



المعهد العالي للإعلامية وتقنيات الإتصال بحمام سوسة  
INSTITUT SUPÉRIEUR D'INFORMATIQUE ET DES TECHNIQUES  
DE COMMUNICATION –HAMMAM SOUSSE

Département Réseaux et Multimédia  
MEMOIRE DE STAGE FIN D'ETUDES

Présenté en vue de l'obtention du diplôme de  
Licence Appliquée Technologies du Multimédia  
et du web

# Système de Pointage Basé sur la Reconnaissance Faciale

Réalisé par :  
Sana DALLEL

Société d'accueil  
Astrolab

Encadré(e)(s) par :  
Mme. Hédia JEGHAM  
Mr. Mohamed DARDOURI

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR  
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DE SOUSSE

المعهد العالي للإعلامية و تقنيات الاتصال بحمام سوسة

INSTITUT SUPERIEUR D'INFORMATIQUE  
ET DES TECHNIQUES DE COMMUNICATION – HAMMAM SOUSSE

Département Réseaux et Multimédia

MEMOIRE DE STAGE FIN D'ETUDES

**Présenté en vue de l'obtention du diplôme de Licence  
Appliquée  
Technologies du multimédia et du Web**

**Réalisé par :**  
Sana Dallel

**Encadrant : Mohamed Dardouri      Date :.....Signature :.....**

**Superviseur : Hédia Jegham      Date :.....Signature :.....**

---

**Année Universitaire 2018 – 2019**

## Résumé

L'image d'un visage dans un passeport est la seule donnée à utiliser pour l'authentification personnelle dans le monde entier, quelle que soit sa nationalité. En outre, la reconnaissance d'image sur le visage offre un avantage unique par rapport aux autres technologies d'authentification personnelle, car elle permet d'authentifier des personnes à distance. Ce rapport présente une technologie de reconnaissance faciale en temps réel à utiliser en un système de pointage.

*Mots clés:* Apprentissage automatique, Reconnaissance des visages, Analyse en temps réel.

...

À mes parents *Fathi et Samia*,  
la raison de ce que je deviens aujourd'hui,  
Merci pour votre tendresse et vos soins continus.

À mes frères *Ahmed, Mohamed et Iyéd*  
que le bon Dieu vous procure la bonne santé, la joie, la  
luminescence et la réussite dans votre vie.

À mes amis *Mariam et spécialement Med yassine*,  
Je vous en suis très reconnaissant.

Je vous souhaite tout le bonheur et le succès dans votre vie  
familiale et professionnelle.

...

## Remerciements

*Avant de présenter mon travail, je souhaite exprimer ma grande gratitude aux personnes qui m'ont soutenu, de près ou de loin, pour montrer leur soutien.*

*Je remercie Mr Anis TOUIBI, le président d'Astrolab, pour m'avoir prodigué l'honneur de travailler au sein de son équipe.*

*Je remercie Mr Mohamed DARDOURI, mon encadrant à Astrolab pour son assistance régulière et son aide par son expérience et ses compétences.*

*J'exprime ma parfaite gratitude et mes remerciements à Madame Hédia JEGHAM, mon superviseur à l'ISITCom, pour le temps qu'elle a bien voulu consacrer à la supervision et au suivi de ce travail, aux conseils qu'elle m'a donnés après ses lectures méticuleuses et aux réunions qui ont ponctué les différentes étapes de la rédaction de ce rapport. Je la remercie beaucoup pour ses efforts, sa disponibilité et surtout pour ses conseils, qui ont grandement contribué à valoriser ce travail.*

*J'exprime mes sincères remerciements à toute l'équipe d'Astrolab.*

*Que les membres du jury trouvent, ici, l'expression de mes remerciements pour l'honneur qu'ils me rendent en acceptant de juger ce travail.*

# Table des Matières

INTRODUCTION GÉNÉRALE	1
CHAPITRE 1 LANCEMENT DU PROJET	2
Introduction du chapitre . . . . .	3
1.1 Présentation de l'entreprise . . . . .	3
1.1.1 Présentation d'Astrolab . . . . .	3
1.1.2 Secteur d'activités . . . . .	3
1.2 Cahier de charge du projet . . . . .	4
1.2.1 Besoins fonctionnels . . . . .	4
1.2.2 Besoins non fonctionnels . . . . .	5
1.2.3 Identification des acteurs . . . . .	6
1.2.4 Diagramme de contexte . . . . .	6
1.3 Méthode de développement . . . . .	7
1.3.1 Méthode adoptée . . . . .	7
1.3.2 Constitution de l'équipe . . . . .	7
1.4 Étude comparative . . . . .	8
1.5 Architecture globale du logiciel . . . . .	9
1.6 Product Backlog de Pointini . . . . .	10
1.7 Diagramme des cas d'utilisation global . . . . .	11
1.8 Diagramme de Gantt . . . . .	12
1.9 Établissement de la Definition of Done . . . . .	13
Conclusion . . . . .	14
CHAPITRE 2 SPRINT1 : RECONNAISSANCE & APPRENTISSAGE	15
Introduction du chapitre . . . . .	16
2.1 Objectif du Sprint 1 . . . . .	16
2.2 Séance du Sprint planning 1 . . . . .	16
2.3 Sprint backlog 1 estimé . . . . .	17
2.4 Démarrage du Sprint . . . . .	17
2.4.1 Analyse . . . . .	17
2.4.2 Conception . . . . .	20
2.4.3 Codage . . . . .	22
2.4.4 Tests . . . . .	24
Conclusion . . . . .	24

CHAPITRE 3 SPRINT2: HABILLAGE DE POINTINI	25
Introduction du chapitre . . . . .	26
3.1 Objectif du sprint . . . . .	26
3.2 Sprint planning . . . . .	26
3.3 Sprint Backlog 2 estimé . . . . .	27
3.4 Démarrage du sprint . . . . .	28
3.4.1 Analyse . . . . .	28
3.4.2 Conception . . . . .	32
3.4.3 Réalisation . . . . .	33
3.4.4 Tests . . . . .	37
Conclusion . . . . .	39
CHAPITRE 4 STATISTIQUES & FACETTE MOBILE	40
Introduction du chapitre . . . . .	41
4.1 Sprint 3: Offre des statistiques de Pointini . . . . .	41
4.1.1 Sprint backlog 3 estimé . . . . .	41
4.1.2 Analyse . . . . .	42
4.1.3 Conception . . . . .	45
4.1.4 Réalisation . . . . .	46
4.1.5 Tests . . . . .	48
4.2 Sprint 4: Pointini facette mobile . . . . .	49
4.2.1 Sprint backlog 4 estimé . . . . .	49
4.2.2 Analyse . . . . .	50
4.2.3 Réalisation . . . . .	51
4.2.4 Tests . . . . .	53
4.3 Déploiement de Pointini en environnement d'exploitation . . . . .	53
Conclusion . . . . .	54
CONCLUSION ET PERSPECTIVE	55
GLOSSAIRE	56
BIBLIOGRAPHY	57

## Liste des Figures

1.1	Logo Astrolab . . . . .	4
1.2	Diagramme de contexte . . . . .	7
1.3	Exemples de systèmes de pointages . . . . .	9
1.4	Architecture globale du logiciel . . . . .	10
1.5	Diagramme des cas d'utilisation . . . . .	12
1.6	Diagramme de Gantt . . . . .	13
2.1	Sprint planning 1 . . . . .	17
2.2	Logo de Dlib . . . . .	18
2.3	Logo OPenCV . . . . .	19
2.4	Positions de visages possible . . . . .	19
2.5	Diagramme de séquence : Marque présence . . . . .	21
2.6	Table employé . . . . .	22
2.7	Code capture et détection de visages . . . . .	22
2.8	Code reconnaissance faciale . . . . .	23
2.9	Capture: visage détecté . . . . .	24
3.1	Sprint planning 2 . . . . .	26
3.2	Use case: Capturer image employé . . . . .	28
3.3	Use case: gérer profil employé . . . . .	29
3.4	Use case: Modifier profil admin . . . . .	30
3.5	Diagramme de séquence : Authentification . . . . .	32
3.6	Diagramme de classe de sprint 2 . . . . .	33
3.7	Interface 1 : Consulter la liste des employés . . . . .	34
3.8	Interface 2.1 : Ajouter un employé . . . . .	34
3.9	Interface 2.2 : Ajouter un employé . . . . .	35
3.10	Interface 2.3 : Capturer des images . . . . .	35
3.11	Interface 3.1 : Profile employé . . . . .	36
3.12	Interface 3.2 : Consulter calendrier d'un employé . . . . .	36
3.13	Interface 4 : Suivre caméras . . . . .	37
3.14	Interface 1 : Tester login . . . . .	37
3.15	Interface 2 : Tester modification profil admin . . . . .	38
3.16	Interface 3 : Autorisation d'ouvrir la caméra . . . . .	38
3.17	Interface 4 : Tester ajout des images . . . . .	39
4.1	Use case: Consulter statistiques . . . . .	42
4.2	Use case: Consulter états de présences . . . . .	43



4.3	Use case: Calcul salaires . . . . .	44
4.4	Diagramme des classes globale . . . . .	45
4.5	Interface 1.1 : Statistique . . . . .	46
4.6	Interface 1.2 : Statistique . . . . .	46
4.7	Interface 1.3 : Statistique . . . . .	47
4.8	Interface 2 : Calcule salaire . . . . .	47
4.9	Interface 3 : État présence . . . . .	48
4.10	Interface 1 : Salaire calculé . . . . .	48
4.11	Interface 2 : Fiche salaire . . . . .	49
4.12	Diagramme des cas d'utilisation pour les employés . . . . .	50
4.13	Interface connexion . . . . .	51
4.14	Interface paramètres . . . . .	52
4.15	Interface de modification mot de passe . . . . .	52
4.16	Interface de profil d'un employé . . . . .	53
4.17	Interface du calendrier d'un employé . . . . .	53
4.18	Diagramme de déploiement . . . . .	54

## Liste des Tableaux

1.1	Tableau des acteurs . . . . .	6
1.2	Product Backlog . . . . .	11
2.1	Sprint Backlog . . . . .	17
3.1	Sprint Backlog de l'habillage du Pointini . . . . .	27
3.2	Description textuelle:«Capturer images» . . . . .	28
3.3	Description textuelle:«Ajouter employé» . . . . .	29
3.4	Description textuelle:«Modifier profil admin» . . . . .	31
3.5	Description textuelle:«Suivre caméras» . . . . .	31
4.1	Sprint Backlog de sprint 3 . . . . .	41
4.2	Description textuelle:«Consulter statistiques» . . . . .	42
4.3	Description textuelle:«Calcul salaires» . . . . .	44
4.4	Sprint Backlog de sprint 4 . . . . .	49
4.5	Description textuelle:«Modifier mot de passe» . . . . .	50

# Introduction Générale

De nos jours, les domaines d'application des algorithmes d'apprentissage automatique n'ont pas de limite en raison du focus porté par la recherche scientifique et de son grand impact de confort dans notre vie quotidienne. Fondamentalement, avec l'Intelligence Artificielle (IA), l'ordinateur est devenu capable de faire des tâches impossibles dans le passé. La force et performances de traitement de l'ordinateur mariée à Intelligence Artificielle a contribué à l'émergence d'une nouvelle intelligence qui est pointue et qui dépasse de loin celle de l'être humain.

Si par exemple une société recrute un ouvrier portier rien que pour reconnaître ses employés et surveiller leurs heures d'entrées et sorties, cet être humain pourrait manquer à la surveillance par absence ou par distraction. Il pourrait ne pas reconnaître le visage d'un ancien employé ou bien même fausser des horaires d'entrées et sorties. Un objet rendu intelligent et entraîné à la reconnaissance des visages des employés d'une entreprise ne commettra que de rarissimes défaillances !

C' est de cet aspect que traite notre sujet : Développer une application qui permet d'apprendre à une caméra des visages d'employés d'une société afin de noter leurs horaires d'entrées et sorties à la manière d'une badgeuse.

Au cours de notre stage de quatre mois, il nous a été confié donc de concevoir et développer une application web et mobile dédiée au chef du personnel permettant d'effectuer un pointage basé sur la reconnaissance faciale. Nous avons baptisé cette application « Pointini ».

Il est convient de signaler que ce projet a été dirigé selon le cadre de processus Scrum et qu'il a nécessité une recherche théorique dans le domaine des avancées scientifiques et techniques de la reconnaissance faciale.

Le rapport de ce PFE a été structuré ainsi : **Le premier chapitre** sera consacré à la présentation de l'organisme d'accueil Astrolab et pour le cahier des charges de notre application. Il sera clôturé par la "méthode de développement" adopté et la description de l'architecture du système.

**Les autres chapitres** expliciteront l'effort de travail investi dans les sprints décidés pour le projet.

À ce titre, chaque sprint faisant l'objet d'un chapitre, exposera les user stories qui lui ont été affectés à partir du Product Backlog, il présentera aussi le dossier conceptuel et des captures écran de l'incrément potentiellement livrable.

Il importe aussi d'indiquer que dans la rédaction de ce PFE nous avons préféré garder les termes usuels anglais de Scrum et de toute autre technique utilisée pour bien garder le sens voulu à l'origine.

Nous clôturerons ce rapport par une **conclusion générale** résumant et évaluant notre travail puis lançant des perspectives à ce projet.

# Chapitre 1

## Lancement du projet

Introduction du chapitre . . . . .	3
1.1 Présentation de l'entreprise . . . . .	3
1.1.1 Présentation d'Astrolab . . . . .	3
1.1.2 Secteur d'activités . . . . .	3
1.2 Cahier de charge du projet . . . . .	4
1.2.1 Besoins fonctionnels . . . . .	4
1.2.2 Besoins non fonctionnels . . . . .	5
1.2.3 Identification des acteurs . . . . .	6
1.2.4 Diagramme de contexte . . . . .	6
1.3 Méthode de développement . . . . .	7
1.3.1 Méthode adoptée . . . . .	7
1.3.2 Constitution de l'équipe . . . . .	7
1.4 Étude comparative . . . . .	8
1.5 Architecture globale du logiciel . . . . .	9
1.6 Product Backlog de Pointini . . . . .	10
1.7 Diagramme des cas d'utilisation global . . . . .	11
1.8 Diagramme de Gantt . . . . .	12
1.9 Établissement de la Definition of Done . . . . .	13
Conclusion . . . . .	14

# Introduction

Dans ce chapitre, Il nous commençons par traiter le contexte général du projet. Dans un premier temps, nous présenterons l'organisme d'accueil ainsi que ses domaines d'activités. Ensuite, l e cahier des charges du sujet, suivi d'une étude comparative de l'existant sur le marché suivront. L'architecture logicielle préconisé et le Product Backlog priorisé seront au suivant rang. Enfin, sera le tour de la planification des sprints pour la réalisation de notre application.

## 1.1 Présentation de l'entreprise

### 1.1.1 Présentation d'Astrolab

Astrolab est une agence digitale fondée en 2015 par Anis TOUIBI. L'agence est spécialisée dans l'accompagnement des porteurs d'idées de projets numériques en fournissant une assistance complète et une variété de services.

L'agence s'engage à renforcer les exigences particulières des ces clients afin de fournir des solutions personnalisées et performantes. Avec une approche créative et innovante, centrée sur les besoins et les objectifs des publics cibles, les services de la société sont conçus pour aider ses clients et les guider à travers toutes les étapes tout en externalisant leur projet pour obtenir des résultats et des produits exceptionnels. Grâce à une approche qualitative orientée vers la satisfaction des clients et son expertise dans le domaine digital la société offre aujourd'hui ses services pour des clients en Asie et en Europe. Fort de solutions technologiques remarquables, Astrolab fournis des services dans les domaines du développement Web et mobile, de la conception graphique, de la conception UI / UX et du marketing digitale.

### 1.1.2 Secteur d'activités

**Le web et les réseaux sociaux :** Avec une équipe réactive Astrolab guide et aide les startups digitales en démarrage à transformer leurs idées en un produit qui dépasse les attentes. Grâce à la maîtrise des méthodes de développement agile et les dernières technologies, la société offre la mise en œuvre des produits digitales conformément aux objectifs commerciaux.

**Développement Web et Mobile :** Une dizaine d'ingénieurs et de spécialistes compétents définissent, conçoivent et développent des applications personnalisées et des produits numériques utilisant des technologies de pointe, tous animés par la passion, l'intelligence et le dévouement.

**Conception graphique et interface utilisateur / UX:** Astrolab travaille avec ces clients sur des expériences novatrices, axées sur l'utilisateur, basées sur des outils intelligents et diverses ressources, tout en se concentrant sur les besoins des utilisateurs et en résolvant des problèmes complexes pour répondre aux objectifs et fournir des solutions numériques intelligentes [2].



