

Rapport de stage de fin d'étude en vue de l'obtention du diplôme de brevet de technicien supérieur

Conception et réalisation d'une application de suivi des
absences avec option de reconnaissance faciale

Réalisé par :

El QACIMI Fatima Ezzahrae

ZARTOUNI Salma

Encadré par :

M.ELFID Zaid

Année académique : 2025-2026

Dédicace

Ce projet est le reflet d'un parcours marqué par le soutien indéfectible et l'amour inconditionnel de mes parents. Leur patience, leurs sacrifices et leur foi constante en mes capacités ont été essentiels pour mener à bien ce travail. Les encouragements de ma mère et les précieux conseils de mon père ont été des repères solides qui m'ont permis de persévérer, même dans les moments de doute.

Les liens familiaux ont également joué un rôle primordial : mes frères et sœurs ont été une source de réconfort, de sourires et de complicité à chaque étape. Ma famille élargie a apporté sa présence rassurante, ses prières et cette fierté partagée qui m'a motivé(e) à aller toujours plus loin.

Les encouragements et la bienveillance de mes amis ont rendu cette aventure plus légère et plus agréable, apportant des moments de rires et de partage indispensables à l'équilibre de ce parcours.

Les conseils avisés et l'accompagnement précieux de mes enseignants et encadrants ont contribué à la concrétisation de ce travail, en renforçant ma confiance et en clarifiant mes orientations.

Enfin, ce projet est l'aboutissement d'un soutien collectif : celui de tous ceux qui, de près ou de loin, ont apporté leur inspiration, leur foi en moi et leurs encouragements. Il est le fruit de cette dynamique familiale, amicale et académique qui a rendu possible sa réalisation.

Remerciement

La conduite et la réalisation de ce projet de fin d'études ont été rendues possibles grâce à l'aide, au soutien et à l'accompagnement de plusieurs personnes. Il serait difficile de toutes les nommer, mais qu'elles trouvent ici l'expression de ma profonde gratitude.

Je tiens à exprimer mes sincères remerciements à **M. Zaid El Feid**, mon encadrant pédagogique, pour sa bienveillance, sa disponibilité et la qualité de son accompagnement. Grâce à ses orientations claires, ses remarques constructives et sa rigueur académique, j'ai pu structurer ce projet avec méthode et sérieux. Ses conseils avisés et sa confiance ont été des leviers essentiels dans la réussite de ce travail.

Je remercie également **l'ensemble des professeurs** qui m'ont accompagné durant ma formation. Chacun d'eux, par ses enseignements, son exigence et sa passion pour la transmission du savoir, a contribué à mon développement académique et personnel. Leur engagement constant m'a permis d'acquérir les compétences.

Mes remerciements vont aussi à **l'équipe administrative** de l'établissement pour leur soutien, leur réactivité et leur disponibilité. Leur gestion efficace et leur accompagnement quotidien ont fortement contribué à créer un environnement propice au travail et à l'épanouissement.

Enfin, je remercie les membres du jury pour l'attention portée à ce travail. J'espère que ce rapport saura refléter l'implication, la rigueur et la motivation investies tout au long de ce projet de fin d'études.

Table des matières

Table des figures

Liste des tableaux

Tableaux des abréviations

Abréviation	Signification
OCP	Office Chérifien des Phosphates
BTS	Brevet de Technicien Supérieur
SGBD	Système de Gestion de Base de Données
UML	Unified Modeling Language
GANTT	Diagramme de Gantt (outil de planification)
PERT	Program Evaluation and Review Technique
GUI	Graphical User Interface (Interface Graphique)
API	Application Programming Interface
DB	Database (Base de données)
SQLite	Structured Query Language Lite (SGBD léger)
CV	Curriculum Vitae
ID	Identifiant
UI	User Interface (Interface utilisateur)
OCR	Optical Character Recognition
AI	Artificial Intelligence
ML	Machine Learning
RAM	Random Access Memory
CPU	Central Processing Unit

Introduction Générale

Dans le cadre de notre formation de brevet de technicien supérieur, nous avons été amenées à réaliser un projet de fin d'études visant à mettre en pratique les compétences acquises tout au long de notre cursus. Ce projet, intitulé « Conception et réalisation d'une application de suivi des absences avec option de reconnaissance faciale », s'inscrit dans une dynamique de modernisation des processus de gestion et de suivi des absences, un enjeu crucial pour les établissements éducatifs et les entreprises.

Le choix de ce sujet s'explique par l'importance croissante des solutions basées sur l'intelligence artificielle et la reconnaissance faciale dans l'optimisation des tâches administratives. Grâce à ce projet, nous avons pu allier théorie et pratique en développant une application performante et innovante, qui répond à des besoins réels et concrets.

Ce rapport présente les différentes étapes de réalisation du projet, de l'étude du besoin jusqu'à la mise en œuvre technique, en passant par la conception et les tests. Il met également en lumière les compétences mobilisées, les difficultés rencontrées et les solutions apportées, offrant ainsi un aperçu complet et structuré de notre travail.

Gestion du projet

1. Cahier des charges

Sachant qu'un système d'information se constitue de deux aspects, **dynamique et statique**, l'informatisation consistera à la réalisation d'une **base de données centralisée**, contenant toutes les informations nécessaires et interagissant avec une application **web intelligente, moderne et bien structurée**. Ainsi, notre projet de fin d'études « **Gestion des absences par IA** » a pour objectif de développer une application en **Python** avec **Streamlit** pour la gestion des absences des étudiants, intégrant des fonctionnalités avancées de **reconnaissance faciale** grâce à l'intelligence artificielle. La base de données est conçue en **SQLite** pour centraliser et sécuriser toutes les données liées aux absences, aux justificatifs et aux indicateurs d'alerte, tandis que l'interaction entre l'application et la base est assurée directement via des requêtes optimisées en Python.

