4 Planification Des Sprints

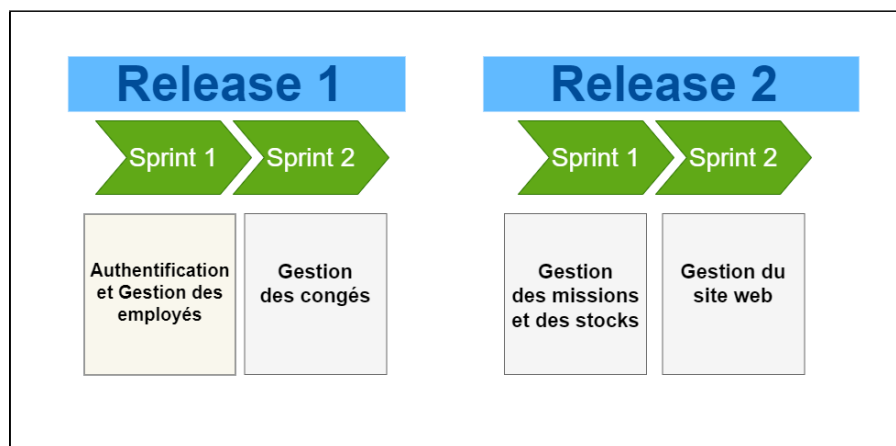


FIGURE 2.2 – découpage en Release

Release	Sprint	Nom Sprint	Durée
1	1	Authentification et Gestion des employés	Du 01/02/2023 au 14/03/2023
1	2	Gestion des congés	Du 14/03/2023 au 06/04/2023
2	3	Gestion des missions et des stocks	Du 07/04/2023 au 06/05/2023
2	4	Gestion du site web(eventement , candudature)	Du 07/05/2023 au 28/05/2023

TABLE 2.4 – Planification des Sprints

2.3 Architecture

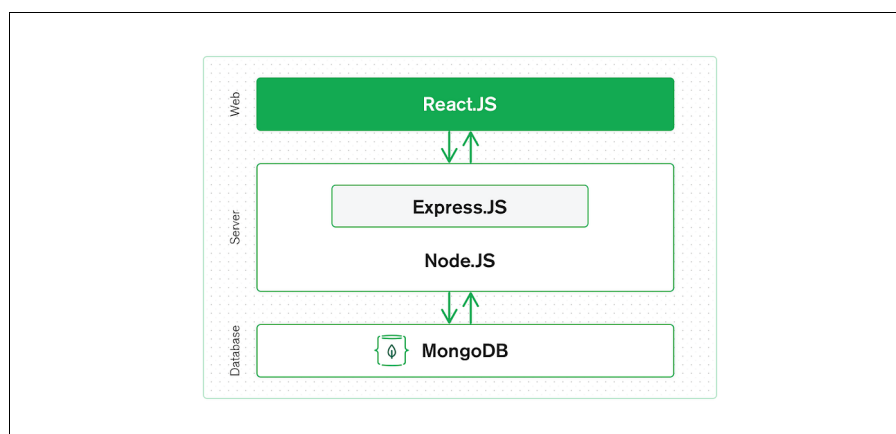


FIGURE 2.3 – Architecture MERN Stack

2.4 Environnement de travail

Dans cette partie, nous présenterons l'environnement matériel et technique relatif à la réalisation de l'application.

2.4.1 Environnement matériel

Nous avons développé notre application en utilisant un ordinateur portable dont les caractéristiques techniques sont décrites ci-dessous :

Premier ordinateur :

Pc Portable	Asus
RAM	8 Go
Microprocesseur	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz
Disque dur	1 To
Système d'exploitation	Windows 10 Professionnel (64 bits)
Prestataire	Belkahla Ahmed

Deuxième ordinateur :

Pc Portable	a
RAM	8 Go
Microprocesseur	
Disque dur	1 To
Système d'exploitation	Windows 10 (64 bits)
Prestataire	Cherni fraj

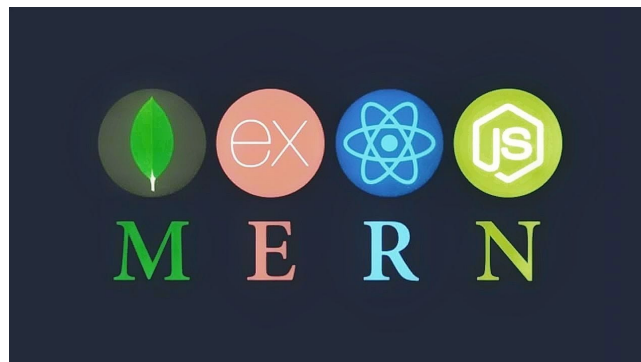
2.4.2 Environnement Technique**MERN**

FIGURE 2.4 – La pile technologique MERN

MERN est un acronyme qui représente quatre technologies open-source populaires utilisées pour créer des applications Web modernes. Il comprend MongoDB, Express.js, React et Node.js.

MongoDB est une base de données NoSQL flexible et évolutive qui stocke des données sous forme de documents JSON. Express.js est un framework pour créer des applications Web côté serveur en utilisant Node.js. React est une bibliothèque JavaScript pour créer des interfaces utilisateur interactives et dynamiques. Node.js est un environnement d'exécution JavaScript côté serveur construit sur le moteur JavaScript V8 de Google Chrome.

En utilisant ces quatre technologies ensemble, MERN permet aux développeurs de créer des applications Web modernes avec une pile complète JavaScript.

Visual Studio Code

Visual Studio Code est un éditeur de code source léger mais puissant qui s'exécute sur votre bureau et est disponible pour Windows, macOS et Linux. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et possède un riche écosystème d'extensions pour d'autres langages (tels que C++, C, Java, Python, PHP, Go) et des runtimes (tels que .NET et Unity).

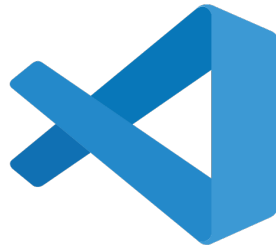
*Visual*

FIGURE 2.5 – Visual studio code

Draw.io

FIGURE 2.6 – Draw.io

draw.io pro est une application création de diagrammes compatible avec Google Drive et entièrement gratuite vous permettant de dessiner des organigrammes, des diagramme UML, des schémas réseaux, des Business Process Models, des Maquetage d'interface etc.. Avec draw.io les données sont stockées uniquement dans Google Drive, donc aucun tiers supplémentaire pour faire confiance à ses données.

PostMan

FIGURE 2.7 – PostMan

Postman est un outil de collaboration de développement d'API (Application Programming Interface) qui permet aux développeurs de tester, de documenter et de partager leurs API. Il offre une interface utilisateur conviviale pour envoyer des requêtes HTTP (GET, POST, PUT, DELETE, etc.) à une API et recevoir des réponses sous forme de JSON, XML ou tout autre

format de données. Postman est largement utilisé pour faciliter le développement d'applications qui utilisent des API et pour simplifier les tests et la documentation des API

Bootstrap



FIGURE 2.8 – Bootstrap

Bootstrap est une boîte à outils open source pour le développement avec HTML, CSS et JS. Prototypiez rapidement vos idées ou créez l'intégralité de votre application avec nos variables et mixins Sass, un système de grille réactif, de nombreux composants préconstruits et des plugins puissants construits sur jQuery.(Bootstrap · The most popular HTML, CSS, and JS library in the world., s. d.) .

Canva



FIGURE 2.9 – Canva

CANVA est un outil graphique qui permet aux utilisateurs de réaliser des logos, des invitations, des publications pour vos réseaux sociaux, des CV, des flyers et beaucoup d'autres choses. En choisissant CANVA, vous pouvez aussi bénéficier de divers outils en matière de vidéo, d'audio et d'animation

2.5 Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel.

Nous présentons par la figure ci-dessous une vue globale concernant le comportement fonctionnel en général du système.

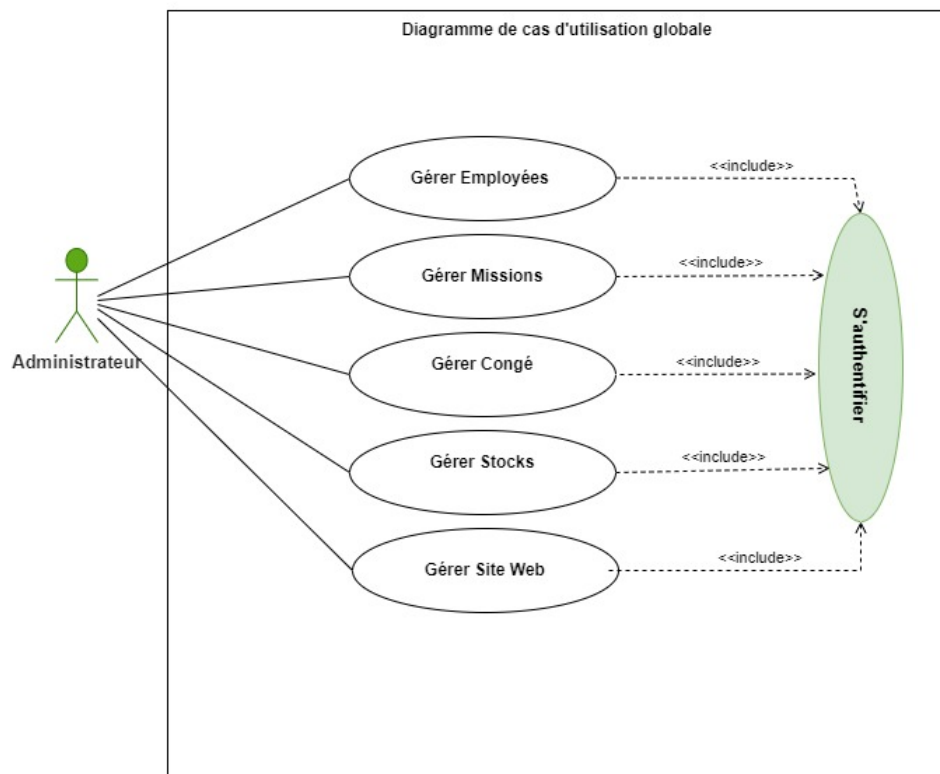


FIGURE 2.10 – Diagramme de cas d'utilisation globale

2.6 Diagramme de Classes

Le diagramme dans la figure 2.8 présente les interactions globales entre les classes dans notre application :

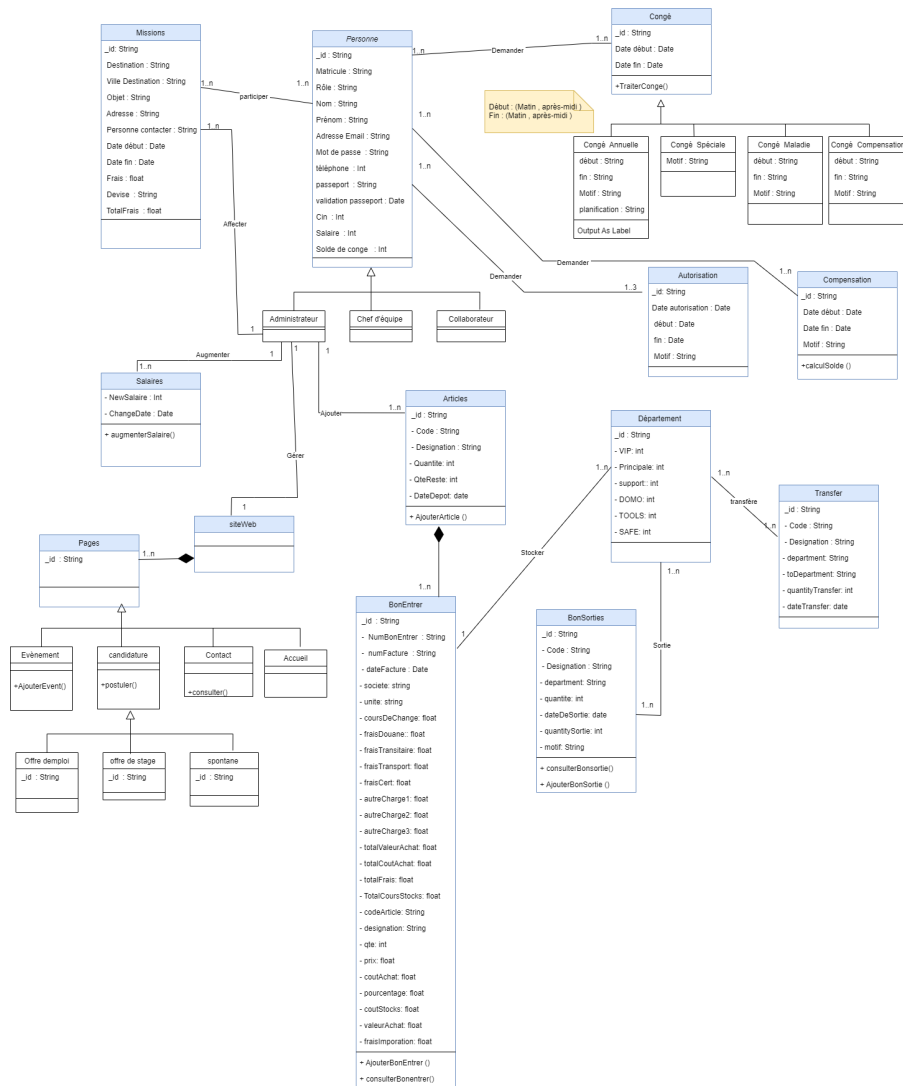


FIGURE 2.11 – Diagramme de classes

Conclusion

Dans ce chapitre, on a planifié notre travail, identifié les besoins fonctionnels et non fonctionnels de notre application, les rôles des utilisateurs, par la suite on a présenté le Backlog de notre système avec le diagramme de cas d'utilisation globale. Ainsi on a détaillé la phase de planification des sprints. Enfin on a choisi l'architecture de notre projet et l'environnement du travail.

Introduction

Ce chapitre fait l'objet d'une présentation du premier « release » du projet. En effet, chaque release, qui est l'ensemble d'itérations (Sprint), représente une vision distribuée de la période de la production du livrable. Ce premier Release comprend deux sprints :

- **Sprint 1** : Authentification et gestion des employées.
- **Sprint 2** : gestion des congés .

Le développement de chaque sprint passe par les étapes d'analyse de conception et de réalisation

3.1 Sprint 1 : Authentification est Gestion des employées

Le sprint est le cœur de Scrum. Il s'agit d'un bloc de temps durant lequel un incrément du produit sera réalisé.

Ce sprint a pour but de développer la première partie de notre projet qui est :

- L'authentification.
- La gestion des employées

3.1.1 Backlog du sprint

ID	User Story	ID tâche	Tâche	Estimation (jour)
1	Authentication	1.1	Interface Authentication	5 jours
		1.2	Développement d'un API d'authentification	
2	gestion des employées	2.1	Interface de gestion des employées	11 jours
		2.2	Ajouter, afficher, modifier, chercher et imprimer l'attestation d'un employé	
3	gestion d'accès	3.1	Interface de gestion d'accès	7 jours
		3.2	Gérer les droits d'accès utilisateur selon rôle	
4	Gestion de salaires	4.1	Interface de gestion de salaires	7 jours
		4.2	Augmenter les salaires et voir statistiques d'augmentation	

TABLE 3.1 – Backlog du Sprint 1

3.1.2 Analyse

Dans cette partie nous présentons la phase d'analyse par un diagramme des cas d'utilisation puis la description textuelle des plus importants d'entre eux.

3.1.2.1 Diagramme de cas d'utilisation Sprint 1

Les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Ils identifient également les interactions entre le système et ses acteurs.

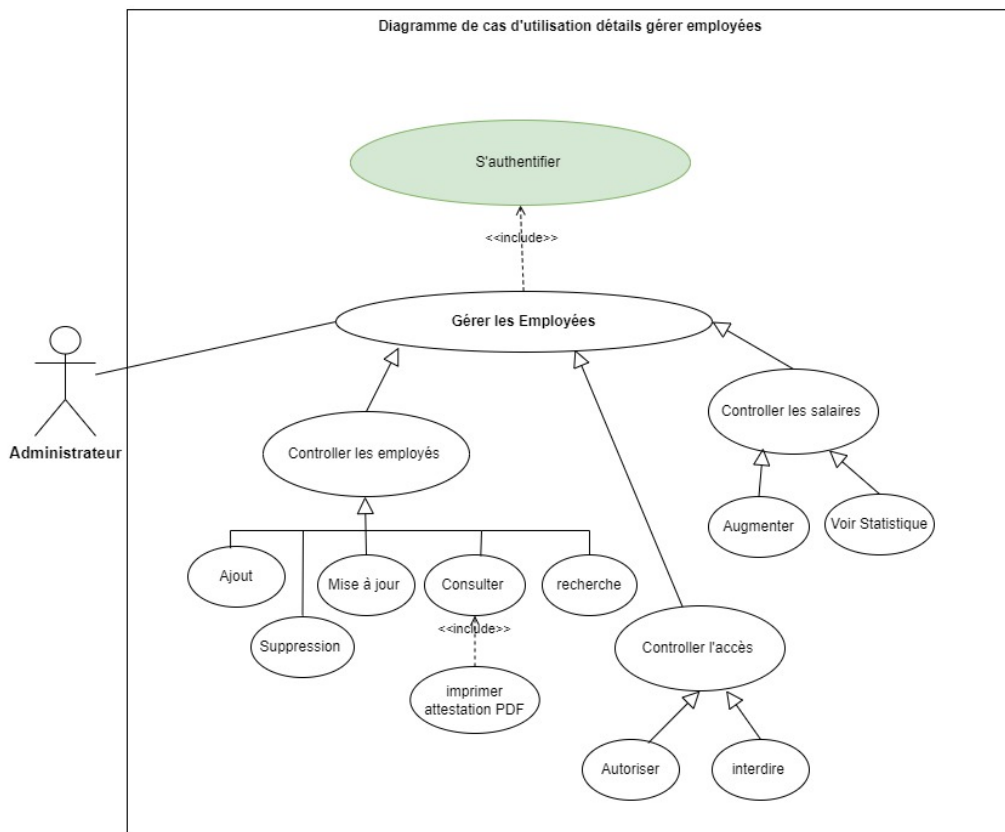


FIGURE 3.1 – Diagramme de cas d'utilisation du Sprint 1

3.1.2.2 Description textuelle

L'objectif de cette activité est de décrire textuellement, pour chaque Sprint, les scénarios des cas d'utilisation. Il faut indiquer comment ce scénario démarre, comment il se termine et les interactions de l'utilisateur avec l'application.