Figure 1.1: Logo Astrolab

Le développement de l'application Pointini qui nous a été confié relève des deux derniers pistes d'activités ci-dessus d'Atrolab.

## 1.2 Cahier de charge du projet

### 1.2.1 Besoins fonctionnels

L'application Pointini doit être capable d'apprendre des visages inconnus par le biais d'une caméra et de les stocker dans sa base pour pouvoir les reconnaître à tout moment. A la reconnaissance d'un visage d'un employé elle doit tout de suite l'associer à son identité personnelle et professionnelle. Elle doit aussi comptabiliser l'heure de son passage à l'occasion d'une entrée ou d'une sortie. Les heures de travail passées par un employé dans l'entreprise peuvent faire l'objet d'heures supplémentaires qui doivent pour certains grades être payées en plus du salaire officiel. Les heures de travail non effectuées et non justifiées doivent aussi être comptabilisées. Un chef de personnel ou directeur de ressources humaines que nous considérons administrateur doit pouvoir piloter ces états grâce à un tableau de bord via une interface web. De même qu'un employé doit pouvoir suivre ses états d'absences ou d'heures supplémentaires via une interface mobile. En somme l'application Pointini sera sollicitée par un administrateur et un employé.

#### Concernant l'administration :

- S'authentifier: L'administrateur peut accéder à son compte avec une adresse mail et mot de passe spécifique pour lui.
- Gérer les comptes des employés :
  - Ajouter un employé: l'administrateur aura la possibilité d'ajouter un nouvel employé. Si un employé est absent le jour où l'admin l'ajoute dans la base, il peut ajouter les données textuelles sans inclure les photos requises. Il aura la possibilité de capturer les photos prochainement.
  - Supprimer un employé: l'administrateur aura la possibilité de supprimer un salarié définitivement de la liste des employés.
  - Consulter profil employé: l'administrateur aura la possibilité de consulter le profil de chaque employé.
  - Gestion des profils: l'administrateur aura la possibilité de mettre à jour toutes les données des salariés en dehors du numéro de la carte d'identité nationale et de l'historique de présence/absence de chaque.

- Ajouter images d'employés: l'administrateur doit capturer trois images (face, droite, gauche) pour chaque salarié pour faciliter à la caméra de bien connaître les employés.
- Gérer statistiques: L'administrateur aura la possibilité de :
  - Consulter les états de présence et les états d'absence individuel ou par service.
  - Visualiser la liste des employés présents ou absents en temps réel.
  - Visualiser des pourcentages de présences et d'absences d' employés par période et par des par service.
- Gérer salaire des employés : L'administrateur aura la possibilité de :
  - Consulter le salaire de chaque employé.
  - Ajouter des primes.
  - Modifier Salaire.
- Gérer services: L'administrateur aura la possibilité de :
  - Ajouter un service.
  - Consulter les services avec le pourcentage des employés affectés au service.
  - Supprimer un service.
  - Modifier un service.

#### **Concernant l'employé :**

- S'authentifier: L'employé doit s'authentifier pour qu'il puisse accéder à son profil.
- Consulter son profil: L'employé a le droit de :
  - Consulter son salaire.
  - Consulter le nombre de jours de sa présence ou absence.
  - Envoyer une réclamation.
- Modifier son mot de passe.

### **1.2.2 Besoins non fonctionnels**

les besoins non fonctionnels décrivent les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement d'où l'application doit présenter une interface agréable et facile à utiliser. Le système doit répondre aux exigences suivantes :

**l'ergonomie des interfaces :** L'application doit être compatible sur différents systèmes d'exploitation (LINUX, Windows...) pour être mieux adaptée aux utilisateurs visés.

#### **La sécurité :**

- Il est nécessaire de s'authentifier avant de faire n'importe quelles tâches.
- Tous les mots de passe doivent être cryptés dans la base.

**Audit :** La vérification des champs est importante lors de la saisie pour éviter de saisir des caractères incorrects.

**La fiabilité et la facilité à administrer :** Il faut que le système conçu soit facile à administrer et capable de fonctionner sans erreur pour améliorer l'interaction entre l'utilisateur et la machine.

### 1.2.3 Identification des acteurs

Un acteur est une personne, un matériel ou un logiciel qui interagit avec le système dans le but de réaliser une ou plusieurs fonctions concernant les cas d'utilisations. Pour l'application Pointini nous les avons évoqué dans ce qui précède mais nous donnons de plus amples détails dans le tableau les principaux acteurs de cette application sont réalisés dans le tableau 1.1:

Table 1.1: Tableau des acteurs

Acteur	Rôle
Administrateur	un admin a pour rôle d'entraîner la caméra (entrer trois images pour chaque employé selon trois positions : de face, de gauche et de droite), gérer les employés et gérer les statistiques. Dépendant de la taille de l'entreprise, il peut être le dirigeant ou bien le chef du personnel ou bien le responsable des ressources humaines ou encore un gardien portier qualifié.
Employé	Tout employé qui a pour droit de consulter son état (présence/absence), ses informations personnelles et son calcul de salaire. Ce sont surtout les ouvriers et les salariés en dehors des cadres qui, en effectuant des heures supplémentaires bénéficient de primes et/ou de paiement double et à l'heure.

### 1.2.4 Diagramme de contexte

La figure 1.2 présente le diagramme de contexte qui montre les acteurs interagissant avec le système "Pointini". Pointini est un système de pointage basé sur la reconnaissance faciale, qui permet à un admin de gérer la liste des employés et de visualiser des statistiques. Il permet à un employé de consulter son état de présence et son salaire.



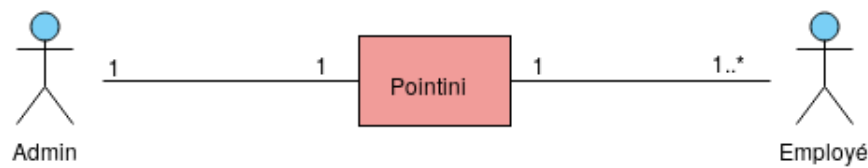


Figure 1.2: Diagramme de contexte

## 1.3 Méthode de développement

### 1.3.1 Méthode adoptée

Le majeure hantise de toute équipe de développement est de ne pas terminer le projet dans les dates prévues à cause d'une mauvaise spécification ou d'un changement brusque des besoins cela conduit à un environnement de travail stressant et à la peur de dépassement de capacités et des délais de livraison. Le recours à l'Agile permet de surpasser ces types de problèmes. Heureusement que l'agence Astrolab est Agile et utilise Scrum pour la gestion de ses projets. Le cadre de processus Scrum est connu pour ses nombreux vertus dont le numéro un est la production de logiciels de qualité s'ajoute la maîtrise de l'évolution des besoins du marché et de l'utilisateur [3]. Nous ajoutons que Scrum est :

- Facile à comprendre.
- Donne une souplesse dans le développement d'un projet.
- Mène vers un produit s'adapte le plus aux exigences du client et des utilisateurs.
- Permet par ses rôles et ses événements une coopération étroite entre clients et équipes de développement [1].

### 1.3.2 Constitution de l'équipe

est un framework composé de trois principaux rôles qui sont :

**Product Owner (PO):** C'est le représentant du client, c'est lui qui connaît parfaitement et sait définir les fonctionnalités du produit, c'est lui qui les priorise. C'est à lui d'accepter ou de rejeter les tâches réalisées. Il est le maître de la réunion de planification de Sprint. Dans le cas de réalisation de mon projet, le PO est monsieur Mohamed DARDOURI.

**Scrum master (SM):** Madame Hédia JEGHAM a chargée de prendre le rôle du Scrum master puisqu'elle assure le suivi de processus, assure un travail complet et productif, entraîner l'équipe de développement dans des environnements organisationnels dans lesquels Scrum n'est pas encore adopté et compris et les aider à créer un produit de grande valeur ajoutée.

**Équipe de développement (DevTeam):** Nous, Sana Dallel, représentons l'équipe de développement puisque nous réalisons toutes les fonctionnalités. Nous sommes tenus de présenter chaque daily meeting les résultats de notre travail. Nous maintenons à jour les spécifications détaillées de notre application et nous sommes entièrement engagés et responsable de la réalisation des User Stories du Product Backlog [4].

## 1.4 Étude comparative

Notre système Pointini s'apparente à un système de pointage. Un système de pointage est un outil simple qui permet de mesurer les heures de travail des salariés, cet outil a du évoluer au cours du temps. Il existe des différents types de systèmes de pointage tels que: la feuille de pointage, la pointeuse par les empreintes digitales, la pointeuse mécanique, la pointeuse mobile, la badgeuse, le eCheck...

**Feuille de pointage:** Comme tout système, ce système permet d'ajuster les heures de travail avec précision de manière que chaque salarié reçoive le salaire qu'il mérite. Ce système n'est pas payant, il est peu coûteux, mais présente de nombreux inconvénients qui font que les entreprises évitent de les utiliser, surtout les grandes entreprises. Nous citons ces inconvénients comme suit:

- L'interface est limitée : n'est pas pratique pour les entreprises qui ont un grand nombre de salariés.
- Les heures d'entrée et de sortie des employés ne peuvent pas être enregistrées.
- Beaucoup de temps est perdu.
- Un employée spécialisé pour marquer les absences [5].

**Système de reconnaissance par les empreintes digitales:** Ce système est un peu plus innovant que la feuille de pointage, fiable, performant et facilite le marquage des absences des salariés. Toutefois il présente un gros inconvénient qui est le manque d'hygiène, les microbes des doigts sales se dispersent sur le lecteur [6].

**Badgeuse:** Pour justifier les horaires du personnel, l'employeur doit se mettre en place devant une badgeuse. Une badgeuse permet de mieux gérer le temps de travail d'un employé, elle peut même ouvrir la porte mais le jour où un employé oublie ou perd son badge il ne peut pas accéder. Récupérer un autre badge peut causer une perte de temps. De même qu'un employé peut facilement badger pour un autre.

Afin de contourner tous les inconvénients de ces types de systèmes de pointage, nous proposons de réaliser un système basé sur la reconnaissance faciale, cette technique est peu coûteuse, performante, conviviale, sécurisée et bien acceptée par le public.

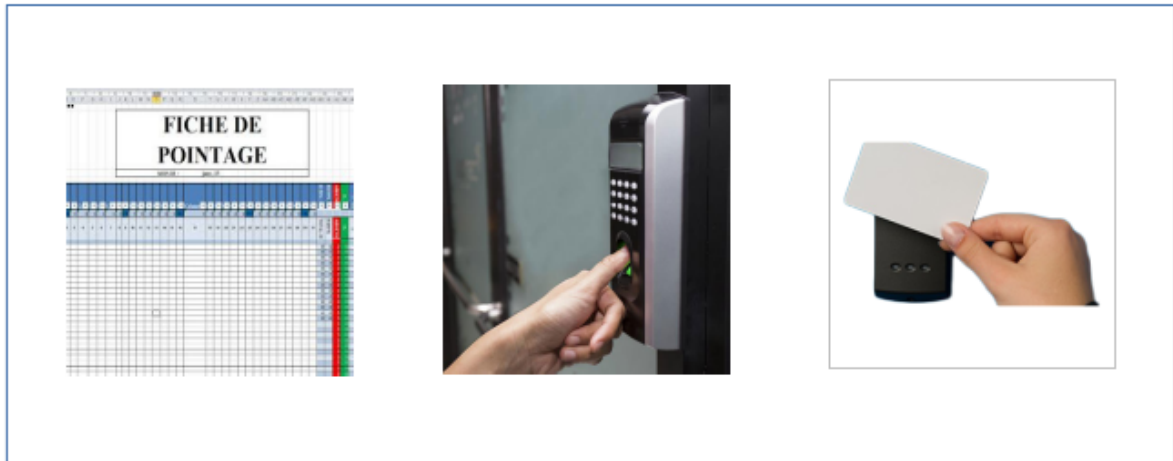


Figure 1.3: Exemples de systèmes de pointages

## 1.5 Architecture globale du logiciel

Dans Scrum il est formellement déconseillé pour la DevTeam de consacrer tout un sprint à étayer les infimes détails de l'architecture d'une application. Celle-ci se construira au fur et à mesure de l'intégration des incréments des différents sprints. Toutefois le Product Owner de par sa maîtrise de son produit, dispose d'une vision claire et globale de son architecture physique et logique. C'est l'architecture globale que nous expliquons dans ce qui suit. Dans Pointini, l'employé et l'administrateur peuvent réagir avec le système.

Dans le premier niveau, l'administrateur utilise un ordinateur pour accéder à ses interfaces Web qui permettent l'envoi des requêtes au serveur et l'affichage des réponses. L'employé de son côté, peut se connecter par son smartphone pour gérer son profil en envoyant des requêtes. Cet employé est constamment capturé par des caméras, d'entrée et de sortie, lors de son passage afin d'informer le serveur de sa présence.

Dans le deuxième niveau, le serveur d'applications Python va recevoir, de part son rôle, toutes les requêtes des clients pour réaliser son traitement et envoyer les réponses à l'employé ou à l'administrateur. Ce serveur souvent a besoin de prendre ou de mettre à jour quelques données et il doit interroger la base. Dans le dernier niveau, la base de données va stocker les données et les fournir au serveur sur demande.

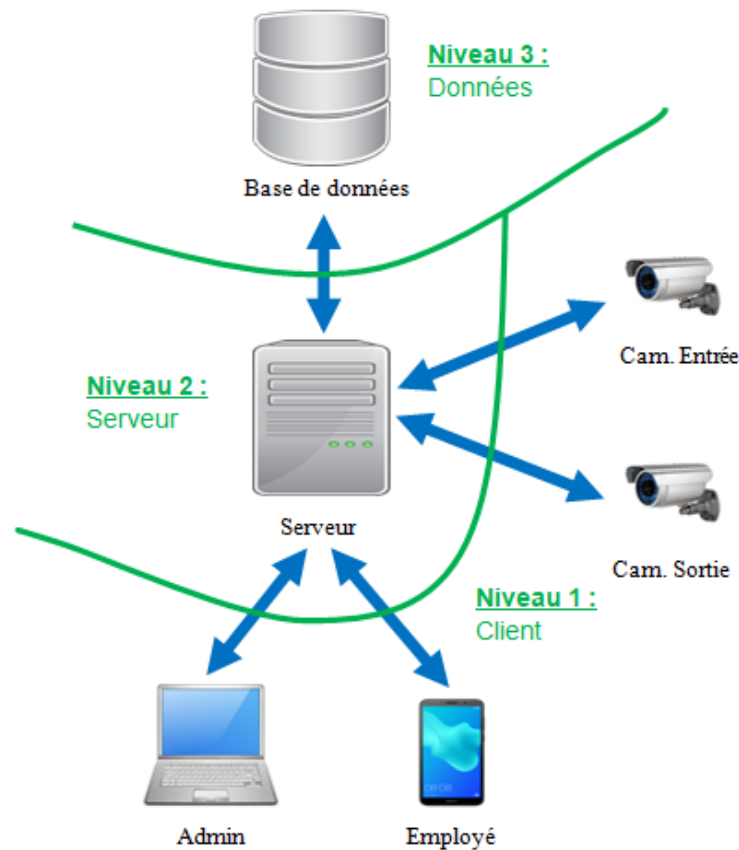


Figure 1.4: Architecture globale du logiciel

## 1.6 Product Backlog de Pointini

Un Product Backlog contient une liste des éléments fonctionnels nécessaires pour l'accomplissement du produit. La DevTeam interviendra avec ses compétences complémentaires, par sa créativité, par son engagement et implication, par sa solidarité et par son grand sens de responsabilité à traduire le Product Backlog en un produit compétitif à placer sur le marché.

Le Product Backlog de Pointini est dressé dans le tableau 1.2 . Il contient les champs suivants :

**ID :** C'est le nombre unique pour chaque user story.

**User stories :** Ce sont les phrases qui décrivent les fonctionnalités souhaitées par le client.

**Priorité :** C'est la priorité du développement de chaque histoire utilisateurs. J'ai pu établir une étude afin de répartir les histoires par priorité selon les normes suivantes:

- La complexité

- La priorité :
  - Élevée : doit être fait.
  - Moyenne : devrait être fait dans la mesure du possible.
  - Faible : pourrait être fait dans la mesure où cela n'a pas d'impact sur les autres tâches [7].