Cette application doit prendre en compte les trois profils d'utilisateurs (administrateurs, enseignants et étudiants), chacun ayant des droits spécifiques.

Chaque **administrateur** a le droit de :

- Affecter des classes aux enseignants
- Ajouter, modifier, supprimer un enseignant
- Ajouter, modifier, supprimer une classe
- Ajouter, modifier, supprimer un étudiant
- Consulter les absences des étudiants
- Valider ou rejeter les justificatifs d'absence
- Activer ou désactiver la reconnaissance faciale
- Envoyer des alertes en cas d'absences répétées

Chaque **enseignant** a le droit de :

- Enregistrer les absents et les présents pendant ses séances
- Marquer les étudiants comme absents, exclus ou ayant quitté la salle
- Consulter les indicateurs d'absence de ses cours

Chaque **étudiant** a le droit de :

- Consulter ses absences avec des indicateurs visuels (vert, jaune, rouge)
- Soumettre des justificatifs d'absence (images ou PDF)
- Suivre l'état de validation de chaque justificatif
- Consulter son emploi du temps
- Enregistrer automatiquement sa présence via la reconnaissance faciale si activée par l'administration .

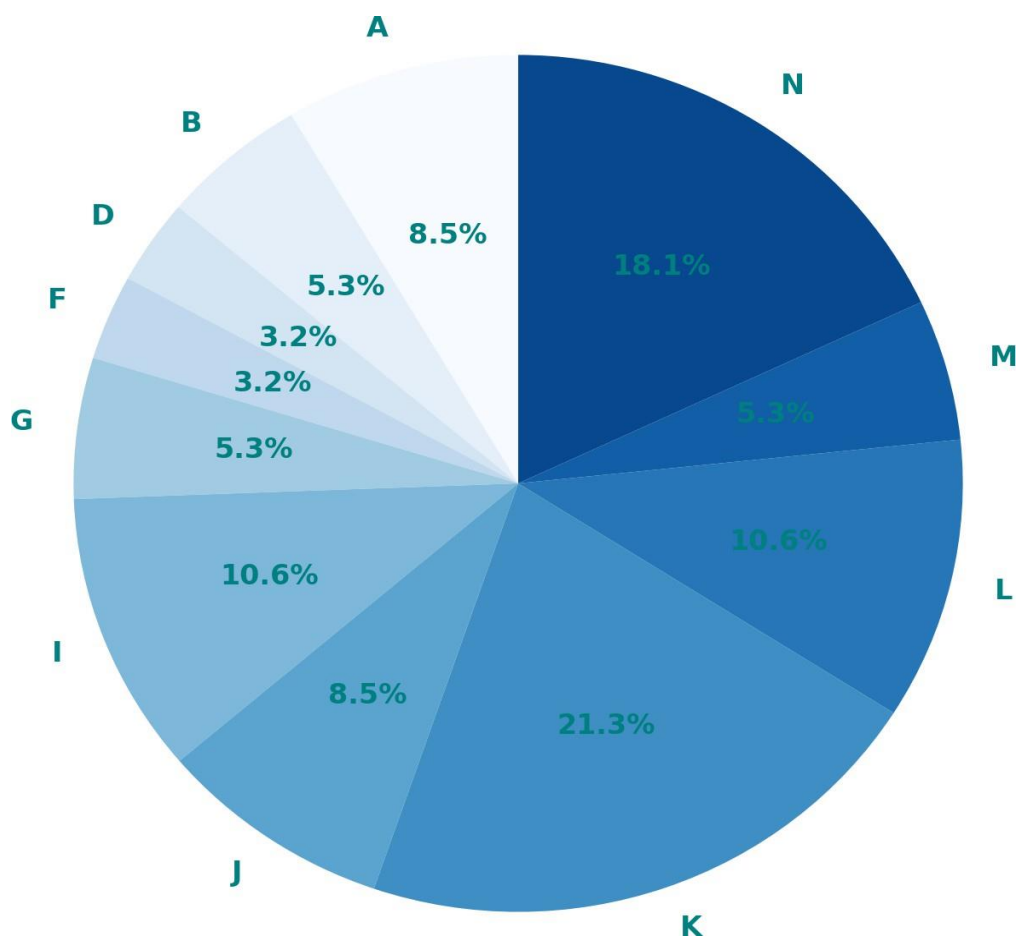
Cette solution innovante, intégrant l'intelligence artificielle, garantit une **gestion efficace, fluide et sécurisée des absences**, répondant ainsi aux besoins actuels des établissements scolaires.

2. Planification de Projet

La liste des tâches de notre projet :

	Tâches	description	Durée	Tâches antérieures
	A	Choix de projet	8	–
	B	Rédaction d'un cahier des charges	5	A
Conception	C	Choix de méthodologies de Conception	4	B
	D	Diagramme des cas d'utilisation	3	C
	E	Diagramme de classe	3	C
	F	Le diagramme de séquence	3	C
Réalisation	G	Conception de la base de données	5	A
	H	Développement des interfaces	3	G
	I	Développement de la logique métier	10	H
	J	Intégration de la caméra	8	I
	K	Développement de la gestion des absences	20	I
	L	Intégration et finalisation des modules	10	J, K
Test	M	Tester l'application	5	L
Rapport	N	Rédiger le rapport	17	–

Le diagramme ci-dessous illustre la répartition du temps attribué à chacune des grandes étapes du projet. C'est une conclusion graphique résultant de notre gestion de projet.