- Description textuelle du cas d'utilisation « Authentication »

Cas d'utilisation	Authentication
Description	L'administrateur ou l'employé doit fournir des identifiants valides pour accéder au système.
Acteurs	Administrateur / Employé
Préconditions	Acteur authentifié.
Postconditions	L'acteur est authentifié et peut accéder aux fonctionnalités autorisées.
Déroulement nominal	<ol style="list-style-type: none">1. L'acteur demande l'authentification.2. Le système affiche le formulaire d'authentification contenant le login et le mot de passe.3. L'acteur remplit les champs nécessaires et valide.4. Le système vérifie les données saisies.5. Le système crée l'espace personnel de l'acteur et l'authentifie.6. L'acteur est redirigé vers la page d'accueil du système et peut accéder aux fonctionnalités autorisées.
Déroulement alternatif	<p>4-a. L'acteur saisit des données incorrectes.</p> <p>4-a-1. Le système affiche des messages d'erreurs.</p> <p>4-a-2. Reprise de l'étape 3 du scénario nominal.</p>

TABLE 3.2 – Cas d'utilisation : Authentication

- Description textuelle du cas d'utilisation « Gérer les Employés»

Cas d'utilisation	Controler les Employees
Acteurs	Administrateur
Préconditions	L'acteur est authentifié.
Description	La gestion des Employeeer consiste à : Ajouter un employer, modifier un employer existant ,consulter les details ,imprimer atestation du travail supprimer un employer.

TABLE 3.3 – Cas d'utilisation : Gérer les Employeeer

- Description textuelle du cas d'utilisation « Ajouter un Employer»

Cas d'utilisation	Ajouter un Employeur
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	employeur Ajoutée.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur click sur le button ajouter un employeur. 2. Le système affiche le formulaire d'ajout. 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée le nouveau employeur.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 3.4 – Cas d'utilisation : Ajouter un Employée

- Description textuelle du cas d'utilisation « Modifier un Employeur »

Cas d'utilisation	Modifier un Employée
Acteurs	Administrateur
Préconditions	Administrateur authentifié et méthode déjà créée.
Postconditions	méthode mise à jour .
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur demande la page « employées ». 2. Le système affiche la liste des employées . 3. L'administrateur choisi l'opération « Modifier » à un employé spécifique. 4. Le système affiche le formulaire « Modifier employeur » . 5. L'acteur change les données qu'il désire. 6. Le système affiche une notification de mise à jour .
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 5 du scénario nominal.

TABLE 3.5 – Cas d'utilisation : modifier un Employée

- Description textuelle du cas d'utilisation « Gérer des salaires »

Cas d'utilisation	Gestion des salaires
Description	Permet à l'administrateur de consulter les salaires des employés et d'effectuer des opérations de gestion des salaires, tel que l'augmentation des salaires existants.
Acteurs	Administrateur
Préconditions	L'administrateur est authentifié.
Postconditions	Les modifications de salaire sont enregistrées dans le système.
Déroulement nominal	<ol style="list-style-type: none">1. L'administrateur accède à la page de gestion des salaires.2. Le système affiche une liste des employés et de leurs salaires actuels.3. L'administrateur peut augmenter le salaire pour un employeur en remplissant les champs nécessaires et valide.4. Le système enregistre les modifications de salaire.
Déroulement alternatif	<ol style="list-style-type: none">3-a. L'administrateur saisit des données incorrectes.3-a-1. Le système affiche des messages d'erreurs.3-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 3.6 – Cas d'utilisation : Gestion des accès

- Description textuelle du cas d'utilisation « Gérer des accès »

Cas d'utilisation	Gérer les accès
Acteurs	Administrateur
Préconditions	L'acteur est authentifié.
Description	La gestion des accès consiste à : <ul style="list-style-type: none">- Autorisation ou interdiction d'accès : l'administrateur peut autoriser ou interdire l'accès à une page pour un rôle.

TABLE 3.7 – Cas d'utilisation : Gérer les accès

- Description textuelle du cas d'utilisation « Autoriser / interdire d'accès »

Cas d'utilisation	Autoriser / interdire d'accès
Description	Permet à l'administrateur d'autoriser ou interdire les accès des employés en modifiant leur permissions.
Acteurs	Administrateur
Préconditions	L'administrateur est authentifié.
Postconditions	Les modifications des accès sont enregistrées dans le système.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'administrateur accède à la page de gestion des accès. 2. Le système affiche le tableau contenant les pages et les permissions avec le rôle qui correspond. 3. L'administrateur choisit la case à cocher indiquant la page, permission, et le rôle choisit ensuite l'admin choisit entre interdire ou autoriser l'accès. 4. Le système enregistre les modifications d'accès.
Déroulement alternatif	<ol style="list-style-type: none"> 3-a. L'administrateur saisit des données incorrectes. 3-a-1. Le système affiche des messages d'erreurs. 3-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 3.8 – Cas d'utilisation : Gestion des salaires

3.1.3 Conception

La conception est la troisième activité dans un sprint, elle se traduit par le diagramme de séquence.

3.1.3.1 Organigramme de conception

L'organigramme ci-dessous montre la démarche avec laquelle nous avons travaillé notre conception :

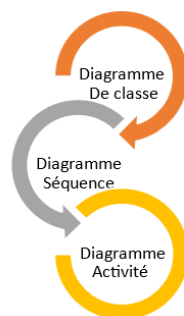


FIGURE 3.2 – Organigramme de conception

Diagramme de classes

La figure ci-dessous représente le diagramme de classes utilisé pour le développement du Sprint

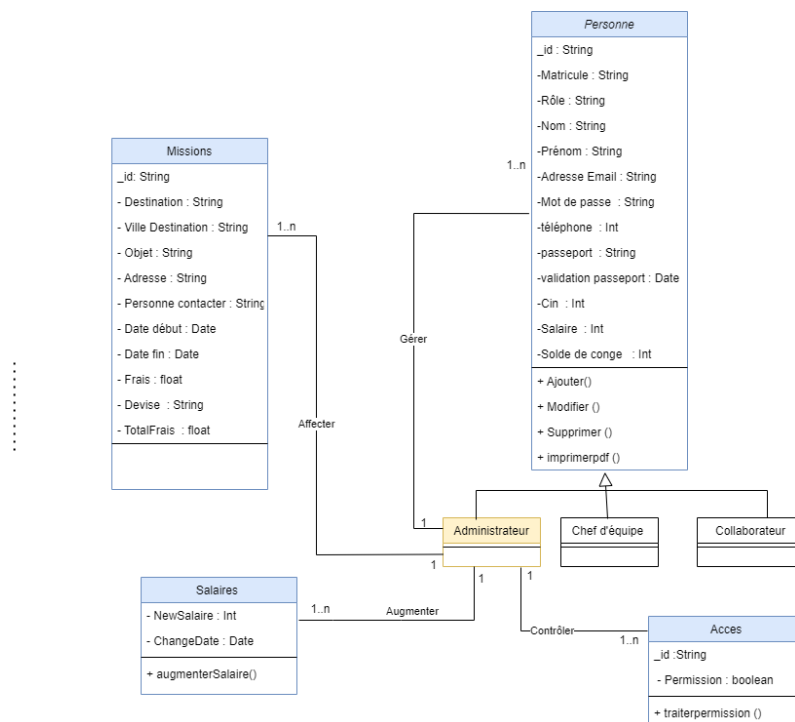


FIGURE 3.3 – Diagramme de classe du Sprint 1

Diagramme de séquence « Authentication »

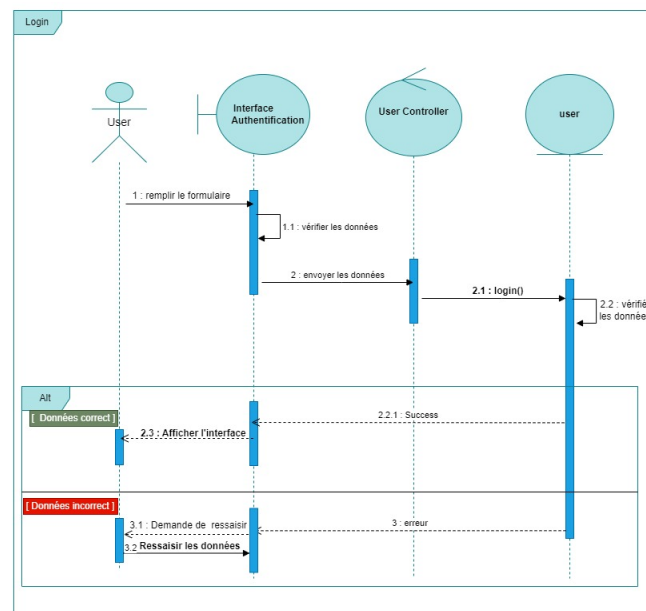


FIGURE 3.4 – Diagramme de Séquence "Authentication"

Diagramme de séquence « Ajouter un employée »

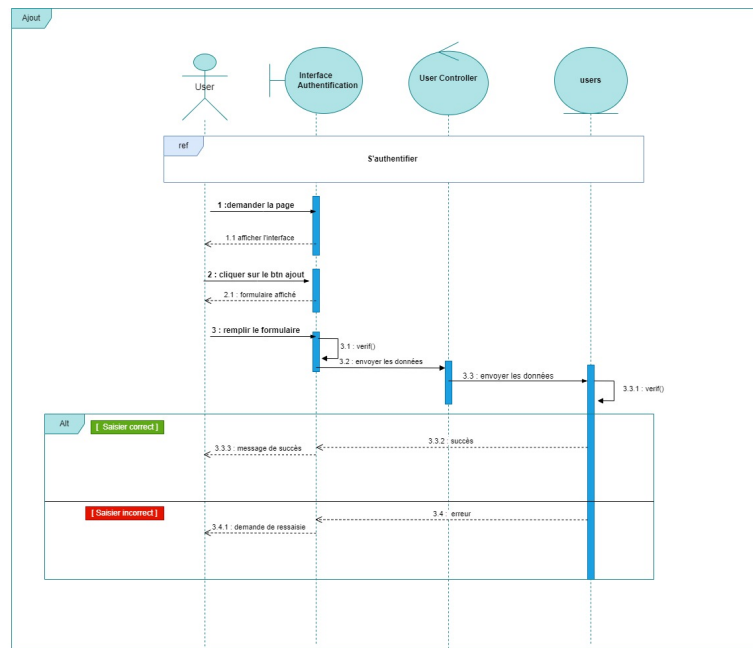


FIGURE 3.5 – Diagramme Séquence "Ajouter un employée"

Diagramme de séquence « Contrôler les accès »

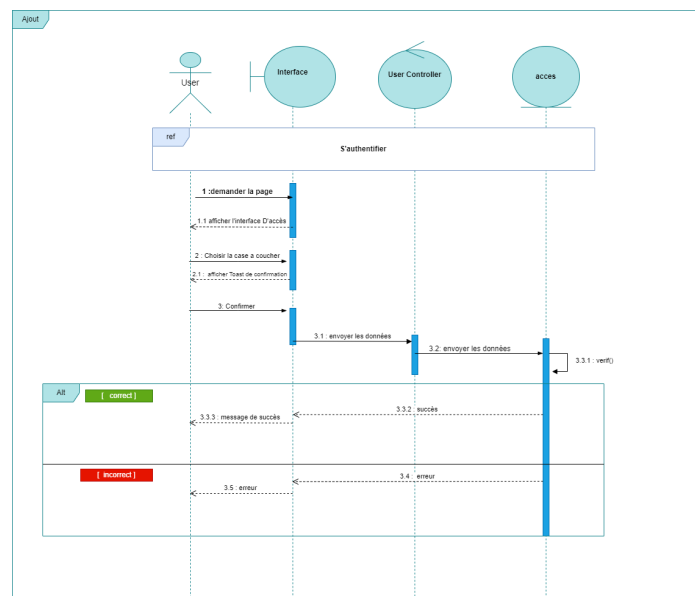


FIGURE 3.6 – Diagramme Séquence "Contrôler les accès"

Diagramme de séquence « Déconnexion »

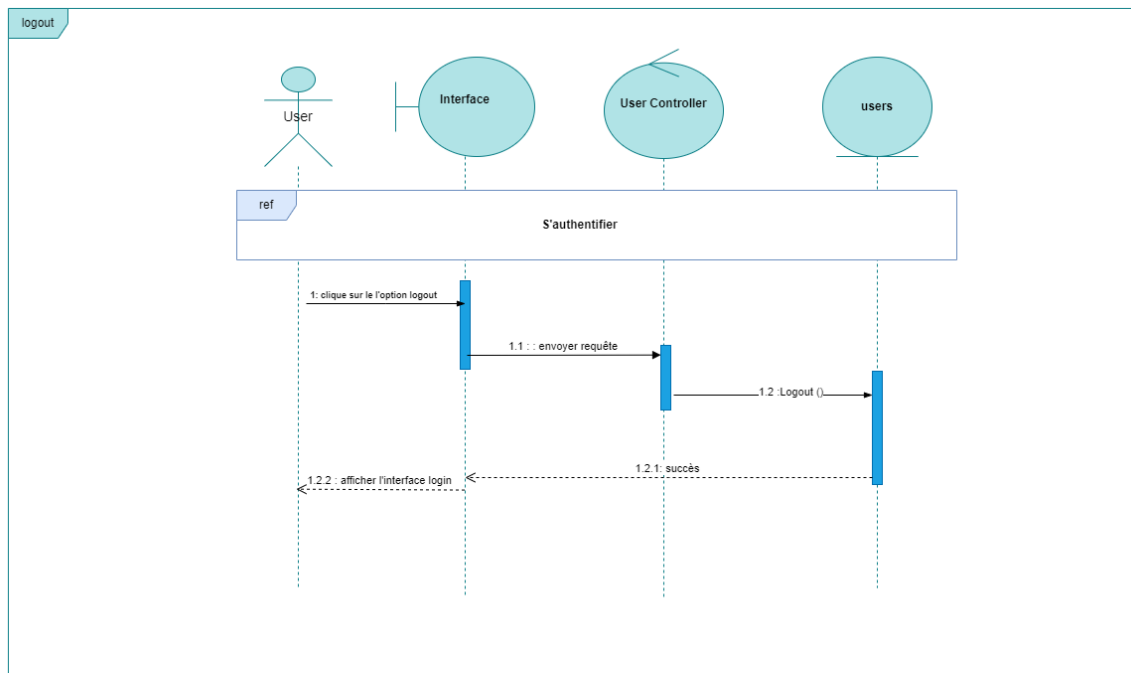


FIGURE 3.7 – Diagramme Séquence "Déconnexion "

Diagramme d'activité « Authentification »

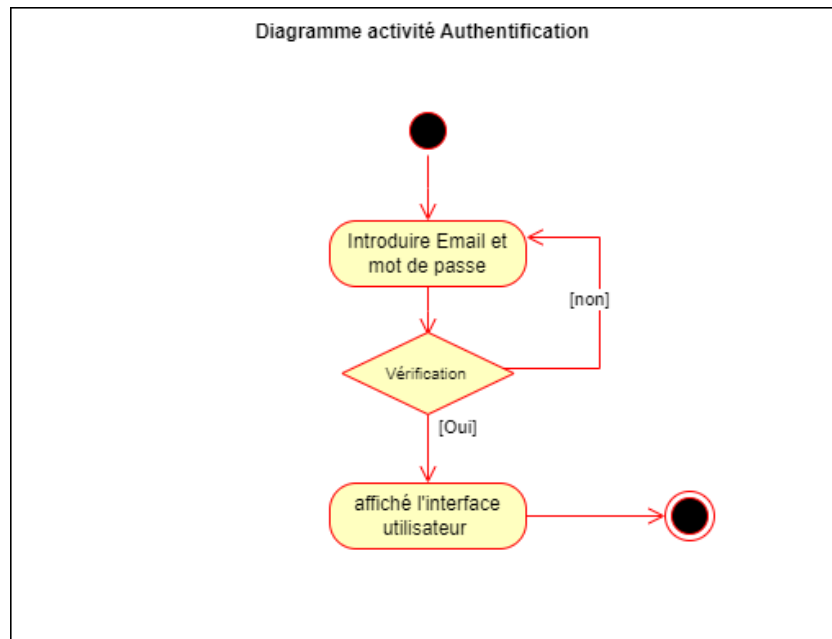


FIGURE 3.8 – Diagramme d'activité "Authentification "

Diagramme d'activité « Ajouter un employée »

