

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A/Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique



Mémoire de fin de cycle

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

Option : Génie Logiciel

Thème

Mise en œuvre d'une solution de monitoring de
présence sous Android

Cas d'étude « CEVITAL BEJAIA »

Réalisé par

Melle. *BENSLIMANE* Ferial Melle. *BENKHELLADI* Hanane

Devant le jury composé de :

Président : Dr. LAHLAH Souad

Examineur : Dr. MOHAMMEDI Mohamed

Encadrant : Dr. YESSAD Nawal

Année universitaire 2020/2021

Remerciements

Nos remerciements vont à notre encadrant **Mme.YESSAD Nawel** pour la préciosité de ses conseils, sa disponibilité et son orientation qui ont constitué un apport considérable grâce auquel ce travail a pu être mené à bon port.

Nous tenons à remercier chacun des membres du jury pour l'intérêt qu'ils ont porté à notre étude en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

Nos remerciements s'étendent à tous nos enseignants et aux membres du département d'informatique de l'université **ABDERRAHMANE MIRA**.

Ainsi qu'à tous ceux et celles qui ont contribué de près ou de loin à l'accomplissement de ce travail.

Dédicaces

A nos parents, pour leurs sacrifices déployés à notre égard, pour leur patience, leur amour et leur confiance. Qu'ils trouvent dans ce document, le témoignage de notre profonde affection et de notre attachement indéfectible ; nulle dédicace ne puisse exprimer ce qu'on leur doit.

A nos frères et soeurs et tous nos amis pour chaque mot reçu, chaque geste d'amitié, à chaque main tendue et pour toute attention témoignée.

Table des matières

	Page
Liste des figures	iv
Liste des tableaux	vi
Liste d'abréviation	vii
Introduction générale	1
1 Méthodologie	3
1.1 Présentation de l'organisme d'accueil	3
1.2 Problématique	4
1.3 Présentation du contexte du projet	4
1.4 CODE QR	5
1.4.1 Caractéristique du code QR	5
1.4.2 Structure du code QR	6
1.4.3 Usage du code QR	7
1.4.4 Générateurs de codes QR	7
1.5 Application mobile	7
1.6 Android	8
1.7 Besoins du système	9
1.7.1 Les besoins fonctionnels	9
1.7.2 Les besoins non fonctionnels	9
1.8 Architecture des fonctionnalités	10
1.9 Méthodologie de conception	10
1.9.1 Présentation du processus unifié UP	10
1.10 Présentation UML	11

2	Analyse des besoins et conception	14
2.1	Identification des cas d'utilisations	14
2.2	Description textuelle des cas d'utilisation	15
2.3	Diagrammes de cas d'utilisation	28
2.3.1	Diagramme de cas d'utilisation général	28
2.3.2	Diagramme de cas d'utilisation « Pointage »	30
2.3.3	Diagramme de cas d'utilisation « Consultation de profil »	30
2.3.4	Diagramme de cas d'utilisation « Gestion des comptes »	31
2.3.5	Diagramme de cas d'utilisation « Gestion des employés »	31
2.3.6	Diagramme de cas d'utilisation « Gestion des dispositifs »	32
2.3.7	Diagramme de cas d'utilisation « Gestion des évènements »	33
2.3.8	Diagramme de cas d'utilisation « Affichage statistiques »	33
2.3.9	Diagramme de cas d'utilisation « Réception/envoie des informations »	34
2.4	Diagramme de séquence	34
2.4.1	Diagramme de séquence du cas d'utilisation « Authentification »	35
2.4.2	Diagramme de séquence du cas d'utilisation « pointage »	36
2.4.3	Diagramme de séquence du cas d'utilisation « Consultation de profil »	38
2.4.4	Diagramme de séquence du cas d'utilisation « Gérer les comptes » .	38
2.4.5	Diagramme de séquence du cas d'utilisation « Gérer les employés »	40
2.4.6	Diagramme de séquence du cas d'utilisation « Gestion des dispositifs »	42
2.4.7	Diagramme de séquence du cas d'utilisation « Gérer les évènements »	44
2.4.8	Diagramme de séquence du cas d'utilisation « Consulter fiche de pointage des employés »	46
2.4.9	Diagramme de séquence du cas d'utilisation « Gestion des plan- nings »	47
2.5	Diagramme de classe	49
2.6	Dictionnaire de données	51
2.7	Le modèle relationnel	52
3	Réalisation	55

3.1	Environnement de développement	55
3.1.1	Plateformes utilisées	56
3.1.2	Logiciels utilisés	56
3.1.3	Langages de programmation utilisés	57
3.2	Les librairies utilisées	59
3.3	Présentation de l'application mobile	59
3.3.1	Arborescence de l'application	60
3.3.2	Aperçu de notre application	60
3.4	Présentation du serveur web	64
3.4.1	Arborescence du serveur web	64
3.4.2	Aperçu des pages web	65
	Coclusion générale	70
	Bibliographie	72

Liste des figures

1.1	Correction des symboles déformés.	6
1.2	La structure du code QR.	6
1.3	Architecture des fonctionnalités.	10
2.1	Diagramme de cas d'utilisation général.	29
2.2	Diagramme de cas d'utilisation «pointage».	30
2.3	Diagramme du cas d'utilisation « Consultation de profil »	30
2.4	Diagramme de cas d'utilisation «Gestion des comptes».	31
2.5	Diagramme de cas d'utilisation «Gestion des employés».	32
2.6	Diagramme de cas d'utilisation «Gestion des dispositifs».	32
2.7	Diagramme de cas d'utilisation «Gestion des évènements».	33
2.8	Diagramme de cas d'utilisation «Affichage statistiques».	33
2.9	Diagramme de cas d'utilisation «Réception/Envoie d'information»	34
2.10	Diagramme de séquence de cas d'utilisation «Authentification».	35
2.11	Diagramme de séquence du cas d'utilisation «Pointage».	37
2.12	Diagramme de séquence du cas d'utilisation «Consultation de profil».	38
2.13	Diagramme de séquence de cas d'utilisation «Gérer les comptes ».	39
2.14	Diagramme de séquence du cas d'utilisation «Gestion des employés».	42
2.15	Diagramme de séquence du cas d'utilisation «Gérer les dispositifs».	44
2.16	Diagramme de séquence de cas d'utilisation «Gérer les évènements».	46
2.17	Diagramme de séquence de cas d'utilisation « Consulter fiche de pointage des employés ».	47
2.18	Diagramme de séquence de cas d'utilisation «Gérer les plannings».	49
2.19	Diagramme de classe.	50
3.1	Arborescence de l'application mobile.	60
3.2	Interface Authentification.	61

3.3	Interface Accueil.	61
3.4	Interface Pointage.	62
3.5	Interface Scanner le QR code.	62
3.6	Interface Consulter mon profil	63
3.7	Interface Consulter mon planning.	63
3.8	Arboraisance seveur web.	64
3.9	Page Authentification.	65
3.10	Page Accueil administrateur.	66
3.11	Page d'ajout d'un compte.	67
3.12	Page d'ajouter d'un planning.	68

Liste des tableaux

- 2.1 Les cas d'utilisations associés au système à développer. 15
- 2.2 Description du cas d'utilisation « Authentification ». 16
- 2.3 Description du cas d'utilisation «Pointage». 17
- 2.4 Description du cas d'utilisation «Consultation de profil ». 18
- 2.5 Description du cas d'utilisation « Consultation fiche de pointage ». 19
- 2.6 Description du cas d'utilisation «Gestion des comptes». 20
- 2.7 Description du cas d'utilisation « Gestion des employés ». 22
- 2.8 Description du cas d'utilisation «Gestion des dispositifs». 24
- 2.9 Description du cas d'utilisation «Gestion des évènements». 25
- 2.10 Description du cas d'utilisation « Affichage statistiques ». 26
- 2.11 Description du cas d'utilisation «Consultation fiche de pointage des em-
ployés». 26
- 2.12 Description du cas d'utilisation «Gestion des plannings». 28
- 2.13 Dictionnaire de données. 52

Liste d'abréviation

2D	2Dimensions
3D	3Dimensions
API	Application Programming Interface
CCD	Charged Coupled Device
CSS	Cascading Style Sheets
EPS	Encapsulated PostScript
JPEG	Joint Photographic Experts Group
JSP	Java Server Pages
HTML	HyperText Markup Language
IDE	Integrated Development Environment
ISO	International Organization for Standardization
GPS	Global Positioning System
OMG	Object Management Group
PHP	Hypertext Preprocessor
PNG	Portable Network Graphics
QR	Quick Response
SMS	Short Message Service
SQL	Structured Query Language
SVG	Scalable Vector Graphics
TIFF	Tagged Image File Format
UML	Unified Modeling Language
UP	Unified Process
URL	Uniform Resource Locato
VSCoDe	Visual Studio Code
WWW	World Wide Web
W3C	World Wide Web Consortium

INTRODUCTION GÉNÉRALE

Un monitoring de présence sous ses différents usages est un dispositif qui permet de contrôler la présence au travail des employés au sein d'une entreprise. A l'origine, les horaires et les accès spécifiques à une zone étaient contrôlés par un employé qui notait scrupuleusement chaque allée et venu des différents employés d'un organisme.

Pour justifier les horaires du personnel d'une entreprise, l'employeur doit mettre en place un système de pointage. Ce dernier peut prendre la forme d'un enregistrement et/ou d'un récapitulatif, un système automatique/simple ou carrément une mention dans un registre des entrées et sorties journalières.

Le système automatique permet de bénéficier de nombreux avantages à savoir un gain de temps dans la gestion des ressources humaines, une facilité de traitement pour les services budgets et ceci grâce à des éditions automatiques de pré-paie (relevé de pointage) et voir aussi une amélioration du climat social où tous les employés subissent le même traitement.

L'entreprise Cevital est une entreprise qui embauche un nombre important d'employés. Afin de gérer la présence de ses fonctionnaires, l'entreprise a mis en place un système statique de pointage. En effet, les employés sont équipés d'un badge qu'ils passent sur une badgeuse à chaque fois qu'ils accèdent ou quittent l'entreprise. Notre projet de fin de cycle consiste à innover ce système pour une meilleure flexibilité. Il s'agit de l'usage des smartphones pour le scan des codes QR comme une alternative des badges à puce.

Les différents services de l'entreprise sont dotés d'un code QR unique que les employés scannent à l'entrée et à la sortie du service. Ceci permettra d'éviter l'encombrement à l'entrée de l'entreprise d'une part et une meilleure gestion par service d'autre part.

Le QR code (Quick Response code) est un type de code à barres 2D qui est utilisé pour faciliter l'accès aux informations par le biais d'un smartphone. Dans ce processus, appelé « marquage mobile », le propriétaire du smartphone pointe le téléphone vers un code QR et ouvre une application de lecture de codes-barres qui fonctionne en conjonction avec l'appareil photo du téléphone. Le lecteur interprète le code et identifie l'employé et sa présence dans l'entreprise.

Afin de réaliser notre projet de fin de cycle, nous avons opté pour un plan de travail composant trois chapitres :

Le premier chapitre est une présentation de l'organisme d'accueil, il introduit également le contexte du projet et sa problématique, ainsi que les concepts clé tels que l'application mobile et le serveur web. Dans la suite du chapitre, nous exprimerons les besoins fonctionnels et non fonctionnels de notre application. Après, nous définirons la méthodologie de conception ainsi que la présentation UML avec laquelle nous avons mené notre projet de façon rationalisée et structurée pour nous aider à accomplir chaque étape du projet.

Le deuxième chapitre qui s'intitule analyse des besoins et conception permettra d'identifier les différents cas d'utilisation afin de modéliser en premier lieu les diagrammes de cas d'utilisation et de les décrire de façon détaillée par le biais de diagrammes de séquence. Enfin, une vue statique de notre système est donnée sous forme d'un diagramme de classes.

Le dernier chapitre intitulé réalisation consiste à présenter l'environnement de développement. Ainsi qu'une description et présentation des principales interfaces de notre application. Nous clôturons le mémoire par une conclusion générale dans laquelle nous exposons quelques perspectives.

Introduction

Ce chapitre a pour objectif de présenter l'organisme d'accueil, à savoir l'entreprise CEVITAL. Ensuite, une présentation du contexte et du projet est abordée.

La problématique relative à la création de l'application mobile et du serveur web est décrite tout en définissant les besoins fonctionnels et non fonctionnels de l'application/serveur. Dans ce chapitre, nous exposons également l'architecture des fonctionnalités et nous définissons le code QR et les applications mobiles ainsi que les méthodologies de conception et la présentation du langage UML.

1.1 Présentation de l'organisme d'accueil

CEVITAL est une entreprise créée par l'entrepreneur ISSAD REBRAB¹ le 02 mai 1998, avec un capital privé 68,76 milliards de dinars, c'est une société par action (SPA) dont les actionnaires principaux son monsieur Issad REBRAB et fils (Omar, Malik, Linda, Yacine et Salim). Elle est implantée à l'extrême Est du port de Bejaia, elle est l'un des fleurons de l'industrie agro-alimentaire en Algérie qui est constituée de plusieurs unités de production, équipées de la dernière technologie et qui poursuit son développement par divers projets en cours de réalisation. Son expansion est son développement durant ses dernières années, fait d'elle un important pourvoyeur d'emplois et de richesses.

1. **Mr ISSAD REBRAB** Président Dircteur général du groupe cevital.

CEVITAL agro-industrie est le plus grand complexe privé en ALGERIE et le leader en Afrique et dans le bassin méditerranéen dans l'industrie du sucre et l'huile végétale. Elle exporte ses produits dans plusieurs pays, notamment en Europe, Numidie, au Moyen Orient et en Afrique de l'Ouest [1].

1.2 Problématique

Nous avons constaté durant notre stage au niveau de l'entreprise CEVITAL que l'état de présence des employés est gérée grâce à un logiciel de gestion de temps de présence. Le logiciel nommé « Kélio intégral » est développé par BODET². Chaque service possède une pointeuse sur laquelle chaque fonctionnaire doit faire passer son badge qui est doté d'une puce à l'entrée et à la sortie de chaque service.

La nouveauté qu'on veut apporter par rapport à ce système existant peut être moins coûteuse et plus facile à utiliser. Nous avons opté pour une nouvelle démarche qui consiste à l'utilisation des smartphones au lieu des badges à puce. Désormais, les employés utiliseront un QR code qui sera scanner grâce à la caméra de leurs téléphone via l'application mobile.

1.3 Présentation du contexte du projet

Notre projet de fin de cycle a pour but de concevoir et de réaliser une application mobile fonctionnelle sur les smartphones munis d'un système ANDROID qui sera lié à un serveur web.

L'application mobile sera installée sur le smartphone de l'employé, et lui permettra de :

- Scanner un QR code grâce à la caméra du téléphone via l'application mobile après l'authentification, ainsi toutes les informations de l'employé seront envoyées à notre serveur web. En effet, chaque service de l'organisme d'accueil aura un QR code qui sera généré spécialement pour ce dernier ;

2. **BODET** Leader international dans la gestion du temps.

- Consulter son profil, son planning et sa fiche de pointage à travers une interface de l'application .

Le serveur web aura pour fonction d'enregistrer les informations des employés reçues à partir de l'application mobile, gérer les comptes des employés, gérer les dispositifs, gérer les événements comme les réunions par exemple, consultation fiche de pointage des employés, gérer les plannings et gérer les employés.

1.4 CODE QR

Le QR code (Quick Response) est un code matriciel 2D conçu en tenant compte de deux points, c'est-à-dire qu'il doit stocker une grande quantité de données par rapport aux codes-barres 1D et il doit être décodé à grande vitesse à l'aide de n'importe quel appareil portable comme les smartphones [2]. Il a été inventé en 1994 par Denso, l'une des principales sociétés du groupe Toyota, et approuvé en tant que norme internationale ISO (ISO/IEC18004) en juin 2000. Ce symbole en deux dimensions a été initialement destiné à être utilisé pour le contrôle de la production de pièces automobiles, mais il s'est répandu dans d'autres domaines. Aujourd'hui, le code QR est vu et utilisé quotidiennement pour plusieurs raisons [3].

1.4.1 Caractéristique du code QR

Le code QR offre une capacité de stockage de données élevée : 7089 caractères numérique ,4896 caractères alphabétiques et signes, 2953 caractères binaires (données codées sur 8 bits) et 1817 données en kanji (caractère chinois de l'écriture japonaise), l'enregistrement à haute densité (environ 100 fois plus élevée en densité que les symboles linéaires) et la lecture à grande vitesse [3][4].

Le QR Code présente d'autres supériorités tant sur le plan des performances que des fonctionnalités [3][4] ,la figure 1.1 represente la correction des sumboles déformés :

- Lecture à grande vitesse dans toutes les directions (360°);
- Résistance aux symboles déformés;
- Correction des erreurs;

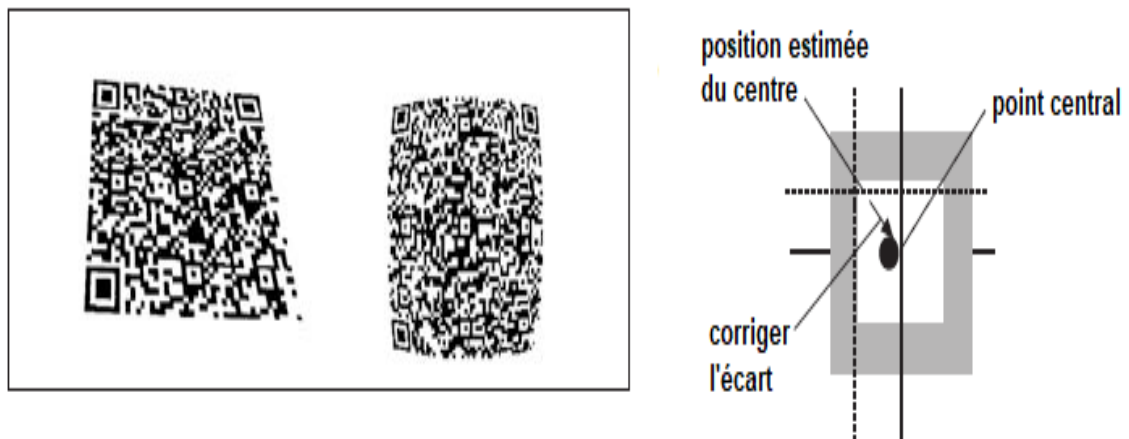


FIGURE 1.1 – Correction des symboles déformés.

- Relier les fonctionnalités des symboles ;
- Marquage direct ;

1.4.2 Structure du code QR

Le QR Code est un symbole de type matriciel avec une structure cellulaire disposée dans un carré. Il se compose des motifs de fonctionnalité pour faciliter la lecture et la zone de données où les données sont stockées. Le code QR a des motifs de recherche, des motifs d'alignement, des schémas de temporisation et une zone tranquille [3] comme le montre la figure 1.2 :

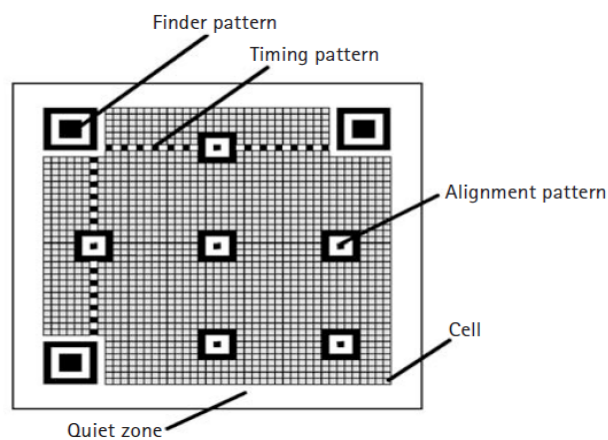


FIGURE 1.2 – La structure du code QR.

1.4.3 Usage du code QR

Un code QR est appliqué dans différents flux d'applications liés au marketing, à la sécurité, etc. et gagne en popularité à un rythme très élevé. Jour après jour, de plus en plus de gens prennent conscience de cette technologie et l'utilisent en conséquence. La popularité du code QR croît rapidement avec la croissance des utilisateurs de smartphones et donc le code QR arrive rapidement à des niveaux élevés d'acceptation dans le monde entier. [2].