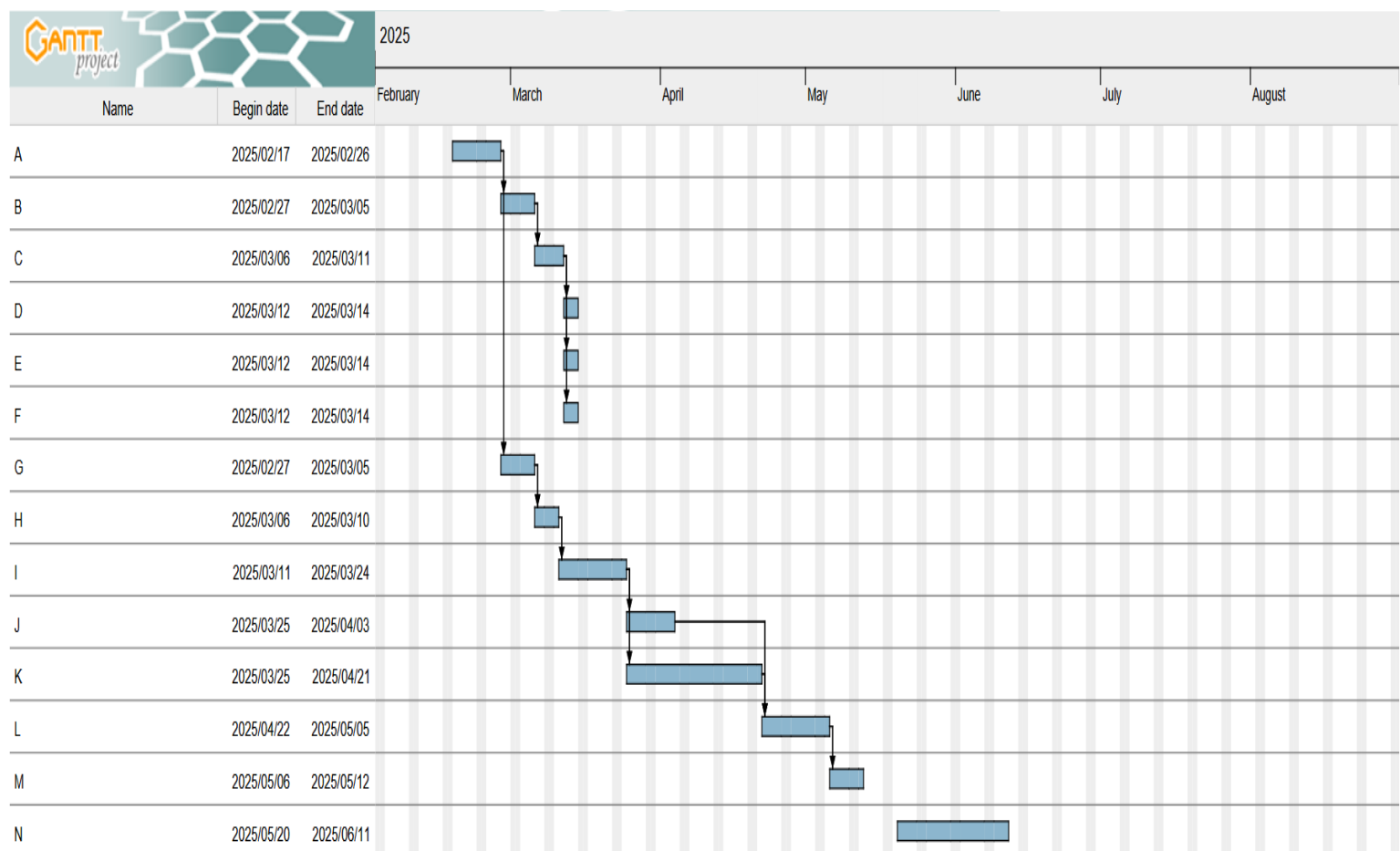


Pour assurer une gestion efficace et optimale de notre projet, nous avons mis en place plusieurs méthodes et pratiques structurées. Ces méthodes visent à garantir le respect des délais, la qualité des livrables et la coordination harmonieuse de l'équipe. Nous avons notamment utilisé des techniques telles que la planification des tâches (diagramme de Gantt et PERT), le suivi rigoureux des délais et des ressources, ainsi que l'évaluation régulière des progrès pour anticiper les risques et les ajustements nécessaires.

3. Diagramme de GANTT

Le diagramme de Gantt est un outil performant qui s'appuie sur des données brutes telles que les dates de début et de fin, ainsi que la durée de chaque tâche, pour offrir une visualisation claire de l'avancement d'un projet. Il fournit une vue d'ensemble des tâches à accomplir, des responsabilités assignées et des ressources mobilisées, depuis l'idée initiale jusqu'à la mise en service. Il couvre les étapes clés comme l'analyse des exigences, l'étude de faisabilité, la conception fonctionnelle, les spécialisations, la réalisation et les tests finaux. En outre, le diagramme de Gantt facilite la gestion des ressources, leur disponibilité et leurs coûts, assurant ainsi un pilotage optimal du projet.

