

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

**Université de la Manouba**

**Institut Supérieur des Arts Multimédias**



**Mémoire de Projet de Fin d'Etudes**

**Code IM02**

---

# **Conception et développement d'une application web de gestion du temps de travail du personnel**

---

**Réalisé par**

Safa Kassous 3IM5

Amir Horchani 3IM5

**Encadré par**

Mohamed Karim Azizi (ISAMM)

Année universitaire 2018 /2019

## *Remerciements*

Nos remerciements sincères vont à toutes les personnes qui ont contribué au succès de notre stage et qui nous ont aidé lors de la rédaction de ce rapport.

Tout d'abord, nous tiendrons à remercier Mr Ahmed Hassini pour son accueil, le temps passé ensemble et le partage de son expertise au quotidien. Grâce aussi à sa confiance on a pu nous accomplir totalement dans nos missions.

Nous remercions vivement notre cher encadrant, Mr Mohamed Karim Azizi, pour le temps qu'il nous a consacré, son encouragement et ses efforts fournis tout au long du travail.

Nous remercions également Mr Amin Kchiche pour les précieuses informations qu'il nous a prodiguées avec intérêt ce qui nous a permis d'avancer dans notre projet.

*Safa & Amir*

## *Dédicaces*

J'aimerai dédier ce travail :

À mes parents,  
qui m'ont doté d'une éducation digne, pour leur amour qui a  
fait de moi ce que je suis aujourd'hui.  
Que dieu vous préserve une longue vie heureuse.

À mon petit cher frère,  
pour son amour et son incontestable appui  
à qui je souhaite une vie pleine de réussite et de bonheur.

À toute ma famille,  
tous ce que j'aime, qui m'aiment et me comblent  
de conseils et d'encouragements

À tous mes amis,  
qui ont été toujours près de moi je vous souhaite  
un avenir à la hauteur de vos ambitions.  
Que notre amitié dure

*Safa Kassous*

## *Dédicaces*

Je dédie ce projet :

À mes parents,  
qui m'ont soutenu et encouragé durant les années d'études.  
Qu'ils trouvent ici le témoignage de ma profonde  
reconnaissance.

À mes frères et sœurs,  
qui ont partagé avec moi tous les moments d'émotions lors  
de la réalisation de ce travail.

À toute ma famille,  
qui m'ont chaleureusement supporté et encouragé tout au  
long de mon parcours.

À mes amis,  
qui m'ont toujours encouragé et à qui je souhaite plus de  
bonheur et de succès.  
À tous ce que j'aime.

*Amir Horchani*

# Table des matières

Introduction générale .....	1
Chapitre 1 : Etat de l'art et Etude de l'existant .....	2
Introduction .....	3
I.    Cadre du stage .....	3
II.   Cadre du projet.....	3
II.1.  Problématique .....	3
II.2.  Présentation du sujet .....	3
II.3.  Description de la cible .....	4
III.  Approche méthodologique .....	4
III.1.  Choix de cycle de vie.....	4
III.2.  Langage de modélisation .....	5
IV.   Etude de l'existant .....	5
IV.1.  Etude de l'application bureau BTManager.....	5
IV.1.1.  Description de BTManager .....	5
IV.1.2.  Analyse graphique.....	6
IV.1.3.  Analyse fonctionnelle.....	6
IV.2.  Etude de l'application web de gestion de pointage MBG.....	7
IV.2.1.  Description de MBG.....	7
IV.2.2.  Analyse graphique.....	7
IV.2.3.  Analyse fonctionnelle.....	8
IV.3.  Etude de l'application de gestion des ressources humaines.....	8
IV.3.1.  Description de l'application .....	8
IV.3.2.  Analyse graphique.....	9
IV.3.3.  Analyse fonctionnelle.....	9
V.    Critique de l'existant.....	10
VI.   Solution proposée .....	10
VI.1.  Volet graphique .....	10
VI.2.  Volet technique .....	11
Conclusion.....	11
Chapitre 2 : Analyse et spécification des besoins .....	12
Introduction .....	13

I.	Identification des besoins .....	13
I.1.	Identification des besoins fonctionnels .....	13
I.1.1.	Identification des acteurs et des cas d'utilisation.....	13
I.1.2.	Le diagramme de cas d'utilisation.....	15
I.1.3.	Description textuelle des principaux cas d'utilisation .....	16
I.2.	Identification des besoins non fonctionnels .....	26
II.	Pilotage du projet avec SCRUM .....	26
II.1.	Backlog du produit.....	27
II.2.	Planification des sprints .....	29
	Conclusion.....	29
	Chapitre 3 : Conception .....	30
	Introduction .....	31
I.	Conception technique.....	31
I.1.	Description de la vue statique .....	31
I.1.1.	Diagramme de classe .....	31
I.1.2.	Dictionnaire des données.....	32
I.2.	Description de la vue dynamique.....	33
I.2.1.	Diagramme de séquence objet.....	33
II.	Conception graphique .....	40
II.1.	Schéma de navigation .....	40
II.2.	La charte graphique.....	40
II.2.1.	Police de caractère .....	40
II.2.2.	Palette graphique .....	41
II.2.3.	Fond de page et template .....	41
II.2.4.	Logo de la société .....	42
III.	Description des sprints.....	42
III.1.	Release 1 .....	42
III.1.1.	Diagramme de cas d'utilisation du premier release.....	42
III.1.2.	Sprint 1 : Gestion des comptes .....	43
III.1.3.	Sprint 2 : Gestion de type d'absence .....	43
III.1.4.	Sprint 3 : Gestion des congés.....	44
III.1.5.	Sprint 4 : gestion des périphériques.....	45
III.1.6.	Sprint 5 : Gestion des usagers.....	45

III.2. Release 2 .....	46
III.2.1. Diagramme de cas d'utilisation du deuxième release.....	46
III.2.2. Sprint 6 : Consultation des heures de présence.....	46
III.2.3. Sprint 7 : Traitement des données de pointage.....	47
Conclusion.....	47
Chapitre 4 : Réalisation et test.....	48
Introduction .....	49
I. Environnement de travail .....	49
I.1. Environnement matériel.....	49
I.2. Environnement logiciel.....	49
I.2.1. Logiciels utilisés .....	49
I.2.2. Technologies et langages utilisés .....	50
II. Choix technique .....	51
II.1. Architecture générale du projet.....	51
II.2. Diagramme de déploiement .....	52
III. Description des interfaces graphiques .....	52
IV. Evaluation de la qualité du produit.....	56
Conclusion.....	56
Conclusion générale .....	57

# Liste des figures

Figure 1 : Eléments de Scrum .....	4
Figure 2 : Interface d'accueil de BTManager .....	5
Figure 3: Interface d'accueil de MBG .....	7
Figure 4: Interface d'accueil de l'application de gestion des ressources humaine .....	8
Figure 5: Diagramme de cas d'utilisation global .....	15
Figure 6: Diagramme de séquence système de "S'authentifier" .....	16
Figure 7: Diagramme séquence système de "Gestion des comptes" .....	18
Figure 8: Diagramme séquence système de "Gestion des usagers" .....	20
Figure 9: Diagramme de séquence système de "Gestion de type d'absence" .....	22
Figure 10: Diagramme de séquence système de "Consulter les heures de présence" .....	23
Figure 11: Diagramme de séquence système de "Récupérer données de pointage" .....	25
Figure 12: Diagramme de classe .....	31
Figure 13: Diagramme de séquence objet de S'authentifier .....	34
Figure 14: Diagramme de séquence objet de Gestion des comptes .....	35
Figure 15: Diagramme de séquence objet de Gestion des usagers.....	36
Figure 16: Diagramme de séquence objet de Gestion de type d'absence .....	37
Figure 17: Diagramme de séquence objet de Consulter les heures de présence .....	38
Figure 18: Diagramme de séquence objet de Récupérer les données de pointage .....	39
Figure 19: Schéma de navigation .....	40
Figure 20: Palette des couleurs.....	41
Figure 21: Maquette de vision globale sur les interfaces .....	41
Figure 22: Logo de l'application .....	42
Figure 23: Diagramme de cas d'utilisation du premier release.....	42
Figure 24: Diagramme de cas d'utilisation du deuxième release.....	46
Figure 25: Architecture du projet .....	51
Figure 26: Diagramme de déploiement .....	52
Figure 27: Réalisation de la page d'authentification.....	52
Figure 28: Réalisation de la page du profil.....	53
Figure 29: Réalisation de la page de création des comptes .....	53
Figure 30: Réalisation de la page de gestion des usagers.....	54
Figure 31: Réalisation de la page de gestion des périphériques .....	54



Figure 32: Réalisation de la page des comptes.....	55
Figure 33: Réalisation de la page de gestion de pointage .....	55
Figure 34: Lancement du serveur.....	56
Figure 35: Récupération de la donnée.....	56
Figure 36: Test de l'affichage des données dans l'interface pointage .....	56

# Liste des tableaux

Tableau 1 : Analyse graphique de BTManager .....	6
Tableau 2 : Analyse graphique de MBG .....	8
Tableau 3 : Analyse graphique de l'application de gestion des ressources humaines .....	9
Tableau 4: Critique de l'existant .....	10
Tableau 5: Identification des acteurs et des cas d'utilisation .....	14
Tableau 6: Description textuelle de "S'authentifier" .....	16
Tableau 7: Description textuelle de "Gestion des comptes" .....	17
Tableau 8: Description textuelle de cas "Gestion des usagers" .....	19
Tableau 9: Description textuelle de cas "Gestion de type d'absence" .....	21
Tableau 10: Description textuelle de cas "Consulter les heures de présence" .....	23
Tableau 11: Description textuelle de cas " Récupérer les données de pointage " .....	24
Tableau 12: Backlog du produit .....	28
Tableau 13: Planification des sprints.....	29
Tableau 14: Dictionnaire des données.....	33
Tableau 15: Backlog du sprint 1 Gestion des comptes .....	43
Tableau 16: Backlog du sprint 2 Gestion de type d'absence.....	44
Tableau 17: Backlog du sprint 3 Gestion des congés.....	44
Tableau 18: Backlog du sprint 4 Gestion des périphériques .....	45
Tableau 19: Backlog du sprint 5 Gestion des usagers .....	45
Tableau 20: Backlog du sprint 6 Consultation des heures de présence.....	46
Tableau 21: Backlog du sprint 7 Traitement des données de pointage .....	47
Tableau 22: Logiciels utilisés.....	50
Tableau 23: Technologies et langages utilisés .....	50

# Introduction générale

Avec la croissance explosive des informations automatisées, la gestion de ces dernières devient de plus en plus importante pour les entreprises qui ne cessent de stocker un tas immense de données critiques à travers leurs réseaux locaux. Plusieurs études ont montré que ces entreprises, face au nombre croissant des employés et de ces informations, vont vers l'allure d'utiliser un système de pointage comme première mesure pour faciliter la gestion de temps de travail du personnel.

C'est dans ce cadre que s'inscrit notre projet de fin d'études, la société a un nombre très important d'employés qui sont obligés de pointer manuellement. Et nous étions amené à développer une application monopage fiable et efficace permettant la gestion du pointage automatique de telle façon que nous pouvons facilement consulter et calculer le nombre des heures de travail et les absences. En plus, nous avons réalisé l'écoute sur le port de la pointeuse pour récupérer les données de pointage des employés en temps réel et les stocker dans la base de données pour avoir un accès à ces données à travers l'application web.

Le rapport qui suit est organisé en quatre chapitres :

**Dans le premier chapitre :** Etat de l'art et étude de l'existant, nous mettons notre sujet dans son contexte général. D'abord nous présentons la société d'accueil, suivi de la problématique et la solution proposée et par la suite le choix de la méthodologie de travail et nous finissons par l'étude de quelques applications similaires.

**Dans le deuxième chapitre :** Analyse et spécification des besoins, nous analysons les besoins fonctionnels et non fonctionnels de notre application. Nous faisons une étude détaillée des différents cas d'utilisation possibles avec leurs descriptions. Et nous finissons par le Backlog du produit et la planification des sprints.

**Dans le troisième chapitre :** Conception, nous présentons l'aspect statique en présentant le diagramme de classe et l'étude détaillée de chaque module avec le diagramme de l'aspect dynamique afin de concevoir une application réalisable et fiable.

**Dans le quatrième chapitre :** La réalisation, nous présentons l'environnement matériel et logiciel dans lequel le projet a été réalisé et l'architecture de l'application. Ainsi que les différents choix techniques liés à cette application, et les contraintes de réalisation. Nous terminons par la présentation des interfaces de l'application.

---

## **Chapitre 1 : Etat de l'art et Etude de l'existant**

---

## **Introduction :**

Avant de débiter le projet, il est indispensable de commencer une étude préliminaire et par la suite discuter des solutions existantes dans le marché afin de détecter leurs qualités et leurs défauts.

### **I. Cadre du stage :**

On a eu de la chance d'y effectuer notre stage au sein de Leeds Engineering qui est fondée en 2007 par Ahmed Hassini, ingénieur en informatique, est une société spécialisée dans les solutions de gestion et de sécurité informatique. Elle développe son savoir-faire au service des grandes entreprises Tunisiennes. Depuis sa création, elle a pris en charge des études et des réalisations de solutions de gestion informatisées pour les entreprises.

Leeds Engineering opère sur plusieurs domaines d'activités : la sécurité informatique, les travaux d'analyse et d'audit et l'intervention sur site.

### **II. Cadre du projet :**

Nul ne peut nier l'importance que revêtent les connaissances théoriques acquises durant les semestres d'études universitaires. Mais, il est essentiel de les compléter avec des connaissances pratiques afin de développer un esprit opérationnel. Pour ce faire, le meilleur moyen est de passer par un stage qui est demandé par notre Institut Supérieur des Arts Multimédias afin de qualifier ses étudiants. Dans ce cadre, nous avons eu l'occasion de passer notre stage de fin d'études durant la période 01/02/2019 – 15/05/2019 au sein de la société Leeds Engineering.

#### **II.1. Problématique :**

Les entreprises ayant une application desktop pour la gestion de temps de travail au sein de leurs locaux risquent un problème d'intégration et de déploiement de l'application et par conséquent une perte du temps.

#### **II.2. Présentation du sujet :**

Leeds engineering souhaite développer une application web de gestion de temps de travail des employés dédié au super administrateur, au responsable et au gestionnaire.

En effet, la société veut transformer la solution existante sur les ordinateurs en une version web simple à utiliser et à intégrer.

En plus de l'application, la société souhaite développer un module qui permet de récupérer les informations d'un employé qui a pointé et de les envoyer en temps réel vers la base de

données de l'application afin de permettre aux utilisateurs de l'application de consulter les heures de travail du personnel.

### II.3. Description de la cible :

Toutes les entreprises propriétaires d'une application bureau qui cherchent à transformer la solution client-serveur en une solution accessible sans besoin d'installation. On cible les entreprises qui ne veulent pas être liées aux informaticiens pour intégrer l'application dans chaque poste de travail.

## III. Approche méthodologique :

Dans cette section, nous allons présenter le choix de cycle de vie et du langage de modélisation afin de pouvoir comprendre la démarche suivie dans la conception et la réalisation de l'application, tout comme la rédaction du rapport.

### III.1. Choix de cycle de vie :

Afin de bien comprendre le fonctionnement de notre système, nous avons d'abord étudié la documentation fournie. Nous avons ainsi utilisé Scrum comme méthodologie de gestion du projet et ce afin de bien conduire notre projet et assurer le bon déroulement des différentes phases. Donc nous avons choisi la méthodologie agile autrement appelée SCRUM.

La figure 1 est une présentation des principaux éléments de SCRUM.

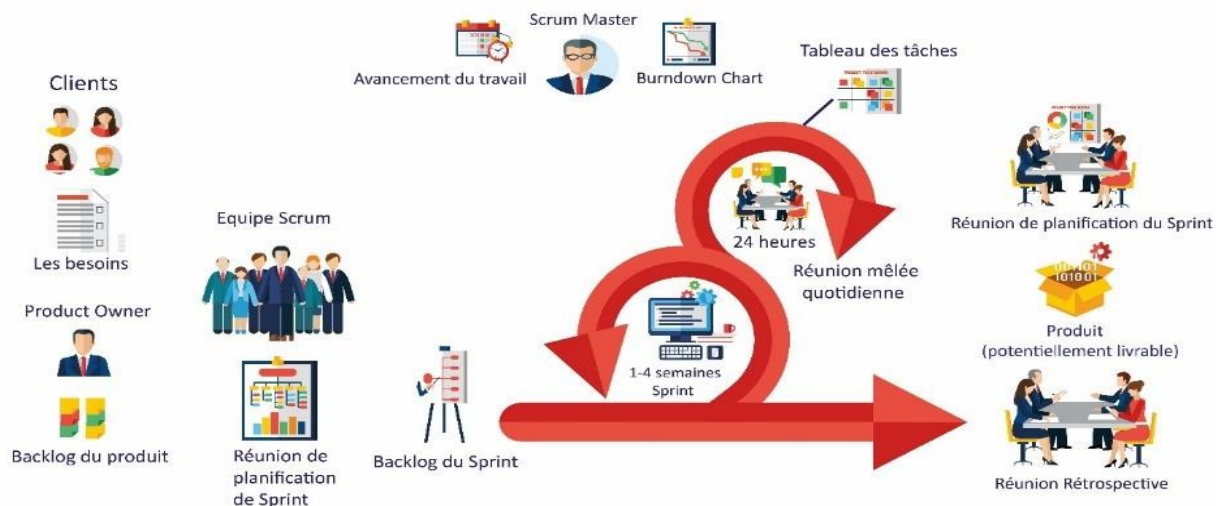


Figure 1 : Eléments de Scrum

### III.2. Langage de modélisation :

Afin de bien réaliser la partie analyse et spécification des besoins et la partie conception du projet nous sommes censé suivre une démarche de modélisation pour définir les fonctionnalités de l'application web. Donc, on a choisi comme langage de modélisation UML<sup>1</sup> qui fait partie des notations les plus utilisées aujourd'hui.

UML nous fournit des diagrammes pour présenter l'application à développer : ses fonctionnalités, les actions à effectuer par l'application, sa mise en route, etc.

## IV. Etude de l'existant :

Nous ne saurions débiter ce travail sans avoir une idée claire et précise sur l'existant quel qu'il soit. Cette phase est très importante pour la mise en route du projet, elle permet de chercher les carences des applications web de gestion de temps déjà existantes afin de les améliorer. Donc, on a choisi pour notre étude trois applications :

- L'application bureau **BTManager** déjà existante au sein de la société.
- L'application web de gestion de pointage **MBG**.
- L'application web de gestion des ressources humaines.

### IV.1. Etude de l'application bureau BTManager :

#### IV.1.1. Description de BTManager :

BTManager [N1] est un logiciel de gestion du temps de travail. Il permet, grâce à la collecte automatique et instantanée des pointages des salariés de calculer plusieurs données (temps de travail, heures supplémentaires ...) afin de faciliter la gestion des collaborateurs.

La figure 2 montre l'interface d'accueil de BTManager.