1.4.4 Générateurs de codes QR

La procédure de codage du Code QR comprend les étapes suivantes. Tout d'abord, les données d'entrée sont encodées selon le mode le plus efficace ce qui forme un flux binaire. Les flux binaires sont divisés en mots de code. Ensuite, les mots de code sont divisés en bloc, et des mots de code de correction d'erreur sont ajoutés à chaque bloc. Tous ces mots de code sont placés dans une matrice et sont masqués par un motif de masque. Enfin, des motifs de fonction (modèle de positionnement, modèle d'alignement, modèle timing) sont ajoutés au symbole QR [6].

Il existe plusieurs outils de création de codes QR qui sont disponibles en ligne et gratuitement. Ils se distinguent les uns des autres par [2] :

- Leurs capacités de production : texte, URL, appel téléphonique, SMS, vCard ;
- Correction des erreurs du code produit ;
- La possibilité de personnaliser les couleurs des pixels, du fond et de la bordure ;
- Le format des images (jpeg, png, eps, tiff, svg), un code QR ainsi vectorisé peut être agrandi sans déformation par exemple pour être imprimé sur une bache.

Des générateurs de codes QR sont proposés sous forme d'applications mobiles. Par exemple QRS+16, QR Code Génération, etc.

1.5 Application mobile

Une application mobile est un logiciel applicatif téléchargeable que l'on installe facilement sur nos smartphones (téléphones mobiles intelligents) via une plateforme de téléchargement

adaptée au système d'exploitation (Google Play, App Store, Windows Store)[7].

Il existe trois types d'applications mobiles selon leurs spécificités techniques :

- **Applications Natives** : ces applications sont liées au système d'exploitation sur lequel elles sont installées, car elles utilisent des caractéristiques reliées à celui-ci. Elles sont écrites dans un langage adapté au système d'exploitation en question ;
- **Applications Web** : ce sont toutes les applications conçues grâce aux outils de développement web actuel (HTML, CSS, JavaScript, etc.). Elles sont accessibles sur tous les mobiles via un navigateur Web ce qui les rend plus intéressantes du point de vue financier, car les coûts de développement sont réduits vu qu'on développe une seule application qui est compatible avec tous les smartphones quelques soient leurs systèmes ;
- **Applications Hybrides** : sont des applications qui incorporent les deux principes de développement précédemment cités : les caractéristiques des applications web et celles des applications natives. Elles pourront être distribuées sur les plateformes de téléchargement telles que l'Apple Store (iOS), Play Store (Android) ou encore Windows Store (Windows Phone). L'utilisateur peut donc installer ces applications et consulter leur contenu sans avoir à passer par un navigateur web, car l'application elle-même, incorpore un navigateur web intégré.

1.6 Android

Android est un système d'exploitation mobile basé sur Linux pour smartphones, tablettes et autres appareils dits « intelligents ». Lancé en 2003, racheté par Google en 2005, et distribuée sous licence open source, aujourd'hui dans sa dixième version Android est l'OS mobile le plus utilisé au monde [8].

1.7 Besoins du système

1.7.1 Les besoins fonctionnels

— L'application mobile :

- Grâce a une interface d'authentification, l'employé saisie son identifiant et son mot de passe pour pouvoir accéder à l'application ;
- A partir de l'application, l'employé scanne le QR code grâce à la caméra du téléphone, il peut également consulter son profil, son planning et sa fiche de pointage ;
- L'application doit obligatoirement être utilisée dans un téléphone qui est connecté à internet ;

— Le serveur web :

- L'ajout des employés, la modification de leurs informations ainsi que la possibilité de la suppression et l'archivage des données par un administrateur ;
- Le serveur récupère les informations des employés à partir de l'application mobile et il envoie également les informations relatives à son profil, son planning et à sa fiche de pointage ;
- Gestion des plannings, des dispositifs et des évènements ainsi que la gestion des comptes ;

1.7.2 Les besoins non fonctionnels

- Ergonomie, cohérence de l'application (interface intuitive) ;
- Sécurité de l'application aux intrusions.

1.8 Architecture des fonctionnalités

La figure 1.3 représente l'architecture des fonctionnalités de notre projet

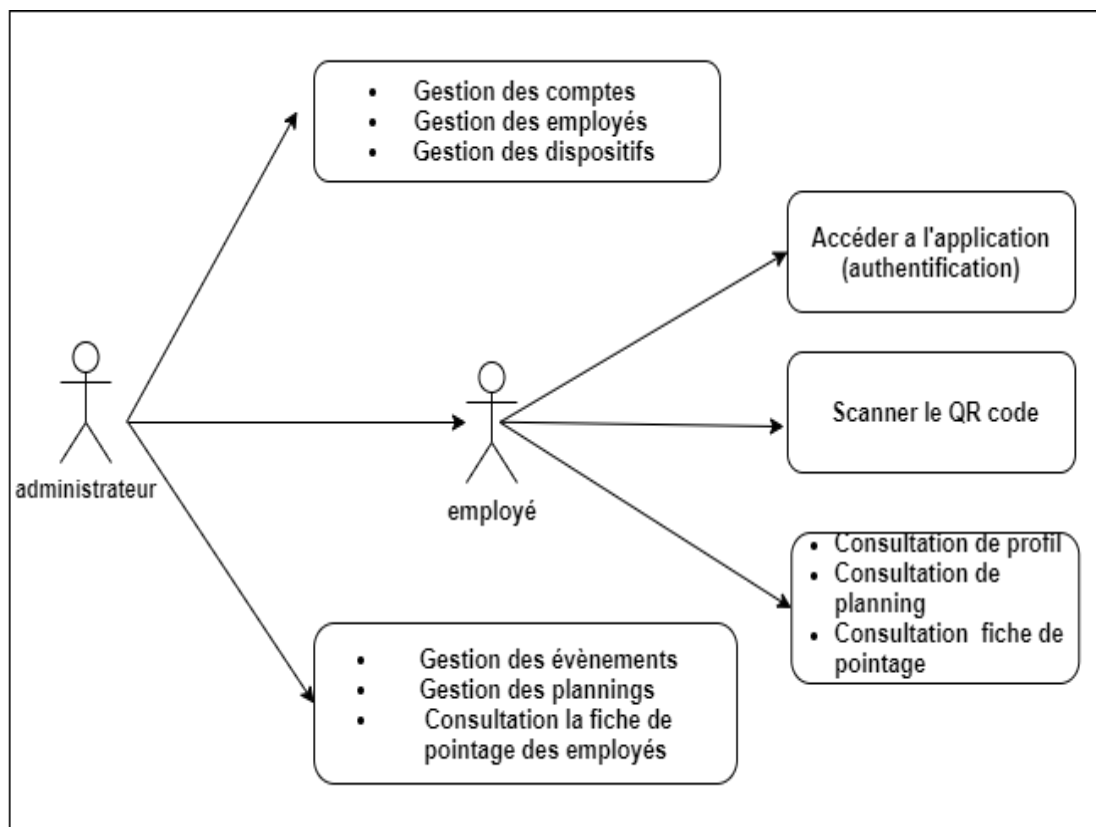


FIGURE 1.3 – Architecture des fonctionnalités.

1.9 Méthodologie de conception

1.9.1 Présentation du processus unifié UP

Le processus unifié est un processus de développement logiciel « itératif et incrémentale, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques » [9] :

- **Itératif et incrémentale :** le projet est découpé en itérations de courte durée (environ 1 mois) qui aident à mieux suivre l'avancement global. À la fin de chaque itération, une partie exécutable du système final est produite, de façon incrémentale ;

- **Centré sur l'architecture** : tout système complexe doit être décomposé en partie modulaires afin de garantir une maintenance et une évolution facilitées. Cette architecture (fonctionnelle, logique, matérielle, etc.) doit être modélisée en UML et pas seulement documentée en texte ;
- **Piloté par les risques** : les risques majeurs du projet doivent être identifiés au plus tôt, mais surtout levés le plus rapidement possible. Les mesures à prendre dans ce cadre déterminent l'ordre des itérations ;
- **Conduit par les cas d'utilisation** : le projet est mené en tenant compte des besoins et des exigences des utilisateurs. Les cas d'utilisation du futur système sont identifiés, décrits avec précision et priorisés.

Afin de concrétiser notre projet, nous allons suivre la méthodologie suivante :

- Nous commencerons d'abord par définir les besoins fonctionnels et non fonctionnels du projet ;
- Ensuite, nous allons élaborer les diagrammes nécessaires à l'analyse des besoins et à la conception à savoir le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classe ;
- Enfin, nous entamerons l'implémentation de l'application et du serveur web.

1.10 Présentation UML

UML (Unified Modeling Language), est un langage permettant de modéliser nos classes et leurs interactions. Autrement dit, c'est un ensemble de notations graphiques s'appuyant sur des diagrammes et permettant de spécifier, visualiser et de documenter les systèmes logiciels orientés-objet[10].

UML s'impose aujourd'hui comme langage de modélisation objet standardisé par l'OMG en 1997 pour la conception des logiciels. La dernière version de la spécification validée par l'OMG est UML 2.5.1[11].

UML propose 14 types de diagrammes dont sept structurels et sept comportementaux, et il est destiné à faciliter la conception des documents nécessaires au développement d'un logiciel orienté objet, comme standard de modélisation de l'architecture logicielle. Les différents éléments représentables sont : activité d'un objet/logiciel, acteurs, processus, schéma de base de données, composants logiciels et réutilisation de composants[11].

Conclusion

Ce premier chapitre est une présentation générale du projet qui consiste à la création d'une application mobile liée à un serveur web pour une solution de monitoring de présence plus facile à manier et moins cher. Cette application offre dix besoins fonctionnels nécessaires pour une meilleure gestion du temps de présence des fonctionnaires. Dans le chapitre suivant nous allons traité la partie analyse des besoins et conception.

CHAPITRE 2

ANALYSE DES BESOINS ET CONCEPTION

Introduction

Dans ce chapitre, nous avons défini la description des principales étapes décrivant le processus de développement de l'application à réaliser. Cela consiste à identifier en premier lieu les diagrammes de cas d'utilisation qui décrivent les scénarios nominaux de chaque acteur interagissant avec le système. En second lieu, nous avons présenté les diagrammes de séquence qui représente les interactions entre acteurs et les différents objets. Enfin, le diagramme de classes de notre application, suivi de la présentation des classes et leurs attributs ont été menés à la fin de ce chapitre.

2.1 Identification des cas d'utilisations

Dans le système à développer, nous avons identifié les cas d'utilisations récentes dans le tableau 2.1 :

N°	Cas d'utilisation	Acteur
1	Authentification	Administrateur/Employé
2	Pointage	Administrateur/Employé
3	consultation de profil	Administrateur/Employé
4	consultation fiche de pointage	Administrateur/Employé
5	Gestion des comptes	Administrateur
6	Gestion des employés	Administrateur
7	Gestion des dispositifs	Administrateur
8	Gestion des évènements	Administrateur
9	Affichage statistiques	Administrateur
10	Consulter la fiche de pointage des employés	Administrateur
11	Gestion des plannings	Administrateur
12	Envoie/Réception des informations	Administrateur

TABLE 2.1 – Les cas d'utilisations associés au système à développer.

2.2 Description textuelle des cas d'utilisation

Une description textuelle des interactions entre l'acteur, le système, et les actions que le système doit réaliser doivent être associées à chaque cas d'utilisation en vue de produire les résultats attendus par les acteurs.

— Description textuelle du cas d'utilisation « Authentification »

Cas d'utilisation N°1	Authentification
Objectif	Vérification de l'identité de l'administrateur/employé
Acteur	Administrateur/employé
Pré condition	- Application fonctionnelle ; - Connexion internet.

Scénario nominal	<p>[Début]</p> <ol style="list-style-type: none"> 1. L'acteur demande l'accès à la page d'authentification ; 2. Le système affiche la page d'authentification ; 3. L'administrateur saisit son login et mot de passe ; 4. Le système vérifie la conformité des informations fournies. <p>[Fin]</p>
Scénario alternatif	<ol style="list-style-type: none"> 1. Les informations fournies sont incorrectes ; 2. Le système affiche un message d'erreur ; 3. le système réaffiche le formulaire d'authentification et attend que l'administrateur ressaisisse ses informations.

TABLE 2.2 – Description du cas d'utilisation « Authentification ».

— Description textuelle du cas d'utilisation « Pointage »

Cas d'utilisation N2		Pointage
Entrée	Objectif	Scanner le QR code pour marquer l'entrée.
	Acteur	Administrateur/employé.
	pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	<p>[Début]</p> <ol style="list-style-type: none"> 1. L'administrateur/l'employé demande l'interface de scanner pour l'entrée ; 2. Le système affiche l'interface demandée ; 3. L'administrateur/l'employé scanne le QR code ; 4. Le système envoie les informations de l'administrateur / l'employé au serveur. <p>[Fin]</p>
Scénario alternatif	<ol style="list-style-type: none"> 1. Le QR code scanner n'est pas conforme ; 2. Le système affiche un message d'erreur. 	

Sortie	Objectif	Scanner le QR code pour marquer la sortie
	Acteur	Administrateur/employé
	Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur/l'employé demande l'interface de scanner pour la sortie ; 2. Le système affiche l'interface demandée ; 3. L'administrateur/l'employé scanne le QR code ; 4. Le système envoie les informations de l'administrateur/l'employé au serveur. [Fin]
	Scénario alternatif	1. Le QR code scanner n'est pas conforme ; 2. Le système affiche un message d'erreur.
Évènement	Objectif	Scanner le QR code pour un évènement
	Acteur	Administrateur/employé
	Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur/l'employé demande l'interface de scanne pour l'évènement ; 2. Le système affiche l'interface demandée ; 3. L'administrateur/l'employé scanner le QR code ; 4. Le système envoie les informations de l'administrateur/l'employé au serveur. [Fin]
	Scénario alternatif	1. Le QR code scanner n'est pas conforme ; 2. Le système affiche un message d'erreur.

TABLE 2.3 – Description du cas d'utilisation «Pointage».

— Description textuelle du cas d'utilisation « consultation de profil »

Cas d'utilisation N°3		Consultation de profil
Consultation mon profil	Objectif	Permettre à l'administrateur/l'employé de consulter son profil.
	Acteur	Administrateur/employé.
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur/l'employé demande de consulter son profil ; 2. Le système affiche le profil. [Fin]
	Scénario alternatif	Aucun
Consultation mon planning	Objectif	Permettre à l'administrateur/l'employé de consulter son profil.
	Acteur	Administrateur/employé.
	Pré conditions	- Application fonctionnelle ; - connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur/l'employé demande de consulter son planning ; 2. Le système affiche le planning. [Fin]
	Scénario alternatif	Aucun.

TABLE 2.4 – Description du cas d'utilisation «Consultation de profil ».

— Description textuelle du cas d'utilisation « Consultation fiche de pointage »

Cas d'utilisation N4	Consultation fiche de pointage
Objectif	Permettre à l'administrateur/l'employé de consulter sa fiche de pointage.
Acteur	Administrateur/employé.
Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
Scénario nominal	[Début] 1. L'administrateur/l'employé demande de consulter sa fiche de pointage ; 2. Le système affiche la fiche de pointage. [Fin]
Scénario alternatif	Aucun

TABLE 2.5 – Description du cas d'utilisation « Consultation fiche de pointage ».

— Description textuelle du cas d'utilisation « Gestion des comptes »

Cas d'utilisation N5	Gestion des comptes	
Ajouter	Objectif	Ajouter un compte
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur demande la page d'ajout d'un compte ; 2. Le système affiche le formulaire d'ajout ; 3. L'administrateur remplit le formulaire ; 4. L'administrateur confirme l'envoi des informations saisis ; 5. Le système confirme l'ajout. [Fin]
	Scénario alternatif	En cas de champs vides le système affiche un message d'erreur.

Modifier	Objectif	Modifier les informations d'un compte
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un compte et demande de modifier les informations du compte sélectionné ; 2. Le système affiche le formulaire de modification ; 3. L'administrateur effectue les modifications désirées ; 4. L'administrateur confirme l'envoi des informations saisie ; 5. Le système confirme la modification. [Fin]
	Scénario alternatif	En cas de saisie de données non compatibles aux champs le système affiche un message d'erreur.
Supprimer	Objectif	Supprimer un compte
	Acteur	Administrateur
	Pré condition	- Application fonctionnelle ; - Connexion internet - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un compte et demande sa suppression ; 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun

TABLE 2.6 – Description du cas d'utilisation «Gestion des comptes».

— Description textuelle du cas d'utilisation « Gestion des employés »

Cas d'utilisation N6		Gestion des employés
Ajouter	Objectif	Ajouter un employé
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; -Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur demande la page d'ajout d'un employé ; 2. Le système affiche le formulaire d'ajout ; 3. L'administrateur remplit le formulaire ; 4. L'administrateur confirme l'envoi des informations saisis ; 5. Le système confirme l'ajout. [Fin]
	Scénario alternatif	En cas de champs vides le système affiche un message d'erreur.
Modifier	Objectif	Modifier les informations d'un employé
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un employé et demande de modifier les informations de l'employé sélectionné ; 2. Le système affiche le formulaire de modification ; 3. L'administrateur effectue les modifications désirées ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme la modification. [Fin]
	Scénario alternatif	En cas de saisie de données non compatibles aux champs le système affiche un message d'erreur.

Supprimer	Objectif	Supprimer un employé
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un employé et demande sa suppression ; 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. Le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun.
Archiver	Objectif	Archiver un employé
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un employé et demande son archivage 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. Le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun

TABLE 2.7 – Description du cas d'utilisation « Gestion des employés ».

— Description textuelle du cas d'utilisation « Gestion des dispositifs »

Cas d'utilisation N7		Gestion des dispositifs
Ajouter	Objectif	Ajouter un dispositif
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un employé et demande l'ajout d'un dispositif ; 2. Le système affiche le formulaire d'ajout ; 3. L'administrateur remplit le formulaire ; 4. L'administrateur confirme l'envoi des informations saisis ; 5. Le système confirme l'ajout. [Fin]
	Scénario alternatif	En cas de champs vides le système affiche un message d'erreur.
Modifier	Objectif	Modifier les informations d'un dispositif
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un dispositif et demande de modifier les informations du dispositif sélectionné ; 2. Le système affiche le formulaire de modification ; 3. L'administrateur effectue les modifications désirées ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme la modification. [Fin]
	Scénario alternatif	En cas de saisie de données non compatibles aux champs le système affiche un message d'erreur.

Supprimer	Objectif	Supprimer un dispositif
	Acteur	Administrateur
	Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un dispositif et demande sa suppression ; 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun

TABLE 2.8 – Description du cas d'utilisation «Gestion des dispositifs».

— Description textuelle du cas d'utilisation « Gestion des événements »

Cas d'utilisation N8		Gestion des évènements
Ajouter	Objectif	Ajouter un évènement
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur demande l'ajout d'un évènement ; 2. Le système affiche le formulaire d'ajout ; 3. L'administrateur remplit le formulaire ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme l'ajout. [Fin]
	Scénario alternatif	En cas de champs vides le système affiche un message d'erreur.

Modifier	Objectif	Modifier les informations d'un évènement
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un évènement et demande de modifier les informations de l'évènement sélectionné ; 2. Le système affiche le formulaire de modification ; 3. L'administrateur effectue les modifications désirées ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme la modification. [Fin]
	Scénario alternatif	En cas de saisie de données non compatibles aux champs le système affiche un message d'erreur.
Supprimer	Objectif	Supprimer un évènement
	Acteur	Administrateur
	Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un évènement et demande sa suppression ; 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun

TABLE 2.9 – Description du cas d'utilisation «Gestion des évènements».