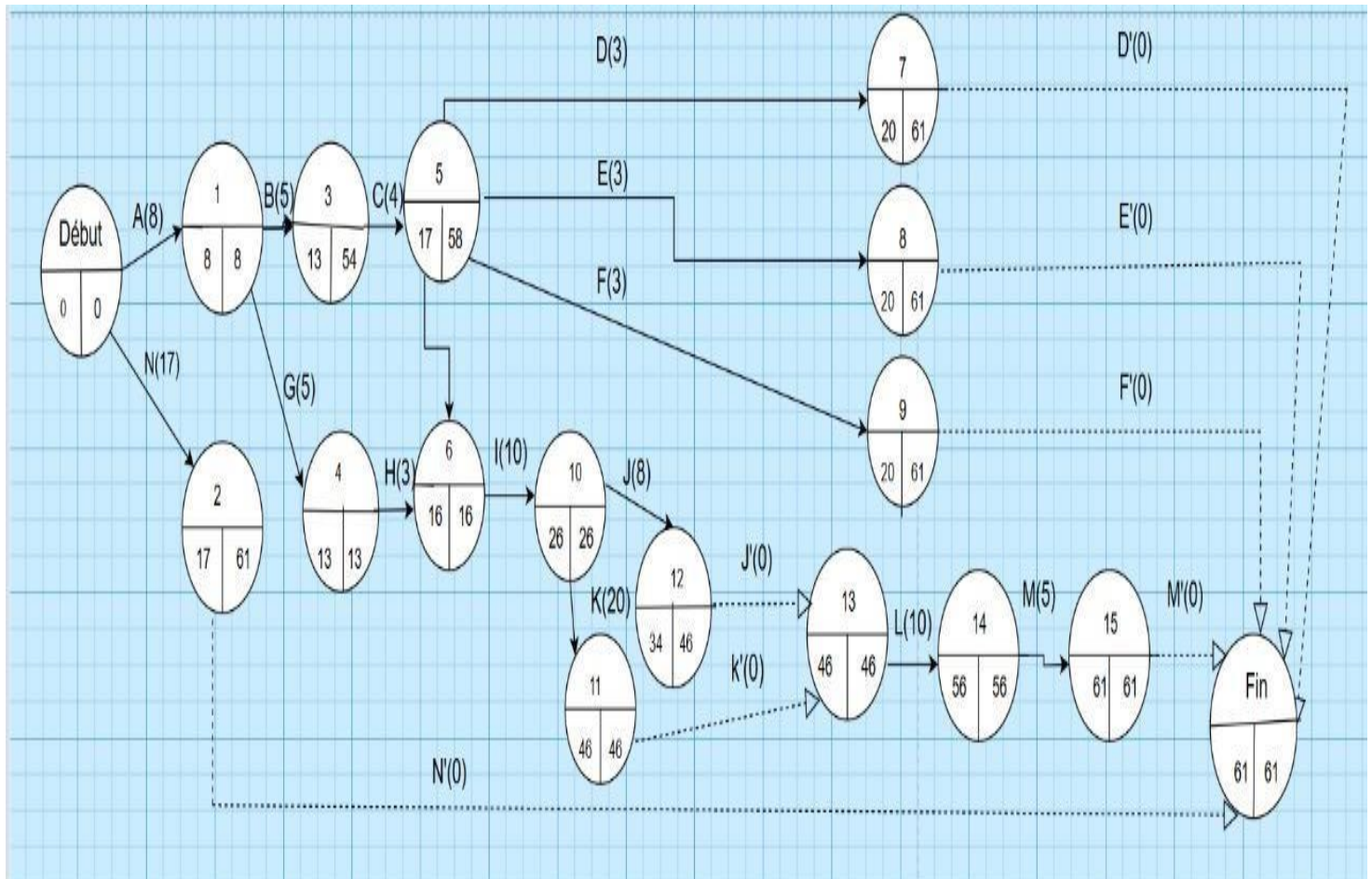
Notre diagramme de GANTT :



4. Diagramme de PERT

Associé au diagramme GANTT, le diagramme PERT permet de mettre en évidence les liaisons entre les différentes tâches afin d'en déduire un chemin dit « critique ». L'état de chacune des tâches évolue ensuite en fonction de la progression du projet et met en évidence la succession des livrables. Pour une question de lisibilité, nous avons seulement détaillé ci-dessous une partie du diagramme PERT à travers la hiérarchie de deux des phases de notre projet.

Notre diagramme de PERT :



Analyse et Conception

1. Introduction

Le Modèle conceptuel de données est une représentation statique du système d'information. Il a comme objectif de constituer une représentation claire et cohérente des données manipulées dans le système d'information. Cette section, sera présentée comme suit : nous commençons par le choix de la méthodologie de conception et justification. Ensuite nous identifions les acteurs et les diagrammes des cas d'utilisation, puis nous présentons le diagramme de classe, diagramme de collaboration et enfin les diagrammes d'état transition.

2. Choix de la méthode de conception

Dans le cadre de notre projet, nous avons adopté le langage UML comme approche de conception. UML, ou Unified Modeling Language, est un langage graphique universel de modélisation objet. Il a été conçu pour représenter les systèmes complexes de manière standardisée et indépendante des langages de programmation. Grâce à sa richesse et à sa souplesse, UML permet de modéliser les différentes facettes d'un système, de la description des besoins fonctionnels à la représentation dynamique et structurelle.