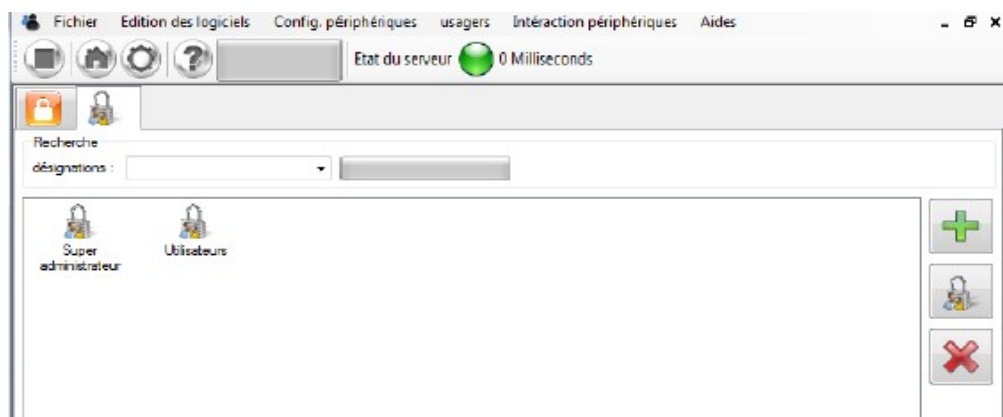


Figure 2 : Interface d'accueil de BTManager

<sup>1</sup>UML: Unified Modeling Language

### IV.1.2. Analyse graphique :

Dans cette partie, nous allons analyser les éléments graphiques de BTManager.


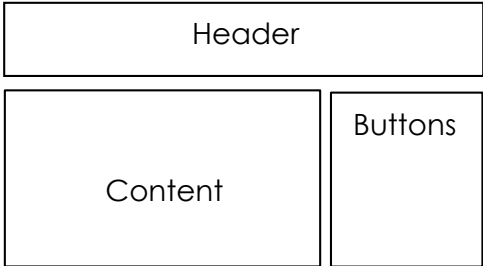
	Elément graphique	Sens conceptuel
<b>Logo</b>		<ul style="list-style-type: none"> <li>- Le logo est composé d'un sigle, du nom BTManager écrit en anglais et d'un slogan signifiant la gestion de temps.</li> <li>- Il est mis sur un fond bleu hussard pour faire apparaître les éléments en couleur clair.</li> <li>⇒ Le logo est clair et reflète le thème du logiciel.</li> </ul>
<b>Structure</b>		<ul style="list-style-type: none"> <li>- Un header contient une barre de navigation.</li> <li>- Un contenu qui contient les informations principales de la page.</li> <li>- Les boutons représente l'ajout, la modification ou la suppression.</li> <li>⇒ Un telle structure facilite la navigation et offre une apparence visuelle claire.</li> </ul>
<b>Composition</b>	La composition est basée sur les formes géométriques avec insertion de quelques icônes en relation avec la fonction du bouton.	⇒ Les rectangles donne de la stabilité à l'interface mais les cercles et les images ajoute du dynamisme aux interfaces.
<b>Couleurs</b>	Les couleurs dominantes sont le gris et le blanc ou changement avec insertion des icônes.	⇒ L'ajout des couleurs sombres ou la modification de la couleur de l'arrière-plan rend l'application plus dynamique.
<b>Typographie</b>	Typographie simple utilisant « Arial »	Typographie clair et visible.

Tableau 1 : Analyse graphique de BTManager

### IV.1.3. Analyse fonctionnelle :

Ce qui rend BTManager une application riche est l'optimisation des ressources humaines et l'offre de la fiabilité.

Parmi les fonctionnalités, les plus intéressantes :

- Gestion des employés.
- Gestion de la présence.
- Gestion des comptes rendus (calcul des heures supplémentaires ...).
- Gestion de paie.



## IV.2. Etude de l'application web de gestion de pointage MBG :

### IV.2.1. Description de MBG :

MBG est une application web qui permet la gestion de temps de travail des employés dans la société. Elle permet aux managers de contrôler les heures de pointage et de gérer l'indisponibilité afin de gagner du temps et aux employés de consulter ses heures de travail.


La figure 3 montre l'interface d'accueil de MBG.



Figure 3: Interface d'accueil de MBG

### IV.2.2. Analyse graphique:

Dans cette partie, nous allons analyser les éléments graphiques de MBG comme illustre la tableau 2.

	Elément graphique	Sens conceptuel
<b>Logo</b>		<ul style="list-style-type: none"> <li>- Le logo est composé par le nom de la société et le domaine de travail de cette dernière.</li> <li>- Il est mis sur un fond noir.</li> </ul>
<b>Structure</b>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Header</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">Content</div>	<ul style="list-style-type: none"> <li>- Un header qui comprend le logo de la société et une barre de navigation.</li> <li>- Un contenu qui contient les informations principales de la page. ⇒ Un telle structure facilite la navigation et offre une apparence visuelle claire.</li> </ul>
<b>Composition</b>	La composition manque des formes. On remarque qu'il y a que des textes et des liens.	⇒ L'absence des formes donne de la stabilité aux interfaces.

<b>Couleurs</b>	Plusieurs couleurs sont utilisées : bleu, rouge et noir.	⇒ L'utilisation des couleurs différentes qui ne sont pas harmonieuses entraîne une mauvaise visibilité.
<b>Typographie</b>	Typographie simple utilisant « Arial » avec des couleurs et des tailles différentes.	Typographie clair malgré la variété des tailles.

Tableau 2 : Analyse graphique de MBG

### IV.2.3. Analyse fonctionnelle :

La richesse de MBG, c'est de donner la possibilité de générer des statistiques de pointage pour montrer encore la fiabilité en collectant et analysant les données.

D'autres fonctionnalités peuvent être dégagées, citons :

- Récupération des données.
- Contrôle le pointage.
- Affectation des programmes de travail aux employés.
- Suivi des heures supplémentaires de travail.

### IV.3. Etude de l'application de gestion des ressources humaines :

#### IV.3.1. Description de l'application :

C'est une application web qui permet la gestion des ressources humaines dans la société (les congés, la présence ...). Elle permet ainsi aux utilisateurs de l'application de recruter des nouveaux employés, d'ajouter des utilisateurs et d'afficher la fiche d'un salarié.

La figure 4 illustre l'interface d'accueil de l'application.

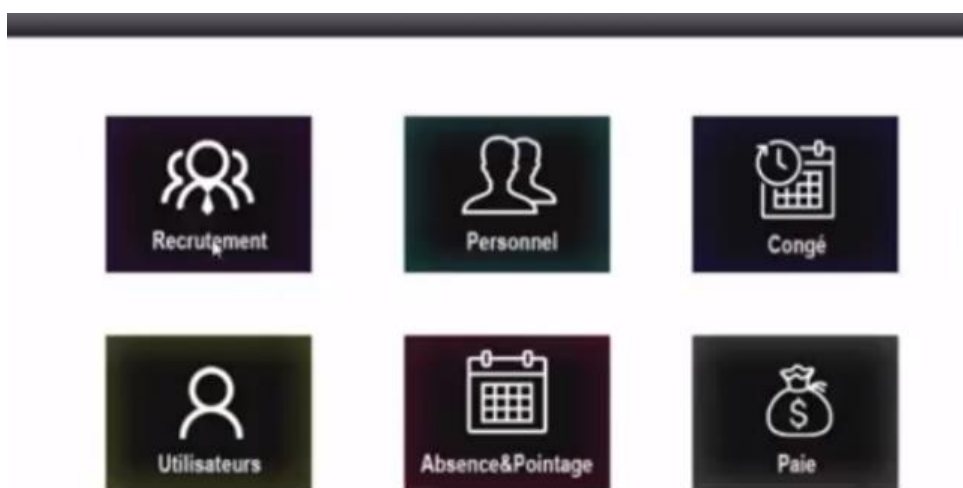


Figure 4: Interface d'accueil de l'application de gestion des ressources humaine

### IV.3.2. Analyse graphique :

Dans cette partie, nous allons analyser les éléments graphiques de l'application comme montre le tableau 3.

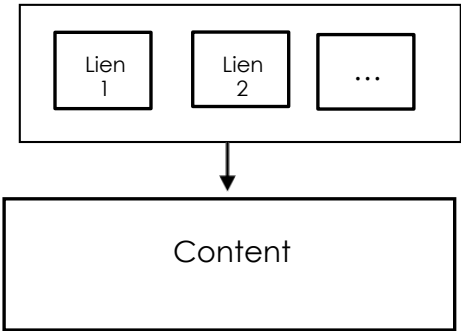
	Elément graphique	Sens conceptuel
<b>Logo</b>	Pas de trace du logo	La présence du logo dans les applications est très importante puisqu'il serve à distinguer entre les différentes sociétés et il reflète l'objectif de l'application.
<b>Structure</b>		<ul style="list-style-type: none"> <li>- Une barre de navigation qui contient des liens vers d'autres pages et des informations supplémentaires.</li> <li>- Un contenu qui contient les informations principales de la page.</li> </ul> <p>⇒ Un telle structure n'offre pas la rapidité de navigation. Il y'a une possibilité de mettre une barre latérale avec le contenu d'une page dans une même interface.</p>
<b>Composition</b>	La composition est basée sur les formes géométriques et les icônes dans la barre latérale.	⇒ Les formes rectangulaires utilisées donnent de la stabilité mais l'ajout des icônes offre du dynamisme.
<b>Couleurs</b>	Les couleurs dominantes sont le noir et le blanc.	⇒ Les couleurs utilisées sont harmonieuses.
<b>Typographie</b>	Typographie simple utilisant « Arial » avec des couleurs et des tailles différentes.	Typographie clair et visible

Tableau 3 : Analyse graphique de l'application de gestion des ressources humaines

### IV.3.3. Analyse fonctionnelle :

Parmi ses fonctionnalités, les plus intéressantes :

- Gestion de pointage.
- Gestion de recrutement du personnel.
- Gestion des congés.
- Gestion de paie.

## V. Critique de l'existant :

Le tableau 4 récapitule les points forts et les pointes faibles des applications étudiées précédemment.

Applications	Points forts	Points faibles
<b>BTManager</b>	<ul style="list-style-type: none"> <li>• Possibilité de calcul des salaires.</li> <li>• Collecte des informations instantanées.</li> </ul>	<ul style="list-style-type: none"> <li>• Absence de l'authentification.</li> <li>• Intégration de la solution dans chaque poste.</li> <li>• Design manque de créativité.</li> </ul>
<b>MBG</b>	<ul style="list-style-type: none"> <li>• Espace employé et espace administration.</li> <li>• Gestion des statistiques de pointage.</li> </ul>	<ul style="list-style-type: none"> <li>• Absence de gestion des périphériques.</li> <li>• Interface n'est pas bien organisé.</li> <li>• Mauvaise vision.</li> </ul>
<b>Application de Gestion des Ressources humaines</b>	<ul style="list-style-type: none"> <li>• Accès à une fiche de paie.</li> <li>• Possibilité de recruter du personnel en ligne.</li> </ul>	<ul style="list-style-type: none"> <li>• Absence de gestion des périphériques.</li> <li>• Manque de modification des informations.</li> <li>• Design très simple en noir et blanc.</li> </ul>

Tableau 4: Critique de l'existant

## VI. Solution proposée :

Afin de surmonter les problèmes observés dans la partie précédente, on a recouru à une solution qui est le développement d'une application web performante et facile à utiliser dédiée aux sociétés qui veulent améliorer leur productivité dans la gestion de temps de travail.

### VI.1. Volet graphique :

- Design des interfaces montre la créativité et la modernité.
- Des interfaces non chargées et bien organisées.
- Contenu net et compréhensible.

## **VI.2. Volet technique :**

- Intégration dans n'importe quel navigateur.
- Application monopage.
- Offrir aux utilisateurs de l'application un livrable facile à manipuler.
- Possibilité aux utilisateurs de s'authentifier.
- Communication avec la pointeuse en temps réel.

## **Conclusion :**

Ce chapitre introductif avait pour but de présenter l'environnement de notre travail : présentation de la société, description de la problématique, les objectifs à atteindre et l'approche méthodologique choisie. Par la suite, on a étudié l'existant à l'aide de la société pour dégager les points faibles et enfin proposer des nouvelles solutions plus optimisées.

---

## **Chapitre 2 : Analyse et spécification des besoins**

---

## **Introduction :**

Nous allons étudier dans ce chapitre l'analyse fonctionnelle et non fonctionnelle des besoins et la gestion de notre projet avec la méthodologie SCRUM.

Dans l'analyse fonctionnelle, nous cherchons les fonctions offertes par l'application pour satisfaire les besoins du client.

Dans l'analyse non fonctionnelle, nous dégagons les besoins caractérisant le système en terme de performance et de sécurité.

## **I. Identification des besoins :**

Dans cette partie, nous allons identifier les exigences du client : les besoins fonctionnels et les besoins non fonctionnels de l'application.

### **I.1. Identification des besoins fonctionnels :**

Les besoins fonctionnels sont des besoins obligatoires, ce sont les fonctionnalités que l'application doit offrir aux utilisateurs pour satisfaire leurs besoins. Pour identifier ces besoins, nous allons d'abord commencer par l'identification des acteurs puis l'identification des cas d'utilisation et on finit par montrer l'interaction entre les acteurs et leurs cas d'utilisation.

#### **I.1.1. Identification des acteurs et des cas d'utilisation :**

L'acteur [N2] est celui qui déclenche le cas et active le système. Il interagit directement avec le système. On distingue deux différents type d'acteur :

- Les acteurs primaires sont le super administrateur, le responsable, le gestionnaire et l'employé.
- L'acteur secondaire est le système de gestion des employés.

Un cas d'utilisation [N2] est la manière d'utiliser le système de point de vue de l'acteur. Il permet de décrire les besoins fonctionnels du client.

On distingue différents cas d'utilisations dans notre système qui doivent répondre à l'idée générale du projet : La gestion de temps de travail.

Le tableau 5 représente les principaux acteurs et leurs cas d'utilisation.

Acteur	Cas d'utilisation
<p><b><u>Super administrateur</u></b> : c'est la personne possédant le privilège de plus haut niveau. Il est capable de manipuler toutes les fonctionnalités.</p>	<ul style="list-style-type: none"> <li>- S'authentifier : pour accéder à son profil et aux fonctionnalités de l'application. Il doit introduire son identifiant et son mot de passe.</li> <li>- Consulter son profil.</li> <li>- Gérer les comptes : soit par ajout, ou par suppression. Il doit gérer les comptes des utilisateurs de l'application.</li> </ul>
<p><b><u>Responsable</u></b> : c'est la personne qui peut superviser les heures de présence des employés</p>	<ul style="list-style-type: none"> <li>- S'authentifier : pour accéder à son profil et pour suivre la présence des employés. Il doit introduire son identifiant et son mot de passe.</li> <li>- Consulter son profil.</li> <li>- Consulter les heures de présence des employés : à chaque fois qu'un employé pointe on peut accéder à ses informations via l'application web en temps réel afin de gérer son pointage.</li> <li>- Imprimer la liste des employés qui ont pointé.</li> <li>- Chercher des employés.</li> </ul>
<p><b><u>Gestionnaire</u></b> : c'est la personne qui est capable d'enrôler les employés, de gérer les périphériques et les absences.</p>	<ul style="list-style-type: none"> <li>- S'authentifier : pour accéder à son profil et aux fonctionnalités accessibles pour lui. Il doit introduire son identifiant et son mot de passe.</li> <li>- Consulter son profil.</li> <li>- Gérer les périphériques : soit par ajout, soit par modification ou soit par suppression.</li> <li>- Gérer les types d'absences : soit par ajout, soit par modification ou soit par suppression.</li> <li>- Gérer les congés : soit par ajout ou par suppression.</li> <li>- Gérer les usagers : soit par ajout ou soit par suppression.</li> </ul>
<p><b><u>Employé</u></b> : c'est la personne qui va interagir avec la pointeuse. C'est celui qui va déclencher l'application.</p>	<ul style="list-style-type: none"> <li>- Pointer sur la pointeuse : pour pouvoir enregistrer sa présence.</li> </ul>

Tableau 5: Identification des acteurs et des cas d'utilisation



## I.1.2. Le diagramme de cas d'utilisation :

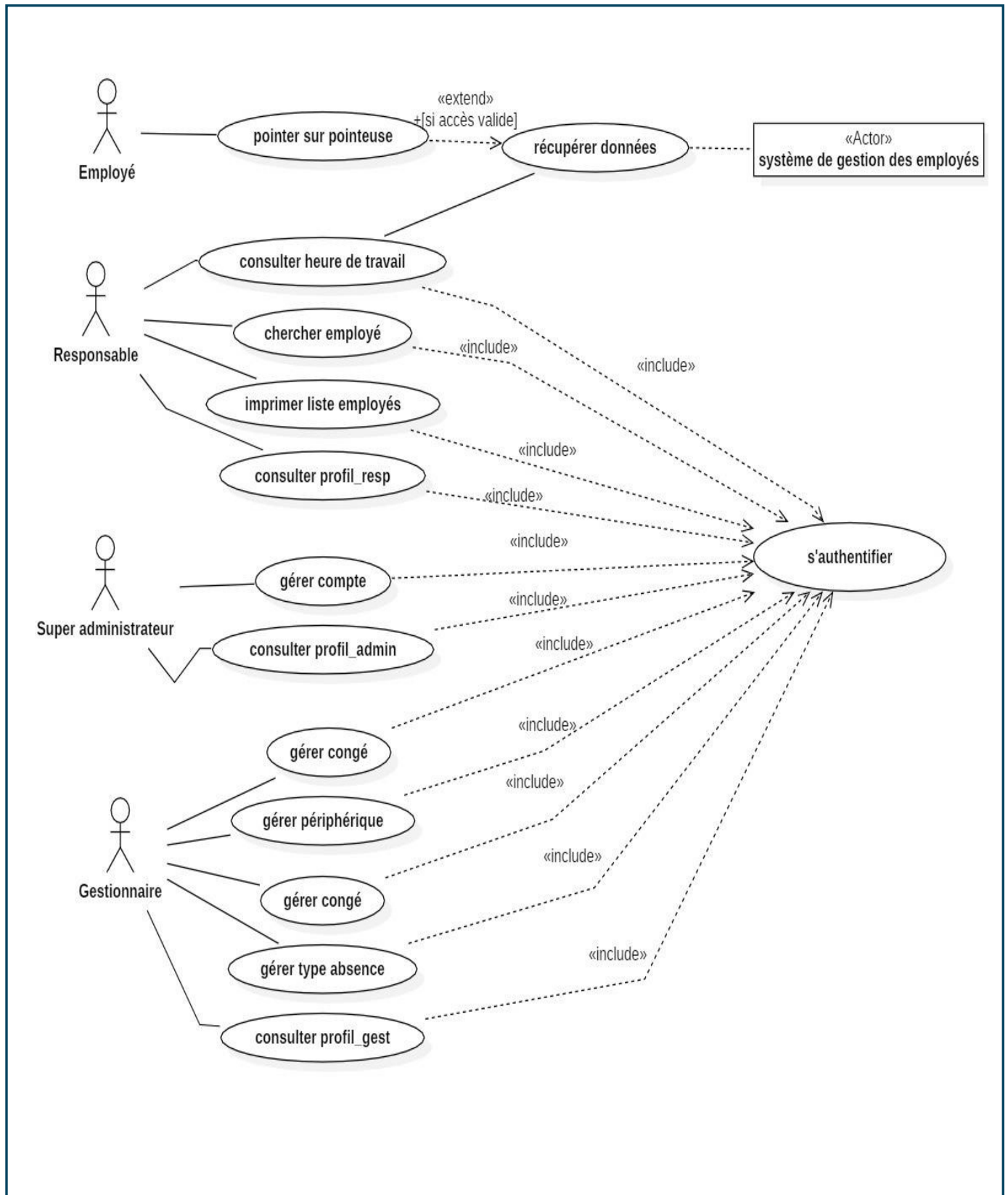


Figure 5: Diagramme de cas d'utilisation global

### I.1.3. Description textuelle des principaux cas d'utilisation :

Dans cette partie, nous allons étudier les principaux cas d'utilisation.

