

UNIVERSITE DE SOUSSE
**INSTITUT SUPERIEUR D'INFORMATIQUE ET DES
TECHNOLOGIES DE COMMUNICATION**



Rapport de stage de fin d'études

Présenté en vue de l'obtention du diplôme de Licence en
Technologies de l'Information et de la Communication
Spécialité : Télécommunications

Web application for users onboarding

Réalisé par :
TABBABI JAWHER ET BAYACH IHEB

Encadrant académique : M. FATHALLAH WYSSEM

Encadrant professionnel : Mme. GMIZA WISSAL

Société d'accueil

LEONI

Année universitaire : 2023/2024

LEONI

Je soussignée **Mme. Gmiza Wissal**, autorise l'étudiants **Tabbabi Jawhar et Bayach Iheb** à déposer leur rapport de projet de fin d'études (PFE).

Signature et cachet



Je soussigné **M. Fathallah Wyssem**, autorise les étudiants **Tabbabi Jawhar et Bayach Iheb** à déposer leur rapport de projet de fin d'études (PFE).

Signature

Dédicace

Nous dédions ce travail à :

À nos chers parents, nos sœurs, nos frères, nos amis, à nos enseignants et à toute notre famille, nous dédions humblement ce travail. Leurs encouragements constants, leur soutien indéfectible et leurs sacrifices désintéressés ont été les fondations invisibles mais solides sur lesquelles nous avons construit ce projet. Leur amour inconditionnel et leur foi en notre potentiel nous ont guidés à travers les défis et les succès de ce parcours. Sans eux, rien de tout cela n'aurait été possible.

Iheb Bayach & Jawhar Tabbabi

Remerciements

C'est avec une immense gratitude que je prends la plume pour exprimer mes remerciements les plus sincères à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet.

*Je souhaite tout particulièrement exprimer ma reconnaissance envers mes tuteurs de stage au sein de l'entreprise **LEONI**, Mme. **Gmiza Wissal** et Mme. **Arfaoui Maroua**, pour leur précieux soutien, leurs conseils avisés et leur encouragement constant tout au long de cette aventure.*

*Un immense merci également à mon encadrant académique, M. **Fathallah Wyssem**, pour sa disponibilité, sa bienveillance et son expertise qui ont grandement enrichi ce travail.*

Je saisirai également cette occasion pour exprimer ma gratitude envers tous les membres du jury qui ont pris le temps d'examiner et d'évaluer ce projet avec attention.

Enfin, je tiens à remercier mes proches, ma famille et mes amis, pour leur soutien indéfectible et leur compréhension tout au long de cette période intense.

Merci du fond du cœur.

Table des matières

Liste des abréviations	XII
Introduction générale	1
I Contexte général	2
1 Présentation de l'organisme d'accueil :	2
1.1 Chiffres clés :	2
1.2 Les services offerts :	3
2 Etude de l'existant :	3
3 Critique de l'existant :	4
4 Problématique :	6
5 La solution proposée	6
6 La méthodologie adoptée :	6
6.1 Méthodologie agile :	7
6.2 Méthodologie scrum :	7
6.3 Méthode en cascade :	7
7 Présentation de Scrum :	8
7.1 caractéristiques de la méthodologie scrum :	8
8 Les Intervenants :	9
8.1 Planification d'un projet avec Scrum :	9

TABLE DES MATIÈRES

8.2	Artefacts Scrum :	10
8.3	Événements :	10
II	Analyse et spécification des besoins	13
1	Capture des besoins :	13
1.1	Identification des acteurs :	13
1.2	Spécification des besoins :	14
2	Product Backlog :	15
2.1	Technique de priorisation du Product Backlog (MoSCoW) :	16
2.2	Les fonctionnalités du backlog :	17
2.3	Diagramme de cas d'utilisation global :	17
2.4	Diagramme de classe globale :	18
3	Planification des sprints du projet :	19
4	Environnement de travail :	19
4.1	Environnement technologique :	19
4.2	Environnement logiciel :	22
5	Architecture générale de l'application :	25
6	Renforcement de la Sécurité :	25
III	sprint 1 "S'authentifier"	29
1	Backlog de sprint 1 :	29
2	Spécifications fonctionnelle :	30
2.1	Diagramme de cas d'utilisation :	30
2.2	Description textuelle des cas d'utilisations :	31
3	Diagramme des séquences détaillés :	32
3.1	Diagramme des séquences "S'authentifier" :	32
4	Diagramme de classe :	34

TABLE DES MATIÈRES

5	Réalisation :	34
5.1	Interface d'authentification :	34
IV sprint 2 "Consulter le Dashboard"		37
1	Backlog de sprint 2 :	37
2	Spécifications fonctionnelles :	37
2.1	Diagramme de cas d'utilisation :	38
2.2	Description textuelle des cas d'utilisation :	38
3	Diagramme de séquence détaillé :	39
3.1	Diagramme de séquence "Consulter le dashboard" :	39
4	Diagramme de classe :	40
5	Réalisation :	40
5.1	Interface du dashboard :	40
V sprint 3 "Gérer utilisateurs"		45
1	Backlog de sprint 3 :	45
2	Spécifications fonctionnelles :	45
2.1	Diagramme de cas d'utilisation :	46
2.2	Description textuelle des cas d'utilisation :	47
3	Diagramme de séquence détaillé :	49
3.1	Diagramme de séquence "Gestion des utilisateurs" :	50
4	Diagramme de classe :	51
5	Réalisation :	52
5.1	Interface de gestion des utilisateurs :	52
VI Sprint 4 "Gérer les designs"		55
1	Backlog de sprint 4 :	55
2	Spécifications fonctionnelles :	56

TABLE DES MATIÈRES

2.1	Diagramme de cas d'utilisation :	56
2.2	Description textuelle des cas d'utilisation :	57
3	Diagramme de séquence détaillé :	63
3.1	Diagramme de séquence "Gestion des designs" :	63
4	Diagramme de classe :	64
5	Réalisation :	64
5.1	Interface ajout des designs et affectation des tâches :	64
Conclusion générale		75
WEBOGRAPHIE		76

Table des figures

I.1	logo de la société LEONI	2
I.2	Tableau de suivi de la migration de la design de production	4
I.3	tableau de suivi de la migration de la design de production	4
I.4	statut de progression de phase de pilotage	5
I.5	Dashboard d'exécution	5
I.6	Exemple de projet en scrum	8
II.1	Diagramme des cas d'utilisation global	18
II.2	Diagramme de class global	27
II.3	Architecture MVC	28
III.1	Diagramme de cas d'utilisation de l'Authentification	30
III.2	Diagramme de séquence de l'authentification	33
III.3	Diagramme de classe de sprint 1	34
III.4	interface Login	35
III.5	Interface login avec un message d'erreur de remplir tous les champs	35
III.6	Interface login avec un message d'erreur	36
IV.1	Diagramme de cas d'utilisation : Consultation du dashboard	38
IV.2	Diagramme de séquence de la consultation du dashboard	40
IV.3	Diagramme de classe du Sprint 2	41

TABLE DES FIGURES

IV.4 Interface du dashboard 1	42
IV.5 Interface du dashboard 2	43
IV.6 Interface du dashboard 3	44
V.1 diagramme de cas d'utilisation sprint 3	46
V.2 Diagramme de séquence de consultation et suppression des utilisateurs	50
V.3 Diagramme de séquence d'ajout d'un utilisateur	51
V.4 Diagramme de séquence de modification d'un utilisateur	52
V.5 Diagramme de classe sprint 3	53
V.6 Interface de gestion des utilisateurs	54
V.7 Interface de gestion des utilisateurs	54
VI.1 Diagramme de cas d'utilisation : Gestion des designs	57
VI.2 Diagramme de séquence de la gestion des tâches	64
VI.3 Diagramme de classe : Gestion des tâches	65
VI.4 Diagramme de classe : Gestion des tâches	66
VI.5 Diagramme de classe : Gestion des tâches	67
VI.6 Diagramme de classe : Gestion des tâches	68
VI.7 Diagramme de classe : Gestion des tâches	69
VI.8 Diagramme de classe : Gestion des tâches	70
VI.9 Diagramme de classe : Gestion des tâches	71
VI.10Interface du liste des designes	72
VI.11Interface du liste des designes	72
VI.12Interface d'ajout des designes	73
VI.13Interface d'ajout des designes	73
VI.14Interface d'affectation des taches	74
VI.15Interface de gestion des taches "admin"	74

Liste des tableaux

I.1	Tableau comparatif des différentes méthodologies	7
I.2	Composition de l'équipe Scrum avec les rôles	11
II.1	Backlog de produit :	17
II.2	sprint de projet	19
II.3	Environnement technologique	21
II.4	Environnement logiciel	24
III.1	Backlog sprint 1	29
III.2	Description textuelle du cas d'utilisation "S'authentifier"	31
IV.1	Backlog du Sprint 2	37
IV.2	Description textuelle du cas d'utilisation "Consulter le dashboard"	39
V.1	Backlog de sprint 3	45
V.2	Description textuelle du cas d'utilisation "Consulter et supprimer des utilisateurs"	47
V.3	Description textuelle du cas d'utilisation " Ajouter un utilisateur"	48
V.4	Description textuelle du cas d'utilisation " Consulter et modifier un utilisateur"	49
VI.1	Backlog de sprint 4 :	55

LISTE DES TABLEAUX

VI.2 Description textuelle du cas d'utilisation "Gestion des designs"	58
VI.3 Description textuelle du cas d'utilisation "L'utilisateur consulte et supprime des designs"	58
VI.4 Cas d'utilisation : Modification de Designs	59
VI.5 Description textuelle du cas d'utilisation "L'utilisateur consulte et valide des designs"	60
VI.6 Description textuelle du cas d'utilisation "L'utilisateur consulte et modifie des taches"	61
VI.7 Description textuelle du cas d'utilisation "L'utilisateur consulte et supprime des taches"	62
VI.8 Description textuelle du cas d'utilisation "L'utilisateur consulte et travaille une tâche"	63

Liste des abréviations

API Application Programming Interface

HTTP Hypertext Transfer Protocol

IDE Integrated Development Environment

JSON JavaScript Object Notation

UML Unified Modeling Language

JS Java Script

MVC Modèle-Vue-Contrôleur

JWT JSON Web Token

HTTPS Hypertext Transfer Protocol Secure

SQL Structured Query Language

MoSCoW Must, Should, Could, Would

CRUD Create, Read, Update, Delete

Introduction générale

Avec le développement continu de la technologie, les entreprises sont confrontées à de plus en plus de défis dans la gestion des données et l'optimisation des processus. Les applications web offrent de nombreux avantages par rapport aux applications traditionnelles, telles que la portabilité, l'accessibilité, la scalabilité et la sécurité.

Dans le cadre de ce Projet, j'ai réalisé une application web qui vise à digitaliser un processus de travail et à faire le suivi des employés au sein de l'entreprise "LEONI". Ce projet consiste à développer une application web qui permet de gérer les tâches, les projets, les équipes, les évaluations des employés de LEONI. Grâce à ses fonctionnalités, l'application web propose également une dashboard et un environnement de suivi de progression des tâches et du projet pour faciliter le contrôle et la prise de décision au sein de l'entreprise.

Le but de ce rapport est de fournir un aperçu complet des réalisations du projet. Il est divisé en quatre chapitres pour plus d'organisation et de clarté. Le premier chapitre présente l'organisation d'accueil et donne un aperçu général du sujet traité. Dans le deuxième chapitre, nous approfondissons les exigences fonctionnelles et non fonctionnelles spécifiques de l'application. Passant au troisième chapitre, nous explorons les différents aspects conceptuels du projet. Enfin, le quatrième chapitre approfondit les détails du processus de mise en œuvre, y compris les outils utilisés, les différentes étapes de développement du système et un aperçu des résultats obtenus. Pour conclure, une section "Conclusion générale" résume notre travail et présente les perspectives d'avenir potentielles.

Chapitre I

Contexte général

1 Présentation de l'organisme d'accueil :

1.1 Chiffres clés :

LEONI est une société globale spécialisée dans la fourniture de produits, de solutions et de services pour la gestion de l'énergie et des données, ciblant principalement l'industrie automobile et d'autres secteurs industriels. Avec une présence mondiale de plus de 95 000 collaborateurs répartis dans 32 pays et plus de 100 sites, LEONI se positionne comme un leader dans son secteur. LEONI Tunisie, créé en 1977, a été le premier établissement de l'entreprise en dehors de l'Europe[1].



FIGURE I.1 – logo de la société LEONI

CHAPITRE I. CONTEXTE GÉNÉRAL

Informations de contact :

- Adresse : Zone Industrielle de Messadine, BP, 67,4000 Sousse, Tunisie.
- Téléphone : +216 73 323 820 / +216 73 32 35 77.
- Site Web : www.leoni.com.
- Premier Responsable : Monsieur Mohamed LAARBI ROUIS.
- Nombre d'Employés : 140 000 cadres et ouvriers.
- Branches de l'Entreprise : Trois sites de production situés à Messadine (Sousse), Mateur (Bizerte) et Ezzahra (Ben Arous).

1.2 Les services offerts :

La chaîne de valeur s'étend des câbles standardisés, des câbles spéciaux et des câbles de données aux systèmes de câblage hautement complexes et aux composants associés, du développement à la production. LEONI soutient ses clients en tant que partenaire d'innovation et fournisseur de solutions avec une expertise distinctive en développement et en systèmes sur la voie de concepts de mobilité de plus en plus durables et connectés. Le développement de solutions de câbles de nouvelle génération et de systèmes de câblage permet, par exemple, grâce à l'architecture zonale, une complexité réduite et des niveaux d'automatisation plus élevés.

2 Etude de l'existant :

Actuellement, le suivi de la progression du travail des designers repose sur l'utilisation d'un tableau Excel centralisé. Les chefs d'équipe assignent les tâches aux designers manuellement. Pour la visualisation des statistiques, ils importent les données à un autre logiciel pour compléter l'analyse. Chaque designer met régulièrement à jour ce tableau pour refléter les avancements sur les tâches assignées.

CHAPITRE I. CONTEXTE GÉNÉRAL

3 Critique de l'existant :

Cependant, en raison de la nature statique de ce tableau, il devient difficile pour les chefs d'équipe de suivre efficacement l'état d'avancement global du travail et de prendre des décisions rapides. La taille et la complexité du tableau rendent la visualisation laborieuse, tandis que les fonctionnalités limitées d'analyse entravent la capacité à identifier rapidement les tendances et les domaines nécessitant une attention particulière. Il est impératif de développer une solution plus efficace et conviviale pour permettre une gestion plus transparente et proactive de l'avancement du travail au sein de l'équipe de design.

Tracking List of Production Design Migration												
				Maintained by Admin	Maintained by User	Maintained by Key User						
Plant	Account	Project name	Design Description	Used System before migration	Design Nbr	Count of Leads	Target Design time assigned [H]	Planned Start Date [dd/mm/yy]	Design import responsible [Design Owner]	Components loaded to LP DB [Yes/No]	Product Design Completed [dd/mm/yy]	
LTN3C	BMW Group	G2X	MZ ZB LTG FRONTEND G2X EFZ	LD	7887562	50	7,2	17/06/23	Atef	Yes	16/06/2023	

FIGURE I.2 – Tableau de suivi de la migration de la design de production

Design Index	Start Date [dd/mm/yy]	DRC completed status [OK/NOK]	Lead Length updated [OK/NOK]	ECNs workpackage received	Processing TVM completed status [OK/NOK]	sBOM upload status [OK/NOK]	Checked by	Design Migrationstatus	Comments
01	30/06/2023	OK	ok	TC2	OK	OK	Sarhan	Completed	None

FIGURE I.3 – tableau de suivi de la migration de la design de production

CHAPITRE I. CONTEXTE GÉNÉRAL

Report Status & Progress

Progress status of Pilot Phase

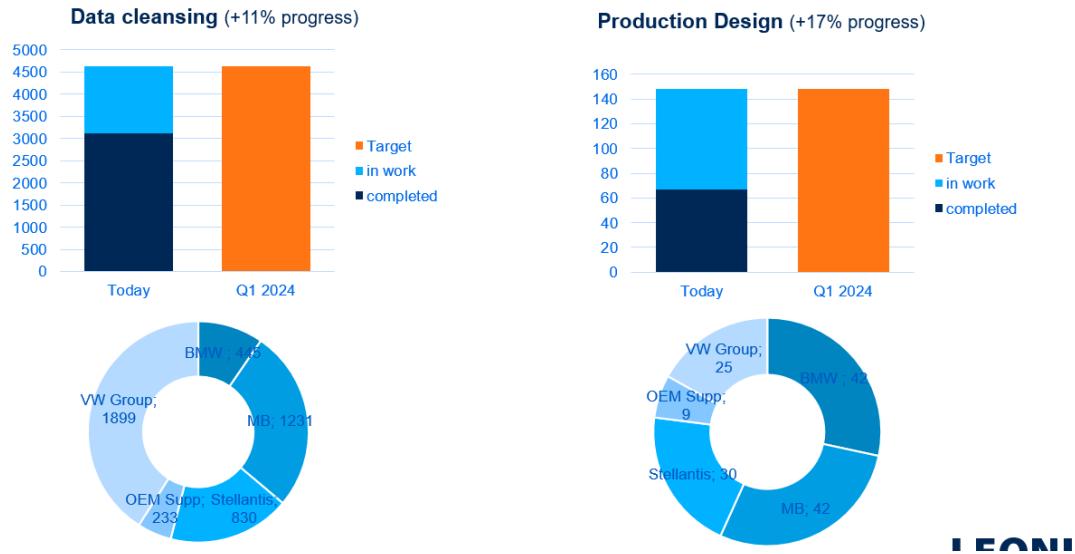


FIGURE I.4 – statut de progression de phase de pilotage

Report Status & Progress

Progress status of Pilot Phase

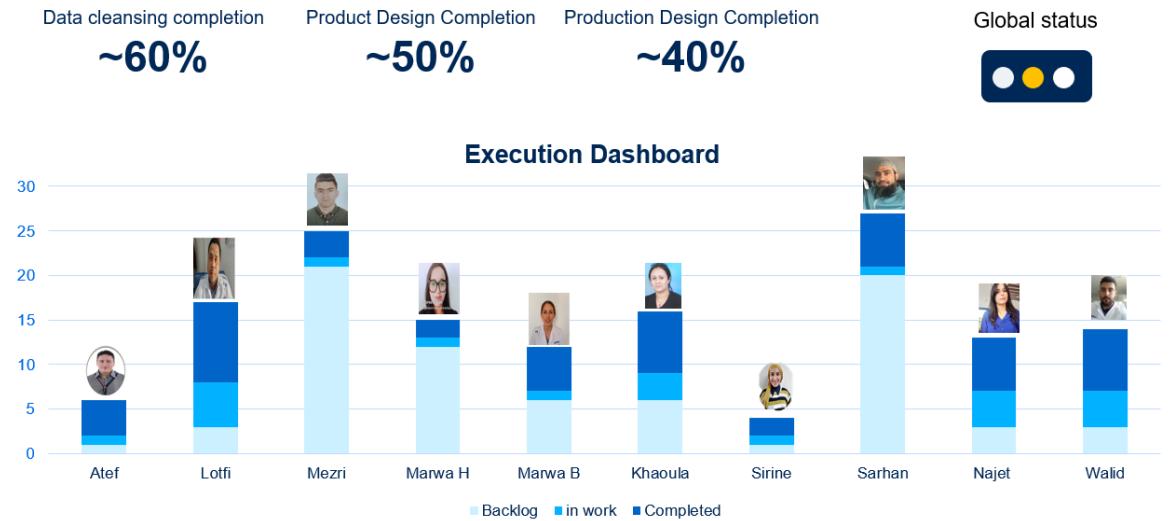


FIGURE I.5 – Dashboard d'exécution

4 Problématique :

La méthode actuelle de suivi des designers via Excel présente des limites significatives en termes d'efficacité et de gestion des données. Les feuilles de calcul Excel, bien qu'elles puissent être utiles pour certaines tâches, sont souvent soumises à des erreurs humaines, difficiles à mettre à jour en temps réel, et elles sont limitées dans leur capacité à fournir des analyses approfondies. De plus, la dispersion des données sur plusieurs feuilles de calcul complique la centralisation et la cohérence des informations sur les employés.

5 La solution proposée

Afin de surmonter ces défis, nous visons à établir une solution de suivi des employés plus robuste et intégrée. En combinant différentes technologies, notre objectif est de créer une plateforme offrant une collecte de données plus précise et une analyse approfondie des performances individuelles et d'équipe.

En adoptant une approche plus systématique et automatisée du suivi des employés, nous cherchons à optimiser l'allocation des effectifs et à favoriser un environnement de travail plus collaboratif et productif.

6 La méthodologie adoptée :

Pour exécuter un projet, il est crucial de le faire sur la base d'un plan de travail solide. Par conséquent, il est important de comprendre les différentes méthodologies de gestion de projet. Nous comparerons les critères des trois méthodologies suivantes pour les distinguer : méthodologies agiles, méthodologie Scrum et méthodologie cascade.

CHAPITRE I. CONTEXTE GÉNÉRAL

6.1 Méthodologie agile :

La méthodologie Agile est une approche qui nous permet de diviser notre projet en petits morceaux et de les gérer. L'objectif est d'atteindre le résultat final défini par les objectifs en petites étapes itératives et pratiques. Elle est principalement utilisée dans les projets de développement de logiciels en raison de son accent pratique sur les résultats et les produits finaux.