Dans cette section, nous allons tout d'abord présenter brièvement le langage UML en mettant en avant ses principaux atouts : la clarté des représentations, la standardisation des notations et la capacité à communiquer efficacement les choix de conception. Ensuite, nous détaillerons les différents diagrammes que nous avons utilisés pour illustrer et formaliser notre projet : diagrammes de cas d'utilisation, diagrammes de classes, diagrammes de séquence, diagrammes de collaboration, et diagrammes d'états-transitions. Chacun de ces diagrammes sera expliqué et justifié en fonction de sa contribution à la compréhension et à la réalisation de notre solution.

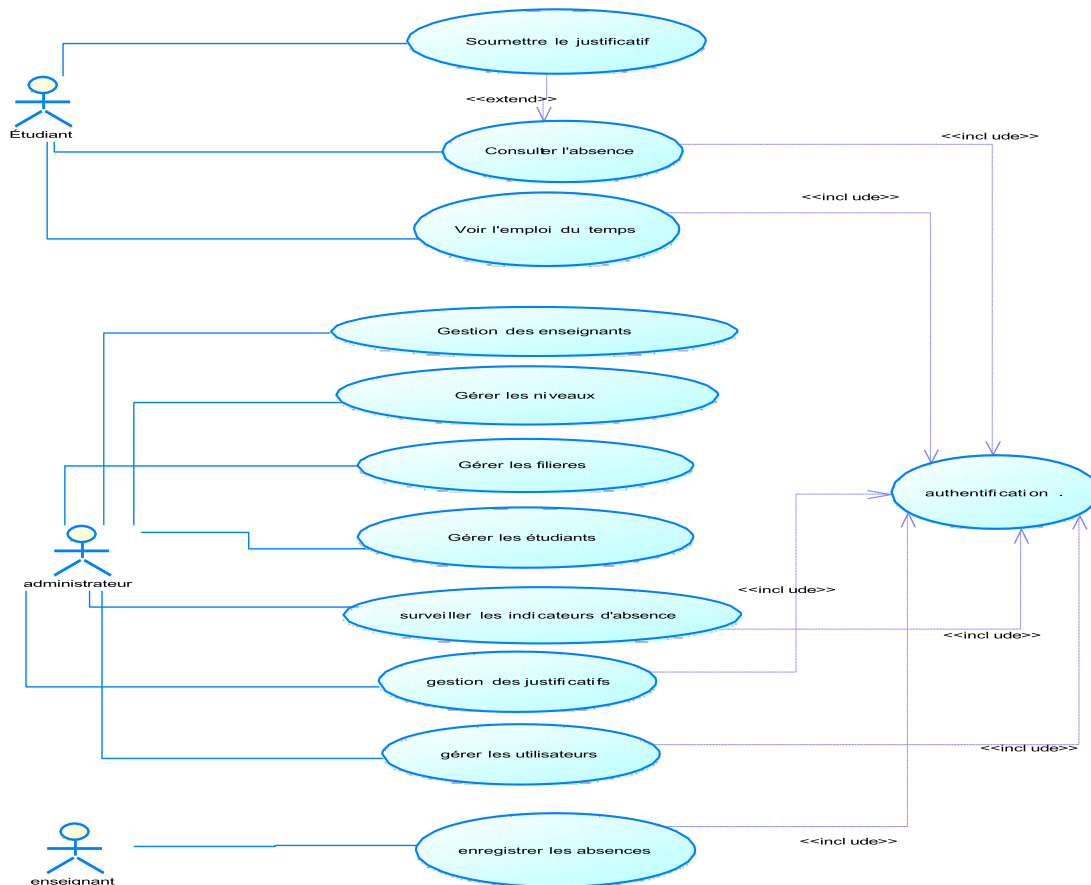
3. Présentation d'UML

UML (Unified Modeling Language) est un langage formel et normalisé dédié à la modélisation objet. Son indépendance vis-à-vis des langages de programmation, des domaines d'application et des processus, ainsi que sa polyvalence et sa souplesse, en font un langage universel. De plus, UML constitue avant tout un support de communication efficace, facilitant la représentation et la compréhension des solutions orientées objet. Grâce à sa notation graphique, il permet d'exprimer visuellement une solution, ce qui facilite la comparaison et l'évaluation des alternatives. Cette notation contribue à réduire l'ambiguïté et les risques d'incompréhension. Enfin, UML offre un moyen ingénieux de représenter différentes vues d'une même modélisation, permettant ainsi d'obtenir une vision complète et cohérente du système.

4. Diagramme de cas d'utilisation

Les cas d'utilisation décrivent un ensemble d'actions réalisées par le système, en réponse à une action d'un acteur.