— Description textuelle du cas d'utilisation « Affichage statistiques »

Cas d'utilisation N9	Affichage statistiques
Objectif	Afficher les statistiques des employés
Acteur	Administrateur
Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
Scénario nominal	[Début] 1. L'administrateur demande les statistiques ; 2. Le système affiche les statistiques. [Fin]
Scénario alternatif	Aucun

TABLE 2.10 – Description du cas d'utilisation « Affichage statistiques ».

— Description textuelle du cas d'utilisation « Consultation fiche de pointage des employés »

Cas d'utilisation N10	Consultation fiche de pointage des employés
Objectif	Permettre à l'administrateur de consulter la fiche de pointage des employés.
Acteur	Administrateur
Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
Scénario nominal	[Début] 1. L'administrateur demande l'interface consultation fiche de pointage des employés ; 2. Le système affiche l'interface demandée ; 3. L'administrateur selectionne un employé ; 4. Le système affiche la fiche de pointage de l'employé [Fin]
Scénario alternatif	Aucun

TABLE 2.11 – Description du cas d'utilisation « Consultation fiche de pointage des employés ».

— Description textuelle du cas d'utilisation « Gestion des plannings »

Cas d'utilisation N11		Gestion des plannings
Ajouter	Objectif	Ajouter un planning
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur demande l'ajout d'un planning ; 2. Le système affiche le formulaire d'ajout ; 3. L'administrateur remplit le formulaire ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme l'ajout. [Fin]
	Scénario alternatif	En cas de champs vides le système affiche un message d'erreur.
Modifier	Objectif	Modifier un planning
	Acteur	Administrateur
	Pré conditions	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un planning et demande de le modifier ; 2. Le système affiche le formulaire de modification ; 3. L'administrateur effectue les modifications désirées ; 4. L'administrateur confirme l'envoi des informations saisies ; 5. Le système confirme la modification. [Fin]
	Scénario alternatif	En cas de saisie de données non compatibles aux champs le système affiche un message d'erreur.

Supprimer	Objectif	Supprimer un planning
	Acteur	Administrateur
	Pré condition	- Application fonctionnelle ; - Connexion internet ; - Authentification.
	Scénario nominal	[Début] 1. L'administrateur sélectionne un planning et demande sa suppression ; 2. Le système affiche la demande de confirmation ; 3. L'administrateur confirme ou refuse la demande ; 4. le système exécute le choix de l'administrateur. [Fin]
	Scénario alternatif	Aucun

TABLE 2.12 – Description du cas d'utilisation «Gestion des plannings».

2.3 Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation est un graphe d'acteurs, et un ensemble de cas d'utilisation, il est destiné à représenter les besoins des utilisateurs par rapport au système et il constitue l'un des diagrammes les plus structurants dans l'analyse d'un système [12].

2.3.1 Diagramme de cas d'utilisation général

Ce diagramme 2.1 regroupe tous les cas d'utilisations relatifs aux acteurs de notre système qui sont l'employé et l'administrateur, chaque scénario de ces cas d'utilisation, doit passer par le cas d'utilisation « Authentification », ce qui explique l'utilisation de la relation « include ».

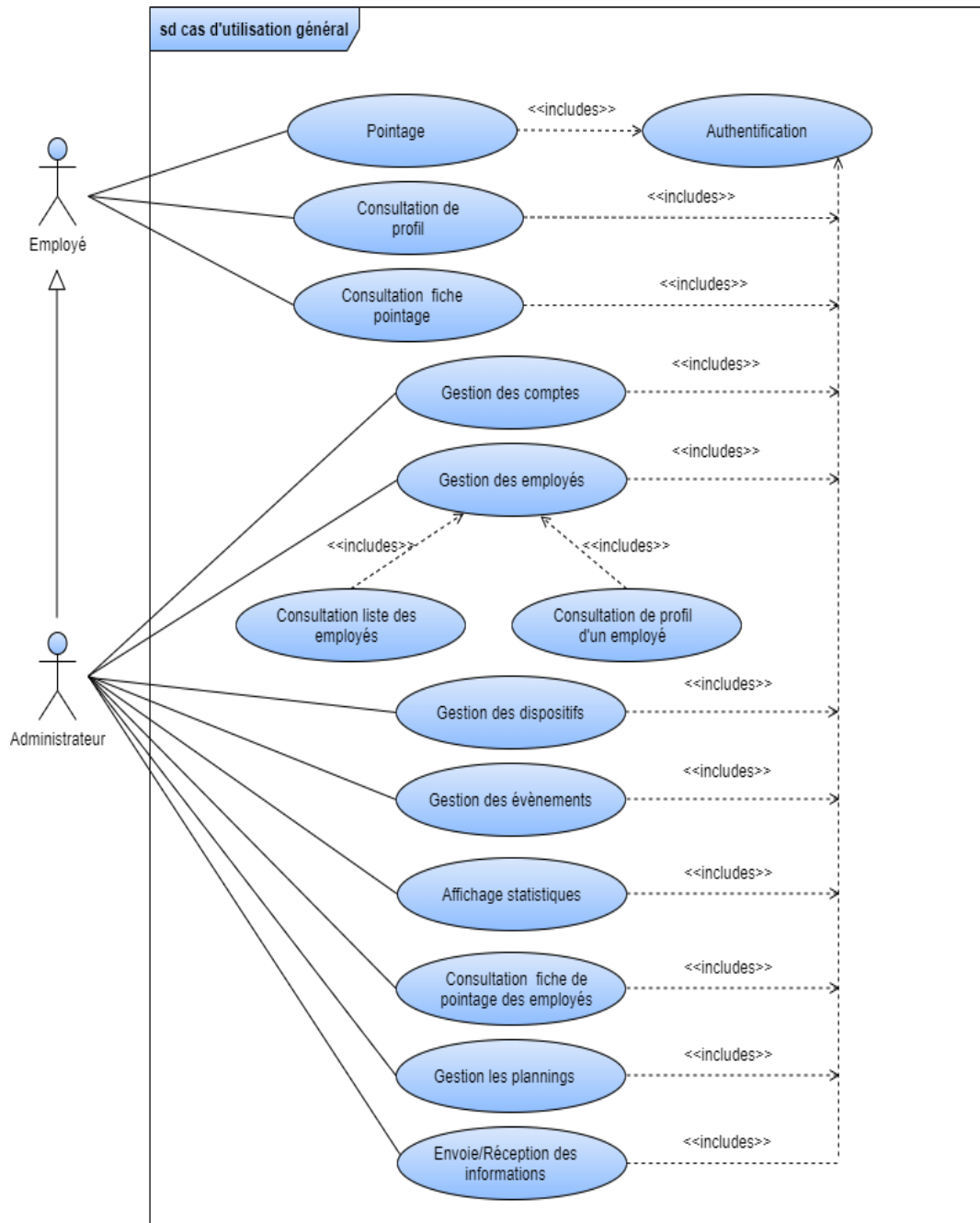


FIGURE 2.1 – Diagramme de cas d'utilisation général.

2.3.2 Diagramme de cas d'utilisation « Pointage »

Ce présent diagramme représente le cas d'utilisation pointage associé à l'acteur « administrateur ».

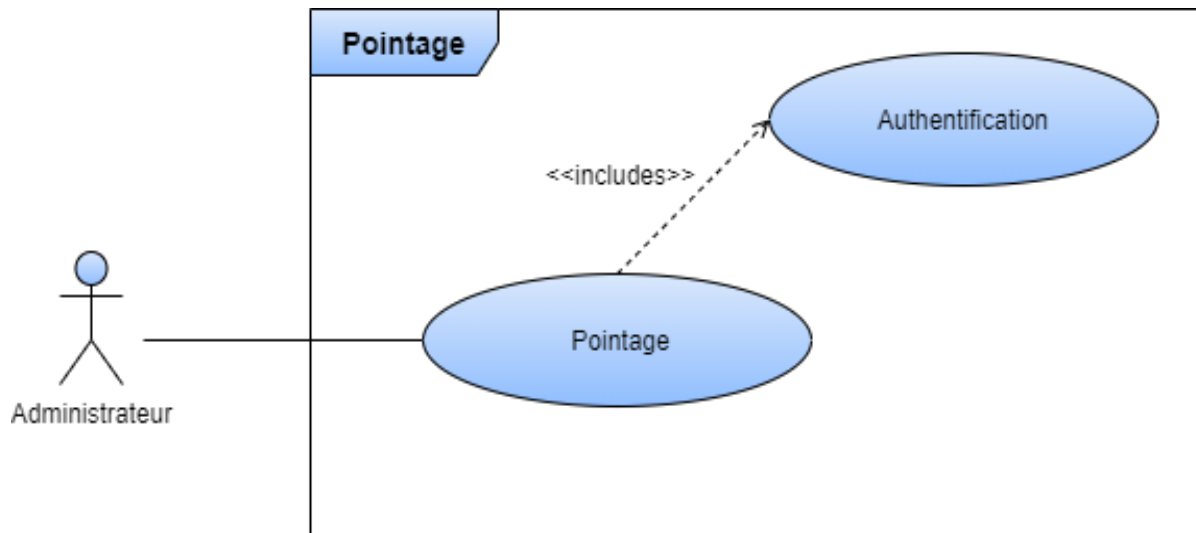


FIGURE 2.2 – Diagramme de cas d'utilisation «pointage».

2.3.3 Diagramme de cas d'utilisation « Consultation de profil »

Ce diagramme représente le cas d'utilisation consultation de profil.

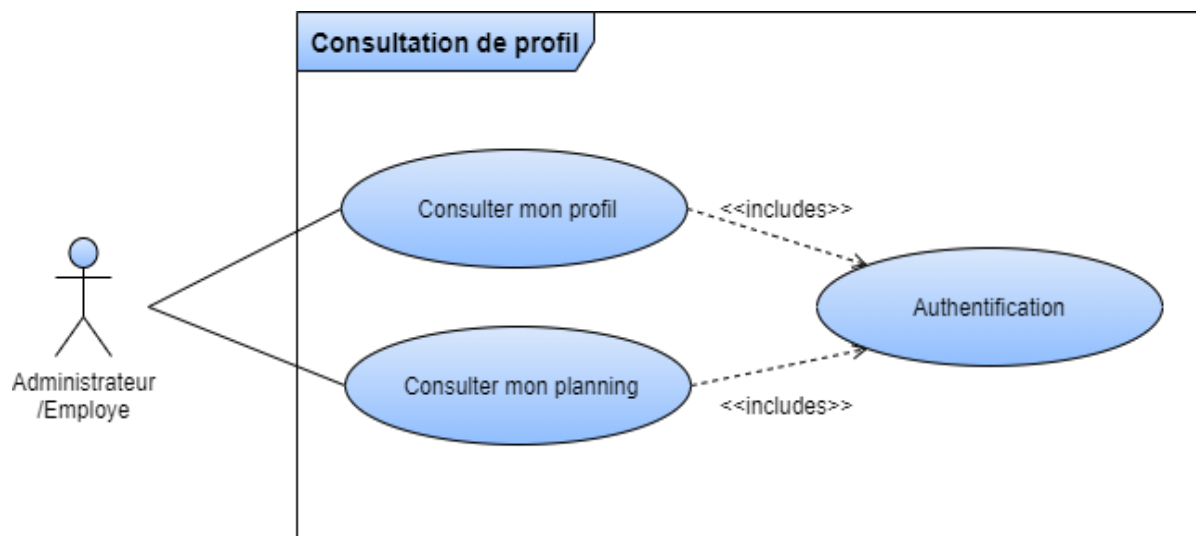


FIGURE 2.3 – Diagramme du cas d'utilisation « Consultation de profil »

2.3.4 Diagramme de cas d'utilisation « Gestion des comptes »

Ce diagramme décrit la façon avec laquelle est faite la gestion des comptes par l'administrateur une fois précédée par une authentification. La gestion de compte comporte trois fonctionnalités à savoir l'ajout, la modification et la suppression.

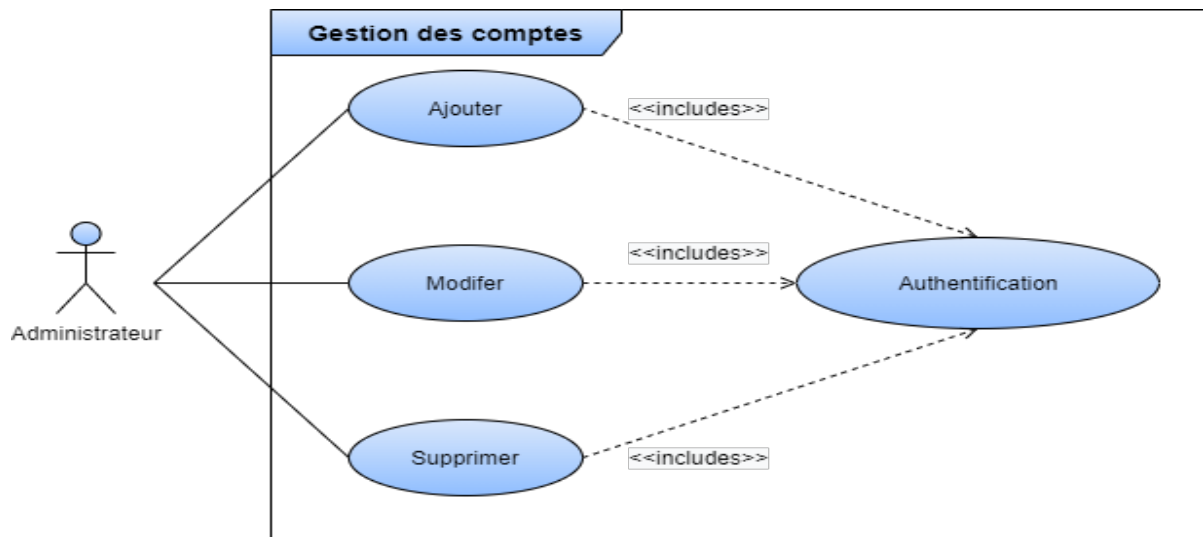


FIGURE 2.4 – Diagramme de cas d'utilisation «Gestion des comptes».

2.3.5 Diagramme de cas d'utilisation « Gestion des employés »

Ce diagramme décrit la façon avec laquelle est faite la gestion des employés par l'administrateur après son authentification. De plus des fonctionnalités citées précédemment la gestion des employés inclue ainsi le service d'archivage.

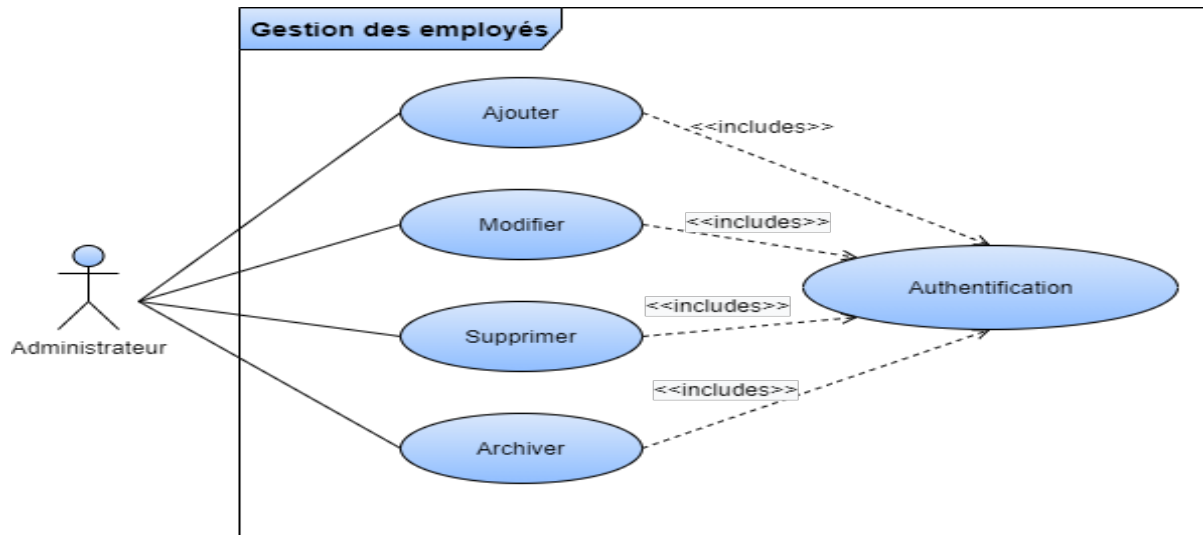


FIGURE 2.5 – Diagramme de cas d'utilisation «Gestion des employés».

2.3.6 Diagramme de cas d'utilisation «Gestion des dispositifs »

Le diagramme ci-après, illustre le cas d'utilisation « gestion des dispositifs » qui comprend l'ajout, la modification et la suppression d'un ou plusieurs dispositifs.

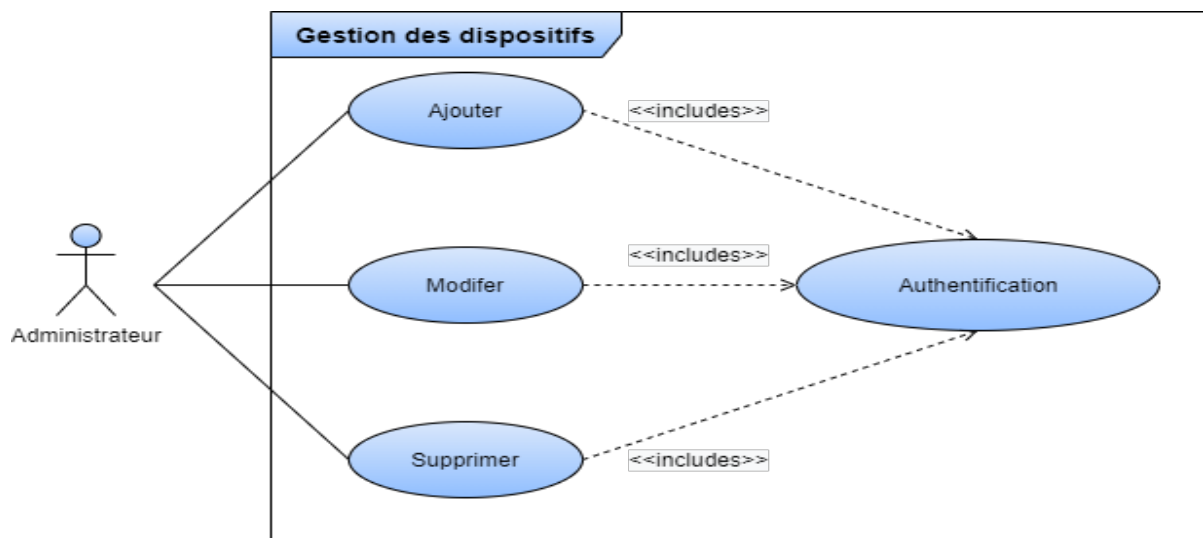


FIGURE 2.6 – Diagramme de cas d'utilisation «Gestion des dispositifs».

2.3.7 Diagramme de cas d'utilisation «Gestion des évènements»

Ce diagramme représente la gestion des évènements qui consiste à l'ajout, la modification et la suppression d'un ou plusieurs évènements. Un évènement peut être : une réunion, une absence, une sortie anticipée, ou un congé.