Cas d'utilisation 1 : S'authentifier	
<b>Acteurs principales</b> : super administrateur, responsable, gestionnaire	
<b>Objectif</b> : permettre à l'acteur de se connecter afin de pouvoir accéder aux fonctionnalités accessibles.	
Description des enchainements	
Pré condition	Post condition
L'acteur doit avoir une connexion internet et un compte.	Le cas démarre après le point 4 de l'enchainement nominal où l'acteur s'authentifie.
Scénario nominal	
<ol style="list-style-type: none"> <li>1. l'acteur lance l'application.</li> <li>2. le système affiche la page d'authentification.</li> <li>3. le système demande à l'utilisateur d'entrer son nom d'utilisateur et son mot de passe.</li> <li>4. l'acteur saisit le nom d'utilisateur et le mot de passe.</li> <li>5. le système vérifie ces informations.</li> <li>6. le système dégage le rôle de l'acteur.</li> <li>7. l'acteur est redirigé vers son profil.</li> </ol>	
Scénario alternatif	
<p>A1 : le nom d'utilisateur ou le mot de passe est incorrect. L'enchainement démarre après le point 5.</p> <ol style="list-style-type: none"> <li>1. le système informe l'acteur que les données saisies sont incorrectes et le scénario reprend au point 3 du scénario nominal.</li> </ol>	

Tableau 6: Description textuelle de "S'authentifier"

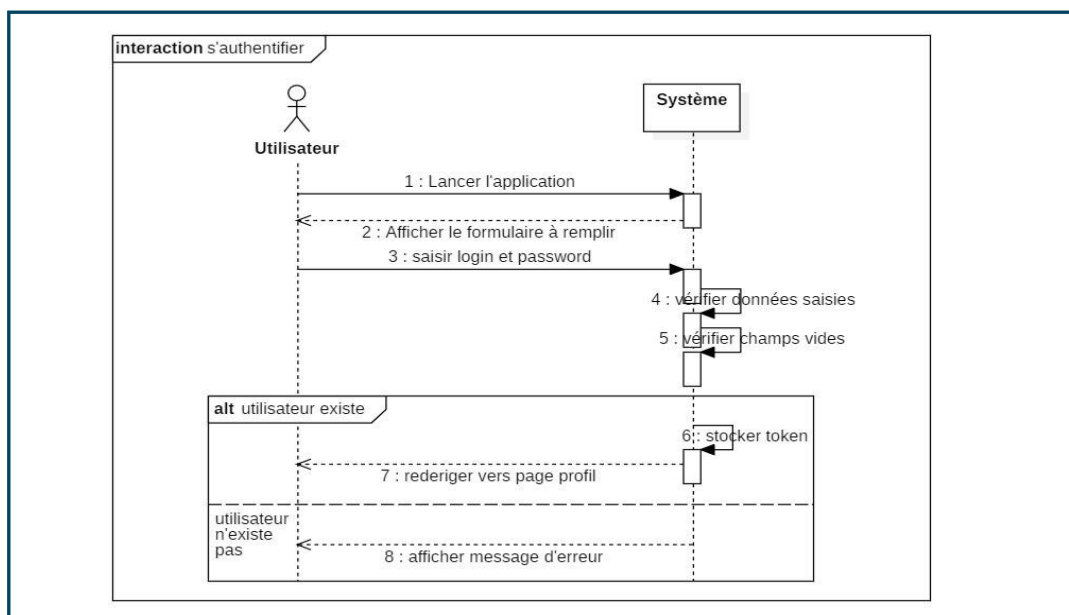


Figure 6: Diagramme de séquence système de "S'authentifier"

Le tableau 7 présente la description textuelle du cas d'utilisation Gestion des comptes.

Par la suite, la figure 7 présente le diagramme de séquence système du cas d'utilisation : Gestion des comptes.

<b>Cas d'utilisation 2 : gestion des comptes</b>	
<b>Acteurs principales</b> : super administrateur	
<b>Objectif</b> : permettre au super administrateur de créer ou de supprimer des utilisateurs.	
<b>Description des enchainements</b>	
<b>Pré condition</b>	<b>Post condition</b>
L'acteur doit avoir une connexion internet et succès d'authentification.	Le cas démarre après le point 3 de l'enchainement nominal où l'acteur choisit l'opération.
<b>Scénario nominal</b>	
<ol style="list-style-type: none"> <li>1. l'acteur choisit la page « création des comptes ».</li> <li>2. le système affiche le formulaire de création des comptes.</li> <li>3. l'acteur remplit le formulaire et clique sur le bouton créer.</li> <li>4. le système vérifie les données saisies.</li> <li>5. le système enregistre les informations dans la base de données.</li> <li>6. le système affiche la liste des comptes.</li> <li>7. l'acteur choisit l'opération supprimer.</li> <li>8. l'acteur confirme la suppression.</li> <li>9. le système supprime le compte de la base de données.</li> </ol>	
<b>Scénario alternatif</b>	
<p>A1 : les informations sont manquantes ou incorrectes. L'enchainement démarre après le point 4.</p> <ol style="list-style-type: none"> <li>1. le système informe l'acteur que les données saisies sont incorrectes, garde les informations précédentes et le scénario reprend au point 3 du scénario nominal.</li> </ol>	

Tableau 7: Description textuelle de "Gestion des comptes"

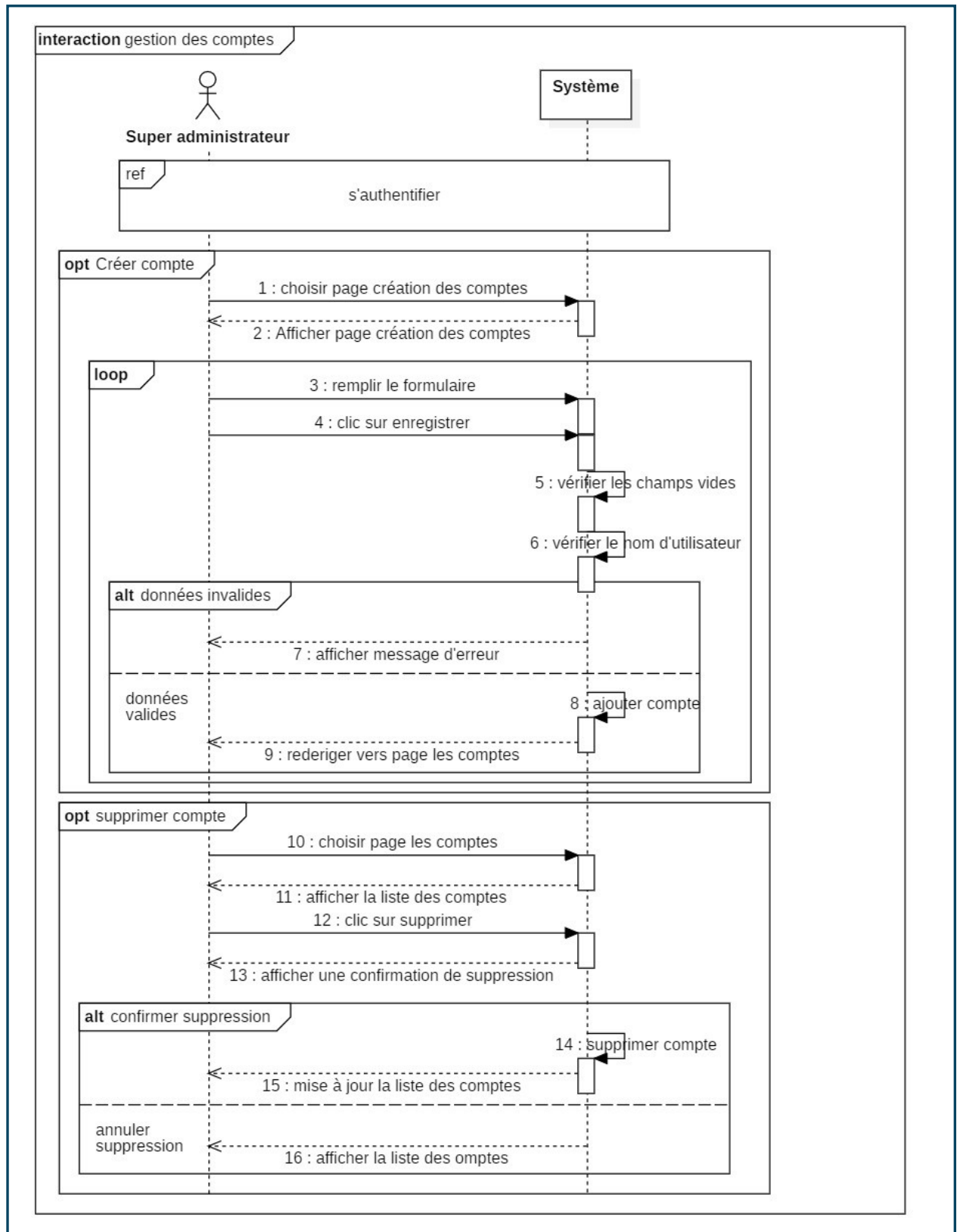


Figure 7: Diagramme séquence système de "Gestion des comptes"

Le tableau 8 présente la description textuelle du cas d'utilisation Gestion des comptes.  
Par la suite, la figure 8 présente le diagramme de séquence système du cas d'utilisation :  
Gestion des comptes.

<b>Cas d'utilisation 3: gestion des usagers</b>	
<b>Acteurs principales :</b> gestionnaire	
<b>Objectif :</b> permettre à l'acteur d'ajouter ou de supprimer un usager.	
<b>Description des enchainements</b>	
<b>Pré condition</b>	<b>Post condition</b>
L'acteur doit avoir une connexion internet et succès d'authentification.	Le cas démarre après le point 5 de l'enchainement nominal où l'acteur ajoute l'usager.
<b>Scénario nominal</b>	
<ol style="list-style-type: none"> <li>1. l'acteur choisit la page « gestion des usagers ».</li> <li>2. le système affiche la page correspondante.</li> <li>3. l'acteur choisit une des trois opérations (ajout, suppression).</li> <li>4. l'acteur remplit le formulaire affiché.</li> <li>5. le système vérifie le type des données saisies.</li> <li>6. le système effectue l'opération et enregistre.</li> </ol>	
<b>Scénario alternatif</b>	
<p>A1 : les données saisies ne sont pas valides. L'enchainement démarre après le point 5.</p> <ol style="list-style-type: none"> <li>1. le système informe l'acteur que les données saisies sont incorrectes, garde les informations précédentes et le scénario reprend au point 4 du scénario nominal.</li> </ol>	

Tableau 8: Description textuelle de cas "Gestion des usagers"

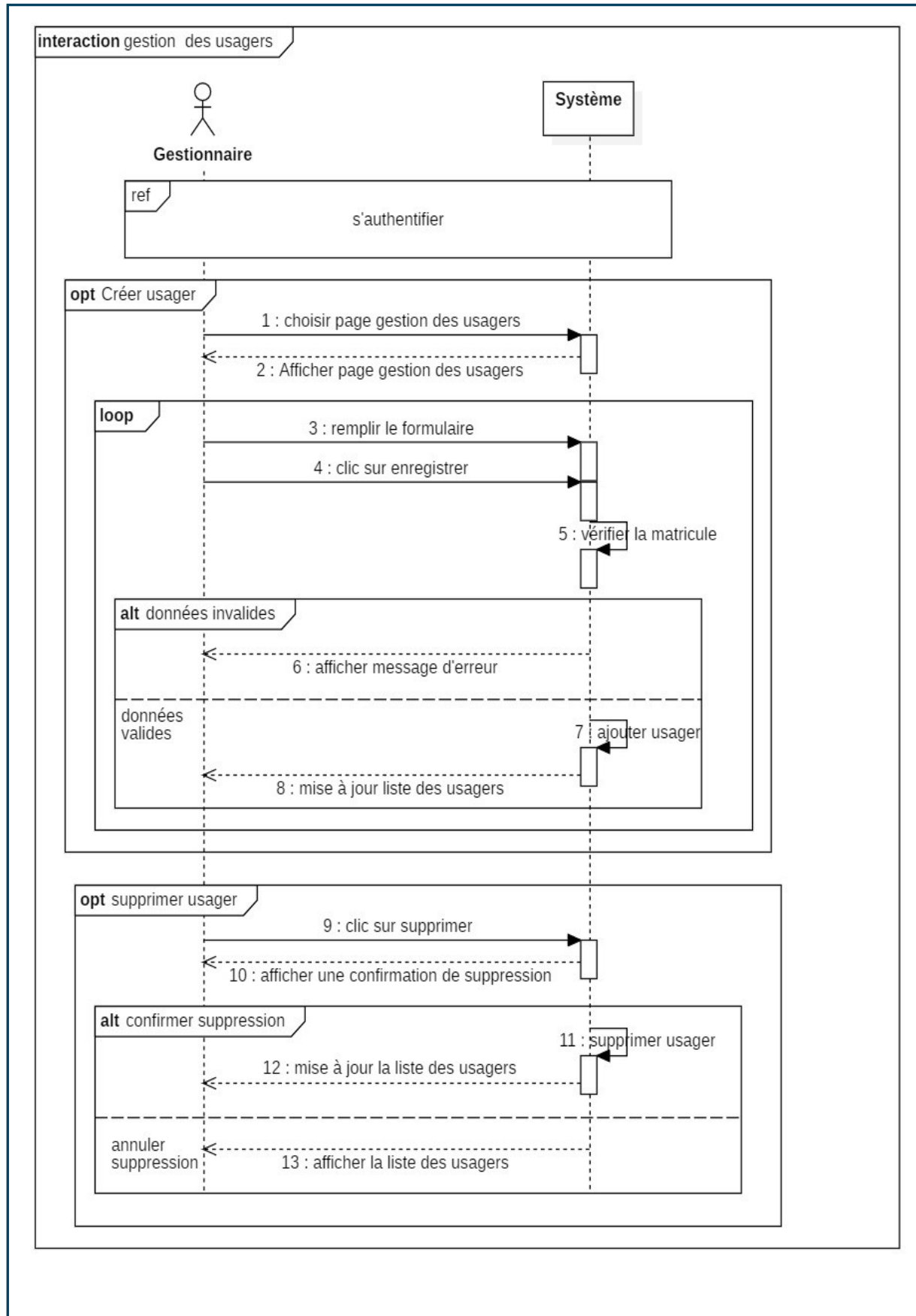


Figure 8: Diagramme séquence système de "Gestion des usagers"

Le tableau 9 présente la description textuelle du cas d'utilisation Gestion de type d'absence. Par la suite, la figure 9 présente le diagramme de séquence système du cas d'utilisation : Gestion de type d'absence.

<b>Cas d'utilisation 4: gestion de type d'absence</b>	
<b>Acteurs principales :</b> gestionnaire	
<b>Objectif :</b> permettre au gestionnaire d'ajouter, de modifier ou de supprimer un type d'absence.	
<b>Description des enchainements</b>	
<b>Pré condition</b>	<b>Post condition</b>
L'acteur doit avoir une connexion internet et succès d'authentification.	Le cas démarre après le point 5 de l'enchainement nominal où l'acteur ajoute l'utilisateur.
<b>Scénario nominal</b>	
<ol style="list-style-type: none"> <li>1. l'acteur choisit la page « gestion de type d'absence ».</li> <li>2. le système affiche la page correspondante.</li> <li>3. l'acteur choisit une des trois opérations (ajout, modification, suppression).</li> <li>4. l'acteur remplit le formulaire affiché.</li> <li>5. le système vérifie le type des données saisies.</li> <li>6. le système effectue l'opération et enregistre.</li> </ol>	
<b>Scénario alternatif</b>	
<p>A1 : les données saisies ne sont pas valides. L'enchainement démarre après le point 5.</p> <ol style="list-style-type: none"> <li>1. le système informe l'acteur que les données saisies sont incorrectes, garde les informations précédentes et le scénario reprend au point 4 du scénario nominal.</li> </ol>	

Tableau 9: Description textuelle de cas "Gestion de type d'absence"

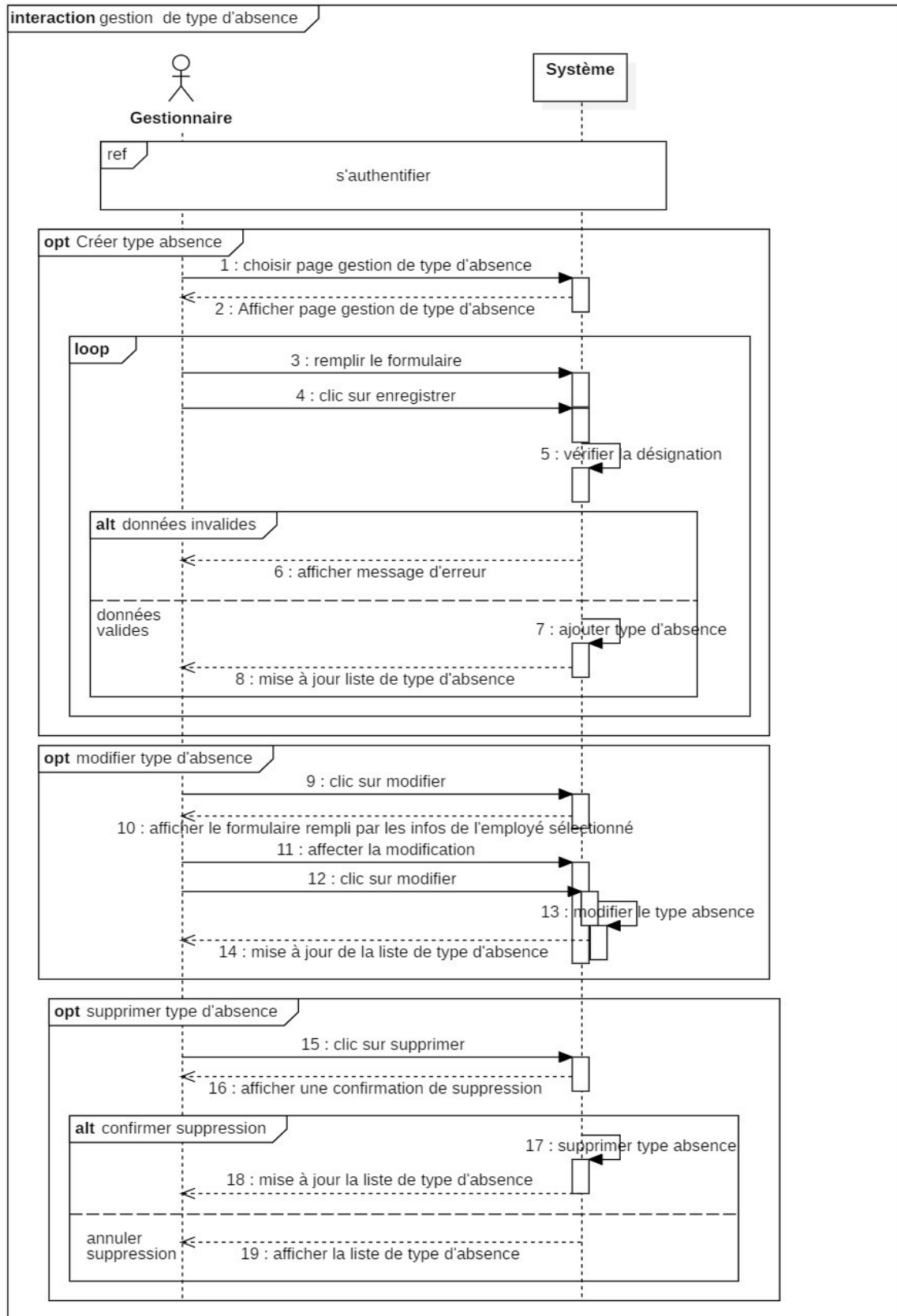


Figure 9: Diagramme de séquence système de "Gestion de type d'absence"



<b>Cas d'utilisation 5 : consulter les heures de présence</b>	
<b>Acteurs principales</b> : responsable.	
<b>Objectif</b> : permettre au responsable de consulter les heures de présence des employés.	
<b>Description des enchainements</b>	
<b>Pré condition</b>	<b>Post condition</b>
L'acteur doit avoir une connexion internet et succès d'authentification.	Le cas démarre après le point 1 de l'enchainement nominal.
<b>Scénario nominal</b>	
1. l'acteur choisit consulter horaire présence. 2. Le système affiche la liste des informations des employés qui ont pointé. 3. l'acteur choisit l'opération. 4. le système effectue l'opération.	