6.2 Méthodologie scrum :

Scrum est davantage axé sur le maintien de la continuité productive des structures complexes déjà établies et sur la formation de groupes autonomes similaires à ceux que l'on trouve dans les méthodologies Agile, mais cette fois-ci, il est organisé autour de périodes de travail appelées sprints.

6.3 Méthode en cascade :

La méthode en cascade est une méthode de gestion de projet séquentielle qui divise le projet en plusieurs phases distinctes. Chaque phase ne peut commencer que si la phase précédente est terminée.

Méthodologies	Avantages	Inconvénients
En cascade	Elle utilise une structure claire. Elle a fixé l'objectif ultime plus tôt. Elle transmet effacement information.	Elle exclut le client. Elle repousse les tests à la fin de la mise en œuvre du projet.
Agile	Forcer le travail d'équipe. Établir une relation plus étroite et plus complète avec chaque client. La mise en œuvre du logiciel est plus rapide.	Peut-être difficile à comprendre. Il ne peut être véritablement mis en œuvre que lorsque les clients sont disponibles.
Scrum	Encourage le travail d'équipe. Plus grande précision, car l'article est souvent vu par le client. Le projet est plus adaptable et flexible.	Difficile à mettre en œuvre, en raison de la prédisposition et de la culture de l'organisation, du top management aux clients.

TABLE I.1 – Tableau comparatif des différentes méthodologies

7 Présentation de Scrum :

Scrum est une méthode agile largement utilisée dans le développement de logiciels. Elle se base sur un processus itératif et incrémental. Scrum est un cadre adaptable, rapide, flexible et efficace qui vise à fournir de la valeur au client tout au long du développement du projet. En effet la méthodologie Scrum s'adapte parfaitement à la décomposition de notre projet de fin d'études. Par conséquent, nous avons opté pour la méthode agile Scrum afin de garantir le bon déroulement des différentes phases de design et développement de notre système de suivi[2].

7.1 caractéristiques de la méthodologie scrum :

On peut citer :

- La simplicité et la légèreté.
- La transparence du scrum.
- Difficulté de la maîtrise.

Scrum Process

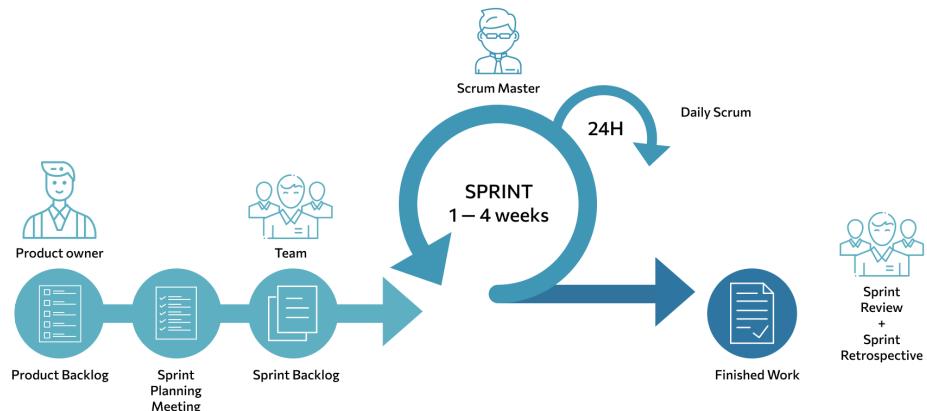


FIGURE I.6 – Exemple de projet en scrum

8 Les Intervenants :

Scrum est un processus empirique qui s'appuie sur l'expérience du terrain. Ce cadre se compose de définitions de rôles, de réunions et de produits livrables. En fait, Scrum définit trois rôles :

Product Owner : la personne (généralement représentant le client) qui détient la vision du produit à réaliser.

Scrum Master : le garant de l'application de la méthodologie Scrum.

Équipe : composée de développeurs, testeurs, architectes et autres personnes indispensables à la réalisation du projet. Les équipes sont organisées et ne changent pas pendant le sprint.

8.1 Planification d'un projet avec Scrum :

Pour utiliser correctement Scrum, nous devons comprendre le cycle de vie d'un sprint dans le processus Scrum. Le processus est décrit ci-dessous :

- Product Owner : La méthodologie Agile Scrum utilise le terme "propriétaire de produit" pour décrire le rôle du chef de produit. Il fait partie de l'équipe de projet et sa responsabilité est de définir un produit viable et maximiser le temps et le budget du projet. Les PO combinent souvent un savoir-faire technique pour communiquer avec les équipes de développement, ainsi qu'un savoir-faire pour gérer toutes les interactions avec les clients et les autres parties prenantes du projet.
- Lors de la planification du sprint, l'équipe sélectionne un ensemble d'user stories.
- L'équipe passe deux à quatre semaines à implémenter une "user story". Dans notre projet, les sprints durent deux semaines.
- Pendant le sprint, l'équipe se réunit quotidiennement en "Daily Scrum" pour synchroniser les tâches.

CHAPITRE I. CONTEXTE GÉNÉRAL

8.2 Artefacts Scrum :

Les artefacts Scrum sont spécifiquement conçus pour maximiser la transparence des informations clés afin que tout le monde ait la même compréhension. Les artefacts Scrum sont :

- Product Backlog : répertorie toutes les fonctionnalités, les exigences, les améliorations et les correctifs qui reflètent les modifications de produit prévues pour les futures versions. Les éléments du backlog de produit incluent les descriptions, les priorités, les estimations de complexité et les valeurs. Le Product Owner est responsable du contenu, de la disponibilité et du calendrier du Product Backlog. Le Product Backlog évolue en fonction du produit et du contexte dans lequel il est utilisé.
- Sprint Backlog : L'ensemble des éléments sélectionnés pour le sprint et le plan de livraison des incrément de produit et d'atteinte des objectifs du sprint. Le Sprint Backlog est la prédition de l'équipe de développement sur les fonctionnalités qui figureront dans le prochain incrément et sur le travail qui sera nécessaire pour fournir cette fonctionnalité dans l'incrément "Terminé".

8.3 Événements :

Nous pouvons distinguer plusieurs événements qui peuvent se produire au cours d'un projet Scrum. Les événements sont :

- **Sprint** : un bloc de durée fixe limité à 4 semaines. Pendant ce temps, vous pouvez créer des portions (incrément) de votre produit qui sont disponibles pour vos utilisateurs et potentiellement livrables.
- **Réunion de planification de sprint** : Toute l'équipe se réunit pour planifier le travail du sprint et répondre aux questions suivantes :

CHAPITRE I. CONTEXTE GÉNÉRAL

- Qu'est-ce qui peut être livré à la suite du sprint à venir ?
- Comment le travail sera-t-il effectué et quels sont les requis pour terminer l'incrément ?
- **Réunion quotidienne** : Il s'agit de la réunion quotidienne pour examiner l'avancement du travail de sprint, résoudre les problèmes qui surviennent et fixer des objectifs pour la journée.
- **Revue de Sprint** : Une revue de sprint a lieu à la fin d'un sprint pour visualiser l'incrément réalisé et ajuster le backlog produit si nécessaire. Lors de la revue de sprint, l'équipe Scrum et les parties prenantes vérifient la qualité du produit.
- **Rétrospective de Sprint** : Une rétrospective de sprint est une réunion à la fin de chaque sprint où l'équipe Scrum utilise les leçons apprises des sprints passés pour améliorer l'organisation et devenir plus efficace.

Product Owner	C'est le représentant du projet. Il traduit la vision du projet à l'équipe, valide les bénéfices dans des histoires à intégrer dans le carnet de commande du produit et les priorise régulièrement.	Madame Wafa Arfaoui
Scrum Master	La personne qui dirige l'équipe en l'aidant à se conformer aux règles et aux processus de la méthodologie. Le Scrum master est chargé de tenir Scrum à jour, de fournir un encadrement, un mentorat et une formation aux équipes au cas où elles en auraient besoin, et gère la réduction des obstacles du projet et travaille avec le Product Owner.	Madame Wissal Gmiza
Scrum Team	Un groupe de professionnels possédant les connaissances techniques nécessaires qui développent le projet en réalisant conjointement les histoires auxquelles ils s'engagent au début de chaque sprint.	Développeur Web : Iheb Bayach et Jawher Tabbabi

TABLE I.2 – Composition de l'équipe Scrum avec les rôles

CHAPITRE I. CONTEXTE GÉNÉRAL

Conclusion :

Ce chapitre présente le contexte général du travail. Nous avons pu identifier les problématiques inhérentes au suivi et au contrôle au sein de la chaîne de production de l'entreprise et visons à présenter les objectifs de la solution proposée. Le chapitre suivant est consacré au lancement du projet et aborde le choix des nouvelles techniques utilisées pour parvenir à la solution souhaitée.

Chapitre II

Analyse et spécification des besoins

Introduction :

La phase d'analyse de projet est une étape essentielle dans tout développement de projet informatique. Elle définit les objectifs du projet, les besoins des utilisateurs et planifie les activités de développement en conséquence. Cette phase comprend différentes étapes telles que la définition de l'architecture du projet, l'identification des acteurs clés et la spécification des exigences fonctionnelles et non fonctionnelles. De plus, il s'agit de créer un backlog produit et de planifier des sprints pour organiser le développement du projet. La phase d'analyse est la base pour assurer le succès du projet et répondre aux attentes des utilisateurs.

1 Capture des besoins :

1.1 Identification des acteurs :

- Admin : Ils supervisent les projets dans leur ensemble et peuvent avoir besoin d'informations sur la progression du travail pour suivre l'avancement des projets,

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

identifier les éventuels retards et prendre des mesures correctives si nécessaire. Ils ont besoin d'accéder aux données sur l'avancement du travail pour prendre des décisions éclairées et assurer la gestion efficace de l'équipe.

- Les designers : Ce sont les principaux acteurs chargés de réaliser les tâches de design et de travailler sur les projets assignés. Ils sont responsables de mettre à jour l'état d'avancement de leurs tâches dans le système.
- Les chefs d'équipe : Les chefs d'équipe sont des designers et en même temps, ils confirment le statut de migration de design.

1.2 Spécification des besoins :

Besoins fonctionnels :

- Gestion des utilisateurs : Permettre à l'admin de gérer les utilisateurs du système, tels que les ajouter dans la base de données avec leurs informations personnelles, les modifier et les supprimer du système.
- Gestion des tâches : Permettre à l'admin de créer, attribuer et gérer les tâches des designers, en spécifiant les détails tels que la description de la tâche, la date d'échéance et les priorités.
- Confirmation de complétion de tâche : Permettre aux moderators de superviser et de confirmer l'achèvement des tâches.
- Suivi de l'avancement : Fournir aux designers un moyen de mettre à jour régulièrement l'état d'avancement de leurs tâches, en indiquant si la tâche est en cours, terminée ou en attente.
- Visualisation des données : Offrir des outils de visualisation conviviaux permettant à l'admin de suivre l'avancement du travail de manière globale et détaillée, en fournissant des tableaux de bord, des graphiques et des rapports personnalisés.

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

Besoins non fonctionnels :

- Convivialité : Assurer une interface utilisateur intuitive et conviviale pour faciliter l'adoption de la solution par les utilisateurs et réduire le besoin de formation.
- Performance : Garantir des performances optimales du système, même lorsqu'il gère un grand volume de données et d'utilisateurs simultanés.
- Sécurité : Mettre en place des mesures de sécurité robustes pour protéger les données sensibles et garantir la confidentialité des informations des utilisateurs.
- Évolutivité : Concevoir une solution évolutive qui peut être facilement adaptée et étendue pour répondre aux besoins futurs de l'entreprise.
- Compatibilité : Assurer la compatibilité avec les différents navigateurs web et les appareils mobiles pour permettre un accès flexible à la solution à tout moment et en tout lieu.
- En capturant ces besoins, nous pouvons établir une base solide pour la conception et le développement de la solution de suivi de la progression du travail des designers, en veillant à ce qu'elle réponde aux attentes et aux exigences des utilisateurs finaux.

En capturant ces besoins, nous pouvons établir une base solide pour la conception et le développement de la solution de suivi de la progression du travail des designers, en veillant à ce qu'elle réponde aux attentes et aux exigences des utilisateurs finaux.

2 Product Backlog :

Le product Backlog, qui est l'unique source des besoins pour tous les changements à effectuer sur le produit, est une liste ordonnée de tout ce qui pourrait être requis dans le produit. Il peut être considéré comme l'ensemble des caractéristiques fonctionnelles qui constituent le produit, également connues sous le nom d'"user stories" dans un souci de cohérence et de clarté, nous conserverons les termes en anglais tout au long de notre

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

analyse et de notre travail sur le product Backlog est déterminé par :

- Identifiant : un numéro unique attribué à chaque tâche.
- Nom : C'est le nom de la tâche à réaliser.
- Description : Besoin du client.
- Critères d'acceptation : comme son nom l'indique, à chaque interrogation du client, il identifie des spécifications et des visions sur le produit. Ces visions peuvent varier en allant loin dans la réalisation du produit.
- Priorité : Est un identifiant qui désigne l'ordre de la réalisation des tâches.
- Complexité : Degré de difficulté déterminé par l'équipe de développement pour pouvoir l'ordonner au sein des sprints.

2.1 Technique de priorisation du Product Backlog (MoSCoW) :

La méthode de MoSCoW permet de prioriser les user stories à moyen terme selon les critères suivants :

M – Must : doit être réalisée.

S – Should have : devrait être réalisée si possible.

C – Could have : pourrait être réalisée s'il n'y a pas d'impact sur les autres tâches en cours.

W – Would have : ne sera pas réalisée tout de suite mais serait souhaitable pour une version ultérieure.

La hiérarchisation des techniques MoSCoW est un bon point de départ, mais cette approche présente certaines limites. Les participants à l'atelier ont expérimenté les expressions de besoin classiques et ont risqué d'être catégorisés comme des histoires "would" ou "should", réelles ou imaginaires, évitant ainsi le piège du "tout compte" qui ne se réalise jamais. Dans cette façon de penser, une mise en œuvre réussie de MoSCoW nécessite un changement de mentalité important et difficile à réaliser. Il est de la responsabilité du

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

Product Owner de promouvoir ces ateliers avec le bon message[2].

2.2 Les fonctionnalités du backlog :

ID Sprint	ID	Fonctionnalité	User Story	Priorité
1	1	S'authentifier	En tant qu'admin, moderator et designer je veux m'authentifier pour accéder à l'application.	M
2	1	Consulter le Dashboard	En tant qu'admin, je veux consulter le Dashboard afin de visualiser la progression des tâches et du projet.	M
3	1	Gérer les utilisateurs	En tant qu'admin, je veux gérer les utilisateurs de système et leurs données (consulter, ajouter, modifier, supprimer).	M
4	1	Fixer les tâches	En tant qu'admin, je veux fixer les tâches pour les utilisateurs de système.	M
	2	confirmer la complétion des tâches	En tant que moderator je veux vérifier le statut de migration de design.	M
	3	Compléter les tâches	En tant qu'admin, moderator et designer je veux Mettre a jours les tâches.	M

TABLE II.1 – Backlog de produit :

2.3 Diagramme de cas d'utilisation global :

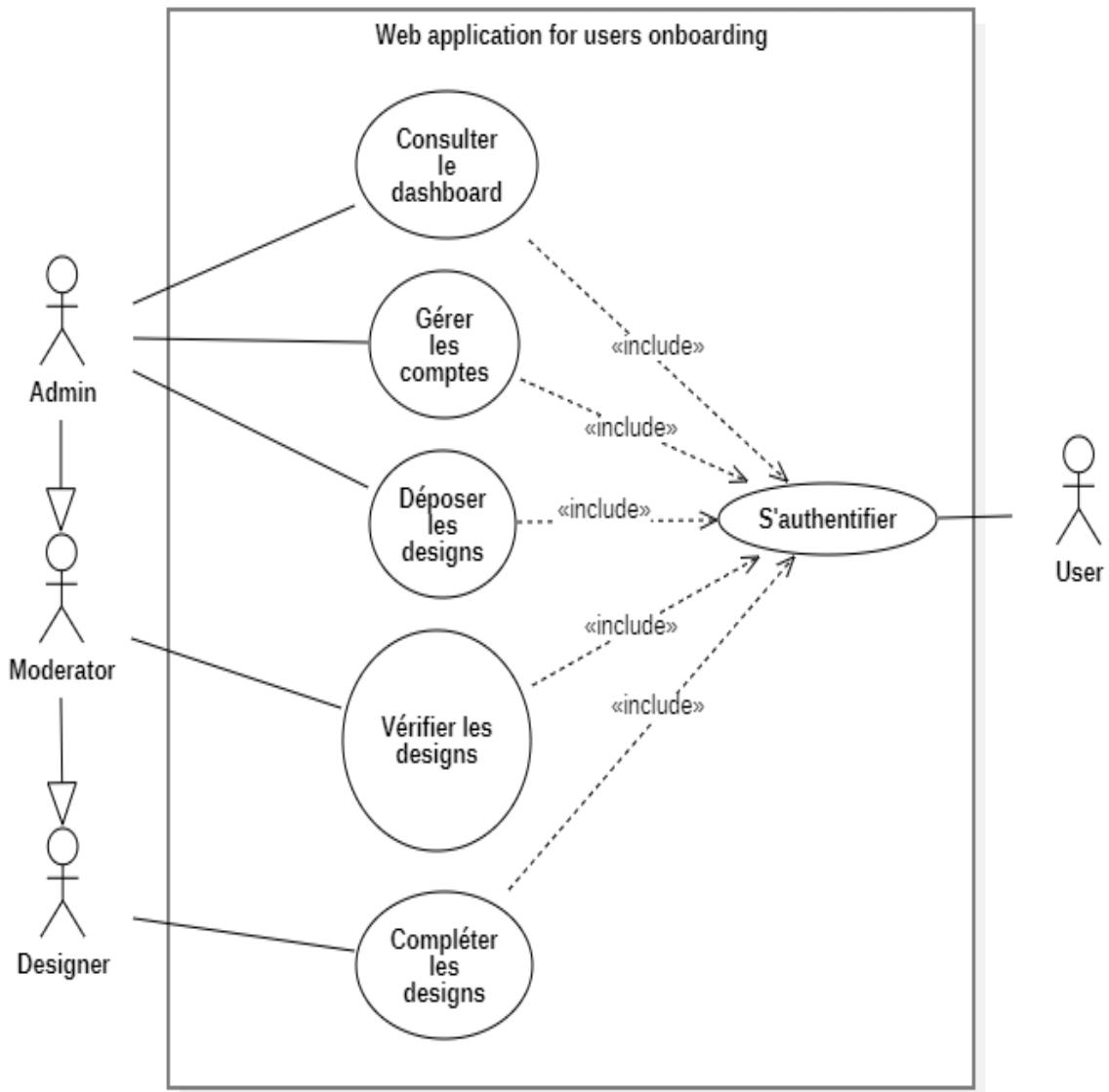


FIGURE II.1 – Diagramme des cas d'utilisation global

2.4 Diagramme de classe globale :

3 Planification des sprints du projet :

Release	Sprint	But de Sprint	durée
1	Sprint 1	S'authentifier	4 Semaines
	Sprint 2	consulter le Dashboard.	2 Semaines
	Sprint 3	Gestion des utilisateurs	2 Semaines
	Sprint 4	Consulter l'interface utilisateurs.	4 Semaines

TABLE II.2 – sprint de projet

4 Environnement de travail :

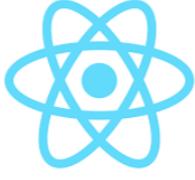
Dans cette section, nous faisons mention du matériel et des outils logiciels nécessaires à l'élaboration de notre système. 2 Ordinateurs pour le développement du code :

- ASUS avec CPU Core i5 10th Gen, 20 Go de RAM, SSD 256 Go, GPU 2 Go.
- HP avec CPU Core i5 10th Gen, 16 Go de RAM, SSD 256 Go.