Le tableau 1.2 présente le Product Backlog de Pointini:

Table 1.2: Product Backlog

<b>Id</b>	<b>User stories</b>	<b>Priorité</b>
<b>1</b>	En tant qu'administrateur, je veux pouvoir à tout moment, consulter l'état des employés (la présence, l'absence)	Élevée
<b>2</b>	En tant qu'administrateur, je veux pouvoir avoir en temps réel, le nombre des employés présents ou absents	Élevée
<b>3</b>	En tant qu'administrateur, je peux entraîner mes caméras sur de nouveaux employés en utilisant la méthode de calcul par distance Euclidienne	Élevée
<b>4</b>	En tant qu'administrateur, je veux pouvoir supprimer des employés	Élevée
<b>5</b>	En tant qu'administrateur, je veux pouvoir modifier des employés	Élevée
<b>6</b>	En tant qu'administrateur, je peux être capable d'activer mes caméras	Élevée
<b>7</b>	En tant qu'administrateur, je veux pouvoir détecter les absences pour toute période souhaité	Élevée
<b>8</b>	En tant qu'utilisateur, je veux pouvoir consulter mon profil	Moyenne
<b>9</b>	En tant qu'un utilisateur, je veux pouvoir consulter mes historiques d'absences	Moyenne
<b>10</b>	En tant qu'administrateur, je veux pouvoir consulter la liste des employés	Faible
<b>11</b>	En tant qu'administrateur, je veux pouvoir modifier la photo de l'employé	Faible
<b>12</b>	En tant qu'un utilisateur, je veux pouvoir envoyer une réclamation	Faible

## 1.7 Diagramme des cas d'utilisation global

Le but de cette section est de décrire le comportement attendu de l'application. Pour ce faire, nous nous basons sur le diagramme de cas d'utilisation globale du langage de

modélisation UML pour l'administrateur ainsi que pour l'employé de la partie web et mobile pour modéliser les fonctionnalités de l'application. La figure 1.5 nous montrons ce diagramme qui présente les différentes activités réalisées par l'administrateur afin de gérer et contrôler l'application en plus des différentes activités offertes pour un employé.

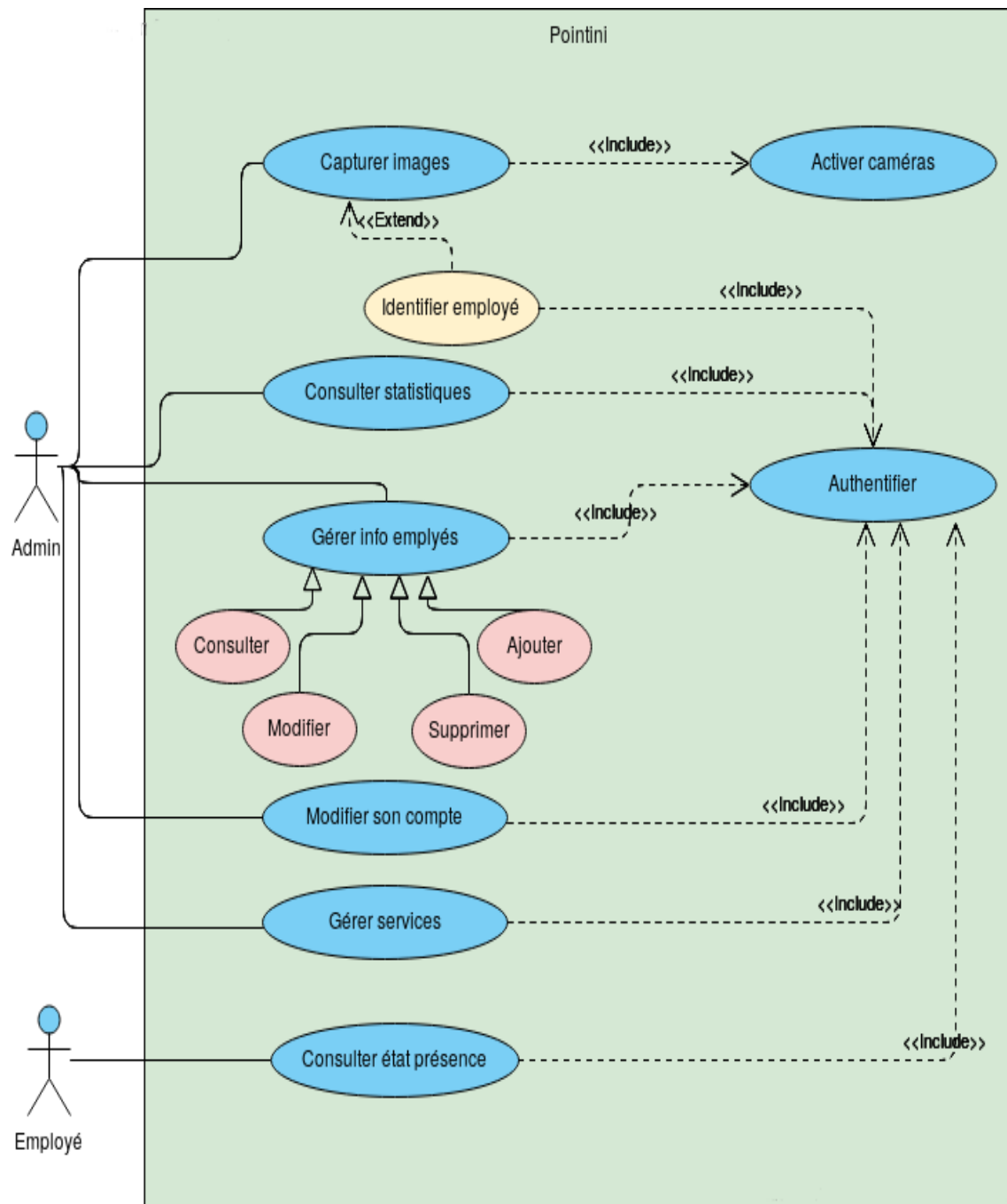


Figure 1.5: Diagramme des cas d'utilisation

## 1.8 Diagramme de Gantt

Les différentes étapes de réalisation de ce projet, ainsi que la période de rédaction du rapport pendant la durée du stage de 4 mois qui s'étend du 01 Février jusqu'au 31 Mai

sont représentées par le diagramme de Gantt de la figure 1.6 suivante.

Il s'agit de la planification des sprint jugés nécessaire pour aboutir à la première release.

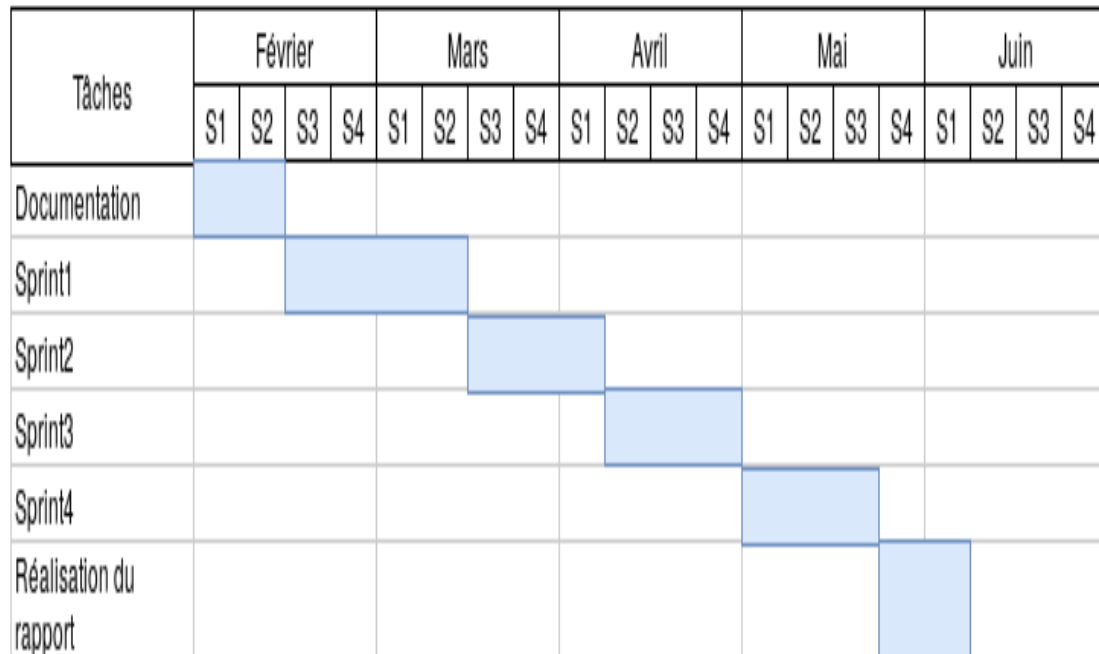


Figure 1.6: Diagramme de Gantt

## 1.9 Établissement de la Definition of Done

Dans le paragraphe qui suit nous établissons notre DoD qui sensée être un standard de qualité de la société. C'est sur la base de la DoD que le PO accepte ou refuse un incrément potentiellement livrable au cours d'un Sprint. Les éléments que peut contenir cette DoD sont :

- L'analyse des besoins pour un objectif d'un Sprint : Il faut que tous les besoins fonctionnels et non fonctionnels soient satisfaits et l'existence sur le marché doit être bien étudiée.

Dans cette application il faut:

- Avoir en temps réel des données exactes sur l'état de présence ou absence des employés.
- La consultation de pointages réalisés par les employés.
- Le stockage et la consultation des données professionnels ou personnelles des employés.
- Le contrôle des options affecté aux différents employés (Ajout, mise à jour, suppression).

- La conception : Pendant cette phase il faut définir l'architecture logicielle ainsi que les interfaces relatifs à un objectif de sprint.
- Le codage et les tests pour un incrément potentiellement livrable : Il faut effectuer la revue de code, vérifier que les tests définis dans les histoires utilisateurs ont été bien réalisés et passés avec succès, bien sûr il faut que le Product Owner voit la démo et valide le fonctionnement.
- La documentation du code, de l'architecture, le help doivent être fournis.

## Conclusion

Dans ce premier chapitre de lancement du projet "Pointini", nous avons introduit l'entreprise d'accueil Astrolab et ses domaines de développement. Nous avons présenté l'équipe Scrum et nous avons détaillé notre cahier de charge. Nous avons aussi transposé le cahier des charges en Product Backlog que nous avons priorisé. A cette étape il y a lieu de rappeler que j'ai joué le rôle du PO. Dans le prochain chapitre l'équipe de développement réunie en ma propre personne, tentera de vider des éléments du PB urgents pour construire son Sprint Backlog selon sa capacité.



## Chapitre 2

### **Sprint1 : Reconnaissance & Apprentissage**

Introduction du chapitre . . . . .	16
2.1 Objectif du Sprint 1 . . . . .	16
2.2 Séance du Sprint planning 1 . . . . .	16
2.3 Sprint backlog 1 estimé . . . . .	17
2.4 Démarrage du Sprint . . . . .	17
2.4.1 Analyse . . . . .	17
2.4.2 Conception . . . . .	20
2.4.3 Codage . . . . .	22
2.4.4 Tests . . . . .	24
Conclusion . . . . .	24

## Introduction du chapitre

Un Sprint est *"le cœur de Scrum"*. Il dure généralement entre 2 et 4 semaines, c'est une suite de tâches que l'équipe Scrum doit bien effectuer pour un objectif préalablement fixé.

Dans ce chapitre nous allons présenter la réalisation du premier Sprint qui se chargera de la réalisation des objectifs des éléments du Product Backlog les plus élevés, pouvant se tenir dans un sprint, et respectant une capacité de travail de DevTeam soutenue.

Rappelons que la DevTeam ce n'est que moi-même. L'organisation du travail se fait sur trois phases principales qui sont l'analyse, la conception et la réalisation.

### 2.1 Objectif du Sprint 1

Chaque Sprint aura des buts aussi déterminés définis auparavant par l'équipe, ce Sprint est entièrement dédié à des technical stories qui ont comme objectif de montrer les importantes étapes à réaliser pour développer la première partie de notre projet qui est la reconnaissance avec apprentissage sur les captures des caméras. les technical stories ce sont des histoires utilisateurs qui n'apportent aucune valeur perçue à l'utilisateur final, mais offre une architecture importante nécessaire pour offrir une valeur future aux utilisateurs.

### 2.2 Séance du Sprint planning 1

Suite à une réunion de planification du Sprint suivie par le product owner qui dure six heures. Nous avons pris la décision suivante pour réussir à effectuer l'apprentissage de la caméra :

- Opencv pour capturer les images à partir des caméras.
- Dlib face detector pour détecter les visages dans une image.
- Dlib face recognition pour extraire le descripteur d'un visage.
- Distance Euclidienne pour trouver le visage le plus proche du visage cible.

La figure 2.1 montre le plan que nous avons créé:



Figure 2.1: Sprint planning 1

## 2.3 Sprint backlog 1 estimé

Le Sprint Backlog est un outil qui facilite la répartition des tâches. Donc pour atteindre l'objectif de ce Sprint, il faut réaliser les tâches présentées dans ce Sprint Backlog sans défaut, ce tableau 2.1 montre les importantes tâches que nous devons les réaliser :

Table 2.1: Sprint Backlog

<b>Id</b>	<b>User stories</b>	<b>Complexity</b>
1	En tant qu'un développeur je dois savoir l'utilisation de Opencv pour capturer par les caméras.	Élevée
2	En tant qu'un développeur je dois savoir l'utilisation de dlib face detector pour la détection des visages.	Élevée
3	En tant qu'un développeur je dois savoir l'utilisation du dlib face recognition pour extraire les descripteurs de visages.	Élevée
4	En tant qu'un développeur je dois maîtriser le calcul de la distance Euclidienne.	Élevée

## 2.4 Démarrage du Sprint

### 2.4.1 Analyse

Plusieurs outils que l'on peut l'utiliser pour établir la reconnaissance faciale telle que l'utilisation de Tensorflow, un framework qui contient des algorithmes qu'on peut utiliser pour faire l'apprentissage des caméras et le training s u r l es images. Tensorflow génère plusieurs calculs et nécessite des capacités de traitement très élevés, d'autant plus qu'il

faut investir beaucoup de temps pour se l'approprier. Notre machine de 4Go et notre durée de stage de 4 mois ne s'apprêtent pas adéquats pour l'utilisation de cette excellente technologie de reconnaissance faciale.

Donc l'idée a été plutôt orientée vers l'utilisation d'une boîte à outils contenant des algorithmes d'apprentissage automatique. Les bibliothèques les plus reconnues qu'on pourrait exploiter pour notre besoin sont openFace et Dlib, la bibliothèque openface donne des résultats non précis et parfois faux (selon l'expérience d'Astrolab dans ce contexte). D'un autre côté, avec l'utilisation du modèle de reconnaissance faciale de Dlib, nous obtenons une erreur moyenne de 0,993833. Ce sera donc notre choix [8].

### Définition et installation du Dlib

Dlib est une bibliothèque que l'on peut utiliser gratuitement dans de multiples applications. Elle contient des algorithmes d'apprentissages automatiques et des outils qui nous aident à créer des logiciels complexes. Dlib est utilisée dans le monde universitaire et dans l'industrie, dans les domaines des appareils intégrés, les téléphones mobiles, la robotique et les grands environnements informatiques [8]. L'installation de Dlib dans Linux est simple il faut tout simplement taper cette ligne de commande dans le terminal: **pip install dlib**

Pour qu'on puisse compiler Dlib il faut installer:

- **python** avec la commande : **sudo apt-get install python3.5** (ou n'importe qu'elle version selon le choix)
- **Cmake** avec la commande: **sudo apt-get install cmake**
- **Numpy** avec la commande: **pip install numpy**



Figure 2.2: Logo de Dlib

### Apprentissage de la Caméra

- **Capture d'image:** Pour qu'on puisse ouvrir notre caméra et capturer des images il faut utiliser Open Cv (Open computer Vision). OpenCV est une bibliothèque d'exécutable qui facilite la reconnaissance d'images en mouvement. Elle fournit une interface permettant à Python de capturer des flux vidéo, et également pour extraire les images afin que nous puissions les traiter plus tard [9].



Figure 2.3: Logo OPenCV

- **Détection des visages:** Après avoir lu les images de flux, nous devons détecter les visages dans chaque image dans l'ordre, pour les identifier plus tard. Cette tâche est assez ardue en raison de la complexité de ses calculs, les chercheurs essaient toujours d'optimiser ses performances. C'est à ce stade que nous arrivons à l'utilisation du détecteur du Dlib. Face-Detection-DataBase du dlib est une base de données créée par David King le créateur du modèle de Dlib. Elle englobe des millions d'images à partir des nombreuses bases de données d'images disponibles au public tel que VG, WIDER, ImageNet, VOC, AFLN, . . . pour bien entraîner son modèle sur la détection faciale [10].
- **Extraction du descripteurs:** Le modèle de la reconnaissance faciale de Dlib est une version réseau ResNet-34 or ResNet est un modèle de deep learning basé sur le CNN spécifié pour les classifications d'images. Ce réseau est entraîné sur un ensemble de données d'environ 3 millions de visages, ce qui rend le modèle de Dlib bien fonctionnel même sur des images de mauvaise qualité ou de faible luminosité. Ce modèle nous permettra l'extraction du descripteur d'une image du visage humain. Le descripteur n'est autre qu'un vecteur de 128 dimensions où les images de la même personne sont proches les unes des autres et les images de personnes différentes sont très éloignées. Nous devons après sa récupération envoyer ce vecteur ultérieurement vers une base de données pour calculer la distance Euclidienne.