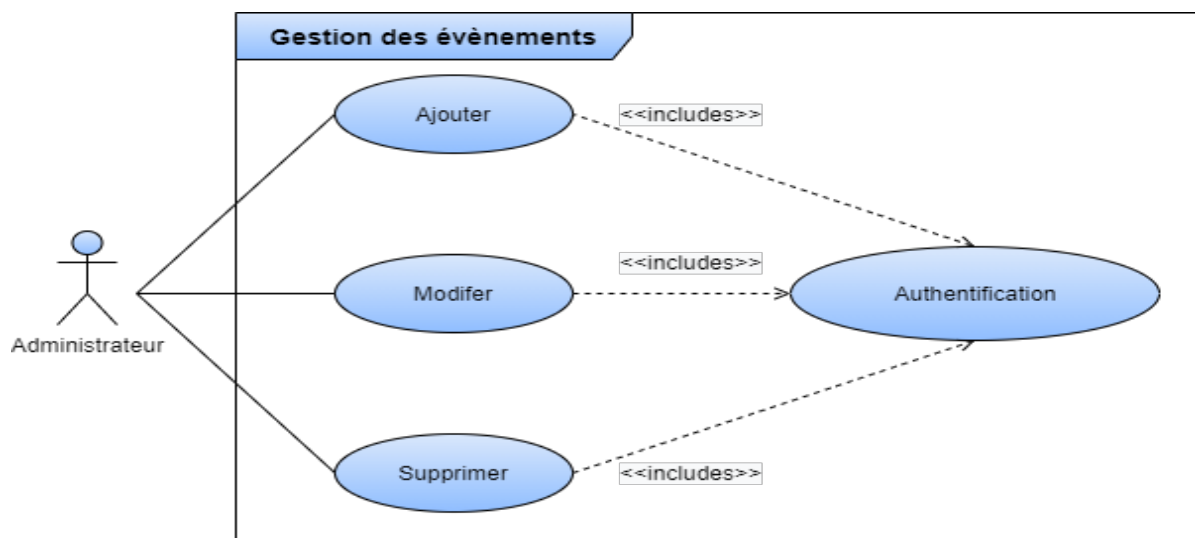


FIGURE 2.7 – Diagramme de cas d'utilisation «Gestion des évènements».

2.3.8 Diagramme de cas d'utilisation « Affichage statistiques »

Ce diagramme représente le cas d'utilisation affichage

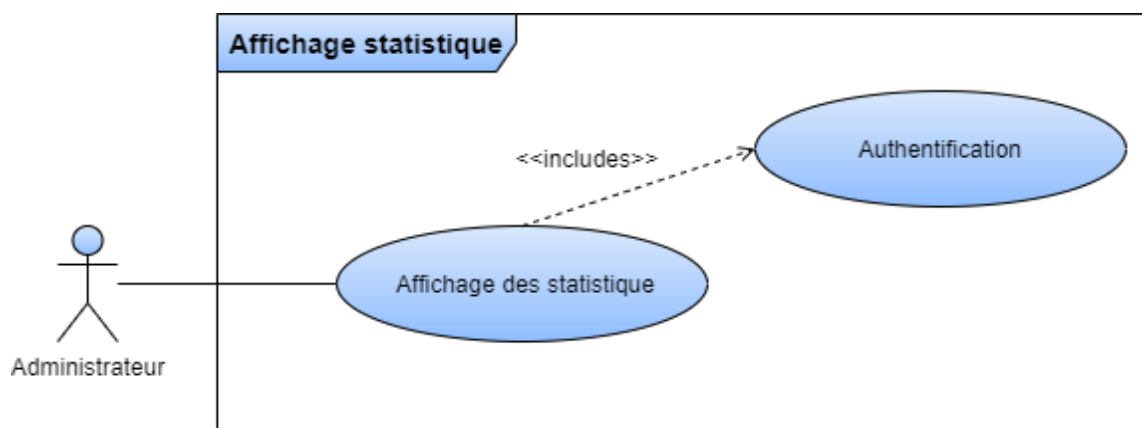


FIGURE 2.8 – Diagramme de cas d'utilisation «Affichage statistiques».

2.3.9 Diagramme de cas d'utilisation « Réception/envoi des informations »

Ce diagramme représente le cas d'utilisation réception/envoi d'informations.

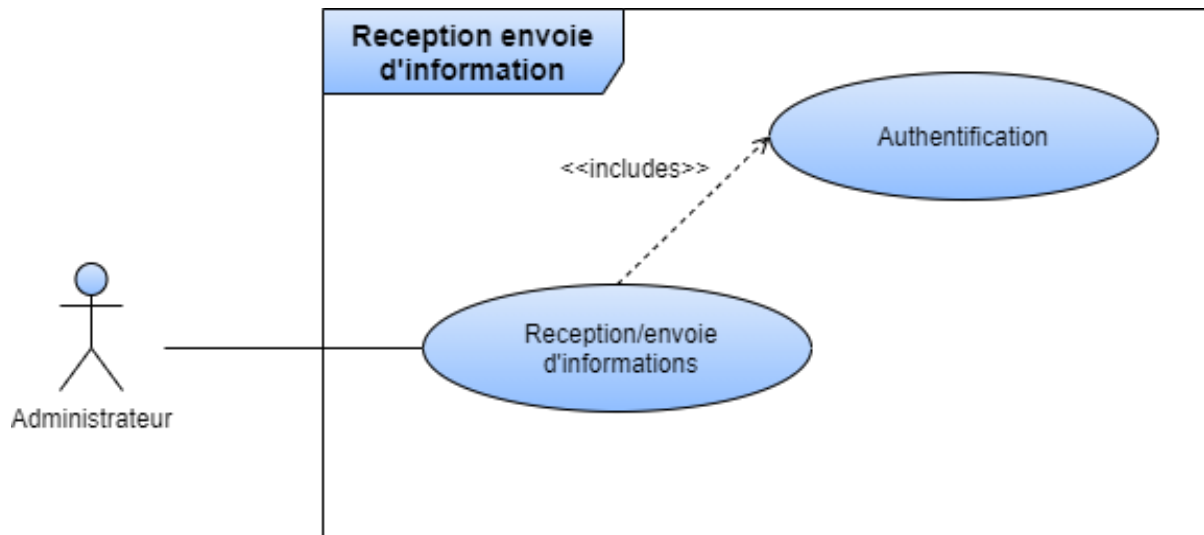


FIGURE 2.9 – Diagramme de cas d'utilisation «Réception/Envoie d'information» .

2.4 Diagramme de séquence

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont.

Les diagrammes de séquence sont organisés en fonction du temps. Le temps s'écoule au fur et à mesure que vous parcourez la page. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence de messages[13]. En se basant sur les opérateurs de UML 2.0, nous présentons les diagrammes de séquence des cas d'utilisation de l'application et du serveur web que nous allons mettre en œuvre.

2.4.1 Diagramme de séquence du cas d'utilisation

« Authentification »

L'authentification consiste à assurer la confidentialité des données, elle se base sur la vérification des informations associées à l'administrateur/employé (généralement un login et un mot de passe). Ces informations sont préétablies dans une base de données. Lors de l'authentification d'un utilisateur, deux cas peuvent se présenter : informations correctes ou informations incorrectes, ce qui explique l'utilisation de l'opérateur "alt". Si les informations fournies sont correctes, alors le système accorde l'accès à l'interface appropriée. En revanche, si l'utilisateur saisit des informations incorrectes, le système génère un message d'erreur et réaffiche la page d'authentification. Ce procédé est exécuté à chaque fois que l'utilisateur tente de s'authentifier, c'est pourquoi nous avons utilisé l'opérateur "loop".

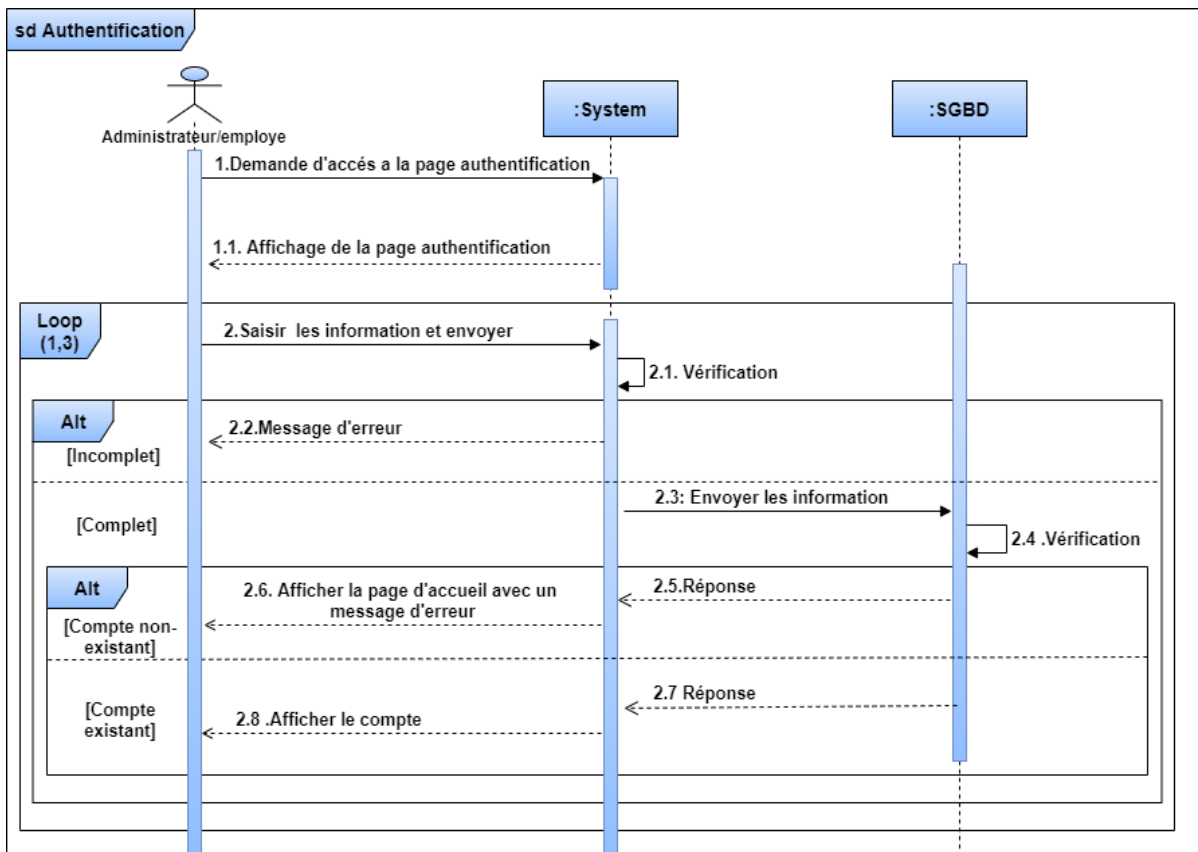


FIGURE 2.10 – Diagramme de séquence de cas d'utilisation « Authentification ».

2.4.2 Diagramme de séquence du cas d'utilisation « pointage »

Après une authentification (utilisation de l'opérateur 'ref') réussie, Le fragment utilisé est de type « opt » pour indiquer que les actions effectuées par l'administrateur/employé peuvent arriver dans n'importe quel ordre d'une manière optionnelle. Trois scénarios sont représentés :le scanne pour une entrée, une sortie, ou un événement. Ensuite une fois l'une des trois propositions est choisi le système affiche l'interface de pointage l'administrateur/employé scanne le code QR qui est envoyé à la base de données pour vérification. Deux cas se présentent à nous (d'où l'utilisation du fragment 'alt'). Si le code est bien scanné et associé à l'évènement alors il l'enregistre dans la base de données, Sinon il renvoie une erreur comme quoi le code n'est pas associé.

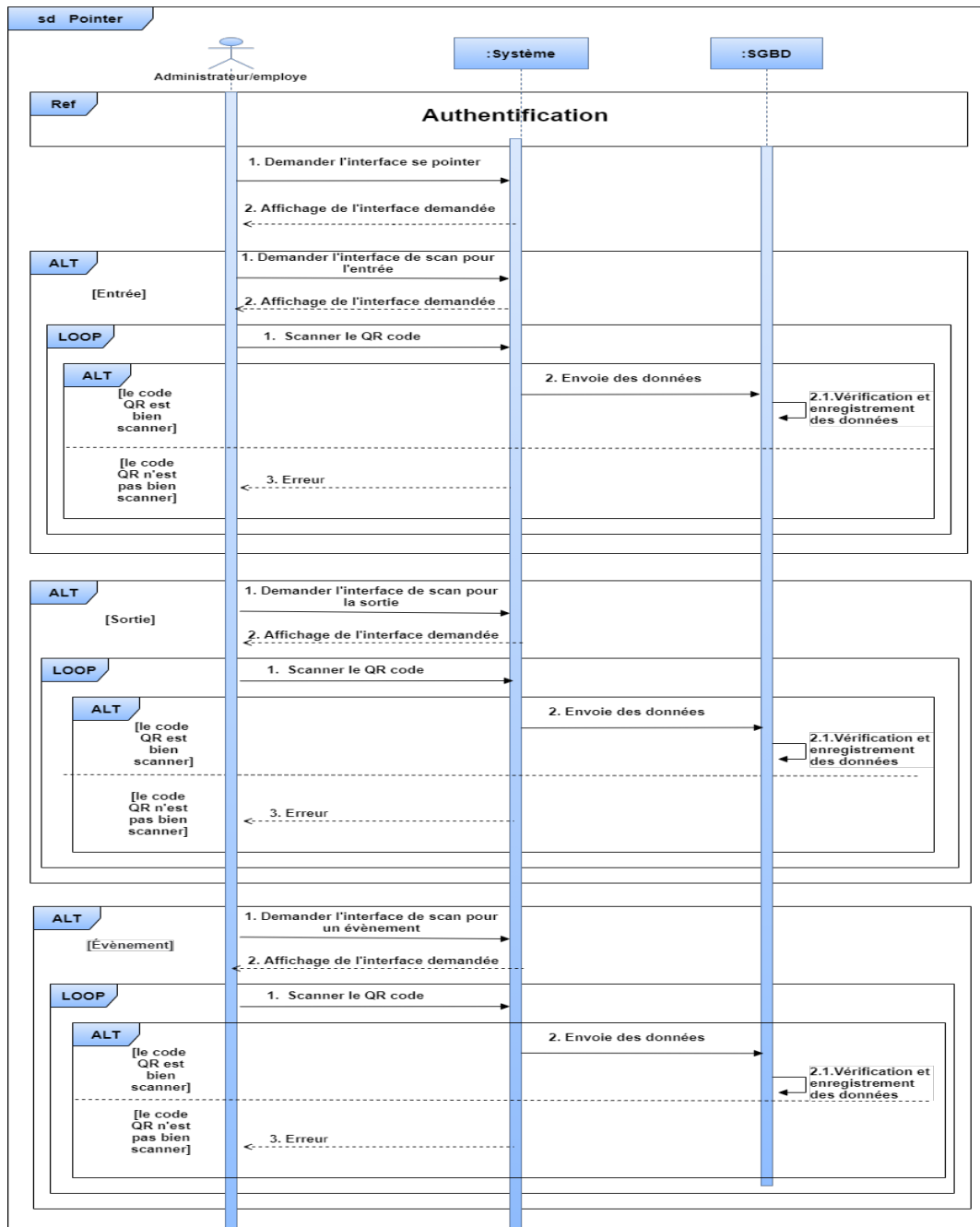


FIGURE 2.11 – Diagramme de séquence du cas d'utilisation «Pointage».

2.4.3 Diagramme de séquence du cas d'utilisation

« Consultation de profil »

Pour consulter le profil , l'administrateur/employé s'authentifie d'abord, puis Deux propositions se présentent (d'où l'utilisation du fragment 'opt'), use fois l'une des deux proposition choisi le système affiche l'interface demandée.

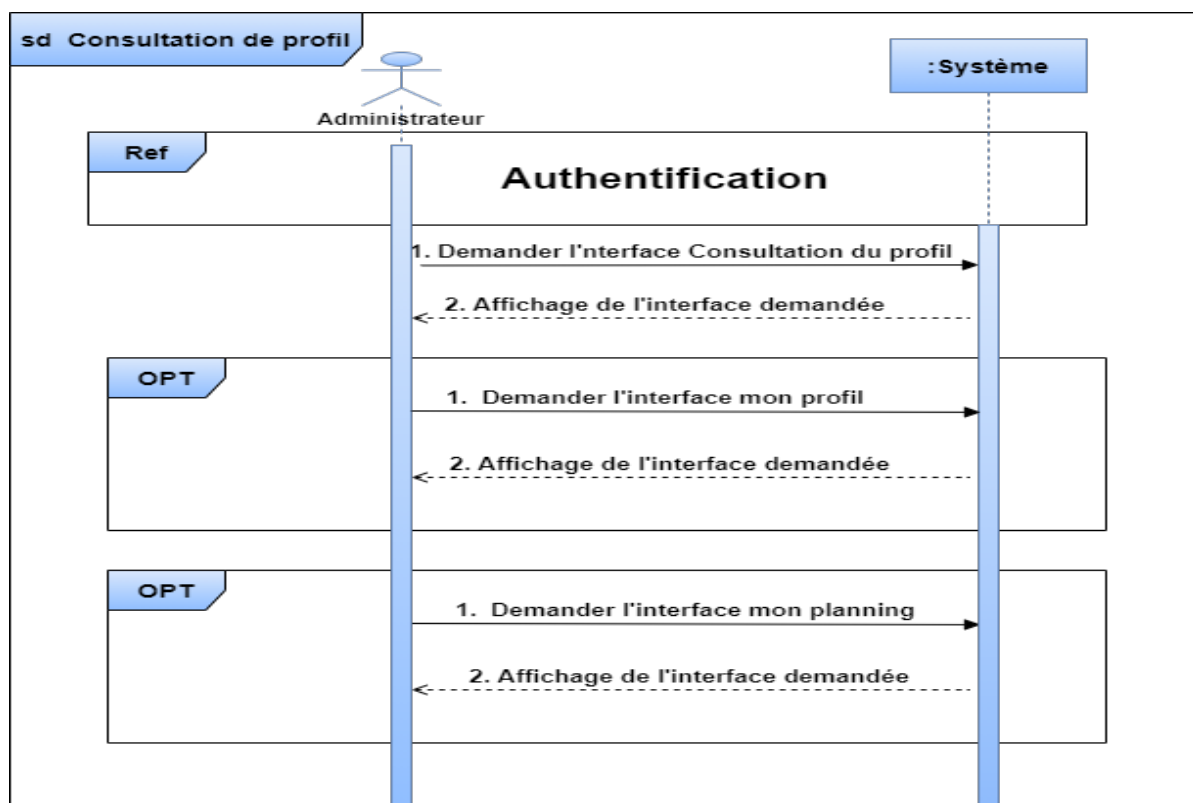


FIGURE 2.12 – Diagramme de séquence du cas d'utilisation « Consultation de profil ».

2.4.4 Diagramme de séquence du cas d'utilisation « Gérer les comptes »

Le fragment utilisé est de type « opt » pour indiquer que les actions effectuées par l'administrateur peuvent arriver dans n'importe quel ordre d'une manière optionnelle. Trois scénarios sont représentés :

- Ajout d'un compte : après que le système affiche le formulaire, l'administrateur introduit les informations qui définissent le compte à ajouter et valide l'opération ;

- Modification d'un compte : après l'affichage du formulaire, l'administrateur a la possibilité d'effectuer des modifications sur un compte déjà existant ;
- Suppression d'un compte : l'administrateur demande au système de supprimer un compte déjà existant.