Ordinateur pour le développement du code

4.1 Environnement technologique :

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

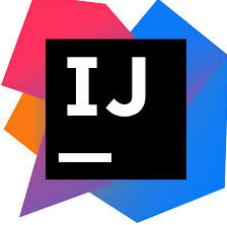
Outil	Définition	Caractéristique
 React js[3]	<p>React est une bibliothèque JavaScript frontale à code source ouvert permettant de créer des interfaces utilisateur. Elle est basée sur le concept de composants réutilisables, ce qui permet aux développeurs de créer des interfaces utilisateur modulaires et efficaces.</p>	<ul style="list-style-type: none"> - React.js est une bibliothèque JavaScript pour la construction d'interfaces utilisateur. Il utilise une architecture basée sur des composants, ce qui permet la réutilisation du code. - React.js est excellent pour rendre efficacement des interfaces utilisateur complexes.
 Spring boot[4]	<p>Spring Boot est un framework open-source basé sur Java qui simplifie le développement d'applications Java. Il fournit un ensemble d'outils et de bibliothèques pour faciliter la création d'applications Java autonomes, prêtes à être déployées, avec un minimum de configuration.</p>	<ul style="list-style-type: none"> - Spring Boot est idéal pour développer des applications de niveau entreprise car il offre un backend puissant et évolutif pour la construction d'API et de microservices. - Il a un support étendu pour diverses sources de données et permet une intégration facile avec d'autres projets, ce qui simplifie la construction d'architectures basées sur des microservices. <p>Le conteneur léger de Spring Boot est idéal pour déployer et exécuter des applications.</p>

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

 MySQL[5]	<p>MySQL est un système de gestion de base de données relationnelle open-source largement utilisé. Il offre un stockage structuré et fiable pour les données dans diverses applications, allant des petites applications web aux grandes entreprises. MySQL est populaire en raison de sa facilité d'utilisation, de sa performance élevée et de sa fiabilité.</p>	<ul style="list-style-type: none">- Langage de requête SQL : MySQL utilise le langage SQL (Structured Query Language) pour interagir avec les données. Les développeurs peuvent utiliser SQL pour effectuer des opérations telles que l'insertion, la mise à jour, la suppression et la récupération de données.- Performance élevée : MySQL est connu pour sa performance élevée, sa rapidité et sa capacité à gérer de grandes quantités de données tout en maintenant des temps de réponse rapides.
---	--	---

TABLE II.3: Environnement technologique

4.2 Environnement logiciel :

Outil	Définition	Caractéristique
 Visual Studio Code[6]	<p>Microsoft Visual Studio est une suite logicielle de développement pour Windows et Mac OS développée par Microsoft. La dernière version s'appelle Visual Studio 2022.</p>	<ul style="list-style-type: none"> - Développement multiplateforme : Visual Studio prend en charge le développement multiplateforme pour une variété de langages de programmation, notamment C++, Python, JavaScript, TypeScript, etc. - Environnement de développement intégré (IDE) : Visual Studio est un IDE complet qui propose une interface utilisateur unifiée pour l'ensemble du processus de développement.
 IntelliJ IDEA Ultimate[7]	<p>IntelliJ IDEA est un environnement de développement intégré (IDE) utilisé principalement pour le développement de logiciels en Java, mais il prend également en charge de nombreux autres langages de programmation.</p>	<ul style="list-style-type: none"> - Productivité améliorée : IntelliJ IDEA offre une gamme de fonctionnalités avancées pour améliorer la productivité des développeurs, y compris une assistance au code, une navigation intelligente, une analyse de code statique, des refactorisations automatisées, etc. - Intégration étroite avec les technologies Java : IntelliJ IDEA offre un support complet pour les technologies Java, y compris les frameworks populaires comme Spring, Hibernate, Maven, etc.

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

 MySQL Workbench[8]	<p>MySQL Workbench est un environnement intégré qui permet aux développeurs de gérer efficacement les bases de données MySQL. Cet outil offre une interface graphique conviviale pour effectuer diverses tâches liées à la base de données.</p>	<ul style="list-style-type: none">- Performances élevées : MySQL est reconnu pour ses performances élevées, sa rapidité et sa capacité à gérer de grandes quantités de données tout en maintenant des temps de réponse rapides.- Fiabilité et stabilité : MySQL est utilisé par de nombreuses entreprises dans des environnements critiques où la disponibilité et l'intégrité des données sont primordiales.
 Postman[9]	<p>Postman est un outil de développement d'API qui permet de concevoir, tester et documenter les API RESTful. Il prend en charge les requêtes HTTP et facilite l'automatisation des tests et la collaboration entre les développeurs.</p>	<ul style="list-style-type: none">- Productivité améliorée : Postman offre une interface intuitive et des fonctionnalités avancées pour améliorer la productivité des développeurs, telles que l'organisation des requêtes en collections, la gestion des environnements, et l'automatisation des tests API.- Intégration étroite avec les technologies d'API : Postman prend en charge divers types de requêtes HTTP et est compatible avec des standards d'API courants, facilitant le travail avec des frameworks et des services RESTful.

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

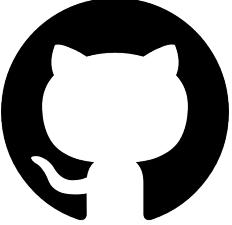
 Git et Github[10]	<p>Git est un système de contrôle de version distribué, et GitHub est une plateforme de développement collaboratif basée sur Git, offrant des fonctionnalités telles que la gestion de projet, le suivi des problèmes et la collaboration entre développeurs.</p>	<ul style="list-style-type: none"> - Git est un système de contrôle de version distribué permettant de suivre les modifications du code, coordonner le travail d'équipe et revenir à des versions antérieures si nécessaire. - GitHub est une plateforme de développement collaboratif basée sur Git, facilitant la gestion de projets, le suivi des problèmes et la collaboration entre développeurs à travers le partage de code source et l'examen des contributions.
 starUML[11]	<p>StarUML est un logiciel de modélisation UML qui permet aux développeurs de créer des diagrammes de classe, des diagrammes de séquence, des diagrammes d'activité et d'autres types de diagrammes utilisés dans le processus de développement logiciel.</p>	<ul style="list-style-type: none"> - Prise en charge de l'UML : StarUML prend en charge les différents types de diagrammes UML, tels que les diagrammes de classe, de séquence, d'activité, de cas d'utilisation, de composants, de déploiement, etc. - Interface conviviale : Il offre une interface utilisateur conviviale avec des outils intuitifs pour créer, modifier et organiser les éléments des diagrammes.

TABLE II.4: Environnement logiciel

5 Architecture générale de l'application :

Spring Boot suit une architecture basée sur le modèle MVC (Modèle-Vue-Contrôleur) qui est couramment utilisée pour construire des services web RESTful. Voici une description de chaque composant de l'architecture MVC (Modèle-Vue-Contrôleur) dans Spring Boot :

- Modèle : Le modèle représente les données et les règles métier de l'application. Il peut interagir avec une base de données pour stocker, récupérer, mettre à jour et supprimer des données[12].
- Vue : La vue est responsable de la présentation des données à l'utilisateur. Dans une application web, la vue est généralement implémentée en tant que pages web[12].
- Contrôleur : Le contrôleur gère les interactions entre le modèle et la vue. Il traite les requêtes HTTP, délègue les tâches au modèle et transmet les données au client via la vue[12].

6 Renforcement de la Sécurité :

Dans le cadre de ce projet, une priorité a été accordée au renforcement de la sécurité afin de garantir la confidentialité des données et l'authentification des utilisateurs. En parallèle de l'architecture MVC, des mesures de sécurité ont été soigneusement mises en œuvre pour répondre à ces exigences fondamentales.

Le support de HTTPS a été rigoureusement configuré au niveau de l'infrastructure serveur. Cette stratégie assure des communications sécurisées entre le client et le serveur, en chiffrant les données échangées. Ainsi, la confidentialité des informations est préservée, réduisant les risques d'interception ou de manipulation des données sensibles[13].

De plus, un système d'authentification basé sur JWT (JSON Web Token) a été intégré pour renforcer la sécurité de l'application. Les JWT sont utilisés pour vérifier l'identité

CHAPITRE II. ANALYSE ET SPÉCIFICATION DES BESOINS

des utilisateurs et garantir l'accès sécurisé aux ressources de l'application, tout en évitant la nécessité de stocker des informations d'authentification côté serveur[14].

Pour renforcer davantage la sécurité, les mots de passe des utilisateurs sont hachés avant d'être stockés dans la base de données. Le hachage des mots de passe rend impossible leur lecture même en cas de compromission de la base de données, offrant ainsi une protection supplémentaire contre les accès non autorisés[15].

Ces mesures renforcées de sécurité sont essentielles pour établir un environnement de confiance pour les utilisateurs de l'application, tout en répondant aux normes de sécurité les plus strictes.

Conclusion :

Ce chapitre a identifié les besoins fonctionnels et non fonctionnels, les backlogs de produits et les parties prenantes. En identifiant le sprint, nous avons établi le temps nécessaire pour le sprint Déploiement du projet. Dans les chapitres suivants, nous allons illustrer la préparation de la première release et établir la partie conception des différents sprints.

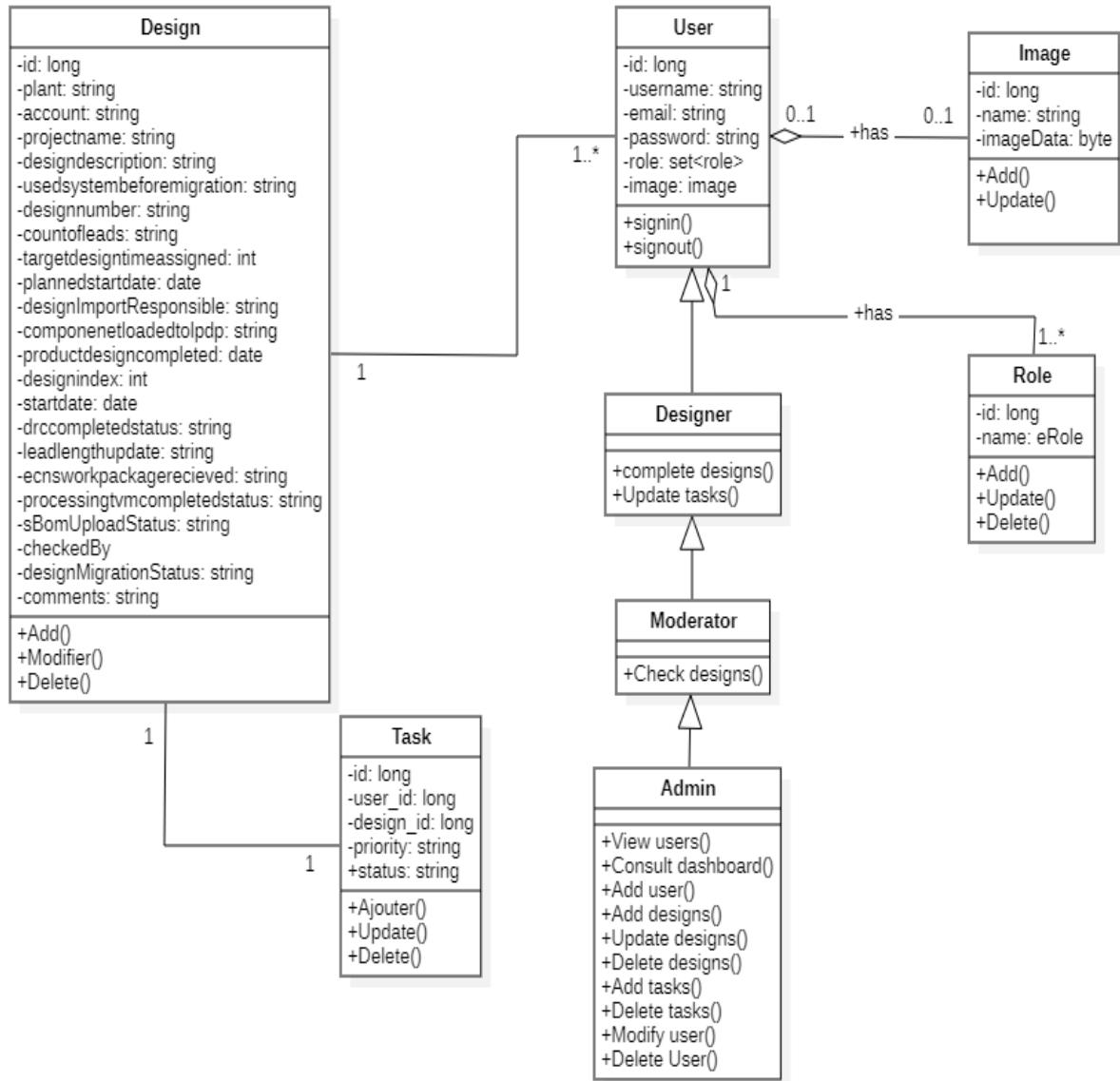


FIGURE II.2 – Diagramme de class global

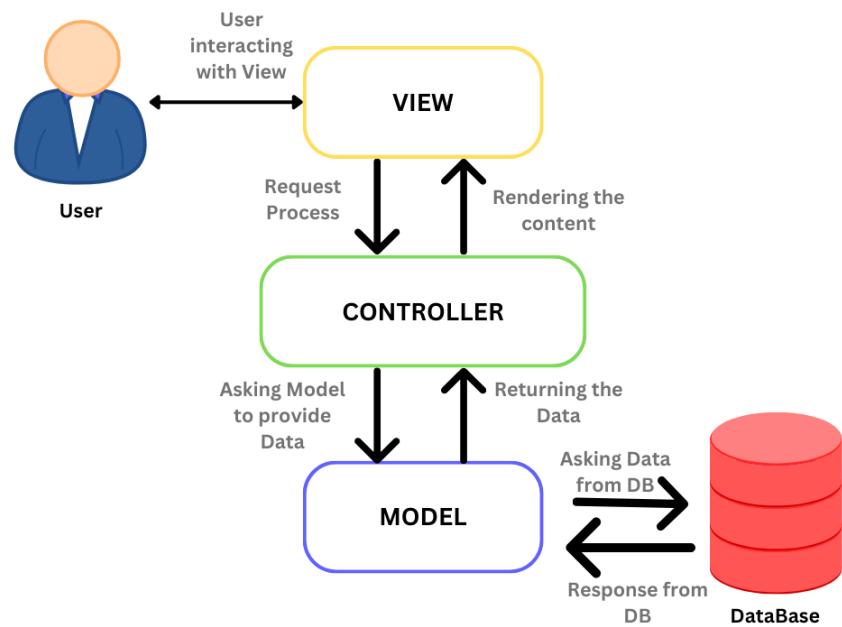


FIGURE II.3 – Architecture MVC

Chapitre III

sprint 1 ”S’authentifier”

Introduction :

Après avoir expliqué la phase initiale du projet et planifié nos sprints dans le chapitre précédent, nous pouvons alors mettre en relief dans ce chapitre le premier sprint qui se compose de la partie authentification.

1 Backlog de sprint 1 :

ID Sprint	ID	Fonctionnalité	User Story	Priorité
1	1	S’authentifier	En tant qu’admin, moderator et designer je veux m’authentifier pour accéder à l’application.	M

TABLE III.1 – Backlog sprint 1

2 Spécifications fonctionnelle :

Cette partie présente les spécifications fonctionnelles du premier sprint dans le but d'interpréter les résultats souhaités en termes de fonctionnalité et de mieux décrire le fonctionnement de chacune des tâches.

2.1 Diagramme de cas d'utilisation :

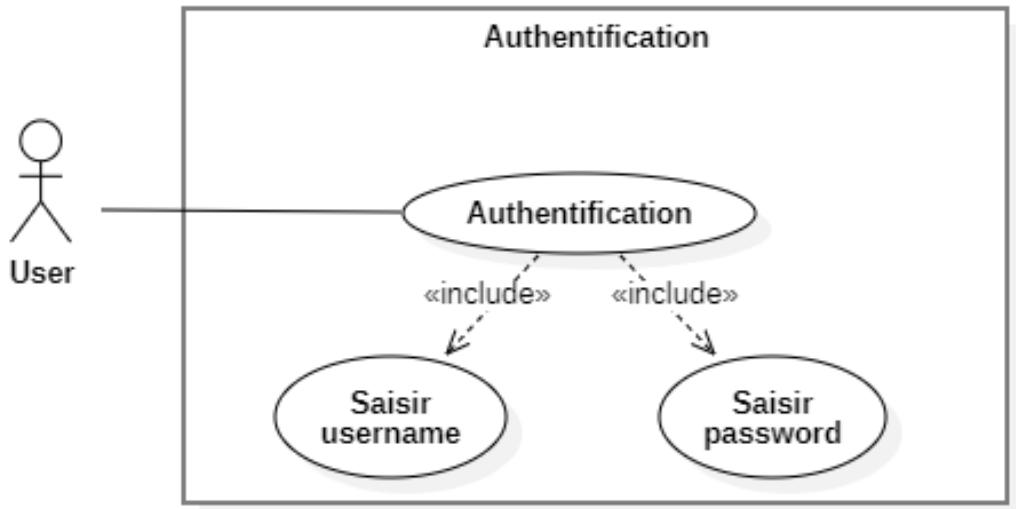


FIGURE III.1 – Diagramme de cas d'utilisation de l'Authentification

CHAPITRE III. SPRINT 1 "S'AUTHENTIFIER"

2.2 Description textuelle des cas d'utilisations :

Raffinement de cas d'utilisation "S'authentifier" :

Cas d'utilisation :	S'authentifier.
Acteurs :	admin, Moderator, designer.
Description :	Lorsque l'utilisateur souhaite accéder à l'application, il doit remplir les champs "username" et "password". Ensuite, le système vérifie s'ils sont corrects pour accorder l'accès à l'application selon son rôle.
Pré-condition :	Compte existant.
Post-condition :	Autorisation d'accès accordée.
Scénario nominal :	<ol style="list-style-type: none">1- L'utilisateur saisit le "username" et le "password".2- L'utilisateur clique sur "Login".3- Le système vérifie les données saisies.4- Si l'utilisateur est identifié, l'application ouvre la page spécifique.5- Le système charge les privilèges et les données du compte courant.
Scénario alternatif :	<p>A1 : Si les données d'authentification sont incorrectes, la page de login est rechargée avec un message d'erreur demandant de saisir un "username" et un "password" valides.</p> <p>A2 : Si un champ est vide, l'application affiche un message demandant de remplir tous les champs de saisie.</p>

TABLE III.2 – Description textuelle du cas d'utilisation "S'authentifier"

3 Diagramme des séquences détaillés :

Les diagrammes de séquence ont pour but de détailler le déroulement des scénarios en présentant les interactions entre les différents acteurs du système, tels que l'administrateur, les interfaces de l'application et la base de données.

3.1 Diagramme des séquences "S'authentifier" :

Dans cette partie, nous allons présenter le processus d'authentification par son diagramme de séquence. Pour faire l'authentification, le système affiche le formulaire où l'employeur peut saisir son username et password. Ensuite, les données saisies seront envoyées vers la base de données afin de les vérifier. En parallèle, le système génère un token JWT (JSON Web Token) qui sera utilisé pour authentifier l'utilisateur par la suite. Une fois les données vérifiées, le résultat est renvoyé à l'application, accompagné du token JWT.

Si les données sont correctes, l'admin aura alors l'accès à l'application et il sera dirigé vers la page Spécifique selon son rôle, tout en conservant le JWT pour les requêtes ultérieures nécessitant une authentification. En cas d'erreur dans les données saisies, la page de username sera rechargée avec un message d'erreur demandant de saisir un username et password valides.

Ce processus d'authentification basé sur JWT permet une gestion sécurisée et efficace des sessions utilisateur, en évitant le stockage de données sensibles côté serveur et en offrant une méthode robuste pour vérifier l'identité de l'utilisateur lors de chaque requête subséquente.

