

MINISTERE DE LA PLANIFICATION DU
DEVELOPPEMENT ET DE LE
COOPERATION



Institut Africain d'Informatique

Représentation du TOGO

Tel : 0022822204700

Email : iaitogo@iai-togo.tg

Site web : www.iai-togo.tg

07 BP 12456 Lomé 07, TOGO.



Tel : 90885976 / 98988818

Email : mivo.energie@entrepreneursdumonde.org

Site web : <https://mivoenergie.org/>

BP 14311 Tokoin Kodomé Lomé,
TOGO.

**PROJET DE FIN DE FORMATION POUR L'OBTENTION DU DIPLOME DE
LICENCE EN INFORMATIQUE**

OPTION : GENIE LOGICIEL & SYSTEME D'INFORMATION

**MISE EN PLACE D'UN SERVICE DE GESTION DE
CONGES ET D'EVALUATION DES PERFORMANCES
DU PERSONNEL D'UNE ENTREPRISE :
CAS DE MIVO ENERGIE**

Période : Du 28 Juin au 17 Septembre 2021

Rédigé et soutenu par :

AYITE Yawo Mawulom Guillaume Florian

Etudiant en troisième année GLSI

Année Scolaire : 2020-2021

SUPERVISEUR

M. EKLOU Kader

Chargé de cours à l'IAI-TOGO.

MAITRE DE STAGE

M. NIMON B. Jean-Pierre

Responsable
administratif et financier

DEDICACES

A ma famille pour son soutien, ses conseils et ses encouragements.

A tous mes professeurs.

Que Dieu vous bénisse.

REMERCIEMENTS

Je tiens tout d'abord à remercier DIEU Tout Puissant pour la vie et la santé qu'il m'a accordé, ainsi qu'à mes proches.

J'adresse mes remerciements à mes parents pour leurs conseils, soutiens et investissements et d'avoir permis ma formation à l'IAI-TOGO.

J'aimerais aussi adresser mes sincères remerciements aux personnes suivantes :

- ❖ M. AGBETI Kodjo, Représentant résidant de l'IAI-TOGO, pour son accompagnement tout le long de notre formation et pour les efforts qu'il ne cesse de déployer pour l'Institut,
- ❖ M. AMEYIKPO Kossi, Directeur des Affaires Académiques et de la Scolarité de l'IAI-TOGO, pour l'encadrement et les conseils fournis,
- ❖ M. DAVON Essé Enseignant à l'IAI-TOGO pour son accompagnement et ses conseils tout le long de notre stage,
- ❖ M. TETE Kosité Senam Enseignant à l'IAI-TOGO pour ses conseils et ses apports,
- ❖ M. MOUTORE Yempabe, Directeur exécutif de l'ONG Mivo Energie,
- ❖ M. EKLOU Kader, Enseignant à l'IAI-TOGO et superviseur de notre travail pour ses conseils, apports, critiques et son encadrement,
- ❖ M. NIMON Jean-Pierre, Responsable des Affaires Financières de l'ONG Mivo Energie,
- ❖ Mes camarades de promotion pour leur soutien, aide et divers apports,
- ❖ Tous ceux qui de près ou de loin ont œuvré au bon déroulement de ce stage.

SOMMAIRE

DEDICACES.....	<i>i</i>
REMERCIEMENTS.....	<i>ii</i>
AVANT-PROPOS.....	<i>iv</i>
RESUME.....	<i>v</i>
ABSTRACT.....	<i>vi</i>
GLOSSAIRE.....	<i>vii</i>
LISTE DES FIGURES.....	<i>viii</i>
LISTE DES TABLEAUX.....	<i>ix</i>
LISTE DES PARTICIPANTS AU PROJET.....	<i>x</i>
INTRODUCTION.....	<i>1</i>
PARTIE 1 : CAHIER DES CHARGES.....	<i>2</i>
1.1 Présentations	<i>3</i>
1.2 Thème de stage	<i>8</i>
1.3 Etude de l'existant.....	<i>11</i>
1.4 Critique de l'existant.....	<i>13</i>
1.5 Propositions et choix de solutions.....	<i>14</i>
1.6 Planning prévisionnel de réalisation	<i>23</i>
PARTIE 2 : ANALYSE ET CONCEPTION.....	<i>25</i>
2.1 Choix de la méthode d'analyse et justification	<i>26</i>
2.2 Choix de l'outil de modélisation et justification	<i>31</i>
2.3 Etude détaillée de la solution	<i>32</i>
PARTIE 3 : REALISATION ET MISE EN ŒUVRE	<i>55</i>
3.1 Matériels et logiciels utilisés.....	<i>56</i>
3.2 Architectures matérielle et logicielle de l'application.....	<i>70</i>
3.3 Sécurité de l'application	<i>73</i>
3.4 Mise en place de la base de données	<i>75</i>
PARTIE 4 : GUIDE D'EXPLOITATION.....	<i>80</i>
4.1 Configuration logicielle et matérielle	<i>80</i>
4.2 Déploiement et suivi	<i>83</i>
4.3 Maintenance	<i>89</i>
PARTIE 5 : GUIDE D'UTILISATION	<i>90</i>
5.1 Description textuelle du logiciel.....	<i>91</i>
5.2 Plan de navigation.....	<i>92</i>
5.3 Présentation des différentes interfaces.....	<i>94</i>
5.4 Présentation des états	<i>99</i>
CONCLUSION.....	<i>103</i>
BIBLIOGRAPHIE INDICATIVE.....	<i>I</i>
WEBOGRAPHIE INDICATIVE.....	<i>II</i>
TABLES DES MATIERES.....	<i>III</i>

AVANT-PROPOS

L'Institut Africain d'Informatique (IAI) est une école inter-état, créée en 1971 à Fort Lamy, actuel Ndjamena et est basée à Libreville au Gabon. Une représentation nationale pour le cycle d'analyste programmeur a été ouverte depuis quelques années dans certains pays membres, notamment au Cameroun, au Niger et en 2002 au Togo dans le but de mettre à la disposition des Etats membres un centre de formation adéquat et conforme aux normes internationales.

La représentation de l'IAI au TOGO localisée actuellement dans l'enceinte du CENETI offre une formation aboutissant à l'obtention d'une **Licence Professionnelle en Informatique** ou le diplôme **d'Ingénieurs des Travaux Informatiques**. La formation s'étale sur une d'une durée de trois ans en options **Génie Logiciel et Systèmes d'Information (GLSI)**, **Administration Systèmes & Réseaux (ASR)** et **Multimédia Technologies Web et Infographie (MTWI)**.

Dans l'optique, d'assumer sa position de référence dans la formation des cadres informatiques l'IAI-TOGO a intégré à la fin de la troisième année un stage pratique en entreprise qui permet à ses étudiants de mettre en pratique les connaissances acquises au cours de leur formation, de se confronter et de se préparer à la vie professionnelle. A la fin de ce stage, les étudiants doivent présenter leur projet devant un jury dans le but de valider leur cursus et obtenir l'un des deux diplômes proposés par l'Institut.

C'est dans ce cadre que nous avons eu à effectuer un stage dans l'ONG **Mivo Energie** sous l'encadrement de notre maître de stage M. NIMON Jean-Pierre et de notre superviseur M. EKLOU Kader.

RESUME

En vue de l'obtention du diplôme de licence professionnelle en Informatique de l'**Institut Africain d'Informatique (IAI-TOGO)**, nous avons eu à effectuer un stage d'une durée de trois mois au sein de l'ONG **Mivo Energie**, où il a été soumis à notre étude le thème : « **Mise en place d'un service de gestion de congés et d'évaluation des performances du personnel d'une entreprise** ». Ce thème consiste à développer une application web et une API afin d'automatiser les processus d'évaluations et de gestion de congés de l'ONG Mivo Energie. Pour se faire nous avons eu à utiliser le langage de modélisation UML couplé au processus de développement 2TUP et l'application Power Designer pour l'analyse et la conception de notre service. Lors du développement nous avons eu à utiliser un certain nombre d'outils notamment un serveur de base de données PostgreSQL pour la persistance des données, le langage Python couplé au framework FastAPI et le framework Angular ajouté aux langages Html, CSS et TypeScript qui nous ont permis de mettre en place respectivement notre API et l'application web.

ABSTRACT

In order to obtain the professional bachelor's degree in Computer Science from the **African Institute of Informatics (IAI-TOGO)**, we had to do a three-month internship within the NGO Mivo Energie, where the theme: **“Establishment of a leave management and performance evaluation service for a company's staff”** was submitted to our study. This theme consists of developing a web application and an API in order to automate assessment and leave management process of the NGO Mivo Energie. To do this we had to use the UML modeling language coupled with the development process 2TUP and the application Power Designer for the analysis and design of our service. During the development we had to use a certain number of tools including PostgreSQL database server for data persistence, the Python language coupled with the FastAPI Framework and the Angular framework added to the Html, CSS and TypeScript languages which allowed us to set up our API and the web application respectively.

GLOSSAIRE

Framework : un ensemble cohérent de composants logiciels qui sert à créer les fondations ainsi que les grandes lignes de tout un d'une partie d'un logiciel.¹

Endpoints : représentent les canaux de communications où les API envoient leurs demandes et où réside la ressource.²

Container : (conteneur en français) est une boîte permettant d'envelopper et d'isoler des autres et d'exécuter une application et ses dépendances.³

Kubernetes : est un logiciel de gestion et de mise en œuvre des conteneurs.⁴

IDE : désigne un ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent un logiciel.

Command Line Interface (CLI) : interface de ligne de commande en français est une interface homme-machine dans laquelle la communication entre l'ordinateur se fait en mode texte avec l'utilisation des commandes.⁵

PAAS : Plateform-as a- Service est un type de cloud computing destinés aux développeurs pour exécuter déployer et gérer des applications sans avoir à construire ni maintenir les processus nécessaires au fonctionnement de ces applications.⁶

Design patterns : sont des solutions typiques à des problèmes communs en développement logiciel : ils ne sont pas une implémentation concrète d'une solution

¹ <https://fr.wikipedia.org/wiki/Framework>

² <https://www.mayasquad.com/glossaire/endpoint/>

³ <https://www.hpe.com/fr/fr/what-is/containers.html>

⁴ <https://fr.wikipedia.org/wiki/Kubernetes>

⁵ <https://fr.wikipedia.org/wiki/CLI>

⁶ <https://www.redhat.com/fr/topics/cloud-computing/what-is-paas>

à un problème, mais plutôt une stratégie à appliquer pour le résoudre de façon élégante et maintenable.⁷

LISTE DES FIGURES

Figure 1 : Plan de localisation de l'ia-togo	4
Figure 2 : Organigramme DE MIVO energie.....	6
Figure 3: Plan de localisation de l'ONG Mivo Energie	7
Figure 4:Diagramme de contexte statique	32
Figure 5 Diagramme de cas d'utilisation général.....	36
Figure 6: Diagramme de cas d'utilisation du Directeur exécutif et du raf	37
Figure 7: Diagramme d'activité "S'authentifier"	41
Figure 8: Diagramme d'activité "Faire un demande de congés"	42
Figure 9: Diagramme d'activité "Evaluer un employé"	44
Figure 10: Diagrammes d'activité "Lancer une campagne d'évaluation"	44
Figure 11: Diagramme de séquence « s'authentifier »	46
Figure 12: Diagramme de séquence « faire une demande de congés »	47
Figure 13: Diagramme de séquence valider une demande de congés (Supérieur hiérarchique).....	48
Figure 14:Diagramme de séquence lancer une campagne d'évaluation	49
Figure 15: Diagramme de séquence évaluer un employé	50
Figure 16: Diagramme de déploiement.....	52
Figure 17: Diagramme de classe	54
Figure 18: Architecture matérielle de l'application	68
Figure 19: Fichier session.py de connexion à la base de données	76
Figure 20: Exemple de modèle SQLAlchemy	77
Figure 21: Classe de crud de base	78
Figure 22: Exemple de classe encapsulant les opérations crud d'un modèle SQLAlchemy.....	79
Figure 23 : FICHER Docker	83
Figure 24 : Fichier de configuration okteto-stack.yaml	84
Figure 25 : Déploiement de l'api avec le cli d'okteto	85
Figure : 26 Déploiement de L'API (Endpoints).....	86
Figure 27: Installation de Heroku CLI sous Ubuntu 20.4.....	87
Figure 28: Création d'une base de données avec Heroku CLI.....	87
Figure 29: Déploiement de l'application web avec Firebase	88
Figure 30:Plan de navigation de l'application	93
Figure 31:Page de connexion	95
Figure 32: Page montrant la liste des employés.....	95
Figure 33:Page de validation d'une demande de congés par un CRH	96

⁷ <https://openclassrooms.com/fr/courses/6031956-creez-une-application-web-en-php-de-qualite-professionnelle/6107816-quest-ce-quun-design-pattern>

Figure 34:Page de validation d'une demande de congés par un supérieur hiérarchique	96
Figure 35:Page d'évaluation.....	96
Figure 36: Page d'attribution des objectifs	97
Figure 37: Page d'ajout et de consultation campagnes d'évaluations	98
Figure 38: Page des statistiques du GRH (1)	99
Figure 39: Page de statistiques du GRH (2)	100
Figure 40: Page des statistiques des congés pour les utilisateurs (1)	101
Figure 41: Page des statistiques des congés pour les utilisateurs (2)	102

LISTE DES TABLEAUX

Tableau 1: Liste des participants au projet	x
Tableau 2 Coût matériel des solutions.....	17
Tableau 3 coûts d'acquisition de factorial	18
Tableau 4 Coût total de FACTORIAL.....	19
Tableau 5 Cout du développement de la solution	20
Tableau 6 Coût total du développement de la plateforme.....	21
Tableau 7: Planning prévisionnel de réalisation	23
Tableau 8: Récapitulatif des cas d'utilisation	34
Tableau 9 Configuration logicielle	81
Tableau 10 configuration matérielle	82

NOM	FONCTION	ROLE
M. AYITE Yawo Mawulom Guillaume Florian	Etudiant en troisième année Option Génie Logiciel et Système d'Information à L'IAI-TOGO	Réalisateur
M. EKLOU KADER	Enseignant à l'Institut Africain d'Informatique représentation Du TOGO. (IAI-TOGO)	Superviseur
M. NIMON B. Jean- Pierre	Responsable administratif et financier	Maître de stage

LISTE DES PARTICIPANTS AU PROJET

TABEAU 1: LISTE DES PARTICIPANTS AU PROJET

INTRODUCTION

L'Informatique est de nos jours, une ressource intervenant dans tous les domaines de notre vie quotidienne. Il permet l'automatisation tant des activités de la vie courante que des tâches métiers plus complexes des entreprises.

La gestion des ressources humaines est un domaine englobant un grand nombre d'activités mises en œuvre afin d'assurer la compétitivité et la performance des entreprises. Afin d'assurer son rôle, elle s'appuie sur des processus dont notamment l'évaluation des performances du personnel et la gestion des congés.

C'est dans l'optique d'optimiser et d'automatiser les processus sus-cités, qu'il nous a été donné pour thème la « **Mise en place d'un service de gestion de congés et d'évaluation des performances d'une entreprise** ». Ce thème consiste tout d'abord à mettre en œuvre une API permettant d'automatiser ces processus, et ensuite à développer une application web l'exploitant et mettre cette dernière à la disposition de l'ONG Mivo Energie.

Le présent mémoire constitue un bref résumé du travail que nous avons eu à effectuer au cours du stage dans l'ONG Mivo Energie du 5 Juillet au 17 Septembre 2021. Il est réparti en cinq grandes parties à savoir : le cahier des charges, l'analyse et la conception, la réalisation et la mise en œuvre, le guide d'exploitation et le guide d'utilisation.

PARTIE 1 : CAHIER DES CHARGES

PARTIE 1 : CAHIER DES CHARGES

Le cahier des charges est un document formel comportant les besoins exprimés par le client, dans notre cas l'ONG Mivo Energie ainsi que les propositions de solutions à mettre en œuvre. Il permet également aux parties prenantes de définir les résultats attendus, d'estimer avec justesse les coûts et les délais du projet.

Notre cahier de charges porte notamment sur les présentations (de l'IAI-TOGO et du centre d'accueil), notre thème de stage, l'étude et la critique de l'existant, les propositions, choix et justifications des solutions ainsi que le planning de réalisation du projet.

1.1 Présentations

1.1.1 Brève présentation de l'IAI-TOGO

L'institut Africain d'Informatique Représentation du TOGO est une école de formation en Informatique. L'IAI-TOGO fait partie du réseau d'écoles IAI créé le 29 Janvier 1971 à Fort Lamy (actuel N'Djamena) en république du TCHAD. Ce réseau est composé de 11 pays membres dont le BÉNIN, le BURKINA-FASO, le CAMEROUN, le CONGO, la COTE d'IVOIRE, le GABON le NIGER, la RÉPUBLIQUE CENTRAFRICAINE le SÉNÉGAL, le TCHAD et le TOGO.

1.1.1.1 Formation à l'IAI-TOGO

A sa création en 2002, l'IAI-TOGO ne formait que des analystes programmeurs (Option Génie Logiciel). En 2007-2008, une nouvelle filière dénommée Maintenance Informatique et Réseaux (MIR) a vu le jour. Celle-ci a été renommée Systèmes et Réseaux (SR) durant l'année académique 2012-2013. Une troisième filière fut créée durant l'année académique 2014-2015, nommée filière Multimédia et Technologie du Web et Infographie.

Les étudiants de l'IAI-TOGO ont la possibilité de suivre l'une des trois formations proposées pour une durée de trois ans afin d'obtenir le diplôme d'Ingénieur des Travaux Informatiques ou la licence professionnelle.

PARTIE 1 : CAHIER DES CHARGES

1.1.1.2 Plan de localisation

L'IAI-TOGO est situé à Lomé, dans les locaux du Centre National d'Études et de Traitements Informatique (CENETI), lui-même situé dans le quartier administratif, à proximité de la Direction Générale de la Communauté Électrique du Bénin CEB et de l'immeuble de SUNU Assurances. Ses coordonnées sont :

59, Rue de la Kozah

07 BP :12456 Lomé-TOGO

Tel:(00228) 22204700 / 22212706

E-mail : iaitogo@iaitogo.tg

Site web : www.iai-togo.tg

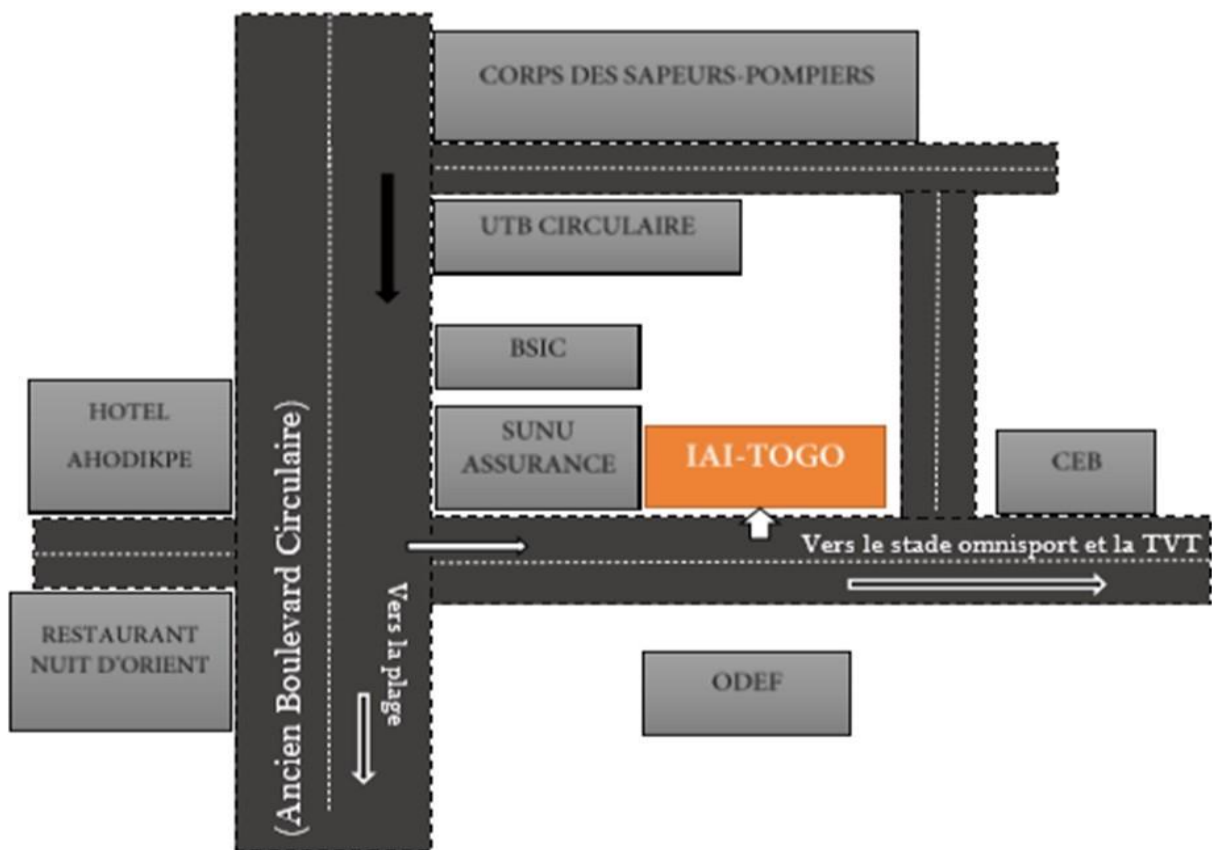


FIGURE 1 : PLAN DE LOCALISATION DE L'IAI-TOGO.