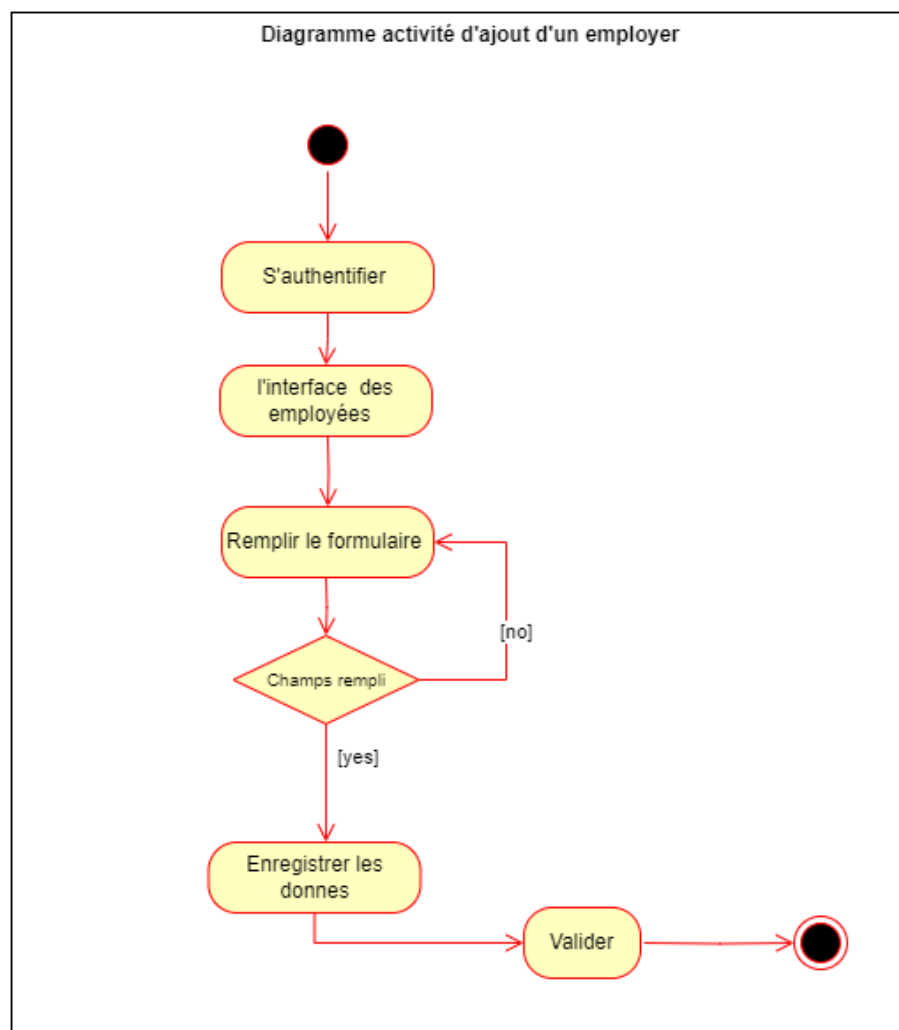


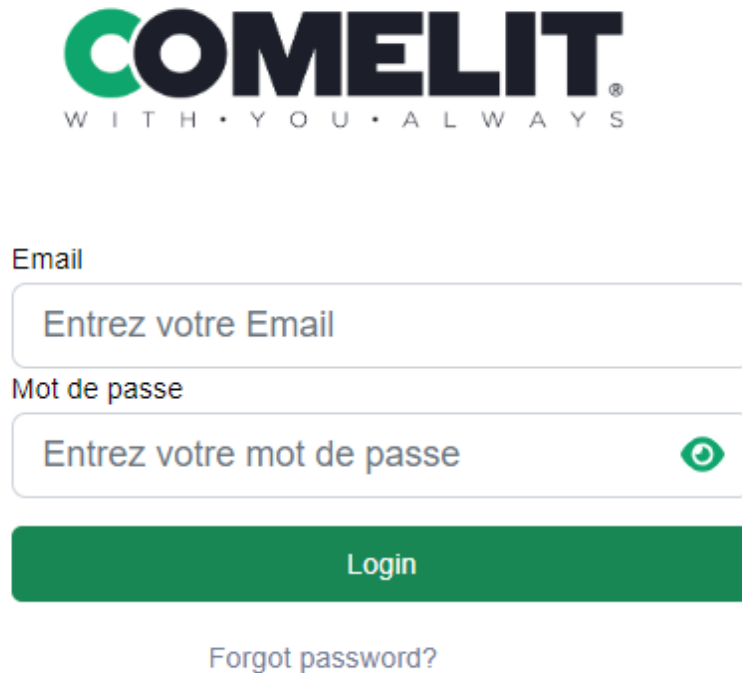
FIGURE 3.9 – Diagramme d'activité "Ajouter un employée "

3.1.4 Réalisation

Cette partie est consacrée à l'exposition du travail achevé à travers des captures d'écran de différentes interfaces développées pendant ce sprint.

Interface Authentication

L'interface Authentication permet à l'utilisateur d'accéder à son espace après une validation de son Email et son mot de passe.



The screenshot displays the authentication interface for COMELIT. At the top, the logo features the word "COMELIT" in a bold, dark blue font, with a green "C". Below it, the tagline "WITH • YOU • ALWAYS" is written in a smaller, grey font. The form consists of two input fields: the first is labeled "Email" and contains the placeholder text "Entrez votre Email"; the second is labeled "Mot de passe" and contains the placeholder text "Entrez votre mot de passe", accompanied by a green eye icon for password visibility. Below these fields is a prominent green "Login" button. At the bottom, there is a blue hyperlink labeled "Forgot password?".

FIGURE 3.10 – Interface " Authentication "

Espace d'accueil d'administrateurs

La figure ci-dessous représente l'espace d'accueil de notre plateforme après l'authentification



FIGURE 3.11 – Interface d'accueil d'administrateurs

Interface de gestion des employés

Un administrateur peut gérer les employés en effectuant les actions suivantes : ajouter, afficher, modifier et supprimer un employé, ainsi que l'impression d'une attestation de travail.

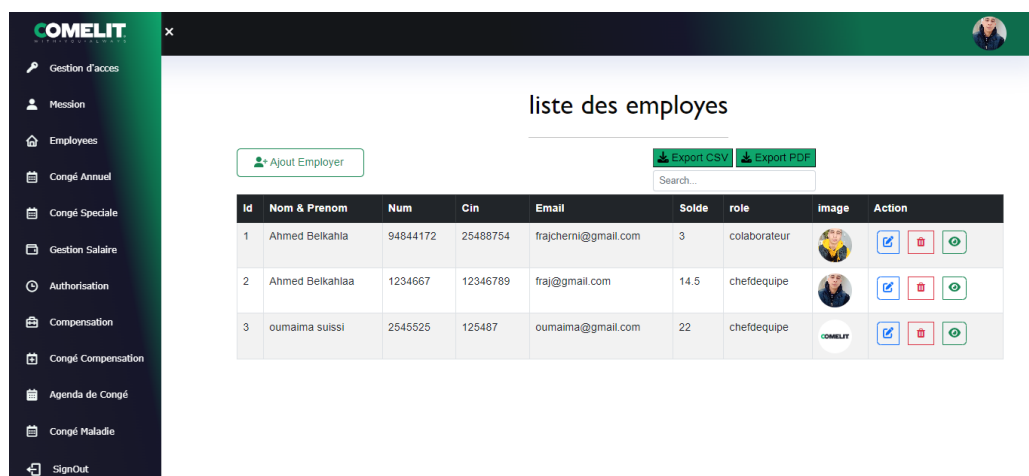


FIGURE 3.12 – Interface de gestion des employés

Interface de gestion des Salaires

Un administrateur peut gérer les salaires en effectuant les actions suivantes : augmenter un salaire et consulter les statistiques d'augmentation de salaire pour un employé.

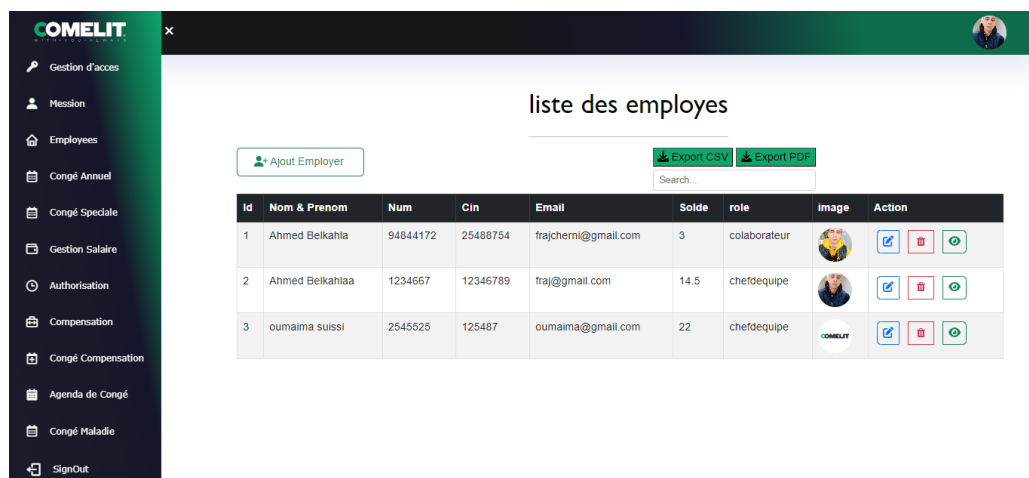


FIGURE 3.13 – Interface de gestion des Salaires

Interface de gestion des accès L'interface de gestion des accès permet à l'administrateur de gérer les autorisations d'accès pour les utilisateurs.

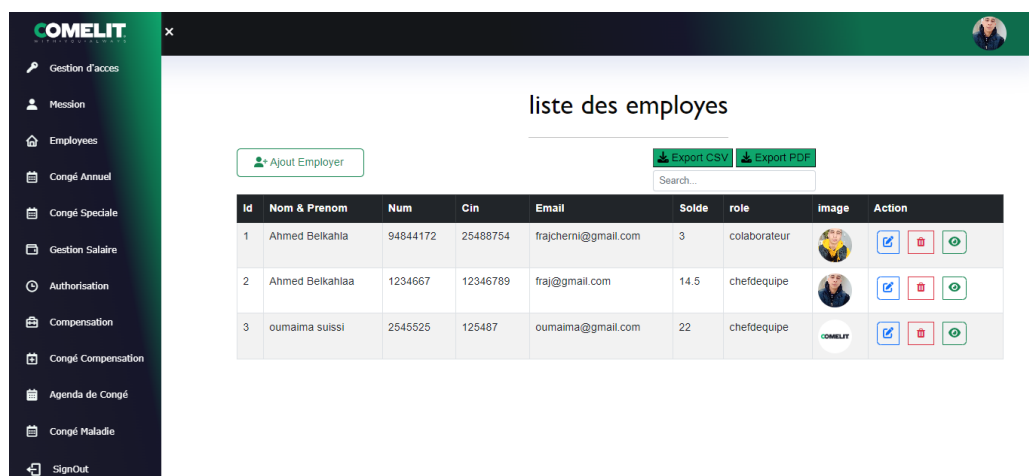


FIGURE 3.14 – Interface de gestion des accès

3.1.5 Test

Le test d'un produit logiciel est un processus consistant qui vise à garantir le bon fonctionnement du système à travers une comparaison des comportements attendus et des résultats obtenus. Avant la fin de chaque Sprint nous avons testé les fonctionnalités du module. Ensuite nous avons validé toutes les fonctionnalités avec le Product Owner. Pour ce fait, nous avons élaboré dans le tableau un ensemble de cas de scénario de tests relatif au sprint

Cas de Test	Demarche	Comportement Attendu	Résultat
Test d'authentification	S'authentifier	Authentification réussie	Conforme
Test de gestion des employees	Ajout, modification, suppression , impression et recherche d'un employeur	Gestion complète fonctionnelle	Conforme
Test de gestion des Salaires	Consulter et augmentation de salaire	Gestion complète fonctionnelle	Conforme
Test de gestion de accès	Consulter et Modification de permission	Gestion complète fonctionnelle	Conforme

TABLE 3.9 – Test du première Sprint

3.2 Sprint 2 « Gestion des Congés »

Ce sprint a pour but de développer les tache suivante :

- Gestion de congé annuelle
- Gestion de congé Spéciale
- Gestion de congé Maladie
- Gestion de congé de Compensation
- Gestion de Compensation
- Gestion d'autorisation

3.2.1 Backlog du Sprint 2

ID	User Story	ID tâche	Tâche	Estimation (jour)
1	Gestion des congé annuelle	1.1	interface de Gestion de congé annuelle	1 jour
		1.2	Demande de conge , liste de congé et gérer les congé (accepter ou refuser)	5 jour
2	Gestion des congé Spéciale	2.1	interface de Gestion de congé spéciale	1 jour
		2.2	Demande de conge , liste de congé et gérer les congé (accepter ou refuser)	5 jour
3	Gestion des congé Maladie	3.1	interface de Gestion de congé Maladie	1 jour
		3.2	Demande de conge , liste de congé et gérer les congé (accepter ou refuser)	5 jour
4	Gestion des congé de Compensation	4.1	interface de Gestion de congé Compensation	1 jour
		4.2	Demande de conge , liste de congé et gérer les congé (accepter ou refuser)	5 jour
5	Gestion de Compensation	5.1	interface de Gestion de Compensation	1 jour
		5.2	Demande de compensation , liste de comensation et gérer les compensation (accepter ou refuser)	2 jour
6	Gestion d'autorisation	6.1	interface de Gestion d'autorisation	1 jour
		6.2	Demande de d'autorisation , liste d'autorisation et gérer l'autorisations (accepter ou refuser)	2 jour

TABLE 3.10 – Backlog du Sprint 2

3.2.2 Analyse

Dans cette partie, nous présentons le diagramme de cas d'utilisation et la description textuelle des plus importants cas d'utilisation.

3.2.2.1 Diagramme du cas d'utilisation « Sprint 2 »

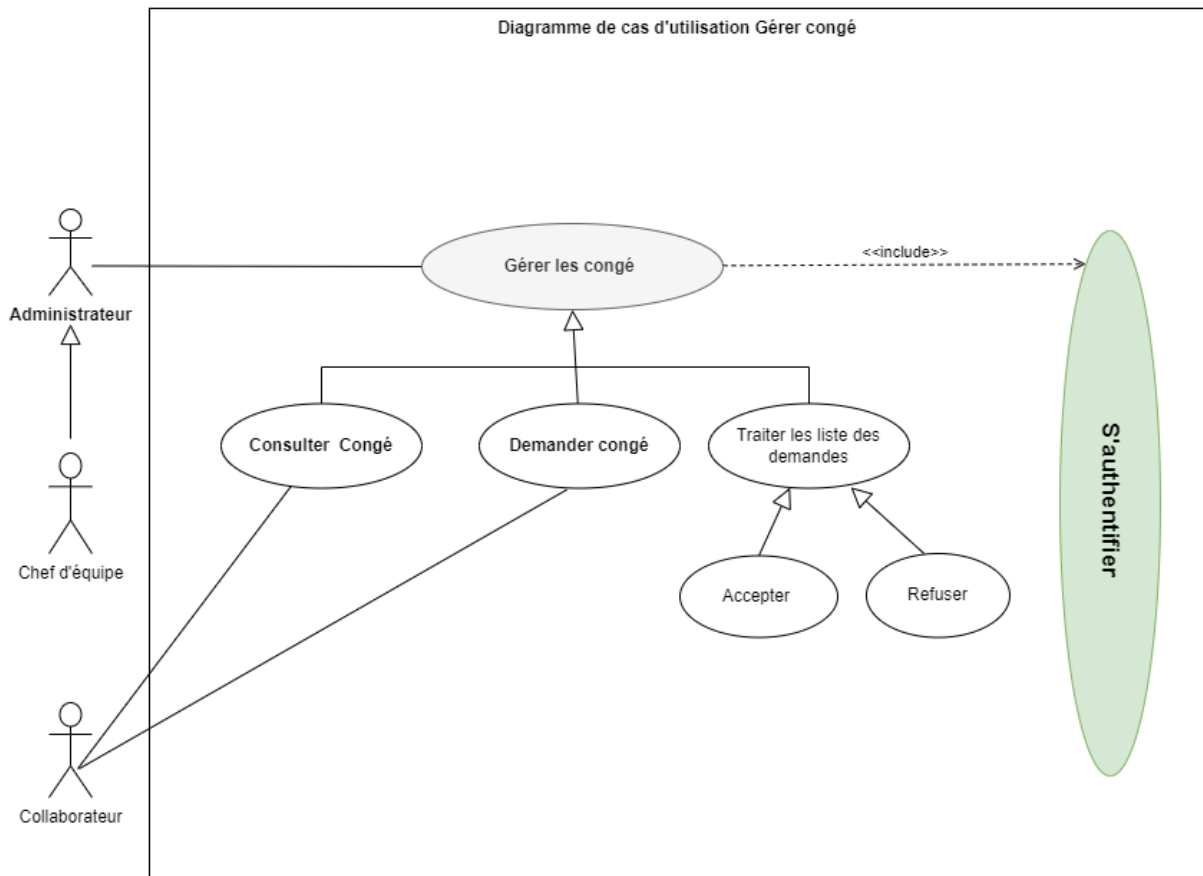


FIGURE 3.15 – Diagramme du cas d'utilisation « Sprint 2 »

3.2.2.2 Description textuelle :

- Description textuelle du cas d'utilisation « Gestion des congés »

Cas d'utilisation	Gestion des congés
Acteurs	Administrateur/ Chef d'équipe
Préconditions	L'acteur est authentifié.
Description	La gestion des congés consiste à : Demander un congé. consulter les Congé. Accepter ou refuser les demandes de congé. .

TABLE 3.11 – Cas d'utilisation : Gérer les congé

- Description textuelle du cas d'utilisation « Demander un congé»

Cas d'utilisation	Demander un congé
Acteurs	Administrateur / Employeur / Chef D'équipe
Préconditions	Acteur authentifié.
Postconditions	Demande enregistrée.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur demande la page de congé . 2. Le système affiche le formulaire d'ajout de demande . 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée le nouveau demande de congé puis il envoi un notification par email à les administrateurs.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. <ol style="list-style-type: none"> 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 3 du scénario nominal. 4-b. le solde de congé est insuffisant . <ol style="list-style-type: none"> 4-b-1. Le système affiche un message d'erreur.