CHAPITRE III. SPRINT 1 "S'AUTHENTIFIEUR"

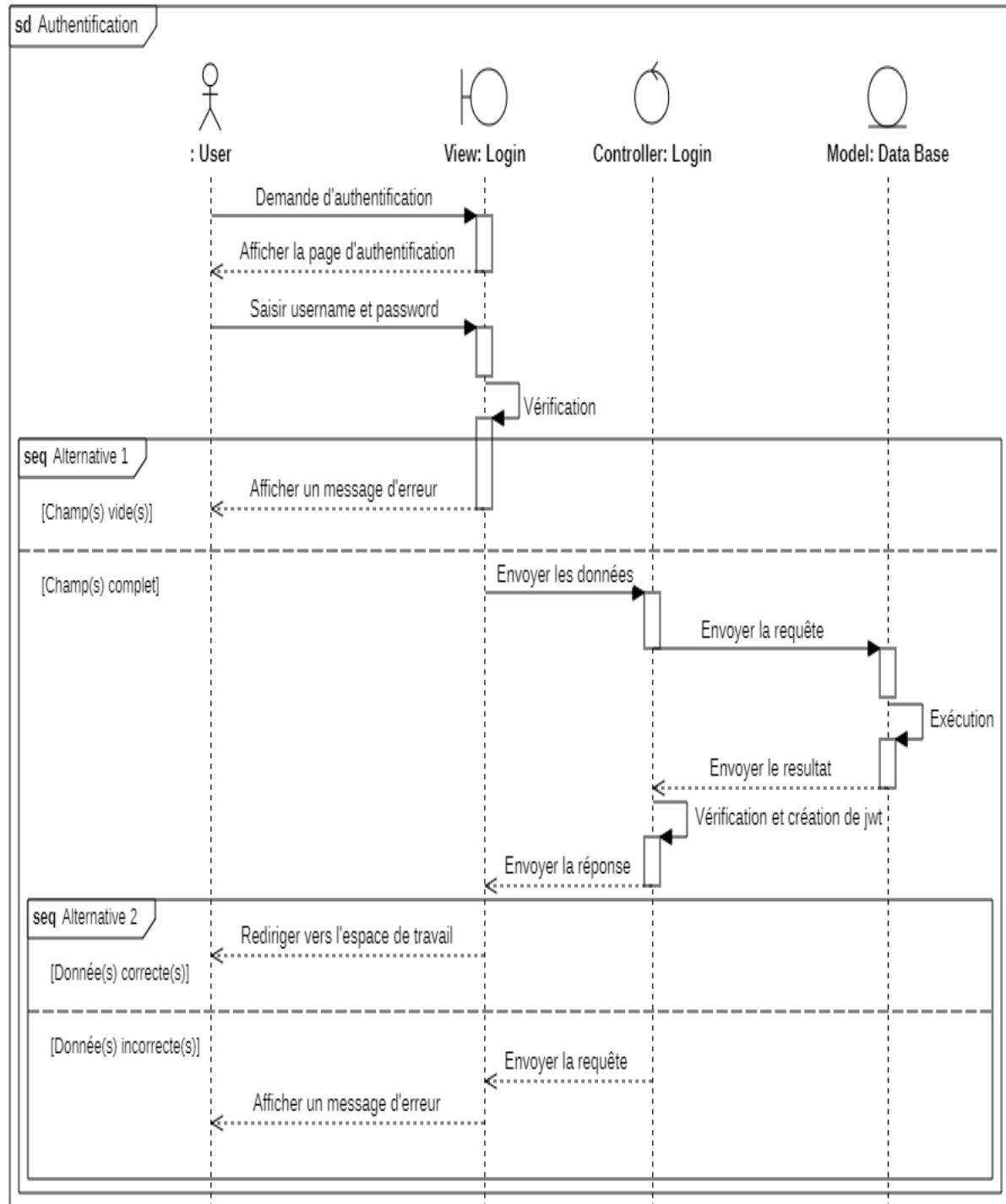


FIGURE III.2 – Diagramme de séquence de l'authentification

4 Diagramme de classe :

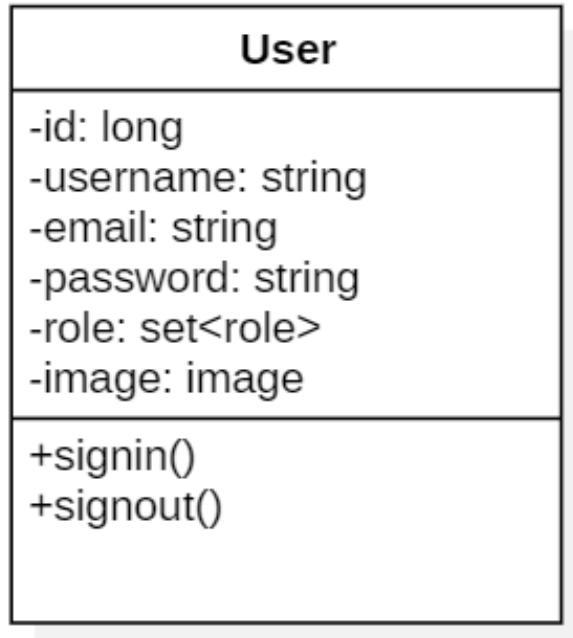


FIGURE III.3 – Diagramme de classe de sprint 1

5 Réalisation :

5.1 Interface d'authentification :

Nous présentons ici l'interface d'authentification : L'utilisateur doit remplir le formulaire d'authentification en passant son username et password.

CHAPITRE III. SPRINT 1 "S'AUTHENTIFIER"

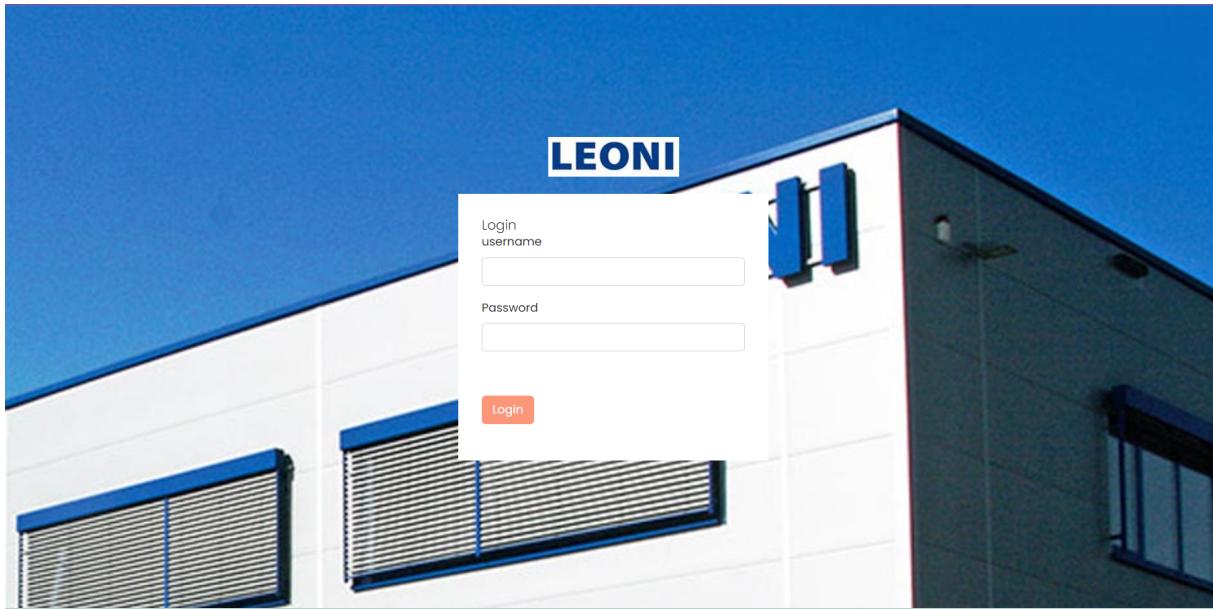


FIGURE III.4 – interface Login

S'il y a un champ vide la page affiche un message qui demande de remplir le champ vide :

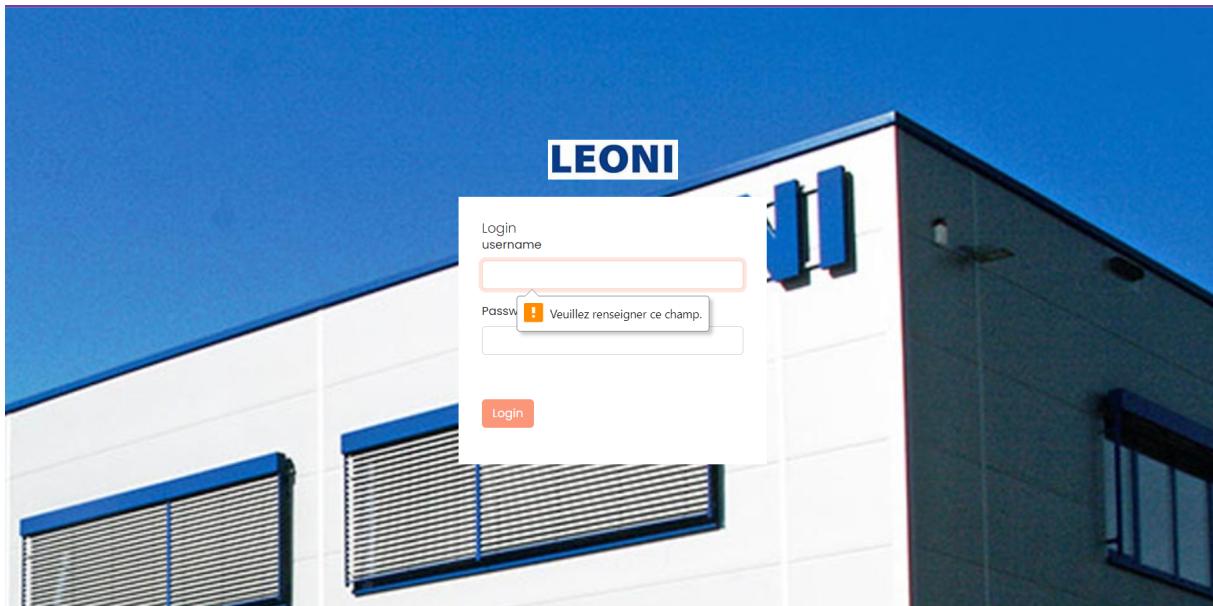


FIGURE III.5 – Interface login avec un message d'erreur de remplir tous les champs

CHAPITRE III. SPRINT 1 "S'AUTHENTIFIEUR"

S'il y a des données incorrectes la page sera réaffichée avec un message d'erreur :

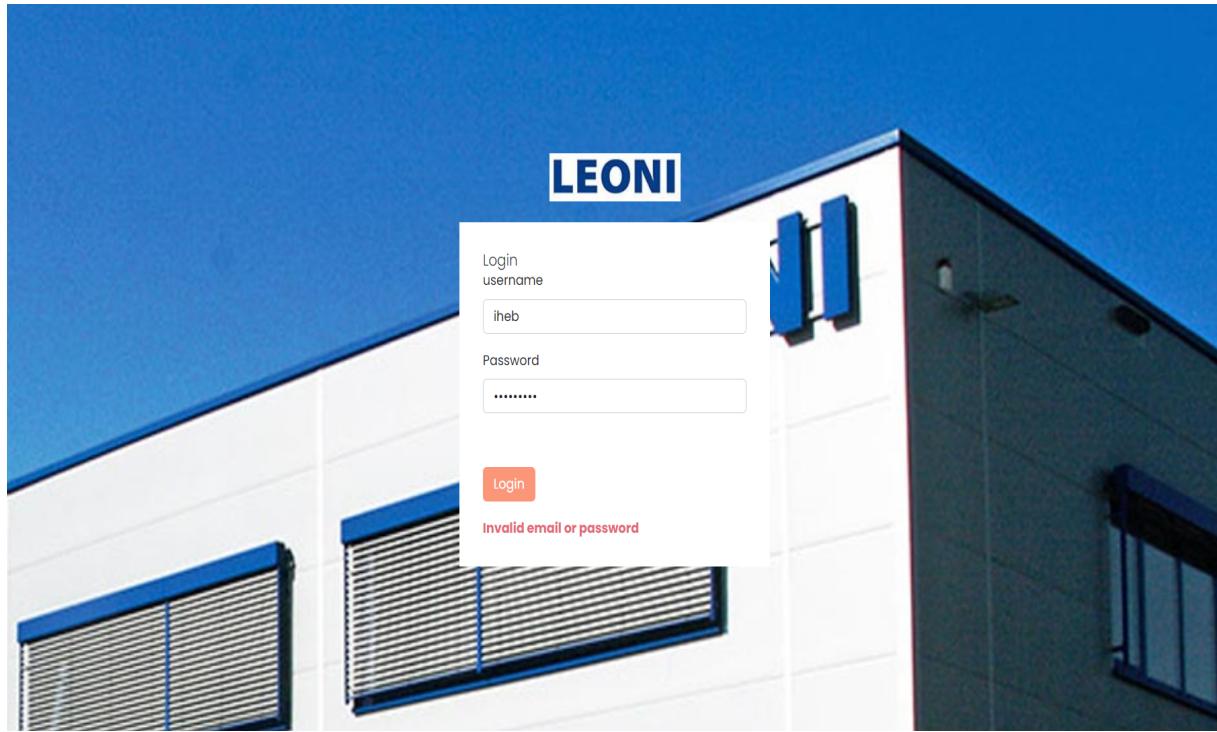


FIGURE III.6 – Interface login avec un message d'erreur

Conclusion :

Dans ce chapitre, nous avons examiné et réalisé le premier sprint, qui comporte l'étape d'authentification, en décrivant le cycle de développement par la conception, et nous avons conclu avec la réalisation de notre sprint en présentant les différentes interfaces. Dans le chapitre suivant, nous allons détailler les besoins du deuxième sprint de notre projet.

Chapitre IV

sprint 2 "Consulter le Dashboard"

Introduction :

Dans ce chapitre, nous allons détailler les besoins du deuxième sprint de notre projet.

Ce sprint se concentre sur la consultation du dashboard, qui présente diverses statistiques essentielles pour le suivi de l'activité des designs et des designers.

1 Backlog de sprint 2 :

ID Sprint	ID	Fonctionnalité	User Story	Priorité
2	1	Consulter le Dashboard	En tant qu'admin, je veux consulter le Dashboard afin de visualiser les statistiques et la progression des designers et des designs.	M

TABLE IV.1 – Backlog du Sprint 2

2 Spécifications fonctionnelles :

Cette section présente les spécifications fonctionnelles du deuxième sprint, afin de décrire les résultats souhaités en termes de fonctionnalités et de détailler le fonctionnement

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

de chaque tâche.

2.1 Diagramme de cas d'utilisation :

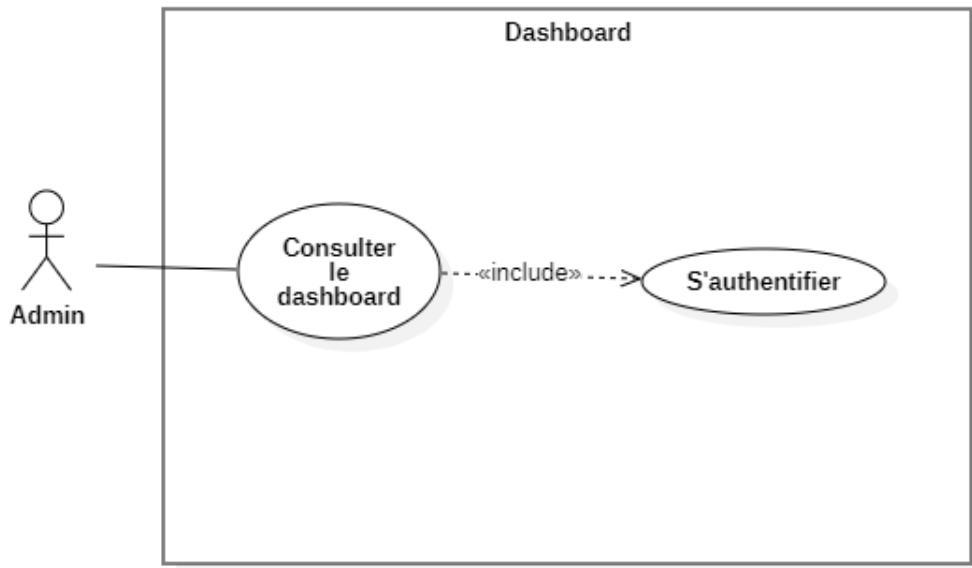


FIGURE IV.1 – Diagramme de cas d'utilisation : Consultation du dashboard

2.2 Description textuelle des cas d'utilisation :

Raffinement des cas d'utilisation pour la consultation du dashboard

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

Cas d'utilisation :	Consulter le dashboard
Acteur :	admin, moderator, designer
Description :	L'admin souhaite consulter le dashboard pour accéder à différentes statistiques sur les designs, les employés, et les performances par marque de voiture.
Pré-condition :	L'admin doit être authentifié.
Post-condition :	Les statistiques sont affichées sur le dashboard.
Scénario nominal :	1- L'admin accède au dashboard. 2- Le système récupère les données concernant les designers et les designs. 3- Le système calcule les statistiques et les affiche sur le dashboard.
Scénario alternatif :	A1 : Si les données nécessaires pour le calcul des statistiques ne sont pas disponibles, le système affiche des statistiques vides.

TABLE IV.2 – Description textuelle du cas d'utilisation "Consulter le dashboard"

3 Diagramme de séquence détaillé :

Les diagrammes de séquence détaillent le déroulement du scénario en présentant les interactions entre les différents acteurs du système et les composants internes.

3.1 Diagramme de séquence "Consulter le dashboard" :

Cette partie présente le processus de consultation du dashboard par son diagramme de séquence. Le système affiche le dashboard où l'admin peut voir les différentes statistiques. Ensuite, les données sont récupérées, calculées et affichées de manière appropriée.

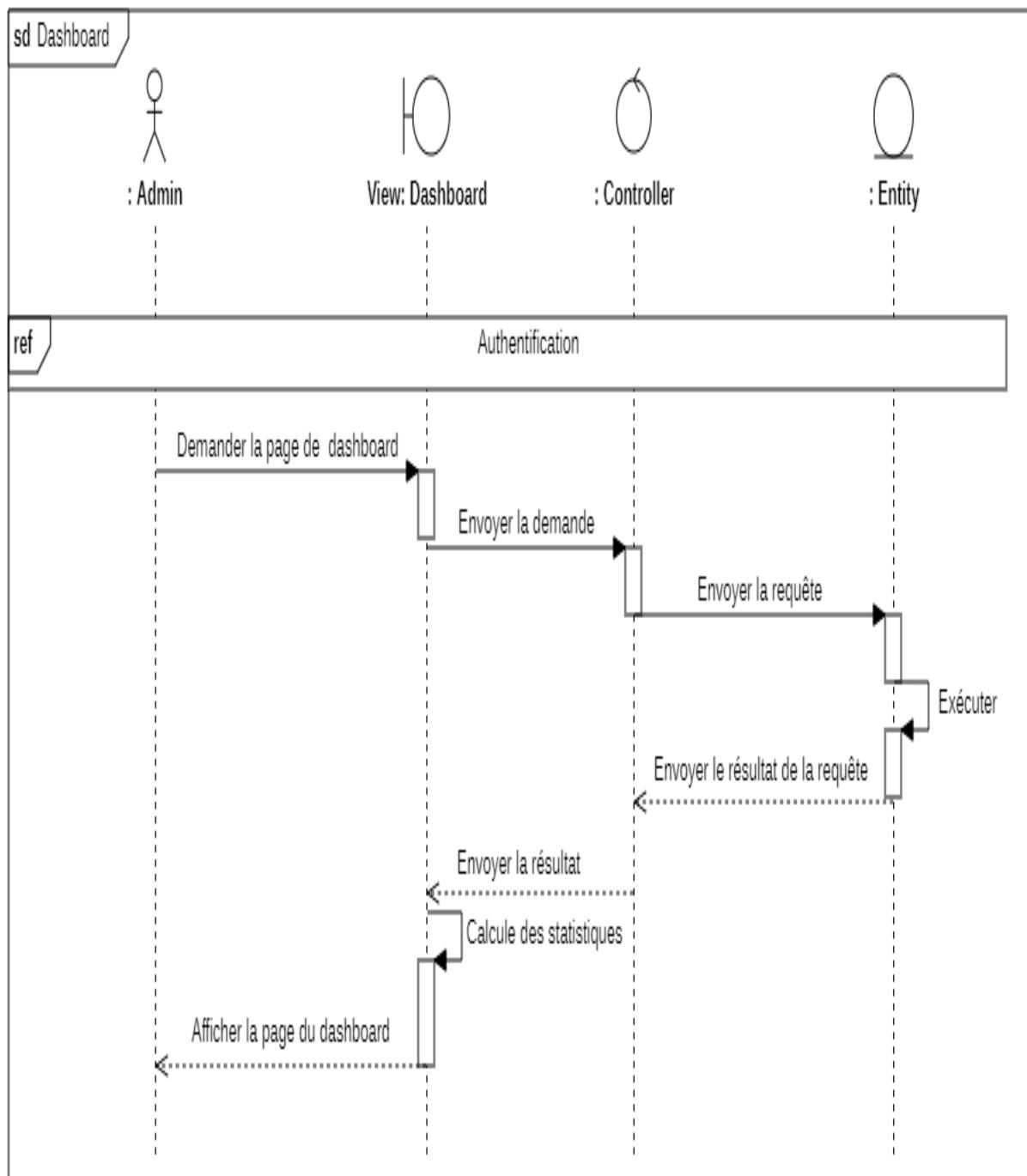


FIGURE IV.2 – Diagramme de séquence de la consultation du dashboard

4 Diagramme de classe :

5 Réalisation :

5.1 Interface du dashboard :

40

Nous présentons ici l'interface du dashboard : l'admin peut consulter diverses statistiques sur le nombre d'employés, le nombre de designs terminés et en cours, ainsi que les

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

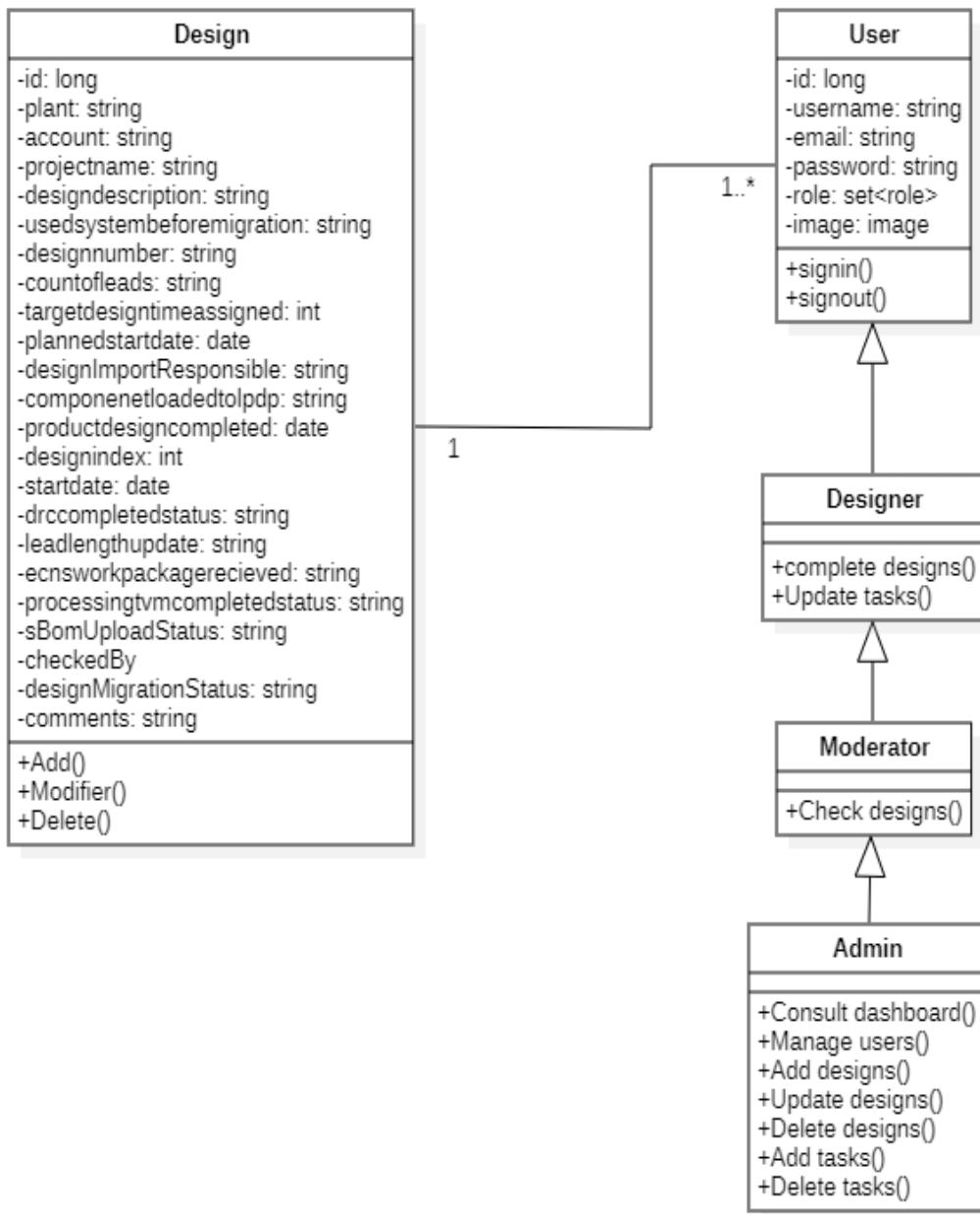


FIGURE IV.3 – Diagramme de classe du Sprint 2

nouveaux designs. Des statistiques supplémentaires incluent le nombre de designs terminés par marque de voiture et le nombre de designs effectués par chaque designer , y compris un classement des meilleurs 5 designers.