1.1.2 Présentation du cadre de stage

1.1.2.1 Statut

L'ONG **Mivo Energie** est un programme créé par l'ONG française Entrepreneurs du Monde, devenu une association de droit togolais en 2017. Mivo Energie repose sur un modèle économique innovant, qui associe un réseau de distribution « au dernier kilomètre » à des services financiers et des outils de marketing social, afin de dépasser l'ensemble des barrières de marché qui entravent l'accès aux produits.

1.1.2.2 Mission

MIVO Energie a pour mission de proposer des solutions d'accès à l'énergie durables et efficaces, aux personnes en situation de précarité énergétique pour améliorer leurs conditions de vie sur le plan économique et sanitaire, et réduire leur impact environnemental.

1.1.2.3 Activités

Mivo Energie, dispose d'un réseau de distribution qui lui permet de mettre à disposition des ménages des équipements modernes : lampes solaires, réchauds à gaz et foyers améliorés économes et combustibles.

L'ONG appuie également des producteurs locaux de réchauds, met en place un réseau de distribution viable et développe des solutions financières adaptées et organise des campagnes de sensibilisation.

1.1.2.4 Quelques réalisations

Mivo Energie intervient notamment dans le domaine de l'énergie et du social. Elle a à son actif quelques réalisations notables telles que :

- La diffusion des équipements économes : lampes solaires, foyers améliorés, gaz ;
- Sensibilisation dans les groupements de la Microfinance Assilassimé, Vision FINANCE PLUS :
- Projet d'entrepreneuriat dans le domaine solaire, basé à Dapaong

PARTIE 1 : CAHIER DES CHARGES

1.1.2.5 Organigramme

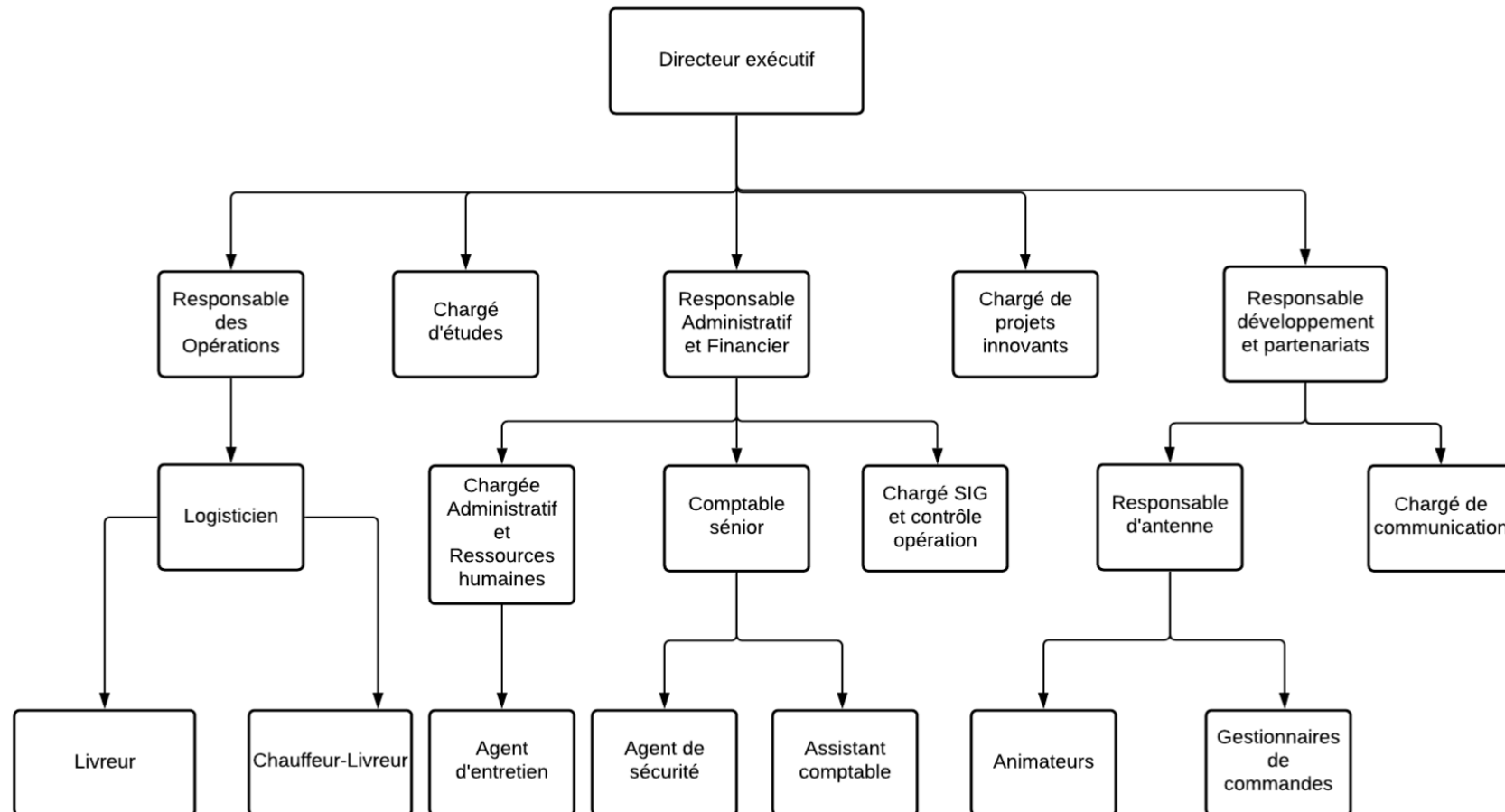


FIGURE 2 : ORGANIGRAMME DE MIVO ENERGIE

1.1.2.6 Service d'accueil

Nous avons effectué notre stage de fin de formation au sein du service des Ressources Humaines et des Affaires Financières de l'ONG Mivo Energie

1.1.2.7 Plan de localisation

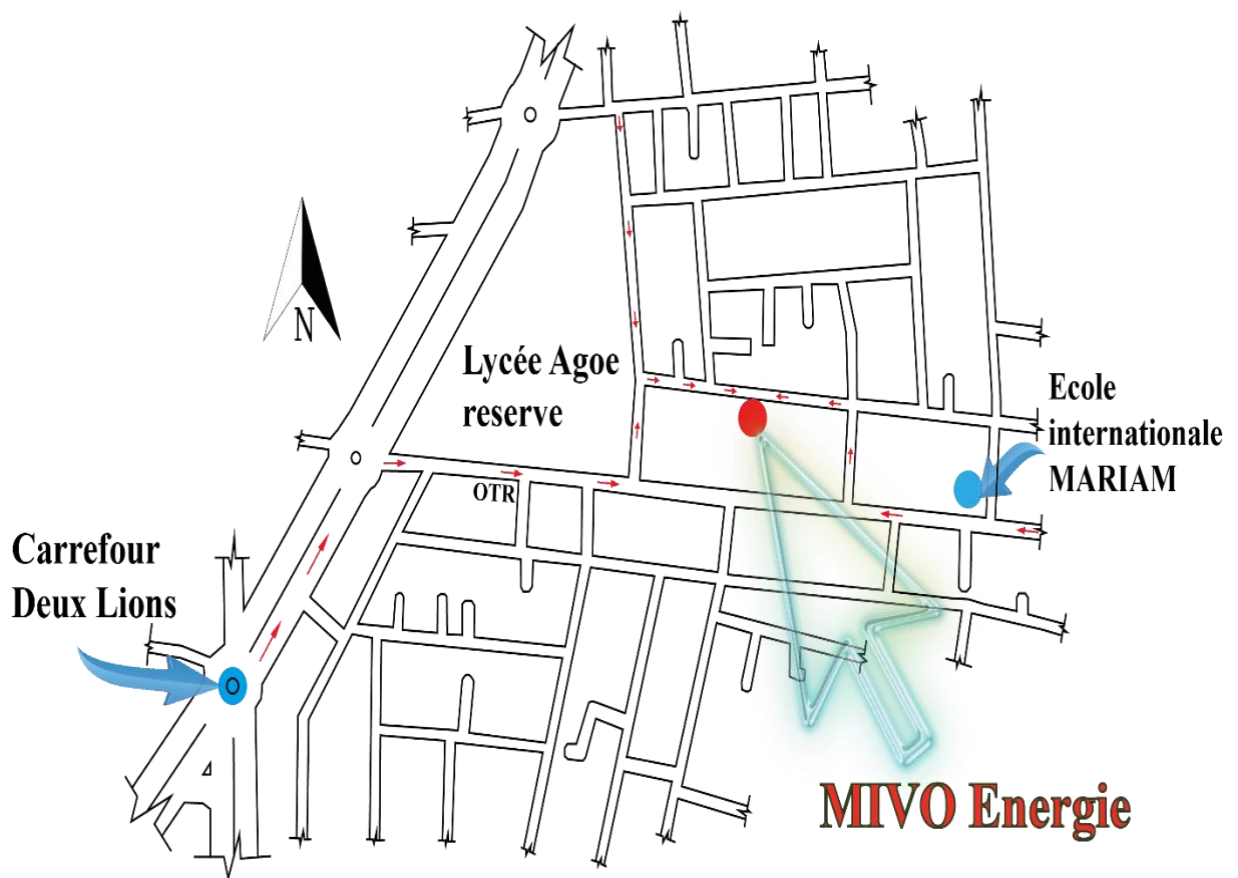


FIGURE 3: PLAN DE LOCALISATION DE L'ONG MIVO ENERGIE

1.2 Thème de stage

1.2.1 Présentation du sujet

Le service des Ressources Humaines se trouve au cœur de la gestion des employés, que ce soit au niveau de l'analyse de leur performance, de la gestion des congés ou même des salaires. C'est dans le but de faciliter cette gestion qu'il nous a été confié dans le cadre de notre stage le thème : « **Mise en place d'un service de gestion de congés et d'évaluation des performances du personnel d'une entreprise** ».

Ce thème consiste à développer et mettre en place une application web et une API permettant de faciliter les processus de demande et d'attribution des congés et d'évaluation des performances du personnel de l'ONG MIVO Energie.

1.2.2 Problématique du sujet

La gestion des congés et des évaluations peut parfois s'avérer complexe surtout lorsqu'on ne dispose pas de moyens efficaces pour la faire.

A l'état actuel la gestion des congés au sein de l'ONG Mivo Energie est faite de façon manuelle, les pièces justificatives des demandes sont conservées sous format papier ; l'analyse des performances quant à elle se fait avec de simples formulaires remplis à la main ce qui implique :

- Une exploitation lourde et fastidieuse des données qui ne permet pas la traçabilité ;
- Une perte de temps liée au circuit de validation d'une demande de congé ;
- De potentiels conflits ou fraudes dans l'obtention des congés ;
- La perte des informations en cas de destruction ou égarement des formulaires d'évaluation, des pièces justificatives ou des feuilles de calcul Excel.

PARTIE 1 : CAHIER DES CHARGES

L'ONG Mivo ayant remarqué certaines insuffisances dans cette gestion s'est tournée vers nous pour lui proposer une solution efficace pour la gestion des congés et l'évaluation des performances du personnel.

Au vu de toutes ces attentes, il se dégage la problématique contenue dans les interrogations ci-dessous :

- Comment faciliter le processus de demande de congés ?
- Comment suivre l'évolution des demandes de congés des employés ?
- Comment améliorer le système d'évaluation existant ?

1.2.3 Intérêt du sujet

1.2.3.1 Objectifs

Divisé en objectifs et en résultats attendus, dans cette partie il est question de montrer le motif et les attentes par rapport à notre sujet de mémoire.

❖ Objectif général

L'objectif général de ce projet est de permettre à l'ONG Mivo-Energie de disposer d'un système qui permettra de faciliter les demandes et la gestion des congés de même que les évaluations du personnel.

❖ Objectifs spécifiques

Ce projet permettra plus précisément à l'ONG Mivo Energie :

- D'enregistrer les informations des employés dans une base de données afin d'y accéder en temps réel ;
- D'attribuer des objectifs à chaque employé suivant son poste ;
- De permettre aux employés d'effectuer des demandes de congés ;
- De permettre aux supérieurs hiérarchiques de consulter les informations de leurs subordonnées ;
- De faciliter l'évaluation des employés sur une certaine période ;
- De pouvoir automatiser le processus de demande de congés ;
- De pouvoir garder un historique des congés et des évaluations de tous les employés ;

PARTIE 1 : CAHIER DES CHARGES

- De pouvoir stocker les différentes pièces justificatives des demandes, les contrats, et la totalité du dossier RH de chaque employé, pour y accéder en temps voulu.

1.2.3.2 Résultats attendus

A la fin de notre stage l'ONG Mivo Energie disposera d'une application web sécurisée qui offrira entre autres :

- un service permettant l'automatisation et l'accélération des demandes de congés et de leur suivi ;
- la dématérialisation des documents et des pièces justificatives pour les rendre disponibles à tout moment grâce au stockage en ligne ;
- une gestion facile et efficace des évaluations du personnel ;
- l'envoi de notification par email ;
- des statistiques fiables et des calendriers donnant des aperçus des congés de chaque employé.

1.3 Etude de l'existant

Dans l'étude de l'existant, nous allons dans un premier temps présenter le processus de gestion des congés et ensuite celui de l'évaluation des performances tels qu'ils nous ont été décrits lors des divers échanges avec la chargée des ressources humaines et le responsable administratif et financier.

1.3.1 Gestion des congés

A l'état actuel la gestion des congés se fait à l'aide de formulaires et de fichiers Excel. Les fichiers Excel permettent entre autres de garder la trace des congés octroyés aux employés

Le processus se déroule comme suit :

- L'employé remplit une demande manuscrite stipulant les raisons de sa demande, la date de départ en congés et la date de retour ;
- Après avoir rempli sa demande l'employé l'envoie à son supérieur hiérarchique pour approbation ;
- Le supérieur hiérarchique doit approuver sa demande et la signer ;
- Après l'approbation de son supérieur hiérarchique, l'employé envoie sa demande aux services des ressources humaines ;
- Le responsable des ressources humaines approuve et signe la demande ;
- L'employé est ensuite notifié de la décision du chargé des RH.

1.3.2 Evaluation des performances des employés

L'évaluation des performances des employés dans la société se fait essentiellement de façon qualitative, et ne dispose d'aucune grille de notation. Les différents documents produits durant ce processus sont stockés en format papier et mis dans des classeurs.

Cette procédure se fait comme décrite ci-dessous :

- Le responsable des ressources humaines lance une campagne d'évaluation ;
- Les employés ont la possibilité de s'auto-évaluer et d'apporter des observations quant à des améliorations à apporter dans l'ONG ;
- Après s'être auto-évalué l'employé a un entretien avec son supérieur hiérarchique qui leur permet d'échanger ;
- À la fin de cette entrevue on fait une copie de l'évaluation faite par le supérieur hiérarchique, qui, doit être signée par ce dernier et l'employé évalué ;
- Après la signature le responsable des ressources humaines fait une copie de l'évaluation signée qu'il garde dans des dossiers.

1.4 **Critique de l'existant**

Comme constaté au niveau de la problématique, l'emploi de formulaire papier et les validations manuelles des différents processus constituent un énorme travail pesant sur le service des ressources humaines, et un risque de perte ou de corruption des informations.

D'après notre étude de l'existant nous avons constaté des manquements tels que mentionnés ci-dessous :

- Les traitements de demandes de congés sont longs et fastidieux ;
- Les contrats et les documents sont sauvegardés au format papier et sont parfois difficile à retrouver ;
- Il n'existe pas de base de données comportant les informations du personnel ;
- Les fichiers Excel de la gestion des congés sont stockés uniquement sur la machine du chargé des ressources humaines ;
- Les évaluations sont elles aussi sauvegardées au format papier.

1.5 **Propositions et choix de solutions**

Afin de pallier les insuffisances précédemment remarquées, nous avons eu à proposer deux solutions :

❖ **Achat, configuration et utilisation de logiciels / modules existants :**

Factorial : une application d'évaluations des performances des employés.

❖ **Développement de la solution :**

Développement d'une plateforme pour gérer les congés et l'analyse des performances, à partir de l'analyse des besoins et des langages de programmation.

1.5.1 **Evaluations techniques des solutions**

1.5.1.1 **Factorial**

La première solution consiste à utiliser des solutions déjà développées permettant de répondre aux besoins énumérés dans le cahier des charges. Factorial est une application multiplateforme permettant la gestion des Ressources Humaines en passant de la gestion des absences, des congés et plannings aux entretiens et évaluations.

❖ **Avantages⁸**

Cette solution présente plusieurs avantages. Elle offre entre autres :

- Une grande palette de fonctionnalités permettant la gestion des ressources humaines ;
- Une solution de gestion électronique des documents ;
- Des indicateurs, tableaux de bords et rapports personnalisés qui permettent de prendre de meilleures décisions pour l'entreprise et son personnel ;
- Une application mobile et web permettant l'accès en temps réel aux différents modules ;
- L'intégration d'application comme Gmail ou Slack.

⁸ Source : <https://factorial.fr/fonctionnalites>

❖ Inconvénients

- Elle comporte des fonctionnalités qui ne suscitent pas l'intérêt de Mivo Energie mais qui sont facturées ;
- La dépendance vis-à-vis des éditeurs de logiciels tiers

1.5.1.2 Développement de la solution

Cette solution consiste à développer une application web et une API pour la gestion des congés et les évaluations des performances.

❖ Avantages

- Elle est parfaitement adaptée aux divers besoins de l'ONG puisque qu'elle a été développée pour répondre à ses exigences ;
- Elle assure à l'ONG Mivo Energie d'être indépendante par rapport à un éditeur de logiciel tiers ;
- Son développement nous permettra en tant que stagiaire de mettre à profit nos compétences de les améliorer et d'en acquérir de nouvelle ;

❖ Inconvénients

- Engendre un grand nombre de dépense en matière de logiciels, main d'œuvre et de matériels informatique ;
- Le développement de la solution et sa mise en place peuvent prendre du temps compte tenu des besoins exprimés.

1.5.2 Evaluations financières des solutions

Dans cette partie, nous étudierons entre autres les dépenses que chacune des solutions pourrait occasionner pour son acquisition et son utilisation. Les différentes estimations sont faites sur une période d'un an.