TABLE 3.12 – Cas d'utilisation : Demander un congé

-Description textuelle du cas d'utilisation « Traiter les congés»

Cas d'utilisation	traiter les congés
Acteurs	Administrateur /Chef D'equipe
Préconditions	Acteur authentifié.
Postconditions	Congé Acceptée ou refusée .
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur demande la page de congé . 2. Le système affiche les liste des demandes de congé . 3. L'acteur choisit entre accepter ou refuser le demande . 4. Le système vérifie le traitement est affichera le message correspond (demande accepter ou demande refuser) puis il envoi un notification par email à le demandeur.

TABLE 3.13 – Cas d'utilisation : Demander un congé

3.2.3 Conception**3.2.3.1 Diagramme de classes**

La figure ci-dessous représente le diagramme de classes utilisé pour le développement du Sprint

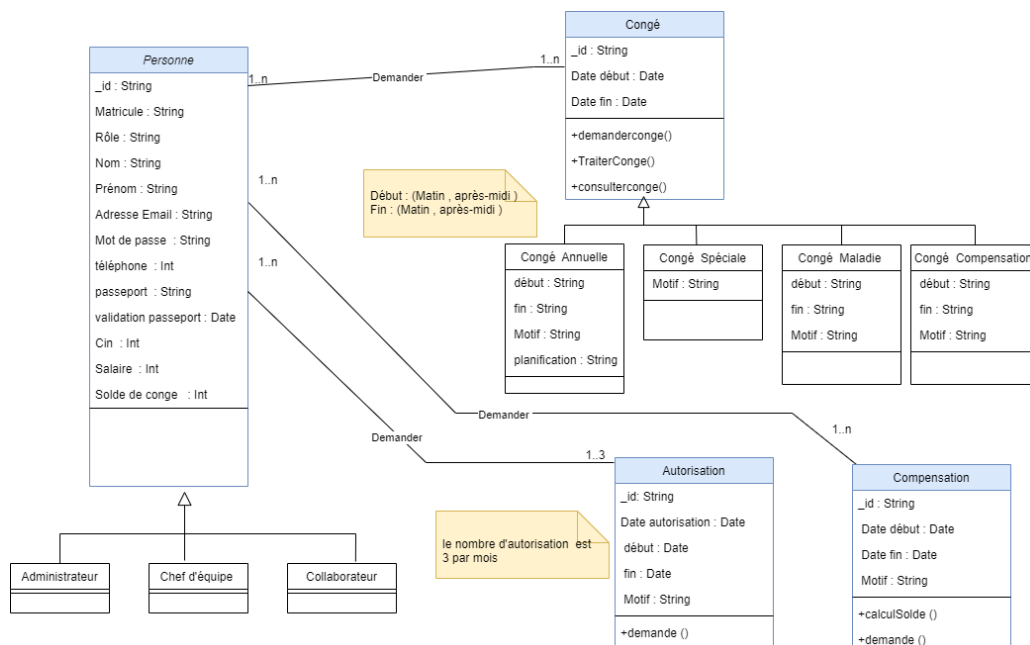


FIGURE 3.16 – diagramme de classe sprint 2

3.2.4 Diagramme de séquence

»

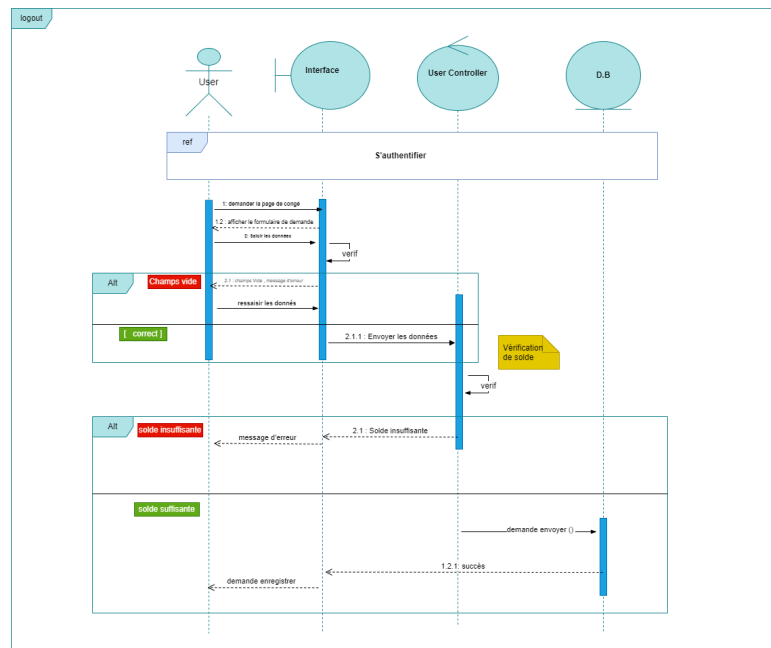


FIGURE 3.17 – Diagramme de séquence " demander un congé "

Diagramme de séquence du cas d'utilisation « Traiter un congé »

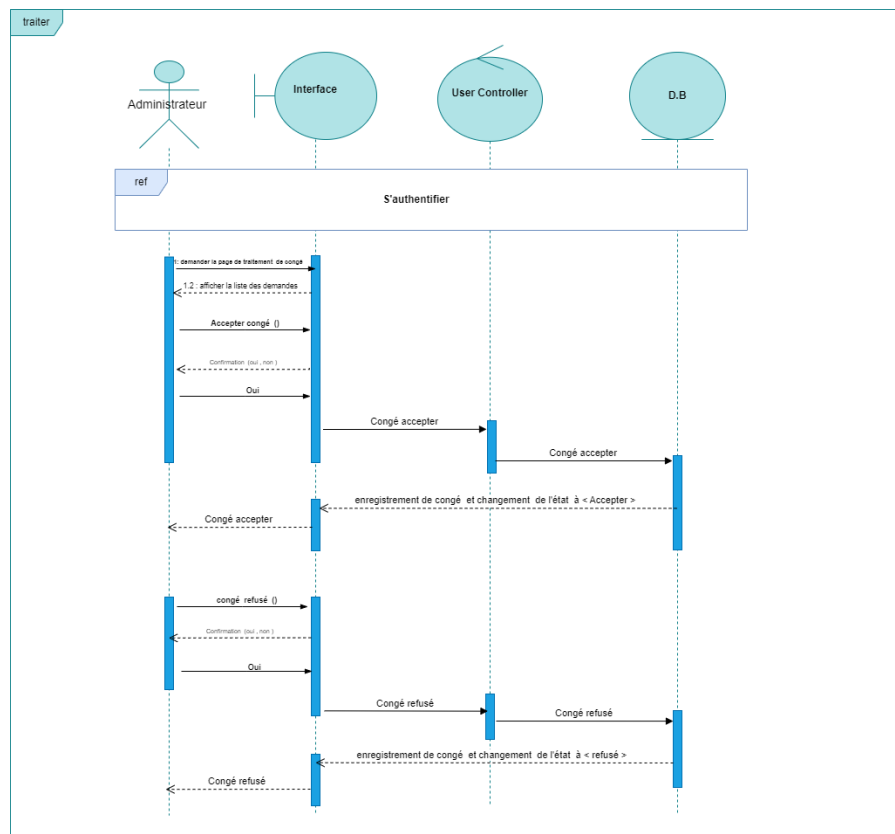


FIGURE 3.18 – Diagramme de séquence " traiter un congé "

Diagramme d'activité « demander un congé»

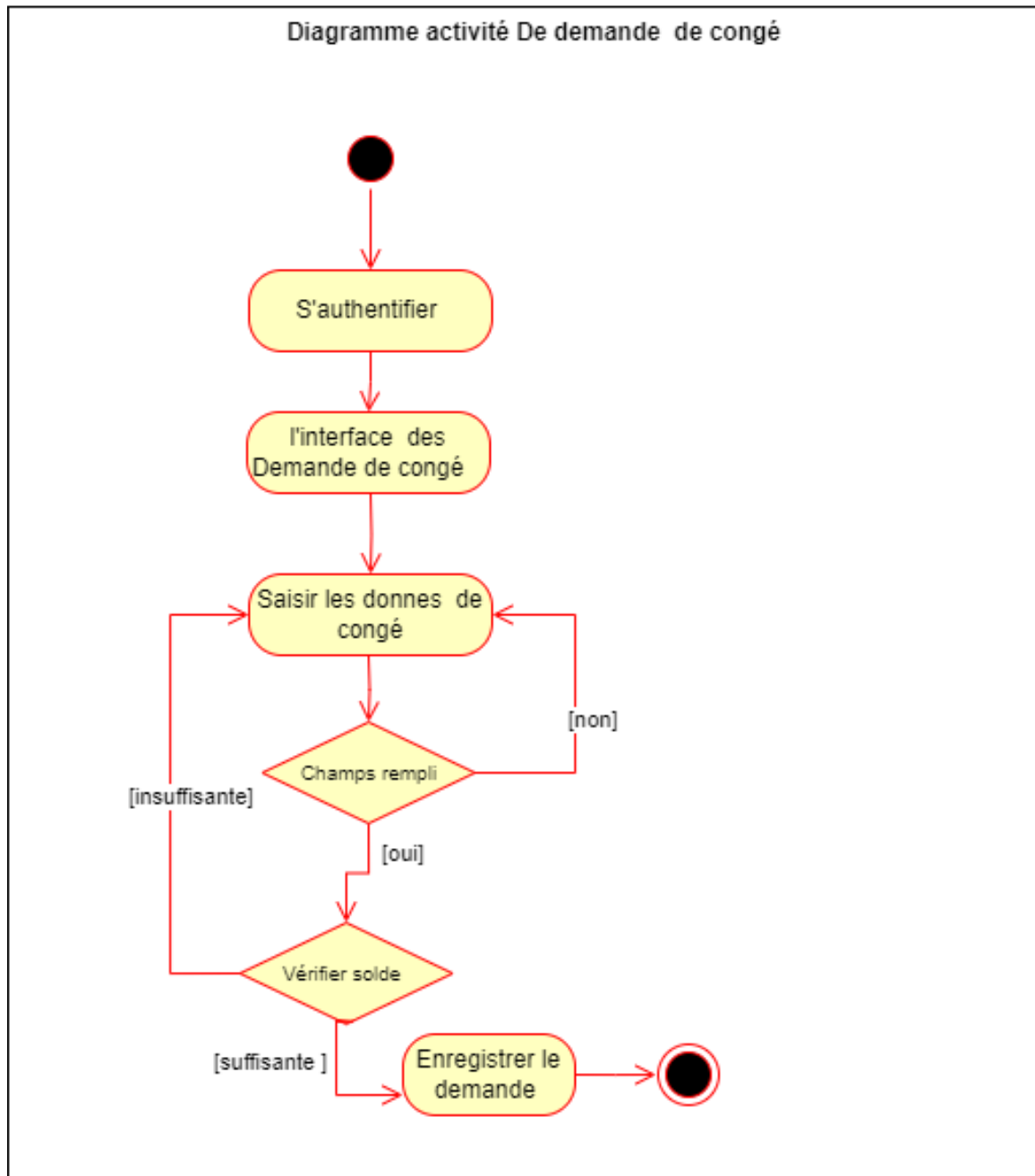


FIGURE 3.19 – Diagramme d'activité " demander un congé "

Diagramme d'activité « Accepter un congé»

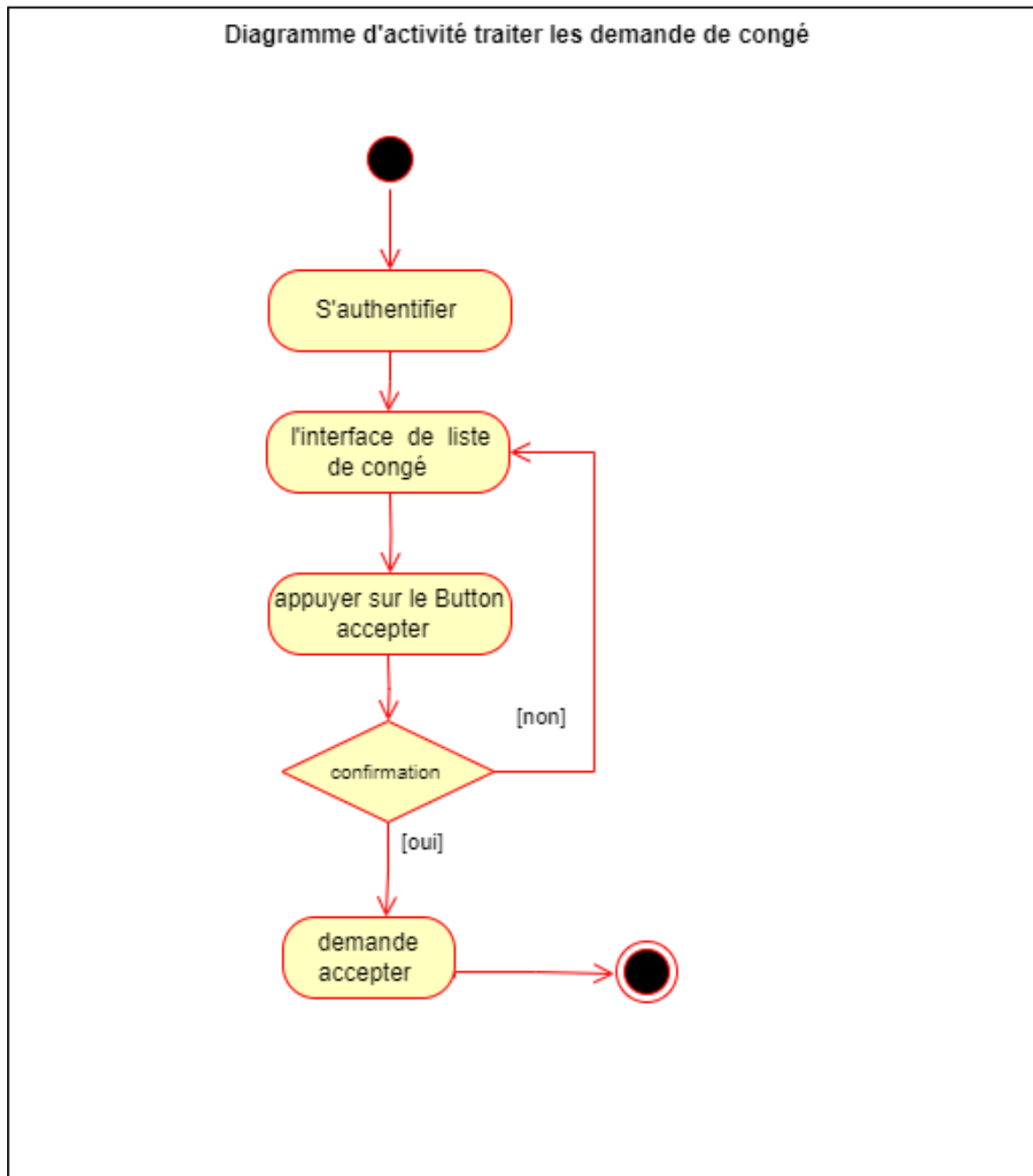


FIGURE 3.20 – Diagramme d'activité " Accepter un congé "

3.2.5 Réalisation

Interface " demander un congé "

L'interface de demande de congé permet à l'utilisateur de demander un congé annuelle

FIGURE 3.21 – Interface demander un conge

liste de congés

L'interface liste de congés permet à l'utilisateur de consulter la liste de congés

Date Debut	Date Fin	Motif	Status
2023-03-25T00:00:00.000Z	2023-03-26T00:00:00.000Z	conge special circoncision enfant 1 jour	refused
2023-04-07T00:00:00.000Z	2023-04-11T00:00:00.000Z	conge special mariage 3 jours	accepted

FIGURE 3.22 – Interface liste de congés

Traitement de congés

L'interface de traitement des congés permet à l'administrateur ou au chef d'équipe d'accepter ou de refuser une demande de congé.



FIGURE 3.23 – Interface de Traitement de congés

3.2.6 Test

Cas de Test	Demarche	Comportement Attendu	Résultat
Test de gestion de congé annuelle	demander un congé , traiter les demande de congés , consulter la liste de congés annuelle	Gestion complète fonctionnelle	Conforme
Test de gestion de congé speciale	demander un congé , traiter les demande de congés , consulter la liste de congés speciale	Gestion complète fonctionnelle	Conforme
Test de gestion de congé maladie	demander un congé , traiter les demande de congés , consulter la liste de congés maladie	Gestion complète fonctionnelle	Conforme
Test de gestion d'autorisation	demander une autorisation , traiter les demande d'autorisation , consulter la liste d'autorisation	Gestion complète fonctionnelle	Conforme
Test de gestion de compensation	demander une compensation , traiter les demande de compensation , consulter la liste de compensations	Gestion complète fonctionnelle	Conforme

TABLE 3.14 – Test du deuxième Sprint

3.2.7 Conclusion

Dans ce chapitre nous avons présenté le premier release qui comporte l' « Authentification », la « Gestion des Employées » ainsi que la « Gestion des congés ». Pour ce faire, nous avons passé par l'analyse, la conception, la réalisation et les tests. Après la validation de ce release nous avons pu passer au release suivant.

Introduction

Ce chapitre a pour objectif d'exposer le travail achevé durant la deuxième itération de notre Projet. Dans cette partie, nous nous intéressons à d'une tâche principale :

Ces sprint ont pour le but de développer les parties :

- **Sprint 1** : Gestion des Missions et aussi que la Gestion du Stock
- **Sprint 2** : Gestion du Site Web .