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

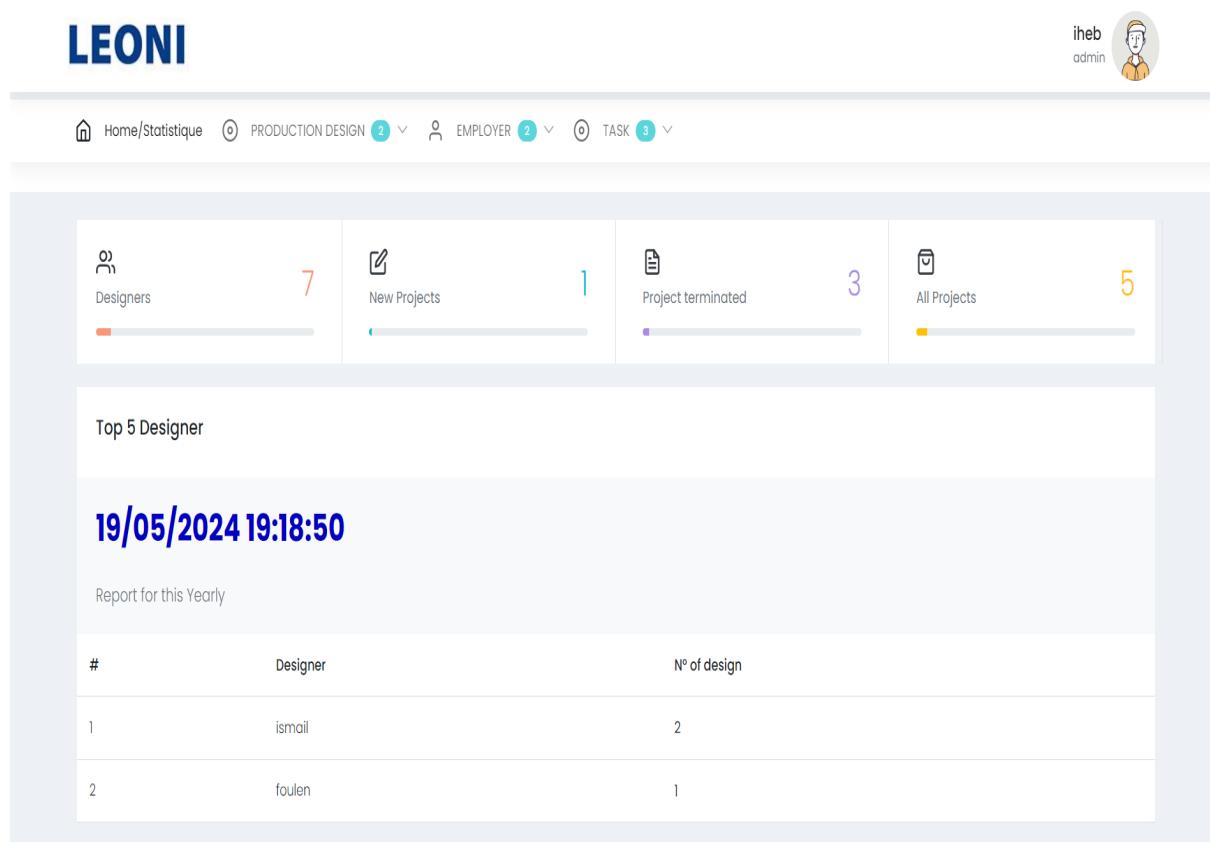


FIGURE IV.4 – Interface du dashboard 1

Conclusion :

Dans ce chapitre, nous avons détaillé et réalisé le deuxième sprint, qui inclut la consultation du dashboard et les différentes statistiques associées. Nous avons décrit le cycle de développement par la conception et la réalisation du sprint, en présentant les différentes interfaces. Dans le chapitre suivant, nous aborderons le troisième sprint de notre projet.

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

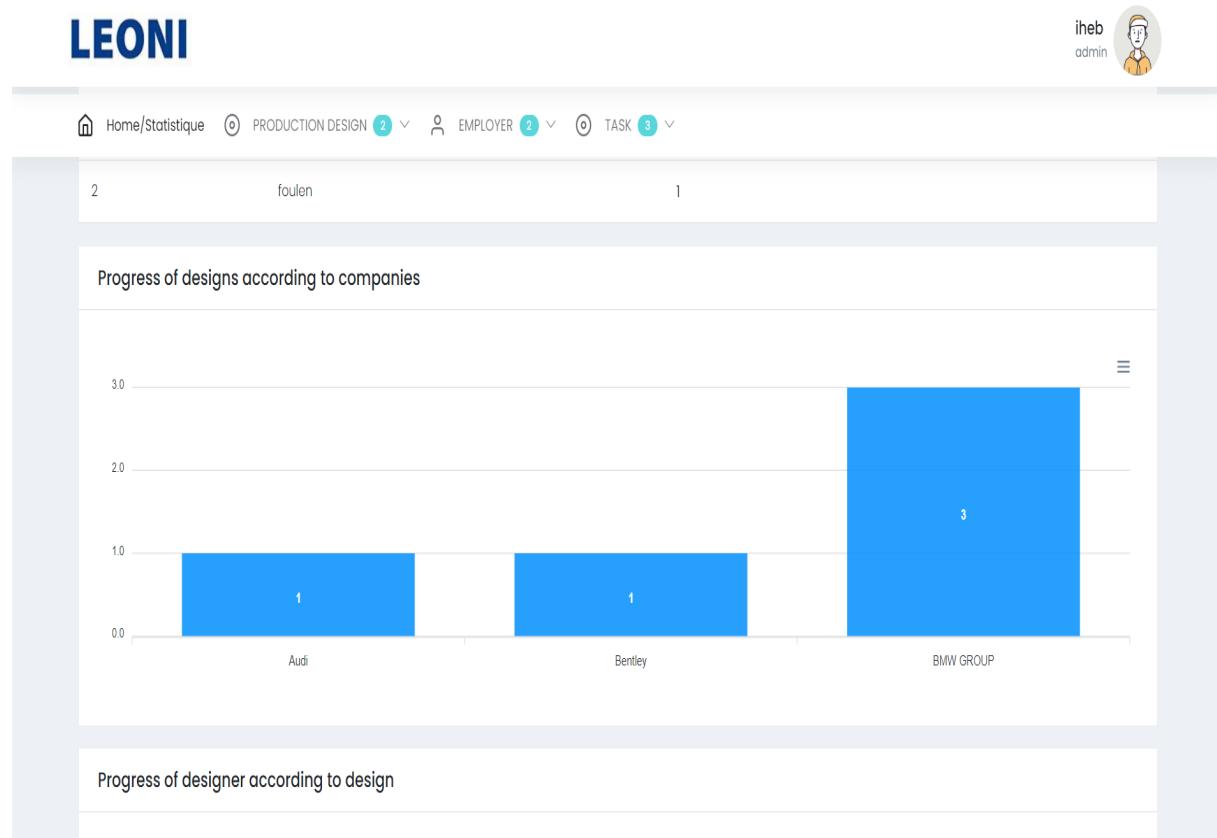


FIGURE IV.5 – Interface du dashboard 2

CHAPITRE IV. SPRINT 2 "CONSULTER LE DASHBOARD"

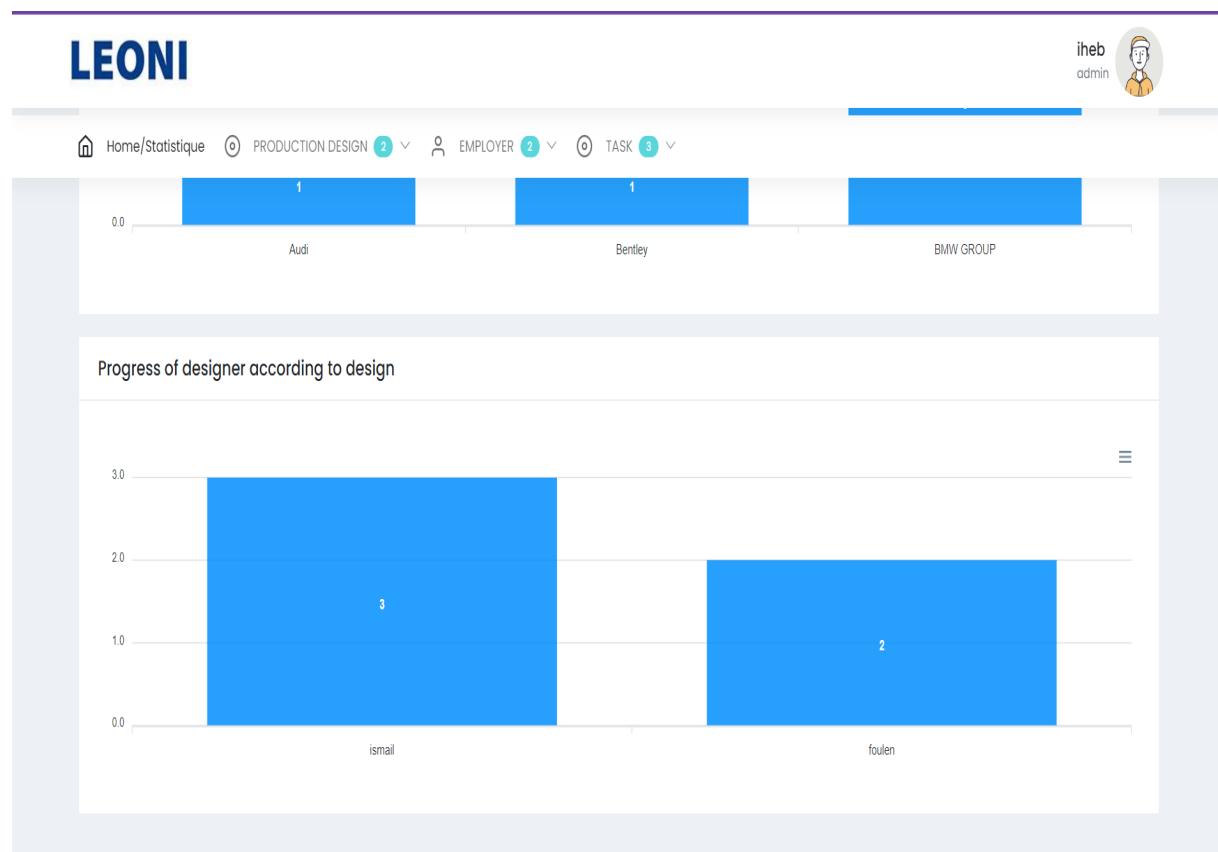


FIGURE IV.6 – Interface du dashboard 3

Chapitre V

sprint 3 "Gérer utilisateurs"

Introduction :

Après Avoir présenté la partie dashboard dans le sprint précédent nous allons maintenant dans ce chapitre traité et exposé le deuxième sprint qui comporte les parties de gestion des utilisateurs.

1 Backlog de sprint 3 :

ID Sprint	ID	Fonctionnalité	User Story	Priorité
3	1	Gérer les utilisateurs	En tant qu'admin, je veux gérer les utilisateurs de système et leurs données (consulter, ajouter, modifier, supprimer).	M

TABLE V.1 – Backlog de sprint 3

2 Spécifications fonctionnelles :

Cette partie est pour présenter les spécifications fonctionnelles du troisième sprint dans le but d'interpréter les résultats souhaités en termes de fonctionnalité et de mieux décrire

le fonctionnement de chacune des tâches.

2.1 Diagramme de cas d'utilisation :

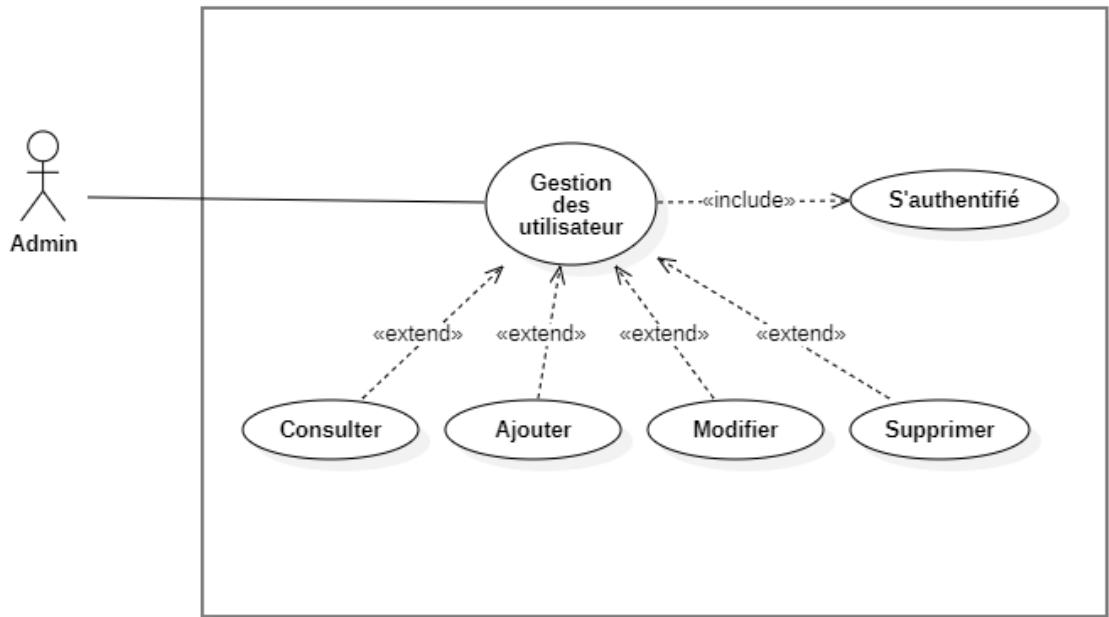


FIGURE V.1 – diagramme de cas d'utilisation sprint 3

CHAPITRE V. SPRINT 3 "GÉRER UTILISATEURS"

2.2 Description textuelle des cas d'utilisation :

Raffinement des cas d'utilisation pour la consultation et la suppression des utilisateurs :

Cas d'utilisation :	Consulter et supprimer un utilisateur de système.
Acteur :	admin
Description :	L'admin peut supprimer des utilisateurs de système.
Pré-condition :	L'admin doit être authentifié.
Post-condition :	L'utilisateur est supprimé.
Scénario nominal :	1- L'admin peut supprimer un utilisateur en cliquant sur le bouton déroulant de la barre de navigation d'en-tête "Employer". 2-La page de tous les utilisateurs de système s'est ouverte. 3-La page des utilisateurs s'affiche 4- L'admin clique sur delete pour supprimer un utilisateur.
Scénario alternatif :	A1 : l'admin annule son choix.

TABLE V.2 – Description textuelle du cas d'utilisation "Consulter et supprimer des utilisateurs"

CHAPITRE V. SPRINT 3 "GÉRER UTILISATEURS"

Cas d'utilisation :	Ajouter un utilisateur de système.
Acteur :	admin
Description :	L'admin peut ajouter des utilisateurs de système.
Pré-condition :	L'admin doit être authentifié.
Post-condition :	L'utilisateur est ajouté.
Scénario nominal :	1- L'admin peut ajouter un utilisateur en cliquant sur le bouton déroulant de la barre de navigation d'en-tête "Employer" et puis choisit "Add". 2- Le formulaire d'ajout s'affiche 3- L'admin remplit le formulaire. 4- L'admin clique sur "Add" pour enregistrer l'utilisateur.
Scénario alternatif :	A1 : Si le username ou l'email existe déjà. Message d'erreur s'affiche.

TABLE V.3 – Description textuelle du cas d'utilisation " Ajouter un utilisateur"

CHAPITRE V. SPRINT 3 "GÉRER UTILISATEURS"

Cas d'utilisation :	Consulter et modifier un utilisateur de système.
Acteur :	admin
Description :	L'admin peut Modifier les données des utilisateurs de système.
Pré-condition :	L'admin doit être authentifié.
Post-condition :	L'utilisateur est modifié.
Scénario nominal :	1- L'admin peut modifier un utilisateur en cliquant sur le bouton déroulant de la barre de navigation d'en-tête "Employer" et puis clique sur "List". 2-La page de l'utilisateur s'ouvre avec un formulaire pour modifier les données. 3- L'admin remplir les champs à modifiés. 4- L'admin clique sur "Update" pour enregistrer l'utilisateur. 5- Un message sera affiché pour informer que les données de l'utilisateur sont modifiées.
Scénario alternatif :	A1 : Si le username ou l'email existe déjà. Message d'erreur s'affiche. A2 : l'admin annule son choix.

TABLE V.4 – Description textuelle du cas d'utilisation ” Consulter et modifier un utilisateur”

3 Diagramme de séquence détaillé :

Dans cette section, nous allons présenter les diagrammes de séquence pour les cas d'utilisation de ce sprint, afin de mieux détailler les différentes fonctionnalités.