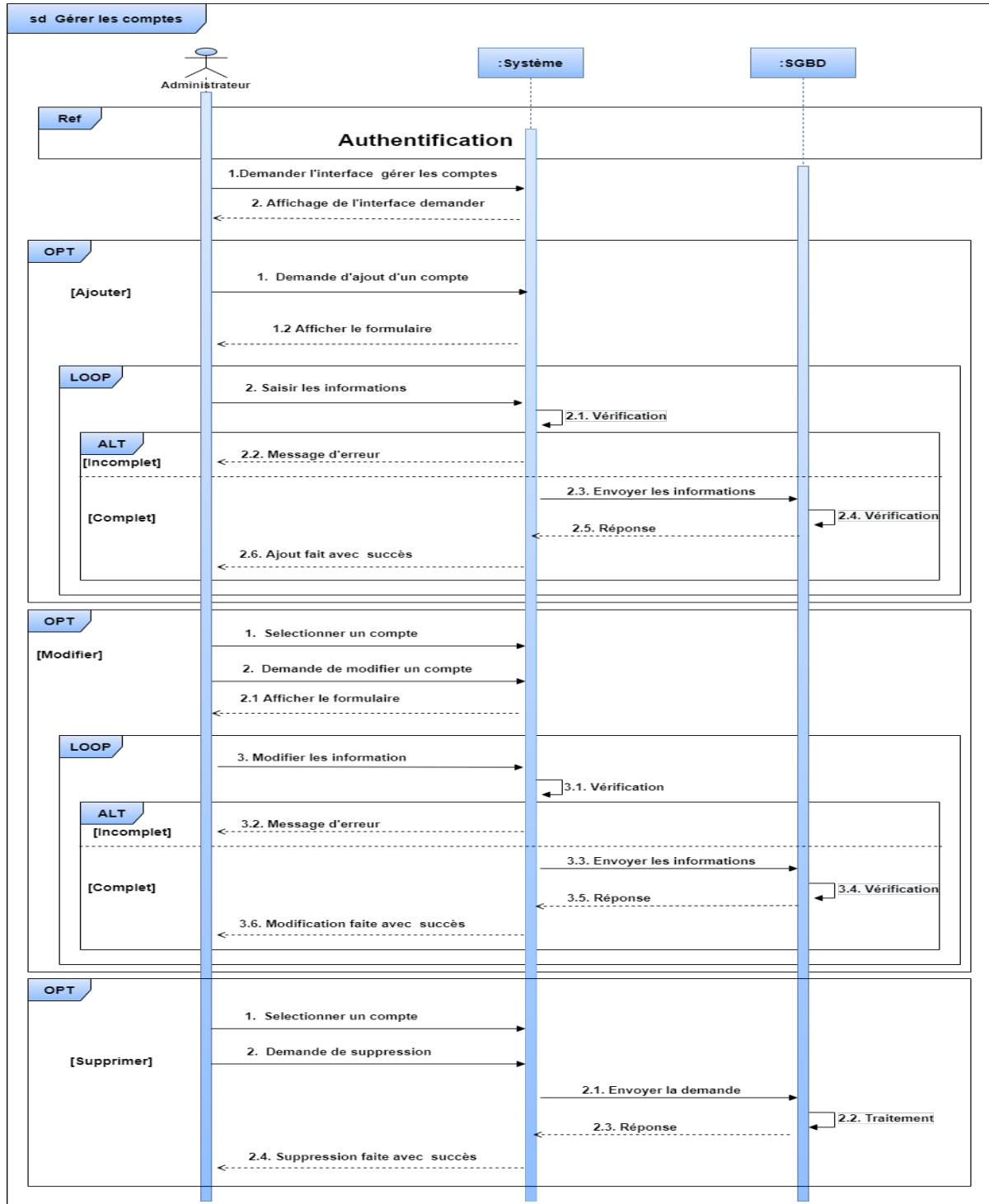
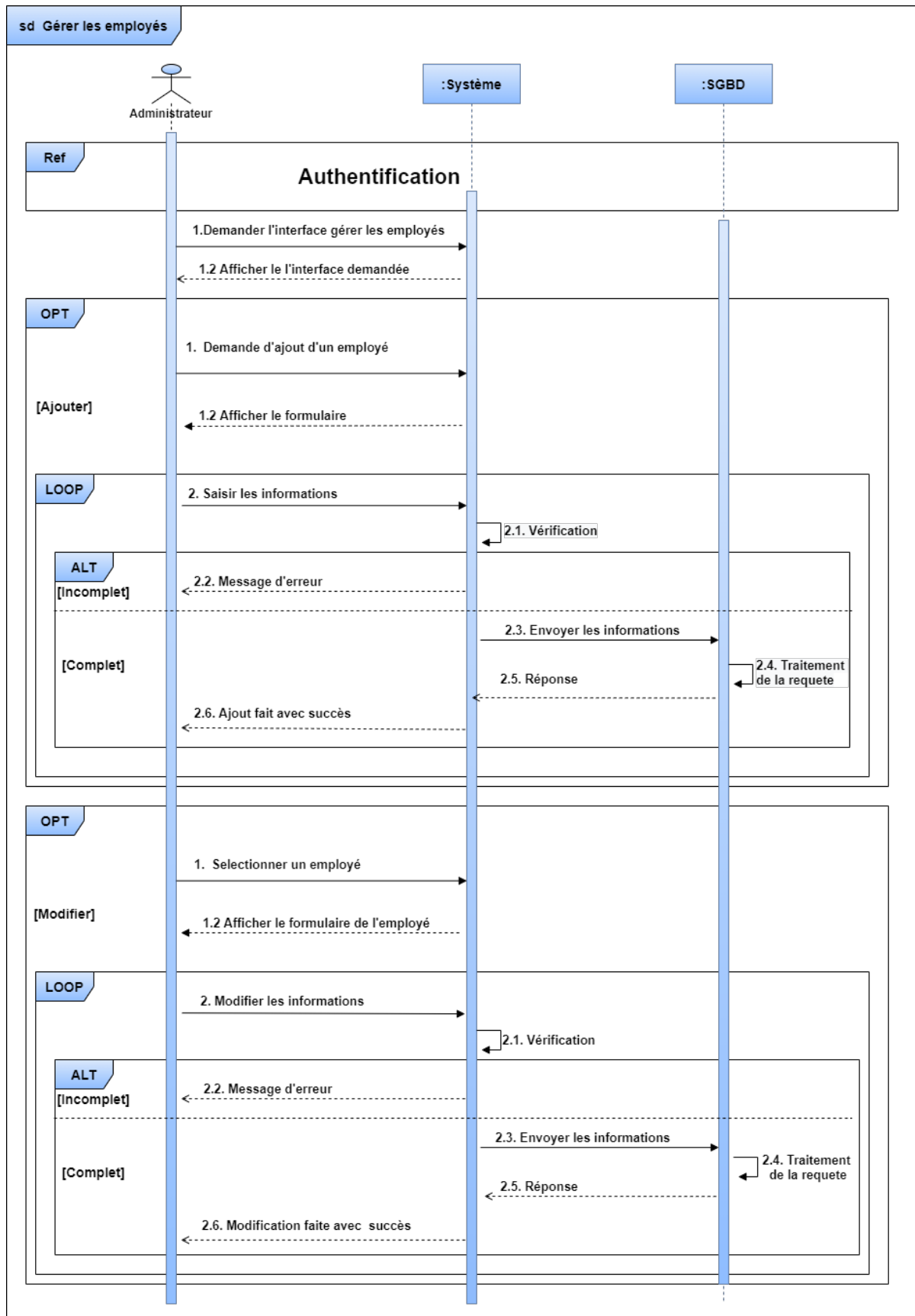


FIGURE 2.13 – Diagramme de séquence de cas d'utilisation «Gérer les comptes ».

2.4.5 Diagramme de séquence du cas d'utilisation « Gérer les employés »

Après authentification, l'administrateur effectue une demande de gestion d'un employé. Nous avons utilisé un fragment de type " opt « pour indiquer que les actions effectuées par l'administrateur peuvent arriver dans n'importe quel ordre d'une manière optionnelle. Quatre scénarios sont représentés :

- Ajout d'un employé : après que le système affiche le formulaire, l'administrateur introduit les informations qui définissent l'employé à ajouter et valide l'opération. Cette dernière est soumise à une vérification d'où le fragment « alt » ;
- Modification d'un employé : après l'affichage du formulaire, l'administrateur a la possibilité d'effectuer des modifications sur un employé déjà défini ;
- Suppression d'un employé : l'administrateur demande au système de supprimer un employé déjà défini ;
- Archivage d'un employé : l'administrateur demande au système d'archiver un employé déjà défini.



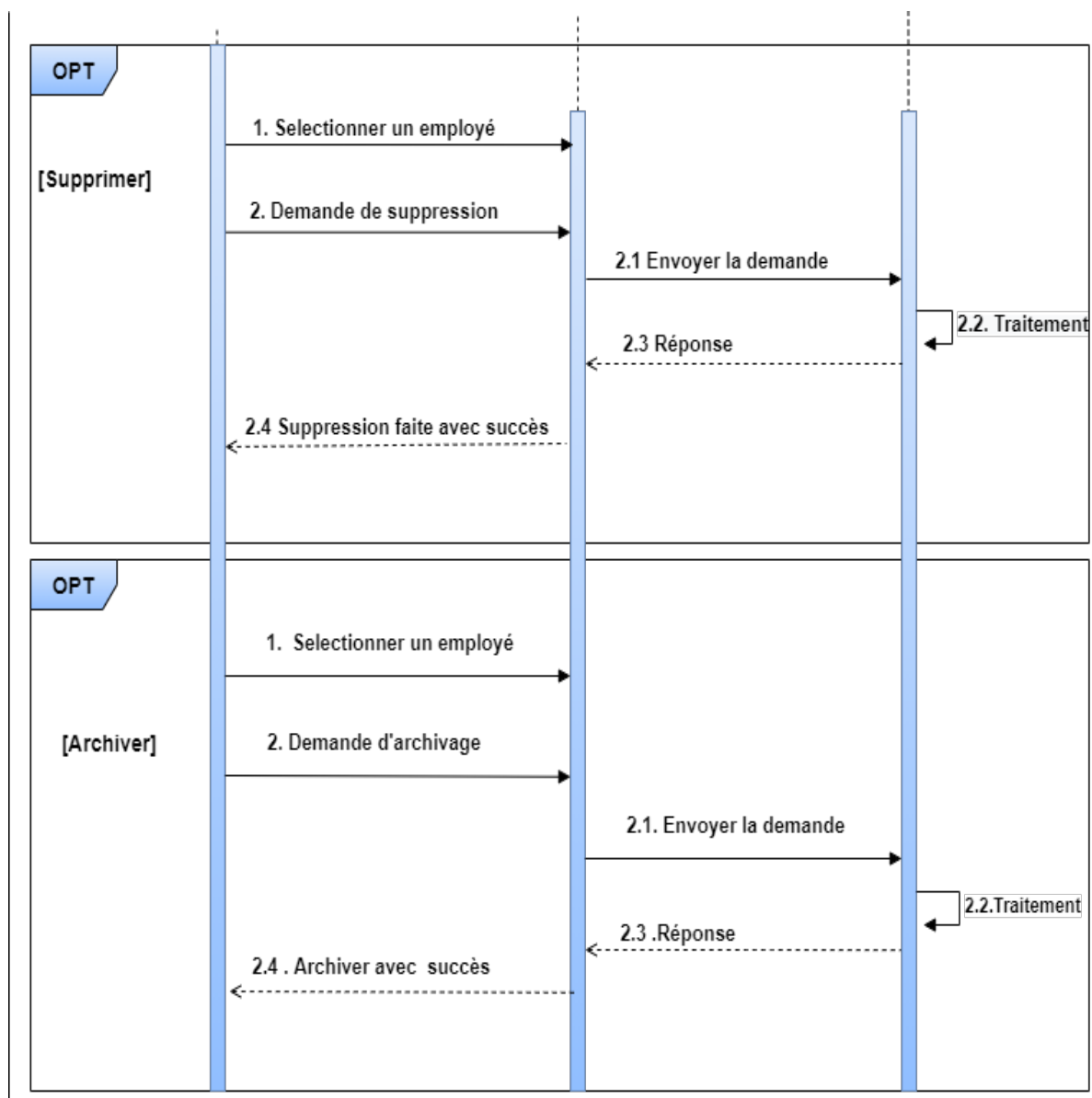
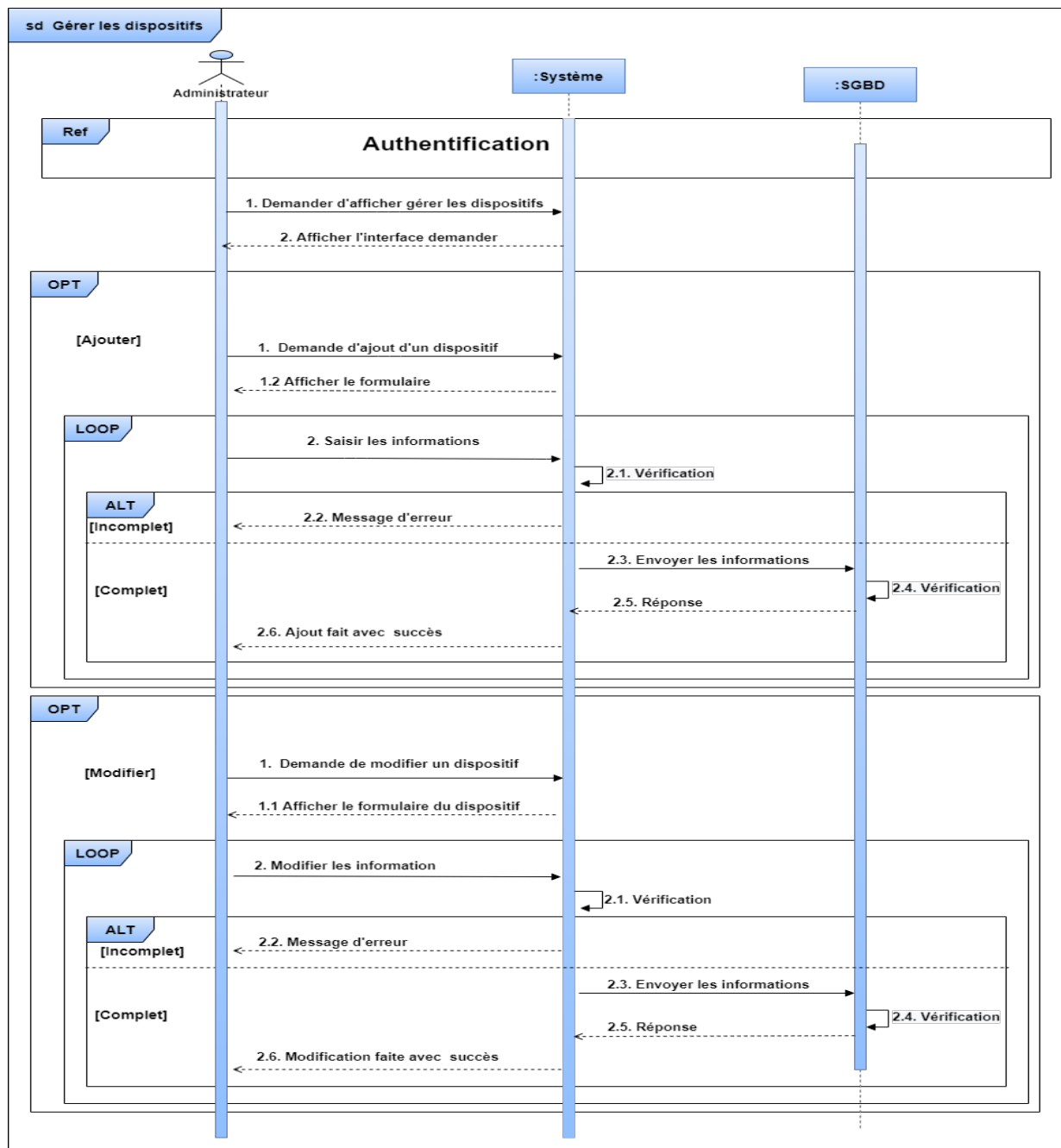


FIGURE 2.14 – Diagramme de séquence du cas d'utilisation «Gestion des employés».

2.4.6 Diagramme de séquence du cas d'utilisation « Gestion des dispositifs »

Après authentification, l'administrateur effectue une demande de gestion d'un dispositif. Nous avons utilisé un fragment de type " opt " pour indiquer que les actions effectuées par l'administrateur peuvent arriver dans n'importe quel ordre. Trois scénarios sont représentés, chacun d'entre eux correspond à une vérification après la validation d'un fragment « alt ».

- Ajout d'un dispositif : après que le système affiche le formulaire, l'administrateur introduit les informations qui définissent le dispositif à ajouter et valide l'opération ;
- Modification d'un dispositif : après l'affichage du formulaire, l'administrateur a la possibilité d'effectuer des modifications sur les informations d'un dispositif déjà défini ;
- Suppression d'un dispositif : l'administrateur demande au système de supprimer un dispositif déjà défini.



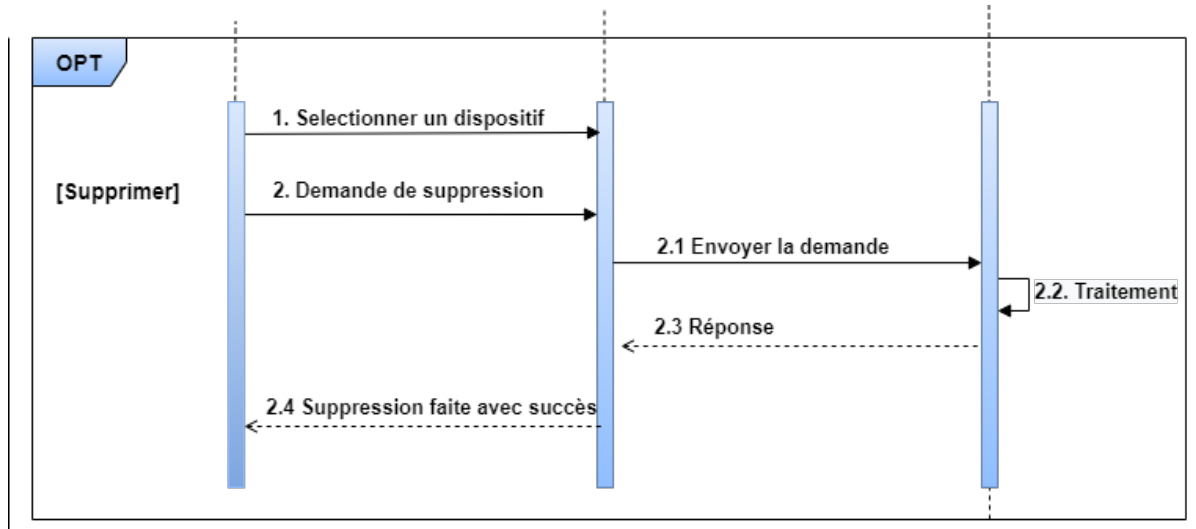
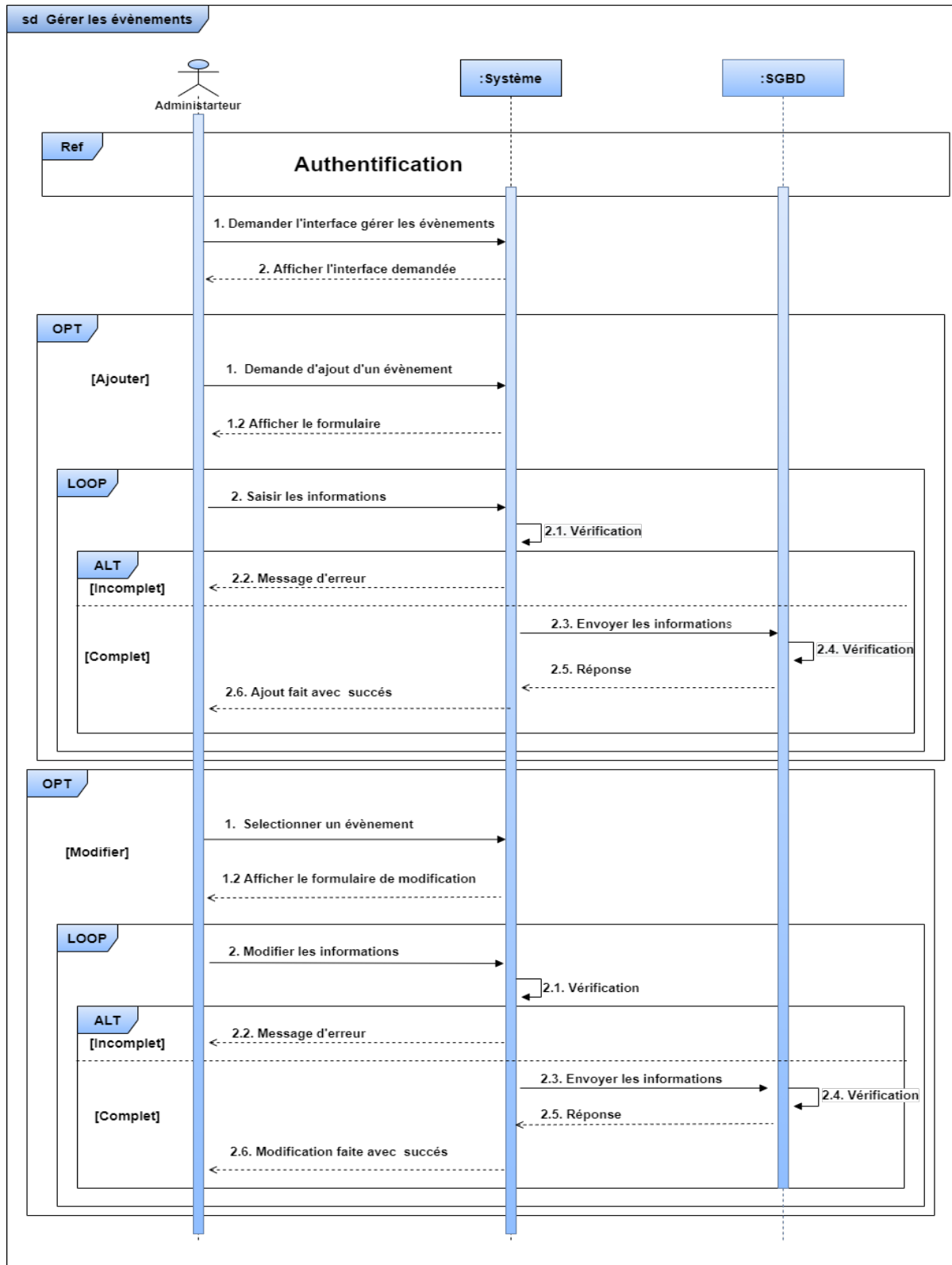


FIGURE 2.15 – Diagramme de séquence du cas d'utilisation «Gérer les dispositifs».