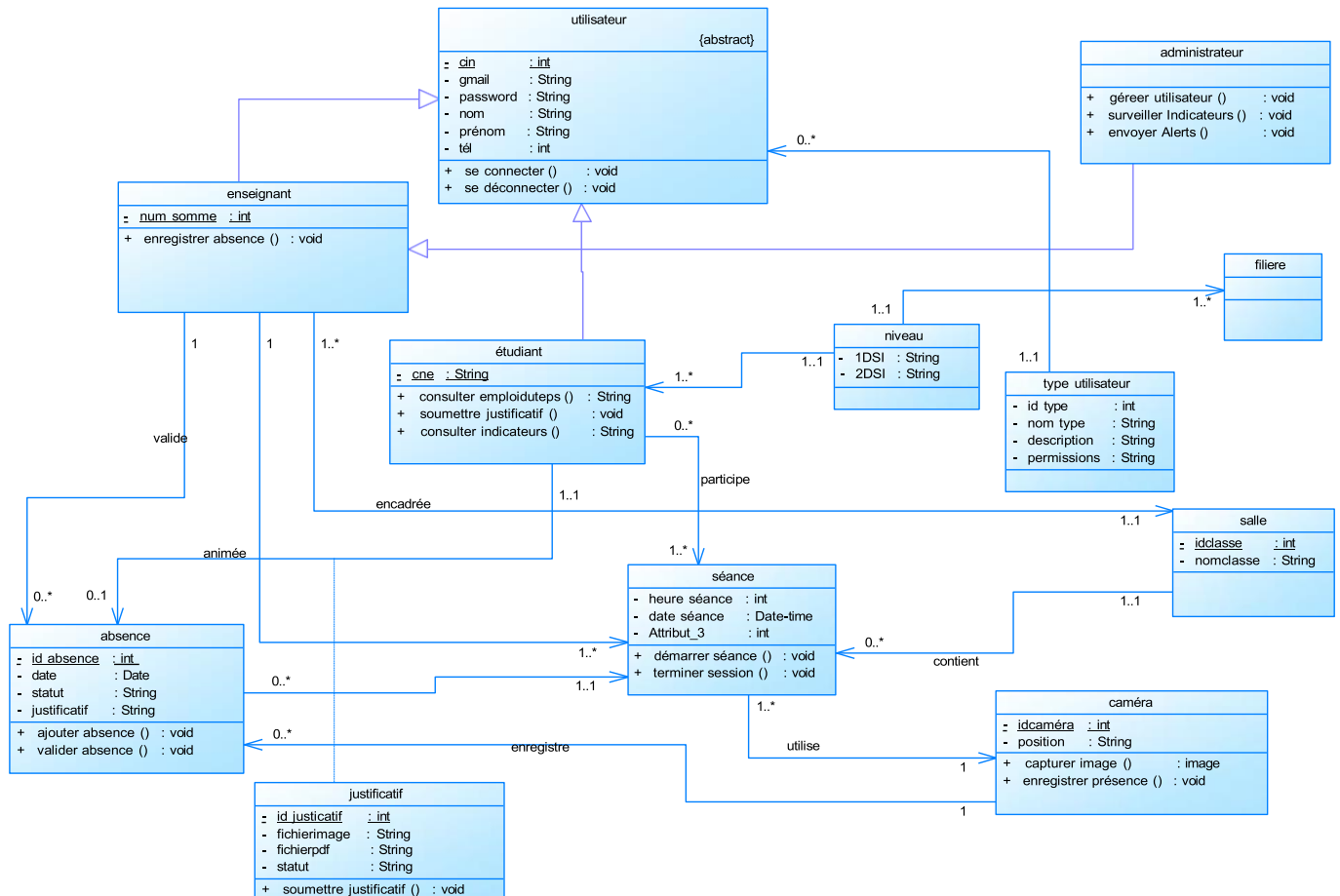
Notre diagramme de cas d'utilisation :



5. Diagramme de classe

Le diagramme de classes est un diagramme statique de l'UML qui illustre la structure d'un système en représentant les classes, leurs attributs, leurs opérations et les relations qui les unissent. Il constitue la base de la modélisation orientée objet et facilite la compréhension de l'architecture du système.

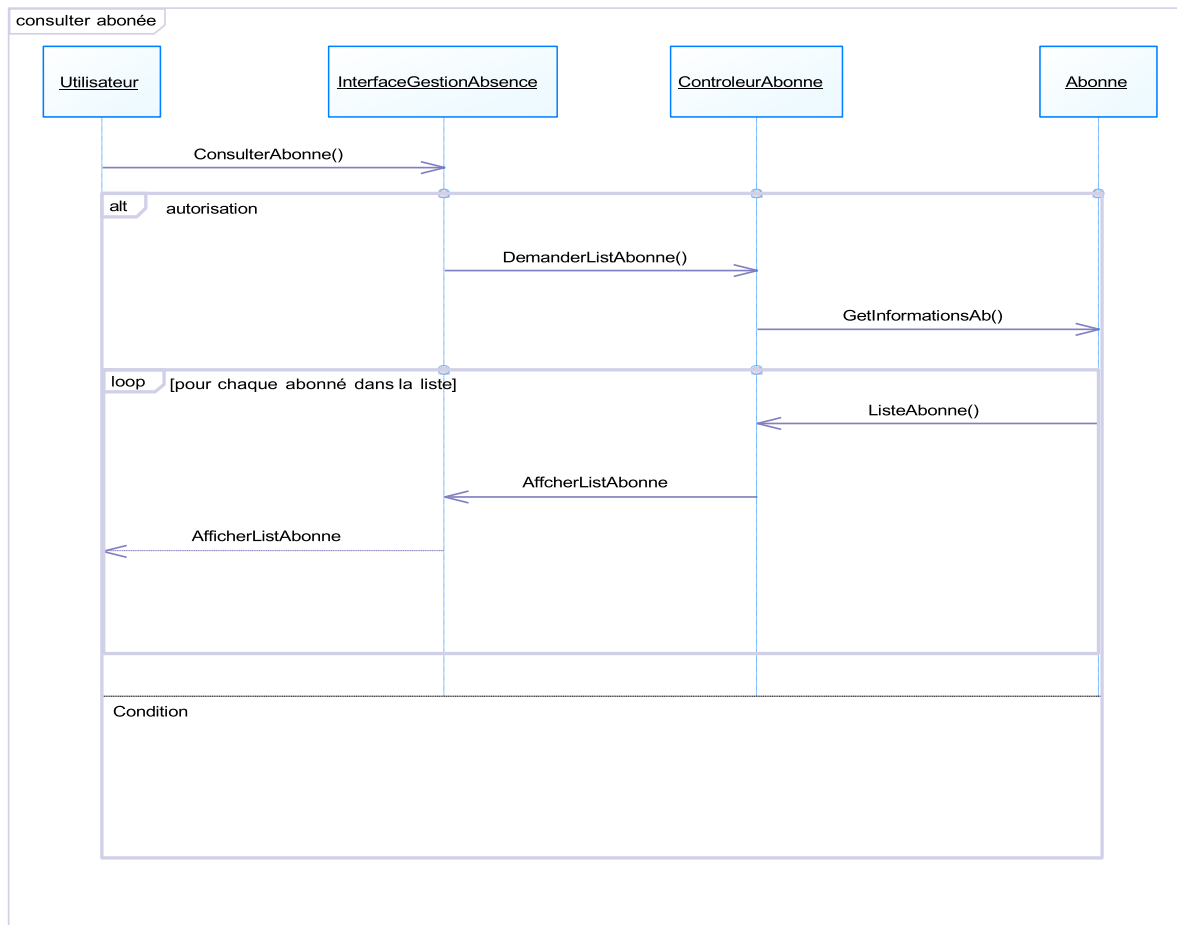
Notre diagramme de classes :