3.1 Diagramme de séquence "Gestion des utilisateurs" :

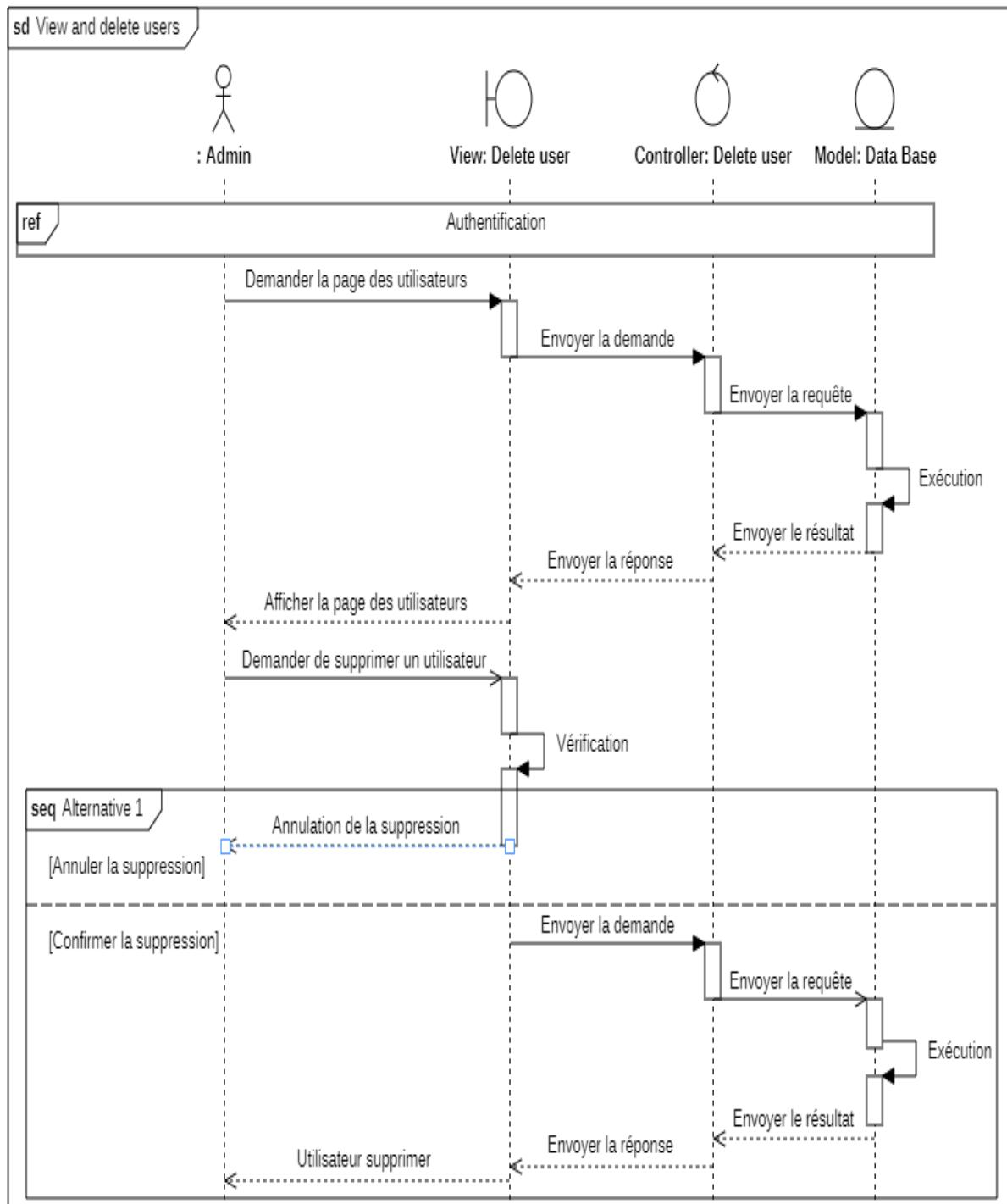


FIGURE V.2 – Diagramme de séquence de consultation et suppression des utilisateurs

CHAPITRE V. SPRINT 3 "GÉRER UTILISATEURS"

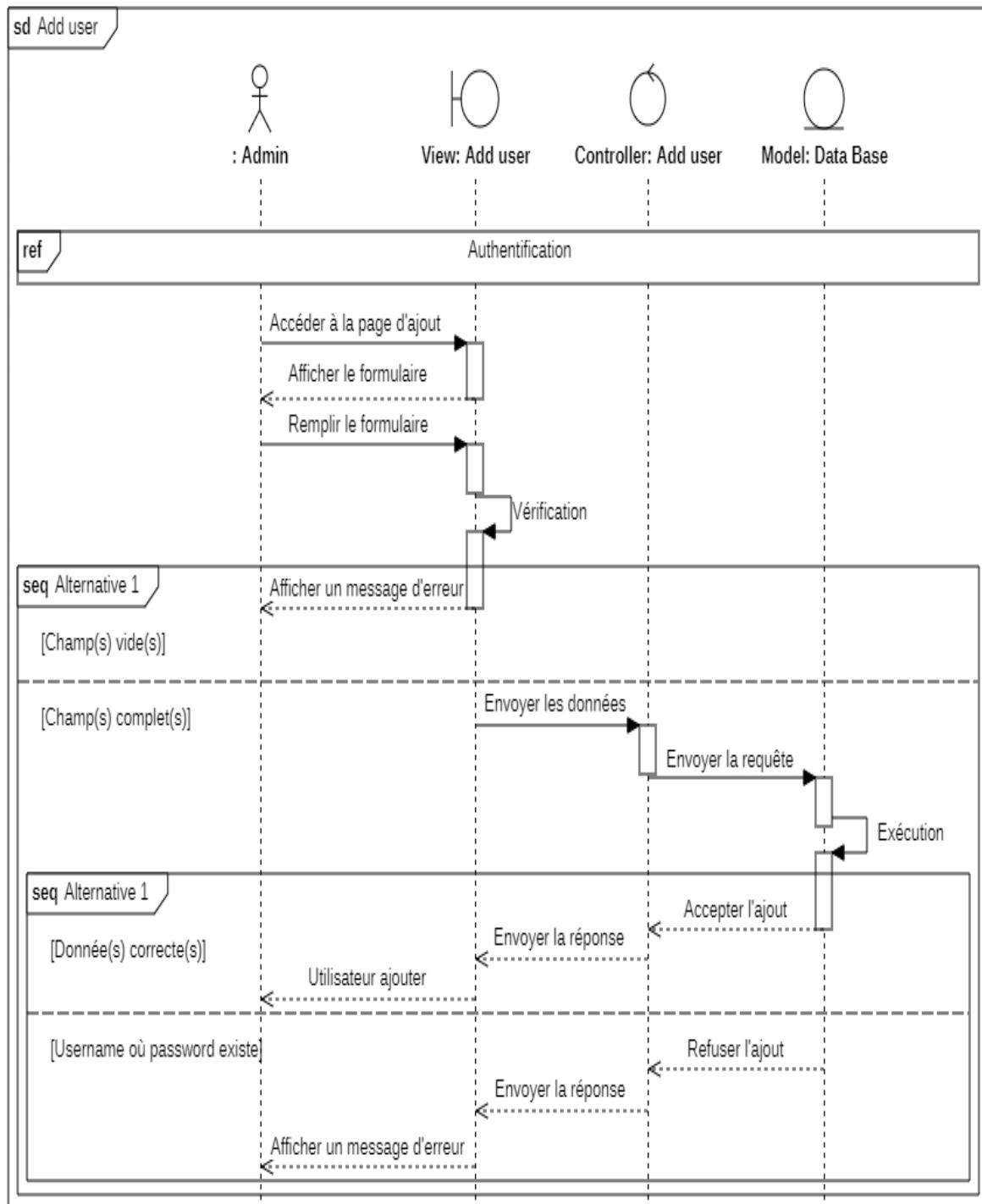


FIGURE V.3 – Diagramme de séquence d'ajout d'un utilisateur

4 Diagramme de classe :

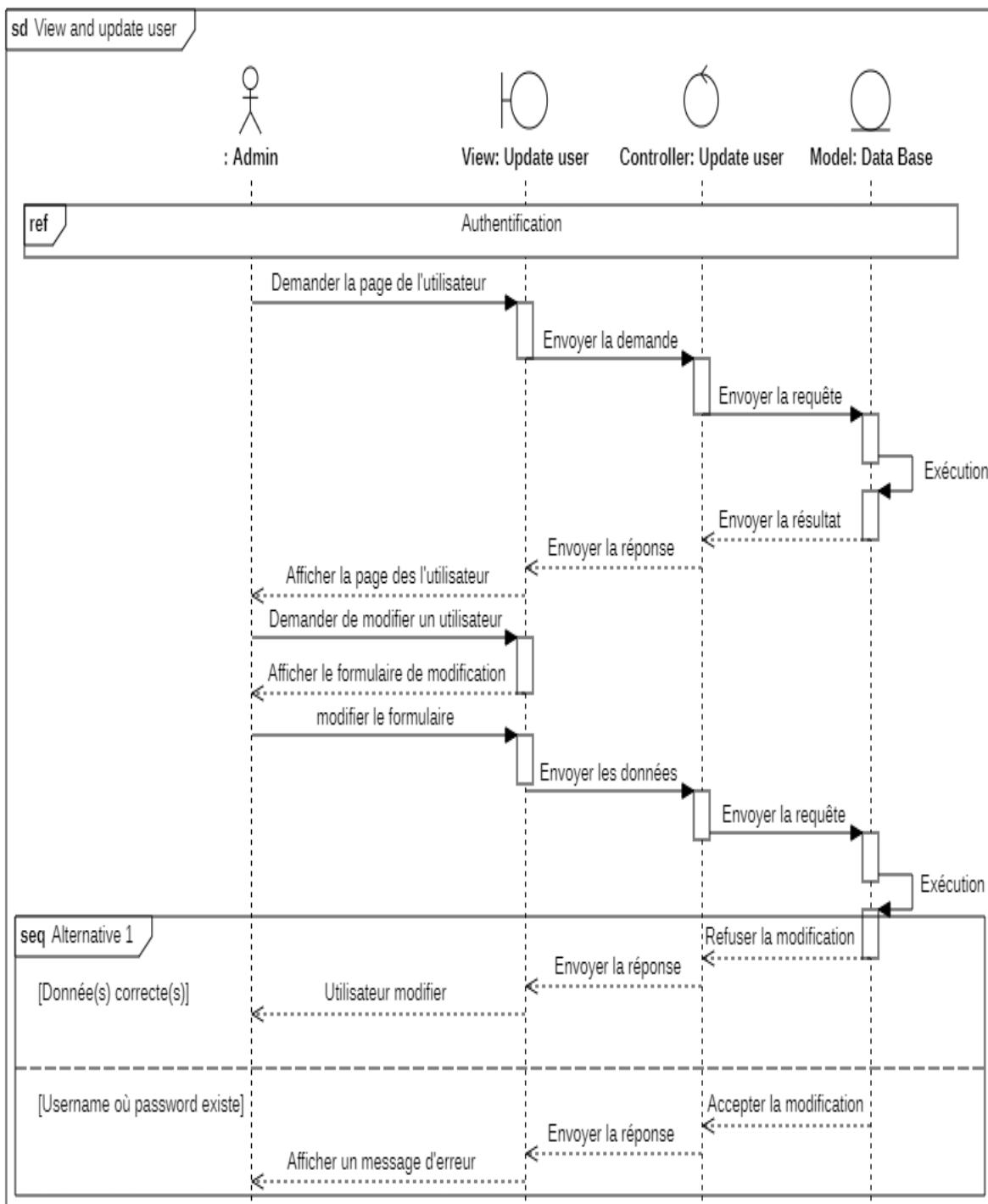


FIGURE V.4 – Diagramme de séquence de modification d'un utilisateur

5 Réalisation :

5.1 Interface de gestion des utilisateurs :

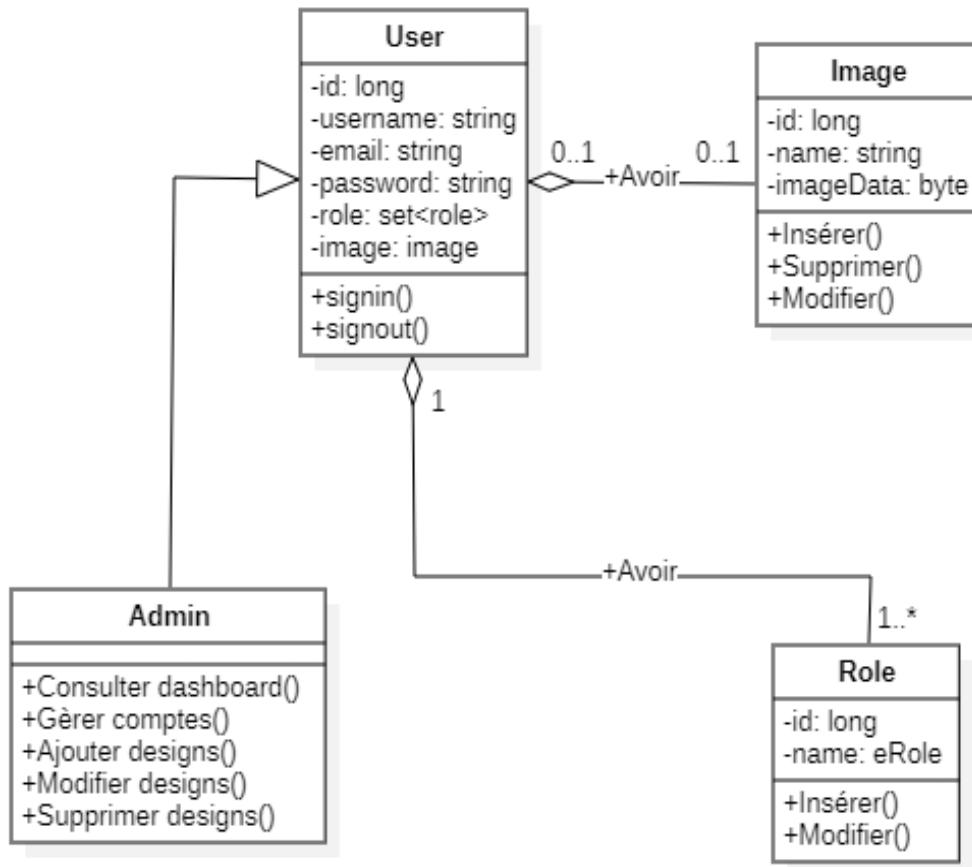


FIGURE V.5 – Diagramme de classe sprint 3

Conclusion :

Dans ce chapitre, nous avons examiné et présenté les différentes fonctionnalités du sprint 3, qui incluent la gestion des utilisateurs du système. Nous avons conclu par la réalisation de ces fonctionnalités. Dans le sprint suivant, nous aborderons la gestion des designs.

CHAPITRE V. SPRINT 3 "GÉRER UTILISATEURS"

The screenshot shows a web-based application interface for managing users. At the top, there is a header with the LEONI logo on the left and a user profile icon on the right. The user profile is labeled "iheb" and "admin". Below the header, there is a navigation bar with links: "Home/Statistique", "PRODUCTION DESIGN", "EMPLOYER", and "TASK".

The main content area is titled "List of Employer" and displays a table with the following data:

ID	User name	E-mail	Role	Image	Actions
28	iheb	iheb@leoni.com	admin		<button>Update</button> <button>Delete</button>
29	jawhertabbabi	jawhertabbabi@leoni.com	moderator		<button>Update</button> <button>Delete</button>
30	aziz	aziz@leoni.com	user		<button>Update</button> <button>Delete</button>
31	ismail	ismail@leoni.com	user		<button>Update</button> <button>Delete</button>
32	wafa	wafa@leoni.com	user		<button>Update</button> <button>Delete</button>

FIGURE V.6 – Interface de gestion des utilisateurs

The screenshot shows a modal dialog box titled "Update user". The form contains the following fields:

- Username: aziz
- Email: aziz@leoni.com
- Roles: A dropdown menu labeled "Select Role".
- Password: A field with a lock icon.
- Confirm Password: A field with a lock icon.

At the bottom of the form are two buttons: "Update user" (green) and "Cancel" (black).

FIGURE V.7 – Interface de gestion des utilisateurs

Chapitre VI

Sprint 4 "Gérer les designs"

Introduction :

Dans ce chapitre, nous allons détailler les besoins du quatrième sprint de notre projet. Ce sprint se concentre sur la gestion des tâches, incluant l'ajout, la modification, la suppression, la consultation et l'affectation des tâches par les admins, ainsi que la consultation, le travail et la validation des tâches par les moderateurs et les designers.

1 Backlog de sprint 4 :

ID Sprint	ID	Fonctionnalité	User Story	Priorité
4	1	Déposer les designs	En tant qu'admin, je veux déposer les designs pour les utilisateurs de système.	M
	2	Vérifier les designs	En tant que admin et moderator je veux vérifier le statut de migration de design.	M
	3	Compléter les designs	En tant qu'admin, moderator et designer je veux mettre à jour les designs.	M

TABLE VI.1 – Backlog de sprint 4 :

2 Spécifications fonctionnelles :

Cette section présente les spécifications fonctionnelles du quatrième sprint, afin de décrire les résultats souhaités en termes de fonctionnalités et de détailler le fonctionnement de chaque designs.

2.1 Diagramme de cas d'utilisation :

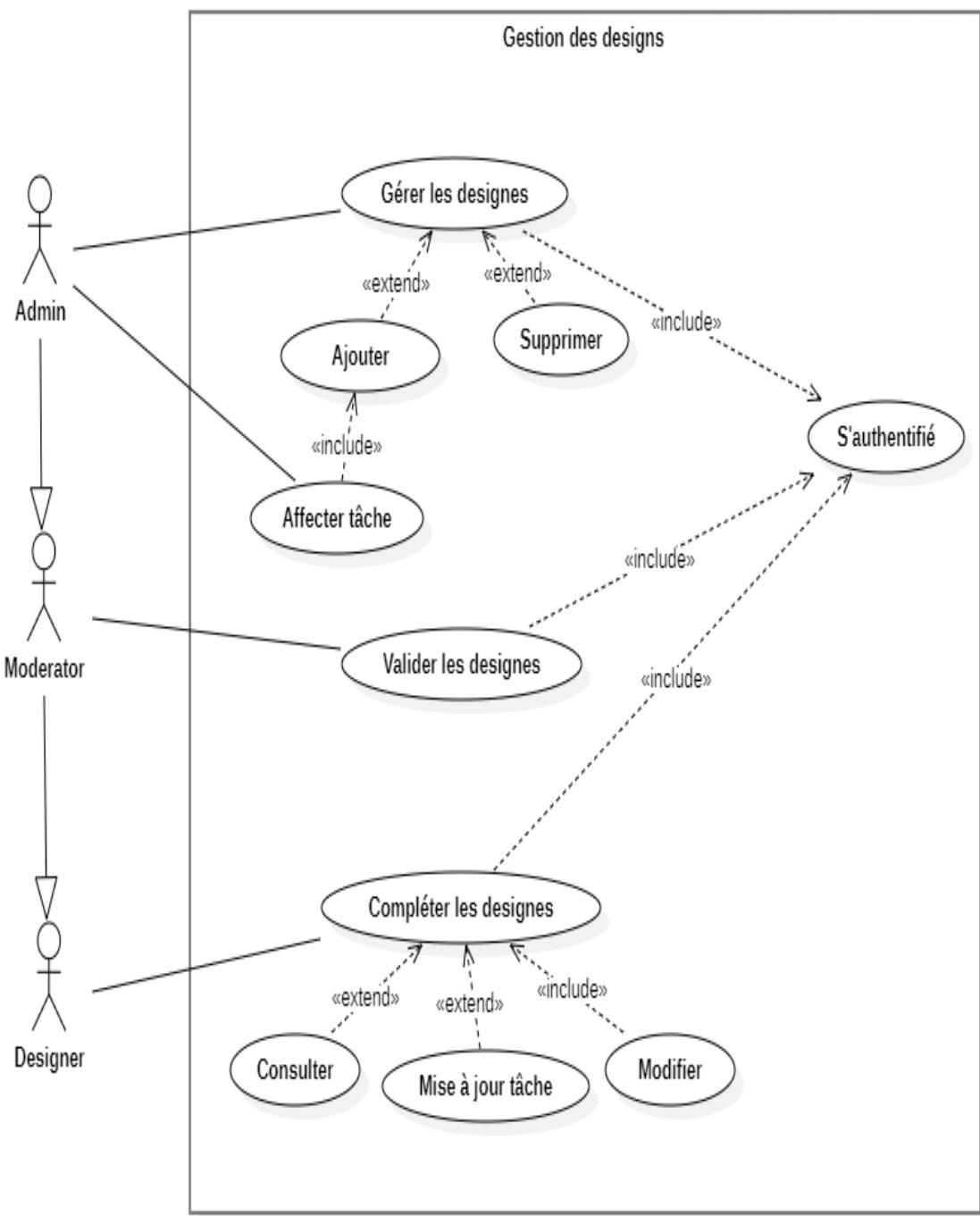


FIGURE VI.1 – Diagramme de cas d'utilisation : Gestion des designs

2.2 Description textuelle des cas d'utilisation :

Raffinement des cas d'utilisation pour la gestion des designs :

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation :	L'utilisateur ajoute et affecte un design.
Acteur :	Admin.
Description :	L'utilisateur souhaite ajouter et affecter des designs.
Pré-condition :	L'utilisateur doit être authentifié. et avoir les permissions nécessaires.
Post-condition :	Les designs sont correctement gérés et mis à jour selon les actions de l'utilisateur.
Scénario nominal :	1- L'admin remplit le formulaire d'ajout des designs. 2- Un formulaire d'affectation des tâches s'ouvre automatiquement. 3- L'admin remplit le formulaire d'affectation des tâches.
Scénario alternatif :	A1 : Si le design existe déjà un message d'erreur s'affiche. A2 : Le formulaire d'affectation ne s'affiche pas.

TABLE VI.2 – Description textuelle du cas d'utilisation "Gestion des designs"

Cas d'utilisation :	L'utilisateur consulte et supprime des designs.
Acteur :	Admin
Description :	L'utilisateur souhaite supprimer des designs.
Pré-condition :	L'utilisateur doit être authentifié et avoir les permissions nécessaires.
Post-condition :	Les designs sont supprimés de la base de données et ne sont plus visibles dans la liste des designs.
Scénario nominal :	1- L'utilisateur consulte la liste des designs disponibles. 2- L'utilisateur sélectionne le design à supprimer en cliquant sur "Delete". 3- L'utilisateur confirme la suppression.
Scénario alternatif :	A1 : L'utilisateur annule la suppression après avoir sélectionné le design.

TABLE VI.3 – Description textuelle du cas d'utilisation "L'utilisateur consulte et supprime des designs"

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation	L'utilisateur consulte et modifie la liste de Designs
Acteur	Admin
Description	L'utilisateur modifie un design existant en apportant des changements.
Pré-conditions	L'utilisateur doit être authentifié.
Post-conditions	Les modifications sont enregistrées et le design est mis à jour.
Scénario principal	1- L'utilisateur accède à la liste des designs disponibles. 2- L'utilisateur sélectionne le design qu'il souhaite modifier. 3- L'utilisateur apporte les modifications nécessaires. 4- L'utilisateur enregistre les modifications.
Scénarios alternatifs	A1 : Si l'utilisateur décide de ne pas enregistrer les modifications, le design reste inchangé.