2.4.7 Diagramme de séquence du cas d'utilisation « Gérer les évènements »

Après authentification, l'administrateur effectue une demande de gestion d'un évènement. Nous avons utilisé un fragment de type " opt " pour indiquer que les actions effectuées par l'administrateur peuvent arriver dans n'importe quel ordre. Trois scénarios sont représentés , chacun d'entre eux correspond à une vérification après la validation du fragment « alt ».

- Ajout d'un évènement : après que le système affiche le formulaire, l'administrateur introduit les informations qui définissent l'évènement à ajouter et valide l'opération ;
- Modification d'un évènement : après l'affichage du formulaire, l'administrateur a la possibilité d'effectuer des modifications sur un évènement déjà défini ;
- Suppression d'un évènement : l'administrateur demande au système de supprimer un évènement déjà défini.



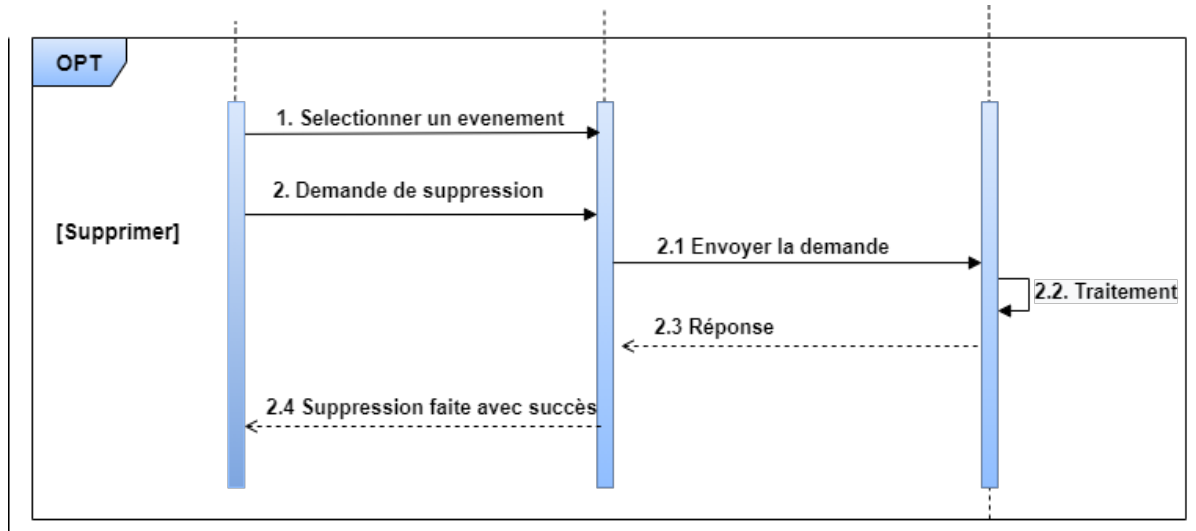


FIGURE 2.16 – Diagramme de séquence de cas d'utilisation «Gérer les évènements».

2.4.8 Diagramme de séquence du cas d'utilisation

« Consulter fiche de pointage des employés »

Pour consulter la fiche de pointage d'un employé, l'administrateur s'authentifie d'abord, puis le système affiche la liste de l'ensemble des employés. L'administrateur sélectionne un seul employé pour voir s'afficher la fiche de pointage de l'employé sélectionné.

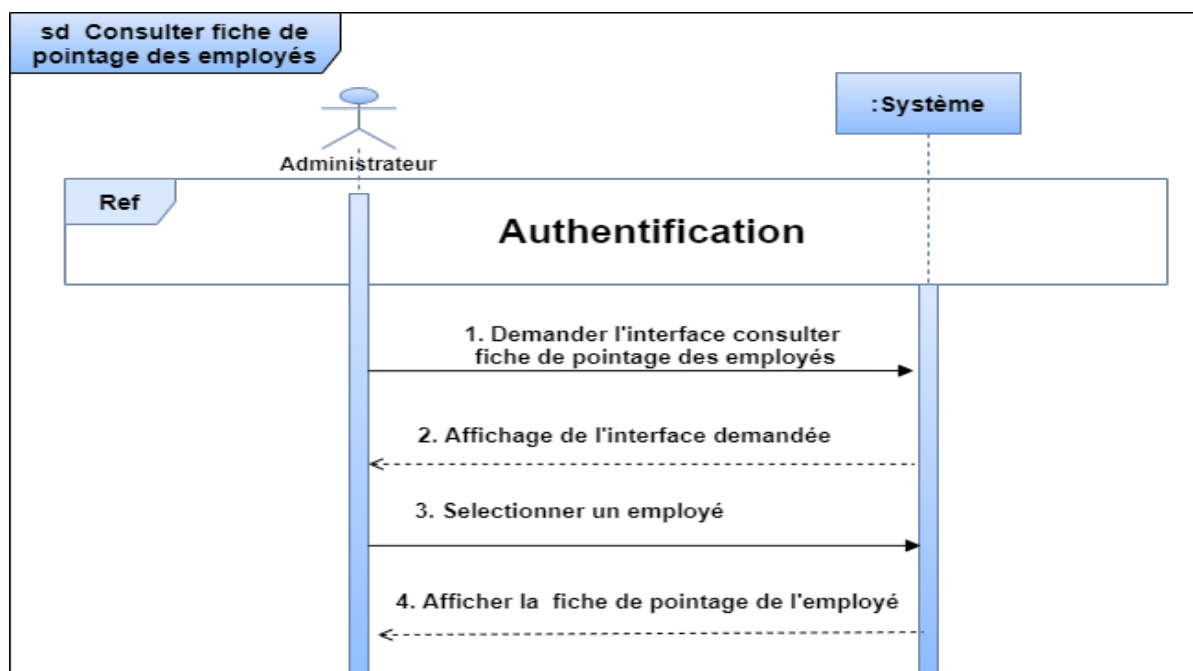
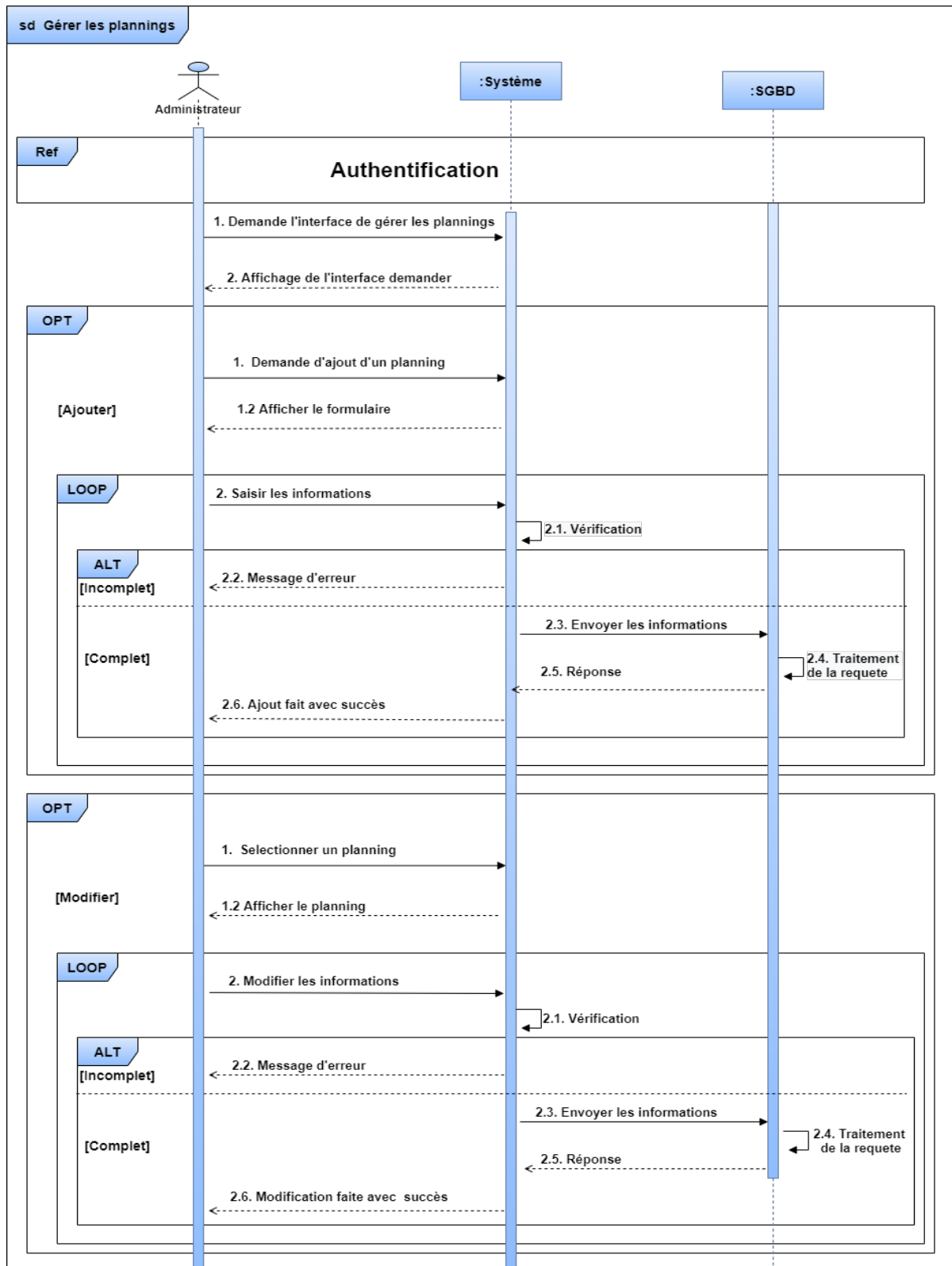


FIGURE 2.17 – Diagramme de séquence de cas d'utilisation « Consulter fiche de pointage des employés ».

2.4.9 Diagramme de séquence du cas d'utilisation « Gestion des plannings »

Après authentification, l'administrateur effectue une demande de gestion d'un planning. (un fragment de type " opt " est utilisé pour indiquer que les actions effectuées par l'administrateur peuvent arriver dans n'importe quel ordre. Cette opération a trois scénarios, chacun d'entre eux correspond à une vérification après la validation du fragment « alt ».

- Ajout d'un planning : après que le système affiche le formulaire, l'administrateur introduit les informations qui définissent le planning à ajouter et valide l'opération ;
- Modification d'un planning : après l'affichage du formulaire, l'administrateur a la possibilité d'effectuer des modifications sur un planning déjà défini ;
- Suppression d'un planning : l'administrateur demande au système de supprimer un Planning déjà défini.



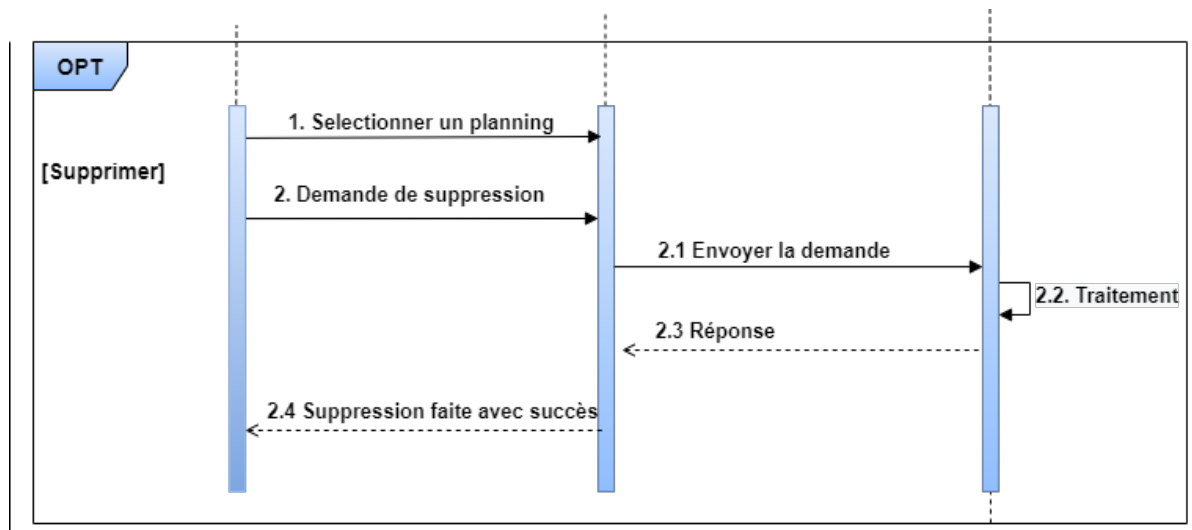


FIGURE 2.18 – Diagramme de séquence de cas d'utilisation «Gérer les plannings».

2.5 Diagramme de classe

Le diagramme de classes est considéré comme une fondation de la modélisation orientée objet. Il s'agit d'une vue statique, car on ne tient pas compte du facteur temporel dans le comportement du système. Le diagramme de classes modélise les concepts du domaine d'application ainsi que les concepts internes créés de toute pièce dans le cadre de l'implémentation d'une application. Chaque langage de Programmation orienté objet donne un moyen spécifique d'implémenter le paradigme objet (pointeurs ou pas, héritage multiple ou pas, etc.), mais le diagramme de classes permet de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier[14].

Les principaux éléments de cette vue statique sont les classes et leurs relations : association, généralisation et plusieurs types de dépendance, telles que la réalisation et l'utilisation. Le diagramme de classes relatif à l'application que nous allons réaliser est représenté dans la figure suivante :

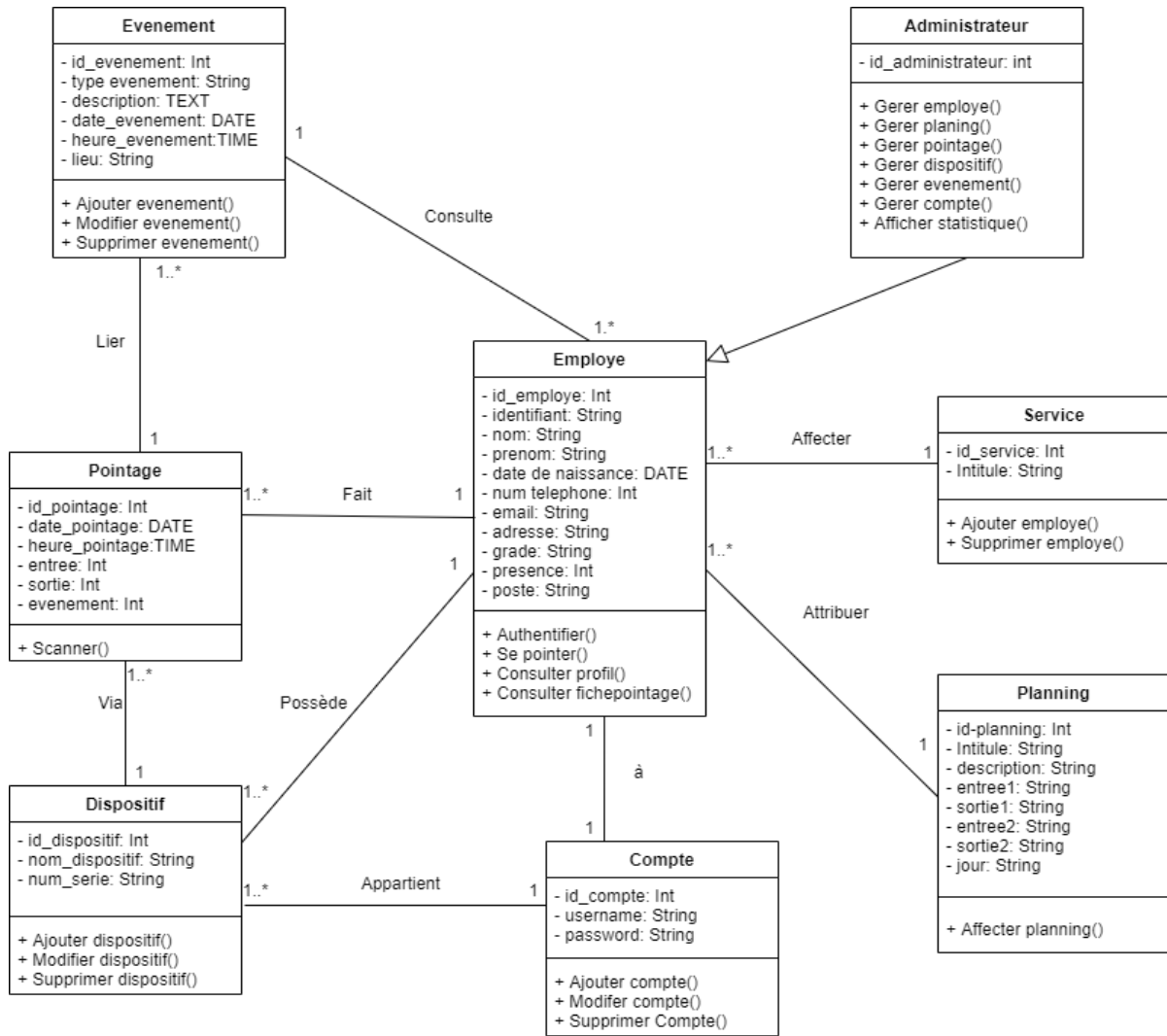


FIGURE 2.19 – Diagramme de classe.