PARTIE 1 : CAHIER DES CHARGES

1.5.2.1 Coût matériel

Les coûts matériels étant communs aux deux solutions ils sont résumés dans le tableau ci-dessous :

TABLEAU 2 COUT MATERIEL DES SOLUTIONS

Désignation	Description	Coût unitaire	Quantité	Montant (FCFA)
Achat d'un ordinateur pour la conception la mise en œuvre et l'administration de la plateforme	Pc HP Pavilion BC506-nf Intel Core i5 8 Go RAM 512 Go SSD	584 110	1	584 110
Connexion Internet	FAI (Fournisseur d'Accès à internet) Canalbox Premium 50 mb/s	30 000 ⁹	3	90 0000

⁹ Source : [Canalbox \(canalplus-afrique.com\)](https://canalplus-afrique.com)

PARTIE 1 : CAHIER DES CHARGES

1.5.2.2 Coût des solutions

1.5.2.2.1 Evaluations financières de Factorial

TABLEAU 3 COUTS D'ACQUISITION DE FACTORIAL

Désignation	Description	Coût unitaire	Quantité	Montant (FCFA)
Abonnement annuel à Factorial	1 an d'abonnement à l'Edition business de Factorial	136 485,6 ¹⁰	12	1 889 799
Mise en place de la solution et de la plateforme	Main d'œuvre du concepteur	6 000	200 (8 heures * 25 Jours)	1 200 000
Formation du personnel de l'ONG	Formation du personnel pour l'utilisation de la plateforme	20 000	12 (4 heures * 3 jours)	240 000

¹⁰ Source : <https://factorial.fr/tarifs>

PARTIE 1 : CAHIER DES CHARGES

❖ Coût total de Factorial

TABLEAU 4 COUT TOTAL DE FACTORIAL

Désignation	Montant (FCFA)
Achat d'un ordinateur	584 110
Connexion Internet	90 000
Abonnement annuel à Factorial	1 889 799
Formation du personnel	240 000
Mise en place de la plateforme	1 200 000
TOTAL	4 003 909

PARTIE 1 : CAHIER DES CHARGES

1.5.2.2.2 Développement de la solution

TABLEAU 5 COUT DU DEVELOPPEMENT DE LA SOLUTION

Désignation	Description	Coût unitaire	Quantité	Montant (FCFA)
Développement de la solution	Analyse, conception, développement et tests de l'application	6 000	592 (8 heures * 74 jours)	3 552 0000
Hébergement annuel de l'API	Plan Developer Pro Okteto	10503,94¹¹	12	126 048
Hébergement annuel de l'application web	Hébergement annuel sur firebase	34 098¹²	12	409 176
Hébergement annuel de la base de données	Plan standard Heroku Postgres (64 Go de stockage et 4 Go)	27641,94¹³	12	331 704
Cloud Storage	Stockage en ligne de fichier sur firebase	19996,18¹⁴	12	239 955
Formation du personnel	Formation du personnel à l'utilisation du logiciel	20 000	12 (4 heures* 3jours)	240 000

¹¹ Source : <https://okteto.com/pricing/>

¹² Source : <https://firebase.google.com/pricing#blaze-calculator>

¹³ Source : <https://elements.heroku.com/addons/heroku-postgresql>

¹⁴ Source : <https://firebase.google.com/pricing#blaze-calculator>

PARTIE 1 : CAHIER DES CHARGES

❖ Coût total de la solution

TABLEAU 6 COUT TOTAL DU DEVELOPPEMENT DE LA PLATEFORME

Désignation	Montant (FCFA)
Achat d'un ordinateur	584 110
Connexion Internet	90 000
Développement de la solution	3 552 0000
Hébergement annuel de l'API	126 048
Hébergement annuel de l'application web	409 176
Hébergement annuel de la base de données	331 704
Cloud Storage	239 955
Formation du personnel	240 000
TOTAL	5 572 993

1.5.3 Choix de la solution

Après avoir fait une étude comparative entre les deux solutions proposées, à travers les avantages et inconvénients ainsi que les coûts d'acquisition de chacune d'elles, nous avons en commun accord avec l'ONG Mivo Energie, retenu la deuxième solution : Mettre en place un système de gestion des congés et des évaluations du personnel.

Nous avons fait ce choix tout d'abord pour des raisons pédagogiques mais aussi parce que la solution choisie répond parfaitement aux besoins exprimés par l'ONG.

1.6 Planning prévisionnel de réalisation

TABLEAU 7: PLANNING PREVISIONNEL DE REALISATION

TÂCHES	DESCRIPTION	PERIODE (2021)	DUREE
Intégration et choix du thème	-Intégration dans le centre d'accueil -Prise de connaissance du thème du stage	05-09 Juillet	5 jours
Recueil des besoins	- Capture des besoins fonctionnels -Capture des besoins techniques -Rédaction du cahier des charges	12-19 Juillet	8 jours
Analyse et conception	-Elaboration des diagrammes et description des cas d'utilisation et du diagramme de classes	19-23 Juillet	5 jours
PHASE DE CODAGE			
Itération N° 1	Mise en place de la base de données, des fonctions de CRUD de base et de l'authentification	24 -29 Juillet	6 jours
Itération N° 2	Développement du module de gestion des demandes de congés	30 Juillet-08 Août	10 jours
Itération N° 3	Configuration du serveur d'envoi d'email et codage du service d'envoi de notification par email	09-12 Août	4 jours
Itération N° 4	Mise en place du module d'évaluations des employés	13-16 Août	4 jours
Itération N° 5	Développement de l'application web	17 Août-02 Septembre	17 jours
Itération N° 6	Intégration de l'API à l'application web	03-06 Septembre	4 jours
Tests et corrections des erreurs		07-15 Septembre	9 jours
Rédaction du mémoire		15 Juillet -09Septembre	56 jours
Correction du mémoire		15-17 Septembre	3 jours

PARTIE 1 : CAHIER DES CHARGES

La première partie de notre document intitulée « Cahier des charges », nous a offert un bref aperçu de la solution que nous aurons à développer, des tâches que nous effectuerons durant toute la durée de notre projet. Le principal objectif du Cahier des charges étant de permettre l'expression des besoins et des résultats qui sont attendus à la fin du projet aux différentes parties prenantes afin de les mettre en accord, a donc été atteint. Nous entamerons la deuxième partie intitulée : « Analyse et conception ».

PARTIE 2 : ANALYSE ET CONCEPTION

2.1 **Choix de la méthode d'analyse et justification**

Une méthode d'analyse et de conception est un procédé qui a pour objectif de permettre de formaliser les étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client.¹⁵ L'analyse et la conception d'un système d'information sont réalisées à travers une méthode de développement et plusieurs ateliers de travail en utilisant un langage de modélisation approprié.

Une méthode d'analyse est scindée en deux parties :

- La phase d'analyse permet de lister les résultats attendus, en termes de fonctionnalités, de performance, de robustesse, de maintenance, de sécurité, d'extensibilité, etc.
- La phase de conception permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation.

Dans la démarche d'analyse et conception des systèmes informatiques on distingue :

- ❖ La méthode fonctionnelle ou structurée
- ❖ La méthode objet ;

2.1.1 **Comparaison des méthodes de modélisation**

❖ Les méthodes fonctionnelles ou structurées

Les méthodes fonctionnelles consistent en la décomposition du système en sous-systèmes. Elles trouvent leur origine dans les langages dit procéduraux. Elles mettent en évidence les fonctions à assurer et proposent une approche hiérarchique descendante et modulaire.

¹⁵ Source : https://fr.wikipedia.org/wiki/M%C3%A9thodes_d%27analyse_et_de_conception

L'approche fonctionnelle dissocie le problème de la représentation des données, du problème du traitement de ces données. Comme méthode utilisant l'approche fonctionnelle nous pouvons citer MERISE, AXIAL, etc.

❖ L'approche orientée objet

L'approche objet désigne l'ensemble des processus et langages utilisés au cours du cycle de vie de l'application, qui reposent sur la manipulation des objets. Ici, une application est vue comme un ensemble d'objets qui coopèrent.

Un autre concept qui est très souvent lié à l'approche objet est celui de classe qui permet de regrouper les propriétés communes des objets. La particularité d'une telle approche est qu'elle rapproche les données et leurs traitements associés au sein d'un unique objet. La difficulté de cette modélisation consiste à créer une représentation abstraite, sous forme d'objets, d'entités ayant une existence. Ainsi, les architectures des logiciels utilisant l'approche objet sont fondées sur les objets du système, plutôt que sur la fonction qu'elles sont censées réaliser.

❖ L'approche fonctionnelle vs l'approche objet

L'approche structurée privilégie la fonction comme moyen d'organisation du logiciel. Ce n'est pas pour cette raison que l'approche objet est une approche non fonctionnelle. En effet, les méthodes d'un objet sont des fonctions.

Ce qui différencie sur le fond l'approche objet de l'approche fonctionnelle, c'est que les fonctions obtenues à l'issue de la mise en œuvre de l'une ou l'autre méthode sont distinctes.

L'approche objet est une approche orientée donnée. Dans cette approche, les fonctions se déduisent d'un regroupement de champs de données formant une entité cohérente, logique, tangible et surtout stable quant au problème traité. L'approche structurée classique privilégie une organisation des données postérieure à la découverte des grandes, puis petites fonctions qui les décomposent, l'ensemble constituant les services qui répondent aux besoins.

En approche objet, l'évolution des besoins aura le plus souvent tendance à se présenter comme un changement de l'interaction des objets.

PARTIE 2 : ANALYSE ET CONCEPTION

S'il faut apporter une modification aux données, seul l'objet incriminé (encapsulant cette donnée) sera modifié. Toutes les fonctions à modifier sont bien identifiées : elles se trouvent dans ce même objet : ce sont ses méthodes. Dans une approche structurée, l'évolution des besoins entraîne souvent une dégénérescence, ou une profonde remise en question, car la décomposition des unités de traitement (du programme principal aux sous-fonctions) est directement dictée par ces besoins. D'autre part, une modification des données entraîne généralement une modification d'un nombre important de fonctions éparpillées et difficiles à identifier dans la hiérarchie de cette décomposition.

Nous choisissons de retenir l'approche objet car elle supporte mieux les modifications, est très flexible quand il s'agit de réutilisation de module ou d'artefact et permet d'obtenir des modèles plus évolutifs comparativement à l'approche fonctionnelle.

C'est un choix important que nous faisons car la méthode a un grand impact sur toutes les étapes de notre projet.

2.1.2 Le langage de modélisation

Un langage de modélisation est un langage artificiel qui peut être utilisé pour exprimer de l'information ou de la connaissance des systèmes dans une structure qui est définie par un ensemble cohérent de règles. Ainsi, ayant choisi l'approche objet comme méthode de modélisation, nous ne pouvons que choisir un langage de modélisation orienté objet.

UML qu'est-ce que c'est ?

Langage de Modélisation Unifié, UML (Unified Modeling Language) est un langage de modélisation graphique à base de pictogrammes conçu comme une méthode normalisée de visualisation dans les domaines du développement logiciel et en conception orientée objet.

Il est une synthèse de langages de modélisation objet antérieurs comme Booch, OMT, OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard adopté par l'Object Management Group (OMG). La dernière version de la spécification validée par l'OMG est UML 2.5.1 (2017) ¹⁶.

Nous avons fait le choix d'utiliser UML car :

- Il facilite la compréhension des aspects liés à la conception et à l'architecture du logiciel au client.
- Il permet de bien définir les besoins des clients et d'éviter de livrer un logiciel qui ne satisfasse pas le client ;
- Il apporte une compréhension rapide du programme à d'autres développeurs externes en cas de reprise et en facilite la maintenance ;¹⁷

¹⁶ Source : [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique))

¹⁷ Source : <https://manurenaux.wp.imt.fr/2013/09/27/interet-de-luml-dans-un-projet-informatique/>

2.1.3 Le processus de développement

Un processus de développement logiciel est le processus de division du travail de développement logiciel en étapes ou sous-processus plus petits, parallèles ou séquentiels pour améliorer la conception, la gestion des produits et la gestion de projet. Pour réaliser un système d'information, nous avons le choix entre différents modèles de développement parmi lesquels nous distinguons :

- Le modèle en cascade ;
- Le modèle en V ;
- Le modèle par prototypage ;
- Les modèles de processus unifié
 - UP
 - RUP
 - 2TUP

Avantages des processus

Un processus unifié préconise une planification itérative dans laquelle « le plan précède l'action ». Le principe est qu'un plan d'ensemble détermine en fonction de la complexité du projet, les itérations nécessaires pour chaque phase et positionne ces phases dans le temps.

Ce plan d'ensemble n'inclut pas de détail par itération. Les itérations sont planifiées de façon progressive au cours de l'avancement du projet. Ainsi, toutes ces caractéristiques des processus unifiés permettent non seulement d'atteindre des objectifs immédiats dans le développement du logiciel mais aussi de mettre le client au centre du processus de développement en lui présentant des versions intermédiaires du logiciel au cours de son développement.

Dans le cadre de notre projet nous avons fait le choix d'utiliser le processus de développement 2TUP car ce dernier est itératif et définit des étapes à suivre lors du développement d'un logiciel. Il fait également une large place à la technologie et à la gestion du risque tout en permettant d'identifier les différents intervenants dans le projet.

2.2 **Choix de l'outil de modélisation et justification**

Ayant désigné UML comme langage de modélisation, nous avons la possibilité de choisir un outil permettant la représentation des diagrammes UML dont nous aurons besoin durant la conception de notre logiciel.

Parmi ces outils nous pouvons citer :

- StarUML ;
- LucidChart ;
- Entreprise Architect ;
- Power Designer.

Pour faire la modélisation de notre service nous avons choisi comme outil PowerDesigner dans sa version 16.5.

❖ **PowerDesigner**

PowerDesigner est un logiciel de conception créé par la société SAP, qui permet de modéliser les systèmes d'informations. Il fonctionne en tant qu'application native sous Windows ou en tant que plugin dans un environnement Eclipse.¹⁸

❖ **Raisons du choix de PowerDesigner**

Nous avons choisi PowerDesigner du fait qu'il est complet, il propose la conception de tous les diagrammes UML, qu'il est facile à prendre en main.

De plus, c'est le logiciel avec lequel nous avons eu à faire les modélisations de tous nos projets durant notre formation à l'IAI-Togo.

¹⁸ Source : <https://fr.wikipedia.org/wiki/PowerAMC>

2.3 Etude détaillée de la solution

2.3.1 Etude préliminaire

Nous détaillerons dans cette partie les différents acteurs qui interagiront avec notre système et nous en ferons une représentation à l'aide d'un diagramme de contexte statique.

2.3.1.1 Les acteurs

Un acteur représente un rôle d'un utilisateur interagissant avec le système modélisé. L'utilisateur peut être une machine, un système externe, une organisation ou un utilisateur humain.

Dans la conception de notre système nous avons recensé comme acteurs :

- Le chargé des ressources humaines ;
- L'employé ;
- Le supérieur hiérarchique ;
- Le Directeur exécutif ;
- Le responsable des affaires financières.

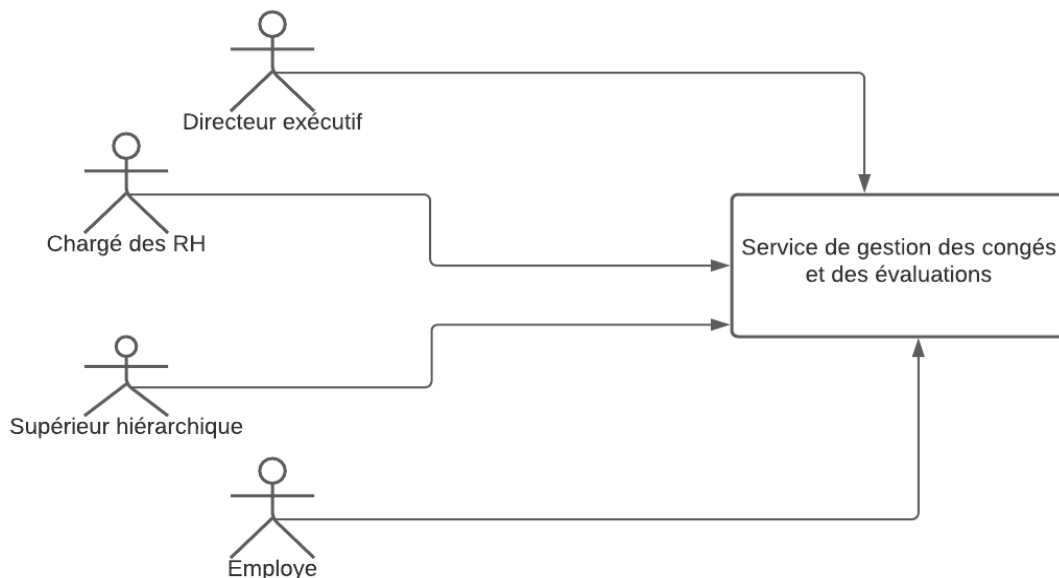


FIGURE 4:DIAGRAMME DE CONTEXTE STATIQUE

2.3.2 Capture des besoins fonctionnels

Dans cette sous-partie nous aurons à détailler les spécifications fonctionnelles du système à l'aide de quelques diagrammes notamment ceux de **cas d'utilisation**, d'**activités** et de **séquences**.

2.3.2.1 Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour une représentation du comportement fonctionnel d'un système logiciel. Ils permettent d'illustrer ce que le système fait et comment les différents acteurs l'utilisent. Ils sont utiles pour des présentations auprès de la direction ou les acteurs d'un projet.

2.3.2.1.1 Les cas d'utilisation

Les cas d'utilisation correspondent à un ensemble d'actions réalisées par le système en interaction avec les utilisateurs qui renvoient des résultats observables et utiles pour ces utilisateurs.

Nous verrons dans le tableau ci-dessous les cas d'utilisation recensés pour notre système.