6. Diagramme de Séquence

Diagramme de séquence permettent de représenter des collaborations eu objets selon un point de vue temporel, on y met l'accent sur la chronologie (envois de messages).

Notre diagramme de séquence d'insertion :



Conclusion

Ce chapitre met en lumière l'importance d'une bonne architecture de données pour le succès du projet, en préparant le terrain pour les prochaines étapes de développement et de déploiement de l'application.

Technologies utilisées

1. Les langages de programmation



Python : est le langage principal utilisé pour le développement de l'application de gestion des absences avec reconnaissance faciale. Grâce à sa polyvalence et à la richesse de ses bibliothèques spécialisées, Python a permis d'implémenter l'ensemble de la logique métier de l'application, notamment la gestion des utilisateurs, la validation des absences et l'authentification sécurisée. Il a également été utilisé pour intégrer les fonctionnalités avancées de la reconnaissance faciale, en s'appuyant sur les bibliothèques **OpenCV** et **face_recognition**. Cette approche offre une solution fiable, performante et évolutive.



SQL/SQLite : La gestion des données au sein de l'application s'appuie sur l'utilisation du langage **SQL**, associé au moteur de base de données **SQLite**. SQL a été employé pour la création des différentes tables (utilisateurs, absences, justificatifs, sessions, caméras, etc.) ainsi que pour la réalisation des requêtes de manipulation des données (création, lecture, mise à jour et suppression). Grâce à SQLite, qui constitue une solution légère et intégrée, l'application bénéficie d'une gestion des données efficace et cohérente.



HTML/CSS : Pour la partie interface web de l'application (lorsqu'elle n'est pas directement assurée par Streamlit), les technologies **HTML** et **CSS** ont été utilisées. HTML a permis de structurer les différentes pages (formulaire de connexion, tableaux des absences, boutons de validation), tandis que CSS a été employé pour en améliorer l'aspect visuel. Cette combinaison garantit une présentation soignée et ergonomique pour les utilisateurs.



JavaScript : pour enrichir l'expérience utilisateur avec des fonctionnalités dynamiques, le langage **JavaScript** a été intégré. Il a permis de gérer des interactions en temps réel, comme la mise à jour automatique des tableaux de données, l'affichage de notifications ou encore l'animation d'éléments de l'interface. JavaScript complète ainsi l'interface web pour offrir une navigation fluide et interactive

2. Les logiciels des programmations



Visual Studio Code (VS Code) : Cet éditeur de code open-source a été utilisé pour écrire et organiser l'ensemble du code source (Python, SQL et parfois HTML/CSS). Il offre de nombreuses extensions facilitant le développement et la gestion des dépendances.



Python (via Anaconda ou installation directe) : Python est le langage principal du projet. L'environnement Python installé localement (via Anaconda ou directement sur l'ordinateur) a permis de gérer les bibliothèques nécessaires, telles que OpenCV, face_recognition et pandas.



SQLite Browser : Pour la création et la gestion de la base de données relationnelle (tables utilisateurs, absences, etc.), l'outil DB Browser for SQLite a été utilisé. Il offre une interface simple et intuitive pour visualiser les données.



Streamlit : Ce logiciel a été utilisé pour la création de l'interface web de l'application. Il permet de transformer rapidement un script Python en une application web interactive et professionnelle.



Git (et GitHub) : Pour la gestion du versionnage du projet, Git a été employé. Cet outil a permis de sauvegarder et de suivre l'évolution du code tout au long du projet.



GanttProject : est un logiciel libre de gestion de projet écrit en Java, ce qui permet de l'utiliser sur divers systèmes d'exploitation. Il permet d'éditer un diagramme de Gantt (<https://fr.wikipedia.org/wiki/GanttProject>).



PowerAMC

Sybase : est un éditeur de logiciels fondé en 1984. Sybase fournit des solutions d'infrastructure d'entreprise, des

solutions de mobilités de logiciel pour la gestion de l'information.