2.6 Dictionnaire de données

Classe	Attribut	Signification	Type	Taille
administrateur	id_administrateur	Identifiant d'administrateur	INTEGER	10
employe	id_employe	Identifiant d'employé	INTEGER	10
	identifiant	Identifiant d'employé unique	VARCHAR	20
	nom	Nom d'employé	VARCHAR	20
	prenom	Prénom d'employé	VARCHAR	20
	date_naissance	Date de naissance d'employé	DATE	/
	num_telephone	Numéro de téléphone de l'employé	INTEGER	10
	email	Email d'employé	MAIL	/
	adresse	Adresse d'employé	VARCHAR	50
	grade	Grade d'employé	VARCHAR	20
	presence	Présence d'employé	INTEGER	10
	poste	Poste d'employé	VARCHAR	20
evenement	id_evenement	Identifiant d'évènement	INTEGER	20
	type_evenement	Type d'évènement	VARCHAR	20
	description	description d'évènement	TEXT	/
	date_evenement	Date d'évènement	DATE	/
	heure_evenement	Heure d'évènement	TIME	/
	lieu	Lieu d'évènement	VARCHAR	50
pointage	id_pointage	Identifiant de pointage	INTEGER	20
	date_pointage	Date de pointage	DATE	/
	heure_pointage	Heure de pointage	TIME	/
	entree	L'heure d'entrée de l'employé	INTEGER	2
	sortie	l'heure de la sortie de l'employé	INTEGER	2
	evenement	Evènement	INTEGER	2

dispositif	id_dispositif	Identifiant de dispositif	INTEGER	20
	nom_dispositif	Nom de dispositif	VARCHAR	40
	num_serie	Numéro de série du dispositif	VARCHAR	30
service	id_service	Identifiant du service	INTEGER	20
	intitule	Intitulé du service	VARCHAR	30
compte	id_compte	Identifiant du compte	INTEGER	10
	username	Nom d'utilisateur	VARCHAR	20
	password	mot de passe d'utilisateur	VARCHAR	10
planning	id_planning	Identifiant du palnning	INTEGER	20
	intitule	Intitule du palnning	VARCHAR	50
	description	Description du planning	TEXT	/
	entree1	L'entrée a l'entreprise le matin	TIME	4
	sortie1	La sortie de l'entreprise le matin	TIME	4
	entree2	L'entrée a l'entreprise l'après-midi	TIME	4
	sortie2	La sortie de l'enreprise l'après-midi	TIME	4
	jour	La journée	VARCHAR	10

TABLE 2.13 – Dictionnaire de données.

2.7 Le modèle relationnel

Le modèle relationnel a vu le jour en 1970 avec les travaux de Edgar Frank Codd. La principale publication est « A Relational Model for Large Shared Data Banks », Communications of the ACM, vol. 13, n° 6, 1970. Ce modèle de données est construit principalement autour des concepts suivants [15] :

- Relation : structure de données qui préfigure la table qui sera créée à l'aide du langage SQL ;
- Attribut : représentation d'une information atomique qui préfigure une colonne d'une table ;
- Un domaine de valeurs devrait être défini ainsi que d'éventuelles règles de validation (contraintes) ;

- Clé primaire : attribut(s) identifiant une relation qui préfigure(nt) la primary Key de la table ;
- Clé étrangère : attribut(s) qui référence(nt) une tierce relation, préfigure(nt) une foreign Key de la table.

Afin de pouvoir implémenter une base de données, il faut pouvoir traduire le modèle conceptuel en modèle logique. Cela signifie qu'il faut pouvoir convertir un modèle UML en modèle relationnel. Les modèles conceptuels sont suffisamment formels pour que ce passage soit systématisé dans la plupart des cas. En appliquant les règles de passage, nous obtenons les relations suivantes[16] :

- Règle 1 : présence de la cardinalité ($? .. 1$) d'un côté de l'association, chaque classe se transforme en une table, Chaque attribut de classe se transforme en un champ de table, l'identifiant de la classe qui est associée à la cardinalité ($? .. 1$) devient la clé étrangère de l'autre classe.
- Règle 2 : présence d'une généralisation. Créer une table pour chaque sous type, chaque table se compose des attributs génériques et d'attributs spécifiques .

Nous obtenons les relations suivantes :

- employe (id_employe, identifiant ,nom, prenom, date_naissance, num_telephone, email, adresse, grade, presence, poste, # id-evenement, #id_service, #id_planning, #id_compte) ;
- administrateur (id_administrateur,id_employe , nom, prenom, date_naissance, num_telephone, email, adresse, grade, presence, poste, # id-evenement, #id_service, #id_planning, , #id_compte) ;
- evenement (id_evenement, type_evenement, description, date_evenement, heure_evenement, lieu, #id_pointage) ;
- pointage (id_pointage, date_pointage, heure_pointage, entree, sortie , evenement, #id_dispositif, #id_employe) ;
- dispositif (id_dispositif, nom_dispositif, num_serie, #id_employe, #id_compte) ;
- service (id_service, intitule) ;
- compte (id_compte, username, password,#id_employe) ;
- planning (id_planning, intitule, description, entre1, sortie1, entre2, sortie2, jour).

Conclusion

Ce chapitre nous a permis en premier lieu d'exprimer et d'analyser les besoins permettant de décrire les fonctionnalités du système de manière globale. Aussi, grâce à l'identification des cas d'utilisation, nous avons défini les diagrammes de cas d'utilisation. En second lieu nous avons décrit les diagrammes de séquences des cas d'utilisation. Tout ceci nous permet d'élaborer le diagramme de classes de notre système, Enfin, pour avoir une vue plus structurée et implémenter une base de données nous avons traduit ce dernier diagramme en un modèle relationnel.

CHAPITRE 3

RÉALISATION

Introduction

Ce chapitre est consacré à la présentation de l'environnement de développement et des bibliothèques utilisées. Ensuite, nous expliquerons l'architecture globale du système ainsi que son principe de fonctionnement. La première partie sera réservée à la présentation de l'application mobile dans ses différents aspects en commençant par sa charte graphique et la hiérarchie de ces interfaces. La deuxième partie sera réservée à la présentation du serveur web. Enfin, nous aborderons brièvement le côté sécurité de l'application réalisée.

3.1 Environnement de développement

Un environnement de développement se réfère à une suite d'applications et d'outils que les développeurs installent sur leurs machines afin de développer leurs applications, gérer les fichiers sources, déboguer du code, et enfin tester leur travail avant de le lancer sur un environnement de test et/ou de production[17].

Un environnement de développement est souvent composé de certains des éléments suivants : un éditeur de code source, pile de développement, le test des applications et un gestionnaire de versions.

Dans cette section nous allons présenter l'environnement de développement utilisé pour développer notre projet.

3.1.1 Plateformes utilisées

— Github

GitHub offre un service d'hébergement qui permet aux développeurs de stocker et de partager, publiquement ou non, le code qu'ils créent. Elle fournit aussi plusieurs fonctionnalités de collaboration, telles que des wikis et des outils de gestion des tâches de base pour chaque projet [18].



— Discord

Discord est une plateforme gratuite qui offre un service de communication par chat vidéo, vocal et textuel, il est donc utile et nécessaire de l'utiliser pour une communication en continu surtout en temps de pandémie [19].



3.1.2 Logiciels utilisés

— Visual studio code

Visual Studio Code (VSCode) est un éditeur de code source, développé par Microsoft en tant que projet open source. Il s'agit de l'outil d'environnement de développement le plus populaire au monde selon un sondage auprès des développeurs réalisés par Stackoverflow [20]. En plus d'être un éditeur open source, il dispose d'un contrôle de gestion intégrée et d'un grand nombre d'extensions [21].



— Android Studio

Android Studio est l'environnement de développement intégré (IDE) officiel pour le développement d'applications Android, basé sur IntelliJ IDEA. En plus, du puissant éditeur de code et des outils de développement d'IntelliJ, Android Studio offre encore plus de fonctionnalités qui améliorent la productivité lors de la création d'applications Android [22].



— Draw.io

Conçu par Seibert Media, draw.io est un logiciel propriétaire permettant de créer des diagrammes et des graphiques. Le logiciel vous permet de choisir parmi une fonction de mise en page automatique ou de créer une mise en page personnalisée. Ils ont un grand choix de formes et des centaines d'éléments visuels pour rendre votre diagramme ou votre graphique unique. La fonction glisser-déposer facilite la création d'un diagramme ou d'un graphique attrayant [23].



3.1.3 Langages de programmation utilisés

Dans cette section, nous allons présenter les différents langages de programmation utilisés pour développer notre application mobile et notre serveur web.

— JAVA

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitations (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp) [24].



— PHP

PHP est un langage de scripts open source, conçu spécialement pour le développement d'applications web. Il peut être intégré facilement au HTML. Il s'exécute de côtés serveur qui permet la génération des pages HTML dynamiquement et permet aussi la connectivité étendue aux bases de données [25].



— HTML

C'est un langage de balisage principal du World Wide Web. À l'origine, HTML était principalement conçu comme un langage pour décrire sémantiquement des documents scientifiques. Sa conception générale lui a toutefois permis de s'adapter, au cours des années suivantes, pour décrire un certain nombre d'autres types de documents et même d'applications [26].



— CSS

Les feuilles de style en cascade sont un mécanisme simple pour ajouter du style (par exemple, polices, couleurs, espacement) aux documents Web. Les standards le définissant sont publiés par le World Wide Web Consortium (W3C) [27].



— JavaScript

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web.

Avec les technologies HTML et CSS, JavaScript est parfois considéré comme l'une des technologies cœur du World Wide Web [28].



JavaScript

— MySQL

MySQL est un système de gestion de bases de données relationnelles SQL open source développée et supportée par Oracle. Son approche relationnelle permet d'organiser les données dans des tableaux à deux dimensions appelés des relations ou tables [29].



3.2 Les librairies utilisées

— Bootstrap

C'est le Framework CSS le plus populaire au monde pour créer des sites responsifs et adaptés aux mobiles. Il dispose d'un système de grille réactif, de nombreux composants prédéfinis et de puissants plugins JavaScript [30].



— JQuery

JQuery est une bibliothèque JavaScript rapide, petite, et riche en fonctionnalités. Il simplifie considérablement le parcours et la l'animation grâce à une API facile à utiliser qui fonctionne sur une multitude de navigateurs [31].



3.3 Présentation de l'application mobile

Notre application mobile offre une solution de monitoring de présence par le biais d'un code QR qui est scanné par la caméra du smartphone de l'employé. Cette dernière sert essentiellement à identifier et à marquer la présence d'un employé au sein de son entreprise. Elle est également pour tout ce qui est consultation de profil, de planning et de fiche de pointage. Il est à noter que toutes les informations de l'employé sont transférées vers le serveur web.

3.3.1 Arborescence de l'application

L'arborescence de l'application mobile est représenté dans la figure 3.1 .

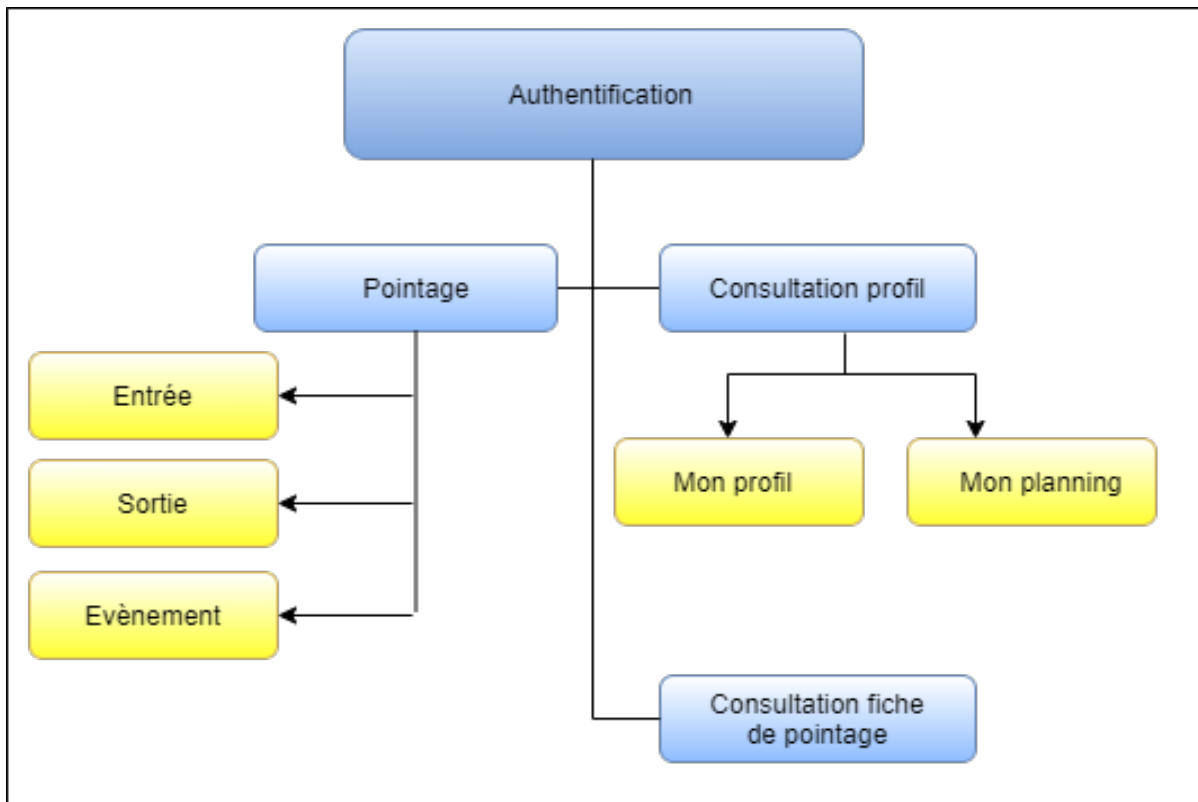


FIGURE 3.1 – Arborescence de l'application mobile.

3.3.2 Aperçu de notre application

Dans ce qui suit, nous allons présenter quelques interfaces de notre application .

— Interface d'authentification

La figure 3.2 représente l'interface d'authentification, c'est la première interface présentée à l'utilisateur au lancement de l'application mobile. Cette interface permettra à l'employé de s'authentifier auprès de l'application via son smartphone, afin d'accéder à ses privilèges.

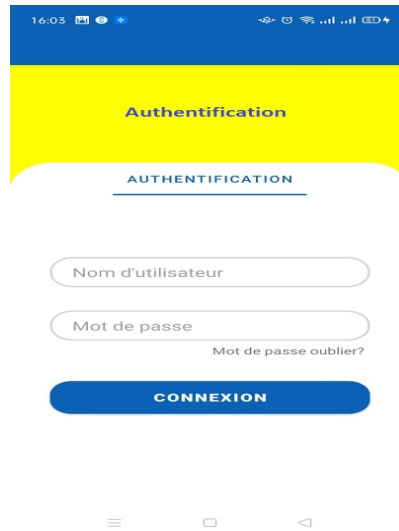


FIGURE 3.2 – Interface Authentification.

— Interface d'accueil

La figure 3.3 représente l'interface d'accueil qui offre une vue globale des différentes fonctionnalités préalablement conçues pour les employés à savoir pointage, consultation de profil et consultation fiche de pointage. Chaque fonctionnalité est affichée avec une icône correspondante.



FIGURE 3.3 – Interface Accueil.

— Interface pointage

la figure 3.4 représente l'interface pointage, l'employé pourra se présenter (entrée, sortie et évènement) dans l'entreprise

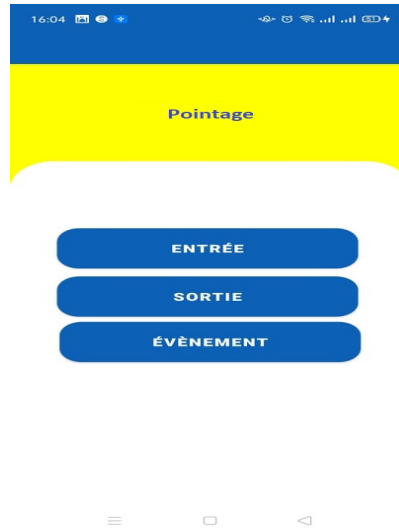


FIGURE 3.4 – Interface Pointage.

Après avoir choisi l'une des options précédentes l'employé devra scanner un QR code comme le montre la figure 3.5 suivante

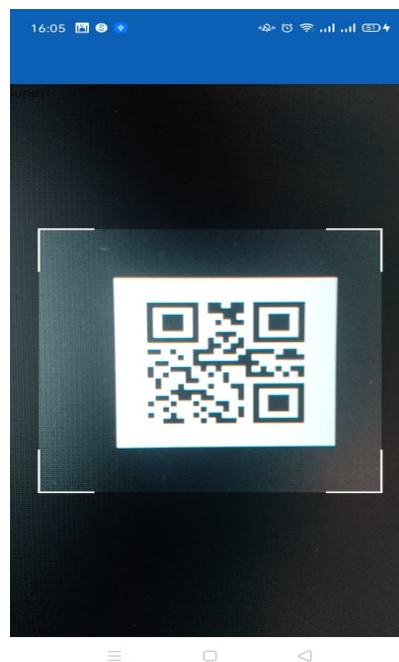


FIGURE 3.5 – Interface Scanner le QR code.

— Interface Consultation profil

Une fois l'employé accède à son compte avec succès il a la possibilité de consulter son profil afin de visualiser les informations personnelles et les coordonnées (figure 3.6) à savoir le nom, le prénom, la date de naissance, le genre, le grade, l'adresse, le téléphone personnel et son planning journalier (figure 3.7).



The screenshot shows a mobile application interface titled "Consultation de profil". At the top, there is a blue header with the time "16:07" and various status icons. Below the header is a yellow banner with the title "Consultation de profil". Underneath, there are two tabs: "MON PROFIL" (selected) and "MON PLANNING". The main content area is titled "Informations perssonnel" (note the typo). It contains several input fields for personal data: "Nom" (nom), "Prénom" (prenom), "Date de naissance" (date de naissance), "Genre" (genre), "Grade" (grade), "Adress" (adress), "Téléphone personnel" (téléphone personnel), "Téléphone professionnel" (téléphone professionnel), and "Email" (email). The fields are arranged in a grid-like fashion. At the bottom, there are three navigation icons: a hamburger menu, a square, and a back arrow.

FIGURE 3.6 – Interface Consulter mon profil



The screenshot shows the same mobile application interface, but with the "MON PLANNING" tab selected. The main content area is titled "Plannig du jour" (note the typo). It is divided into two sections: "matiné" and "Après-midi". Each section has two input fields: "Entrée 1" (entrée 1) and "Sortie 1" (sortie 1) for the morning, and "Entrée 2" (entrée 2) and "Sortie 2" (sortie 2) for the afternoon. The layout is consistent with the previous screenshot, including the blue header, yellow banner, and navigation icons at the bottom.

FIGURE 3.7 – Interface Consulter mon planning.

3.4 Présentation du serveur web

Notre serveur web héberge les informations des employés émises de l'application mobile, et il envoie également les informations relatives à son profil et à sa fiche de pointage. L'administrateur gère les employés à travers l'ajout la modification, la suppression et l'archivage. Il permet également à l'administrateur de gérer les plannings, les dispositifs, les événements, les comptes, consulter la fiche de pointage des employés et afficher les statistiques.

3.4.1 Arborescence du serveur web

L'arborescence du serveur web est représenté dans la figure 3.8 .