PARTIE 2 : ANALYSE ET CONCEPTION

TABLEAU 8: RECAPITULATIF DES CAS D'UTILISATION

Cas d'utilisation		Acteurs
S'authentifier		Tous les utilisateurs
Gérer les employés	Ajouter des employés	Chargé des RH
	Désactiver des employés	
	Modifier des employés	
	Consulter les profils des employés	
	Consulter la liste des employés	
Gérer les contrats et les types de postes, de contrats	Ajouter	Chargé des RH
	Modifier	
	Attribuer aux employés	
Gérer les rôles	Modifier un rôle	Chargé des RH
	Attribuer rôle	
	Ajouter un rôle	
Gérer les dossiers des ressources humaines des employés	Ajouter dossiers RH	Chargé des RH
	Créer dossier	
Gérer les services	Ajouter service Modifier service Attribuer service	Chargé des RH

PARTIE 2 : ANALYSE ET CONCEPTION

Gérer les congés	Valider ou refuser une demande de congés	Chargé des RH, Supérieur hiérarchique
Demander des congés	Créer une demande de congés	Tous les utilisateurs
Gérer les objectifs	Créer un objectif	Chargé des RH
	Modifier un objectif	
	Affecter un objectif	
Gérer les campagnes d'évaluation	Lancer une campagne d'évaluation	Chargé des RH
Procéder à des évaluations	Evaluer les employés suivant leurs objectifs	Supérieur hiérarchique
Consulter les évaluations		Employé, Supérieur hiérarchique et chargé des RH
Consulter les statistiques et les états		Directeur exécutif, Responsable des affaires financières, Chargé des RH, Supérieur hiérarchique
Consulter les informations de la totalité les employés		Directeur Exécutif, Responsable des affaires financières, Chargé des RH

2.3.2.1.2 Représentation des diagrammes de cas d'utilisation

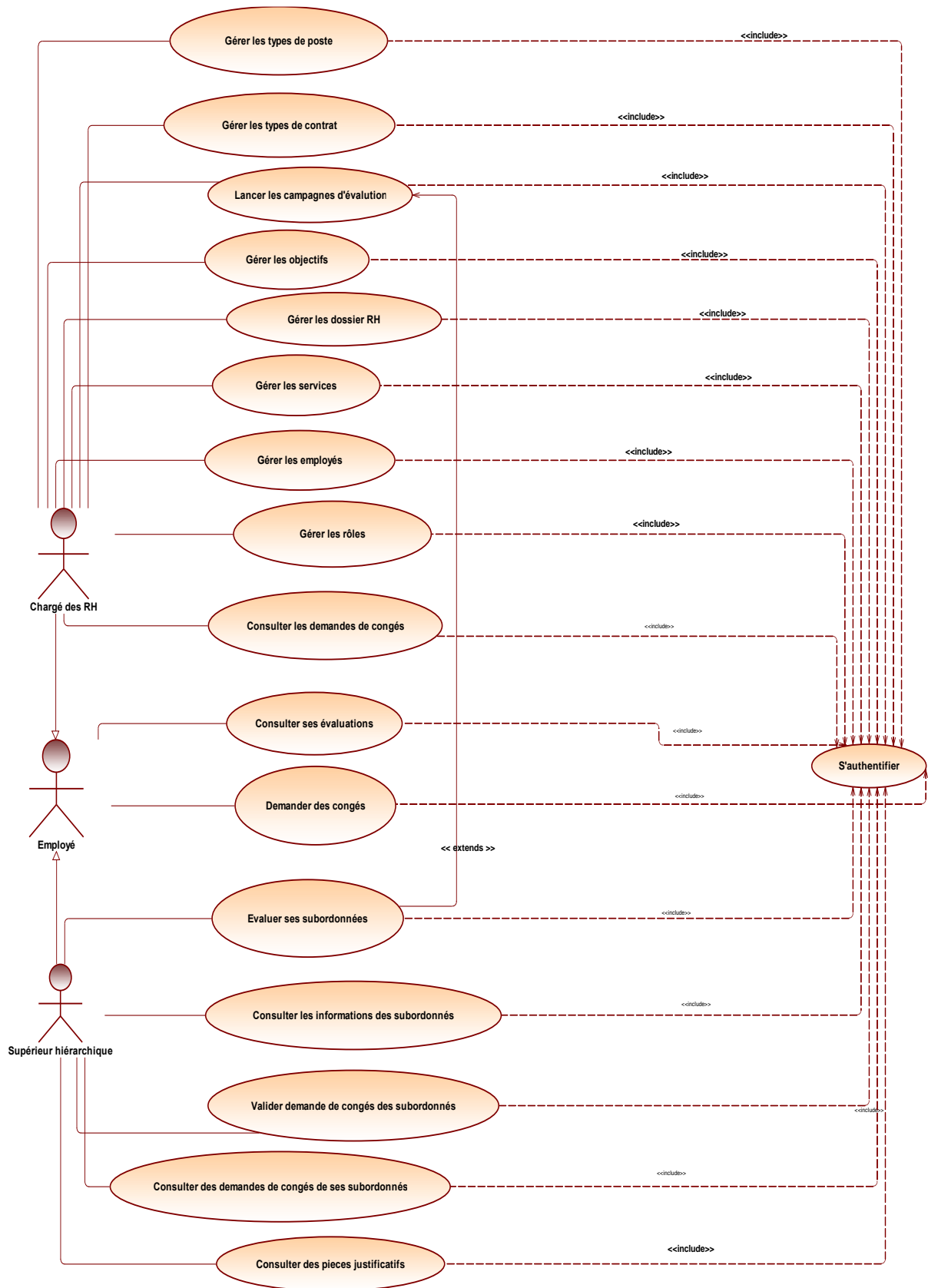


FIGURE 5 DIAGRAMME DE CAS D'UTILISATION GENERAL

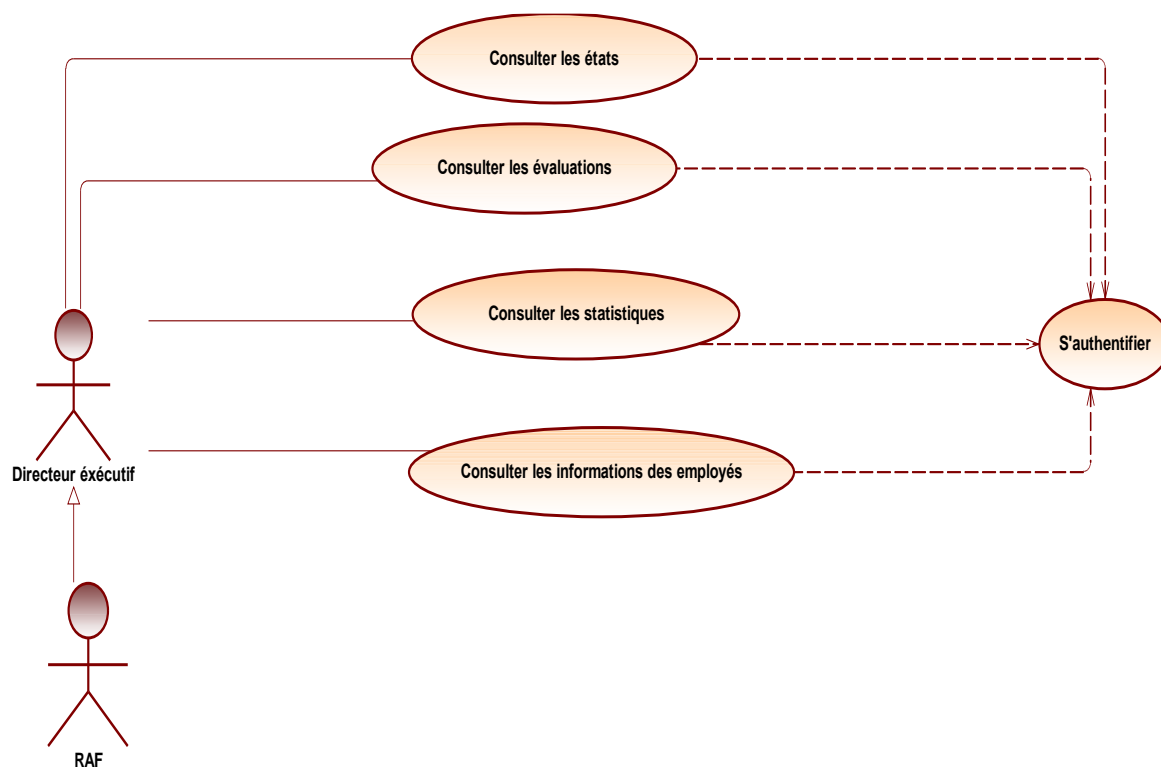


FIGURE 6: DIAGRAMME DE CAS D'UTILISATION DU DIRECTEUR EXECUTIF ET DU RAF

2.3.2.1.3 Description textuelle de quelques cas d'utilisation

2.3.2.1.3.1 Cas d'utilisation « S'authentifier »

Sommaire d'identification	
Résumé : Permet à un utilisateur de se connecter à l'application web	
Acteurs : Tous les utilisateurs	Date de création : 19/7/2021
Responsable : AYITE Yawo M.G.F	Version : 1.0
DESCRIPTION DES ENCHAINEMENTS	
Préconditions	
<ul style="list-style-type: none">✓ L'API est fonctionnelle✓ L'application web est fonctionnelle✓ Le client n'a pas accès aux ressources	
Scénario nominal	
<ul style="list-style-type: none">1) L'utilisateur saisit ses identifiants de connexion2) L'application envoie les identifiants à l'API3) L'API vérifie les informations fournies4) Les informations sont correctes5) L'API vérifie les autorisations6) L'API génère le token de connexion et l'envoie à l'application7) L'application enregistre le token8) L'utilisateur est redirigé vers la page d'accueil	
Enchaînements alternatifs	
<ul style="list-style-type: none">✓ A1 : L'utilisateur ne saisit pas ses informations de connexion ainsi il ne peut être connecté à l'application, ni accéder aux ressources	
Enchaînements d'erreurs	
<ul style="list-style-type: none">✓ E1 : Les identifiants saisis par l'utilisateur sont incorrects. Le système renvoie un message d'erreur. Ce scénario se déroule au point 4✓ E2 : L'utilisateur ne dispose pas des autorisations suffisantes. Un message d'erreur est retourné. Ce scénario se déroule au point 5	
Post conditions	
<ul style="list-style-type: none">✓ L'utilisateur a accès à l'application	

2.3.2.1.3.2 Cas d'utilisation « Evaluer »

Sommaire d'identification	
Résumé : Permet à un supérieur hiérarchique d'évaluer ses subordonnés	
Acteurs : Supérieur hiérarchique	Date de création : 05/08/2021
Responsable : AYITE Yawo M.G.F	Version : 2.0
DESCRIPTION DES ENCHAINEMENTS	
Préconditions	
<ul style="list-style-type: none"> ✓ L'application web et l'API sont fonctionnels ✓ L'utilisateur est connecté ✓ Le token est valide 	
Scénario nominal	
<ol style="list-style-type: none"> 1) Le supérieur sélectionne l'employé à évaluer 2) L'application envoie une requête pour récupérer les informations de l'employé 3) Le supérieur remplit et valide le formulaire d'évaluation 4) La requête est envoyée à l'API 5) L'API vérifie s'il y a une campagne en cours 6) L'API vérifie les informations envoyées 7) L'évaluation est enregistrée 8) Un email de notification est envoyé à l'employé évalué 	
Enchaînements alternatifs	
<ul style="list-style-type: none"> ✓ A1 : Le token généré à la connexion a expiré. L'API renvoie un message d'erreur. Le scénario se déroule au point 1. ✓ A2 : L'utilisateur ne dispose pas des permissions requises. L'API renvoie un message d'erreur. Le scénario se déroule au point 1. 	
Enchaînements d'erreurs	
<ul style="list-style-type: none"> ✓ E1 : Les informations de l'employé ne sont pas disponibles. L'api renvoie un message d'erreur. Ce scénario se déroule au point 3 ✓ E2 : Il n'y a pas de campagne en cours. L'API renvoie un message d'erreur. Le scénario se déroule au point 5 ✓ E3 : Les informations envoyées sont incorrectes. L'API renvoie un message d'erreur. Ce scénario se déroule au point 6 	
Post conditions	
<ul style="list-style-type: none"> ✓ L'évaluation est enregistrée ✓ L'employé est notifié par email 	

2.3.2.1.3.3 Cas d'utilisation « Faire une demande de congés »

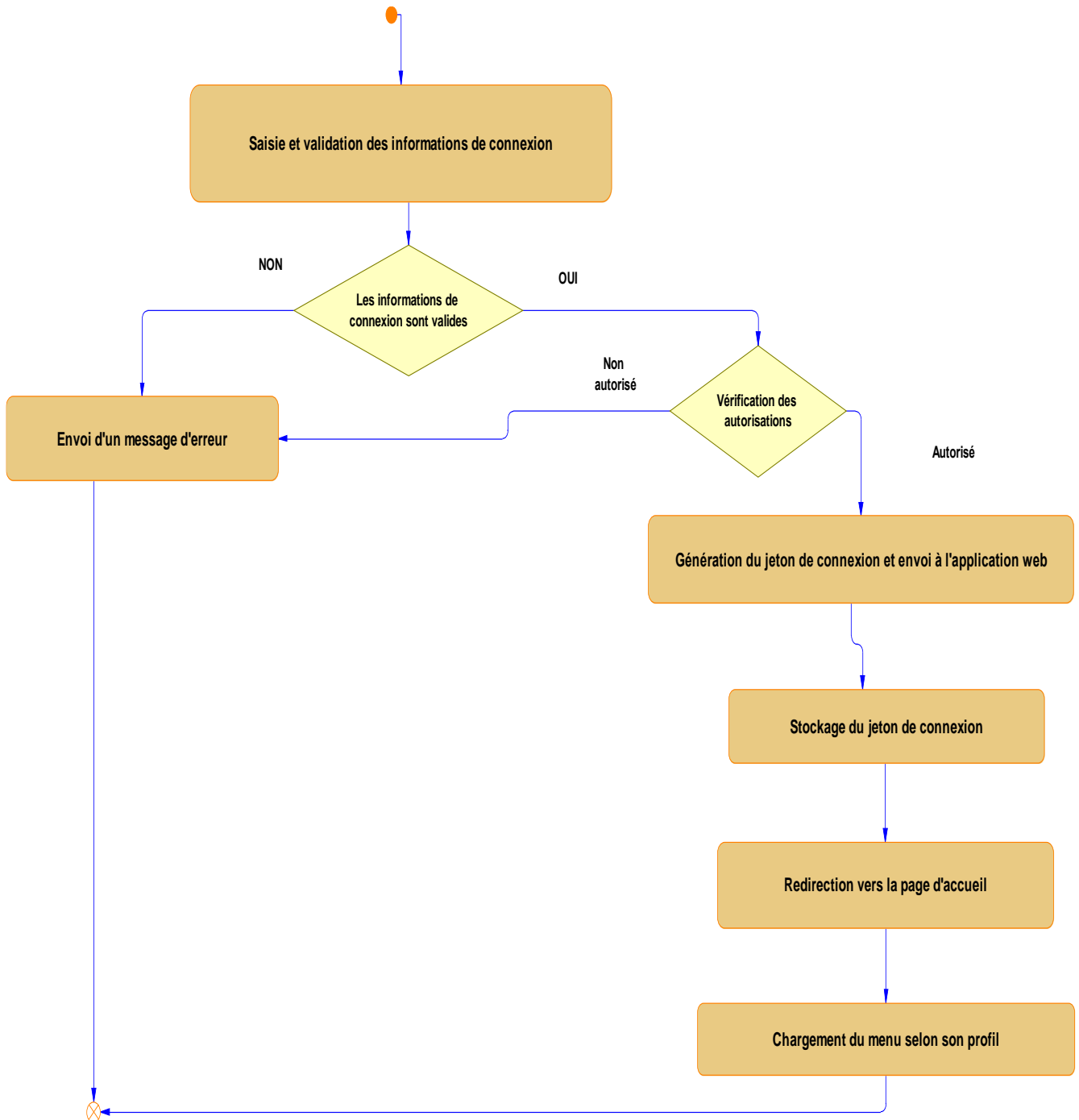
Sommaire d'identification	
Résumé : Permet à un utilisateur de faire une demande de congés	
Acteurs : Tous les employés	Date de création : 20/07/200
Responsable : AYITE Yawo M.G.F	Version : 1.0
Description des enchaînements	
Préconditions	
<ul style="list-style-type: none"> ✓ L'application web et l'API sont fonctionnels ✓ L'utilisateur est connecté ✓ Le token est valide 	
Scénario nominal	
<ol style="list-style-type: none"> 1) L'utilisateur renseigne et valide le formulaire de demande de congés 2) La requête est envoyée vers l'API 3) L'API vérifie les informations 4) La demande est enregistrée 5) Un email de notification est envoyé au supérieur hiérarchique de l'employé 6) Un message de succès est renvoyé par l'API 	
Enchaînements alternatifs	
<ul style="list-style-type: none"> ✓ A1 : Le token généré à la connexion a expiré. L'API renvoie un message d'erreur. Le scénario se déroule au point 1. ✓ A2 : L'utilisateur ne dispose pas des permissions requises. L'API renvoie un message d'erreur. Le scénario se déroule au point 1. 	
Enchaînements d'erreurs	
<ul style="list-style-type: none"> ✓ E1 : Les informations saisies sont incorrectes. L'API renvoie un message d'erreur. Ce scénario se déroule au point 3 	
Post conditions	
<ul style="list-style-type: none"> ✓ La demande est enregistrée ✓ Le supérieur hiérarchique est notifié par email 	

2.3.2.2 Les diagrammes d'activité

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et décrire un flux de travail.

2.3.2.2.1 Diagrammes d'activité « S'authentifier »

FIGURE 7: DIAGRAMME D'ACTIVITE "S'AUTHTIFIER"



2.3.2.2.2 Diagrammes d'activités « Faire une demande de congés »

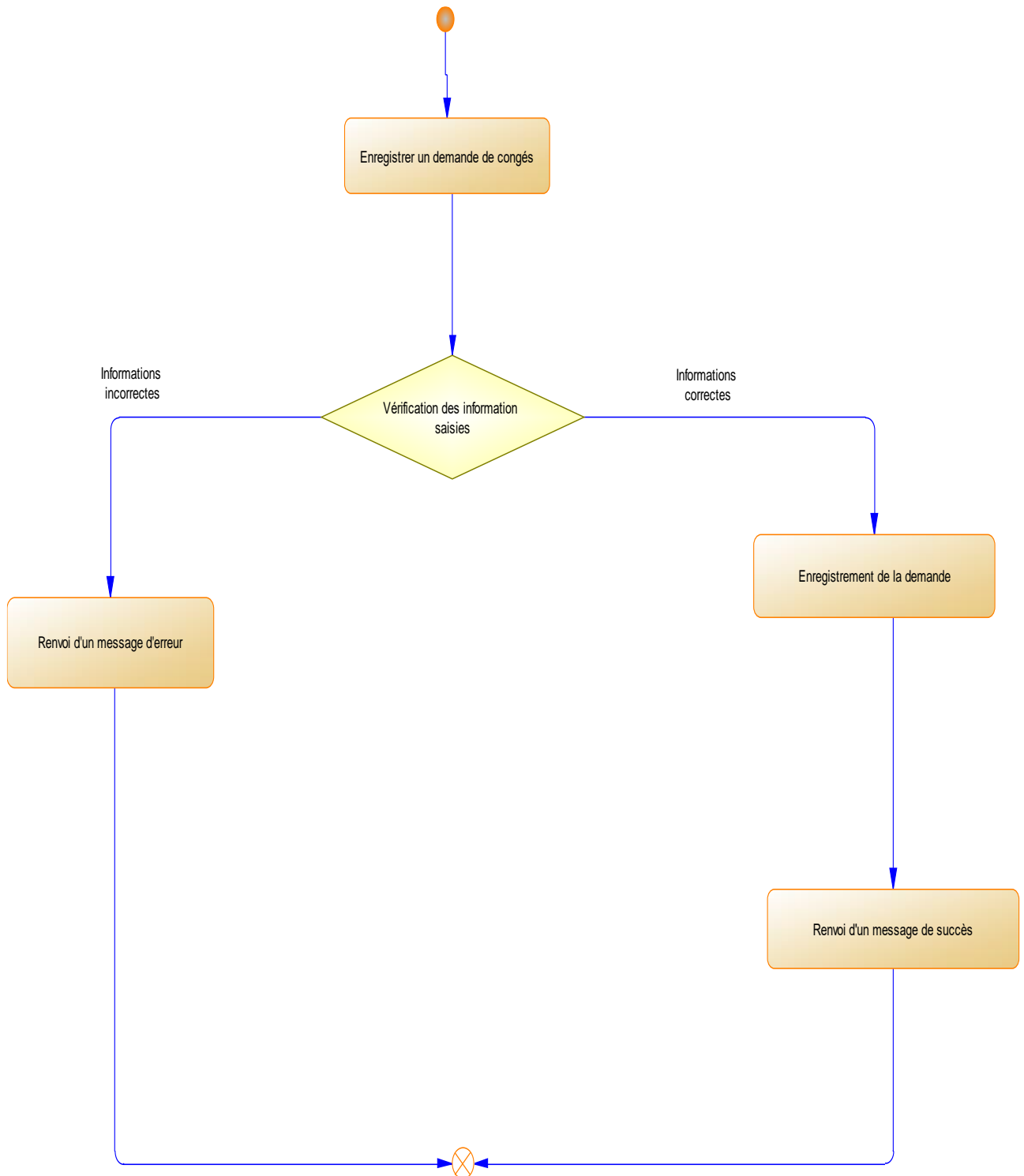


FIGURE 8: DIAGRAMME D'ACTIVITE "FAIRE UN DEMANDE DE CONGES"