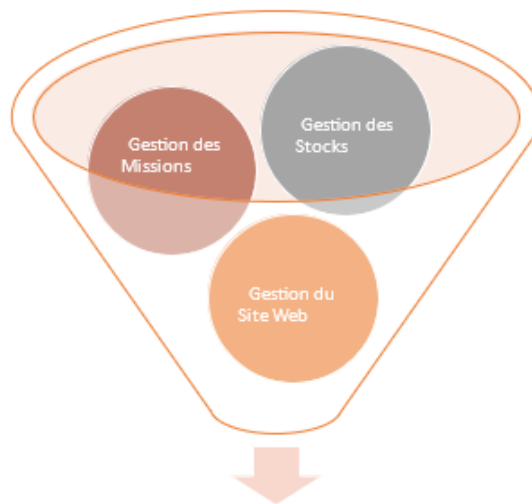


FIGURE 4.1 – Release 2

ID	User Story	ID tâche	Tâche	Estimation (jour)
1	gestion des Missions	1.1	Interface de gestion des Missions	11 jours
		1.2	Ajouter, afficher, modifier, chercher et imprimer l'ordre d'un Mission	
2	gestion de Stock	2.1	les Interfaces de gestion de stock	9 jours
		2.2	Gérer les Articles ,les Suivies des commandes , les bons d'entrés , les bons de transfères et les bons de sorties ainsi que létat de stocks	

TABLE 4.1 – Backlog du Sprint 1 de release 2

4.1 Sprint 1 : Gestion des Missions Et Gestion de Stock

4.1.1 Backlog du sprint

4.1.2 Analyse

Dans cette partie nous présentons la phase d'analyse par un diagramme des cas d'utilisation puis la description textuelle des plus importants d'entre eux.

Diagramme de cas d'utilisation « Sprint 1 »

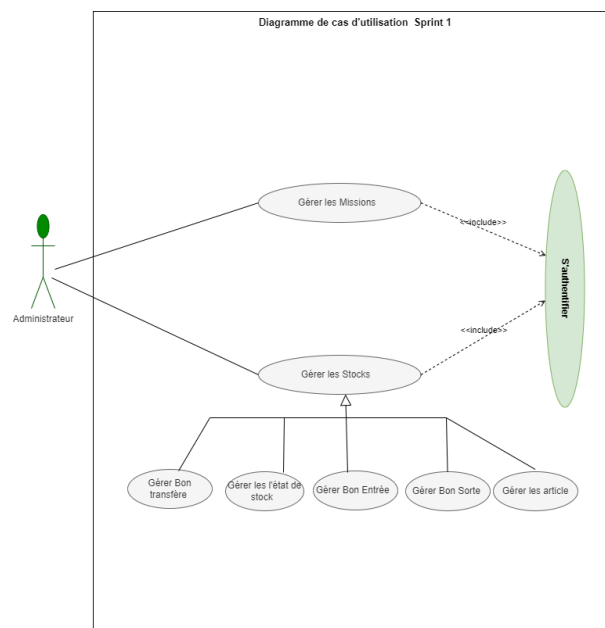


FIGURE 4.2 – Diagramme de cas d'utilisation "Gestion des missions "

Diagramme de cas d'utilisation « Gestion des Missions »

Diagramme de cas d'utilisation « Gestion des Stocks »

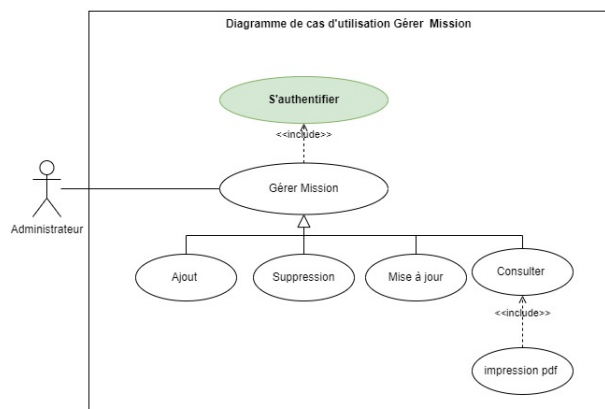


FIGURE 4.3 – Diagramme de cas d'utilisation "Gestion des missions "

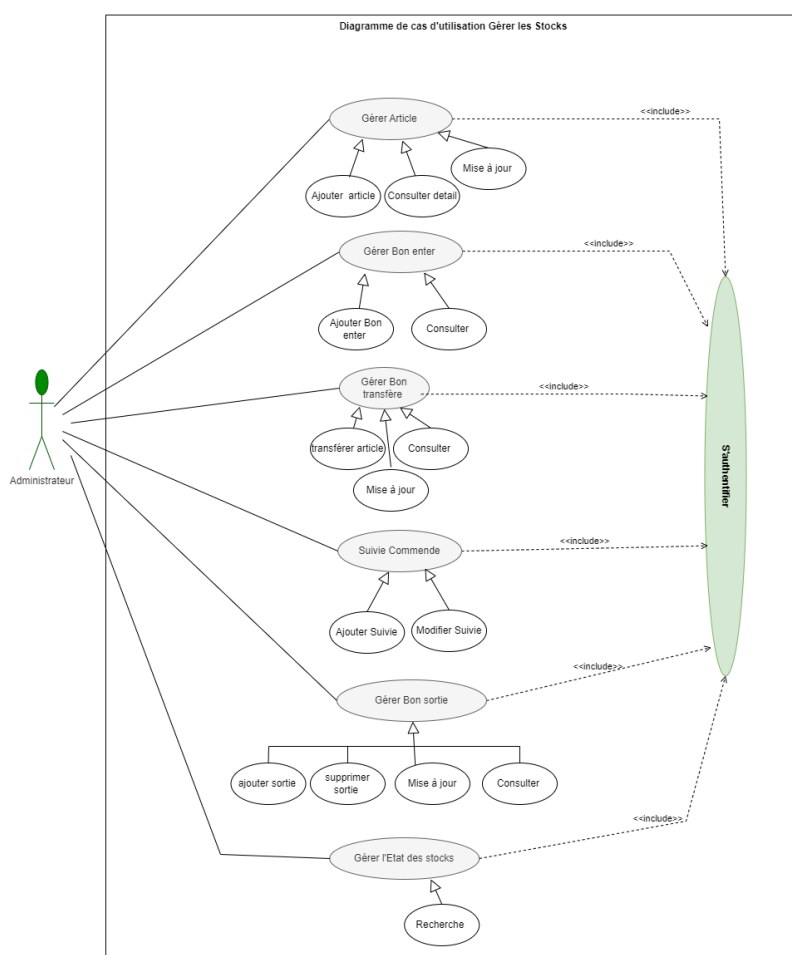


FIGURE 4.4 – Diagramme de cas d'utilisation "Gestion les Stocks "

4.1.3 Description textuelle

- Description textuelle du cas d'utilisation « Gérer les missions»

Cas d'utilisation	Gérer les missions
Acteurs	Administrateur
Préconditions	L'acteur est authentifié.
Description	- La gestion des missions consiste à : Ajouter une mission , Modification d'une mission , suppression d'une mission , consultation de la liste des missions et l'impression d'un ordre de mission.

TABLE 4.2 – Cas d'utilisation : Gérer les missions

- Description textuelle du cas d'utilisation « Ajouter une mission»

Cas d'utilisation	Ajouter une mission
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	mission Ajoutée.
Déroulement nominal	1. L'acteur click sur le bouton ajouter une mission. 2. Le système affiche le formulaire d'ajout d'une mission. 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée une nouvelle mission.
Déroulement alternatif	4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 4.3 – Cas d'utilisation : Ajouter une mission

- Description textuelle du cas d'utilisation « Modifier une mission »

Cas d'utilisation	Modifier une mission
Acteurs	Administrateur
Préconditions	Administrateur authentifié et méthode déjà créée.
Postconditions	méthode mise à jour .
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur demande la page « mission ». 2. Le système affiche la liste des missions . 3. L'administrateur choisi l'opération « Modifier » à une mission spécifique. 4. Le système affiche le formulaire « Modifier d'une mission » . 5.L'acteur change les données qu'il désire. 6.Le système affiche une notification de met à jour .
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 5 du scénario nominal.

TABLE 4.4 – Cas d'utilisation : modifier une mission

- Description textuelle du cas d'utilisation « supprimer une mission »

Cas d'utilisation	supprimer une mission
Acteurs	Administrateur
Préconditions	Administrateur authentifié et méthode déjà créée.
Postconditions	méthode de suppression.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur demande la page « mission ». 2. Le système affiche la liste des missions. 3. L'administrateur choisi l'opération « supprimer » à une mission spécifique. 4. Le système affiche une toast de confirmation « supprimer une mission (oui ou non) » . 5. L'administrateur confirme la suppression de la mission. 6.Le système affiche une toast de success après la suppression .
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'administrateur annulé la suppression de la mission. 4-a-1. le systeme affiche la liste des missions.

TABLE 4.5 – Cas d'utilisation : supprimer une mission

- Description textuelle du cas d'utilisation « Gérer les stocks »

Cas d'utilisation	Gérer les stocks
Acteurs	Administrateur
Préconditions	L'acteur est authentifié.
Description	<ul style="list-style-type: none"> - La gestion des stocks consiste à : - suivie : ajouter , modifier . - article : ajouter , modifier , consulter les details. - bon d'entrer : ajouter , modifier , supprimer. - bon de transfer : Ajouter , supprimer , modifier , supprimer . - bon de sortie : ajouter , supprimer , modifier - consulter l'etat d'un article

TABLE 4.6 – Cas d'utilisation : Gérer le stock

- Description textuelle du cas d'utilisation « Ajouter un Article »

Cas d'utilisation	Ajouter un Article
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	Article a été Ajoutée.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur click sur le button ajouter un article 2. Le système affiche le formulaire d'ajout d'une article. 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée un nouveau article. 6. Le système initialise la quantité de l'article à zéro.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreurs. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 4.7 – Cas d'utilisation : Ajouter un article

- Description textuelle du cas d'utilisation « Ajouter un bon de transfert »

Cas d'utilisation	Ajouter un bon de transfert
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	Bon de transfert a été ajoutée.
Déroulement nominal	1. L'acteur clique sur le bouton "Ajouter un bon de transfert". 2. Le système affiche le formulaire d'ajout d'un bon de transfert. 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée un nouveau bon de transfert. 6. Le système enregistre la quantité qui est transférée du département choisi par l'administrateur vers le département de destination.
Déroulement alternatif	4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreur. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 4.8 – Cas d'utilisation : Ajouter un bon de transfert

- Description textuelle du cas d'utilisation « supprimer un bon de transfert »

Cas d'utilisation	supprimer un bon de transfert
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	mission a été Ajoutée.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur click sur le button supprimer un bon de transfert 2. Le système affiche une toast de confirmation indiquant oui ou non 3. L'acteur confirme la suppression de la bon de transfert. 4. Le système vérifie les données. 5. Le système affiche une toast de success après la suppression. 6. Le système rétablit la quantité à l'article correspondant.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'administrateur annule la suppression. 4-a-1. le système affiche la liste des bons de transfert.

TABLE 4.9 – Cas d'utilisation : supprimer un bon de transfert

- Description textuelle du cas d'utilisation « **Ajouter un bon d'entrer** »

Cas d'utilisation	Ajouter bon d'entrer
Acteurs	Administrateur
Préconditions	Acteur authentifié.
Postconditions	Bon d'entrer a été ajouté.
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur clique sur le bouton "Ajouter un bon d'entrer". 2. Le système affiche le formulaire d'ajout d'un bon d'entrer. 3. L'acteur remplit tous les champs nécessaires et valide. 4. Le système vérifie les données saisies. 5. Le système crée un nouveau bon d'entrer. 6. Le système incrémente la quantité saisie à la quantité initiale de l'article correspondant.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche des messages d'erreur. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 4.10 – Cas d'utilisation : Ajouter un bon d'entrée

4.1.4 Conception

Diagramme de classes La figure ci-dessous représente le diagramme de classe utilisé pour le développement du Sprint

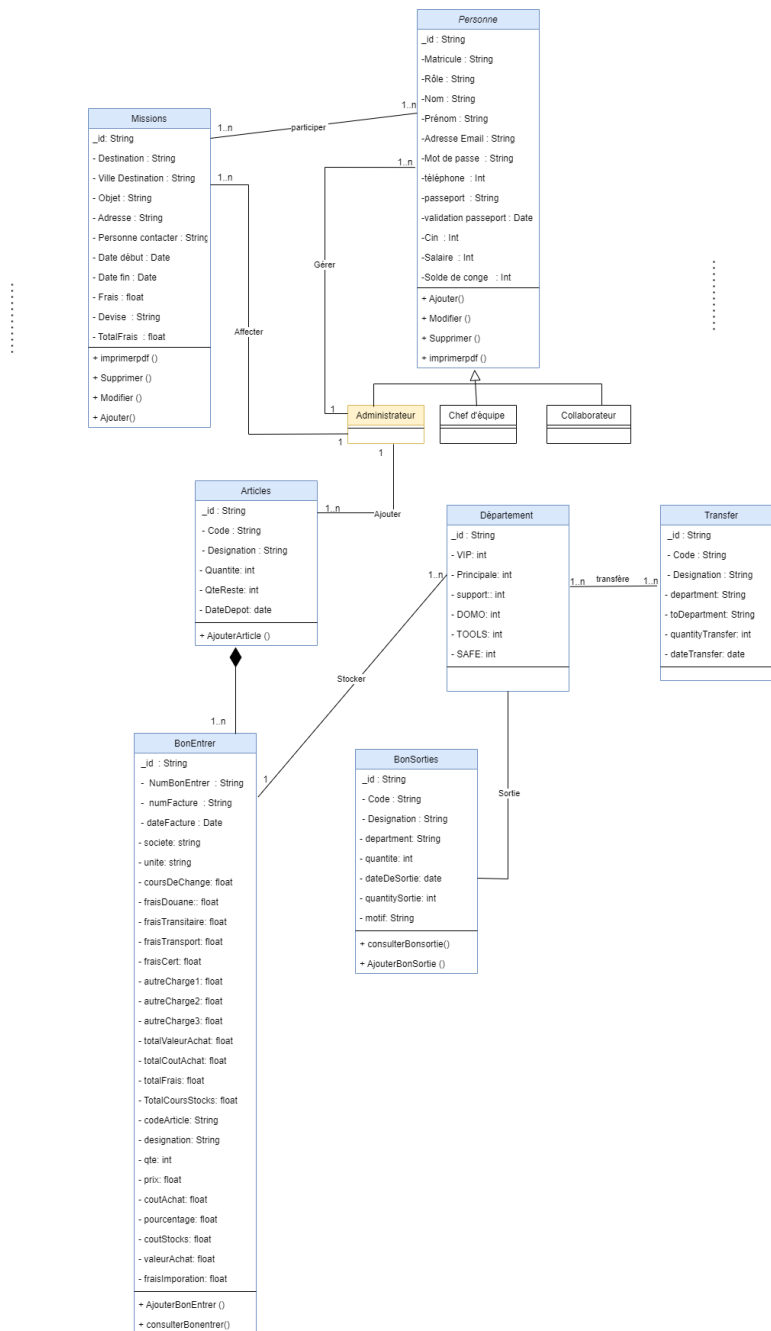


FIGURE 4.5 – Diagramme de classe sprint 1

Diagramme de séquence du cas d'utilisation «Ajouter une mission »

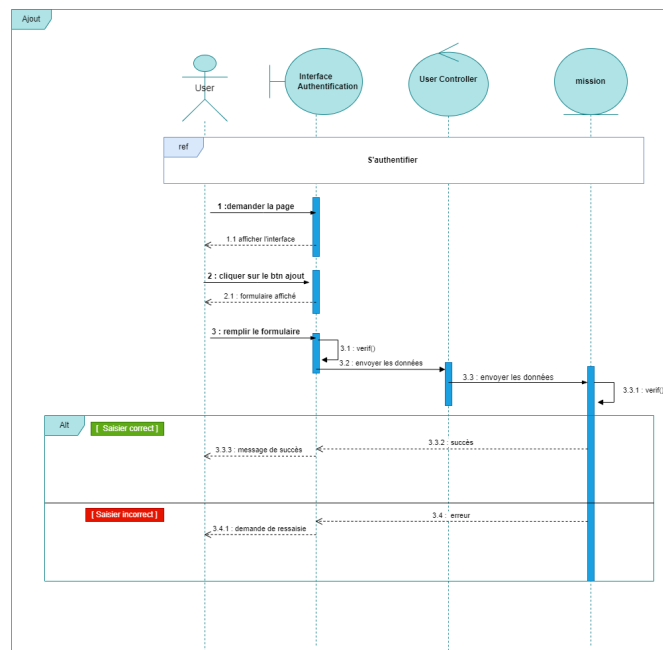


FIGURE 4.6 – Diagramme de séquence "Ajouter une mission "

Diagramme de séquence du cas d'utilisation «supprimer une mission »

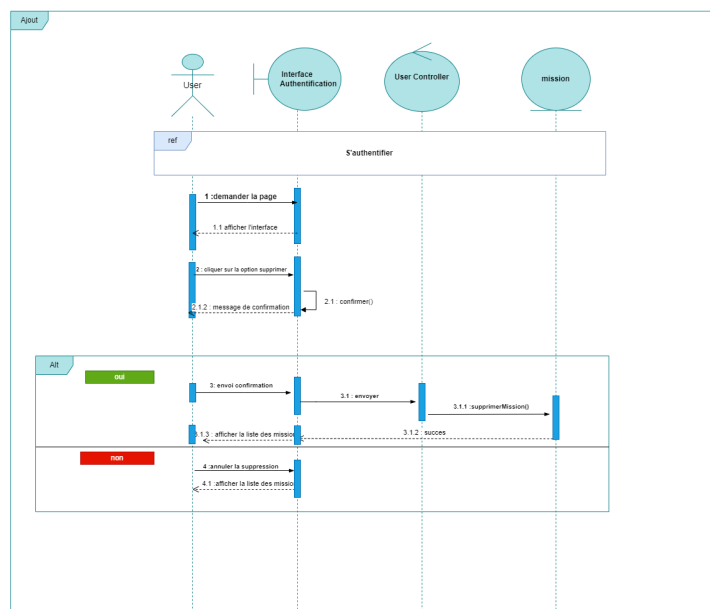


FIGURE 4.7 – Diagramme de séquence "supprimer une mission "