TABLE VI.4 – Cas d'utilisation : Modification de Designs

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation :	L'utilisateur consulte et valide des designs
Acteur :	Admin,Moderator.
Description :	L'utilisateur souhaite valider des designs
Pré-condition :	L'utilisateur doit être authentifié et avoir les permissions nécessaires
Post-condition :	Les designs sont validé et ne sont plus modifiable par les designers.
Scénario nominal :	1- L'utilisateur consulte la liste des taches disponibles. 2- L'utilisateur sélectionne le designs à vérifier en cliquant sur "Check Design". 3-Le formulaire design qui est remplit déjà par l'admin et le designer s'ouvre. 3-L'utilisateur vérifier les champs et change le status de Migration de design vers "Completed".
Scénario alternatif :	A1 : L'utilisateur annule la validation après avoir vérifier le designs.

TABLE VI.5 – Description textuelle du cas d'utilisation "L'utilisateur consulte et valide des designs"

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation :	L'utilisateur consulte et modifier l'affectation des designs
Acteur :	Admin.
Description :	L'utilisateur souhaite modifier la priority d'une tache
Pré-condition :	L'utilisateur doit être authentifié et avoir les permissions nécessaires
Post-condition :	Les taches sont modifier et le changement sera effectuer dans la base de donnée.
Scénario nominal :	1- L'utilisateur consulte la liste des taches disponibles. 2- L'utilisateur sélectionne la taches à modifier en cliquant sur "Update Task". 3-Le formulaire d'affectation s'affiche. 3-L'utilisateur modifie la priority de cette tache.
Scénario alternatif :	A1 : L'utilisateur annule la modification.

TABLE VI.6 – Description textuelle du cas d'utilisation "L'utilisateur consulte et modifie des taches"

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation :	L'utilisateur consulte et supprime une tache.
Acteur :	Admin.
Description :	L'utilisateur souhaite supprimer la tache.
Pré-condition :	L'utilisateur doit être authentifié et avoir les permissions nécessaires.
Post-condition :	Les taches sont supprimées de la base de données et ne sont plus affichées dans la liste des taches.
Scénario nominal :	<ol style="list-style-type: none">1- L'utilisateur consulte la liste des taches disponibles.2- L'utilisateur sélectionne la tache à supprimer en cliquant sur "Delete Task".3- Message de confirmation de suppression s'affiche.4- L'utilisateur confirme la suppression de cette tache.
Scénario alternatif :	A1 : L'utilisateur annule la suppression.

TABLE VI.7 – Description textuelle du cas d'utilisation "L'utilisateur consulte et supprime des taches"

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Cas d'utilisation :	L'utilisateur consulte et travaille une tâche.
Acteur :	Moderator, Designer.
Description :	L'utilisateur souhaite travailler une tâche.
Pré-condition :	L'utilisateur doit être authentifié et avoir les permissions nécessaires.
Post-condition :	Les designs existent dans la base de données.
Scénario nominal :	1- L'utilisateur consulte la liste des tâches disponibles. 2- L'utilisateur sélectionne la tâche à travailler en cliquant sur "take design". 3- Redirection vers le formulaire de modification "update design". 4- L'utilisateur modifie les champs.
Scénario alternatif :	A1 : L'utilisateur annule la modification.

TABLE VI.8 – Description textuelle du cas d'utilisation "L'utilisateur consulte et travaille une tâche"

3 Diagramme de séquence détaillé :

Les diagrammes de séquence détaillent le déroulement du scénario en présentant les interactions entre les différents acteurs du système et les composants internes.

3.1 Diagramme de séquence "Gestion des designs" :

Cette partie présente le processus de gestion des designs par son diagramme de séquence. Le système permet à l'admin d'ajouter, modifier, supprimer et d'affecter des designs. Les moderators et les designers peuvent consulter et travailler sur les designs assignés, et les moderators peuvent valider les tâches travaillées par les designers.