Tableau 10: Description textuelle de cas "Consulter les heures de présence"

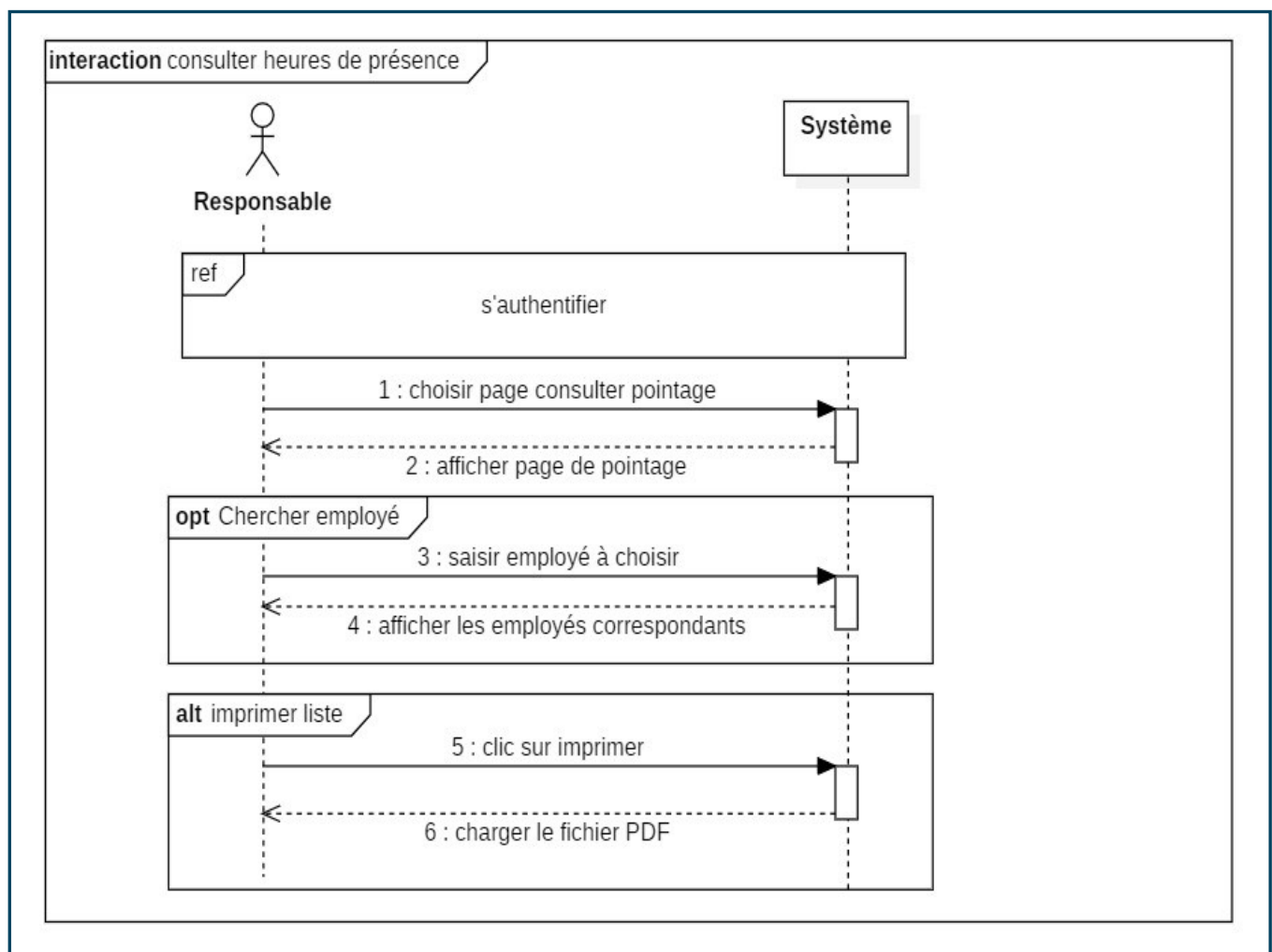


Figure 10: Diagramme de séquence système de "Consulter les heures de présence"

Le tableau 11 présente la description textuelle du cas d'utilisation Récupérer les données de pointage qui a été réalisé avec une communication entre les sockets.

Par la suite, la figure 11 présente le diagramme de séquence système du cas d'utilisation : Récupérer les données de pointages.

<b>Cas d'utilisation 6 : Récupérer les données de pointage</b>	
<b>Acteurs principales :</b> système de gestion des employés.	
<b>Objectif :</b> permettre au socket serveur de récupérer données du socket client.	
<b>Description des enchainements</b>	
<b>Pré condition</b>	<b>Post condition</b>
On doit mettre en place une connexion internet.	Le cas démarre après le point 8 de l'enchainement nominal.
<b>Scénario nominal</b>	
<ol style="list-style-type: none"> <li>1. Créer socket serveur.</li> <li>2. Attacher port au socket serveur.</li> <li>3. Créer socket client.</li> <li>4. Attacher port au socket client.</li> <li>5. Attacher l'adresse IP au socket client.</li> <li>6. lancer le serveur sur l'écoute.</li> <li>7. le client demande l'ouverture d'une connexion.</li> <li>8. le serveur accepte la connexion.</li> <li>9. le client écrit des données.</li> <li>10. le serveur reçoit le flux.</li> <li>11. le serveur enregistre le flux dans la base de données.</li> <li>12. le serveur ferme la connexion.</li> </ol>	
<b>Scénario alternatif</b>	
<p>A1 : La connexion n'est pas acceptée. L'enchainement démarre après le point 7.</p> <ol style="list-style-type: none"> <li>1. le système informe le client que le temps de d'ouverture d'une connexion a été dépassé et le scénario reprend au point 7 du scénario nominal.</li> </ol>	

Tableau 11: Description textuelle de cas " Récupérer les données de pointage "

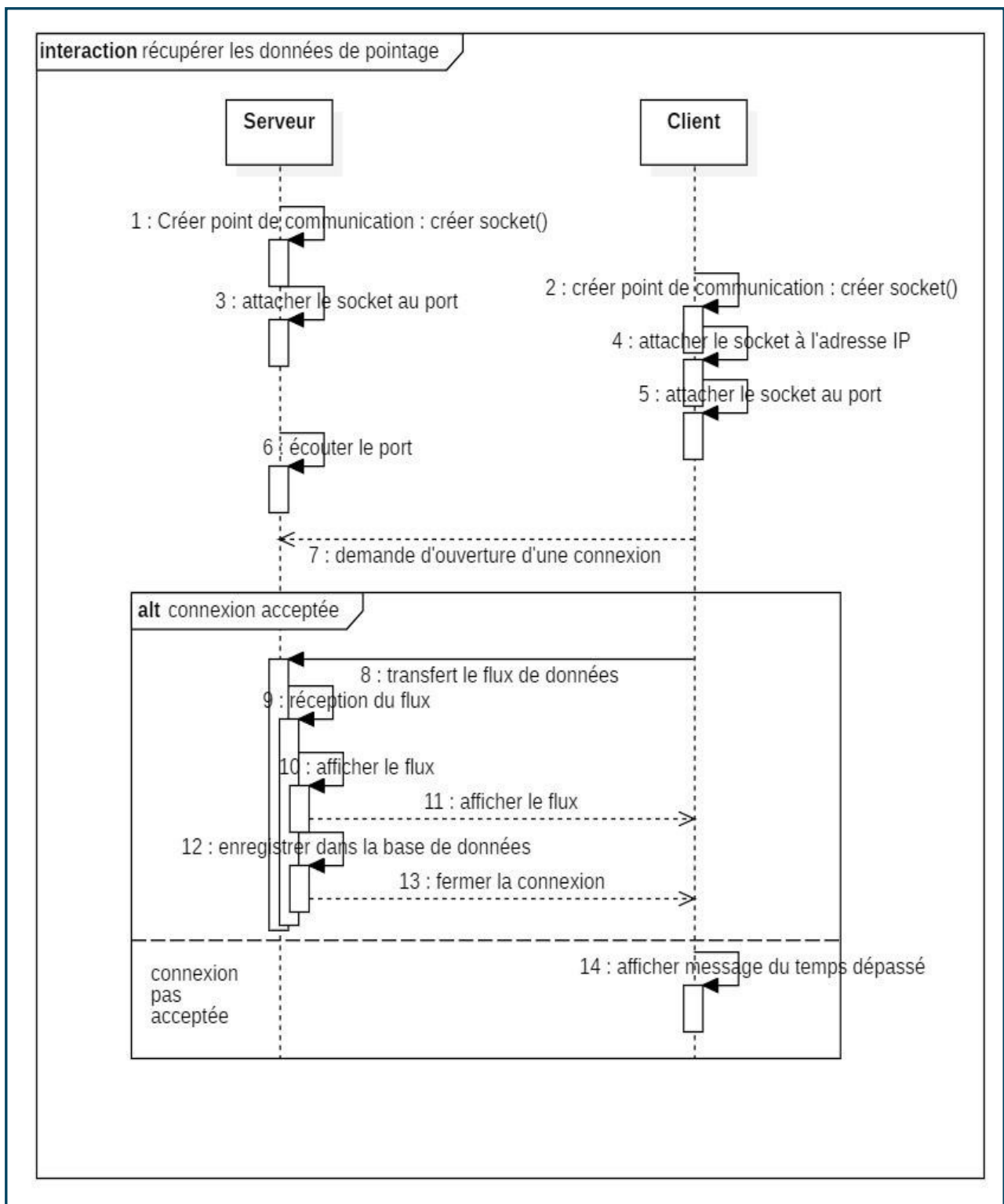


Figure 11: Diagramme de séquence système de "Récupérer données de pointage"

## **I.2. Identification des besoins non fonctionnels :**

La qualité d'une application est définie par sa capacité à satisfaire les besoins de l'utilisateur. C'est le facteur le plus important dans notre projet : avoir un produit performant avec les meilleures fonctionnalités. Pour cela, on va se baser sur la norme ISO 9126 pour définir les principales règles à appliquer. Donc, nous définissons les caractéristiques qualité attendu d'un logiciel :

- Disponibilité : capacité d'un logiciel à accomplir sa fonction dans des conditions données.
- Ergonomie : Le but de l'ergonomie est d'adapter l'environnement logiciel et matériel au visiteur. L'application doit avoir des interfaces bien lisibles pour faciliter l'utilisation.
- Fiabilité : capacité d'un logiciel accomplir sans défaillance les fonctions à faire, à fonctionner même dans des mauvaises conditions sans mettre en cause les informations du système. Enfin, il doit être tolérant aux pannes.
- Flexibilité : facilité d'ajouter, supprimer ou modifier dans un programme opérationnel sans causer des problèmes.
- Intégrité : aptitude d'un logiciel à être protégé contre un accès des utilisateurs non autorisés.
- Maintenabilité : facilité à identifier et corriger un défaut ou à transformer le logiciel.
- Validité : réponse aux exigences du client.
- La performance : il comporte plusieurs termes tel que temps de réponse, débit, fluidité.

## **II. Pilotage du projet avec SCRUM :**

Scrum[N3] est une méthodologie agile dédiée à gestion des projets là où le travail est en équipe.

Le développement se fait en un ensemble d'itérations appelées sprints qui sont des tâches réalisables par l'équipe en un temps donné.

Nous avons réparti les rôles comme suit :

- ✓ L'équipe : qui travaille chaque jour et qui a une responsabilité de délivrer à la fin de chaque sprint. L'équipe est constituée par Safa Kassous et Amir Horchani.
- ✓ Le Scrum Master : c'est la personne qui aide à supprimer les obstacles et qui veille l'avancement sur le tableau des tâches : Monsieur Mohamed Karim Azizi.

- ✓ Le Product Owner : c'est la personne qui est chargée par la tenue du backlog du produit. C'est monsieur Ahmed Hassini.

## II.1. Backlog du produit :

La backlog du produit[N4] est l'artefact le plus important dans le processus Scrum puisqu'il contient une liste des fonctionnalités attendues d'un produit et les éléments nécessaires pour l'élaboration du rapport.

Dans notre cas, le cas d'utilisation le moins prioritaire a la priorité 1.

Thème	User story	Priorité	Sprint
Gérer utilisateurs	En tant qu’administrateur, je souhaite créer un nouvel utilisateur.	2	Sprint 1 ( 14 jours )
	En tant qu’administrateur, je souhaite afficher la liste des utilisateurs.		
	En tant qu’administrateur, je souhaite supprimer un utilisateur bien déterminé.		
S’authentifier	En tant qu’utilisateur, je souhaite m’authentifier afin d’accéder à mon espace personnel.	3	
Attribuer rôle	En tant qu’administrateur, je souhaite attribuer un rôle au nouvel utilisateur.	3	
Gérer profil	En tant qu’utilisateur, je souhaite consulter mon profil.	1	
Gérer type d’absence	En tant que gestionnaire, je souhaite créer un nouveau type d’absence.	2	Sprint 2 ( 7 jours )
	En tant que gestionnaire, je souhaite modifier un type d’absence bien déterminé.		
	En tant que gestionnaire, je souhaite supprimer un type d’absence bien déterminé		
	En tant que gestionnaire, je souhaite consulter la liste de type d’absence.		
Gérer congé	En tant que gestionnaire, je souhaite ajouter un congé.	2	Sprint 3

	En tant que, je souhaite consulter la liste des congés.		(5 jours)
	En tant que gestionnaire, je souhaite supprimer un congé bien déterminé.		
Gérer périphérique	En tant que gestionnaire, je souhaite ajouter un périphérique.		
	En tant que gestionnaire, je souhaite consulter la liste des périphériques.		
	En tant que gestionnaire, je souhaite modifier un périphérique bien déterminé.	2	Sprint 4 (7 jours)
	En tant que gestionnaire, je souhaite supprimer un périphérique bien déterminé.		
Gérer usager	En tant que gestionnaire, je souhaite ajouter un usager.		
	En tant que gestionnaire, je souhaite consulter la liste des usagers.		
	En tant que gestionnaire, je souhaite supprimer un usager bien déterminé.	2	Sprint 5 (6 jours)
Consulter heure de présence	En tant que responsable, je souhaite consulter la liste des employés qui ont pointé.		
	En tant que responsable, je souhaite chercher un employé bien déterminé.	3	Sprint 6 (14 jours)
	En tant que responsable, je souhaite imprimer liste des employés.		
Traiter les données de pointage	En tant que système de gestion des employés, je souhaite récupérer les données de la pointeuse.		
	En tant que système de gestion des employés, je souhaite afficher les données de pointage dans l'application en temps réel.	3	Sprint 6 (20 jours)

Tableau 12: Backlog du produit

## II.2. Planification des sprints :

Le regroupement des tâches en sprints est une étape très importante. D'où nous avons réuni les tâches dans plusieurs sprints dans le tableau 12 précédant afin de bien organiser le travail. Dans cette partie, nous allons planifier les sprints en release d'où la réussite du projet.

<b>Release 1</b>	Sprint 1 : gestion des utilisateurs
	Sprint 2 : gestion de type d'absence
	Sprint 3 : gestion des congés
	Sprint 4 : gestion des périphériques
	Sprint 5 : gestion des usagers
<b>Release 2</b>	Sprint 6 : consultation des heures de présence
	Sprint 7 : traitement des données de pointage

Tableau 13: Planification des sprints

## Conclusion :

Dans ce chapitre, nous avons identifié les besoins fonctionnels et les besoins non fonctionnels en se basant sur la norme ISO 9126 et nous avons présenté le pilotage du projet avec la méthodologie Scrum.

---

## **Chapitre 3 : Conception**

---



## Introduction :

Après avoir éclaircir les besoins du client en les spécifiant dans le chapitre précédent, nous allons mettre maintenant l'accent sur la phase de conception. C'est une phase qui s'appuie dans le cycle de vie d'un logiciel pour développer les meilleures solutions à partir d'un besoin donné. Dans cette partie, nous allons donc présenter la vue statique à travers le diagramme de classe et la vue dynamique à travers les diagrammes de séquences détaillés.

### I. Conception technique :

Avant tout développement, on doit d'abord étudier notre vue statique par énumération des objets et les relations entre eux.

#### I.1. Description de la vue statique :

Dans cette partie, nous mettons en évidence la structure statique de notre projet en recourant au diagramme de classes pour détailler les différentes associations entre les objets mis en œuvre.

##### I.1.1. Diagramme de classe :

Le diagramme de classe[N5] est considéré comme le point central dans un développement orienté objet. Il permet de bien décrire la structure des entités manipulées par les utilisateurs.