Figure 2.4: Positions de visages possible

La figure 2.4 illustre les repères des visages de Dlib, les lignes vertes ne font que relier des points de repère juxtaposés. Ce sont ces repères que Dlib nous aide à définir leurs positions sur un visage pour ensuite pouvoir le rapprocher à ceux d'un visage de référence pour une même personne. La position prend la forme de 68 points de repère. Ce sont des points (X et Y coordonnées) sur le visage tel que les coins de la bouche, le long des sourcils sur les yeux, etc...

- **Distance Euclidienne:** Arrivés à ce niveau de calcul Dlib nous a déjà aidé à définir un vecteur à 128 dimensions pour une image nouvellement capturée. Il faut passer maintenant au calcul de similarité entre ce vecteur et un autre vecteur d'image enregistré sur la base de données à l'occasion de la phase de l'apprentissage de la caméra et de l'enregistrement des données personnelles de l'employé pour la toute première fois. Le calcul de similarité est basé sur le calcul de la distance Euclidienne entre les vecteurs des images enregistrés dans la base de données et le vecteur d'image capturé par la caméra, dans un espace vectoriel de 128 dimensions, puis comparé la distance Euclidienne à un seuil de distance fixe de 0,6. Donc si la distance Euclidienne est suffisamment petite et inférieure à 0,6 la distance seuil, alors ces deux vecteurs ne sont pas pour la même personne sinon ils sont pour la même personne. Le calcul de la distance Euclidienne se fait comme suit :

$$AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

Avec  $A(x_A, y_A)$  et  $B(x_B, y_B)$  sont deux points dans le plan. Dans notre cas la formule de la distance Euclidienne est prédéfinie dans python donc nous invoquons le calcul comme suit:

**distance.euclidean(V1,V2)**

Avec **V1** et **V2** sont les deux vecteurs contenant les descripteurs du visages.

### 2.4.2 Conception

La meilleure solution pour décrire comment Pointini interagit avec les employés est l'utilisation du diagramme de séquence. Un diagramme de séquence permet de décrire comment les éléments du système interagissent entre eux et avec les acteurs dans un ordre chronologique. La figure 2.5 montre le scénario de la reconnaissance d'un employé et le marquage de son état de présence.

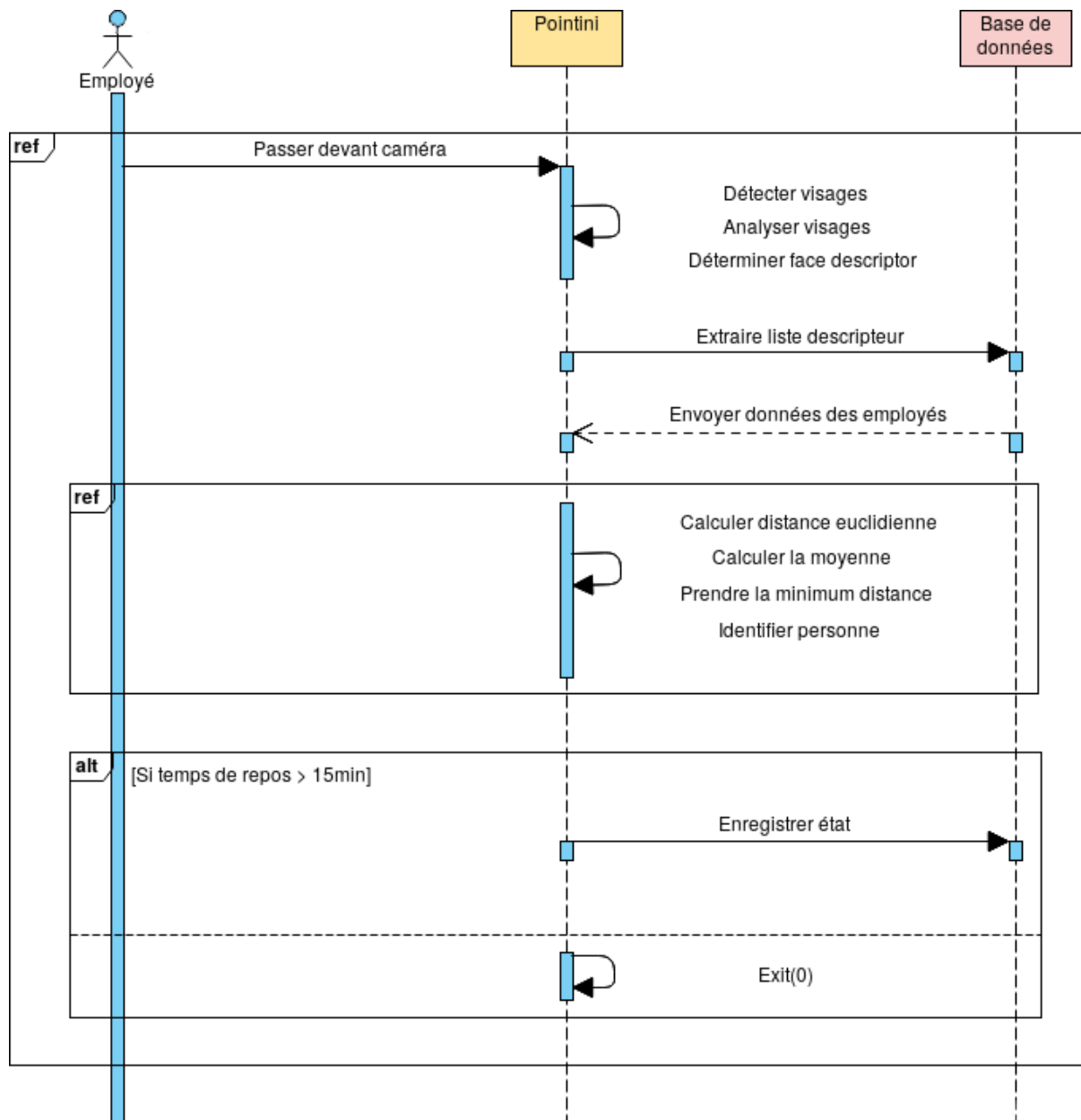


Figure 2.5: Diagramme de séquence : Marque présence

Pour marquer son état de présence, un employé doit passer devant la caméra. Pointini alors peut détecter, analyser et déterminer les descripteurs du son visage. Suite à la réception des données des employés, notre système passe à calculer la distance Euclidienne pour identifier l'employé. Si un employé reste debout devant la caméra pendant une période de 15 minutes ou moins, notre système ne continue pas à marquer l'état de présence de cet employé plusieurs fois, il marque son absence ou présence une seule fois. Sinon, notre système marque l'état et passe au visage capturé qui suit. Bien sur tout Cela ne prend qu'une fraction de seconde. Tout cela c'est pour montrer le principe de la détection et la reconnaissance faciale, mais avant tout ça, il faut enregistrer les données des employés dans notre base de données. La figure 2.6 montre la classe d'objet employé requise.

Employé
<ul style="list-style-type: none"><li>- cin</li><li>- nom</li><li>- prénom</li><li>- mail</li><li>- telephone</li><li>- heure début officiel</li><li>- heure fin officiel</li><li>- début de travail</li><li>- image</li><li>- face descriptor</li><li>- image gauche</li><li>-left face descriptor</li><li>- image droite</li><li>- right face descriptor</li></ul>

Figure 2.6: Table employé

Dans cette classe nous trouvons les données personnelles et professionnelles pour les employés. Ce qui nous intéresse pour l'objectif de ce sprint ce sont les trois images de face, gauche et droite pour le visage d'un employé. Lors de la phase d'apprentissage de la caméra de Pointini et l'enregistrement de ces trois images, notre système calcule le descripteur de visage pour chaque image automatiquement et l'enregistre pour qu'on puisse l'utiliser prochainement.

### 2.4.3 Codage

Dans cette partie nous allons montrer une partie d'un code pour expliquer l'utilisation du modèle du Dlib. Deux parties sont capturées, la première montre comment capturer des images de visages alors que la deuxième partie montre la fonction que nous avons utilisée pour faire la reconnaissance faciale.

```
cap=cv2.VideoCapture(0)
cap2=cv2.VideoCapture(1)

while True:
    _, myimg = cap.read()
    win.set_image(myimg)
    dets = detector(myimg, 1)
    if dets:
        compareMe(myimg,dets,win,records,1)

    _, myimg = cap2.read()
    win2.set_image(myimg)
    dets = detector(myimg, 1)
    if dets:
        compareMe(myimg,dets,win,records,0)
    win.clear_overlay()
    win2.clear_overlay()
```

Figure 2.7: Code capture et détection de visages



Tout d'abord, il faut créer un objet vidéo capture pour capturer une vidéo, pour cela nous déclarons deux variables cap et cap2 puisque nous avons besoins d'utiliser deux caméras, la première pour capturer l'entrée et la deuxième pour capturer la sortie. 0 et 1 sont les index de notre périphérique pour spécifier quelle caméra.

cap.read() nous renvoie les images capturé image par image, donc nous déclarons la variable myimg pour prendre ces images capturées puis cette image sera affichée dans notre fenêtre. Ensuite nous utilisons le variable dets qui va prendre la liste des visages détectés dans cette image pour que nous puissions les comparer avec les visages enregistrés des employés. La fonction de comparaison est présentée dans la figure 2.8.

La première chose à faire est de prendre la liste des visages détectés et puis nous prendrons le visage le plus proche de la caméra. Comment ?, il faut seulement tracer un rectangle sur chaque visage détecté et nous prendrons le visage cadré par le rectangle qui a la plus grande surface.

À cette étape-là, nous fixons le visage à comparer, nous passons alors à calculer son descripteur. Du fait que les descripteurs de visages enregistrés dans la base de données sont prêts, nous devons alors calculer la distance Euclidienne entre ce visage détecté et les trois visages de chaque employé. Enfin nous prenons chaque cinq (id) d'un employé avec la minimale distance entre ses trois visages et le visage détecté.

Maintenant, nous obtenons une liste de distances. Dans une autre partie du code nous passons à la comparaison de chaque distance au regard du meilleur seuil de distance utilisé par Dlib qui est 0.6 . Nous prenons alors l'employé qui a une similarité de 40% ou plus avec l'image de l'employé. Nous arrivons alors à faire la reconnaissance faciale, selon l'état de l'employé (entrée ou sortie) on marque son présence/absence.

```
def compareMe(myimg,dets,win,records,etat):
    print("Number of faces detected: {}".format(len(dets)))
    max=0
    smax=None
    for k, d in enumerate(dets):
        print("Detection {}: Left: {} Top: {} Right: {} Bottom: {}".format(k, d.left(), d.top(),
            d.right(), d.bottom()))
        surf=abs(d.right()-d.left())*abs(d.bottom()-d.top())
        if surf>max:
            shape = sp(myimg, d)
            win.add_overlay(d)
            win.add_overlay(shape)
            max=surf
            smax=shape
    my_face_descriptor = facerec.compute_face_descriptor(myimg, smax)
    scin=[]
    ssim=[]
    for record in records:
        cin=record[u'cin']
        face_descriptor=record[u'face_descriptor']
        face_left_descr=record[u'face_left_descr']
        face_right_descr =record[u'face_right_descr']
        print("Nom et Prénom: {} {}".format(record[u'nom'],record[u'prenom']))
        s = distance.euclidean(face_descriptor,my_face_descriptor)
        sl = distance.euclidean(face_left_descr,my_face_descriptor)
        sr = distance.euclidean(face_right_descr ,my_face_descriptor)
        scin.append(cin)
        ssim.append(min(s,sl,sr))
    imax=np.argmin(ssim)
    mcin=scin[imax]
    msim=ssim[imax]
    print("La personne {} avec distance {}".format(mcin, msim))
```

Figure 2.8: Code reconnaissance faciale

### 2.4.4 Tests

Nous arrivons à la phase du test où nous avons testé notre code pour vérifier si il y'a une partie à corriger ou à améliorer. Suite a les teste effectué à ce code nous avons décidé alors de comparer tous les visages détectés par la caméra et non seulement le visage le plus proche de la caméra. La figure 2.9 montre le visage détecté avec son nom, prénom et CIN qui les correspondant et aussi la valeur la plus petite de ses trois distances Euclidiennes.

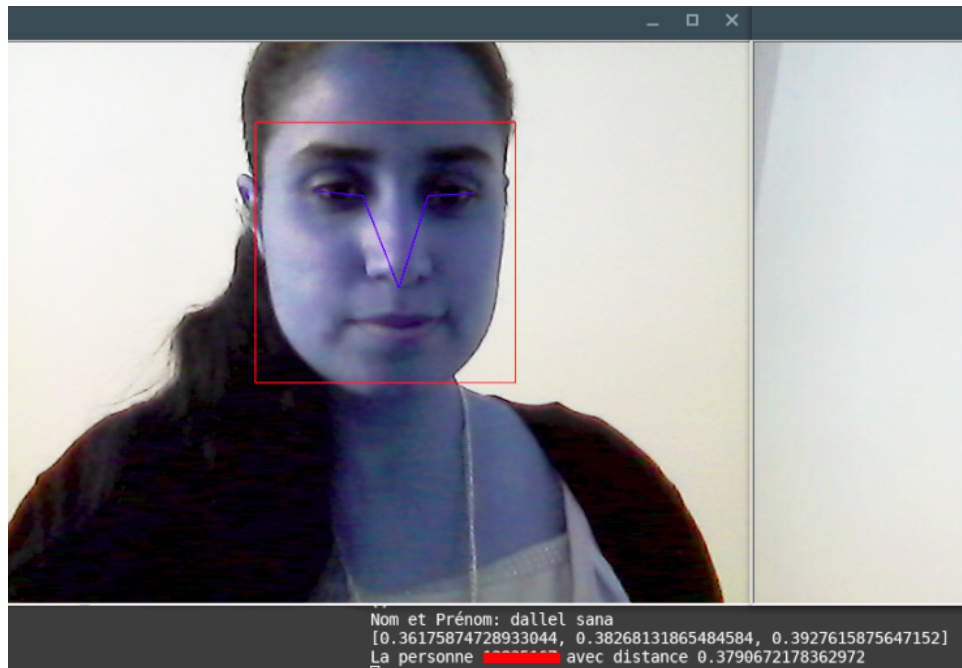


Figure 2.9: Capture: visage détecté

## Conclusion

Ce chapitre est le compte rendu du premier sprint où il y a de l'analyse, la conception, et beaucoup de techniques. Il avait comme objectif spécifique de montrer les étapes à suivre pour réaliser la reconnaissance avec apprentissage sur les captures des caméras. Il est totalement basé sur des technical stories mais à l'occasion de la Sprint Review il a été possible de montrer au client qu'on pouvait déjà reconnaître des visages avec une caméra. Il convient dans les sprints prochains de faire l'habillage de cette fonctionnalité importante.

## Chapitre 3

### Sprint2: Habillage de Pointini

Introduction du chapitre . . . . .	26
3.1 Objectif du sprint . . . . .	26
3.2 Sprint planning . . . . .	26
3.3 Sprint Backlog 2 estimé . . . . .	27
3.4 Démarrage du sprint . . . . .	28
3.4.1 Analyse . . . . .	28
3.4.2 Conception . . . . .	32
3.4.3 Réalisation . . . . .	33
3.4.4 Tests . . . . .	37
Conclusion . . . . .	39

## Introduction

Le chapitre précédent a rapporté le travail fait au cours du premier sprint qui a bouti au premier incrément potentiellement livrable qui n'est autre que le backend de notre produit Pointini. L'infrastructure est prête maintenant pour avancer en faisant sprinter d'autres users stories. Notamment l'habillage de l'application par des interfaces permettant la conviabilité de l'interaction.

### 3.1 Objectif du sprint

Cette partie sera consacrée pour détailler le deuxième sprint où nous allons commencer le développement de notre application web. Notre but sera le suivant : «Terminer les fonctionnalités liées à la gestion des employés, la modification du profil admin et le suivie de caméras».