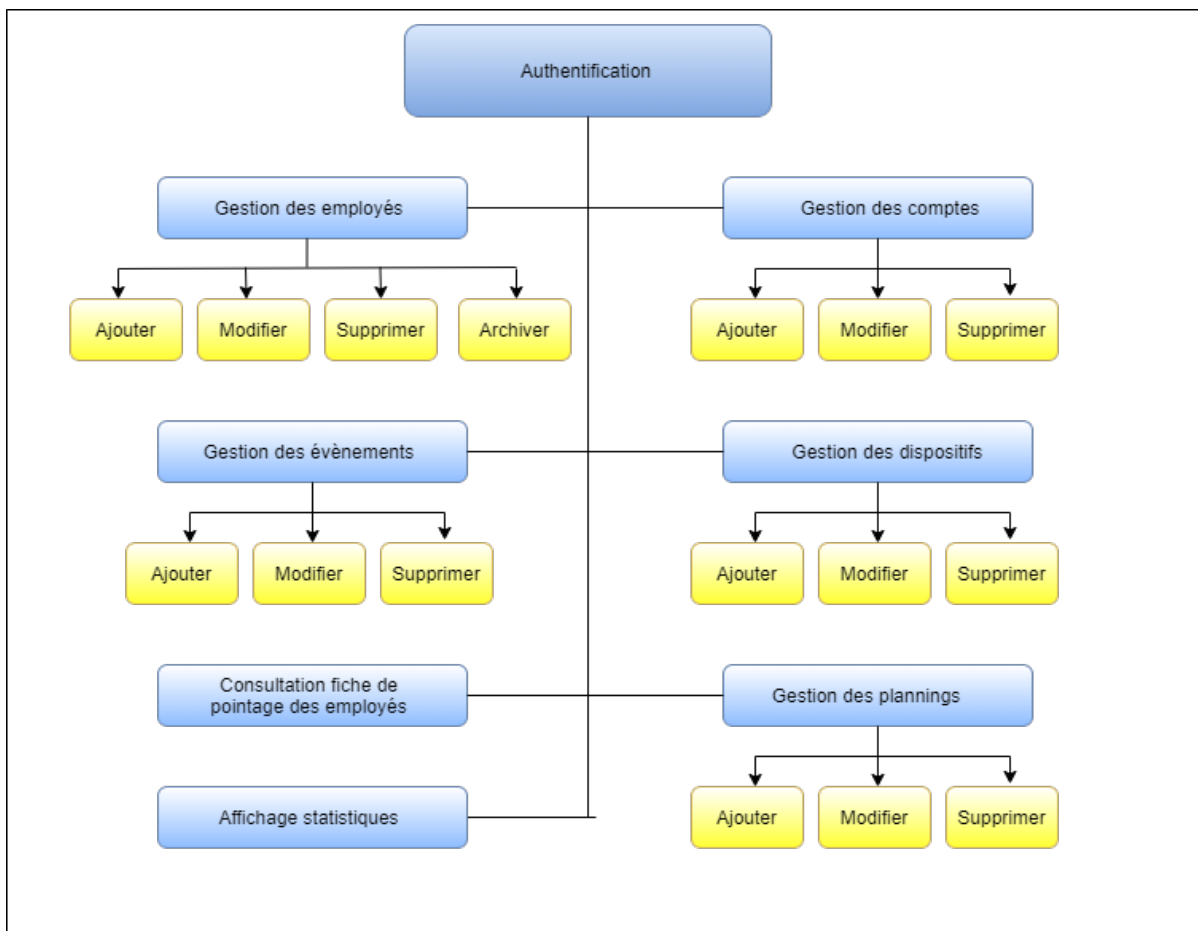


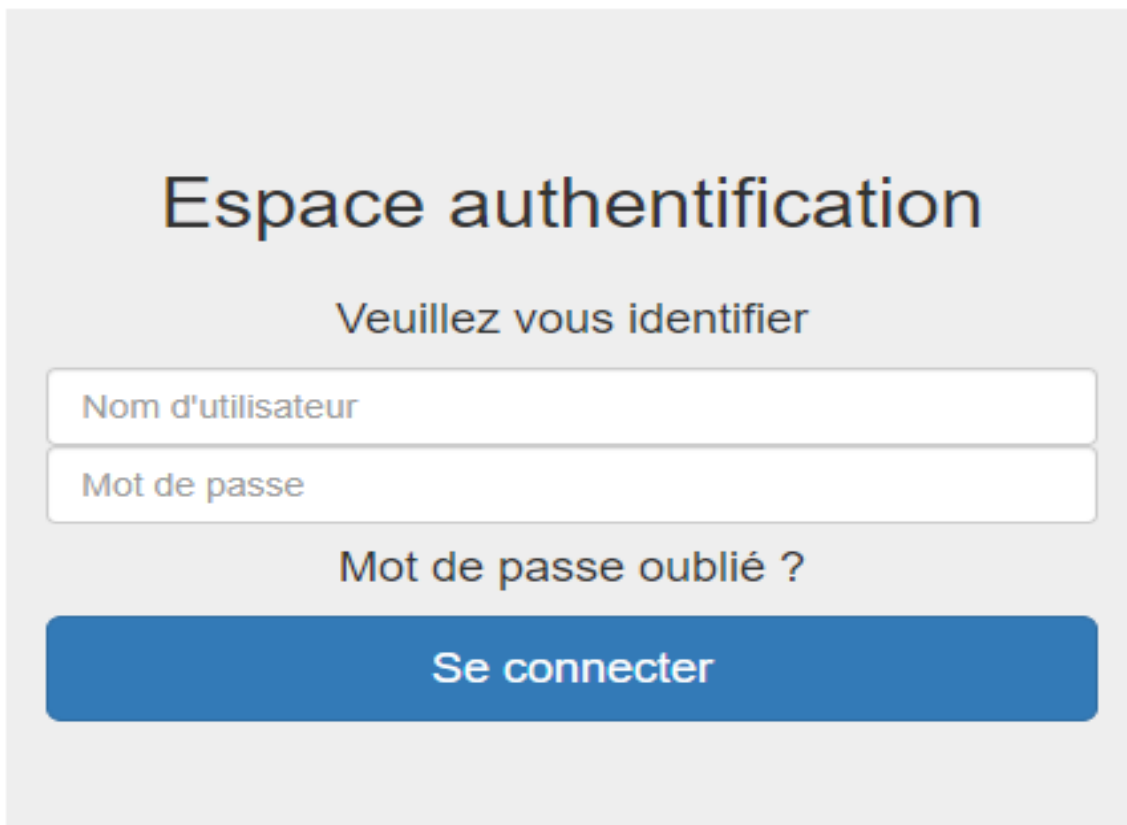
FIGURE 3.8 – Arborescence serveur web.

3.4.2 Aperçu des pages web

Dans ce qui suit, nous allons présenter quelques interfaces de notre serveur web.

— Page d'authentification

La figure 3.9 représente l'interface qui permet à l'administrateur de s'authentifier auprès de notre serveur web.



The image shows a web interface for authentication. At the top, the title 'Espace authentification' is displayed in a large, bold, black font. Below the title, the instruction 'Veuillez vous identifier' is centered. There are two input fields: the first is labeled 'Nom d'utilisateur' and the second is labeled 'Mot de passe'. Below these fields, there is a link that says 'Mot de passe oublié ?'. At the bottom of the form, there is a prominent blue button with the text 'Se connecter' in white.

FIGURE 3.9 – Page Authentification.

— Page Accueil

La figure 3.10 montre la page d'accueil représentant l'interface principale de notre serveur web. Par le biais de cette page, l'administrateur pourra accéder aux autres interfaces (gestion des comptes, gestion des employés, gestion des dispositifs, affichage statistique, gestion des événements, gestion des plannings et consultation fiche de pointage employé) pour gérer son entreprise selon ses besoins.

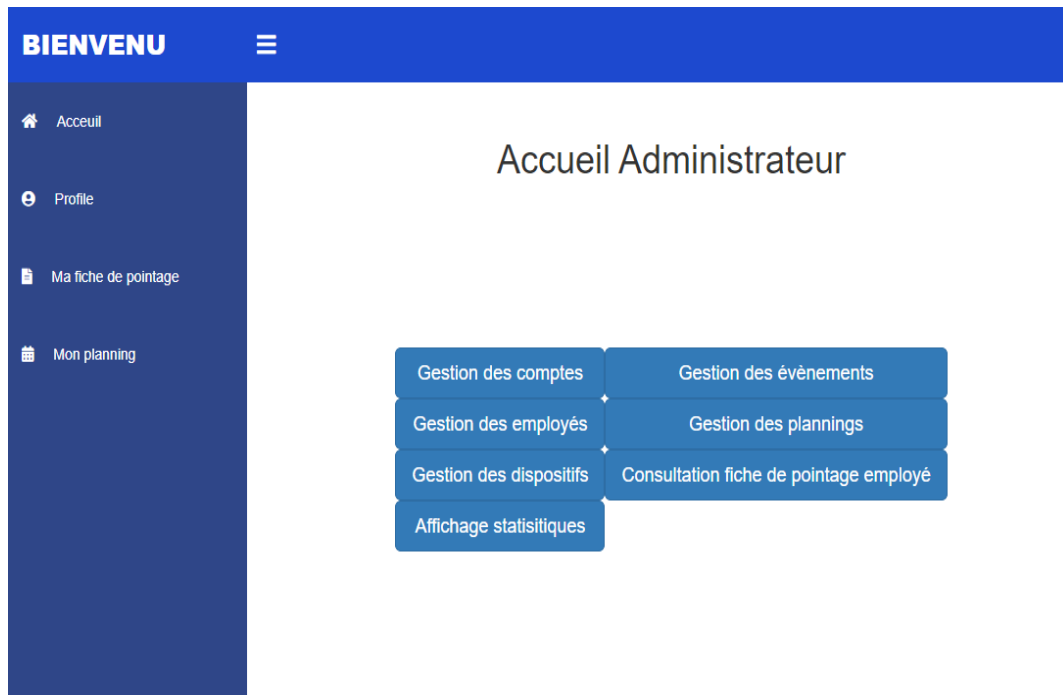
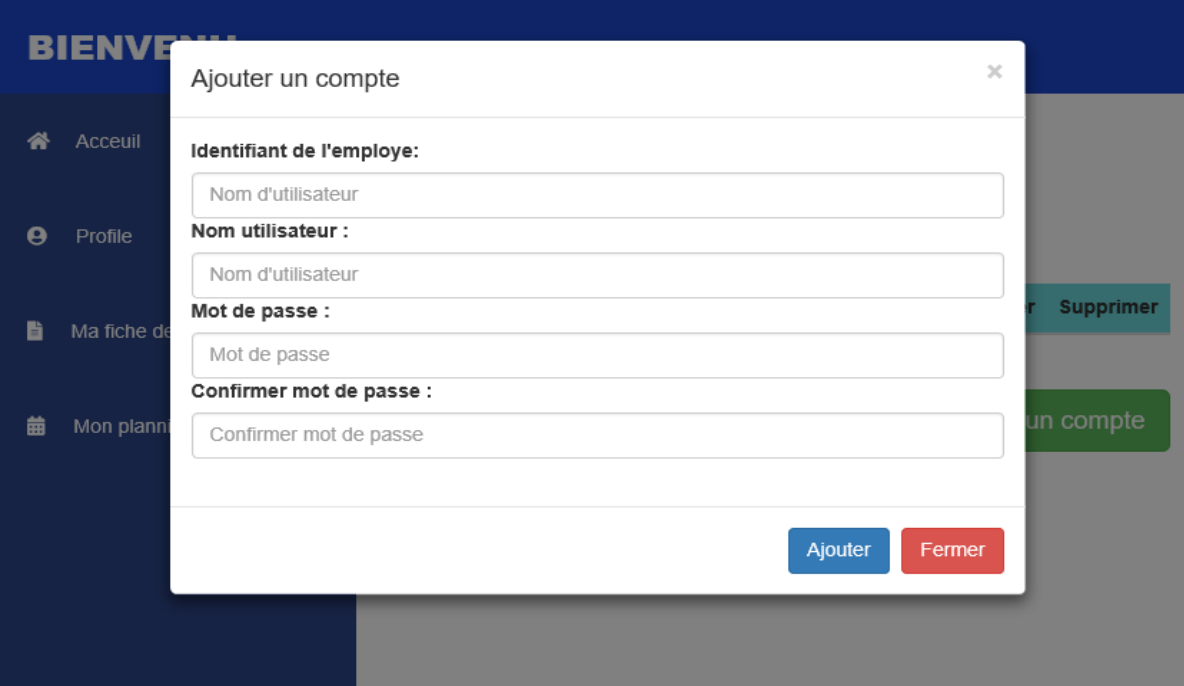


FIGURE 3.10 – Page Accueil administrateur.

— Page Ajouter compte

Une figure 3.11 représente la page d'ajout d'un compte, l'administrateur pourra à travers cette page qui se présente sous la forme d'un formulaire d'ajout d'un compte. Il doit renseigner les informations les plus sensibles pour le bon fonctionnement du système.



The image shows a modal dialog box titled "Ajouter un compte" (Add an account) overlaid on a web application interface. The background interface includes a dark blue header with the word "BIENVENUE" (Welcome) and a sidebar with navigation items: "Accueil" (Home), "Profil" (Profile), "Ma fiche de" (My card), and "Mon planning" (My calendar). The modal dialog has a white background and a close button (X) in the top right corner. It contains the following fields and labels:

- Identifiant de l'employe:** A text input field with the placeholder "Nom d'utilisateur".
- Nom utilisateur :** A text input field with the placeholder "Nom d'utilisateur".
- Mot de passe :** A text input field with the placeholder "Mot de passe".
- Confirmer mot de passe :** A text input field with the placeholder "Confirmer mot de passe".

At the bottom right of the modal, there are two buttons: a blue "Ajouter" (Add) button and a red "Fermer" (Close) button.

FIGURE 3.11 – Page d'ajout d'un compte.

— Page ajouter planning

La figure 3.12 représente le travail effectué par l'administrateur lui permettant de créer des nouveaux plannings.

Créé un nouveau planning

Titre* :

Description :

Dimanche :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Lundi :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Mardi :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Mercredi :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Jeudi :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Vendredi :

Entre 1	Sortie 1	Entre 2	Sortie 2
---------	----------	---------	----------

Samedi :

FIGURE 3.12 – Page d'ajouter d'un planning.

Conclusion

Dans ce chapitre nous avons énoncé les différents logiciels, plateformes, librairies utilisées pour la réalisation de notre application mobile ainsi que le serveur web. Dans un premier temps, nous avons présenté notre application mobile avec son arborescence suivie de quelques interfaces. Ensuite, nous avons donné un aperçu de notre serveur web et quelques pages web qui apparaîtront à l'administrateur lui permettant de gérer efficacement l'emploi du temps des employés.

CONCLUSION GÉNÉRALE

Ce travail a été réalisé dans le cadre de notre projet de fin de cycle Master option Génie logiciel. Il consiste à la mise en œuvre d'une solution de monitoring de présence sous Android constitué d'une application mobile et d'un serveur web. Ce système peut être destiné aux entreprises, administrations et universités dans un but de réduire le coût et le taux d'erreurs, et ce, par un système de gestion présence automatisée.

Durant ce projet, nous avons pu mettre en œuvre les connaissances théoriques et pratiques acquises durant notre cursus. Nous avons réalisé ce travail en combinant conception et programmation pour répondre aux besoins de l'entreprise Cevital en matière de monitoring de présence.

Nous avons réalisé une application mobile liée à un serveur web permettant de faciliter la gestion du personnel de l'entreprise d'accueil. En effet, toute absence, retards, gérés automatiquement à l'aide des informations transmises de l'application vers le serveur.

Grâce à un processus de développement, nous avons pu effectuer une spécification et une analyse des besoins des utilisateurs, et dégager les principaux acteurs et cas d'utilisation pour réaliser les différents diagrammes, à savoir les diagrammes de cas d'utilisation et de séquence.

Ensuite, la phase de conception une fois terminée, nous a permis de générer le diagramme de classes ainsi que le modèle relationnel pour commencer la réalisation de l'application, Enfin, nous avons abordé la dernière phase aisément grâce au travail effectué

dans les phases précédentes. Nous avons pu ainsi utiliser différents outils et plateformes (Java, Android studio, Bootstrap, PHP, etc.) pour implémenter notre solution que nous avons dotée d'une identité graphique reflétant le domaine d'activité de celle-ci.

Malgré tout le travail fourni, nous sommes conscients que plusieurs aspects de notre système peuvent et doivent être améliorés. Nous pouvons citer à titre d'exemple l'optimisation du code pour permettre une meilleure exploitation des ressources matérielles. En guise de perspectives, nous espérons enrichir notre solution avec d'autres fonctionnalités telles que la génération des bulletins de salaire en fonction des heures de travail, et également la génération des attestations de travail.

BIBLIOGRAPHIE

- [1] Cevital, «document interne de l'entreprise,» 2021.
- [2] P. Allard, CODES QR Un gadget ou un nouvel outil?. ABD-BVD, Cahiers de la documentation., 2011.
- [3] T. J. Soon., QR Code. synthesis journal, section Three., 2008.
- [4] K. E. M. Mary Shanthi Rani, DATA SECURITY THROUGH QR CODE ENCRYPTION AND STEGANOGRAPHY., Advanced Computing : An International Journal (ACIJ), 2016.
- [5] M. h. Iris Uitz, “ Der QR-Code – aktuelle Entwicklungen und Anwendungsbereiche, Springer Science+Business Media., 2012.
- [6] H. J. G. Kinjal H. Pandya, A Survey on QR Codes : in context of Research and Application. International Journal of Emerging Technology and Advanced Engineering, (ISSN 2250-2459, ISO 9001 : 2008 Certified Journal)., 2014.
- [7] <https://www.wearecom.fr/dictionnaire/application-mobile/>. [Consulté le 15/04/2021].
- [8] <https://www.futura-sciences.com/tech/definitions/smartphone-android-15017>. [Consulté le 15/04/2021].

- [9] <https://sabricole.developpez.com/uml/tutoriel/unifiedProcess>. [Consulté le 16/04/2021].
- [10] <https://ult.bi/fr/chapitre-ii-presentation-du-langage-de-modelisation-uml>. [Consulté le 16/04/2021].
- [11] [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique)). [Consulté le 16/04/2021].
- [12] https://www.memoireonline.com/06/12/5947/m_Conception-dun-outil-dadministration-reseaux4.html. [Consulté le 19/05/2021].
- [13] <https://docwiki.embarcadero.com>. [Consulté le 06/06/2021].
- [14] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-classes>. [Consulté le 25/06/2021].
- [15] F. B. Christian Soutou, UML 2 pour les bases de données, 2e édition,, Paris, France. : EYROLLES.
- [16] <https://slidetodoc.com/transformation-du-diagramme-de-classe-en-modle-relationnel>. [Consulté le 15/09/2021].
- [17] <https://www.blog-nouvelles-technologies.fr/3461/quest-ce-quun-environnement-de-developpement/>. [Consulté le 21/07/2021].
- [18] <https://github.com>. [Consulté le 02/08/2021].
- [19] <https://discord.com>. [Consulté le 15/08/2021].
- [20] Résultats du sondage auprès des développeurs 2019, insights.stackoverflow.com/survey/2019. [Consulté le 15/07/2021].
- [21] <https://code.visualstudio.com>. [Consulté le 15/07/2021].
- [22] <https://developer.android.com/studio>. [Consulté le 16/07/2021].
- [23] <https://www.computerhope.com/jargon/d/drawio.htm>. Consulté le 12/05/2021].

- [24] <https://www.futura-sciences.com/tech/definitions/internet-java-485>. [Consulté le 04/07/2021].
- [25] Robin nixon, Développer un site web en php, mysql, javascript jquery, css3 et html5, 4ème édition, traduction de William Piette, Edition O'Reilly, 2016.
- [26] <https://www.w3.org/TR/2017/REChtml52-20171214/introduction.html>. [Consulté le 02/08/2021].
- [27] <https://www.w3.org/Style/CSS>. [Consulté le 02/08/2021].
- [28] <https://developer.mozilla.org/fr/docs/Web/JavaScript>. [Consulté le 02/08/2021].
- [29] <https://www.mysql.com/fr/>. [Consulté le 02/09/2021].
- [30] <https://getbootstrap.com>. [Consulté le 03/08/2021].
- [31] <https://jquery.com>. [Consulté le 03/08/2021].

Résumé

Ce document a été rédigé en vue de l'obtention du diplôme de Master en Génie Logiciel. La problématique traitée est la gestion de pointage des employés au sein d'une entreprise.

Après une analyse de la problématique, une solution de monitoring de présence a été proposée. Cette dernière est composée d'une application mobile et d'un serveur web, cette alternative permet une meilleure gestion par service.

Pour concrétiser cela, nous avons eu recours à un processus de développement UP et au langage de modélisation UML pour les phases analyse et conception. La phase réalisation s'est caractérisée par l'utilisation du langage PHP pour l'implémentation du serveur web. Quant à l'application mobile le langage JAVA, le Framework CodeScanner de SwiftUI, ainsi que d'autres librairies ont été utilisés.

Mot clé : présence, code QR, pointage, UP, UML.

Abstract

This document has been written with a view for obtaining the Master's degree in Software Engineering. The issue addressed is the management of worker check-ins within a company.

After the analysis of the problem, a presence monitoring solution was proposed. The latter is composed of a mobile application and a web server, this alternative allows better management by department.

To make this happen, we used a UP development process and the UML modeling language for the analysis and design phases. The realization phase was characterized by the use of the PHP language for the implementation of the web server. As for the mobile application, the JAVA language, SwiftUI's CodeScanner Framework, as well as other library were used.

Keywords : presence, QR code, pointing, UP, UML.