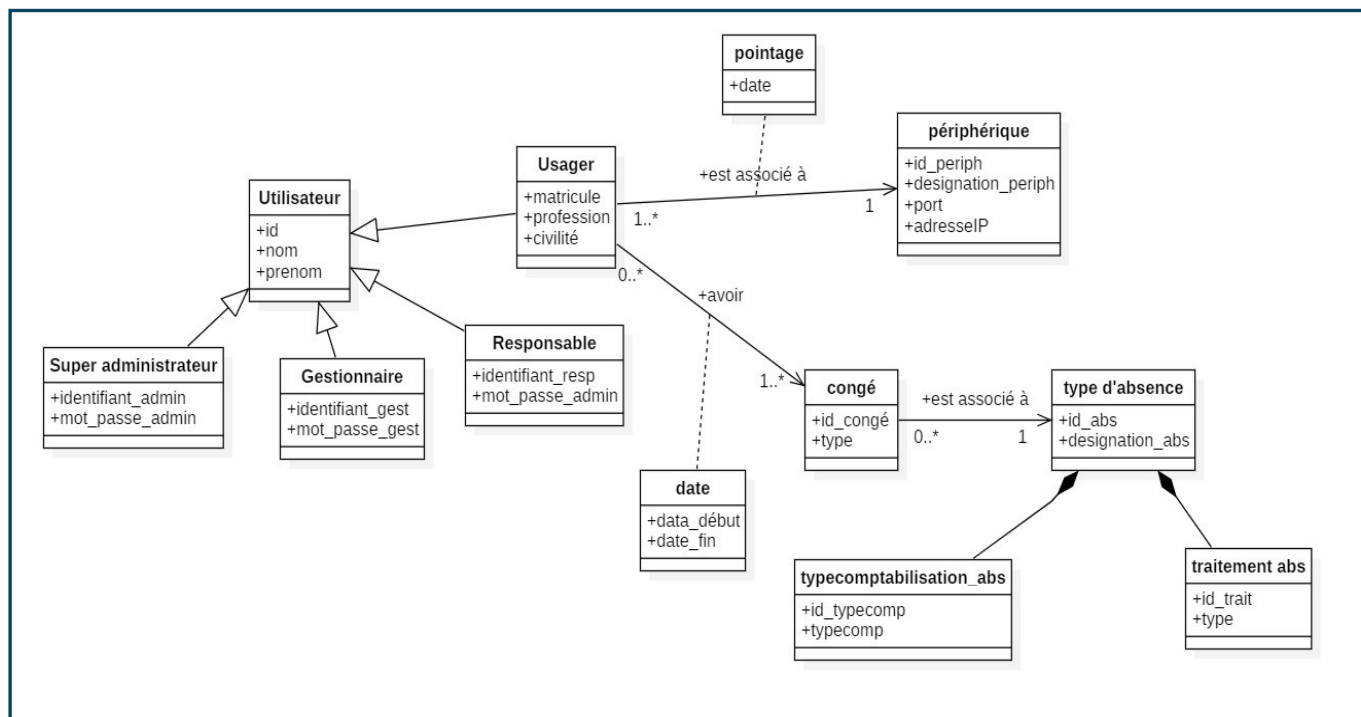


Figure 12: Diagramme de classe

### I.1.2. Dictionnaire des données :

Chaque classe de notre projet a plusieurs attributs de différentes natures. Le tableau 14 décrit les classes de notre projet.

<b>Utilisateur</b> : La table utilisateur contenant toutes les informations nécessaires pour construire nos acteurs, chaque utilisateur peut avoir un seul rôle et qui peut être : administrateur, responsable gestionnaire et usager.		
Code	Désignation	type
id	Identifiant de l'utilisateur	String
Nom	Nom de l'utilisateur	String
Prénom	Prénom de l'utilisateur	String
<b>Super administrateur</b> : la table super administrateur est un héritage de la table utilisateur en spécifiant le rôle de l'utilisateur.		
Identifiant_admin	Login de l'administrateur	String
Mot_passe_admin	Mot de passe de l'administrateur	String
<b>Gestionnaire</b> : la table gestionnaire est un héritage de la table utilisateur en spécifiant le rôle gestionnaire.		
Identifiant_gest	Login de gestionnaire	String
Mot_passe_gest	Mot de passe de gestionnaire	String
<b>Responsable</b> : la table responsable est un héritage de la table utilisateur en spécifiant le rôle responsable.		
Identifiant_resp	Login de responsable	String
Mot_passe_resp	Mot de passe de Responsable	String
<b>Usager</b> : la table usager est un héritage de la table utilisateur. Elle contient les informations des employés qui seront ajoutés par l'administrateur.		
Matricule	Matricule de l'employé	Int
Profession	Profession de l'employé	String
Civilité	Civilité de l'employé	String
<b>Périphérique</b> : la table périphérique contient les informations des périphériques (pointeuses) qui seront ajoutés par le gestionnaire.		
Id_periph	Identifiant de périphérique	String
Designation_periph	Désignation de Périphérique	String
Port	Port de périphérique	Int
AdresseIP	AdresseIP de périphérique	string
<b>Congé</b> : la table congé contenant les informations du congé associé à un employé		
Id_congé	Identifiant du congé	String
Type	Type de l'absence	String

<b>Type absence</b> : la table type d'absence contient le type de l'absence d'un employé		
Id_abs	Identifiant de l'absence	String
Designation_abs	Désignation de l'absence	String
<b>Typecomptabilisation_abs</b> : table contient la liste des type de comptabilisation dans un type d'absence donné.		
Id type_comp	Identifiant du type	String
Typecomp	Type de comptabilisation De l'absence	String
<b>Traitement_abs</b> : table contient la liste de type de traitement d'un type d'absence donné.		
Id_trait	Identifiant du traitement	String
type	Type de traitement	String
<b>Pointage</b> : table associative qui permet de stocker la liste des employés qui ont pointé sur un périphérique avec stockage de la date de pointage.		
Date	Date de pointage	Date
<b>Date</b> : table associative permet de stocker la durée du congé d'un employé donné.		
Date debut	Date debut du congé	Date
Date fin	Date fi du congé	Date

Tableau 14: Dictionnaire des données

## I.2. Description de la vue dynamique :

Dans cette partie on va présenter l'aspect dynamique de notre application en utilisant le diagramme de séquences objets.

### I.2.1. Diagramme de séquence objet :

En se basant sur les descriptions textuelles étudiés dans le chapitre 2, nous allons présenter le diagramme de séquence objet pour présenter le déroulement chronologique de quelques cas d'utilisation comme illustrés dans les figures suivantes.