2.3.2.2.3 Diagramme d'activité « Evaluer un employé »

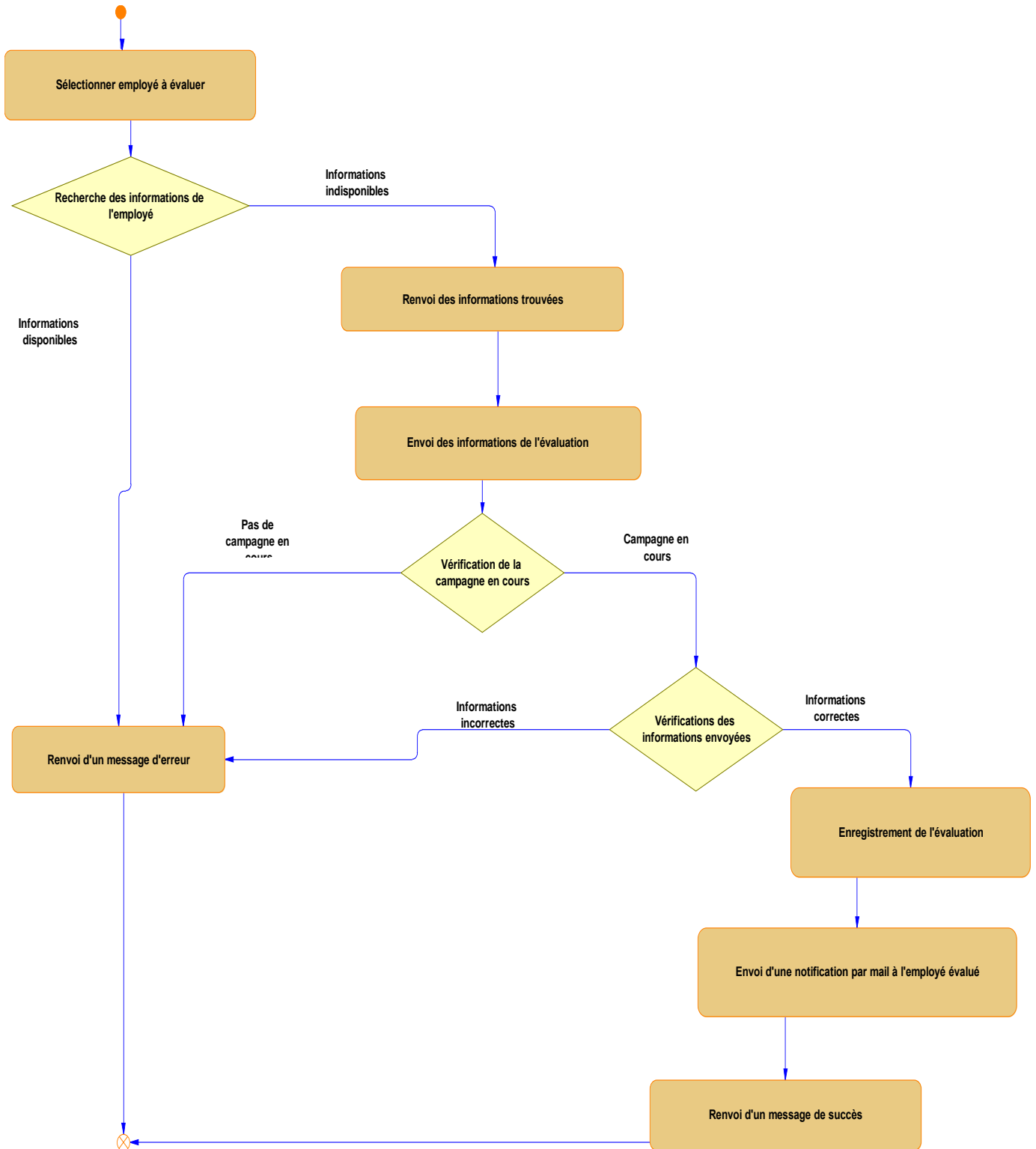


FIGURE 9: DIAGRAMME D'ACTIVITE "EVALUER UN EMPLOYE"

2.3.2.2.4 Diagrammes d'activité « Lancer une campagne

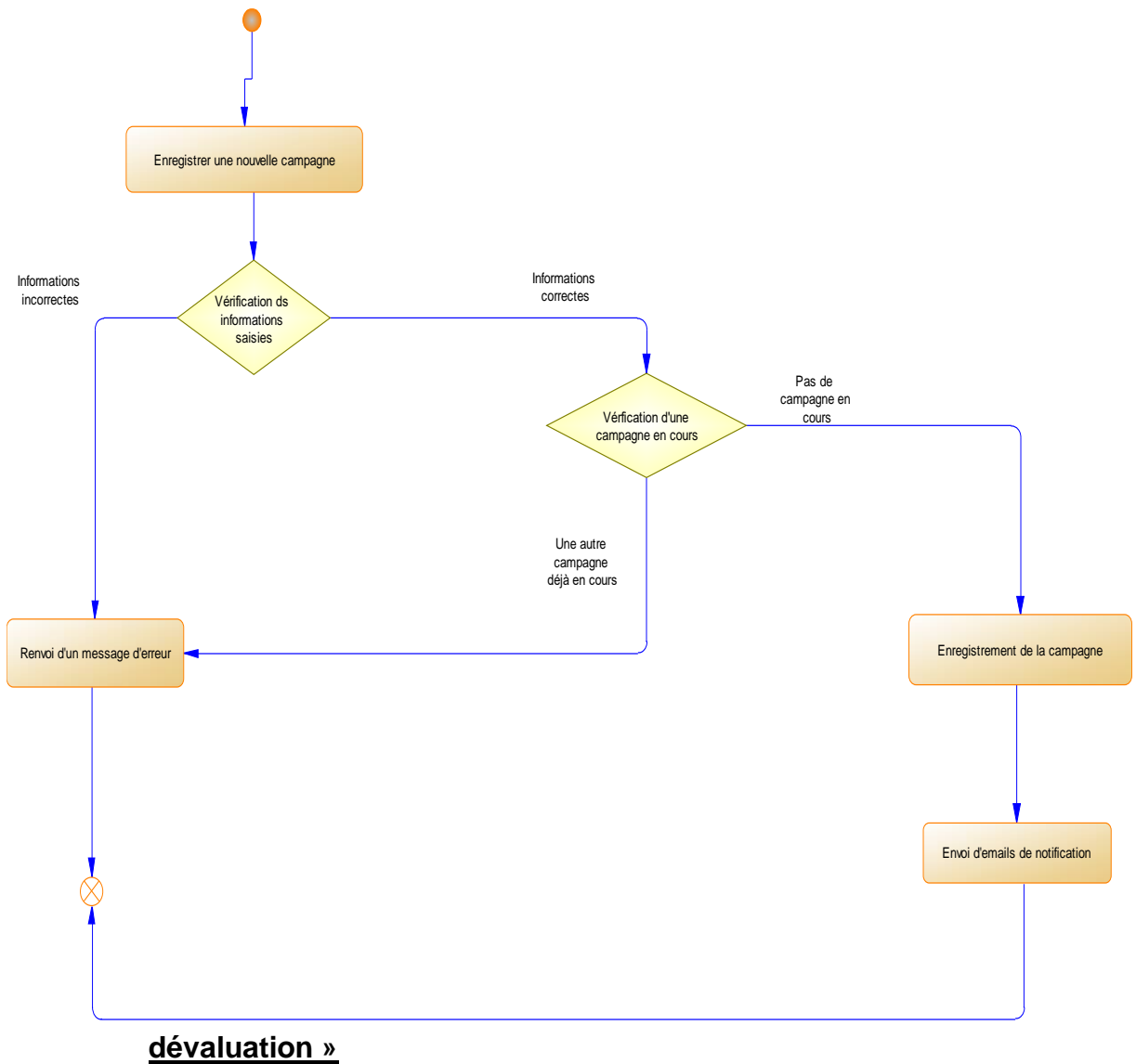


FIGURE 10: DIAGRAMMES D'ACTIVITE "LANCER UNE CAMPAGNE D'EVALUATION"

2.3.2.3 Les diagrammes de séquences

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language.¹⁹

¹⁹ Source : https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence

PARTIE 2 : ANALYSE ET CONCEPTION

2.3.2.3.1 Diagrammes de séquences « S'authentifier »

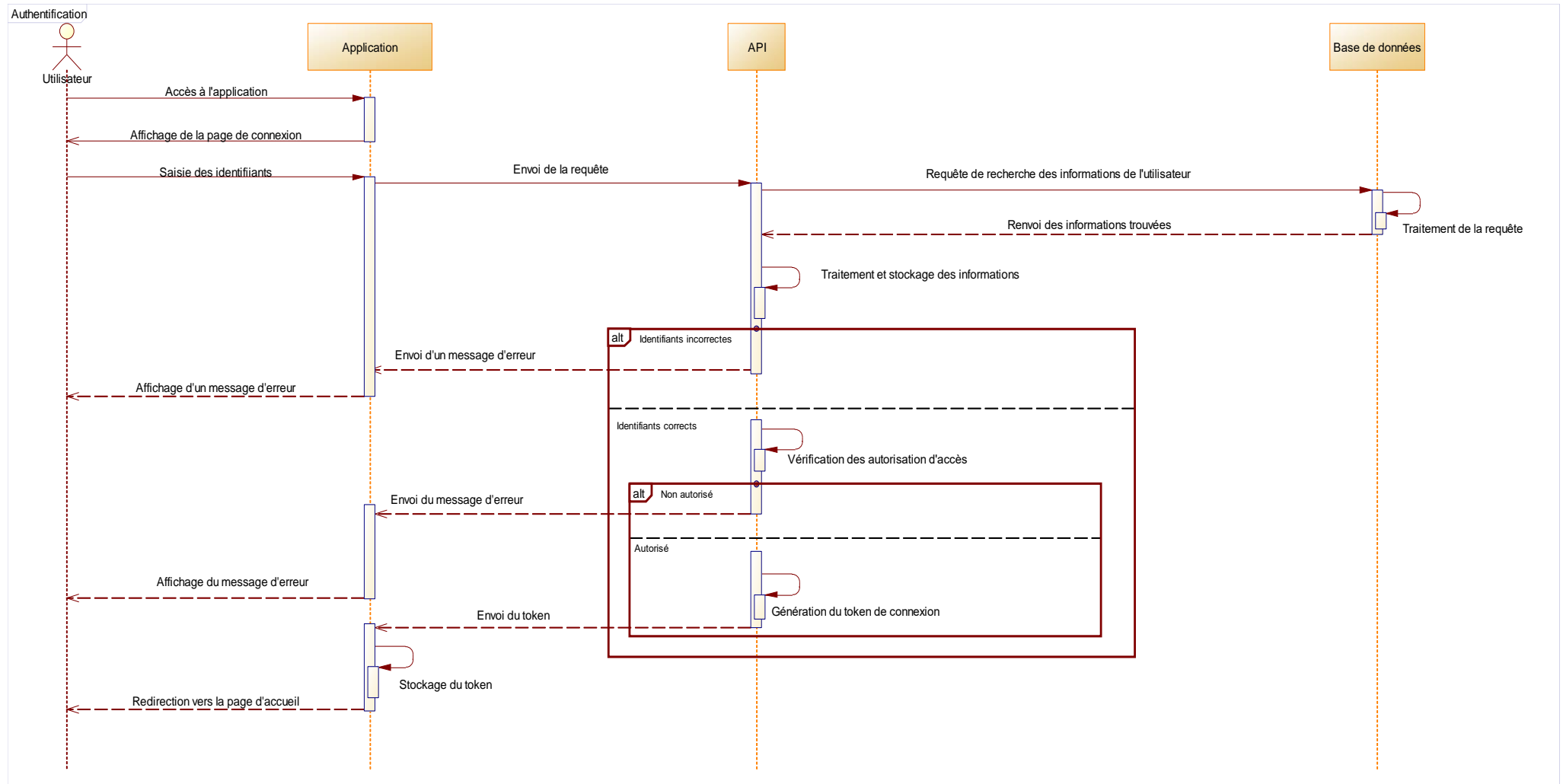


FIGURE 11: DIAGRAMME DE SEQUENCE « S'AUTHTIFIER »

PARTIE 2 : ANALYSE ET CONCEPTION

2.3.2.3.2 Diagramme de séquences « Faire une demande de congés »

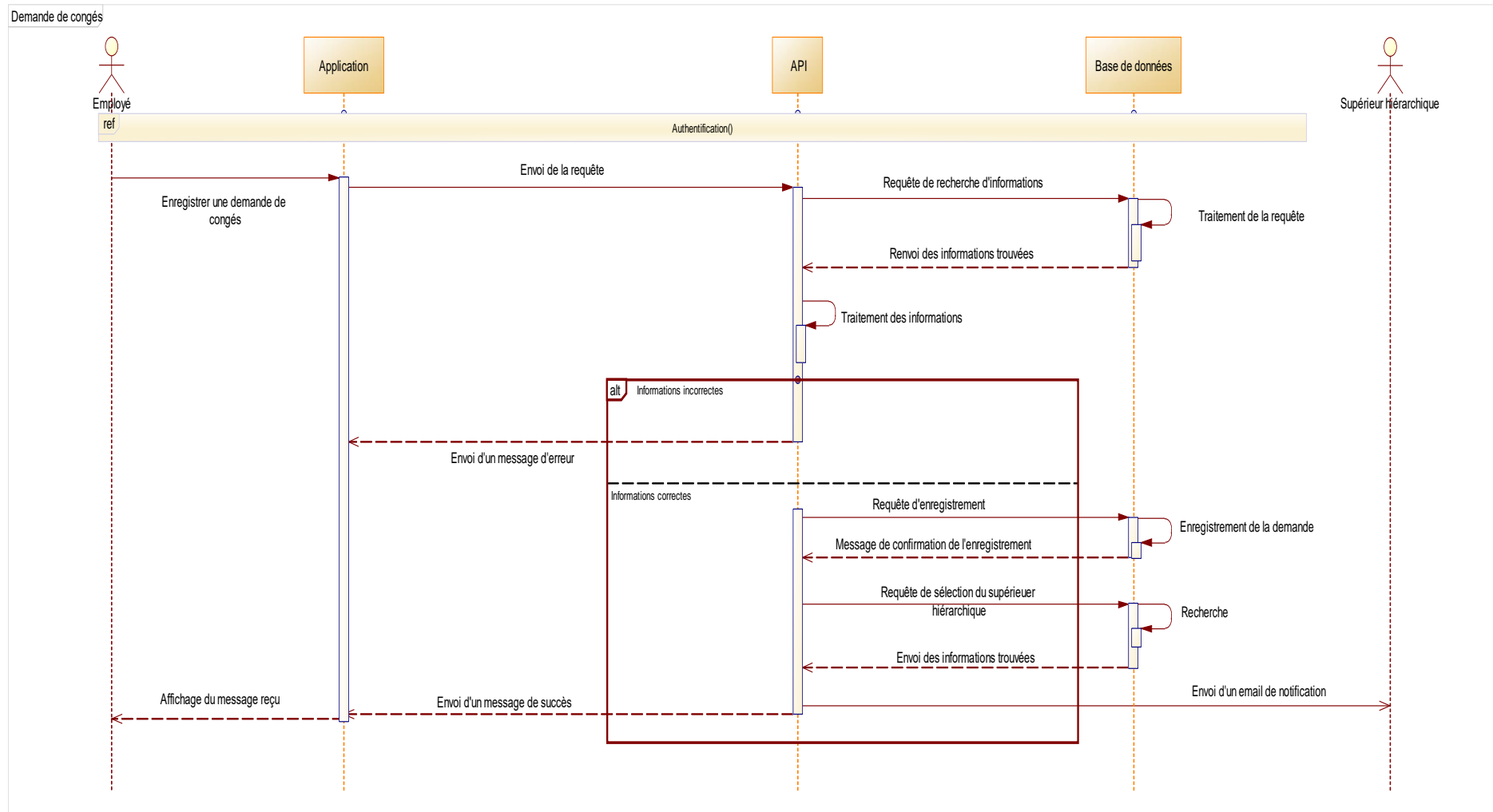


FIGURE 12: DIAGRAMME DE SEQUENCE « FAIRE UNE DEMANDE DE CONGES »

PARTIE 2 : ANALYSE ET CONCEPTION

2.3.2.3.3 Diagramme de séquence « Valider une demande de congés »

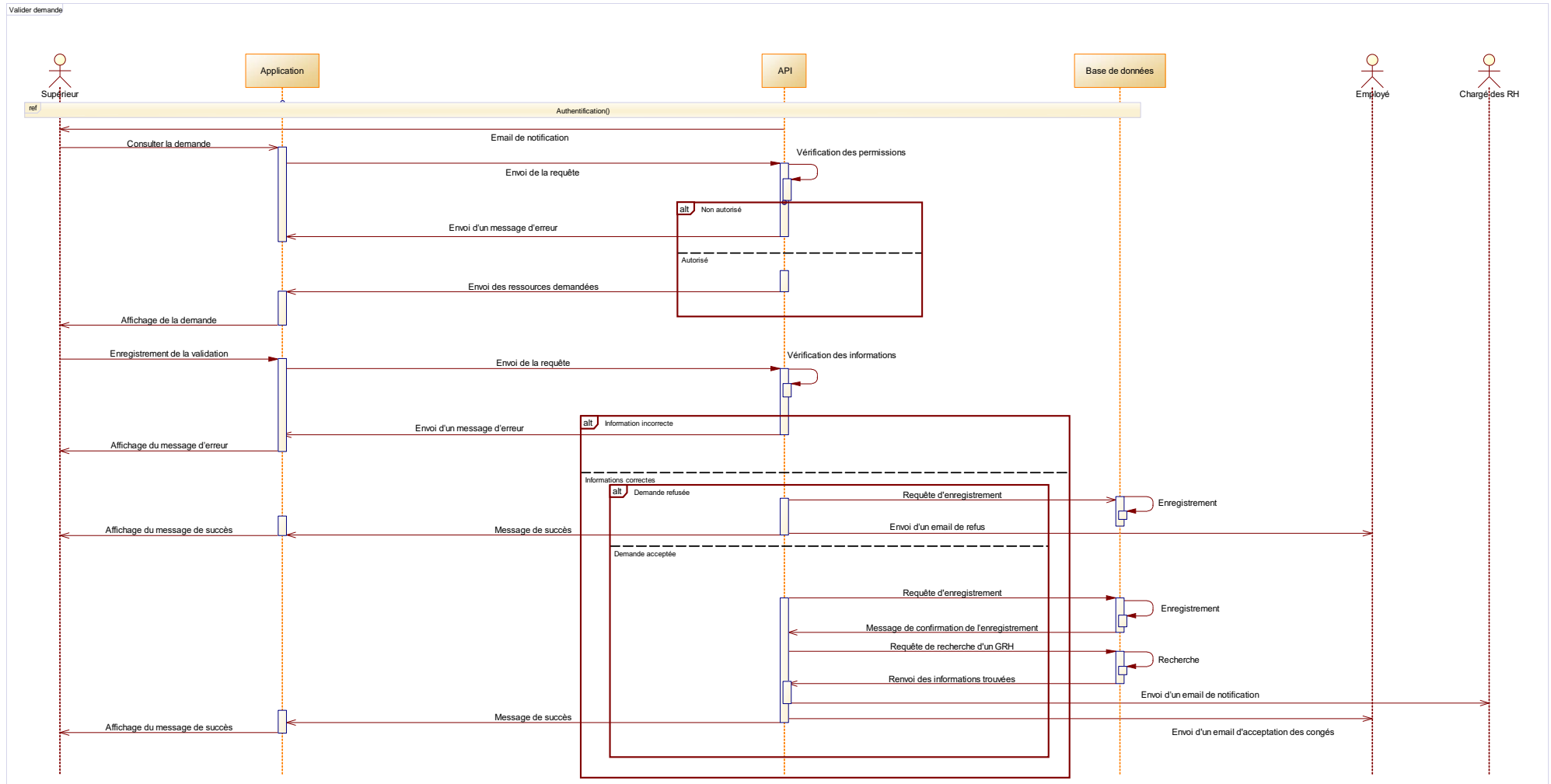


FIGURE 13: DIAGRAMME DE SEQUENCE VALIDER UNE DEMANDE DE CONGES (SUPERIEUR HIERARCHIQUE)