Diagramme de séquence du cas d'utilisation «imprimer une mission »

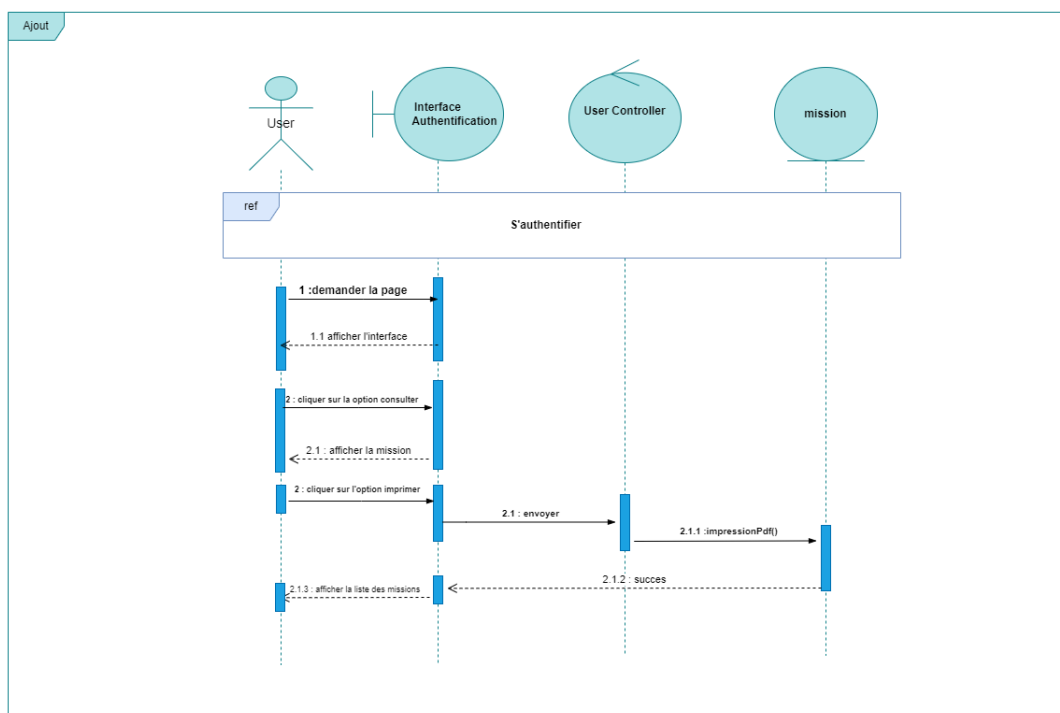


FIGURE 4.8 – Diagramme de séquence "impression d'une mission "

Diagramme de séquence du cas d'utilisation «Ajouter un Bon d' Entrée»

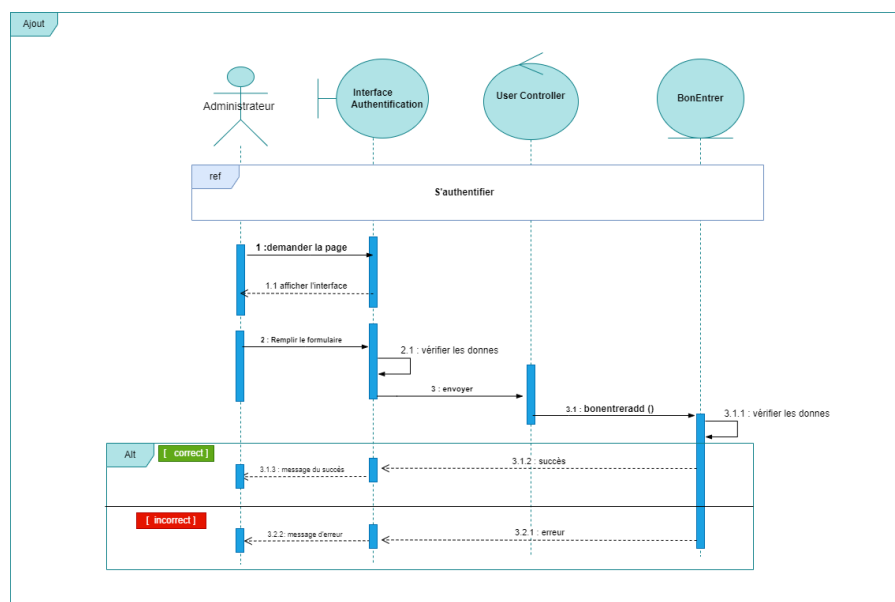


FIGURE 4.9 – Diagramme de séquence «Ajouter un Bon d' Entrée »

Diagramme de séquence du cas d'utilisation «Ajouter un Bon de Transfert »

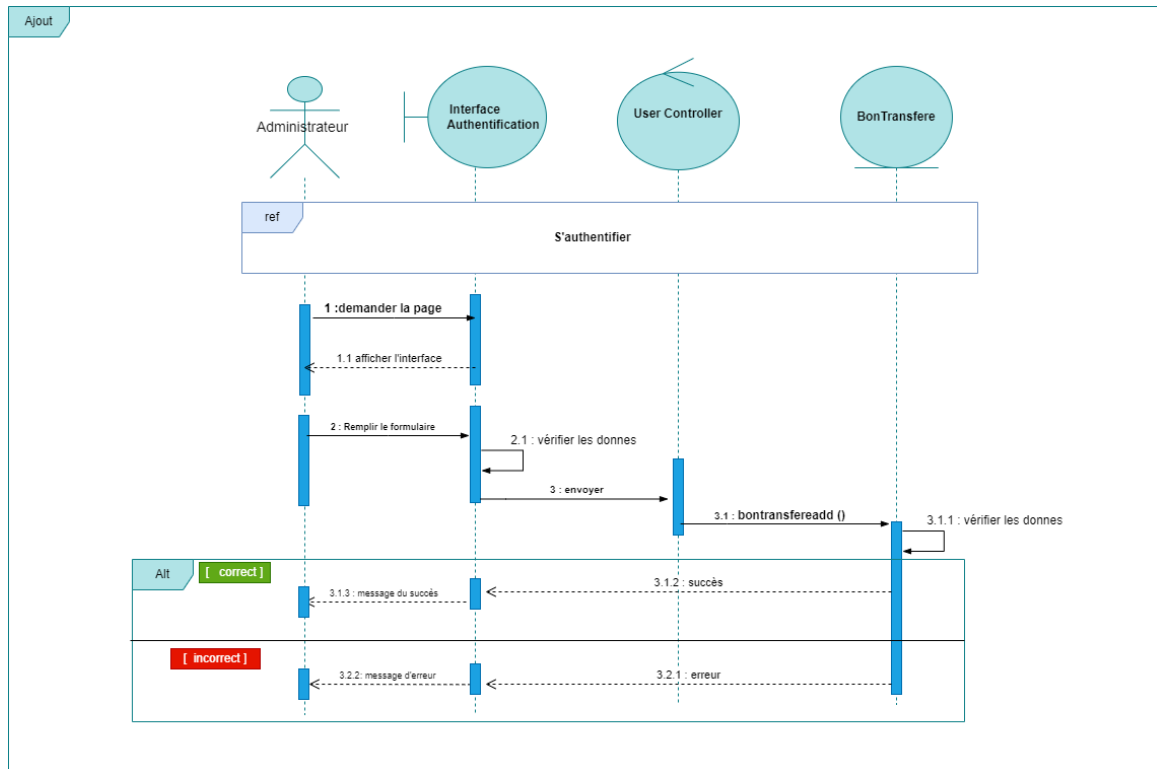


FIGURE 4.10 – Diagramme de séquence «Ajouter un Bon de Transfert »

Diagramme de séquence du cas d'utilisation «Modifier un bon de sortie »

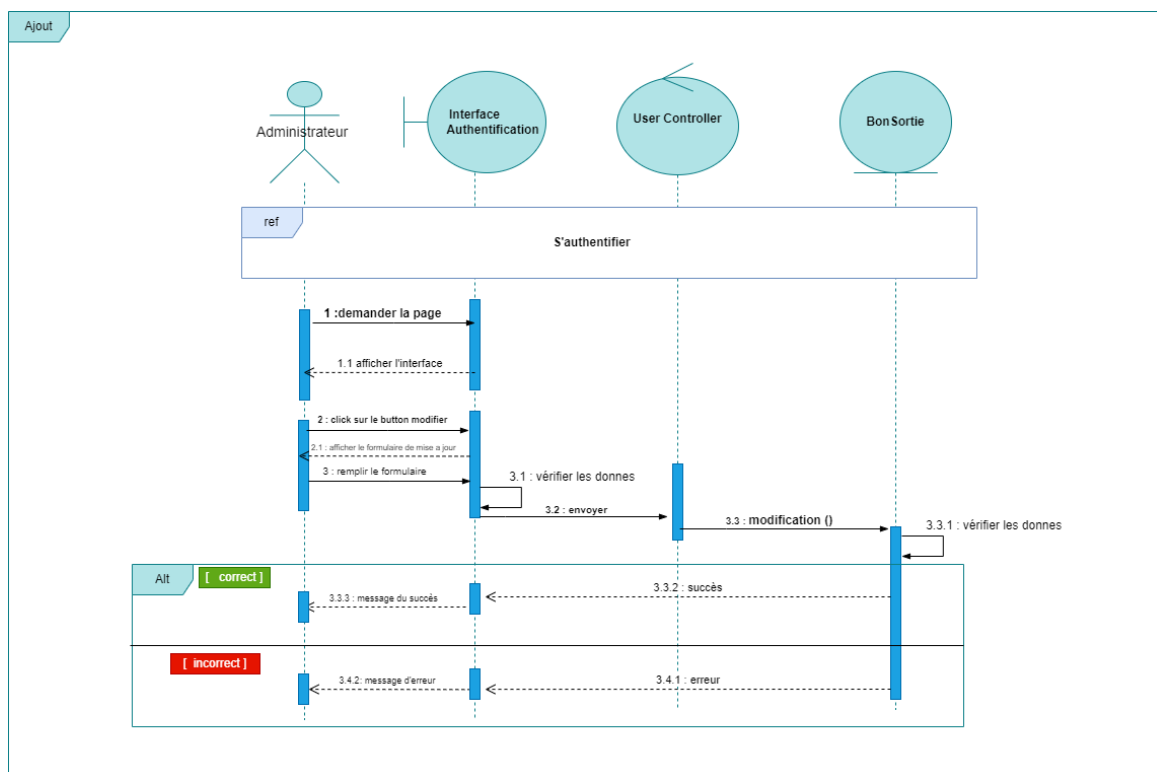


FIGURE 4.11 – Diagramme de séquence «Modifier un bon de sortie »

Diagramme d'activité «Modifier une mission»

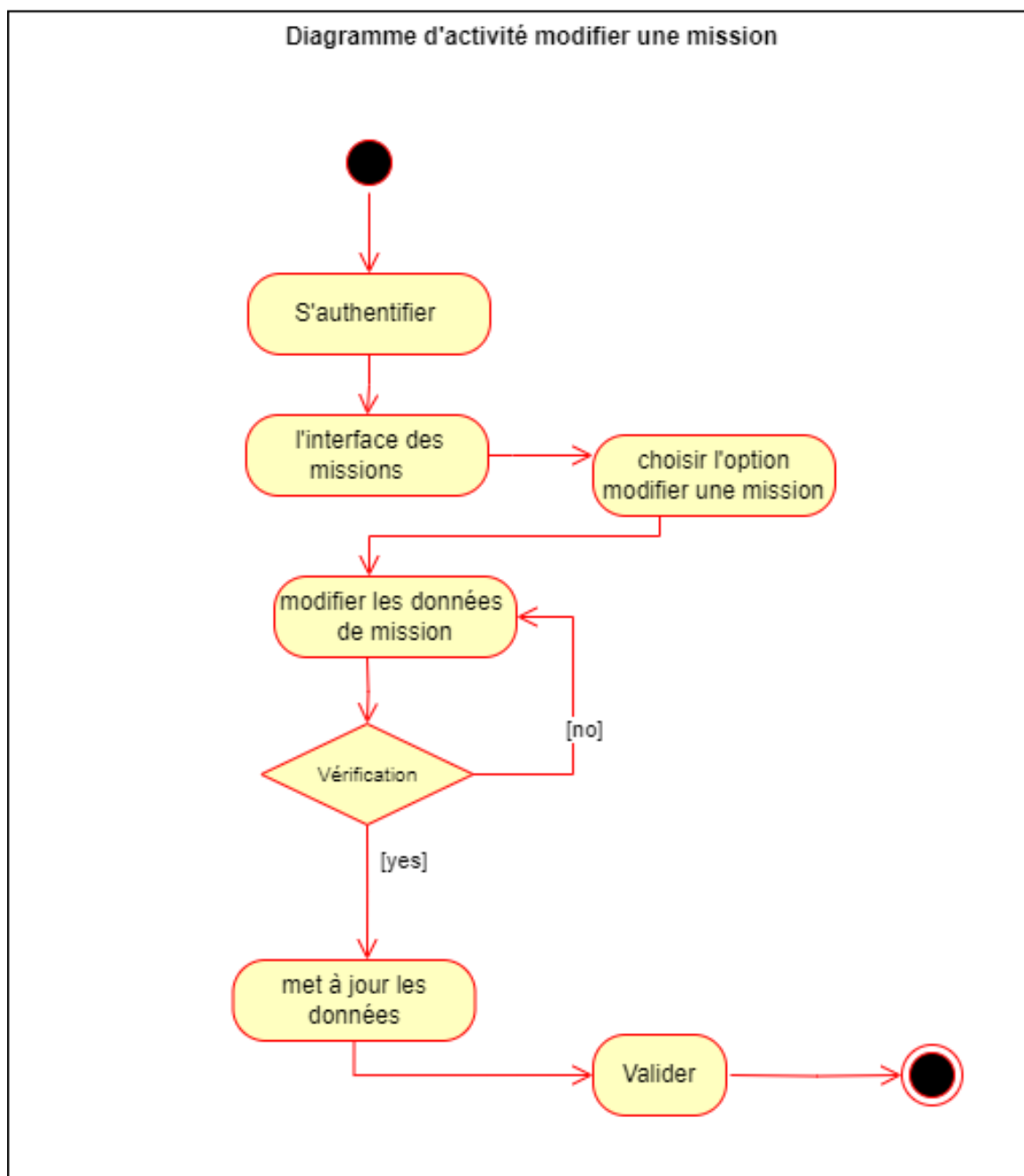


FIGURE 4.12 – Diagramme d'activité «Modifier une mission »

Diagramme d'activité «Modifier un Bon d'entrée »

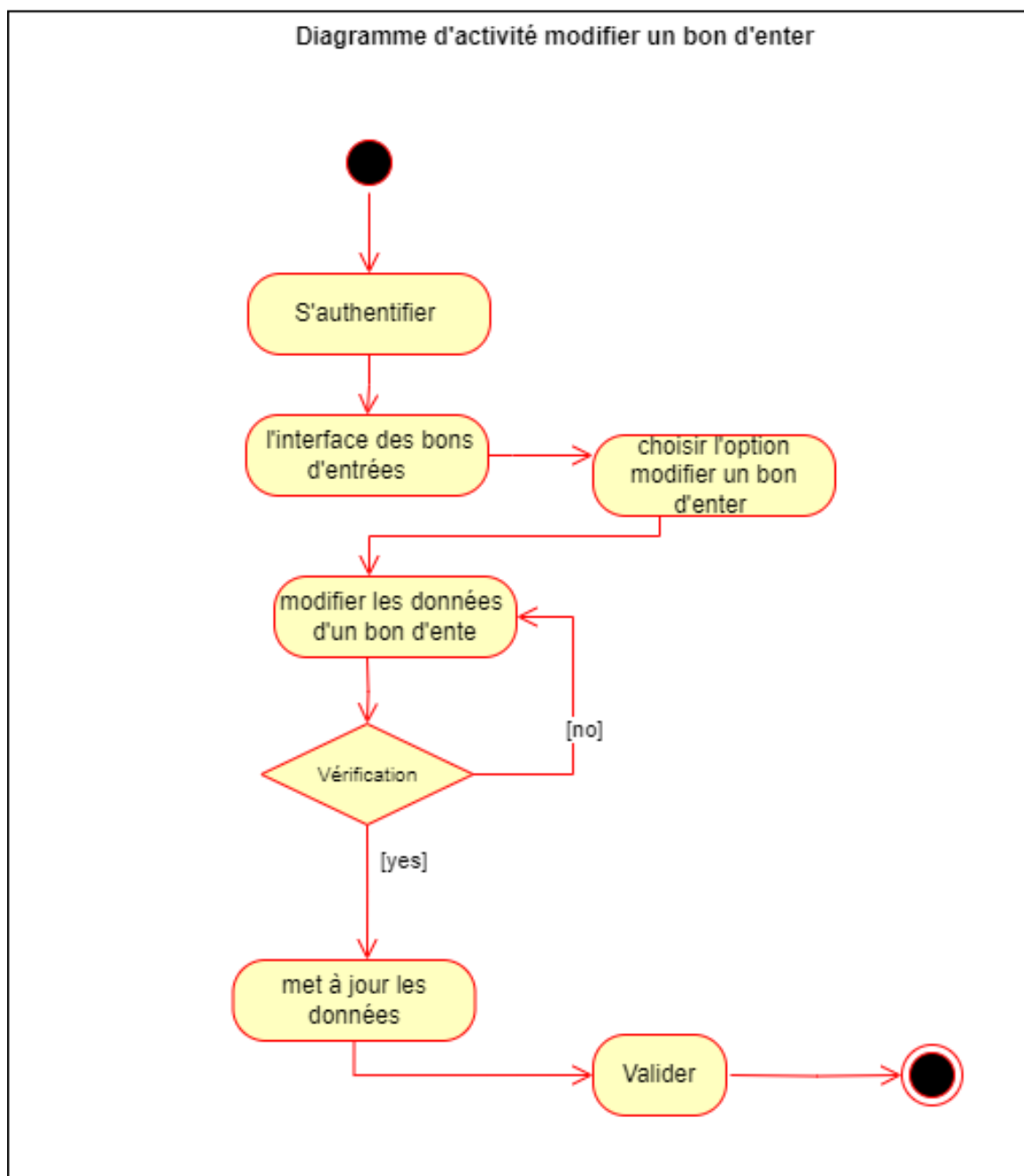


FIGURE 4.13 – Diagramme d'activité «Modifier un Bon d'entrée »