**Diagramme séquence objet de « S'authentifier » :**

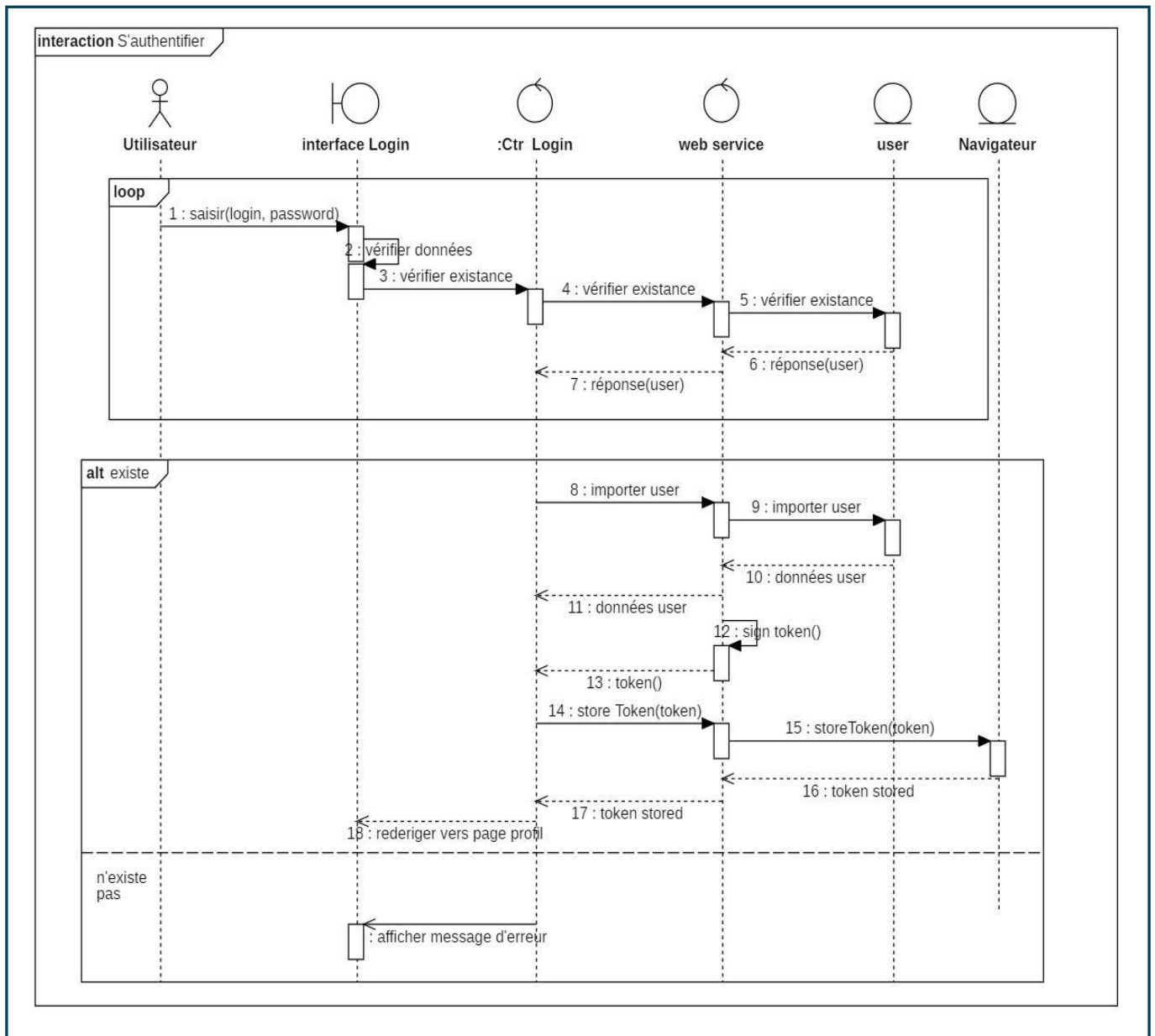


Figure 13: Diagramme de séquence objet de S'authentifier

**Diagramme séquence objet de « Gestion des comptes » :**

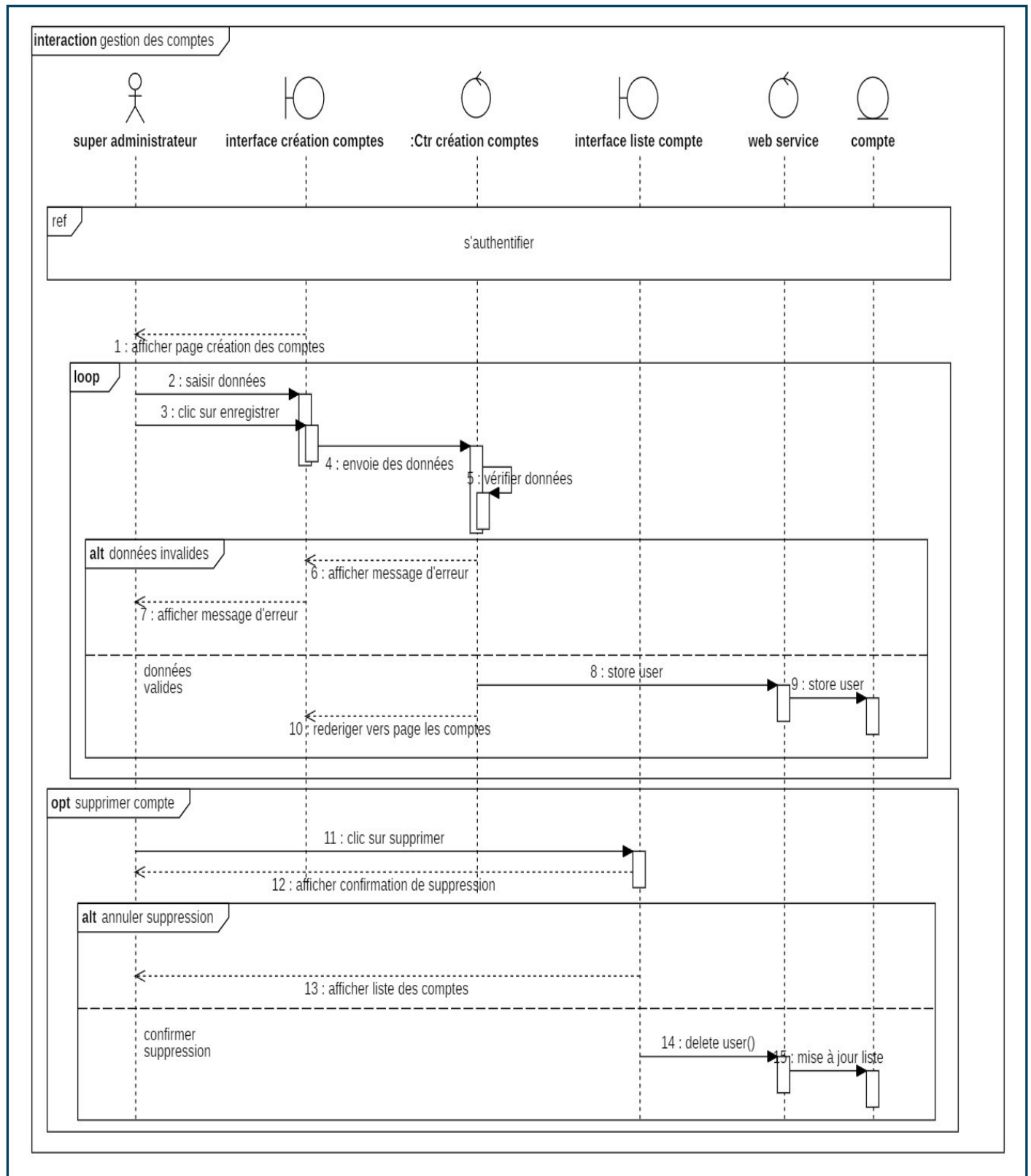


Figure 14: Diagramme de séquence objet de Gestion des comptes

## Diagramme séquence objet de « Gestion des usagers » :

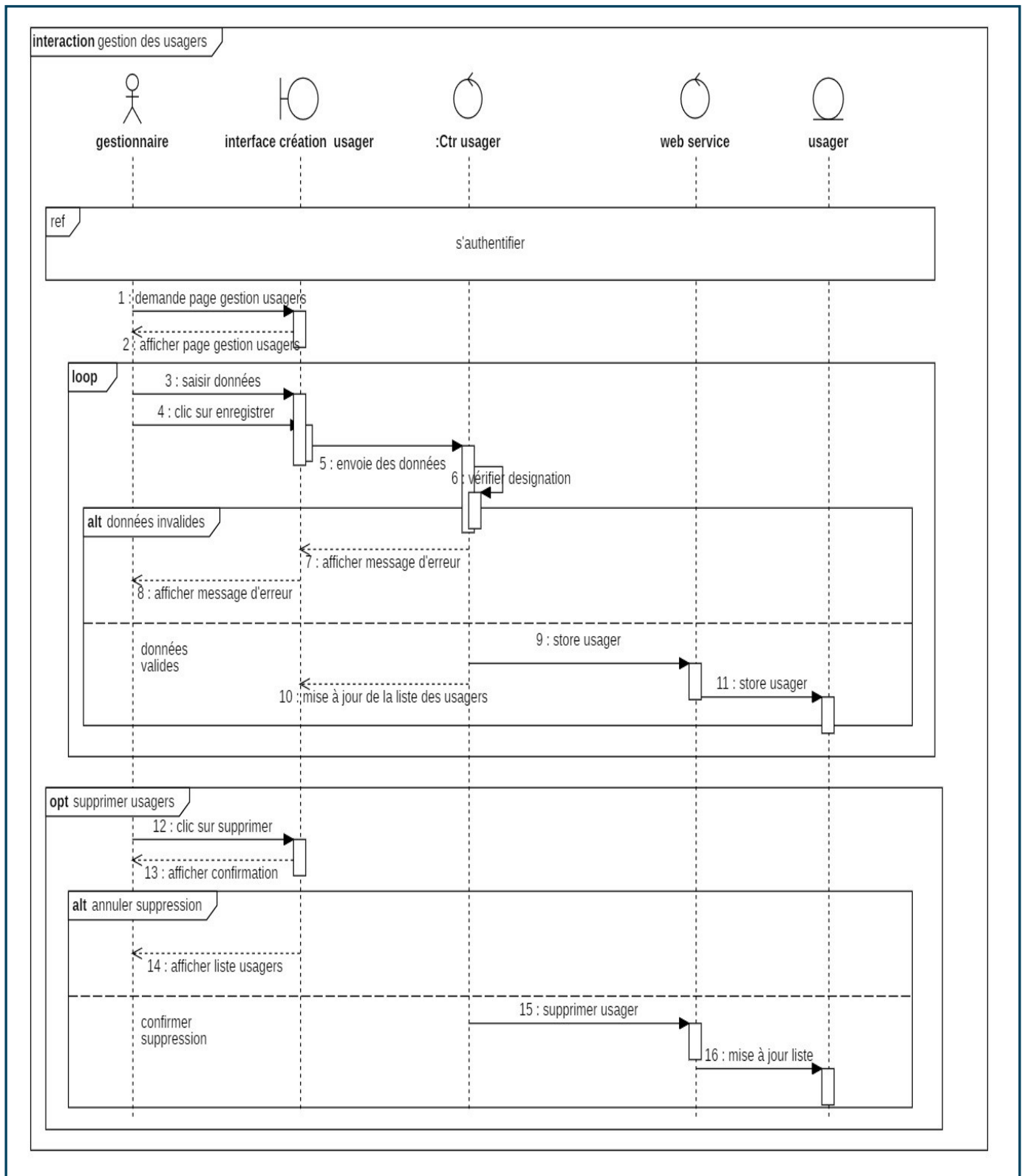


Figure 15: Diagramme de séquence objet de Gestion des usagers

## Diagramme séquence objet de « Gestion de type d'absence » :

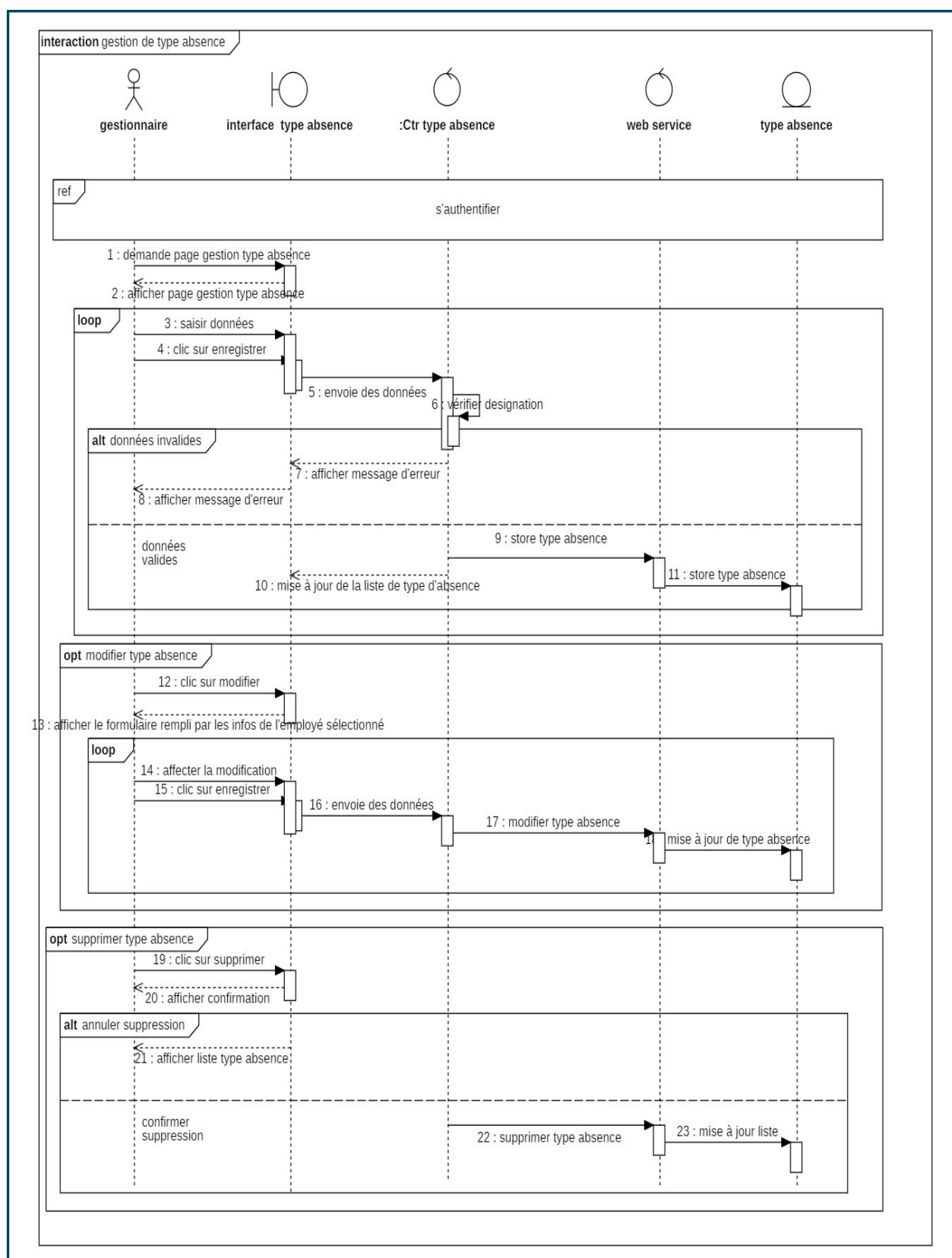


Figure 16: Diagramme de séquence objet de Gestion de type d'absence

**Diagramme séquence objet de « Consulter les heures de présence » :**

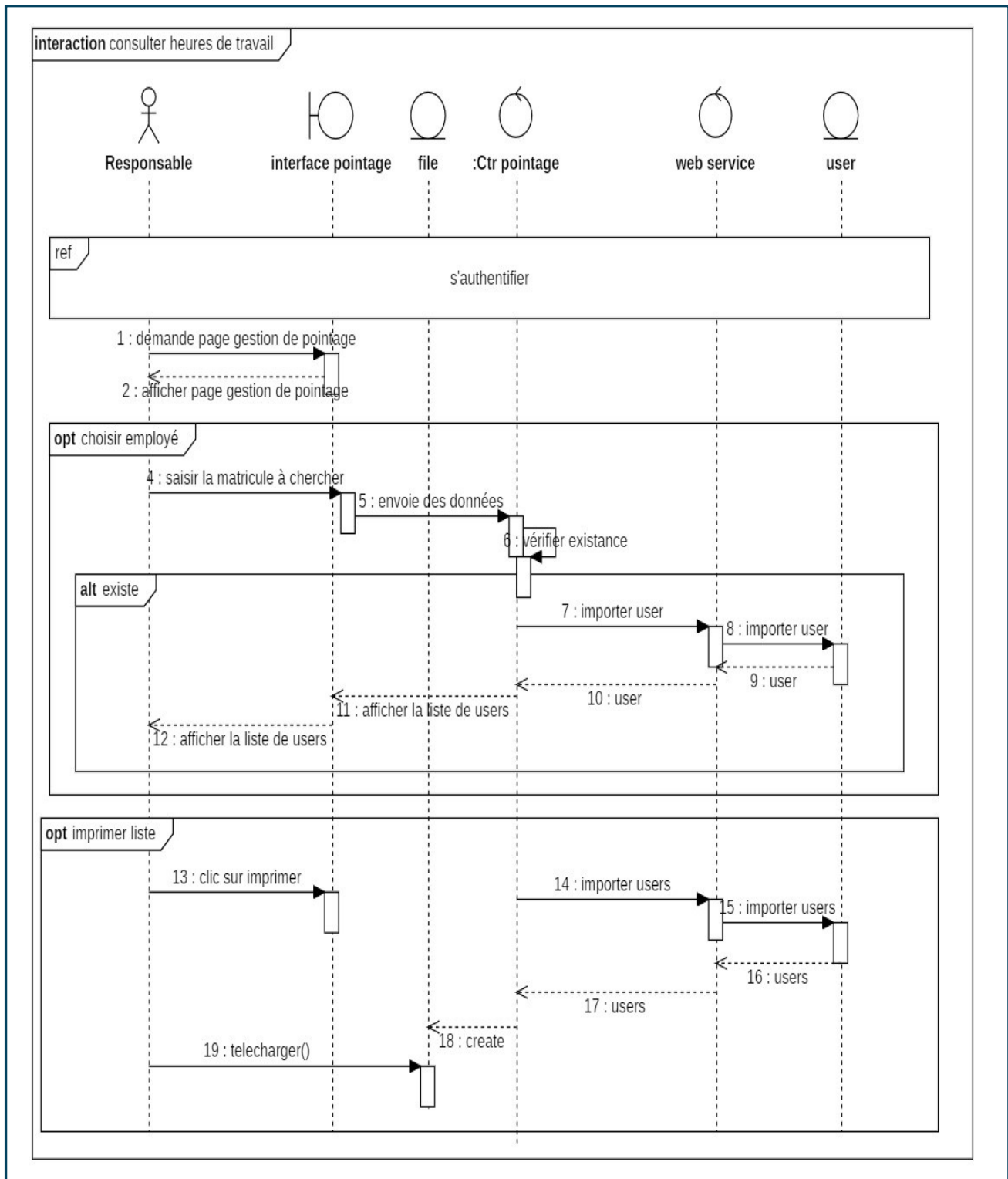


Figure 17: Diagramme de séquence objet de Consulter les heures de présence



**Diagramme séquence objet de « Récupérer données de pointage » :**

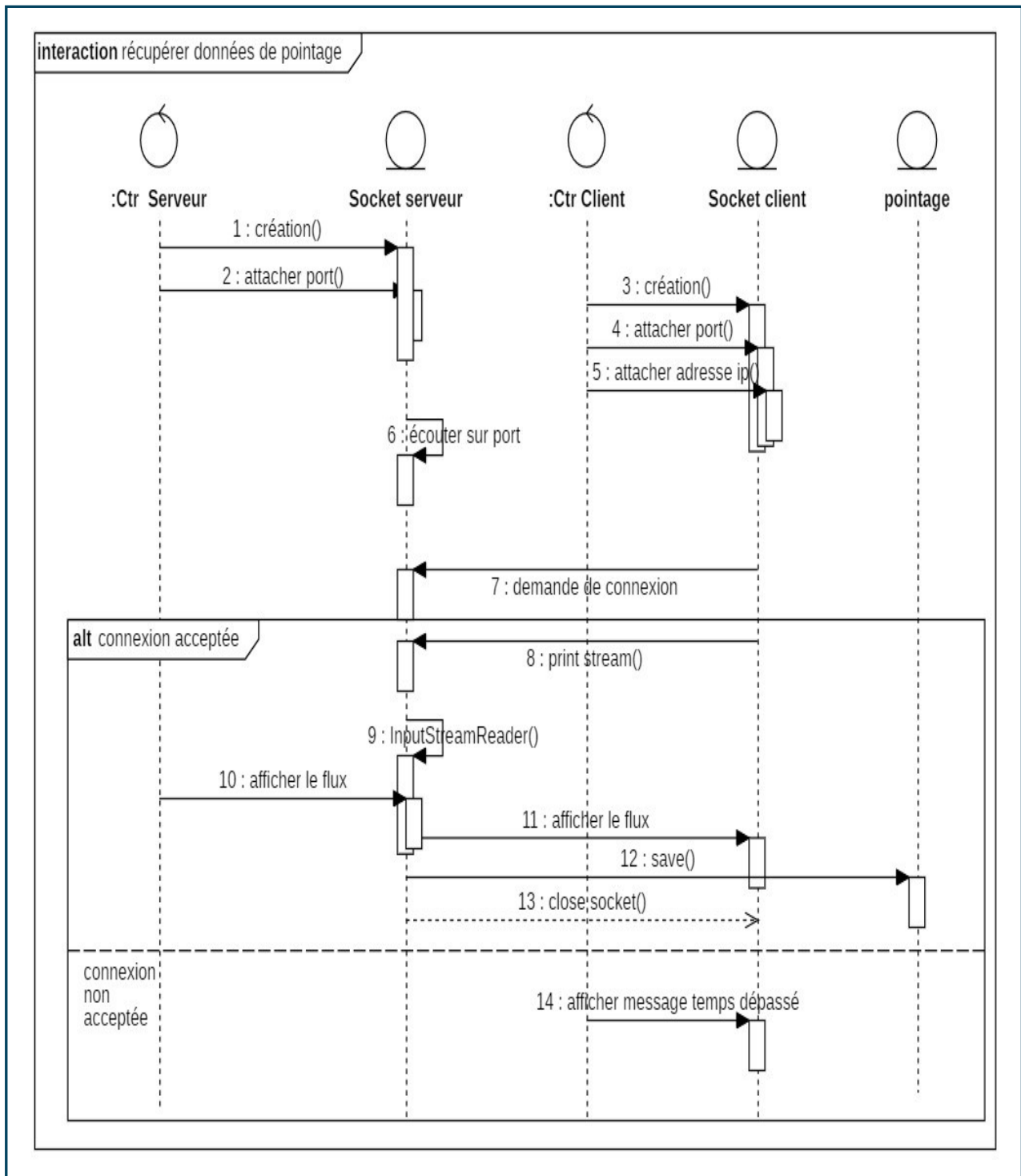


Figure 18: Diagramme de séquence objet de Récupérer les données de pointage

## II. Conception graphique :

La conception graphique est une phase primordiale dans la création des applications web. Elle permet de décrire les étapes que nous suivons pour la réalisation des interfaces de notre projet.

### II.1. Schéma de navigation :

On a adopté comme type de schéma de navigation la structure hiérarchisée : On a divisé notre projet en nœuds. On a montré les interactions entre les nœuds des utilisateurs de l'application et les nœuds constituant les interfaces.

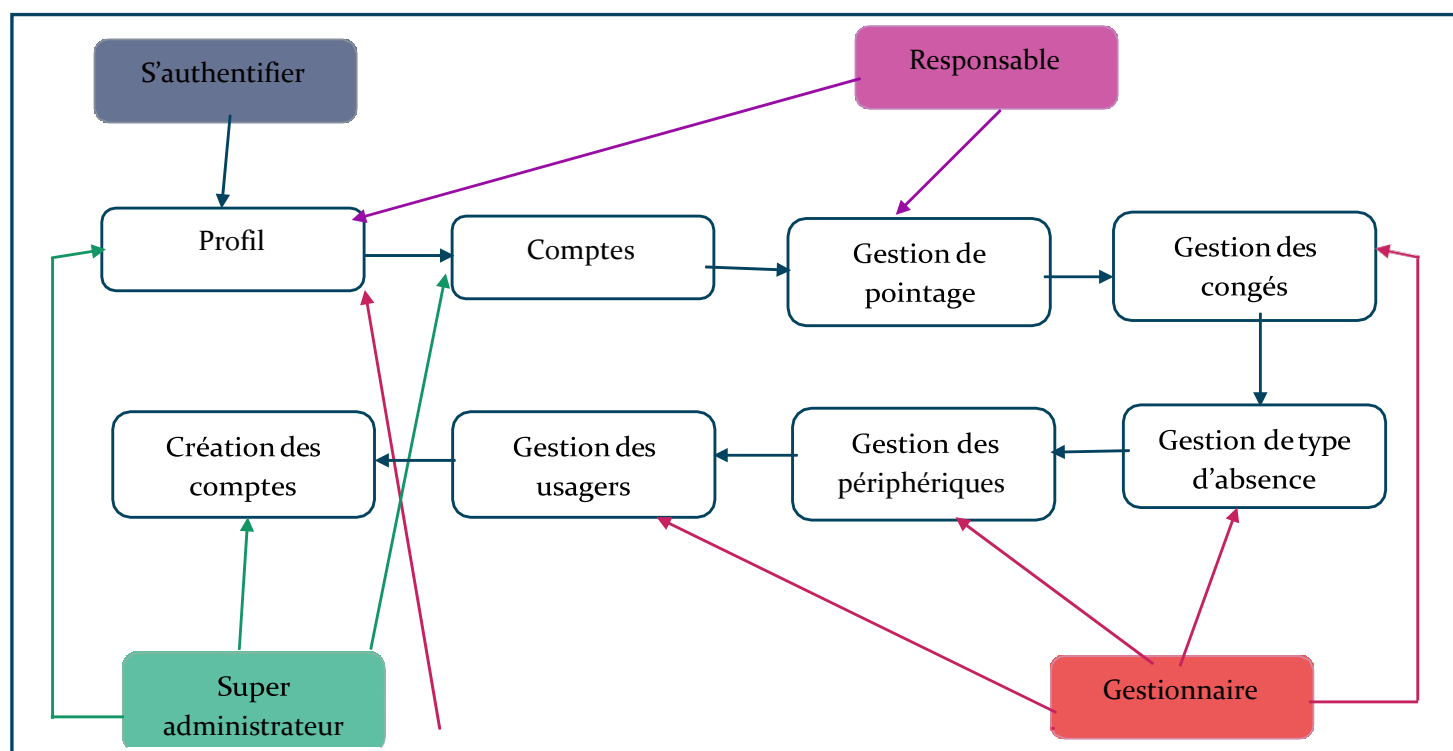


Figure 19: Schéma de navigation

### II.2. La charte graphique :

La charte graphique est un document de travail qui contient l'ensemble des règles fondamentales d'utilisation des signes graphiques et elle définit la position de ces dernières afin d'unifier le visuel de ce projet.

#### II.2.1. Police de caractère :

Les polices qui sont utilisées sur l'application sont au nombre de trois.

Chaque police spécifie des éléments précis. Pour les titres des interfaces ainsi que pour la barre de navigation initiale et les titres des formulaires, nous utilisons la police « Century ».

Pour les titres de la barre de navigation à gauche, nous utilisons la police « Calibri ». Et pour le reste, nous utilisons la police « Bahnschrift ».

### II.2.2. Palette graphique :

Le choix des couleurs dans la réalisation d'une application est une étape très importante. Il faut avoir une composition de couleurs harmonieuses afin d'établir une charte graphique qui plait visuellement et mettre en valeur les informations.

La figure 20 présente les couleurs utilisées dans les interfaces de notre application.



Figure 20: Palette des couleurs

### II.2.3. Fond de page et template :

La figure 21 représente une vision globale sur la structure des interfaces de l'application qui est composé par trois zones principales qui sont :

- **Menu horizontal** : contient la partie gestion du profil qui contient l'identifiant de l'utilisateur connecté et la partie société qui contient le nom de la société et le logo qui la représente.
- **Menu vertical** : c'est le menu principal de notre application qui contient les liens des interfaces auxquelles un utilisateur peut y accéder.
- **Container** : contient le corps de l'application. Les formulaires à remplir ou les informations à consulter.

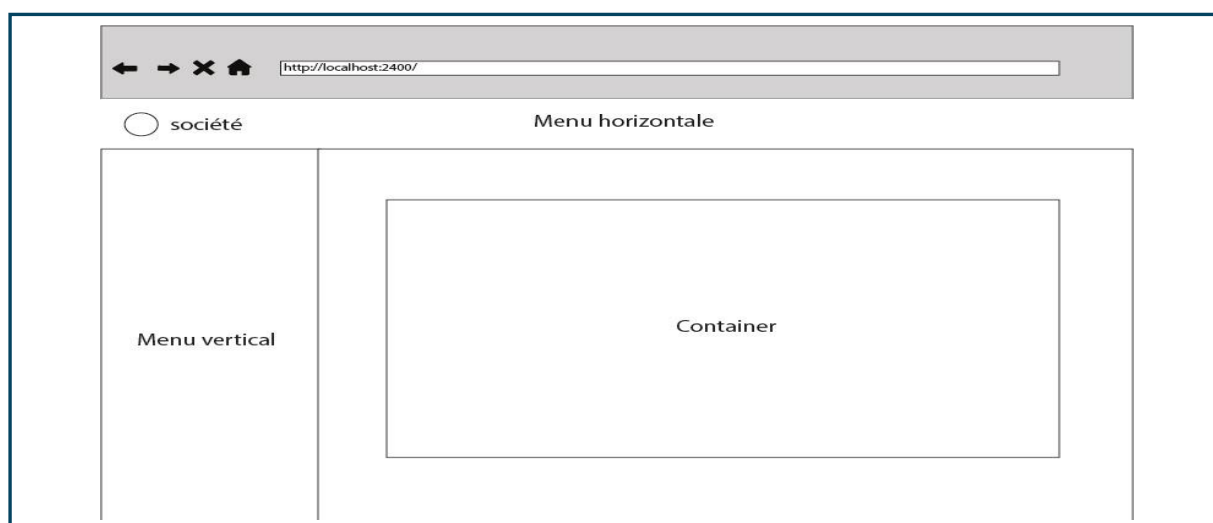


Figure 21: Maquette de vision globale sur les interfaces

## II.2.4. Logo de la société :

Dans cette partie, nous allons montrer l'ancien logo de la société et le nouveau logo qu'on a réalisé avec Illustrator.



Figure 22: Logo de l'application

## III. Description des sprints :

Dans cette partie, nous allons présenter le backlog de chaque sprint.

### III.1. Release 1 :

Tout au long de cette partie, nous traitons les user stories de premier release pour produire à la fin un produit potentiellement livrable.

#### III.1.1. Diagramme de cas d'utilisation du premier release :

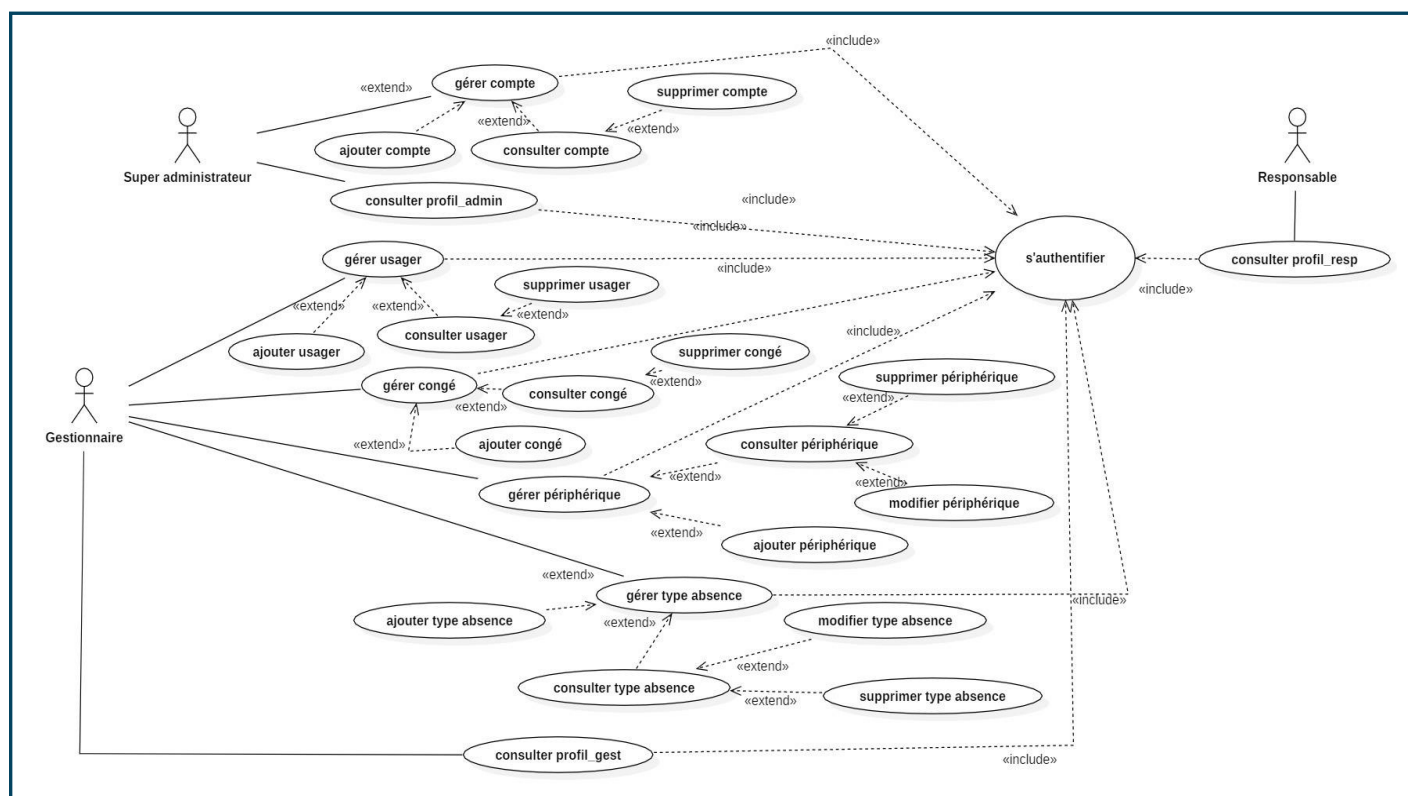


Figure 23: Diagramme de cas d'utilisation du premier release

### III.1.2. Sprint 1 : Gestion des comptes

Dans ce premier sprint, nous avons estimé finalement que l'implémentation du mécanisme d'authentification, la gestion des utilisateurs et la redirection selon les droits d'accès seront un bon point de départ sur lequel nous pourrions compter pour implémenter le reste des fonctionnalités de chaque utilisateur. Pour ce faire, nous commençons par élaborer le backlog du sprint comme illustré par le tableau 15.