2.3.2.3.4 Diagramme de séquence « Lancer une campagne d'évaluation »

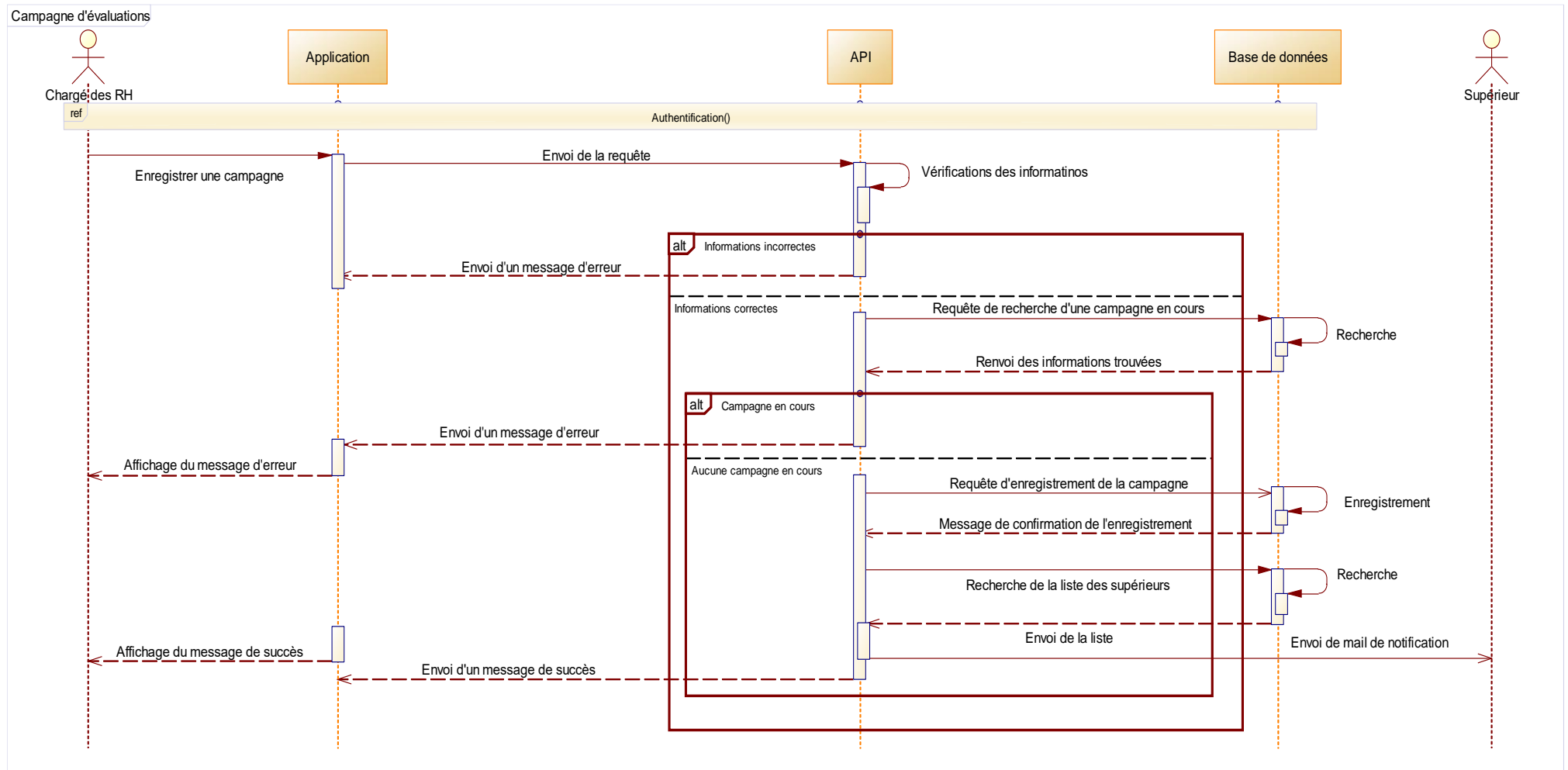


FIGURE 14:DIAGRAMME DE SEQUENCE LANCER UNE CAMPAGNE D'EVALUATION

2.3.2.3.5 Diagramme de séquence « Evaluer un employé »

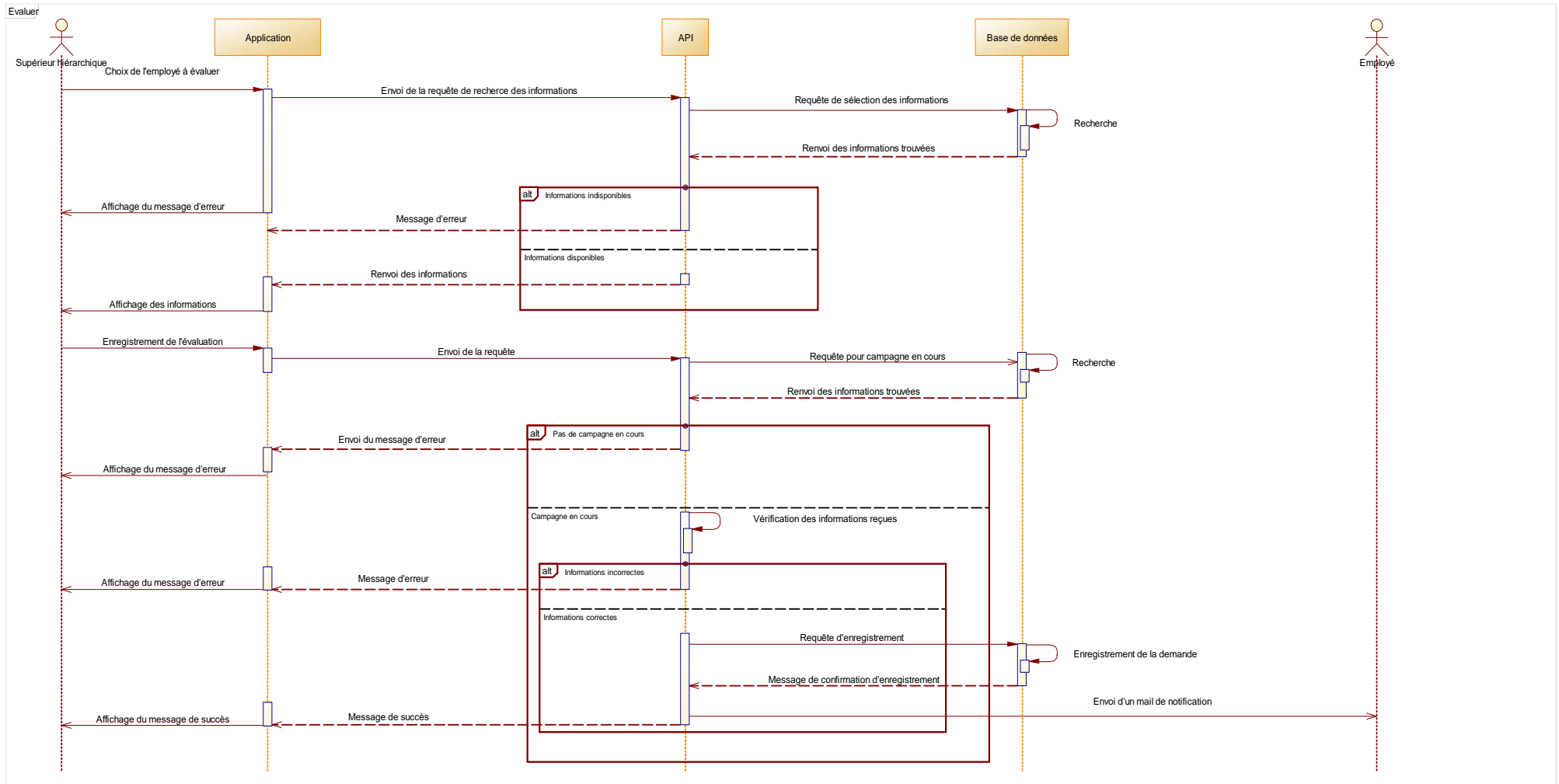


FIGURE 15: DIAGRAMME DE SEQUENCE EVALUER UN EMPLOYE

2.3.3 Capture des besoins techniques

La capture des besoins représente la première étape de la branche droite du processus 2TUP.

Elle permet de compléter la capture des besoins fonctionnels de la branche droite dudit processus en définissant des contraintes indépendantes des utilisateurs que l'application se doit de prendre en compte comme par exemple les contraintes de performance ou de développement.

Recensement des cas d'utilisation techniques

Un cas d'utilisation technique est une séquence d'actions produisant une valeur ajoutée opérationnelle ou purement technique.

Ces derniers sont indépendants des cas d'utilisation obtenus au niveau de la capture des besoins fonctionnels.

- Les données doivent être cohérentes ;
- Les notifications par emails doivent être envoyées de façon rapide ;
- Les ressources doivent être disponibles à tout moment ;
- Plusieurs utilisateurs doivent pouvoir utiliser le service en même temps ;
- Les ressources sensibles du système doivent être protégées par la mise en place de rôles et de permissions ;
- L'utilisateur doit travailler avec des entités sous forme d'objets. Cela nécessite la mise en œuvre de processus de gestion du cycle de vie et de la persistance des objets.

2.3.4 Conception

2.3.4.1 Diagramme de déploiement

C'est un diagramme statique d'UML qui sert à représenter l'utilisation de l'infrastructure physique du système et la manière dont les composants du système sont répartis ainsi que les relations entre eux.²⁰

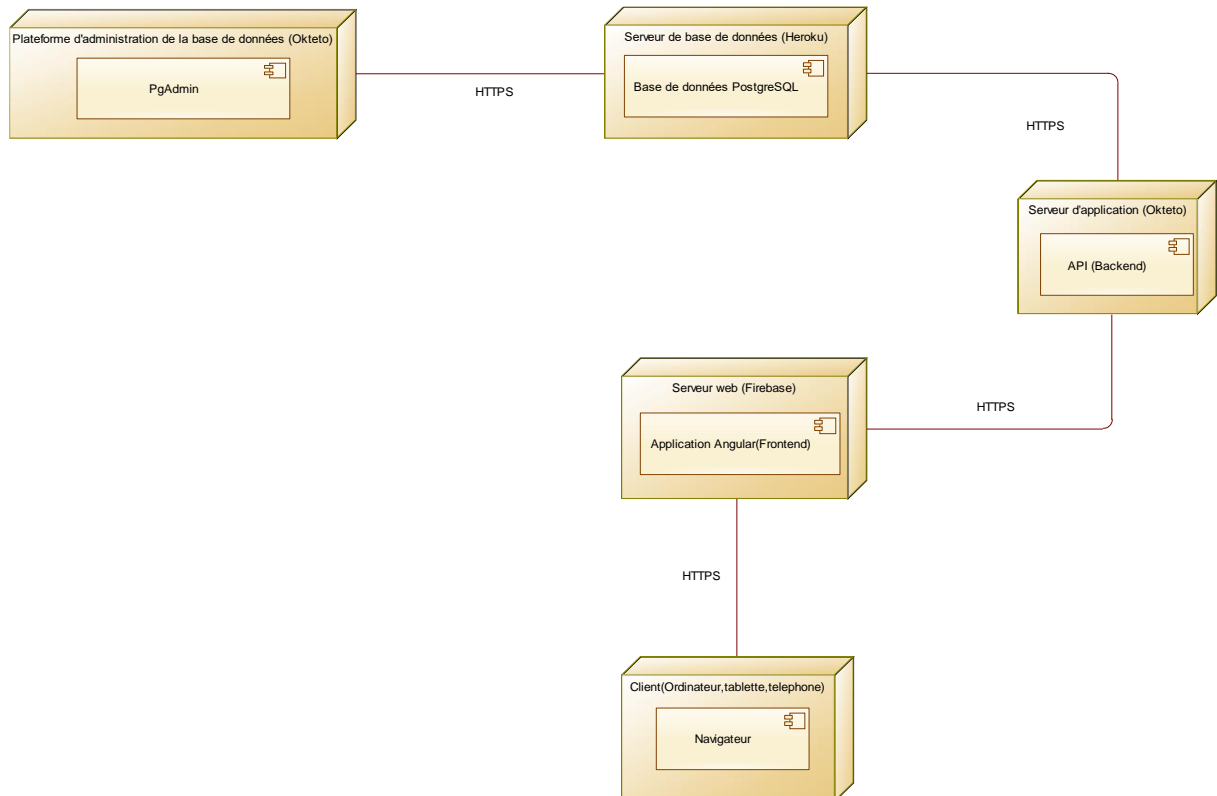


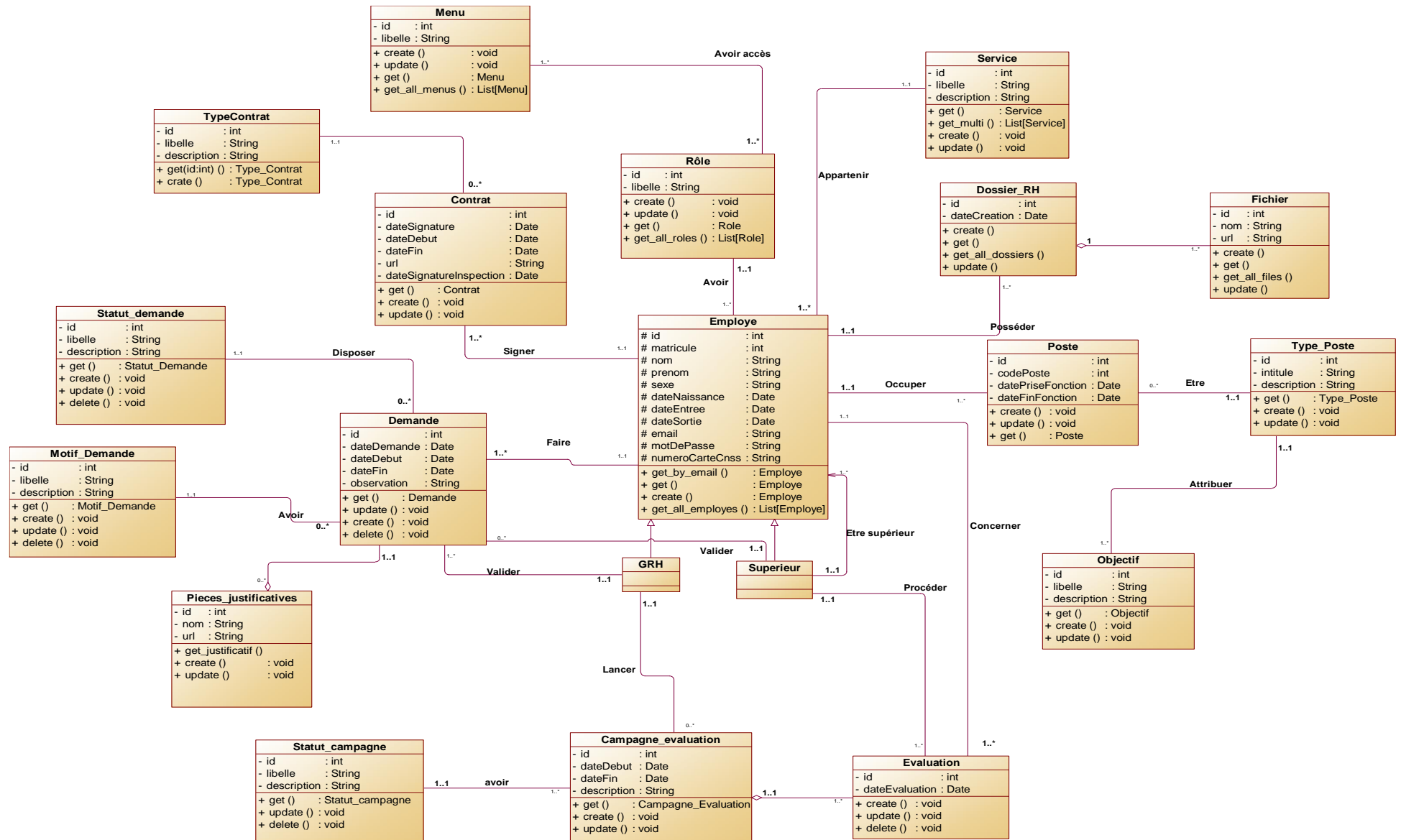
FIGURE 16: DIAGRAMME DE DEPLOIEMENT

2.3.4.1 Diagramme de classes

Le diagramme de classes est un diagramme statique d'UML utilisé pour représenter les classes et les interfaces des systèmes de même que leurs relations.

²⁰ Source : https://fr.wikipedia.org/wiki/Diagramme_de_d%C3%A9ploiement

PARTIE 2 : ANALYSE ET CONCEPTION



PARTIE 2 : ANALYSE ET CONCEPTION

FIGURE 17: DIAGRAMME DE CLASSE

PARTIE 3 : REALISATION ET MISE EN ŒUVRE

La phase de mise en œuvre est celle qui suit immédiatement l'analyse et la conception. Dans cette partie, nous ferons une brève présentation des différents outils tant matériels que logiciels que nous avons eu à utiliser durant la phase de programmation de notre stage.

3.1 Matériels et logiciels utilisés

3.1.1 Matériels utilisés

❖ Ordinateur portable

Notre système a été réalisé avec un ordinateur portable Pavilion 15 bc-506nf de marque HP (Hewlett Packard) disposant des caractéristiques²¹ suivantes :

- Système d'exploitation : Microsoft Windows 10 Professionnel ;
- Type du système d'exploitation : Système d'exploitation 64 bits ;
- Processeur : Intel(R) Core (TM) i5-9300H CPU @ 2.40GHz 2.40 GHz ;
- Mémoire RAM installée : 8,00 Go ;
- Capacité du disque dur : 477 Go ;
- Type du disque dur : SSD PCIe® NVMe™ M.2 ;
- Cartes graphiques : NVIDIA Pascal GeForce GTX 1050 3 Go GDDR5 dédiés, Intel UHD 630 intégrée au processeur et Optimus ;
- Dalle: 15.6" Full HD LED (1920x1080, antireflet/mate).

❖ Moniteur

Nous avons eu à utiliser un moniteur car celui-ci a favorisé un gain en termes de productivité tout en nous permettant de réaliser plus aisément les recherches et les débogages. Les caractéristiques²² du moniteur sont les suivantes :

- Fabricant : Hewlett Packard (HP) ;
 - Modèle: HP ZR2240w 21.5-inch LED Backlit IPS;
 - Résolution optimale : 1920 x 1080 (Full HD) à 60 Hz ;
 - Zone d'image visible(diagonale) : Écran large de 54,52 cm (21,5 po)
- ;

²¹ Source : <https://www.laptopspirit.fr/249100/hp-pavilion-15-bc506nf-15-pouces-jeu-sobre-avec-pack.html>

²² Source : <https://support.hp.com/th-en/document/c03003776>

3.1.2 Logiciels utilisés

Nous présentons dans cette section, les différents logiciels et technologies que nous avons utilisé tout au long de notre stage.

3.1.2.1 Backend

Le backend est un terme généralement utilisé pour décrire les ressources tant matérielles que logicielles mises en œuvre en arrière-plan qui alimentent une application ou un site web.

Il comporte un grand nombre d'éléments comme les serveurs d'application, de base de données, et les langages de programmation.

3.1.2.1.1 Python

Python est un langage de programmation puissant et facile à apprendre. Il dispose de structures de données de haut niveau et permet une approche simple mais efficace de la programmation orientée objet.²³

❖ Justifications

- Facile à prendre en main ;
- Rapide ;
- Syntaxe légère ;
- Compatible avec la plupart des systèmes et plateformes.

❖ Alternatives

- Php ;
- Javascript ;
- Java ;

²³ Source : <https://docs.python.org/fr/3/tutorial/>

3.1.2.1.2 FastAPI

FastAPI est un framework Web moderne, rapide permettant de créer des API avec Python 3.6 ou plus.²⁴

❖ **Justifications**²⁵

- Supporté par les éditeurs de code ;
- Facile à prendre en main ;
- Moins de temps de débogage ;
- Rapide ;
- Facile à apprendre ;
- Dispose d'une documentation détaillée ;

❖ **Alternatives**²⁶

- Django Rest Framework ;
- Flask Rest Framework ;
- Tornado ;

3.1.2.1.3 Pydantic

Pydantic est un outil permettant la validation des données et la gestion de paramètres basé sur le langage de programmation python.