Diagramme d'activité «Ajouter un Bon de transfert »

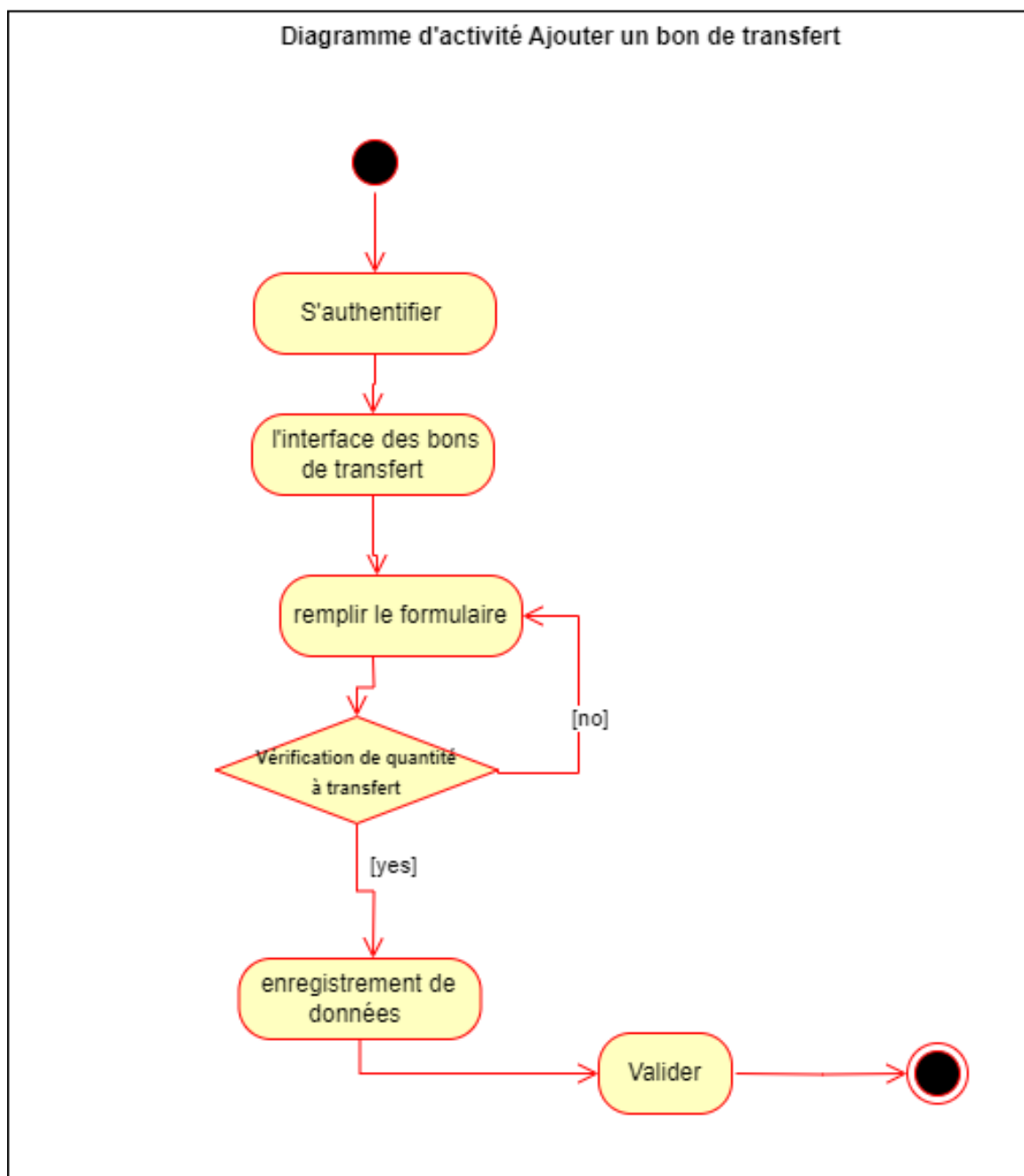


FIGURE 4.14 – Diagramme d'activité «Ajouter un Bon de transfert »

4.1.5 Réalisation


Interface de Gestion des missions

id	Nom	départ	retour	Destination	Passport	Objet	Adresse	Nb jours	contacter	Actions
1	fraj chemili	2023-03-28T00:00:00.000Z	2023-03-30T00:00:00.000Z	aucun Object	111111	aucun Object	aucun Object	2	aucun Object	
2	fraj chemili	2023-03-28T00:00:00.000Z	2023-03-29T00:00:00.000Z	aucun Object	111111	aucun Object	aucun Object	1	aucun Object	
3	Ahmed Beikahlaa	2023-05-31T00:00:00.000Z	2023-06-03T00:00:00.000Z	aucun Object	111111	aucun Object	aucun Object	3	aucun Object	

FIGURE 4.15 – Interface de Gestion des missions

Cette interface permet à l'administrateur de gérer les missions , L'administrateur peut ajouter, modifier supprimer ou imprimer un ordre de mission .

L'interface mission un congé permet à l'utilisateur de demander un congé annuelle



Tunis , Le: 31/05/2023

ORDRE DE MISSION

Le directeur de la société Comelit Recherche & Développement (CRED) atteste que :

Nom Et Prénom :	fraj chernili
Nationalité	Tunisien
Passport N° :	111111
validation passport:	2023-03-22
Date de début :	2023-03-28T00:00:00.000Z
Date de fin :	2023-03-30T00:00:00.000Z
Durée de séjours:	2 jours
Destination :	aucun Object
Ville de destination :	aaa
Frais de mission:	13.26 EUR
Objet de mission :	aucun Object
Adresse	aucun Object
Personne	aucun Object

Les frais de séjour et les frais de voyage sont pris en charge par la société Comelit group SPA durant sa mission en Italie.

COMELIT RECHERCHE ET DEVELOPPEMENT

SARL au capital de 6000 TND Siège
social Pole Technologie El Ghazala Bloque E1
RDC Route de Raoud 2088 Ariana| Tunisie

MF : 1378354 A/A/M00
RC : 801247692014
T + 216 70 834 125
F + 216 70 834 126

CCB Agence UBCI Raoud
RIB:11 053 0050028 034 978 55
web:www.comelitgroupe.comelit
E-mail:cred@comelit.it

www.comelitgroupe.com

FIGURE 4.16 – Interface demander un conge

4.1.6 Test

Cas de Test	Demarche	Comportement Attendu	Résultat
Test de gestion des missions	Ajout, modification, suppression , impression et recherche d'une mission	Gestion complète fonctionnelle	Conforme
Test de gestion de bons d'enter	Ajout, modification, suppression d'un bon d'nter	Gestion complète fonctionnelle	Conforme
Test de gestion de bons de transfert	Ajout, modification, suppression d'un bon de transfert	Gestion complète fonctionnelle	Conforme
Test de gestion de bons de sortie	Ajout, modification, suppression d'un bon de sortie	Gestion complète fonctionnelle	Conforme
Test de gestion de suivie	Ajout, modification d'un suivie	Gestion complète fonctionnelle	Conforme

TABLE 4.11 – Test du deuxième Sprint

4.2 Sprint 2 « Gestion de site web »

Ce sprint a pour but de développer les parties :

- Gestion des offres d'emplois et des stages
- Gestion des candidatures.
- Gestion des contacts et des événements .

4.2.1 Backlog du sprint

4.2.2 Analyse

Dans cette partie, nous présentons le diagramme de cas d'utilisation et la description textuelle des plus importants cas d'utilisation.

ID	User Story	ID tâche	Tâche	Estimation (jour)
1	Gestion de site Web	1.1	Les Interface de site web	11 jours
		1.2	Gérer les offres demploi : ajouter, afficher, modifier,supprimer un offre	
		1.3	Gérer les offres de stage : ajouter, afficher, modifier,supprimer un offre	5 jours
		1.4	Gérer Contact : consulter, supprimer un contact	3 jours
		1.5	traiter les candidatures :consulter , Accepter, refuser une candidature	4 jours

TABLE 4.12 – Backlog du Sprint 1 de release 2

Diagramme de cas d'utilisation « Gestion de site web »

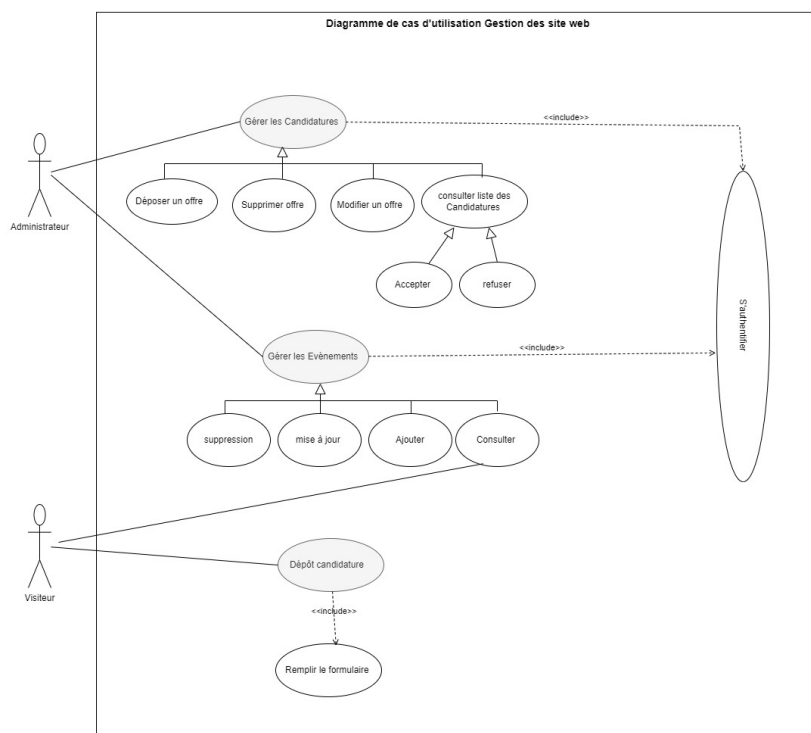


FIGURE 4.17 – Diagramme de cas d'utilisation "Gestion de site web "

- Description textuelle du cas d'utilisation « Gérer le site web »

Cas d'utilisation	Gérer le site web
Acteurs	Administrateur
Préconditions	L'acteur est authentifié
Description	<p>La gestion du site web consiste à :</p> <ul style="list-style-type: none"> - Gérer les offres d'emploi : disposer, modifier, supprimer, traiter les demandes d'emplois - Gérer les offres de stage : disposer, modifier, supprimer, traiter les demandes de stages - Gérer les candidatures spontanée : traiter les demandes de stages - Gérer les événements : ajouter, modifier, supprimer - Gérer les contacts : consulter, supprimer

TABLE 4.13 – Cas d'utilisation : Gérer le site web

- Description textuelle du cas d'utilisation « Disposer une offre d'emploi »

Cas d'utilisation	Disposer une offre d'emploi
Acteurs	Administrateur
Préconditions	Acteur authentifié
Postconditions	Offre d'emploi ajoutée
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur clique sur le bouton « Disposer une offre d'emploi » 2. Le système affiche le formulaire d'ajout d'une offre d'emploi 3. L'acteur remplit tous les champs nécessaires et valide 4. Le système vérifie les données saisies 5. Le système crée une nouvelle offre d'emploi
Déroulement alternatif	<p>4-a. L'acteur saisit des données incorrectes</p> <p>4-a-1. Le système affiche des messages d'erreur</p> <p>4-a-2. Reprise de l'étape 3 du scénario nominal</p>

TABLE 4.14 – Cas d'utilisation : Disposer une offre d'emploi

- Description textuelle du cas d'utilisation « Traiter une demande de stage »

Cas d'utilisation	Traiter une demande de stage
Acteurs	Administrateur
Préconditions	Acteur authentifié
Postconditions	Demande de stage mise à jour
Déroulement nominal	<ol style="list-style-type: none"> 1. Le système affiche la liste des demandes de stage en cours 2. L'acteur choisit entre accepter ou refuser la demande en cliquant sur le bouton correspondant 3. Le système affiche une notification de confirmation indiquant "oui" ou "non" 4. L'acteur confirme l'opération 5. Le système envoie un e-mail pour informer le candidat
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'administrateur annule l'opération 4-a-1. Le système affiche la liste des demandes de stage

TABLE 4.15 – Cas d'utilisation : Traiter une demande de stage

- Description textuelle du cas d'utilisation « Traiter une demande d'emploi »

Cas d'utilisation	Traiter une demande d'emploi
Acteurs	Administrateur
Préconditions	Acteur authentifié
Postconditions	Demande d'emploi mise à jour
Déroulement nominal	<ol style="list-style-type: none"> 1. Le système affiche la liste des demandes d'emploi en cours 2. L'acteur choisit entre accepter ou refuser la demande en cliquant sur le bouton correspondant 3. Le système affiche une notification de confirmation indiquant "oui" ou "non" 4. L'acteur confirme l'opération 5. Le système envoie un e-mail pour informer le candidat
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'administrateur annule l'opération 4-a-1. Le système affiche la liste des demandes d'emploi

TABLE 4.16 – Cas d'utilisation : Traiter une demande d'emploi

- Description textuelle du cas d'utilisation « Supprimer un événement »

Cas d'utilisation	Supprimer un événement
Acteurs	Administrateur
Préconditions	Acteur authentifié
Postconditions	Événement supprimé
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur clique sur le bouton « Supprimer un événement » 2. Le système affiche une notification de confirmation indiquant "oui" ou "non" 3. L'acteur confirme la suppression de l'événement 4. Le système vérifie les données saisies 5. Le système affiche une notification de succès après la suppression
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'administrateur annule la suppression 4-a-1. Le système affiche la liste des événements

TABLE 4.17 – Cas d'utilisation : Supprimer un événement

- Description textuelle du cas d'utilisation « modifier un offre de stage »

Cas d'utilisation	modifier un offre de stage
Acteurs	Administrateur
Préconditions	Acteur authentifié
Postconditions	Un offre de stage a été modifiée
Déroulement nominal	<ol style="list-style-type: none"> 1. L'acteur clique sur le bouton "Modifier un offre de stage". 2. Le système affiche le formulaire "Modifier un offre de stage". 3. L'acteur change les données qu'il désire et valide. 4. Le système vérifie les données saisies. 5. Le système affiche une notification de mise à jour.
Déroulement alternatif	<ol style="list-style-type: none"> 4-a. L'acteur saisit des données incorrectes. 4-a-1. Le système affiche un message d'erreur. 4-a-2. Reprise de l'étape 3 du scénario nominal.

TABLE 4.18 – Cas d'utilisation : modifier un offre de stage

Diagramme de séquence du cas d'utilisation «Diposer une offre d'emploi »

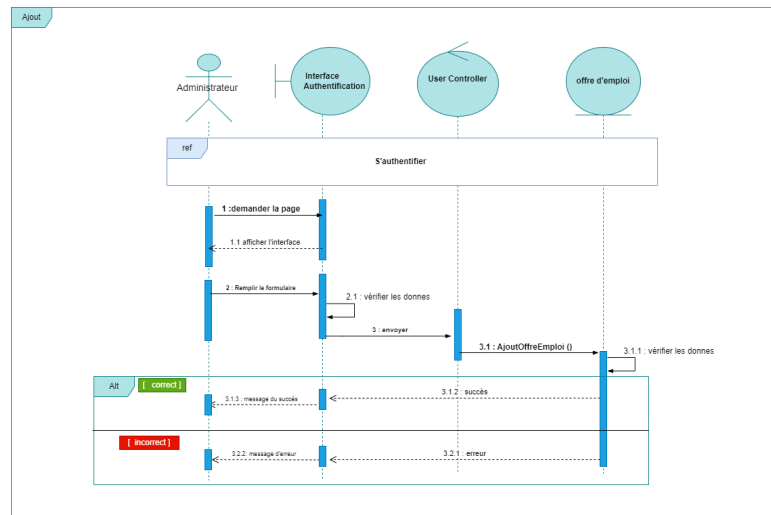


FIGURE 4.18 – Diagramme de séquence «Disposer une offre d'emploi »

4.2.3 Conception

Diagramme de classes

La figure ci-dessous représente le diagramme de classe utilisé pour le développement du Sprint