### 3.2 Sprint planning

La figure 3.1 présente le planning de ce sprint qui a été convenu lors de la réunion où nous répondons à trois questions très importantes qui sont:

**Qu'est-ce qui peut être fait dans ce sprint?**

**Comment sera effectué le travail choisi?**

**Quel est le degré de complexité de son détail ?**

Donc, nous nous penchons sur le développement de la gestion des employés avec l'interface du suivi des caméras, mais commencerons d'abord par l'authentification et puis nous terminerons par l'interface de la modification de profil de l'administrateur.



Figure 3.1: Sprint planning 2

### 3.3 Sprint Backlog 2 estimé

Afin de réaliser notre but pour ce présent sprint, il reste à déterminer les «user stories» qui seront incluses dans le Backlog de ce sprint. Le tableau suivant résume le backlog de notre sprint.

Table 3.1: Sprint Backlog de l'habillage du Pointini

<b>Id</b>	<b>User stories</b>	<b>Tasks</b>	<b>Complexity</b>
1	En tant qu'administrateur, je veux pouvoir m'authentifier	-Affichage de l'interface de l'authentification -Ajouter la méthode de récupération de l'adresse et de mot de passe -Cryptage et décryptage du mot de passe -Vérification de l'adresse et de mot de passe -Tester les cas possibles	Élevée
2	En tant qu'administrateur, je veux pouvoir modifier mon profil	-Ajouter la méthode de modification -Afficher l'interface de modification -Tester la modification du profil de l'admin	Moyenne
3	En tant qu'administrateur, je veux pouvoir ajouter des employés	-Ajouter la méthode d'ajout dans le serveur flask -créer la méthode qui consommera le web service dans le front-office (Python) -Tester l'ajout des employés	Élevée
4	En tant qu'administrateur, je veux pouvoir modifier le profil des employés	-Ajouter la méthode de modification -Afficher l'interface de modification -Tester la modification du profil d'un employé	Élevée
5	En tant qu'administrateur, je veux pouvoir consulter la liste des employés	-Ajouter la méthode de récupération de la liste des employés -Tester la récupération la liste des employés	Élevée
6	En tant qu'administrateur, je veux pouvoir supprimer des employés	Ajouter la méthode de la suppression dans le serveur Flask -Tester la suppression des employés	Élevée

7	En tant qu'administrateur, je veux pouvoir consulter le profil d'un employé bien déterminé	-Ajouter la méthode de récupération du profil d'un employé bien déterminé dans un serveur Flask -Créer la méthode qui consommera le web service dans le front-office (Python) -Tester l'ajout des employés	Moyenne
8	En tant qu'administrateur, je veux pouvoir suivre les caméras	-Ajouter la méthode de récupération du données d'une caméra dans le serveur flask -Tester la récupération des caméras	Élevée

## 3.4 Démarrage du sprint

### 3.4.1 Analyse

#### Capture d'image employé :

L'étape la plus importante lors de l'ajout d'un employé ou pour l'identifier est de capturer trois positions de son visage. Pour ajouter ses images, l'administrateur doit activer la caméra et prendre des photos, l'application alors passe à calculer les descripteurs de ce visage.

La figure 3.2 présente ce cas d'utilisation.

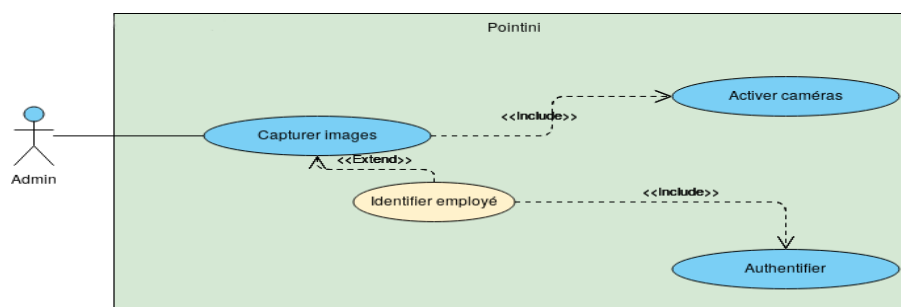


Figure 3.2: Use case: Capturer image employé

La description détaillée de ce cas d'utilisation est donnée par le tableau 3.2 suivant:

Table 3.2: Description textuelle:«Capturer images»

Sommaire d'identification	
<b>Titre</b>	Capturer images employés
<b>Résumé</b>	L'administrateur peut capturer des images pour chaque employé

<b>Acteur</b>	Administrateur
<b>Descripteur des enchaînements</b>	
<b>Pré-conditions</b>	Authentification préalable Caméra ouverte
<b>Post-conditions</b>	Capturer trois visages pour chaque employé
<b>Scénario nominal</b>	1.L'administrateur demande la page d'ajout ou de modifier d'un employé 2.Le système affiche l'interface demandée en questionnant l'admin s'il veut ouvrir la caméra ou nn 3.L'admin accepte d'ouvrir la caméra 4.Le système ouvre la caméra 5.L'admin prendre trois positions de visage 6.Le système ajout les images et calculer le descripteur de visage pour chaque image
<b>Enchaînements alternatifs</b>	<b>E1:</b> Login ou mot de passe invalide 1.Le système affiche un message d'erreur <b>E2:</b> Si le système trouve une erreur lors de calculs de descripteur de visage 1.Le système annule l'action et affiche un message d'erreur 2.Le scénario reprend à partir de scénario 2

### Gérer profil employé:

La figure 3.3 montre le diagramme du cas d'utilisation "Gérer profil employé" destiné à l'administrateur de l'application qui aura le droit de créer des profils des employés, de les consulter, de les modifier ou de les supprimer.

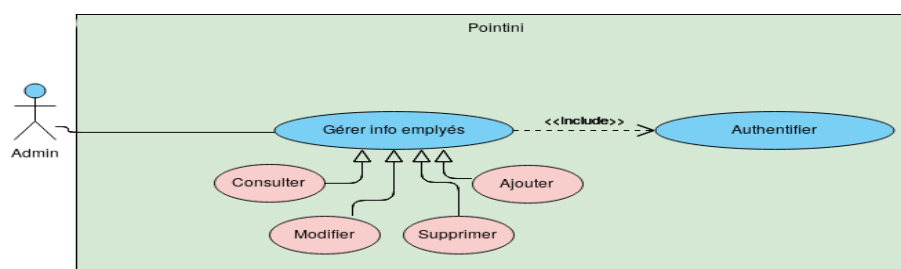


Figure 3.3: Use case: gérer profil employé

La description détaillée du cas d'utilisation "Ajouter employés" est donnée par le tableau 3.3 suivant:

Table 3.3: Description textuelle:«Ajouter employé»

<b>Sommaire d'identification</b>	
<b>Titre</b>	Ajouter des employés
<b>Résumé</b>	L'administrateur peut ajouter des employés

<b>Acteur</b>	Administrateur
<b>Descripteur des enchaînements</b>	
<b>Pré-conditions</b>	Authentification préalable
<b>Post-conditions</b>	Affichage d'un message informant l'administrateur que son action a été effectuée avec succès
<b>Scénario nominal</b>	1.L'administrateur demande la page d'ajout d'un employé 2.Le système affiche le formulaire d'ajout d'un employé en questionnant l'admin s'il veut ouvrir la caméra ou non 3.L'admin remplit soigneusement le formulaire 4.Le système vérifie que les champs sont bien remplis 5.L'administrateur appuie sur le bouton Ajouter 6.Le système vérifie que l'employé n'est pas déjà existant 7.Le système insère l'employé dans la base de données 8.Le système affichait message à l'administrateur l'informant que l'ajout de l'employé a été effectué avec succès
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si l'administrateur n'a pas bien rempli le formulaire: 1.Le système affiche un message d'erreur 2.Le système lui affiche le formulaire de nouveau <b>E2:</b> Si l'employé est déjà existant : 1.Le système annule l'action et affiche un message d'erreur <b>E3:</b> Si l'administrateur n'ajout pas la première date ouvrable de l'employé 1.Le système ajoute automatiquement la première date d'entrée de l'employé

### Modifier profil admin

La figure 3.4 montre le diagramme du cas d'utilisation "Modifier profil administrateur" où l'administrateur de l'application aura le droit de modifier ses données personnels et de modifier son mot de passe.

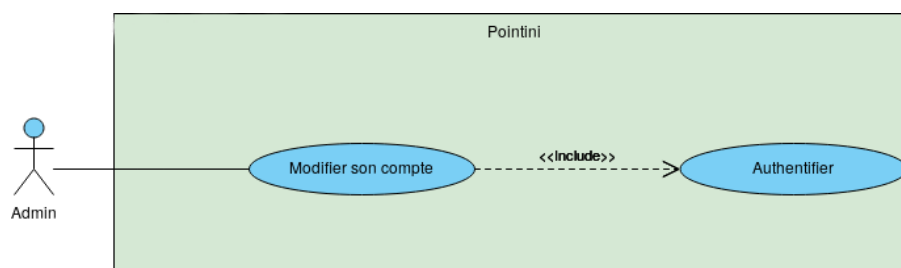


Figure 3.4: Use case: Modifier profil admin

La description détaillée du cas d'utilisation "Modifier profil admin" est donnée par le tableau 3.4 suivant:



Table 3.4: Description textuelle:«Modifier profil admin»

Sommaire d'identification	
<b>Titre</b>	Modifier profil admin
<b>Résumé</b>	L'administrateur a le droit de modifier son profil
<b>Acteur</b>	Administrateur
Descripteur des enchaînements	
<b>Pré-conditions</b>	Authentification préalable
<b>Post-conditions</b>	Affichage d'un message informant l'administrateur que la modification a été effectuée avec succès
<b>Scénario nominal</b>	1.L'administrateur demande à la page de modifier son compte 2.Le système affiche le formulaire de modification du compte admin 3.L'admin remplit soigneusement le formulaire 4.Le système vérifie que les champs sont bien remplis 5.L'administrateur appuie sur le bouton Modifier 6.Le système insère l'employé dans la base de données 7.Le système affiche un message au administrateur l'informant que la modification a été effectuée avec succès
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si l'administrateur a mal inséré son email et son ancien mot de passe: 1.Le système affiche un message d'erreur 2.Le système lui affiche le formulaire de nouveau <b>E2:</b> Si l'administrateur a ajouté deux mots de passe non conformes 1.Le système annule l'action et affiche un message d'erreur

### Suivre caméras

La description détaillée du cas d'utilisation "Modifier profil admin" est donnée par le tableau 3.4 suivant:

Table 3.5: Description textuelle:«Suivre caméras»

Sommaire d'identification	
<b>Titre</b>	Suivre caméra
<b>Résumé</b>	L'administrateur a le droit de suivre les entrées et les sorties des employés
<b>Acteur</b>	Administrateur
Descripteur des enchaînements	
<b>Pré-conditions</b>	Authentification préalable Les deux caméras sont ouvertes
<b>Post-conditions</b>	Affichage de l'interface suivie caméras

<b>Scénario nominal</b>	1.L'administrateur demande la page suivis des caméras 2.Le système affiche l'interface
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si il y'a aucun employé a passé devant la caméra: 1.Le système affiche seulement se qui capte la caméra <b>E2:</b> S'il y a un employé qui passe devant une caméra 1.Le système affiche quelques données personnelles de l'employé avec son état (entré ou sorti) sous les deux fenêtres du caméras

### 3.4.2 Conception

#### Diagramme de séquence: Authentification

Pour accéder à l'application, chaque acteur, employé ou administrateur, est dans l'obligation de s'authentifier avec son email et son mot de passe. L'acteur saisit dans les champs du formulaire l'email et le mot de passe , le système vérifie tout d'abord si les champs sont bien remplis et valides et si ce n'est pas le cas, il avertit l'acteur en affichant un message d'erreur. En cas de succès, le système vérifie l'existence du compte dans la base de données. En cas d'échec, le système affiche un message d'erreur à l'acteur lui demandant de vérifier son email. Si les informations sont correctes, l'authentification est réussie et l'application se dirige vers la page d'accueil qui est l'interface de suivi caméra pour le compte administrateur et l'interface du profil employé pour le compte d'un employé.

La figure 3.5 représente le diagramme de séquence de ce cas d'utilisation.

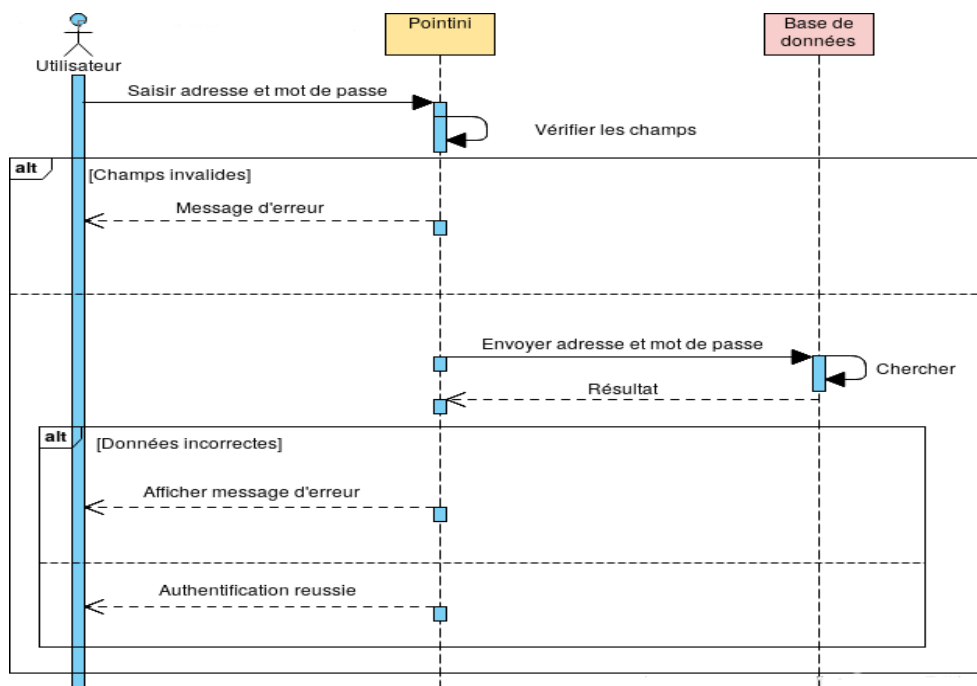


Figure 3.5: Diagramme de séquence : Authentification

### Diagramme de classes

Afin d'identifier la structure de chaque objet du système sous la forme d'une classe, lui donner des propriétés et lui assigner un comportement nous avons réalisé le diagramme de classes participantes pour la gestion des employés. La figure 3.6 suivante représente le diagramme de classes du sprint 2:

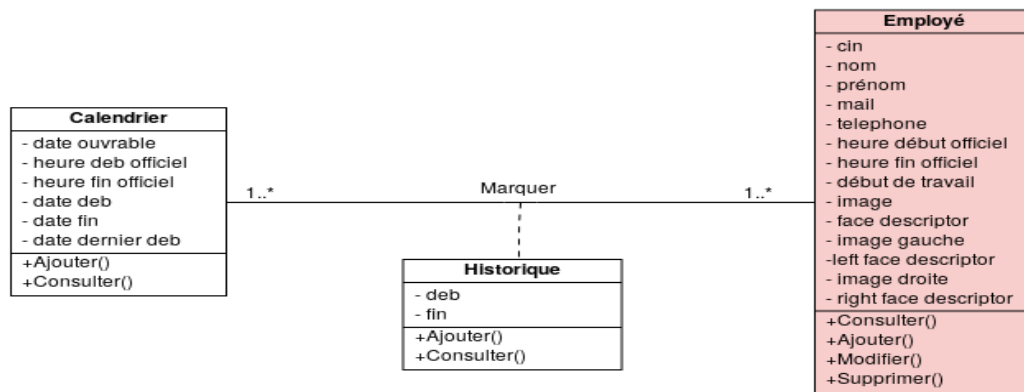


Figure 3.6: Diagramme de classe de sprint 2

### 3.4.3 Réalisation

Dans un premier temps, nous allons présenter les technologies utilisées qui nous ont permis de réaliser notre application web.

Voici la liste des dépendances choisies pour construire notre système:

**OpenCV:** comme son nom l'indique, il s'agit d'une bibliothèque Open Computer Vision, cela nous a beaucoup aidé à saisir les images de flux vidéo. **Dlib:** Bibliothèque de Davis King

**Python:** Le langage parfait pour nos travailleurs. A machine learning friendly language and Dlib provide a good API for it.

**MongoDB:** Un choix parfait pour gérer des bases de données volumineuses.

**Flask:** Framework web en python

**JSON:** JavaScript Object Notation est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est basé sur un sous-ensemble du langage de programmation JavaScript [13].

**Javascript :** est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web [14].

**JQuery:** est une bibliothèque JavaScript rapide, petite et riche en fonctionnalités. Cela rend les choses comme la traversée de documents HTML et la manipulation, la gestion d'événements, l'animation beaucoup plus simple avec une API facile à utiliser qui fonctionne dans une multitude de navigateurs [15].