❖ **Justifications**

- Permet de valider les données entrées ;
- Permet de définir le format des données retournées lors de l'appel des endpoints de l'API ;

❖ **Alternatives**²⁷

- Pygmento ;
- Sqlparse ;
- Pyparsing ;
- Lak ;

²⁴ Source : <https://fastapi.tiangolo.com/>

²⁵ Source : <https://fastapi.tiangolo.com/>

²⁶ Source : <https://nordicapis.com/8-open-source-frameworks-for-building-apis-in-python/>

²⁷ Source : <https://python.libhunt.com/pydantic-alternatives>

3.1.2.1.4 SQLAlchemy

SQLAlchemy est un ORM écrit en python qui offre aux développeurs d'application de s'adonner au développement avec tous les outils nécessaires à la communication avec la base de données prédéfinis.

❖ **Justifications**

- Permet d'écrire en utilisant le langage python des classes, qui représente une table dans la base de données. Chaque attribut de la classe représentant une colonne de la table dans la base de données ;
- Facilite les migrations des clés étrangères et les relations entre les différentes tables de la base de données ;
- Fonctionne avec les bases de données relationnelles comme PostgreSQL et Mysql

❖ **Alternatives²⁸**

- Ormar ;
- Django-models ;
- Tortoise orm ;
- Peewee ;

3.1.2.1.5 Pipenv

Pipenv est un outil de packaging qui résout certains problèmes courants associés au flux de travail typique utilisant pip, virtualenv et le requirements.txt.²⁹

❖ **Justifications**

- Permet de gérer les packages et les dépendances des projets avec python ;
- Permet de fixer les versions des dépendances utilisées et de créer facilement un environnement lors du développement ;
- Permet que les dépendances de l'environnement de développement soient identiques à celles de l'environnement de production ;
- Dispose de la gestion d'environnement virtuel intégrée ;

❖ **Alternatives³⁰**

²⁸ Source : <https://python.libhunt.com/sqlalchemy-alternatives>

²⁹ Source : <https://realpython.com/pipenv-guide/>

³⁰ Source : <https://python.libhunt.com/pipenv-alternatives>

- Pyenv ;
- Poetry ;
- Autoenv ;
- Virtualenv ;

3.1.2.1.6 Postman

Postman est un outil essentiel permettant entre autres de tester son API au travers des requêtes http.

❖ **Justifications**³¹

- Facilite le test des endpoints d'une api, avec l'envoi des requêtes http ;
- Permet de construire des API grâce à son API Builder ;
- Il offre la possibilité de simuler une API pour le développement ;

❖ **Alternatives**³²

- Insomnia REST Client ;
- Paw ;
- Hoppscotch ;

3.1.2.1.7 Swagger UI

Swagger UI offre la possibilité de générer une documentation visuelle de l'API, et permet à quiconque d'interagir avec les ressources d'une API sans qu'il n'ait connaissance de la logique implémentée

❖ **Justifications**

- Offre la possibilité de générer la documentation de l'API et de la mettre à disposition d'autres développeurs
- Directement intégré dans le framework FastAPI
- Il couvre la totalité des schémas pydantic et des endpoints d'une api.

❖ **Alternatives**³³

- Apiary ;
- Slate ;

³¹ Source : <https://practicalprogramming.fr/postman>

³² Source : <https://alternativeto.net/software/postman/>

³³ Source : <https://alternativeto.net/software/swagger-io/>

- Readme.io ;
- ReDoc ;

3.1.2.1.8 Uvicorn

Uvicorn est un serveur ASGI (Asynchronous Server Gateway Interface) qui va permettre à notre application de communiquer de manière asynchrone avec le serveur que l'on va déployer.

❖ **Justifications**

- Uvicorn est le serveur recommandé par la documentation officielle pour l'utilisation du framework FastAPI

❖ **Alternatives**³⁴

- Gunicorn

3.1.2.1.9 Google Cloud Storage for Firebase

Google Cloud Storage for Firebase est conçu pour vous aider à stocker et à diffuser rapidement et facilement du contenu généré par les utilisateurs, tel que des photos et des vidéos.

❖ **Justifications**

- Facilité d'utilisation ;
- Il dispose d'une excellente documentation ;
- Il est de confiance et scalable ;
- Permet de sauvegarder n'importe quel type de fichiers.

❖ **Alternatives**³⁵

- Visual FoxPro
- IBM Db2

3.1.2.1.10 Docker

Selon la firme de recherche sur l'industrie 451 Research, « Docker est un outil qui peut emballer une application et ses dépendances dans un conteneur isolé, qui pourra être exécuté sur n'importe quel serveur ».

³⁴ Source : <https://fastapi.tiangolo.com/>

³⁵ Source : <https://www.g2.com/products/google-cloud-storage-for-firebase/competitors/alternatives>

❖ Justifications

- Permet de déployer rapidement et facilement des environnements de serveur dans des containers permettant ainsi de rendre l'appli indépendante du système sur lequel on l'installe ;
- Permet d'isoler l'application dans un container, afin que des paramètres et des dépendances qui lui sont propres n'affectent ni les paramètres ni les installations sur la machine ;
- Permet d'assurer la sécurité en séparant les différents composants d'une application (Si un de ces containers est compromis les autres restent non atteints) ;

❖ Alternatives³⁶

- LXC ;
- Hyper-v containers ;
- Runc ;
- Containerd

3.1.2.1.11 PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle et objet.

❖ Justifications

- Il est libre ;
- Il est plus avancé dans la conformité aux standards SQL
- Il fonctionne sur diverses plates-formes matérielles et sous différents systèmes d'exploitation ;

❖ Alternatives³⁷

- Redis ;
- Mysql ;
- MongoDB ;
- Cubrid ;

³⁶ Source : <https://frog.com/knowledge-base/the-basics-7-alternatives-to-docker-all-in-one-solutions-and-standalone-container-tools/>

³⁷ Source : <https://alternativeto.net/software/postgresql/>

3.1.2.1.12 PgAdmin

PgAdmin est une plateforme Open Source d'administration pour PostgreSQL.

❖ Justifications

- Il dispose d'un large panel de fonctionnalités et est la plateforme d'administration la plus populaire pour PostgreSQL ;
- Il fonctionne sur plusieurs systèmes d'exploitation ;

❖ Alternatives³⁸

- Adminer ;
- Navicat ;
- Valentina Studio ;

3.1.2.1.13 SQLite

SQLite est une bibliothèque écrite en langage C qui propose un moteur de base de données relationnelle accessible par le langage SQL.

Il implémente en grande partie le standard SQL-92 et les propriétés ACID.³⁹

❖ Justificatifs

- Sqlite nous a servi durant tout le développement et permis de disposer d'une base de données locale, nous facilitant ainsi les tests ;
- Il utilise un seul fichier, et python dispose d'un support intégré

3.1.2.1.14 Mailtrap.io

Mailtrap.io est un service qui offre un faux serveur SMTP pour tester les envois d'emails pendant la phase de développement de votre projet.

❖ Justifications

- Permet d'éviter d'envoyer des emails à de vrais clients
- Offre la possibilité de tester l'envoi des emails sur un environnement local

³⁸ Source : <https://alternativeto.net/software/pgadmin/>

³⁹ Source : <https://fr.wikipedia.org/wiki/SQLite>

❖ Alternatives⁴⁰

- Litmus ;
- MailChimp ;
- Mailflow Monitoring ;

3.1.2.1.15 Gmail

Gmail est un service de messagerie électronique proposé par Google.

❖ Justification

- Gmail nous a permis la mise en place et l'envoi des emails grâce à son serveur SMTP.

3.1.2.1.16 Visual Studio Code

Visual Studio Code est un éditeur de code développé par la société Microsoft pour Windows, Linux et MacOS.

❖ Justifications

- Il offre une multitude d'extension permettant de faciliter notamment le contrôle de version, la détection et l'affichage de message d'erreurs, la complétion intelligente de code ou la containerisation des projets

❖ Alternatives⁴¹

- Atom ;
- Pycharm ;
- Notepad ++ ;

3.1.2.2 Frontend

3.1.2.2.1 Angular

Angular est un framework open-source de Google permettant de développer des applications multiplateformes.

❖ Justifications

- Facilité d'apprentissage du framework ;
- Est maintenu par Google ;

⁴⁰ Source : <https://www.g2.com/products/mailtrap/competitors/alternatives>

⁴¹ Source : <https://alternativeto.net/software/visual-studio-code/>

- Offre un grand nombre de bibliothèques facilitant entre autres le management des formulaires et la connexion aux API ;
- Facilité de connexion et de la consommation des API

❖ Alternatives⁴²

- React JS ;
- Vue JS ;
- Aurelia ;

3.1.2.2.2 TypeScript

D'après Wikipédia, TypeScript est un langage de programmation open source développé par Microsoft qui a pour but d'améliorer et de sécuriser la production de code Javascript. Il s'agit d'un sur-ensemble syntaxique strict de JavaScript (c'est-à-dire que tout code JavaScript correct peut être utilisé avec TypeScript).

❖ Justifications

- TypeScript est le langage utilisé par défaut utilisé par le framework Angular ;
- Il permet d'écrire les différents contrôles derrière les composants des applications Angular ;
- Il nous a permis l'écriture des services pour la consommation de l'API

3.1.2.2.3 Npm

Node Package Manager est un outil de gestion et de référencement des packages qui permettent d'ajouter diverses fonctionnalités aux projets Javascript.

❖ Justifications

- Il automatise la gestion des dépendances et des paquets des projets Javascripts ;
- Met à la disposition des développeurs une plateforme fiable (npmjs.com) qui leur permet l'accès à du code open-source

⁴² Source : <https://alternativeto.net/software/angularjs/>

❖ Alternatives⁴³

- Yarn ;
- Turbo ;
- PNPM

3.1.2.2.4 Sass

Selon Wikipédia, SASS (**S**yntactically **a**wesome **s**tylesheets) est un langage de script préprocesseur qui est compilé ou interprété en CSS (Feuilles de styles en cascades).

❖ Justifications

- Dispose d'une large communauté et est bien documenté ;
- Offre des options que le CSS seul ne propose pas comme : l'emploi des variables, boucles, fonctions et héritage.

❖ Alternatives⁴⁴

- Css ;
- Stylus ;
- LESS

3.1.2.2.5 TailwindCSS

TailwindCSS est un framework CSS pour le design graphique de sites et applications web.

❖ Justifications

- Offre l'option de personnalisation et est simple à configurer, on peut définir les thèmes, les polices de texte et les couleurs qu'on souhaite à l'aide d'un fichier de configuration tailwind.config.js ;

⁴³ Source : <https://alternativeto.net/software/npm/>

⁴⁴ Source : <https://alternativeto.net/software/sass/>

PARTIE 3 : REALISATION ET MISE EN OEUVRE

- Le design final ne ressemble pas à un résultat obtenu grâce à un framework CSS comme Bootstrap

❖ Alternatives⁴⁵

- Bootstrap ;
- Chakra UI ;

3.1.2.2.6 Node.JS

Node.js est une plateforme logicielle libre qui permet l'exécution de JavaScript côté serveur

❖ Justifications

- Il est rapide et dispose d'une grande communauté active ;
- Il est évolutif et extensible ;

❖ Alternatives

- Deno

3.1.2.2.7 WebStorm

Selon Wikipédia, WebStorm est un IDE pour les langages Web (HTML, CSS et JavaScript) développé par l'entreprise JetBrains, basé sur la plateforme IntelliJ IDEA.

❖ Justifications

- C'est un éditeur spécialisé en JavaScript qui offre un grand nombre de fonctionnalités dont la refactorisation de code.

❖ Alternatives⁴⁶

- Vim ;
- Brackets ;

⁴⁵ Source : <https://alternativeto.net/software/tailwind-css/>

⁴⁶ Source : <https://easypartner.fr/blog/les-meilleurs-ide-javascript/>

3.1.2.3 Gestion de version

3.1.2.3.1 Git

Git est un logiciel libre et open source de gestion de versions. Il est utilisable via un terminal ou une application desktop disposant d'une interface utilisateur graphique.

❖ **Justifications**

- Permet de conserver un historique des modifications et des versions de tous les fichiers ;
- Rend facile le retour en arrière en cas de problème ;
- Offre la possibilité de travailler à plusieurs sur un même projet sans risquer de supprimer les modifications des autres collaborateurs

❖ **Alternatives⁴⁷**

- Azure Devops Server ;
- Helix Core ;
- AWS Code Commit

3.1.2.3.2 Github

Selon Wikipédia, est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git.

❖ **Justifications**

- Permet d'avoir un historique des modifications et des versions de tous les fichiers et de les sauvegarder dans le cloud
- Permet de suivre les changements dans le code à chaque version

❖ **Alternatives⁴⁸**

- GitLab ;
- Bitbucket ;

⁴⁷ Source : <https://alternativeto.net/software/git/>

⁴⁸ Source : <https://alternativeto.net/software/github/>

3.1.2.4 Choix des hébergeurs

3.1.2.4.1 Okteto (API & PGADMIN)

Okteto est un service d'hébergement cloud basé sur la technologie Kubernetes, il convient pour le déploiement de conteneurs génériques qui requiert une configuration minimale.

❖ Justifications

- Il offre la possibilité d'avoir des endpoints HTTPS pour les services déployés ;
- Il permet aussi de déployer son application à partir d'un répertoire git, ou des fichiers Docker compose ;

❖ Alternatives⁴⁹

- Vercel ;
- Amazon Web Service (AWS) ;

3.1.2.4.2 Firebase (Frontend)

Firebase est un service d'hébergements proposé par Google.

❖ Justifications

- Dispose d'une documentation complète et est facile à utiliser

3.1.2.4.3 Heroku (Bases de données)

Heroku est une PaaS (Platform as A Service) qui permet aux développeurs de créer, manipuler et utiliser des applications totalement dans le cloud.

❖ Justifications

- Il nous a permis d'héberger notre base de données PostgreSQL dans le cloud ;
- Il possède une excellente offre d'hébergement, peu coûteux

❖ Alternatives⁵⁰

- OpenShift ;
- Linode ;

⁴⁹ Source : <https://stackshare.io/okteto/alternatives>

⁵⁰ Source : <https://www.saashub.com/heroku-alternatives>

3.2 Architectures matérielle et logicielle de l'application

3.2.1 Architecture matérielle

L'architecture de notre système est essentiellement constituée de serveurs hébergés dans le **Cloud**. Ces derniers, communiquent entre eux à l'aide du protocole HTTPS afin d'assurer la sécurité des informations transmises.

Nous avons notamment :

- Le serveur d'application hébergé sur la plateforme **Firebase**, sur lequel est déployé notre application web ;
- Le serveur hébergé sur **Okteto**, où est déployée l'API faisant office de pont entre les serveurs de base de données et d'applications ;
- Le serveur de base de données PostgreSQL hébergé sur **Heroku** ;
- Le poste client permettant l'affichage des interfaces et les interactions des utilisateurs avec l'application web

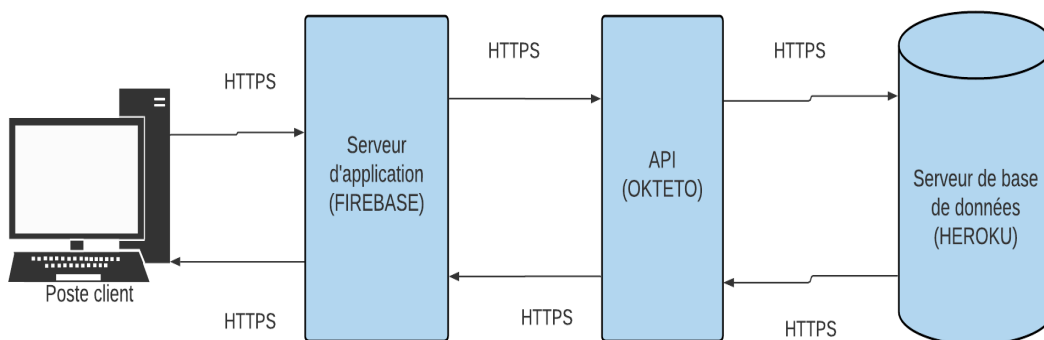


FIGURE 18: ARCHITECTURE MATERIELLE DE L'APPLICATION

3.2.2 Architecture logicielle

3.2.2.1 API

Notre API suit une architecture logicielle **REST (Representational State Transfer)** ou **RESTFul**.

REST est un style d'architecture permettant de construire des applications (Web, Intranet, Web Service).

Il impose un certain nombre de contraintes de conception :

- **Client-serveur** : le client et le serveur doivent être séparés l'un de l'autre. Le serveur dispose d'un ensemble d'informations ;
- **Stateless (Sans état)** : les informations du client ne sont jamais stockées entre les différentes requêtes ;
- **Une interface uniforme** : entre les composants qui permet la communication et le transfert des informations entre le serveur et le client dans un seul langage ;
- **Mise en cache des données** : le serveur peut mettre en cache les réponses pour améliorer la performance du système ;
- **Code à la demande** : offre la possibilité d'envoyer du code exécutable depuis le serveur vers le client ;
- **Un système à couche** : consiste à rendre invisible au client l'architecture matérielle du service auquel il a accès/

Comment implémentions-nous les contraintes de REST ?

❖ **Contrainte client-serveur**

Pour notre application, le serveur représenté par notre API hébergé sur Okteto est séparé du client qui est hébergé sur Firebase. Le client interagit avec le serveur grâce à des appels aux points d'entrées fournis par ce dernier.

❖ **Sans état**

Les appels effectués entre le client et le serveur de notre application, ne sont pas sauvegardées. A chaque appel du client, le serveur lui fournit les ressources nécessaires.

❖ Interface uniforme

Pour la communication entre le client et le serveur, nous avons choisi comme interface le langage JSON. Notre choix s'est porté sur ce langage tout d'abord parce qu'il est supporté par les langages de programmation Python et TypeScript que nous avons utilisés durant le développement respectif de l'API et de l'application web.

3.2.2.2 Application web

Notre application web étant développée avec la technologie Angular elle suit l'architecture logicielle **Model-View-ViewModel**.

Qu'est-ce que le MVVM ?

Le Model-View-ViewModel est un design pattern dérivé du MVC, qui présente quelques différences dont notamment la séparation de l'interface utilisateur et de la logique métier.

Ce pattern identifie trois objets distincts :

- **Le modèle :** Il constitue la couche des données métier et n'est lié à aucune représentation graphique
- **La vue :** Elle ne contient aucune logique applicative. C'est l'interface que les utilisateurs auront à l'écran.
- **La Vue-Modèle :** ce composant fait le lien entre le modèle et la vue. Il s'occupe de gérer les liaisons de données et les éventuelles conversions. C'est ici qu'intervient le **binding**.

Qu'est-ce que le binding ?

Le binding ou data binding est un mécanisme qui permet la synchronisation instantanée entre la vue (HTML) et les données de l'application (généralement le modèle).

3.3 **Sécurité de l'application**

Notre application manipulant un grand nombre d'informations confidentielles, il constitue un service critique dont la protection doit être une priorité. Pour se faire nous avons eu à utiliser un certain nombre de règles et technologies afin d'assurer la sécurité de notre application.

3.3.1 **Authentification et autorisation**

Pour accéder aux diverses ressources de notre application nous avons défini des règles que chaque utilisateur doit respecter. Pour se faire nous avons eu à utiliser la technologie JWT.

Qu'est-ce que JWT ?

JWT (JSON Web Token) est un standard ouvert qui permet l'échange sécurisée de jetons (tokens) entre plusieurs parties. Cette sécurité de l'échange se traduit par la vérification de l'intégrité et de l'authenticité des données. Elle s'effectue à l'aide de l'algorithme HMAC ou RSA.⁵¹

Dans le cas de notre application les jetons générées par l'API que nous avons eu à développer, sont générés dès qu'un utilisateur se connecte. Ces jetons sont alors envoyés aux navigateurs de l'utilisateur qui les emploie pour accéder aux ressources protégées de notre API.