Ce sprint dure 112 heures.

Elément du backlog	Les tâches à faire	Estimation
Préparation de l'environnement du travail	<ul style="list-style-type: none"> <li>- Création de la base de données</li> <li>- Créations des tables.</li> <li>- Création des models et des routes en nodejs.</li> <li>- Création des composants.</li> <li>- Création des services et des models en Angular.</li> </ul>	50h
Ajouter utilisateur	<ul style="list-style-type: none"> <li>- Création de l'interface d'ajout.</li> <li>- Implémenter la méthode d'ajout en nodejs et angular.</li> </ul>	10h
S'authentifier	<ul style="list-style-type: none"> <li>- Création de l'interface d'authentification.</li> <li>- Implémenter la méthode d'authentification en nodejs et angular.</li> </ul>	14h
Lister utilisateurs	<ul style="list-style-type: none"> <li>- Création l'interface d'affichage des utilisateurs.</li> <li>- Ajouter la méthode d'affichage en ndoejs et angular.</li> </ul>	10h
Supprimer utilisateur	<ul style="list-style-type: none"> <li>- Création du bouton de suppression.</li> <li>- Ajouter la méthode de suppression avec nodejs et angular.</li> </ul>	10h
Attribuer rôle	<ul style="list-style-type: none"> <li>- Implémentation des guards d'attribution des rôles.</li> </ul>	10h
Consulter profil	<ul style="list-style-type: none"> <li>- Création de l'interface profile.</li> <li>- Ajouter la méthode d'accès et d'affichage du profil.</li> </ul>	8h

Tableau 15: Backlog du sprint 1 Gestion des comptes

### III.1.3. Sprint 2 : Gestion de type d'absence

Dans ce sprint, nous allons mettre en place les fonctionnalités présentées dans ce sprint telles que l'ajout d'un type d'absence, la suppression d'un type d'absence, la modification d'un type d'absence ou la consultation d'un type d'absence. Le tableau 16 résume le backlog de notre sprint.

Ce sprint dure 56h.

Elément du backlog	Les tâches	Estimation
Créer un type d'absence	<ul style="list-style-type: none"> <li>- Création de l'interface d'ajout.</li> <li>- Implémenter la méthode d'ajout.</li> </ul>	18h
Modifier un type d'absence	<ul style="list-style-type: none"> <li>- Ajouter le bouton de modification.</li> <li>- Ajouter la méthode de modification.</li> </ul>	15h
Supprimer un type d'absence	<ul style="list-style-type: none"> <li>- Création le bouton de suppression.</li> <li>- Ajouter la méthode de suppression.</li> </ul>	15h
Consulter un type d'absence	<ul style="list-style-type: none"> <li>- Création de la liste des type d'absence.</li> <li>- Implémenter la méthode de consultation.</li> </ul>	8h

Tableau 16: Backlog du sprint 2 Gestion de type d'absence

### III.1.4. Sprint 3 : Gestion des congés

Dans ce sprint, nous allons mettre en place les fonctionnalités présentées telles que l'ajout d'un congé, la suppression d'un congé ou a consultation d'un congé. Le tableau 17 résume le backlog de notre sprint.

Ce sprint dure 40h.

Elément du backlog	Les tâches	Estimation
Créer congé	<ul style="list-style-type: none"> <li>- Création de l'interface d'ajout.</li> <li>- Implémenter la méthode d'ajout avec nodejs et angular.</li> </ul>	20h
Supprimer congé	<ul style="list-style-type: none"> <li>- Création du bouton de suppression.</li> <li>- Implémenter la méthode de suppression en nodejs et angular.</li> </ul>	10h
Consulter congé	<ul style="list-style-type: none"> <li>- Création de la liste des congés.</li> <li>- Implémenter la méthode d'affichage de la liste des congés.</li> </ul>	10h

Tableau 17: Backlog du sprint 3 Gestion des congés

### III.1.5. Sprint 4 : gestion des périphériques

Dans ce sprint, nous allons mettre en place les fonctionnalités présentées telles que l'ajout d'un périphérique, la suppression d'un périphérique, la modification d'un périphérique ou la consultation. Le tableau 18 résume le backlog de notre sprint. Ce sprint dure 56h.

Elément du backlog	Les tâches	Estimation
Créer un périphérique	<ul style="list-style-type: none"> <li>- Création de l'interface d'ajout.</li> <li>- Implémenter la méthode d'ajout.</li> </ul>	20h
Modifier un périphérique	<ul style="list-style-type: none"> <li>- Création du bouton de modification d'un périphérique.</li> <li>- ajouter la méthode de modification.</li> </ul>	14h
Supprimer un périphérique	<ul style="list-style-type: none"> <li>- Création du bouton de suppression d'un périphérique.</li> <li>- Ajouter la méthode de suppression.</li> </ul>	14h
Consulter un périphérique	<ul style="list-style-type: none"> <li>- Ajouter la méthode de consultation.</li> </ul>	8h

Tableau 18: Backlog du sprint 4 Gestion des périphériques

### III.1.6. Sprint 5 : Gestion des usagers

Dans ce sprint, nous allons mettre en place les fonctionnalités présentées telles que l'ajout d'un usager, la suppression d'un usager... Le tableau 19 résume le backlog de notre sprint. Ce sprint dure 48h.

Elément du backlog	Les tâches	Estimation
Créer un usager	<ul style="list-style-type: none"> <li>- Création de l'interface d'ajout.</li> <li>- Implémenter la méthode d'ajout.</li> </ul>	20h
Supprimer un usager	<ul style="list-style-type: none"> <li>- Création du bouton de suppression d'un usager.</li> <li>- Ajouter la méthode de suppression.</li> </ul>	18h
Consulter un usager	<ul style="list-style-type: none"> <li>- Ajouter la méthode de consultation.</li> </ul>	10h

Tableau 19: Backlog du sprint 5 Gestion des usagers

### III.2. Release 2 :

Notre deuxième release est constitué par deux sprints « consultation des heures de pointage » et « traitement des données de pointage ». C'est la dernière version du projet.

#### III.2.1. Diagramme de cas d'utilisation du deuxième release :

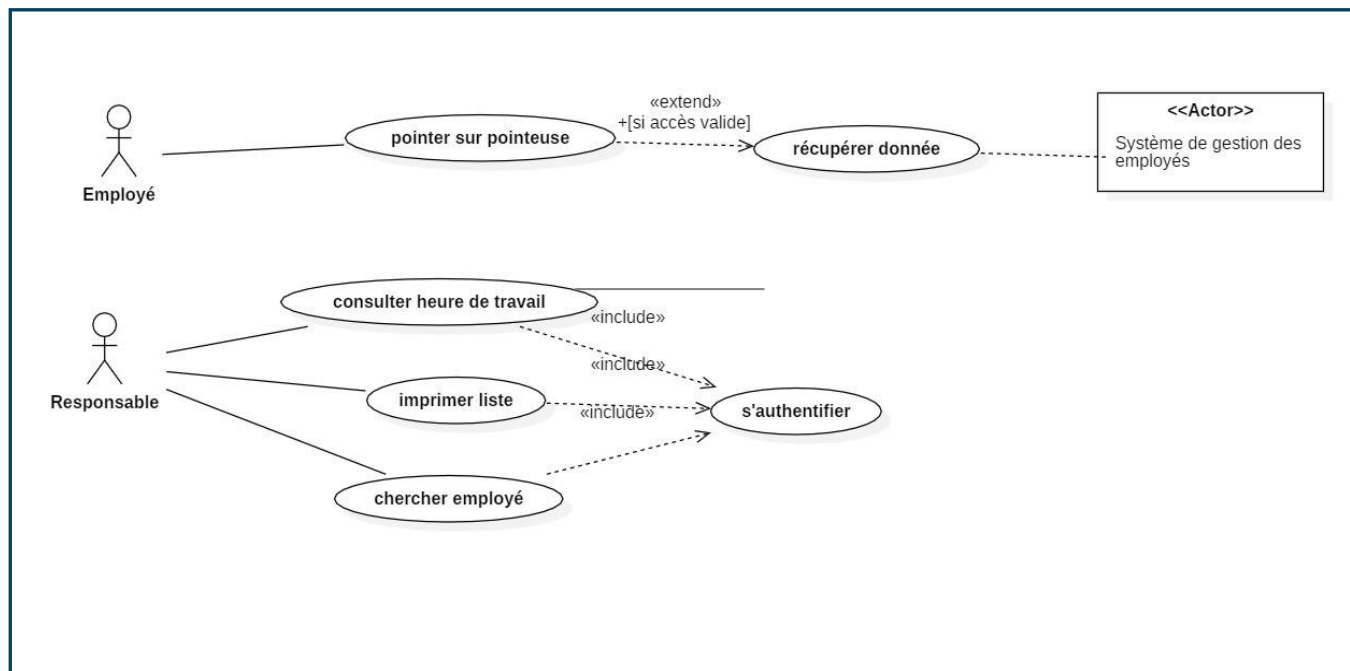


Figure 24: Diagramme de cas d'utilisation du deuxième release

#### III.2.2. Sprint 6 : Consultation des heures de présence

Ce sprint est l'une de la partie importante de l'application web qui comporte des fonctionnalités indispensables telle que la consultation des heures de pointage en temps réel. Le tableau 12 présente le backlog du sprint.

Ce sprint dure 112 heures.

Elément du backlog	Les tâches	Estimation
Consulter la liste des employés qui ont pointé	- Création de l'interface d'affichage. - Ajouter la méthode d'affichage	55h
Chercher un employé	- Création du champ de recherche. - Ajouter la méthode du recherche en nodejs et angular.	30h
Imprimer la liste des employés	- Création du bouton imprimer. - Ajouter la méthode d'impression	27h

Tableau 20: Backlog du sprint 6 Consultation des heures de présence



### III.2.3. Sprint 7 : Traitement des données de pointage

Dans ce deuxième sprint, nous allons mettre en place la partie la plus importante de l'application qui comporte la communication avec la pointeuse en temps réel afin de récupérer les données de pointage et les envoyer vers la base de données. Le tableau 21 représente la backlog de ce sprint.

La durée de sprint est de 160 heures.

Elément du backlog	Les tâches	Durée
Créer des sockets	<ul style="list-style-type: none"> <li>- Création du socket serveur.</li> <li>- Création du socket client.</li> <li>- Ajouter la méthode de communication en réseau local.</li> </ul>	60h
Stocker données dans la base de données	- Ajouter la méthode de stockage des données dans mongodb.	20h
	- Ajouter la méthode d'envoi de données à l'application en temps réel.	40h
Afficher données dans l'application	- Ajouter la méthode d'affichage des données en temps réel.	40h

Tableau 21: Backlog du sprint 7 Traitement des données de pointage

## Conclusion :

Dans ce chapitre, nous avons procédé au choix de la méthodologie Scrum pour présenter la conception de notre système tout en utilisant les diagrammes de UML pour la modélisation de la vue statique et la vue dynamique de notre projet.

---

## **Chapitre 4 : Réalisation et test**

---

## **Introduction :**

Dans ce chapitre, on présente notre environnement de travail logiciel ou matériel et l'architecture de notre projet. Par suite, nous proposons le travail réalisé par l'équipe et les résultats finals.

### **I. Environnement de travail :**

Nous présentons dans cette partie l'environnement matériel et logiciel de notre travail.

#### **I.1. Environnement matériel :**

➤ Machine 1 :

- Fabricant : HP Omen.
- Processeur : Processeur Intel® Core™ i7 de 7e génération.
- Mémoire installée (RAM) : 8 Go.
- Type de système : Système d'exploitation 64 bits.
- Système d'exploitation : Windows 10.

➤ Machine 2 :

- Fabricant : DELL.
- Processeur : Processeur 3<sup>e</sup> Gen Intel Core i3-3217U.
- Mémoire installée (RAM) : 4 Go.
- Type de système : Système d'exploitation 64 bits.
- Système d'exploitation : Windows 10.

#### **I.2. Environnement logiciel :**

##### **I.2.1. Logiciels utilisés :**

Nous allons présenter les logiciels utilisés au cours de développement de l'application.

La tableau 22 montre les logiciels utilisés tout au long de notre projet.








Logo	Description
	<b>Star Uml</b> est un logiciel de conception qui permet de modéliser les diagrammes.
	<b>Visual Studio</b> est l'environnement de développement de notre application.
	<b>GitHub</b> est un service web d'hébergement utilisé pour la gestion de travail à distance.
	<b>Postman</b> est un outil qui nous permet de tester rapidement des requêtes HTTP.
	<b>Illustrator</b> est un logiciel de création des graphiques. On l'a utilisé pour le design de quelques figures et du logo.
	<b>Eclipse</b> est l'environnement utilisé pour le développement des sockets.
	<b>Trello</b> est un outil utilisé pour la gestion d'un projet (avancement du travail avec tableau des tâches..)

Tableau 22: Logiciels utilisés

### I.2.2. Technologies et langages utilisés :

Le tableau 23 présente l'ensemble de technologies que nous avons utilisées.




Logo	Description
	<b>Angular 6</b> est utilisé pour la création de notre projet.
	<b>Node.js</b> est utilisé pour créer le serveur d'application afin d'assurer une communication en temps réel
	<b>Bootstrap</b> est utilisé pour la création du design de l'application.
	<b>MongoDB</b> est un système de gestion de base de données orientée documents qui est utilisé pour stocker les données.
	<b>ExpressJS</b> est un framework utilisé pour demander des données de la base et recevoir ses données afin de les transmettre au nodejs.
	<b>Java</b> est un langage de programmation orienté objet utilisé avec les sockets pour réaliser l'écoute sur un port spécifique.

Tableau 23: Technologies et langages utilisés

## II. Choix technique :

Dans cette partie, on va justifier les choix techniques du langage de programmation, de la plateforme de développement et des Framework utilisés.

### II.1. Architecture générale du projet :

Durant la réalisation de notre projet, on a recours à ce qu'on appelle MEAN STACK qui est un ensemble de technologies formant un environnement complet pour une application. En effet, MEAN signifie « MongoDB Express Angular Node ». Quant au module de l'écoute, on a utilisé les sockets Client – Serveur qui assurent la communication à travers un réseau local. L'un qualifié du client qui envoie des requêtes et l'autre qualifié du serveur qui attend les requêtes du client. Et finalement, quand le serveur reçoit les données du client il les stocke dans la base de données de notre application. La figure 25 montre les interactions entre ces différentes technologies et le rôle de chacune.

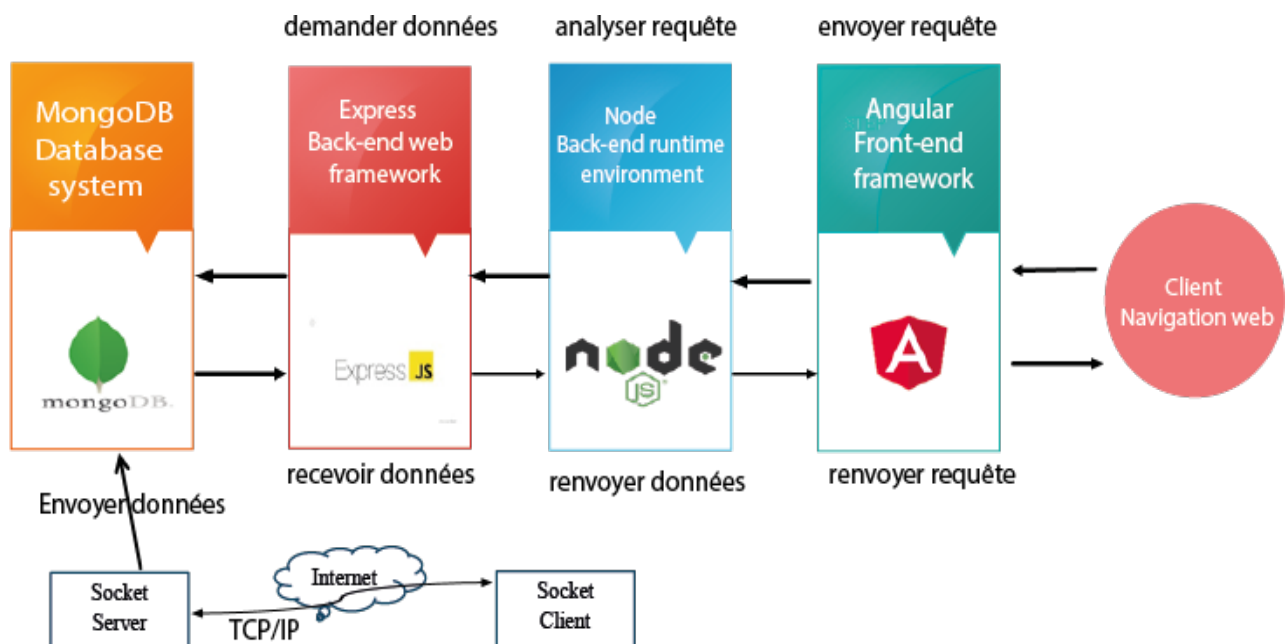


Figure 25: Architecture du projet

## II.2. Diagramme de déploiement :

Le diagramme de déploiement est une vue statique qui sert à représenter les composants du système et la manière dont ils sont répartis ainsi que leurs relations entre eux.