**HTML5:** HyperText Markup Language 5 est la dernière révision majeure du HTML.

**Bootstrap:** C'est une infrastructure de développement frontale, gratuite et open source pour la création de sites et d'applications Web [11].

**React Native:** Vous permet de créer des applications mobiles en utilisant uniquement JavaScript [12].

Dans Notre application, l'administrateur peut consulter la liste des employés en cliquant sur le bouton «Gestion employés». Ensuite il peut modifier, effacer ou ajouter un ou plusieurs employés.

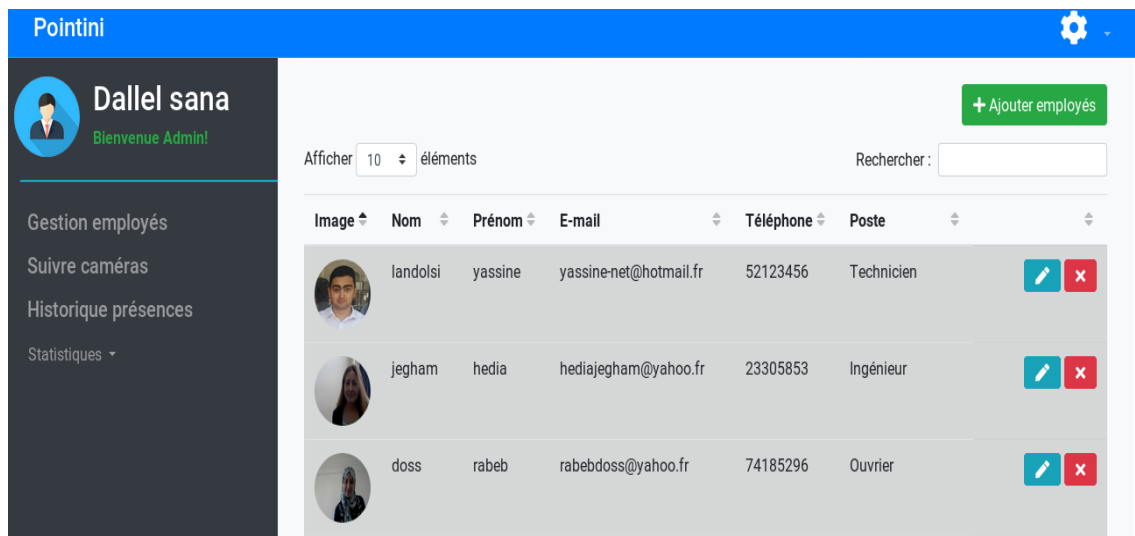


Figure 3.7: Interface 1 : Consulter la liste des employés

Dans les captures présentées par les figures 3.8, 3.9 et 3.10 il s'agit d'un formulaire d'inscription d'un employé dans une entreprise équipée par le système pointini. On constate qu'il y a une partie spécialisée pour capturer les 3 images d'un employé. Si une image est mal capturée on la visualise dans l'une des trois premières images (captures anciennes) et on peut la recapturer une autrefois, cette nouvelle image on le trouve dans l'une des trois dernières images.

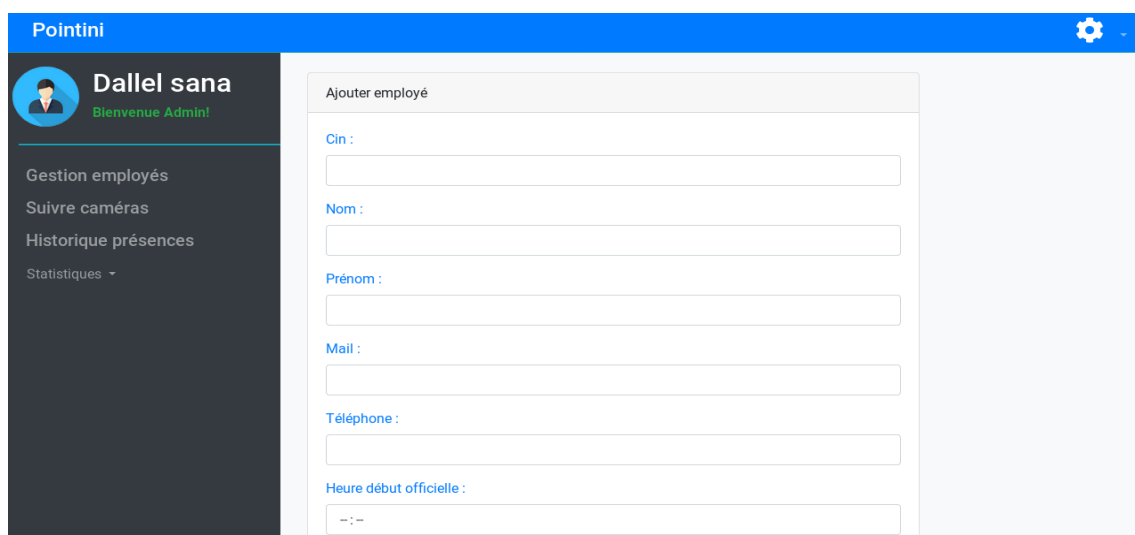


Figure 3.8: Interface 2.1 : Ajouter un employé

Heure début officielle :

Heure fin officielle :

Première date ouvrable (si existe) :

Service :

Poste :

Capturer

Figure 3.9: Interface 2.2 : Ajouter un employé

Capturer

Gauche Face Droite

Enregistrer

Figure 3.10: Interface 2.3 : Capturer des images

L'administrateur aussi peut consulter le profil de chaque employé à travers cette interface en lui présentant ses informations personnelles et professionnelles qui présentés dans les figures 3.11 et 3.12

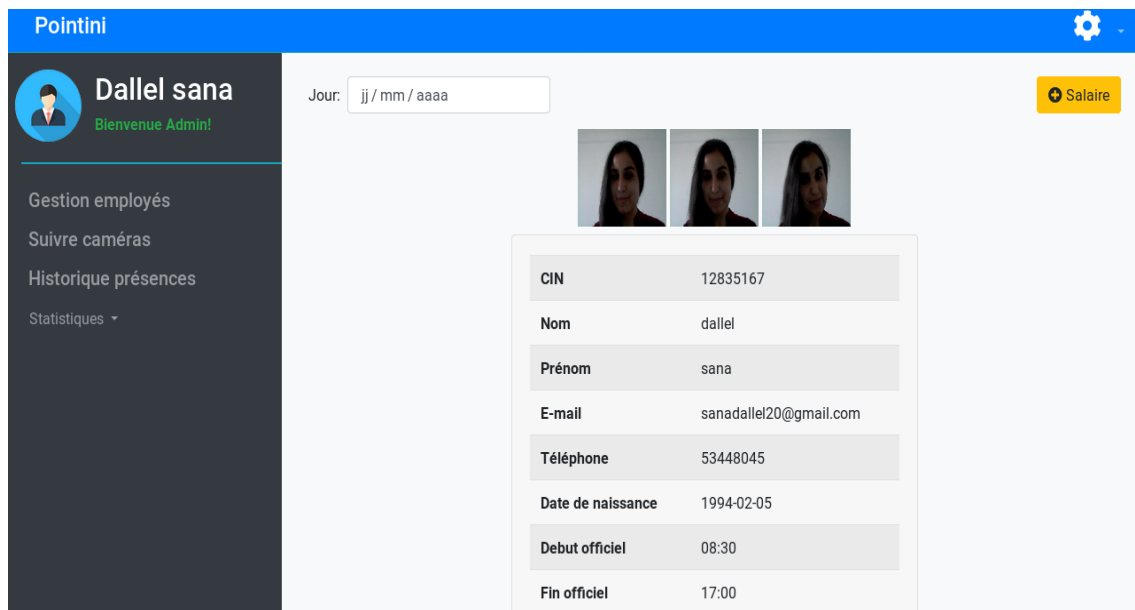


Figure 3.11: Interface 3.1 : Profile employé

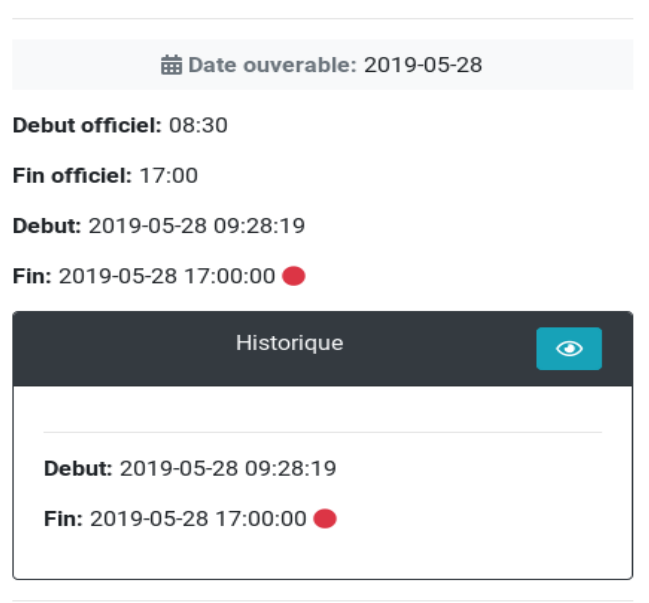


Figure 3.12: Interface 3.2 : Consulter calendrier d'un employé

Le fruit de notre travail trouvé dans le premier sprint qui est la reconnaissance avec apprentissage est présenté dans la figure 3.13 suivante, ce qui assure une interface de suivie de deux caméras avec l'historique des entrées/sorties des employés.

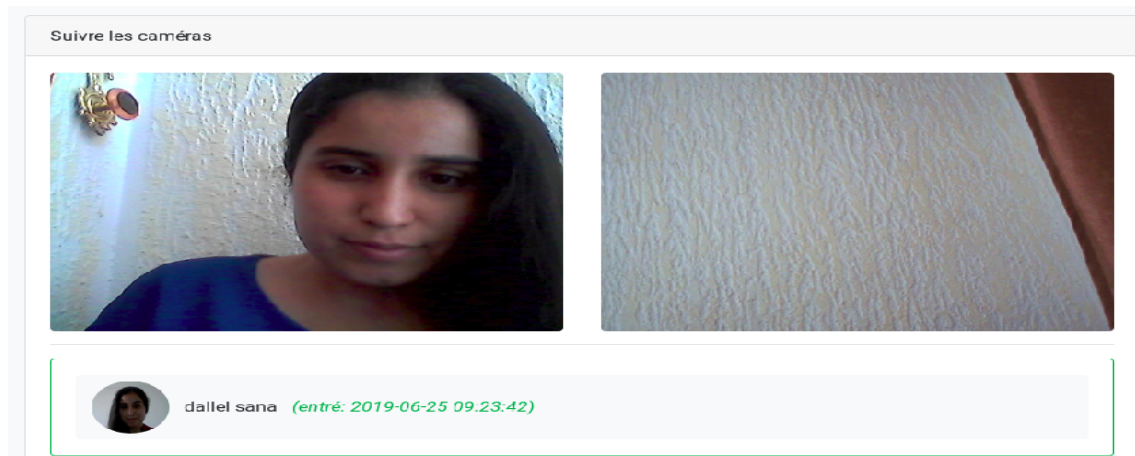


Figure 3.13: Interface 4 : Suivie caméras

### 3.4.4 Tests

Dès le démarrage de l'application, l'interface « Authentification » se présente à l'utilisateur. Ce dernier doit entrer son email et son mot de passe correctement pour accéder à l'application, sinon un message d'erreur sera indiqué comme le représente la figure 3.14 suivante:

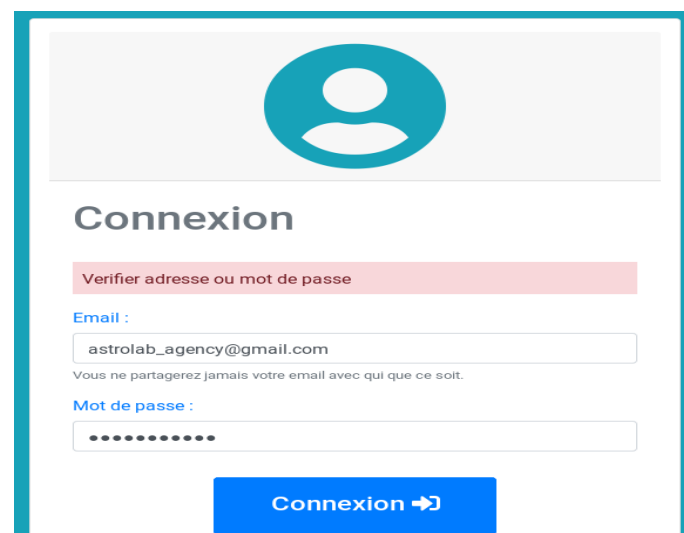


Figure 3.14: Interface 1 : Tester login

Dès que l'admin clique sur le bouton «Modifier compte», l'interface de modification s'affichera . Lorsque l'admin a bien saisi son email et mot de passe un message de réussite s'affichera comme le représente la figure 3.15 et les informations seront bien modifiées.

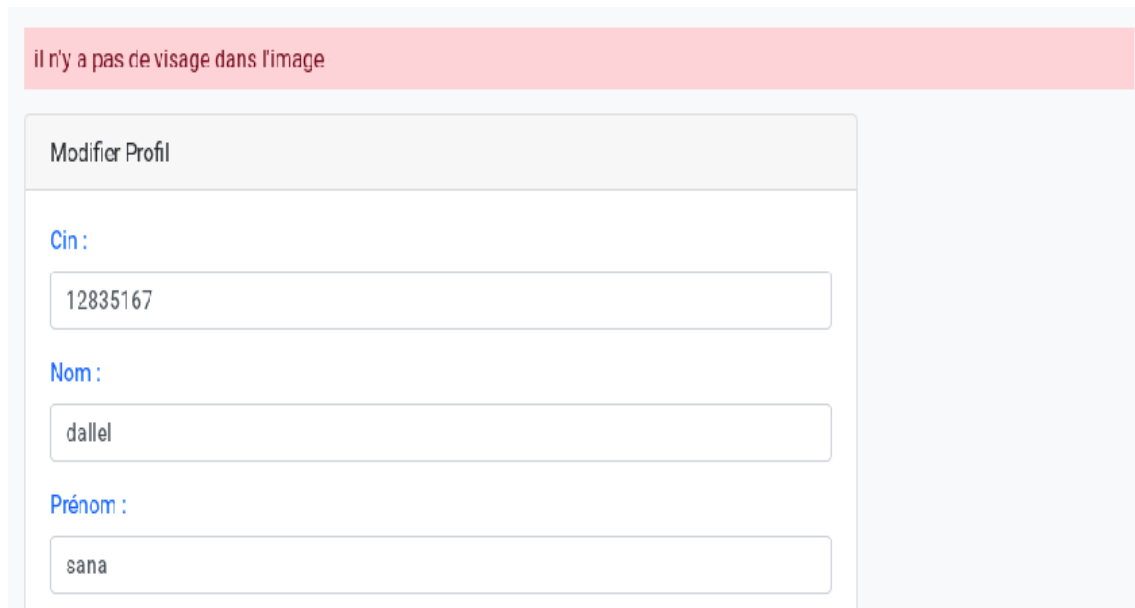
Figure 3.15: Interface 2 : Tester modification profil admin

Lorsque l'admin veut ajouter ou modifier un profil d'un employé, un message d'autorisation pour ouvrir la caméra sera affiché comme suite:

Figure 3.16: Interface 3 : Autorisation d'ouvrir la caméra

Si l'image capturée par l'admin ne contient pas un visage ou elle contient un visage pas clair un message d'erreur sera affiché comme le montre le capture 3.17 suivant:





il n'y a pas de visage dans l'image

Modifier Profil

Cin :

12835167

Nom :

dallel

Prénom :

sana

Figure 3.17: Interface 4 : Tester ajout des images

## Conclusion

Jusqu'à ce stade nous avons développé une bonne partie de notre application web en arrivant à un produit complet et fonctionnel. Ce chapitre avait pour objectif de faire la description, la conception et la réalisation de cette partie qui consiste à terminer la gestion des employés ainsi que l'authentification et la modification du profil admin.