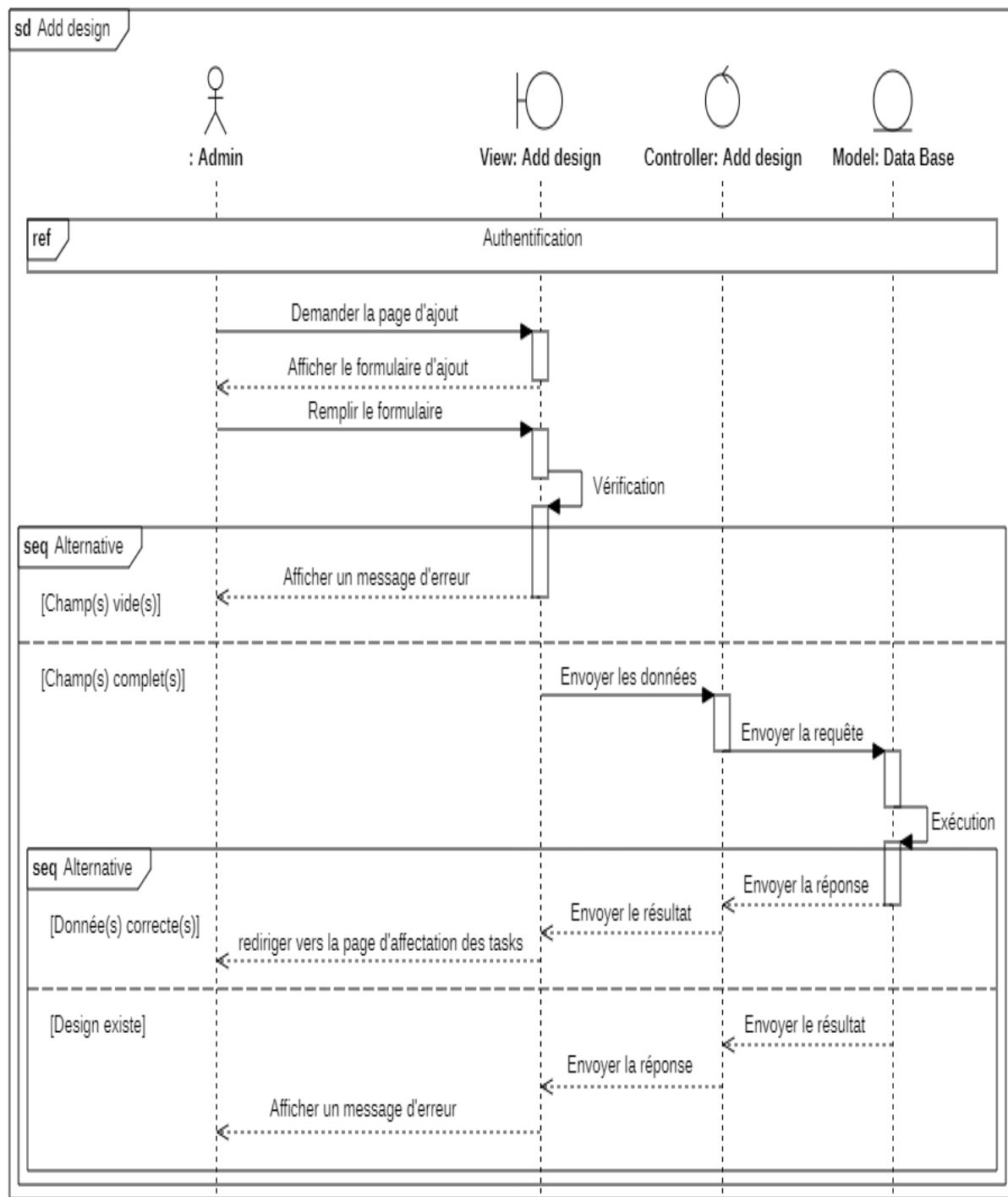


FIGURE VI.2 – Diagramme de séquence de la gestion des tâches

4 Diagramme de classe :

5 Réalisation :

5.1 Interface ajout des designs et affectation des tâches :

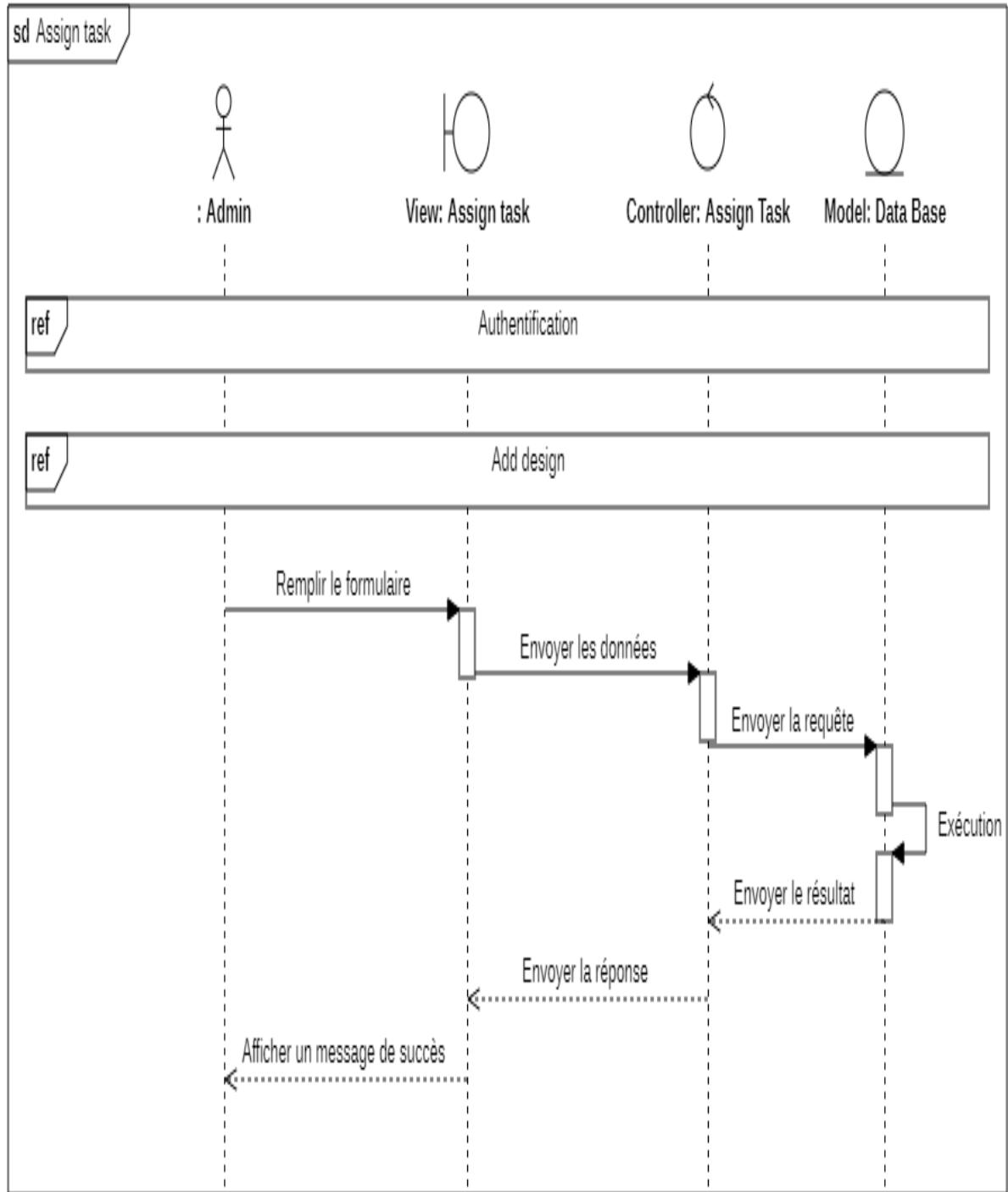


FIGURE VI.3 – Diagramme de classe : Gestion des tâches

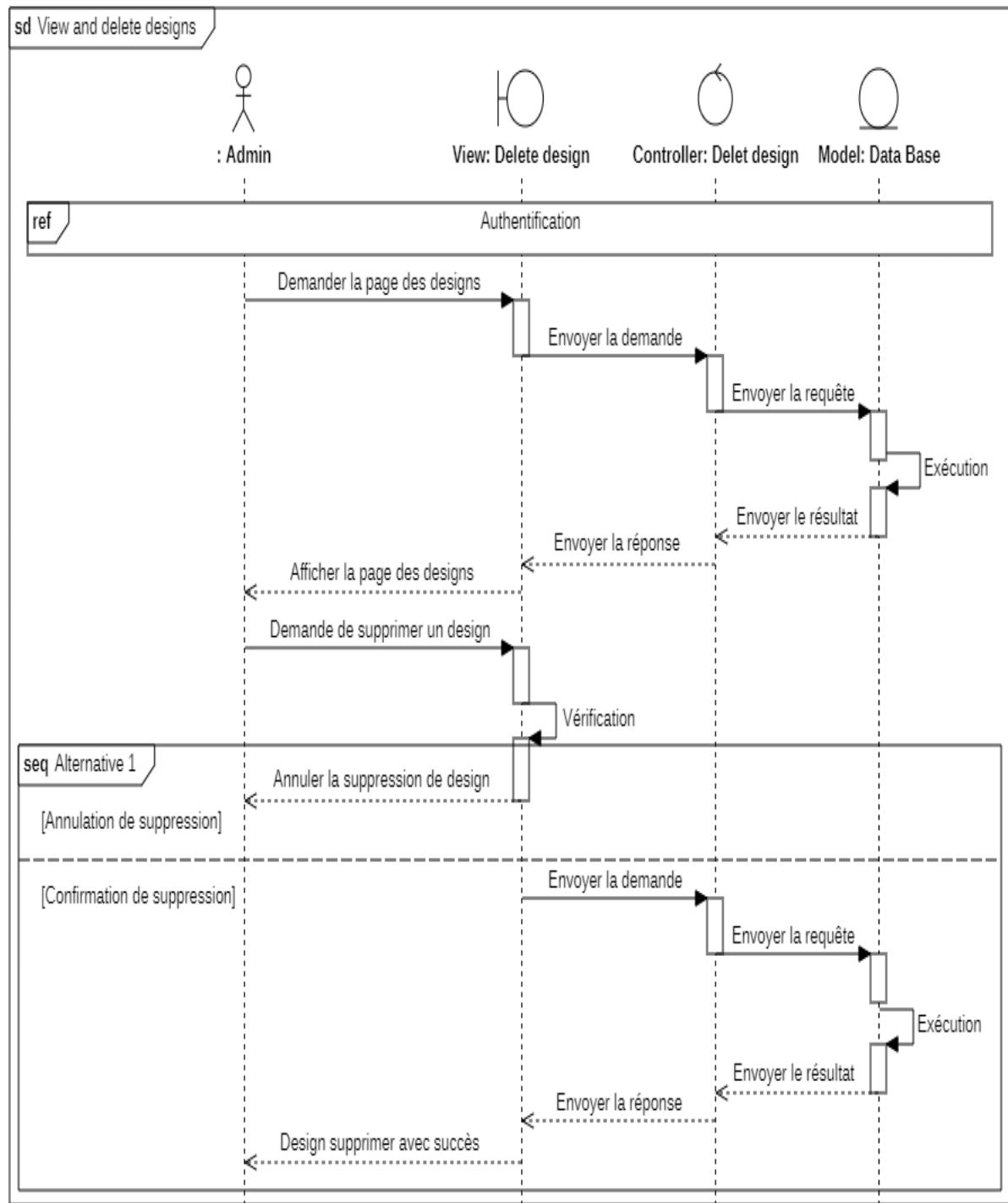


FIGURE VI.4 – Diagramme de classe : Gestion des tâches

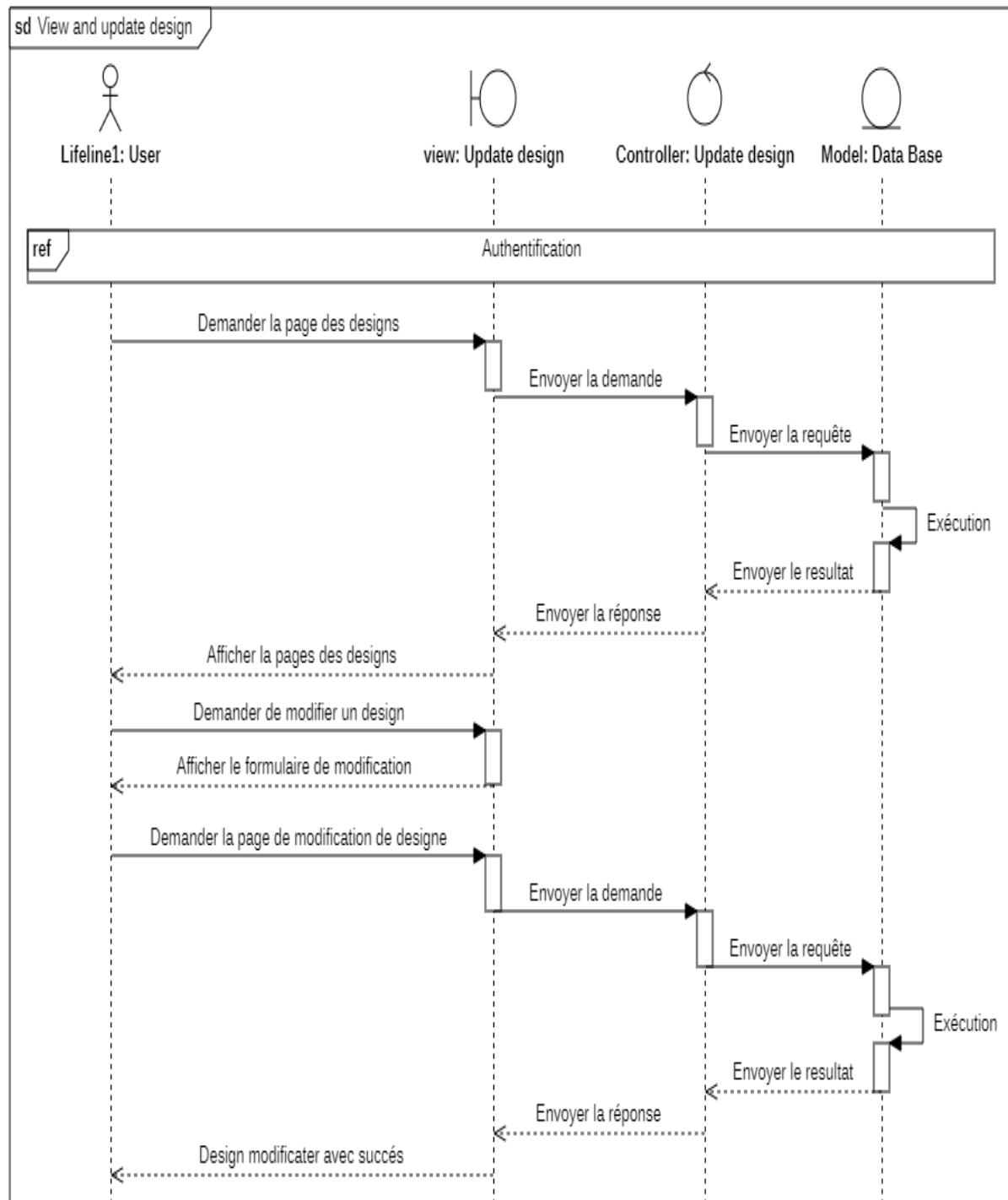


FIGURE VI.5 – Diagramme de classe : Gestion des tâches

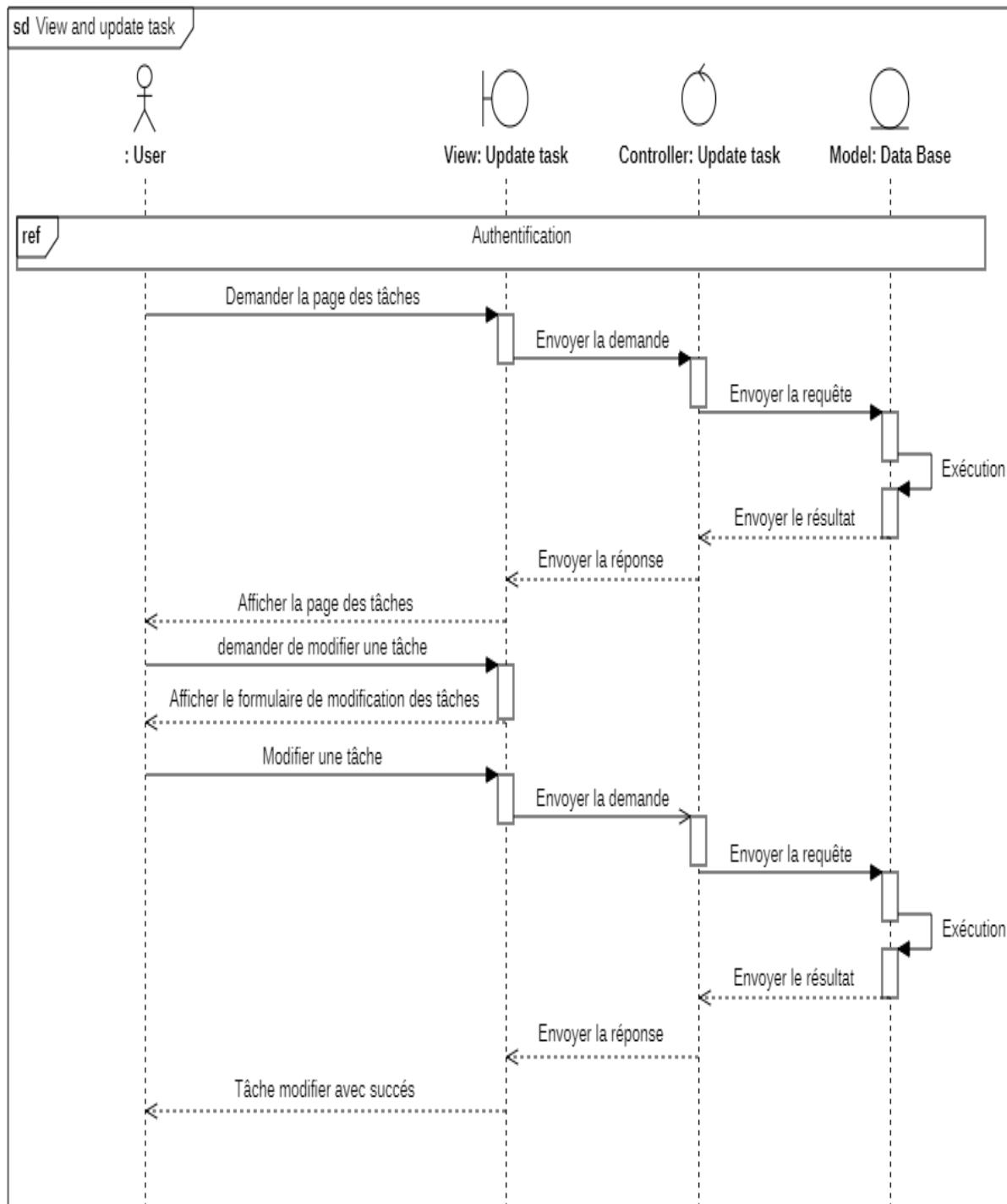


FIGURE VI.6 – Diagramme de classe : Gestion des tâches

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

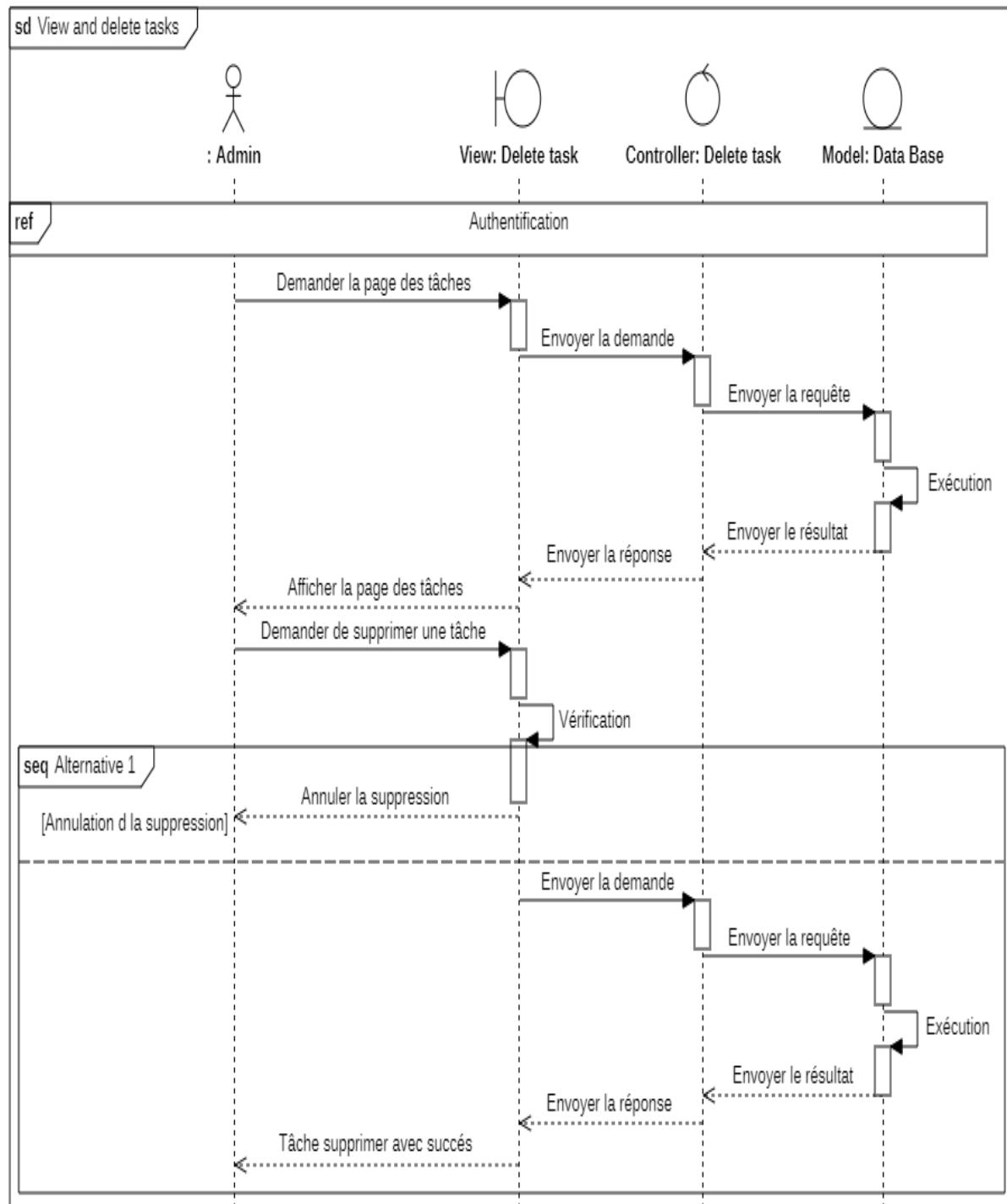


FIGURE VI.7 – Diagramme de classe : Gestion des tâches

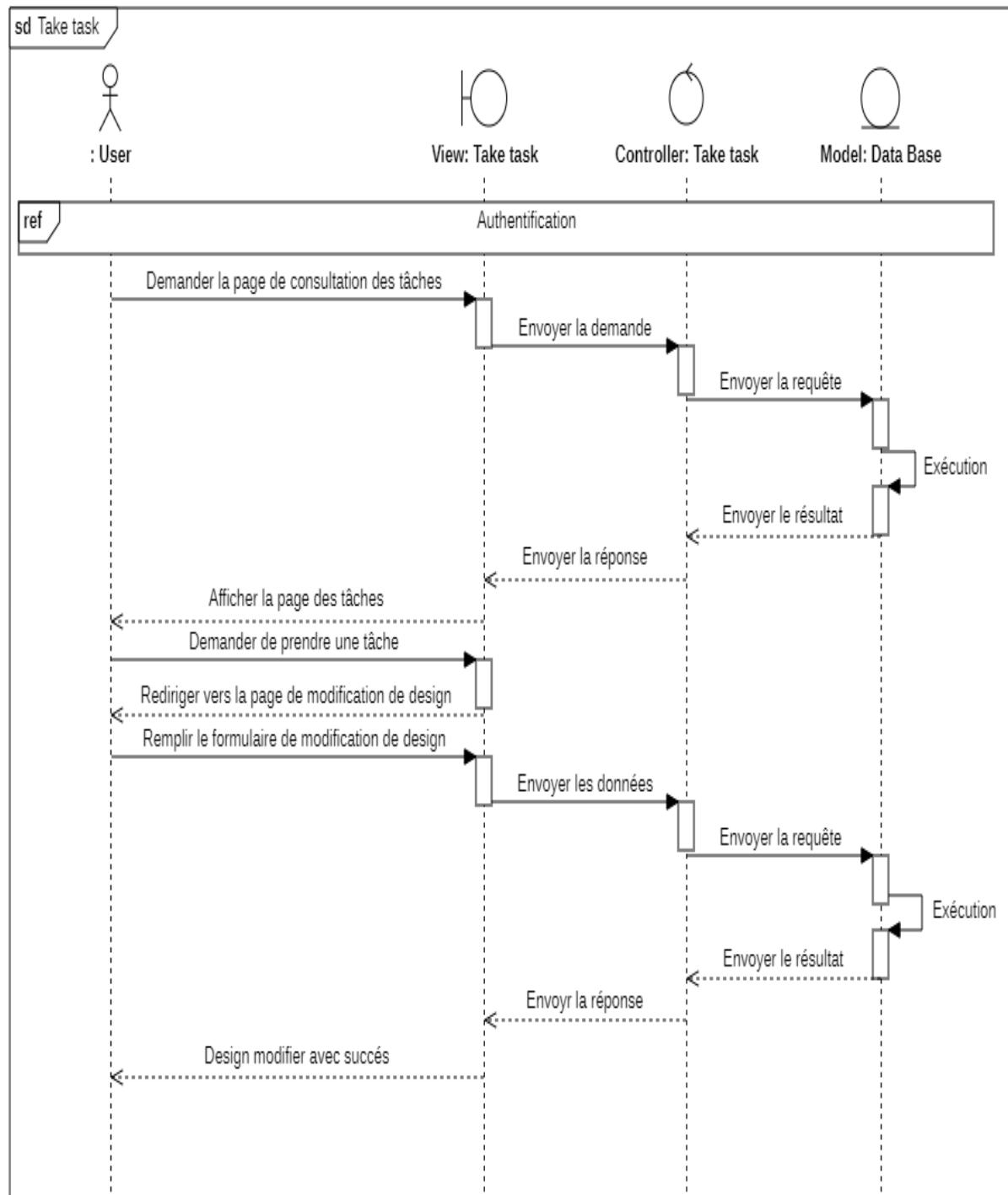


FIGURE VI.8 – Diagramme de classe : Gestion des tâches

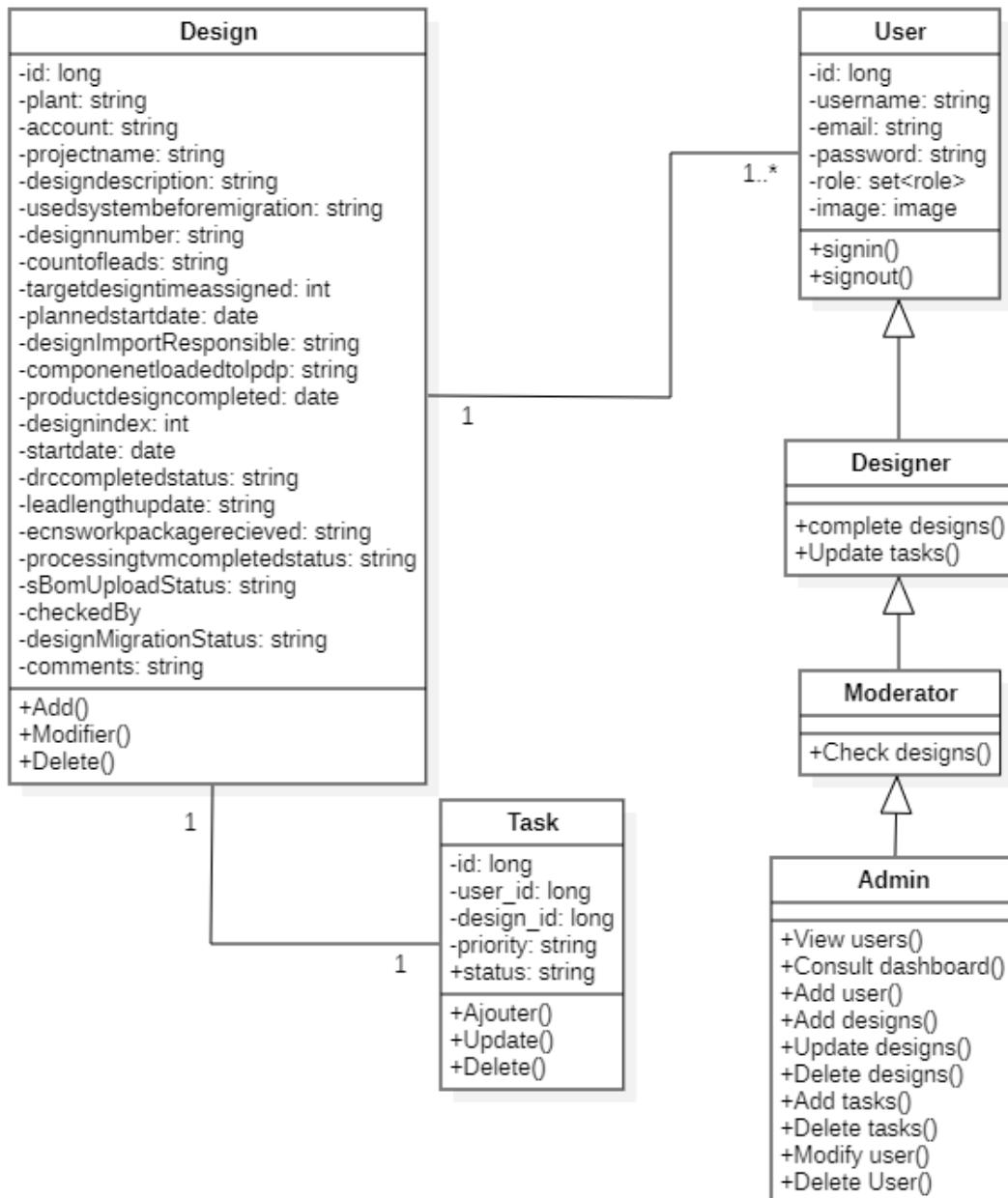


FIGURE VI.9 – Diagramme de classe : Gestion des tâches

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

id	plant	account	projectNa...	designDes...	usedSyste...	designNbr	countOfLea...	targetDesi...	plannedSt...	designImp...	componen...	productDe...
21	LTN3C	BMW Group	G2X	MZ ZB LTG F...	LD	7887562	50	7,2	2023-06-17...	aziz		2023-06-16...
23	GBR651654...	BMW GROUP	G2X8	FD651561DF...	ih	5	50	5	2024-05-31...	wafa	no	2024-05-01...
24	GBR651654...	BMW GROUP	G2X	MZ ZB LTG F...	ER654FR65...	50	50	9	2024-06-0...	yousef	no	2024-05-0...

FIGURE VI.10 – Interface du liste des designes

n...	productDes...	designIndex	start date	drcComple...	leadLength...	eCNsWork...	processing...	sBomUploa...	checkedBy	designMigr...	comments	actions
									refka	Completed		<button>Update</button> <button>Delete</button>
			2023-06-16...	1	2023-06-3...	OK	OK	TC2	ok	OK	Backlog	<button>Update</button> <button>Delete</button>
			2024-05-01...								Backlog	<button>Update</button> <button>Delete</button>
			2024-05-0...									

FIGURE VI.11 – Interface du liste des designes

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

The screenshot shows a web-based application interface for adding production designs. At the top, there is a header with the LEONI logo, a navigation bar with links like 'Home/Statistique', 'PRODUCTION DESIGN', 'EMPLOYER', and 'TASK', and a user profile icon for 'iheb admin'. Below the header, the main form is titled 'Add Production Design'. It contains several input fields: 'Plant' (dropdown), 'Account' (dropdown), 'projectName' (text input), 'Design Description' (text input), 'usedSystemBeforeMigration' (text input), 'Design Nbr' (text input), 'Count Of Leads' (text input), 'Target Design Time Assigned [H]' (text input), 'Planned Start Date' (date input field with placeholder 'jj/mm/aaaa'), and 'Design Import Responsible' (dropdown). A 'Select Design Import Responsible' button is also present.

FIGURE VI.12 – Interface d'ajout des designs

This screenshot shows the same 'Add Production Design' interface as Figure VI.12, but with more fields visible. The additional fields include: 'Start Date' (date input field with placeholder 'jj/mm/aaaa'), 'DRC Completed Status' (radio buttons for 'OK/NOK' and 'OK'), 'Lead Length Update' (text input), 'Gete CNs Workpackage Received' (text input), 'Processing Tvm Completed Status' (radio buttons for 'OK/NOK' and 'OK'), 'Gets Bmn Upload Status' (text input), 'Checked By' (text input), 'Design Migration Status' (dropdown), and 'Comments' (text input). At the bottom left of the form is a green 'Add' button.

FIGURE VI.13 – Interface d'ajout des designs

CHAPITRE VI. SPRINT 4 "GÉRER LES DESIGNS"

Add new task

Nº Design Id
25

User Name
wajdi

Priority
High Priority

State
Backlog

Add Task

FIGURE VI.14 – Interface d'affectation des tâches

Nº Task	ID Design	ID User	Priority	Status	Actions
1	24	yousef	High Priority	Backlog	Delete Task Update Task
2	25	wajdi	High Priority	Backlog	Delete Task Update Task
3	23	wafa	Medium Priority	In progress	Delete Task Update Task
4	21	aziz	High Priority	Completed	Delete Task Update Task

FIGURE VI.15 – Interface de gestion des tâches "admin"

Conclusion générale

Ce projet de développement d'une application web pour le suivi des utilisateurs au sein de l'entreprise LEONI représente une avancée significative vers la digitalisation et l'optimisation des processus de gestion. L'application a permis de répondre aux défis liés au suivi manuel des tâches et des performances des employés.

Les fonctionnalités clés de l'application, telles que la gestion des tâches, des projets, des équipes et des évaluations des employés, ont été conçues pour améliorer l'efficacité opérationnelle et la transparence. Le dashboard interactif facilite la prise de décision, créant ainsi un environnement de travail plus dynamique et organisé.

En somme, ce projet illustre les avantages de l'intégration des technologies web modernes dans la gestion d'entreprise. La solution développée offre une plateforme robuste et flexible qui répond aux besoins spécifiques de LEONI, tout en étant adaptable à d'autres environnements industriels. Les perspectives d'avenir incluent l'ajout de nouvelles fonctionnalités, l'amélioration continue de l'interface utilisateur et l'intégration avec d'autres systèmes pour une gestion encore plus harmonieuse et efficace.

Ce projet a non seulement permis de résoudre des problèmes pratiques, mais a également posé les bases pour des innovations futures dans la gestion des ressources humaines et des processus au sein de l'entreprise. Nous espérons que cette application servira de modèle pour d'autres initiatives de digitalisation, contribuant ainsi à l'évolution technologique de LEONI et à son positionnement en tant que leader industriel.

WEBOGRAPHIE

[1] **Leoni.** Documentation for Leoni. Available at :

<https://www.leoni.com/about-us/locations>. Visited February 15th 2024.

[2] **Scrum.** Documentation for Scrum. Disponible sur :

<https://www.scrum.org/resources/what-is-scrum>. Visité le 26 février 2024.

[3] **React JS.** Documentation for React JS. Disponible sur :

<https://reactjs.org/docs/getting-started.html>. Visité le 20 février 2024.

[4] **Spring Boot.** Documentation for Spring Boot. Disponible sur :

<https://spring.io/projects/spring-boot>. Visité le 7 mars 2024.

[5] **MySQL.** Documentation for MySQL. Disponible sur :

<https://dev.mysql.com/doc/>. Visité 7 mars 2024.

[6] **Vscode.** Documentation for VsCode. Disponible sur :

<https://code.visualstudio.com/docs>. Visité le 7 mars 2024.

[7] **IntelliJ IDEA.** Documentation for IntelliJ IDEA. Disponible sur :

<https://www.jetbrains.com/idea/documentation/>. Visité le 7 mars 2024.

[8] **MySQL Workbench 8.0.** Documentation for MySQL Workbench 8.0. Disponible sur :

<https://dev.mysql.com/doc/workbench/en/>. Visité le 7 mars 2024.

[9] **Postman.** Documentation for Postman. Disponible sur :

<https://learning.postman.com/docs/>. Visité le 15 mars 2024.

WEBOGRAPHIE

[10] **GitHub.** GitHub Documentation. Disponible sur :

<https://docs.github.com/en>. Visité le 20 mars 2024.

[11] **StarUML.** StarUML Documentation. Disponible sur :

<http://staruml.io/user-guide>. Visité le 27 février 2024.

[12] **MVC.** Documentation for MVC. Disponible sur :

<https://medium.com/@sadikarahmantanisha/the-mvc-architecture-97d47e071eb2>.

Visité le 5 mars 2024.

[13] **HTTPS.** Documentation for HTTPS. Disponible sur :

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>. Visité le 6 mars 2024.

[14] **JWT.** Documentation for JSON Web Token (JWT). Disponible sur :

<https://jwt.io/introduction/>. Visité le 6 mars 2024.

[15] **Bcrypt.** Documentation pour cryptage de mot de passe(Bcrypt).Disponible sur :

<https://www.baeldung.com/spring-security-registration-password-encoding-bcrypt>.

Visité le 6mars février 2024.

Résumé

Le présent rapport synthétise le travail effectué dans le cadre du projet de fin d'études pour l'obtention du diplôme de Licence en Technologies de l'Information et de la Communication au sein de l'entreprise LEONI. Ce projet vise à digitaliser et à optimiser les processus de gestion des tâches et du suivi des employés. À travers le développement d'une application web, nous avons cherché à remplacer les méthodes manuelles de suivi des tâches par un système intégré offrant des fonctionnalités de gestion des tâches, des projets, des équipes et des évaluations des employés.

Mots clés : Gestion des tâches, Projet, Suivi des employés

Abstract

The present report summarizes the work carried out as part of the end-of-study project for obtaining the Bachelor's degree in Information and Communication Technologies within the company LEONI. This project aims to digitize and optimize task management and employee monitoring processes. Through the development of a web application, we sought to replace manual task tracking methods with an integrated system offering features for task, project, team, and employee evaluation management.

Keywords : Task Management, Project, Employee monitoring