3. la justification de notre choix et les avantages de langages utiliser

PYTHON

Python est un langage de programmation à la fois puissant, simple et lisible, qui facilite la prise en main et le développement rapide de solutions logicielles. Il offre de nombreuses bibliothèques spécialisées, notamment **OpenCV** pour le traitement des images et des vidéos, **face_recognition** pour la détection et la reconnaissance faciale, ainsi que **pandas** pour la manipulation des données sous forme de tableaux. Cette richesse de l'écosystème Python en fait un choix privilégié pour les projets qui nécessitent des calculs avancés ou des analyses en temps réel. De plus, la popularité de Python et l'ampleur de sa communauté garantissent un accès rapide à des ressources de qualité, des tutoriels, ainsi qu'un soutien permanent pour résoudre les problèmes techniques et améliorer continuellement le projet.

SQL / SQLITE

Le langage **SQL** (Structured Query Language) est un standard reconnu pour l'interrogation et la manipulation des bases de données relationnelles. Il permet de réaliser des opérations telles que la création de tables, l'ajout de données, la mise à jour et la suppression d'enregistrements via des requêtes simples et précises. Dans ce projet, l'utilisation de **SQLite** comme moteur de base de données a été un choix judicieux, car il offre une solution intégrée, légère et facile à déployer. SQLite ne nécessite aucun serveur dédié, ce qui simplifie grandement la configuration et la gestion du projet. Cette combinaison de SQL et SQLite permet d'assurer une excellente cohérence des données, une sécurité renforcée et une rapidité d'accès, même dans des environnements de travail contraints.

STREAMLIT

Streamlit est un framework moderne et puissant, entièrement basé sur Python, qui permet de transformer facilement des scripts Python en applications web interactives. Son principal atout réside dans sa simplicité : en quelques lignes de code seulement, il est possible de créer des tableaux de bord dynamiques, des formulaires d'inscription ou des pages de consultation des données. Grâce à Streamlit, il est aussi possible de visualiser des graphiques en temps réel (par exemple, l'évolution des absences ou la participation aux formations) en utilisant des bibliothèques Python comme **matplotlib** ou **altair**. De plus, Streamlit bénéficie d'une grande popularité et d'un développement actif, garantissant des mises à jour régulières et des fonctionnalités toujours à la pointe des besoins modernes. Son intégration transparente avec les autres bibliothèques Python en fait un outil idéal pour créer une expérience utilisateur intuitive, professionnelle et agréable.

Conclusion Générale et Perspectives

Au terme de ce projet de fin d'études, nous avons pu concevoir et réaliser une application innovante de gestion des absences avec option de reconnaissance faciale, répondant aux besoins réels d'automatisation et de fiabilisation des processus administratifs. Ce projet nous a permis de mobiliser l'ensemble des compétences techniques et méthodologiques acquises durant notre formation, tout en développant notre sens de l'organisation et notre esprit d'équipe.

La mise en œuvre de technologies avancées, telles que l'intelligence artificielle et la reconnaissance faciale, a été un défi stimulant et formateur. Nous avons pu constater l'apport concret de ces outils pour la simplification des tâches et la précision des données, tout en restant attentifs aux enjeux éthiques liés à leur utilisation, notamment en matière de confidentialité et de sécurité des données personnelles.

Au-delà de l'aspect purement technique, ce projet a renforcé notre capacité à :

Analyser les besoins et formuler des solutions adaptées.

Concevoir une architecture logicielle robuste et évolutive.

Travailler en équipe en adoptant une démarche collaborative et rigoureuse.

S'adapter aux contraintes et aux évolutions rapides des technologies.

Ce PFE a été une étape déterminante dans notre parcours, consolidant notre envie de poursuivre dans le domaine des technologies émergentes et de l'intelligence artificielle.

Perspectives

Pour prolonger et enrichir ce projet, plusieurs axes d'amélioration et de développement pourraient être envisagés :

Renforcer la sécurité de l'application, en intégrant des protocoles de chiffrement avancés pour la protection des données biométriques.

Améliorer les performances de la reconnaissance faciale, notamment dans des environnements variés ou à faible luminosité.

Étendre les fonctionnalités de l'application, par exemple en y intégrant des modules de statistiques et de rapports automatiques pour les administrateurs.

Adapter l'application à d'autres secteurs, tels que la gestion des présences en entreprise, les événements ou les systèmes de contrôle d'accès.

Développer une version mobile pour une utilisation encore plus souple et intuitive.

Enfin, nous sommes convaincus que la convergence de l'IA et de la gestion intelligente des données constitue un levier majeur pour répondre aux besoins actuels et futurs des entreprises et des institutions. Ce projet n'est donc pas un aboutissement en soi, mais bien un point de départ pour explorer de nouvelles pistes et participer activement aux avancées de la transformation numérique.

Références

Bibliographie

- [1] Kimball, R. (2001). *Concevoir et déployer un Data Warehouse*. Éditions Eyrolles.
- [2] FERNANDEZ, A. (2014). *GMSI : Le projet de business intelligence clé en main*. Dunod.
- [3] Coffing, T. (2001). *Tera-Tom on Teradata Basics*. Coffing Data Warehousing.
- [4] OCP Groupe (2024). *Documentation interne sur la plateforme Formation Manager*.
- [5] OCP Groupe (2024). *Présentation de l'organigramme et des processus RH*.

Webographie

- [W1] <https://www.ocpgroup.ma/fr/media/actualites/organigramme-ocp-2024>
- [W2] <https://www.informatica.com/fr-fr/solutions/data-integration.html>
- [W3] <https://business-intelligence.developpez.com/actu/311154/Concepte-d-entrepot>
- [W4] <https://www.oracle.com/fr/business-analytics/>