## Chapitre 4

# Statistiques & Facette Mobile

Introduction du chapitre . . . . .	41
4.1 Sprint 3: Offre des statistiques de Pointini . . . . .	41
4.1.1 Sprint backlog 3 estimé . . . . .	41
4.1.2 Analyse . . . . .	42
4.1.3 Conception . . . . .	45
4.1.4 Réalisation . . . . .	46
4.1.5 Tests . . . . .	48
4.2 Sprint 4: Pointini facette mobile . . . . .	49
4.2.1 Sprint backlog 4 estimé . . . . .	49
4.2.2 Analyse . . . . .	50
4.2.3 Réalisation . . . . .	51
4.2.4 Tests . . . . .	53
4.3 Déploiement de Pointini en environnement d'exploitation . . . . .	53
Conclusion . . . . .	54

## Introduction

Une fois que nous avons terminé l’habillage de notre application, nous pouvons passer au quatrième chapitre où nous nous allons détailler deux sprints dans le but d’atteindre leurs objectifs : Offre des statistiques pour l’administrateur et la facette mobile pour les employés.

### 4.1 Sprint 3: Offre des statistiques de Pointini

Cette partie sera consacrée pour détailler le troisième sprint où nous allons terminer le développement de notre application web. Notre but sera le suivant: «terminer toutes les fonctionnalités qui restent dans l’application web».

#### 4.1.1 Sprint backlog 3 estimé

Ce sprint englobe 4 user stories du Product Backlog :

- La consultation de l’administrateur des statistiques, des salaires et de l’état de présence
- La modification du salaire par l’administrateur

Le tableau 4.1 suivant résume le backlog de ce sprint.

Table 4.1: Sprint Backlog de sprint 3

<b>Id</b>	<b>User stories</b>	<b>Tasks</b>	<b>Complexity</b>
<b>1</b>	En tant qu’administrateur, je veux pouvoir consulter les statistiques	-Affichage d’une courbe qui présente le nombre de présences des employés -Affichage d’un pie chart qui présente le pourcentage de distribution des employés par service -Affichage d’un histogramme qui présente le nombre des employés présents pour chaque service	Élevée
<b>2</b>	En tant qu’administrateur, je veux pouvoir consulter les salaires des employés	-Ajouter la méthode de récupération des données de salaires -Afficher l’interface de salaire pour chaque employé -Tester la récupération des données correctes du salaire	Élevée

3	En tant qu'administrateur, je veux pouvoir modifier le salaire des employés	-Ajouter la méthode de modification -Afficher l'interface de modification -Tester la modification du salaire d'un employé	Élevée
4	En tant qu'administrateur, je veux pouvoir consulter l'historique de présence des employés en temps réel	-Ajouter la méthode de récupération des données -Tester la récupération des données	Moyenne

4.1.2 Analyse

Consulter statistiques

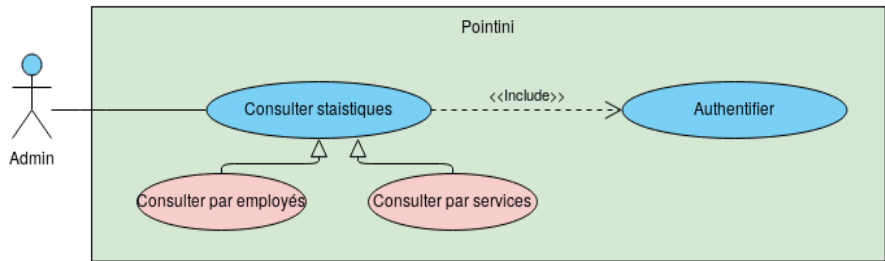


Figure 4.1: Use case: Consulter statistiques

La figure 4.1 précédente présente le diagramme de cas d'utilisation «Consulter statistiques» où nous montrons à l’administrateur qu’il peut consulter les statistiques non seulement par des employés mais aussi par services. La description détaillée de ce cas d’utilisation est donnée par le tableau 4.2 suivant:

Table 4.2: Description textuelle:«Consulter statistiques»

Sommaire d’identification	
Titre	Consulter statistiques
Résumé	L’administrateur a le droit de consulter les statistiques par employés ou par statistiques
Acteur	Administrateur
Descripteur des enchaînements	
Pré-conditions	Authentification préalable
Post-conditions	Affichage de l’interface statistique

<b>Scénario nominal</b>	1.L'administrateur demande l'une des pages de statistiques 2.Le système affiche l'interface
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si l'administrateur demande la page «suivre présences» : 1.Le système affichera une courbe <b>E2:</b> Si l'administrateur demande la page «Employés par service» : 1.Le système affichera un pie chart <b>E3:</b> Si l'administrateur demande la page «Absences/Présences par service» : 1.Le système affichera un histogramme

### Consulter états de présences

La figure 4.2 présente le diagramme de cas d'utilisation «Consulter états de présences» dont l'administrateur peut connaître la liste des employés présents ou absents en temps réel.

Puisque la description textuelle de ce cas d'utilisation a presque le même démarche que les autres descriptions textuelles de consultation, nous pensons qu'il vaut mieux de ne pas le répéter.

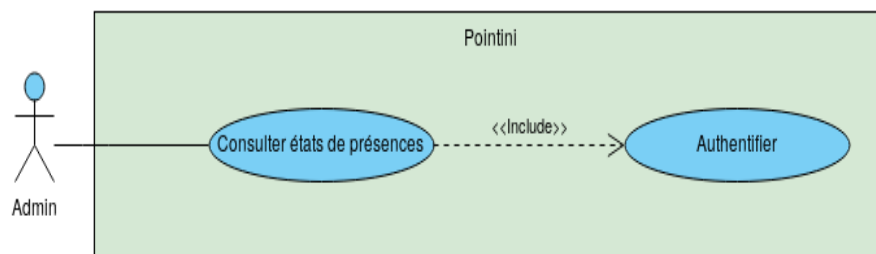


Figure 4.2: Use case: Consulter états de présences

### Calcul salaires

La figure 4.3 montre le diagramme du cas d'utilisation "Calcul salaires" destiné à l'administrateur de Pointini qui aura le droit de calculer des salaires, de les consulter, de les modifier ou de les supprimer.

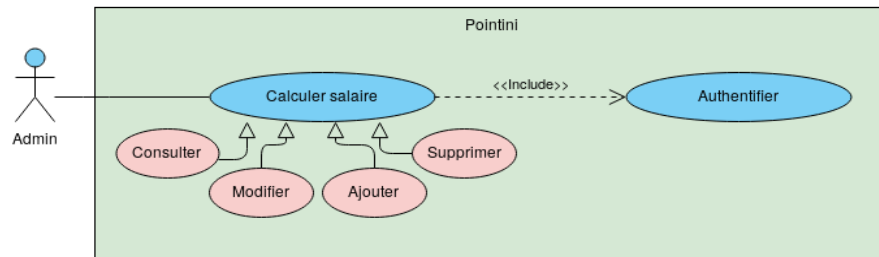


Figure 4.3: Use case: Calcul salaires

La description détaillée du cas d'utilisation «Calculer salaire» est donnée par le tableau 4.3 suivant:

Table 4.3: Description textuelle:«Calcul salaires»

Sommaire d'identification	
<b>Titre</b>	Calculer salaires
<b>Résumé</b>	L'administrateur a le droit de calculer le salaire de chaque employé
<b>Acteur</b>	Administrateur
Descripteur des enchaînements	
<b>Pré-conditions</b>	Authentification préalable L'interface de profile employé est affiché
<b>Post-conditions</b>	Affichage d'un message informant l'administrateur que le salaire a été calculer avec succès
<b>Scénario nominal</b>	1.L'administrateur demande l'interface de salaire d'un employé bien déterminé 2.Le système affiche calcul les heures du travail et affiche l'interface du calcul salaire 3.L'administrateur entre les nombres d'heures supérieurs et puis clique sur le bouton "Montant" 4.Le système calcule le salaire 5.L'administrateur vérifie le montant et clique sur le bouton "Payer" 6.Le système enregistre le salaire et affiche un message de réussite
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si l'administrateur fait une erreur lors de l'ajout des nombres des heures: 1.L'administrateur peut le modifier ou le supprimer <b>E2:</b> Si l'administrateur veut visualiser la fiche de salaire de l'employé : 1.Il clique sur le bouton "Imprimer" 2.Le système affiche la fiche à imprimer

### 4.1.3 Conception

La figure 4.4 illustre le diagramme de classe de la totalité du projet. Ces différentes classes représentent les éléments qui composent le modèle de données et leurs relations.

- La classe employé présente les données personnelles des employés
- La classe calendrier présente les données professionnelles des employés
- La classe historique est une classe intermédiaire qui présente l'heure de début et l'heure de la fin de chaque période quelque soit l'entrée ou la sortie

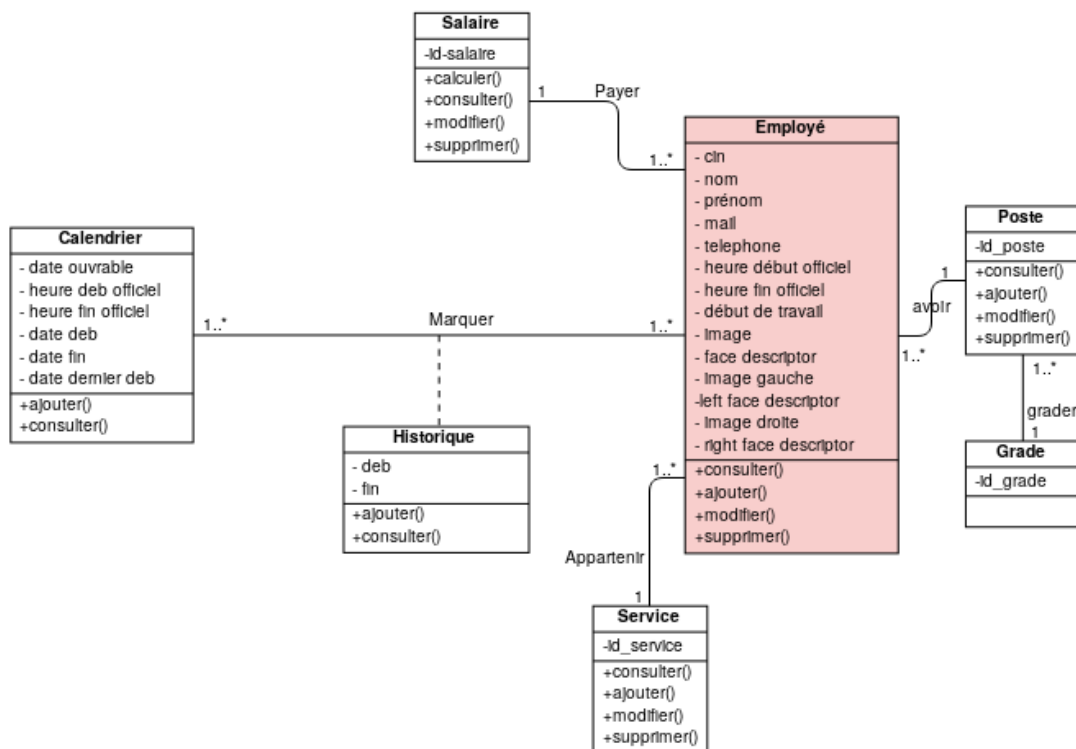


Figure 4.4: Diagramme des classes globale

#### 4.1.4 Réalisation

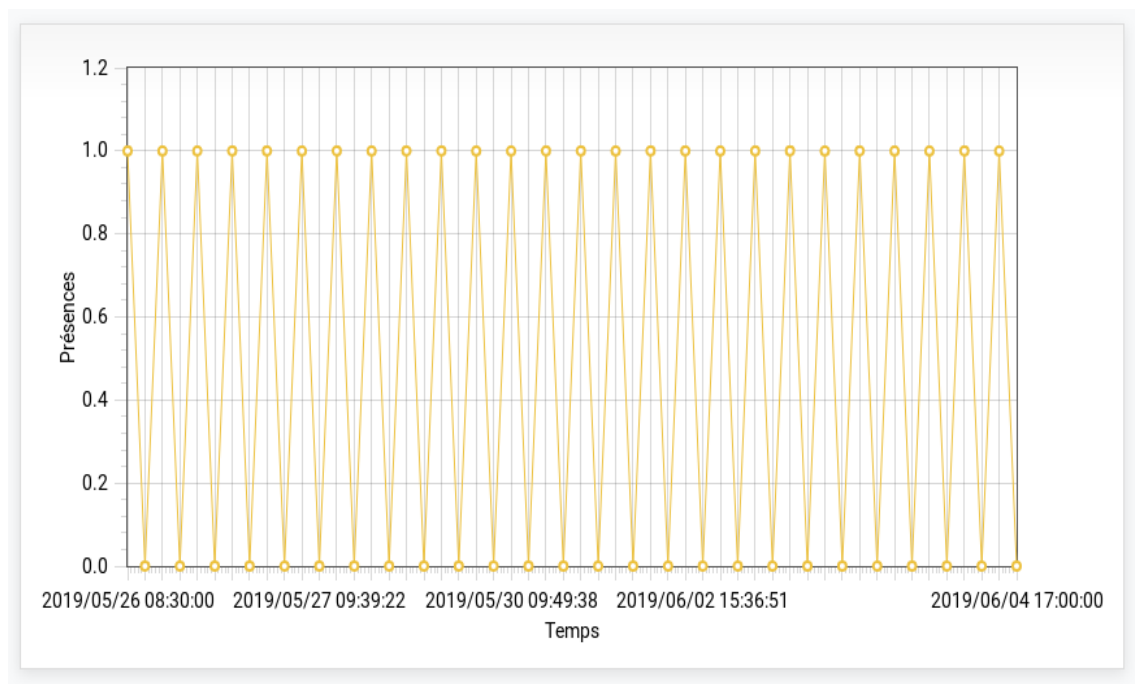


Figure 4.5: Interface 1.1 : Statistique

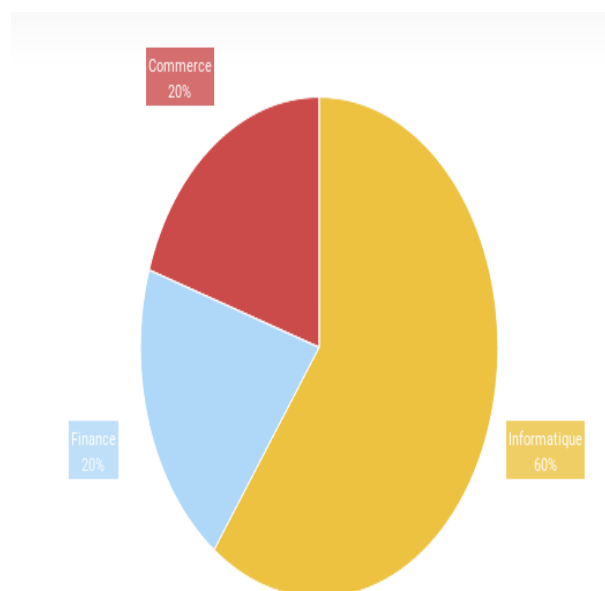


Figure 4.6: Interface 1.2 : Statistique



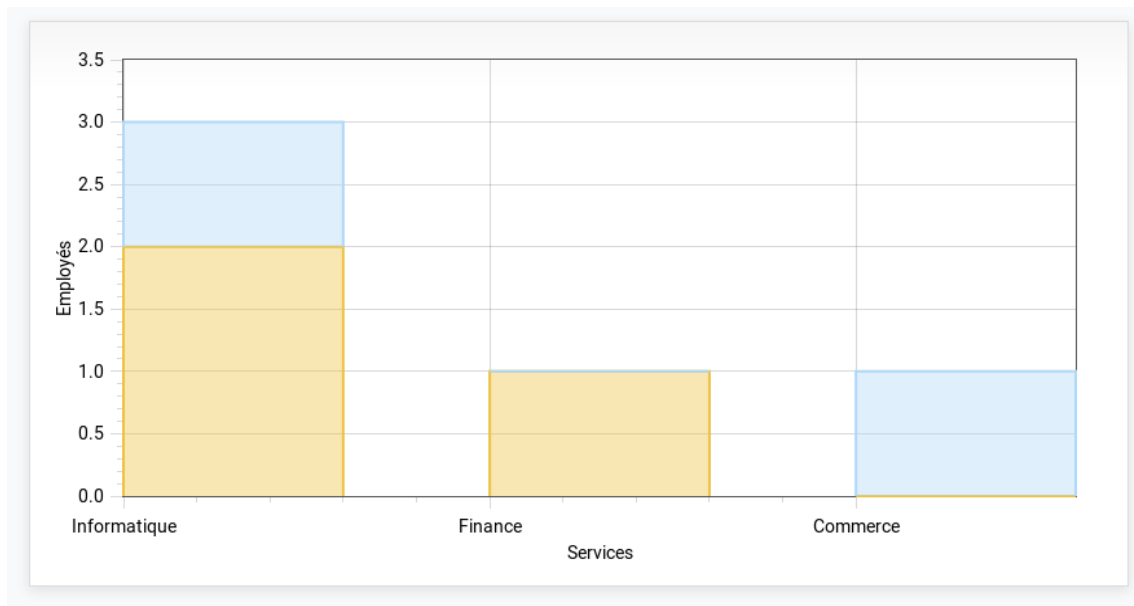


Figure 4.7: Interface 1.3 : Statistique

En accédant aux interfaces 4.5, 4.6 et 4.7 l'administrateur aura une idée sur les nombres des employés présents en temps réel, le pourcentage de distribution des employés par services et le nombre des employés présents par service.