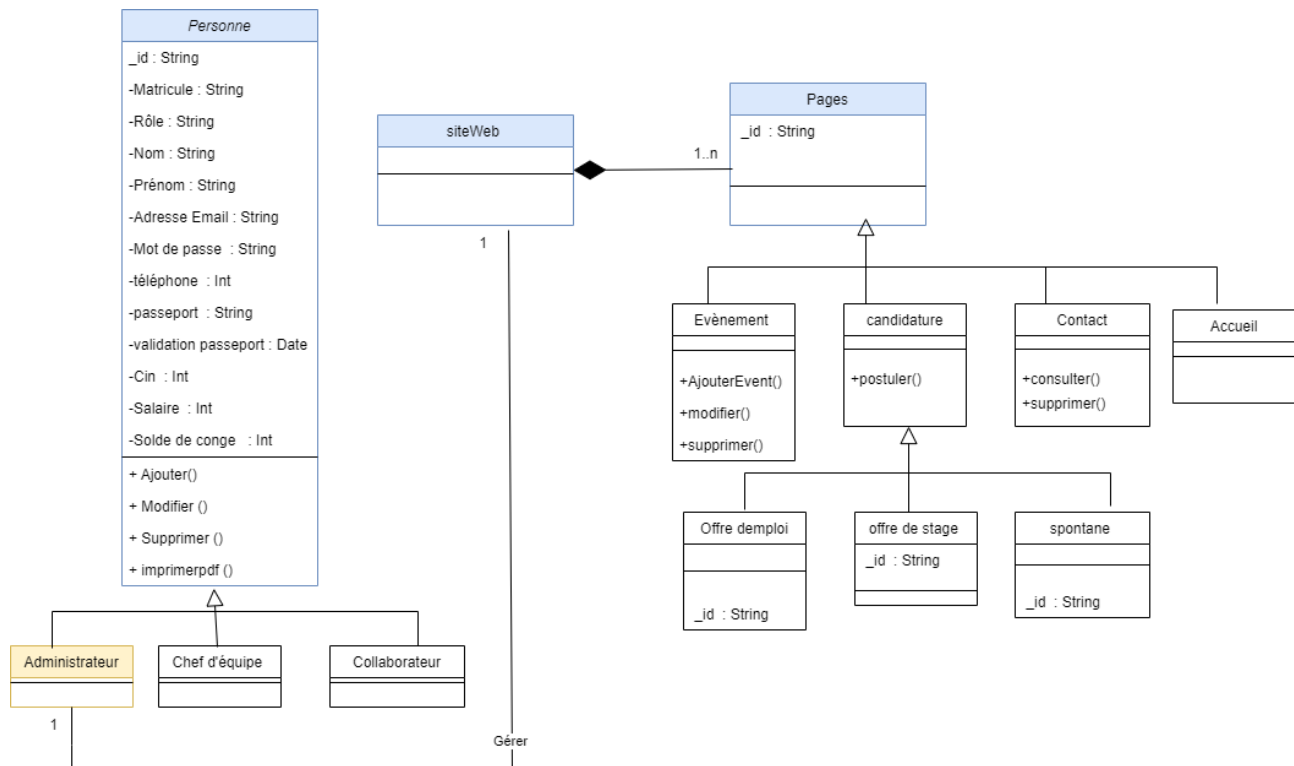


FIGURE 4.19 – Diagramme de classe « Gérer site web »

Diagramme de séquence du cas d'utilisation «Traiter une offre d'emploi »

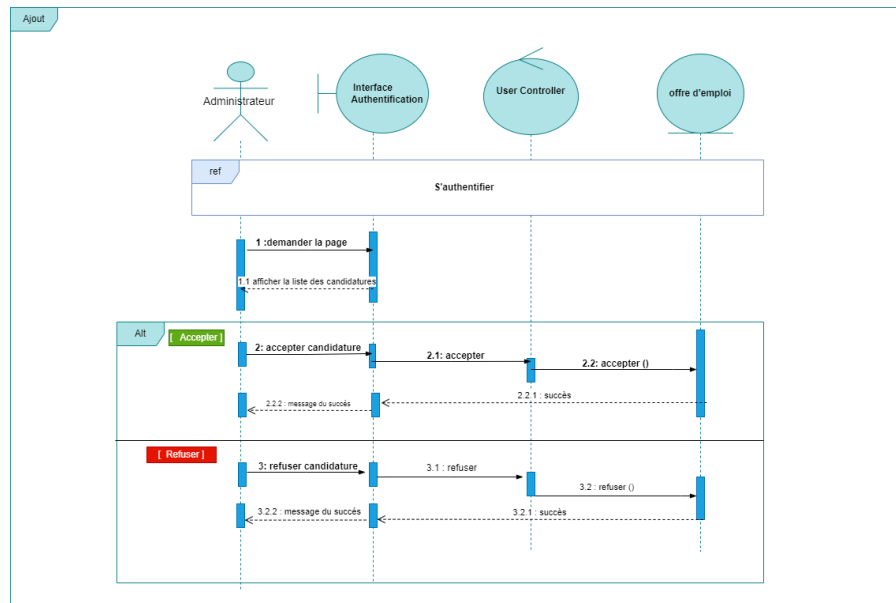


FIGURE 4.20 – Diagramme de séquence «Traiter une offre d'emploi »

Diagramme de séquence du cas d'utilisation «Supprimer un Evenement »

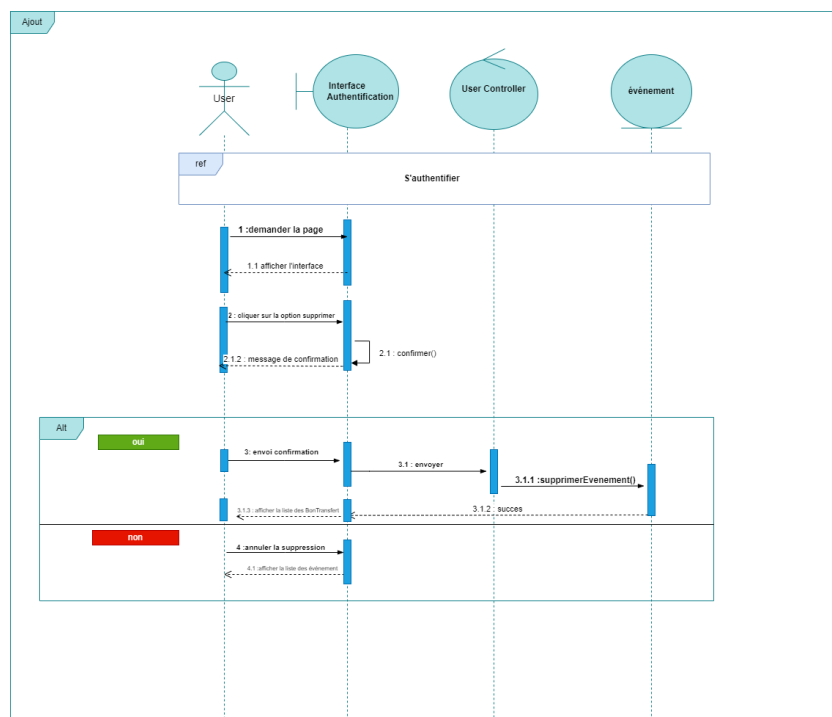


FIGURE 4.21 – Diagramme de séquence «Supprimer un Evenement »

Diagramme de séquence du cas d'utilisation «Traiter une demande de stage »

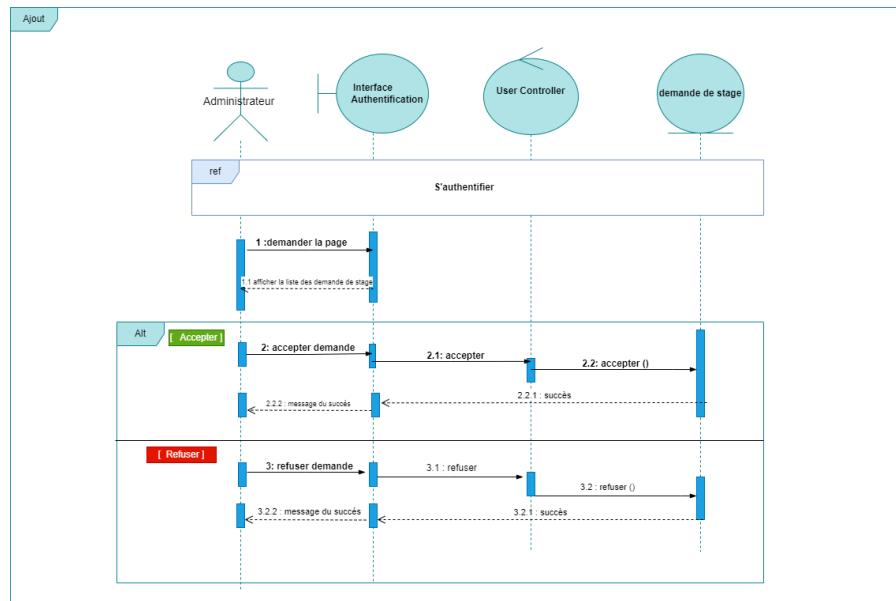


FIGURE 4.22 – Diagramme de séquence «Traiter une demande de stage »

Diagramme d'activité «Ajouter un offre d'emploi »

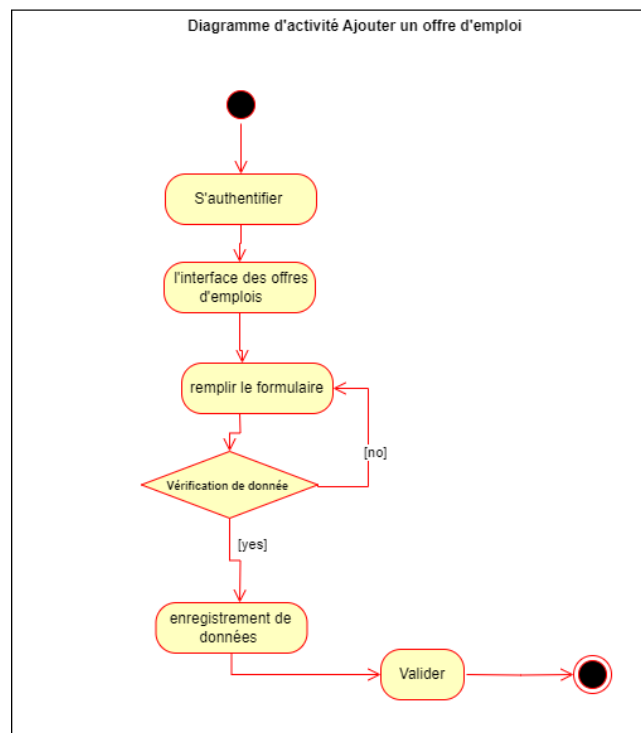


FIGURE 4.23 – Diagramme d'activité «Ajouter une offre d'emploi »

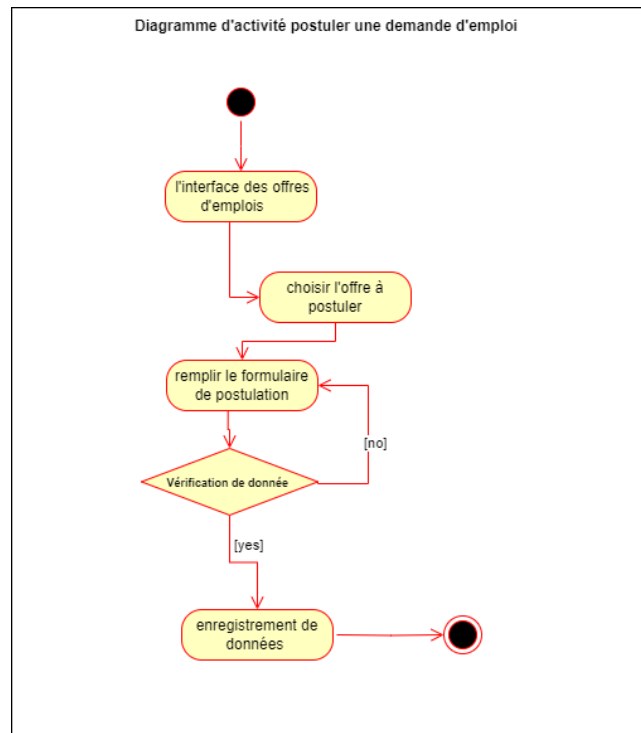
Diagramme d'activité «Postuler une demande d'emploi »

FIGURE 4.24 – Diagramme d'activité «Postuler une demande d'emplo »

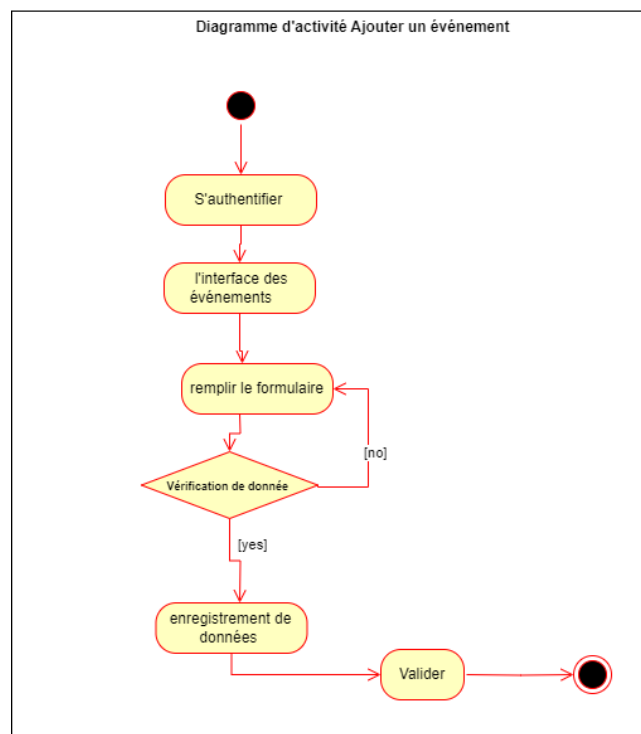
Diagramme d'activité «Ajouter un événement »

FIGURE 4.25 – Diagramme d'activité «Ajouter un événement »

4.2.4 Réalisation

Interface des offres de Comelit

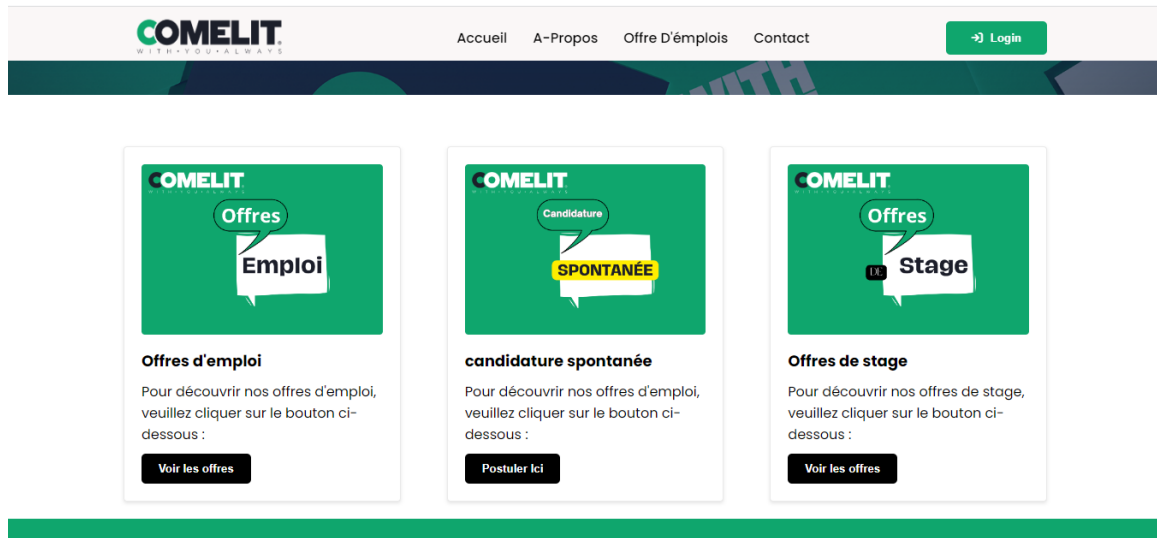


FIGURE 4.26 – Interface d'offre de Comelit

Cette interface permet aux utilisateurs de consulter les offres d'emploi et de stage de Comelit, ainsi que de postuler spontanément.

4.2.5 Conclusion

Dans ce chapitre nous avons présenté le dernier release qui comporte la « Gestion des Missions », la « Gestion des Stocks » ainsi que la « Gestion de site web ». Pour ce faire, nous avons passé par l'analyse, la conception, la réalisation et les tests .

CONCLUSION GÉNÉRALE

Ce projet se dirige dans le cadre De notre projet fin d'étude de Licence nationale en technologie de l'informatique, spécialité développement système d'information System à ISET KEF pour le compte de la société Comelit recherche et développement.

De Point de vue personnel,notre PFE nous a permis d'apprendre MERN dont Node Js, Express,Mongo DB, HTML et CSS , et aussi de travailler avec la méthodologie SCRUM et de mener à bien un projet en suivant les contraintes d'un cahier de charges.

Nous sommes appelés dans ce travail de développer un site web pour la société pour la gestion ressources humaines ,stock et d'offre des emplois pour les employés internes et les externes, nous avons terminé ce stage que nous espérons enrichissant pour nous et pour tous qui consulte ce rapport qui résume quatre mois de travail rigoureux. Pour le moment le site est presque terminé nous souhaitons qu'il trouvera les conditions nécessaires pour entrer en vigueur

Enfin, il faut noter que ce projet est un noyau sur qui plusieurs perspectives peuvent être construites. À titre d'exemple nous pourrions développer au futur un système de pointage dans l'application et une fonction pour envoyer des notifications personnalisées pour les employés internes de la société . Aussi le chatbot qui interagit avec les utilisateurs externes de manière conversationnelle, en utilisant des techniques d'intelligence artificielle (IA).

- [1] Scrum :
Disponible sur : <https://fr.wikipedia.org/wiki/Scrum>
- [2] StarUML :
Disponible sur : <https://fr.wikipedia.org/wiki/StarUML>
- [3] Visual Studio Code :
Disponible sur : https://fr.wikipedia.org/wiki/Visual_studioCode
- [4] Bootstrap :
Disponible sur : [https://fr.wikipedia.org/wiki/Bootstrap\(framework\)](https://fr.wikipedia.org/wiki/Bootstrap(framework))
- [5] Postman :
[https://fr.wikipedia.org/wiki/Postman\(logiciel\)](https://fr.wikipedia.org/wiki/Postman(logiciel))
- [6] Draw.io :
[https://fr.wikipedia.org/wiki/Postman\(logiciel\)](https://fr.wikipedia.org/wiki/Postman(logiciel))
- [7] UML :
Disponible sur : [https://fr.wikipedia.org/wiki/UML\(informatique\)](https://fr.wikipedia.org/wiki/UML(informatique))