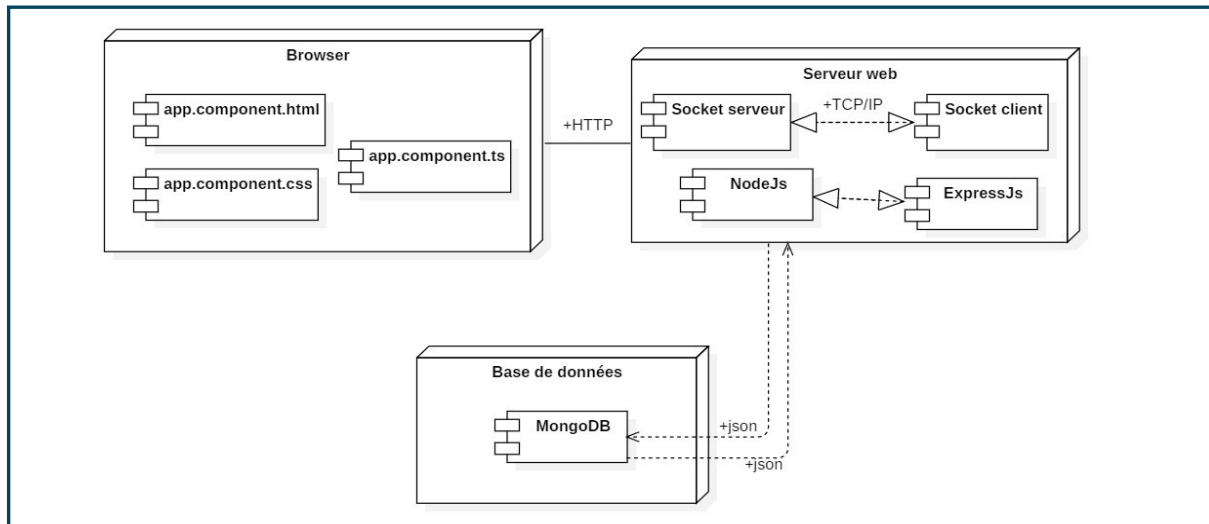


Figure 26: Diagramme de déploiement

## III. Description des interfaces graphiques :

La figure 27 illustre la page d'authentification de l'application. Chaque utilisateur doit entrer son nom d'utilisateur et son mot de passe pour pouvoir accéder à son espace personnel.

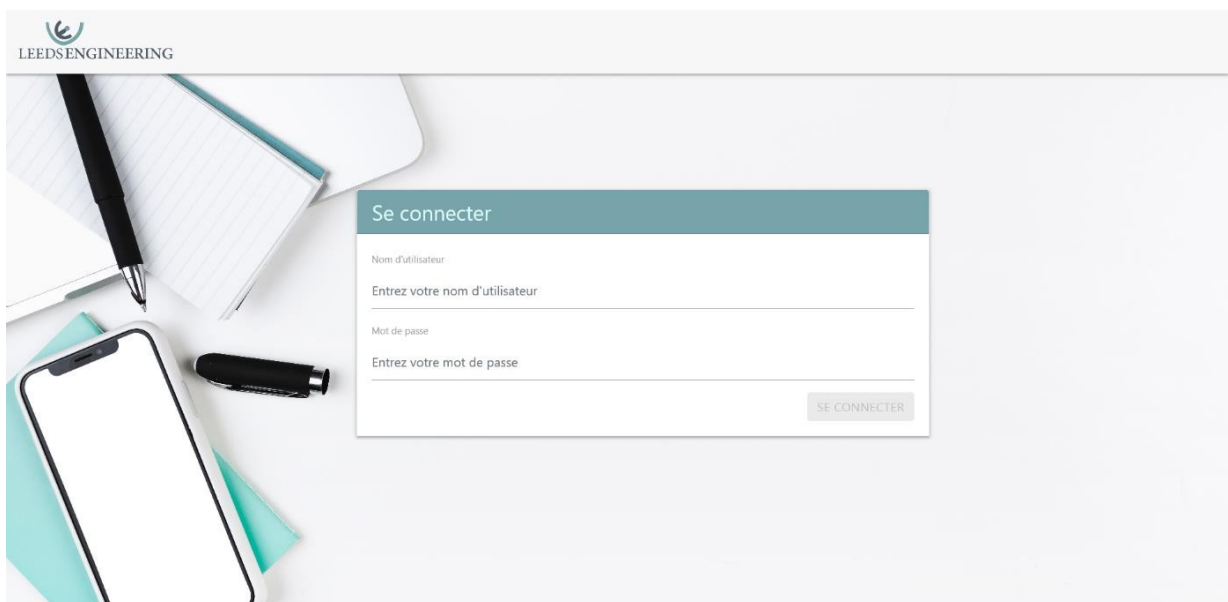


Figure 27: Réalisation de la page d'authentification

La figure 28 représente l'interface de l'utilisateur connecté. Elle contient les informations de ce dernier.

The screenshot shows the 'Mon compte' (My account) page. On the left is a dark sidebar with a menu containing: 'Création d'un compte', 'Gestion des usagers', 'Gestion des périphériques', 'Gestion de types d'absence', 'Gestion des congés', 'Gestion de pointage', 'Les comptes', and 'Mon compte'. The main content area has a header 'Mon compte' and a 'Bienvenue' (Welcome) message. Below this, a table displays the user's profile information:

Nom	safa
Prénom	kassous
Nom d'utilisateur	safakass
Fonction	Super administrateur

At the top right of the page, the text 'safakass Déconnexion' is visible.

Figure 28: Réalisation de la page du profil

La figure 29 représente l'interface de création des comptes, elle contient un formulaire à remplir pour pouvoir ajouter un nouvel utilisateur.

The screenshot shows the 'Créer un compte' (Create account) page. The sidebar is identical to the previous figure. The main content area has a header 'Créer un compte' and a form with the following fields:

- Prénom: Entrez votre prénom
- Nom: Entrez votre nom
- Nom d'utilisateur: Entrez votre nom d'utilisateur
- Fonction: A dropdown menu with the text '-- Sélectionner le rôle de l'utilisateur --'
- Mot de passe: Entrez votre mot de passe

An 'ENREGISTRER' (Register) button is located at the bottom right of the form. The text 'safakass Déconnexion' is visible at the top right of the page.

Figure 29: Réalisation de la page de création des comptes

La figure 30 illustre l'interface de gestion des usagers. Elle contient un formulaire à remplir pour ajouter les usagers et un bouton supprimer pour permettre au gestionnaire de supprimer un usager

Figure 30: Réalisation de la page de gestion des usagers

La figure 31 illustre l'interface de gestion des périphériques. Elle contient un formulaire à remplir pour ajouter un périphérique, un bouton supprimer pour permettre au gestionnaire de supprimer un périphérique et un bouton modifier pour modifier un périphérique.

Figure 31: Réalisation de la page de gestion des périphériques



La figure 32 illustre l'interface des comptes. Elle contient une liste des comptes des utilisateurs et un bouton supprimer pour permettre à l'administrateur de supprimer un utilisateur.

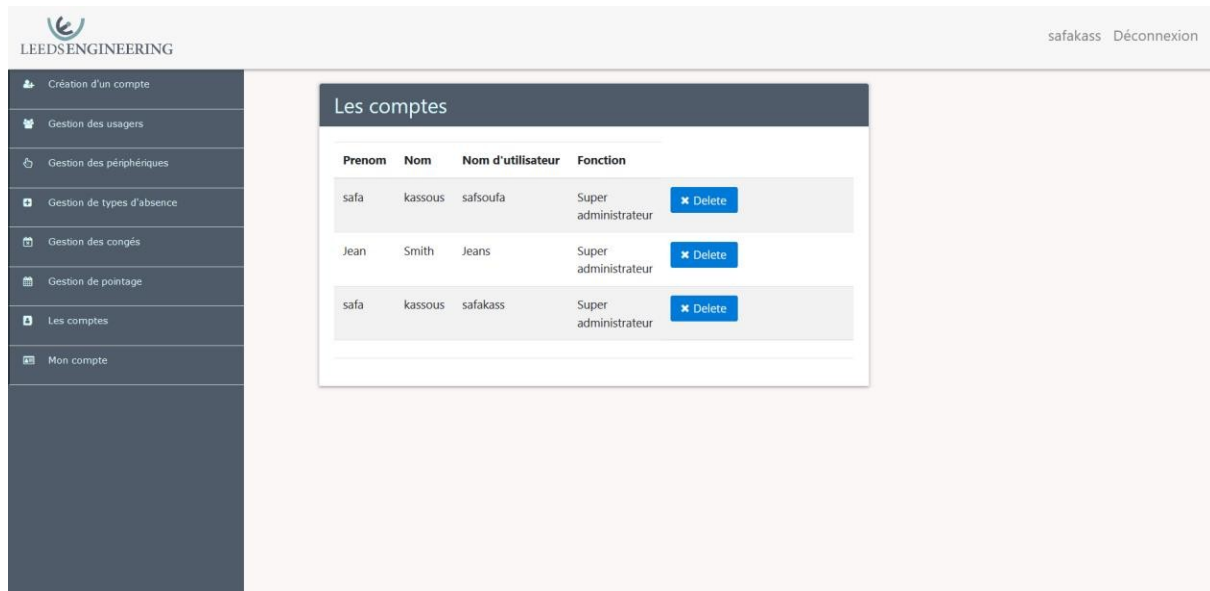


Figure 32: Réalisation de la page des comptes

La figure 33 illustre l'interface de gestion de pointage. Elle contient la liste des utilisateurs qui ont pointé avec la date de pointage, un bouton imprimer qui permet d'imprimer la liste des employés choisis et enfin un champ de recherche qui permet au responsable de chercher un employé bien déterminé.

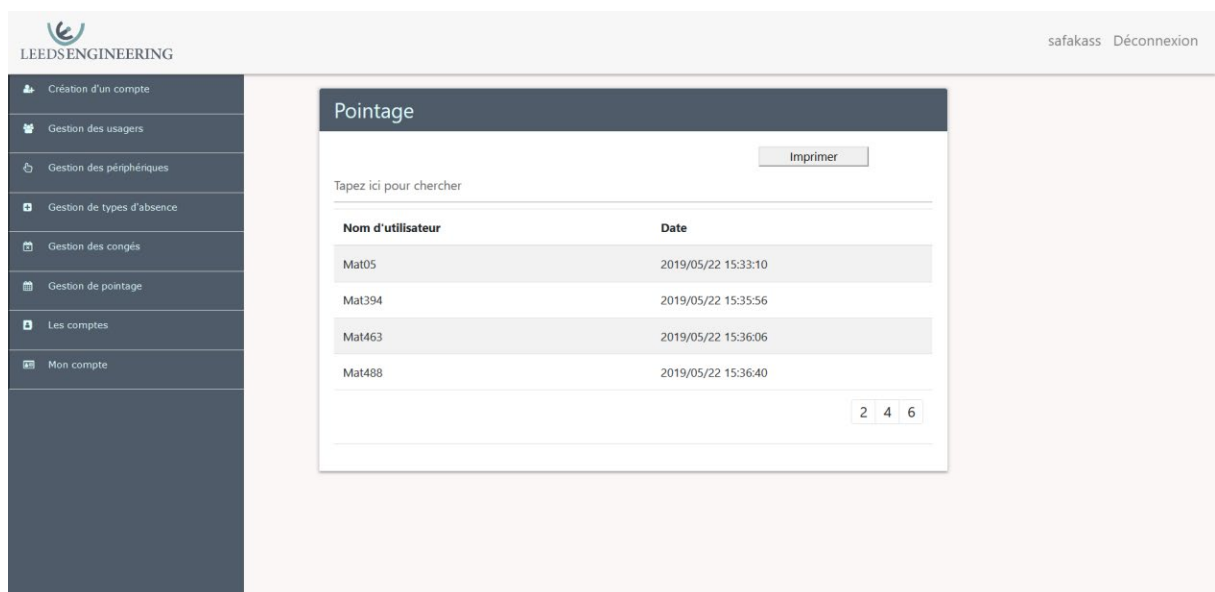
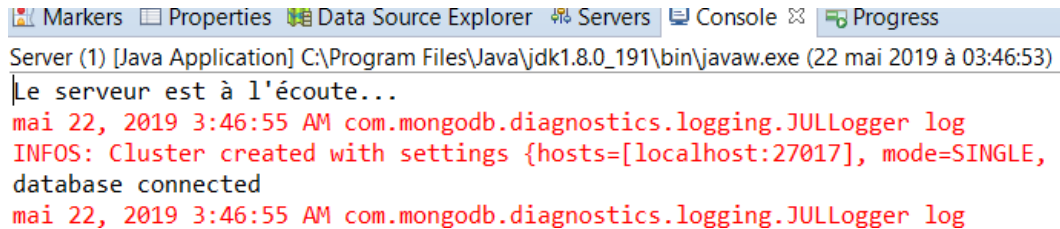


Figure 33: Réalisation de la page de gestion de pointage

## IV. Evaluation de la qualité du produit :

Les figures 34, 35 et 36 représentent le test de l'écoute sur un port spécifique et la récupération de la donnée reçue et par la suite elle montre l'affichage instantanément de la donnée dans l'interface. On voit que la donnée a été bien affichée dans l'application ce qui montre que le test est réussi

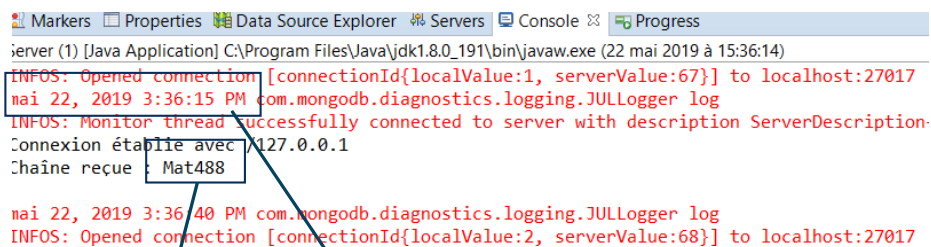


```

Markers Properties Data Source Explorer Servers Console Progress
Server (1) [Java Application] C:\Program Files\Java\jdk1.8.0_191\bin\javaw.exe (22 mai 2019 à 03:46:53)
Le serveur est à l'écoute...
mai 22, 2019 3:46:55 AM com.mongodb.diagnostics.logging.JULLogger log
INFOS: Cluster created with settings {hosts=[localhost:27017], mode=SINGLE,
database connected
mai 22, 2019 3:46:55 AM com.mongodb.diagnostics.logging.JULLogger log

```

Figure 34: Lancement du serveur

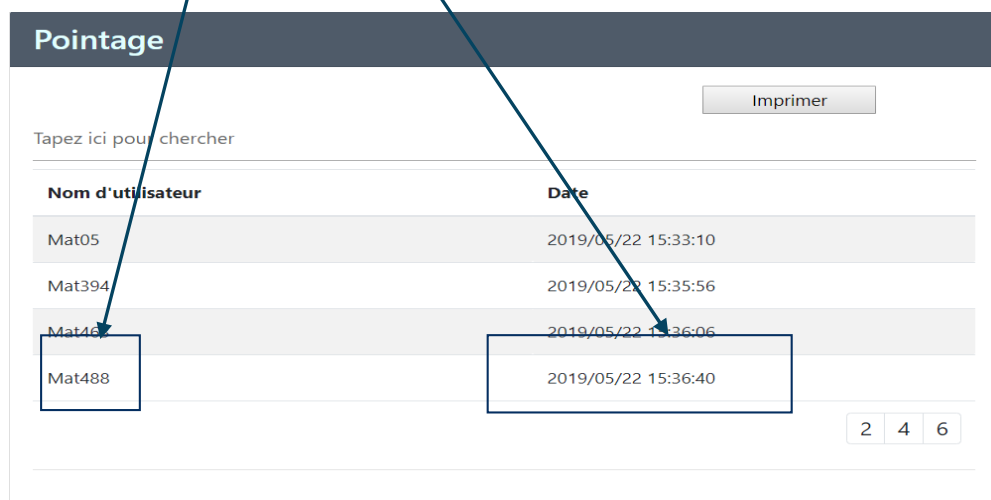


```

Markers Properties Data Source Explorer Servers Console Progress
server (1) [Java Application] C:\Program Files\Java\jdk1.8.0_191\bin\javaw.exe (22 mai 2019 à 15:36:14)
INFOS: Opened connection [connectionId{localValue:1, serverValue:67}] to localhost:27017
mai 22, 2019 3:36:15 PM com.mongodb.diagnostics.logging.JULLogger log
INFOS: Monitor thread successfully connected to server with description ServerDescription:
Connexion établie avec 127.0.0.1
Chaîne reçue : Mat488
mai 22, 2019 3:36:40 PM com.mongodb.diagnostics.logging.JULLogger log
INFOS: Opened connection [connectionId{localValue:2, serverValue:68}] to localhost:27017

```

Figure 35: Récupération de la donnée



Nom d'utilisateur	Date
Mat05	2019/05/22 15:33:10
Mat394	2019/05/22 15:35:56
Mat488	2019/05/22 15:36:06
Mat488	2019/05/22 15:36:40

Figure 36: Test de l'affichage des données dans l'interface pointage

## Conclusion :

Dans ce chapitre, nous avons présenté l'environnement du projet. Par la suite, nous avons exposé le travail réalisé par des captures d'écran de quelques interfaces de notre application.

## **Conclusion générale**

Le travail que nous avons effectué s'inscrit dans le cadre du projet de fin d'études pendant 3 mois au sein de la société Leeds Engineering. L'objectif de notre stage consiste à concevoir et développer une application web de gestion du temps de travail du personnel.

Ce rapport décrit la démarche de notre travail, nous avons procédé dans un premier temps par une étude profonde des applications existantes de le marché dans le but de bien comprendre leurs carences et engendrer une solution efficace et qui répond aux besoins de la société. Nous avons par la suite choisi les méthodologies et les technologies utilisés. En troisième lieu, nous avons passé à la planification de notre travail par l'identification de tâches à réaliser tout en essayant d'appliquer la méthode agile SCRUM et respectant son cycle de vie. En effet, nous avons divisé notre projet en deux releases où chacun contient différents sprints qui aident à opter après chaque release un livrable opérationnel. Cette méthode nous a bien aidés à diminuer la charge de travail en partageant les tâches entre l'équipe de travail et à respecter les deadlines.

Ce stage nous a beaucoup apporté sur le plan professionnel puisqu'il nous a offert l'opportunité pour mettre en pratique les connaissances théoriques que nous avons acquises durant les années universitaires et de les enrichir par des nouveaux savoir-faire. Nous avons aussi travaillé dans un domaine qui nous plaît et dans lequel nous envisageons de nous orienter pour notre poursuite d'études.

Quant à nos efforts fournis tout au long de ce travail sont chercher à découvrir les langages de développement. En fait, on a commencé par l'apprentissage du Framework Angular pour bien organiser nos interfaces et puis nous avons appris la Framework Express et nodeJS pour le backend de notre application et enfin nous avons travaillé pour la première fois avec les sockets pour l'écoute sur un port spécifique. Et pour être plus efficient, nous avons adoptée l'architecture MVC afin de bien organiser l'interface homme-machine.

Enfin, il est vrai que nous sommes arrivés à atteindre nos objectifs tracés, mais cela ne nous empêche pas de penser à des améliorations comme perspectives de notre projet. Nous proposons à ajouter un module qui calcule les salaires d'un employé par rapport à ces absences dans un mois. Et on peut ajouter une application mobile qui permet aux employés de consulter sa présence ou de demander un congé en ligne.

## **Netographie :**

[N1] : <https://www.btmanager.fr/>

[N2] : B. Meyer, Conception et programmation orientées objet, 2e éd., 2000, Eyrolles.

[N3] <https://www.thierry-pigot.fr/scrumbn-en-moins-de-10-minutes/>

[N4] <http://www.aubryconseil.com/post/Le-backlog-de-produit>

[N5] [https://www.eyrolles.com/Chapitres/9782212112467/chap\\_3.pdf](https://www.eyrolles.com/Chapitres/9782212112467/chap_3.pdf)