La figure 4.8 présente l'interface dont laquelle l'administrateur peut agir dessus pour calculer le salaire de chaque employé.


de

jj / mm / aaaa

à

jj / mm / aaaa

  
Montant

  
Heures

Heures off.:

67,9084

Heures sup.:

4,0408

Montant =

144,381

DT

+ Payer

Figure 4.8: Interface 2 : Calcule salaire

La figure suivante présente l'état de présence d'un employé en temps réel où on va le trouver dans l'interface "Historique présence". La boule rouge montre que cet employé est absent à ce moment-là alors que la boule verte montre le cas contraire.

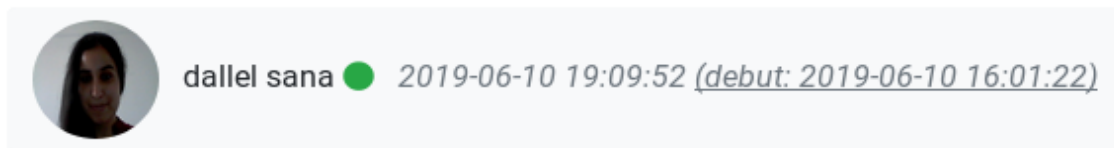


Figure 4.9: Interface 3 : État présence

### 4.1.5 Tests

Lorsque l'admin clique sur le bouton "Montant" suite à l'ajout de nombre d'heures supérieurs, le système calcule le nombre des heures et calcul le salaire de l'employé puis il affiche un message de réussite comme le représente la figure 4.10.



Figure 4.10: Interface 1 : Salaire calculé

La figure suivante présente une fiche de salaire d'un employé où nous pouvons trouver des informations personnelles avec les historiques et les informations sur son salaire.


<div>  <div> <div>CIN</div>12835167 </div> <div> <div>Nom</div>dallel </div> <div> <div>Prénom</div>sana </div> <div> <div>Début du travail</div>2019-05-22 </div> <div> <div>Service</div>Commerce </div> <div> <div>Poste</div>Technicien superieur </div> </div>							
Historique de paiements							
Temps	Début	Fin	Heures off.	Heures sup.	Début du travail	Grade	Montant
2019-06-12 19:49:01	2019-05-01	2019-06-30	59.33	0.0	2019-05-22	0	63.0822 DT

Figure 4.11: Interface 2 : Fiche salaire

## 4.2 Sprint 4: Pointini facette mobile

Cette partie sera consacrée pour détailler le quatrième sprint où nous allons développer la partie mobile du Pointini. Notre but sera comme suit: «Réaliser une application mobile pour l'employé».

### 4.2.1 Sprint backlog 4 estimé

Ce sprint englobe 2 user stories du Product Backlog :

- L'authentification des employés
- La consultation des employés et de leurs états de présences

Le tableau 4.4 suivant résume le backlog de ce sprint.

Table 4.4: Sprint Backlog de sprint 4

<b>Id</b>	<b>User stories</b>	<b>Tasks</b>	<b>Complexity</b>
<b>1</b>	En tant qu'un employé, je veux pouvoir m'authentifier	-Affichage de l'interface de l'authentification -Ajouter la méthode de récupération de l'adresse et du mot de passe -Cryptage et décryptage du mot de passe -Vérification de l'adresse et de mot de passe -Tester les cas possibles	Élevée

2	En tant qu'un employé, je veux pouvoir consulter mon état de présence	-Ajouter la méthode de récupération des données -Tester la récupération des données	Élevée
---	---	--	--------

4.2.2 Analyse

La figure suivante présente les cas d'utilisation d'un employé où nous pouvons trouver que l'employé a le droit de consulter son profil et son état de présence. Aussi, il a le droit de modifier son mot de passe. Tout cela n'est fait que suite à son authentification.

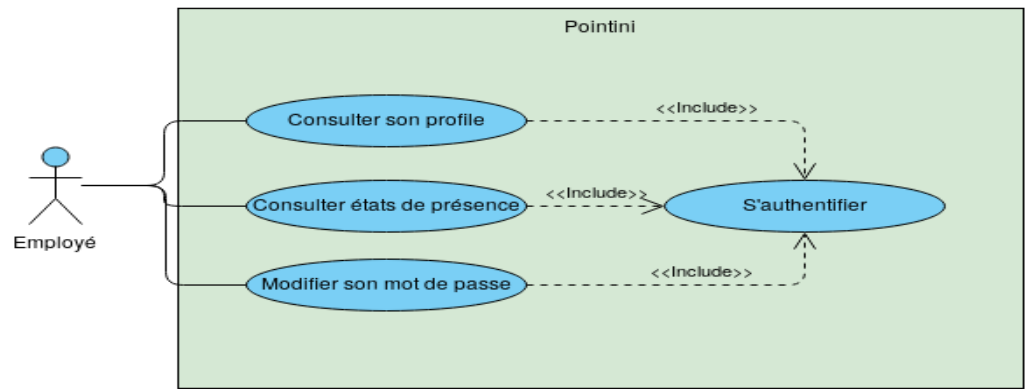


Figure 4.12: Diagramme des cas d'utilisation pour les employés

La description détaillée du cas d'utilisation «Modifier mot de passe» est donnée par le tableau 4.5 suivant:

Table 4.5: Description textuelle:«Modifier mot de passe»

Sommaire d'identification	
Titre	Modifier mot de passe
Résumé	L'employé a le droit de modifier son mot de passe
Acteur	Administrateur, Employé
Descripteur des enchaînements	
Pré-conditions	Authentification préalable
Post-conditions	Affichage d'un message informant l'employé que le mot de passe a été modifié avec succès

<b>Scénario nominal</b>	1.L'employé demande l'interface de modification de mot de passe 2.Le système affiche l'interface de modification de mot de passe 3.L'employé entre son ancien mot de passe et deux fois le nouveau mot de passe 4.Le système enregistre le nouveau mot de passe et affiche un message de réussite
<b>Enchaînements alternatifs</b>	<b>E1:</b> Si l'employé entre un ancien mot de passe incorrecte 1.Le système affiche un message d'erreur <b>E2:</b> Si les deux nouveaux mots de passe ne sont pas conformes : 1.Le système affiche un message d'erreur <b>E3:</b> Si l'employé son ancien mot de passe : 1.L'administrateur peut initialiser le mot de passe de l'employé en cochant "Init. mot de passe". Ici le mot de passe sera changé automatiquement par la date de naissance de l'employé

**Note !** le mail pour accéder à son compte, est le CIN de l'employé.

### 4.2.3 Réalisation

Mise à part l'analyse, ce dernier sprint a été entièrement dédié au développement et test puisqu'il s'agit de solliciter la base de données déjà prête pour des extractions et affichage sur écran mobile. La figure suivante présente une interface de connexion pour l'application mobile où l'employé doit entrer son CIN et son mot de passe pour accéder à son compte.

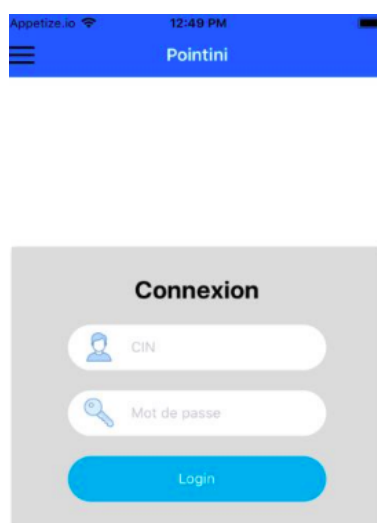


Figure 4.13: Interface connexion

La figure 4.14 présente l'interface de paramètre dans l'application mobile. où l'employé peut consulter son profil, modifier son mot de passe ou bien se déconnecter.



Figure 4.14: Interface paramètres



Figure 4.15: Interface de modification mot de passe

La figure ci-dessus montre l'interface de modification du mot de passe. Ici l'employé doit saisir un ancien mot de passe correcte et deux nouveaux mots de passe conformes.

### 4.2.4 Tests

Les figures 4.16 et 4.17 suivantes présentent le profil d'un employé où il trouve ses données personnelles et professionnelles. Il peut vérifier son état de présence. En cas d'erreur comme la caméra ne marque pas son présence, il peut informer l'administrateur.



Figure 4.16: Interface de profil d'un employé



Figure 4.17: Interface du calendrier d'un employé

## 4.3 Déploiement de Pointini en environnement d'exploitation

Au bout des quatre sprints nous arrivons à la phase de déploiement du produit dans son environnement réel d'exploitation. Le diagramme de déploiement ci-dessous montre en clair les différents composants de Pointini, les nœuds nécessaires et leurs divers modes de communications.

- Le serveur web et le serveur de base de données sont liés par internet pour transférer les données en utilisant le protocole TCP/IP.
- Le serveur web et l'application mobile communiquent par des requêtes et des réponses REST pour que l'application consomme les services web qui sont fournis par le serveur.
- L'application web communique avec le serveur web avec le protocole HTTP en utilisant un navigateur web pour accéder aux interfaces.
- Les caméras sont connectés avec le serveur web par des câbles USB.

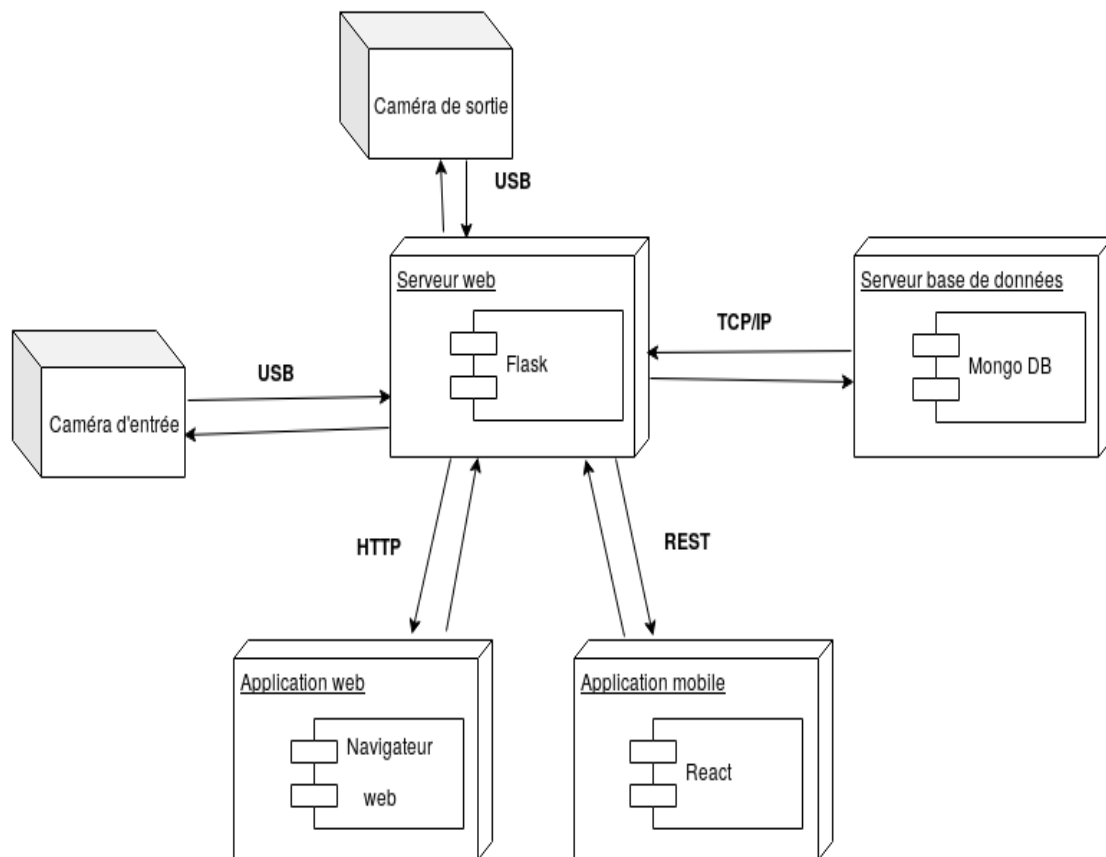


Figure 4.18: Diagramme de déploiement

La figure ci-dessus montre une vue détaillée de l'architecture finale de Pointini.

## Conclusion

À ce stade nous avons développé une bonne partie de nos applications web et mobile pour arriver à un produit complet et fonctionnel. Ce chapitre avait pour objectif de faire l'analyse, la conception et la réalisation de nos deux derniers sprints.



## Conclusion Générale & Perspectives

Ce présent rapport a pour objet de synthétiser le travail réalisé dans le cadre du projet de fin d'études au sein de l'ISITCom Hammam Sousse, et d'un stage effectué au sein d'Astrolab.

Il a porté sur la conception et d'un système de pointage basé sur la reconnaissance faciale : Pointini.

Nous avons commencé dans un premier lieu par comprendre le contexte général de notre application et identifier les différentes exigences de notre futur système en étudiant l'existant. Nous avons préparé par la suite notre planning de travail en construisant le cahier des charges à l'aide de l'avis du PO.

Pour mettre en place notre application, nous avons utilisé et pour la première fois Scrum comme cadre de processus de gestion de projet. Malgré toutes les difficultés rencontrées au niveau de la réalisation du processus métier de l'application et les contraintes de temps, nous avons réussi à convertir notre Product Backlog en un ensemble d'incréments potentiellement livrable jusqu'à l'aboutissement au produit final bien intégré. Nous avons entre les mains la première release de Pointni bien positionnée par rapport à des produits concurrents.

En guise de perspective ce travail reste prêt pour toute amélioration envisageable : Ajouter une partie de reconnaissance vocale pour éviter les problèmes causés par l'environnement (éclairage, expression du visage ) ou par le changement (barbe, moustache, chirurgie...).

En terme de conclusion, nous tiendrons à souligner l'importance et la richesse de cette expérience académique et professionnelle, mais surtout personnelle puisqu'elle nous a ouvert les yeux sur le monde professionnel, sur nos forces et nos faiblesses, et sur nos futures ambitions.

# Glossaire

**IA** = Intelligence Artificielle

**UI** = User Interface (Interface pour l'utilisateur)

**UX** = User eXperience (Expérience de l'utilisateur)

**DevTeam** = Équipe de développement

**SM** = Scrum master

**UML** = Unified Modeling Language (Langage de Modélisation Unifié)

**PO** = Product Owner

**DoD** = Definition of Done (Définition de « fini »)

**PB** = Product Backlog (Backlog produit)

**ResNet** = Residual Network (Réseau résiduel)

**CNN** = convolutional neural network (réseau de neurones convolutifs)

## Bibliographie et Netographie

- [1] *Hédia Jegham, Cours Apprentissage de la méthode scrum, Chapitre 3, ISITCom 2018.*
- [2] *Chihéb S'hili, About Astrolab FR, Astrolab 2019.*
- [3] <http://www.isi.rnu.tn/BIBI/ing/P%20ING%20490>, Consulté le 17 mai 2019.
- [4] <https://agiliste.fr/exemple-dorganisation-projet-agile/>, Consulté le 15 mai 2019.
- [5] <https://www.horloges-huchez.fr/blog/feuille-pointage>, Consulté le 18 mai 2019.
- [6] <http://www.linternaute.com/science/biologie/dossiers/06/0607-biometrie/empreinte-digitale.shtml>, Consulté le 18 mai 2019.
- [7] <https://scrumguides.org/docs/scrumguide/v2017/2017-scrum-Guide-French.pdf>, Consulté le 18 mai 2019.
- [8] <http://dlib.net/>, Consulté le 24 mai 2019.
- [9] <https://www.pobot.org/Decouverte-d-OpenCV.html>, Consulté le 20 mai 2019.
- [10] <https://github.com/davisking/dlib-models>, Consulté le 24 mai 2019.
- [11] <https://whatis.techtarget.com/fr/definition/Bootstrap>, Consulté le 01 juin 2019.
- [12] <https://facebook.github.io/react-native/>, Consulté le 01 juin 2019.
- [13] [www.json.org/json-fr.html](http://www.json.org/json-fr.html), Consulté le 01 juin 2019.
- [14] <https://www.javascript.com>, Consulté le 01 juin 2019.
- [15] <https://jquery.com/>, Consulté le 01 juin 2019.