Ces jetons contiennent des informations qui permettent à l'API d'identifier l'utilisateur et de connaître ses rôles et les ressources auxquelles ce dernier a le droit d'accéder, et qui définissent leur durée de validité.

⁵¹ Source : [JSON Web Token — Wikipédia \(wikipedia.org\)](https://fr.wikipedia.org/wiki/JSON_Web_Token)

3.3.2 Cross-Origin Resource Sharing (CORS)

Le **CORS** est un mécanisme qui consiste à ajouter des en-têtes http afin de permettre à un agent utilisateur d'accéder à des ressources d'un serveur situé sur une autre origine que le site courant.⁵²

Le backend de notre projet étant une API avec laquelle doit communiquer uniquement l'application web, nous avons dû mettre en place ce mécanisme pour le protéger. En effet, un utilisateur malveillant ayant connaissance des points d'entrée de l'API pourrait facilement interagir avec elle en utilisant certains outils.

A cet effet, pour éviter que cela ne se produise nous avons mis en place une liste contenant les services clients dont notre application web, autorisée à interagir avec notre API.

3.3.3 Hypertext Transfer Protocol Secure (HTTPS)

Notre service étant constitué d'une API, les appels qu'elle reçoit de l'application web doivent être vérifiés et les réponses qu'elle lui renvoie doivent être chiffrées. C'est dans cette optique que nous avons mis en place le protocole HTTPS.

Qu'est-ce que le HTTPS ?

HTTPS est une version chiffrée du protocole HTTP. Il permet entre autres de chiffrer l'intégralité des communications entre un client et un serveur. La connexion sécurisée permet aux clients d'échanger de manière sûre des données sensibles avec un serveur.⁵³

⁵² Source: [Cross-origin resource sharing \(CORS\) - HTTP | MDN \(mozilla.org\)](#)

⁵³ Source : [HTTPS - Glossaire | MDN \(mozilla.org\)](#)

3.4 **Mise en place de la base de données**

Pour la mise en place de notre base de données nous avons utilisé l'ORM SQLAlchemy précédemment présenté qui nous permet de créer des tables à partir de classes écrites en python, et de faciliter la connexion à la base de données ainsi que les migrations des clés entre les tables.

Dans cette partie nous ferons la présentation de quelques classes et fonctions qui nous ont permis de mettre en place la base de données obtenue à partir de notre analyse.

3.4.1 **Le fichier session.py**

Ce fichier nous permet entre autres de créer une connexion à notre base de données avec du code python et des identifiants de notre base de données hébergée sur la plateforme Heroku, et de définir une fonction qui sont utilisés lors de la création des modèles SQLAlchemy.

3.4.2 **Les modèles SQLAlchemy**

Toutes les tables de notre base de données sont définies à partir des modèles de bases de données de SQLAlchemy. Ces modèles héritent de la fonction de connexion établie dans le fichier session.py.

3.4.3 **Les opérations de CRUD et les requêtes**

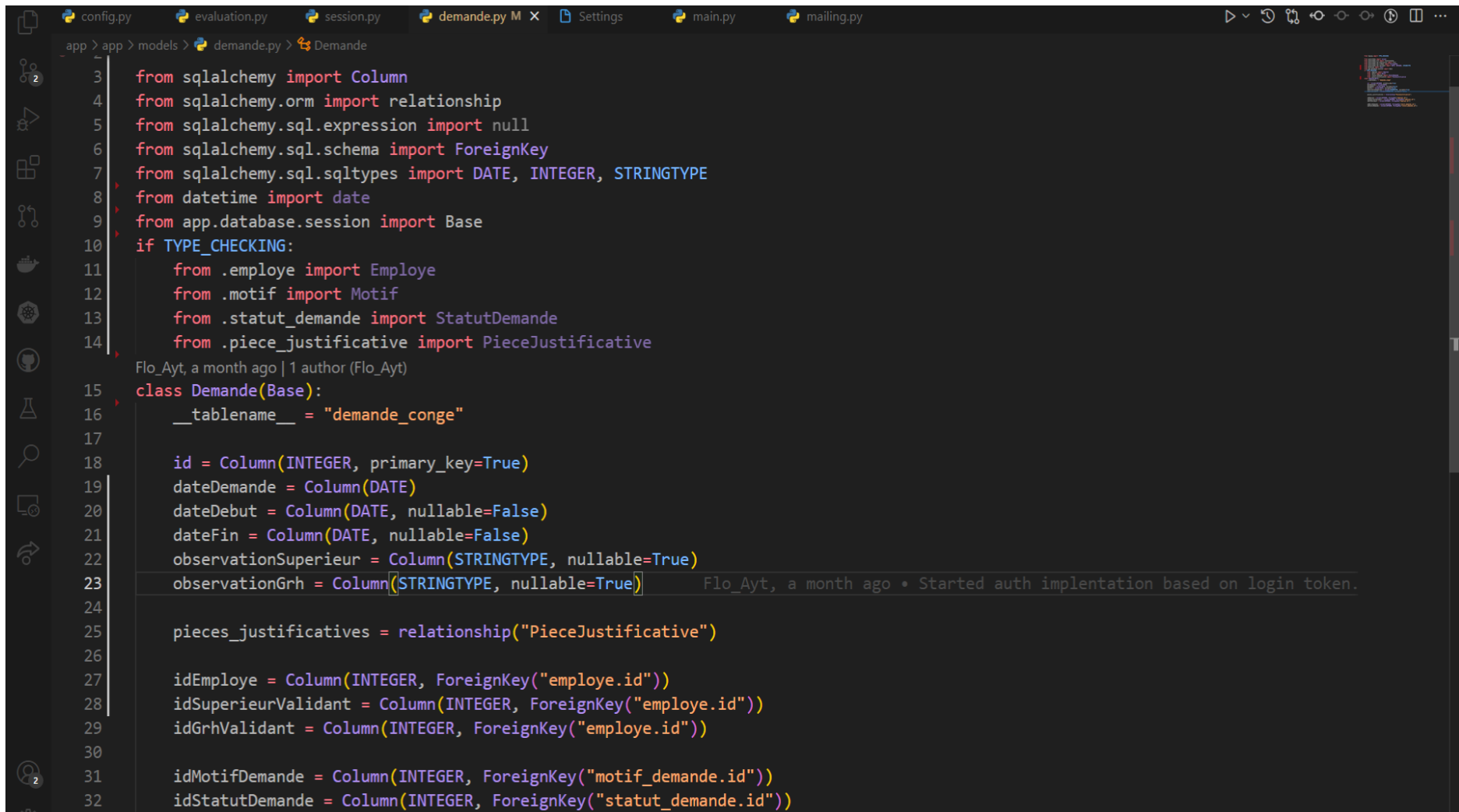
Les opérations de CRUD (Create Read Update Delete), ainsi que toutes les requêtes de bases de données sont écrites dans des classes respectivement pour chaque modèles SQLAlchemy de notre projet. Ces classes encapsulent les méthodes que nous aurons à utiliser pour interroger notre base de données au travers de ses endpoints. Elles héritent elles même de la classe CRUDBase écrite par le créateur de FastAPI qui regroupe les méthodes les plus souvent utilisées.

3.4.4 Quelques exemples montrant la mise en place de notre base de données

```
app > app > database > session.py > ...
Flo_Ayt, a month ago | 1 author (Flo_Ayt)
1  from sqlalchemy import create_engine
2  from sqlalchemy.ext.declarative import declarative_base
3  from sqlalchemy.orm import sessionmaker
4
5
6  SQLALCHEMY_DATABASE_URL = "postgresql://bpmhhdzblvnqyb:4360cd11e5a317df67084593f0b61f4a3b99ad7e460fff52c7862dd0ebcd42c4@ec2-54-211
7  engine = create_engine(SQLALCHEMY_DATABASE_URL)
8  SessionLocal = sessionmaker(autocommit=False, autoflush=False, bind=engine)
9
10 Base = declarative_base()
11
```

FIGURE 19: FICHIER SESSION.PY DE CONNEXION A LA BASE DE DONNEES

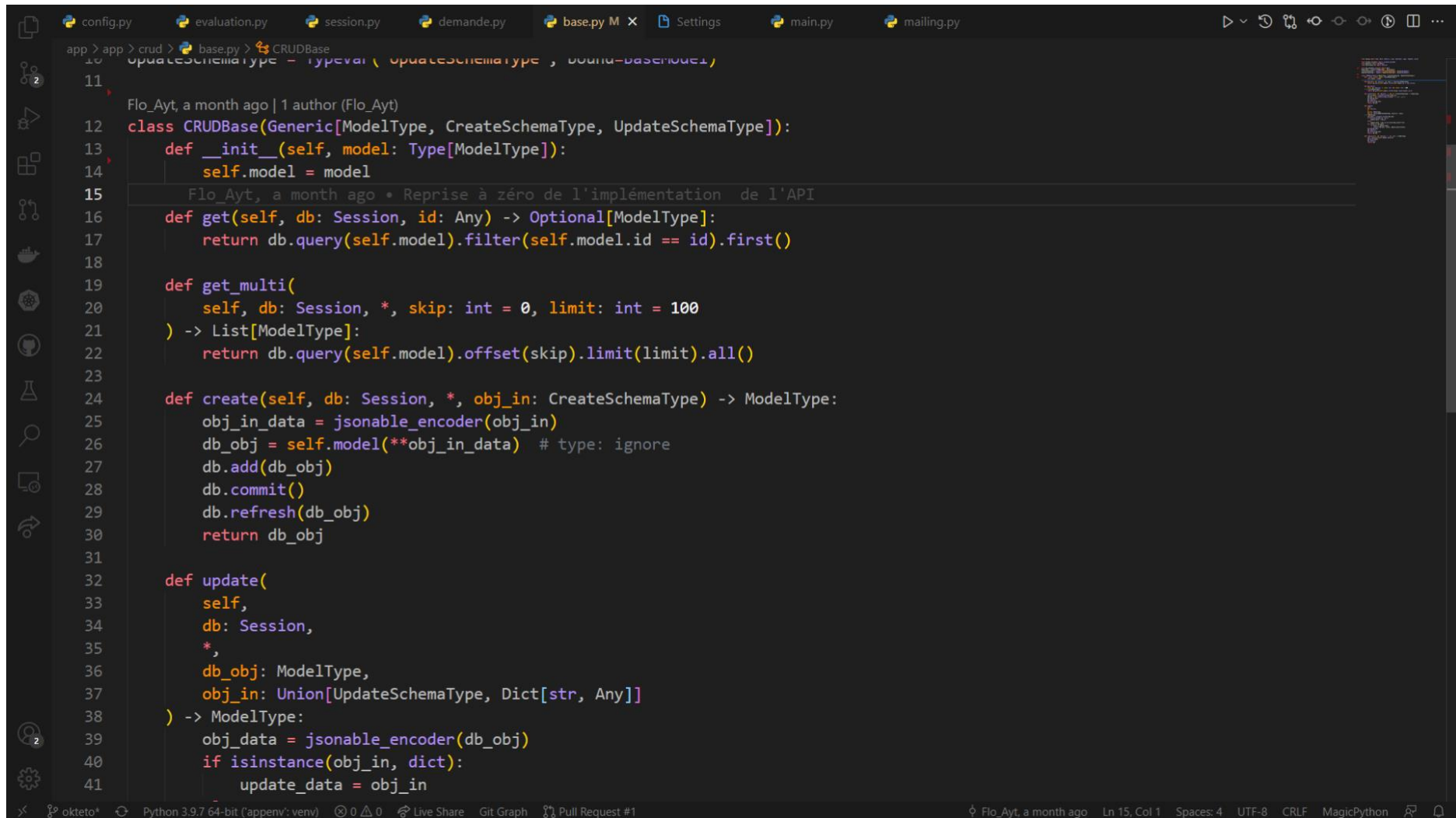
PARTIE 3 : REALISATION ET MISE EN OEUVRE



```
config.py  evaluation.py  session.py  demande.py M  Settings  main.py  mailing.py
app > app > models > demande.py > Demande
3  from sqlalchemy import Column
4  from sqlalchemy.orm import relationship
5  from sqlalchemy.sql.expression import null
6  from sqlalchemy.sql.schema import ForeignKey
7  from sqlalchemy.sql.sqltypes import DATE, INTEGER, STRINGTYPE
8  from datetime import date
9  from app.database.session import Base
10 if TYPE_CHECKING:
11     from .employe import Employe
12     from .motif import Motif
13     from .statut_demande import StatutDemande
14     from .piece_justificative import PieceJustificative
15 Flo_Ayt, a month ago | 1 author (Flo_Ayt)
16 class Demande(Base):
17     __tablename__ = "demande_conge"
18     id = Column(INTEGER, primary_key=True)
19     dateDemande = Column(DATE)
20     dateDebut = Column(DATE, nullable=False)
21     dateFin = Column(DATE, nullable=False)
22     observationSuperieur = Column(STRINGTYPE, nullable=True)
23     observationGrh = Column(STRINGTYPE, nullable=True)
24     pieces_justificatives = relationship("PieceJustificative")
25     idEmploye = Column(INTEGER, ForeignKey("employe.id"))
26     idSuperieurValidant = Column(INTEGER, ForeignKey("employe.id"))
27     idGrhValidant = Column(INTEGER, ForeignKey("employe.id"))
28     idMotifDemande = Column(INTEGER, ForeignKey("motif_demande.id"))
29     idStatutDemande = Column(INTEGER, ForeignKey("statut_demande.id"))
```

FIGURE 20: EXEMPLE DE MODELE SQLALCHEMY

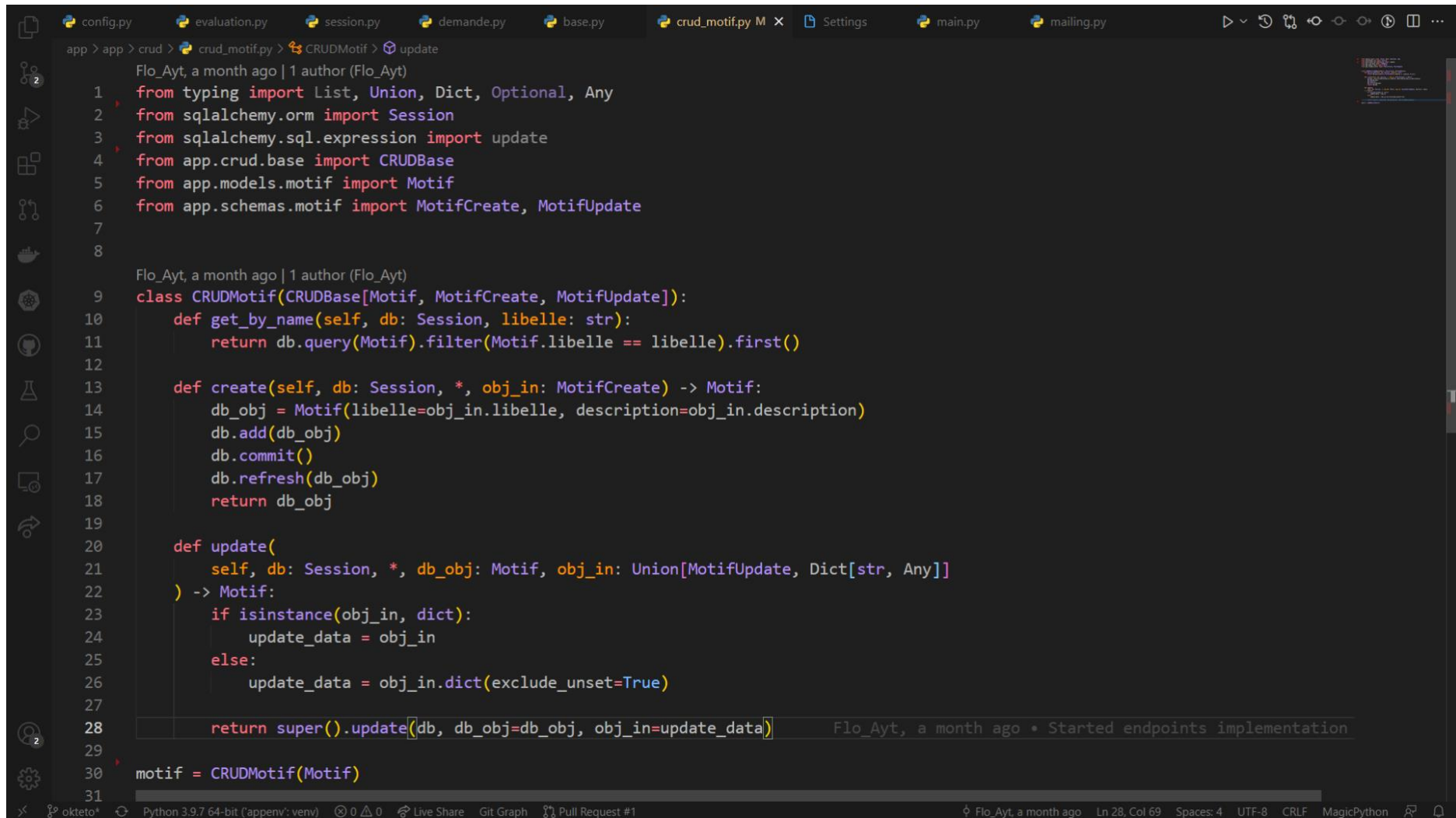
PARTIE 3 : REALISATION ET MISE EN OEUVRE



```
config.py evaluation.py session.py demande.py base.py M X Settings main.py mailing.py
app > app > crud > base.py > CRUDBase
updateSchemaType = typeval(UpdateSchemaType, bound=BaseModel)
11
Flo_Ayt, a month ago | 1 author (Flo_Ayt)
12 class CRUDBase(Generic[ModelType, CreateSchemaType, UpdateSchemaType]):
13     def __init__(self, model: Type[ModelType]):
14         self.model = model
15     Flo_Ayt, a month ago • Reprise à zéro de l'implémentation de l'API
16     def get(self, db: Session, id: Any) -> Optional[ModelType]:
17         return db.query(self.model).filter(self.model.id == id).first()
18
19     def get_multi(
20         self, db: Session, *, skip: int = 0, limit: int = 100
21     ) -> List[ModelType]:
22         return db.query(self.model).offset(skip).limit(limit).all()
23
24     def create(self, db: Session, *, obj_in: CreateSchemaType) -> ModelType:
25         obj_in_data = jsonable_encoder(obj_in)
26         db_obj = self.model(**obj_in_data) # type: ignore
27         db.add(db_obj)
28         db.commit()
29         db.refresh(db_obj)
30         return db_obj
31
32     def update(
33         self,
34         db: Session,
35         *,
36         db_obj: ModelType,
37         obj_in: Union[UpdateSchemaType, Dict[str, Any]]
38     ) -> ModelType:
39         obj_data = jsonable_encoder(db_obj)
40         if isinstance(obj_in, dict):
41             update_data = obj_in
```

FIGURE 21: CLASSE DE CRUD DE BASE

PARTIE 3 : REALISATION ET MISE EN OEUVRE



```
config.py  evaluation.py  session.py  demande.py  base.py  crud_motif.py M  Settings  main.py  mailing.py
app > app > crud > crud_motif.py > CRUDMotif > update
Flo_Ayt, a month ago | 1 author (Flo_Ayt)
1  from typing import List, Union, Dict, Optional, Any
2  from sqlalchemy.orm import Session
3  from sqlalchemy.sql.expression import update
4  from app.crud.base import CRUDBase
5  from app.models.motif import Motif
6  from app.schemas.motif import MotifCreate, MotifUpdate
7
8
9  Flo_Ayt, a month ago | 1 author (Flo_Ayt)
10 class CRUDMotif(CRUDBase[Motif, MotifCreate, MotifUpdate]):
11     def get_by_name(self, db: Session, libelle: str):
12         return db.query(Motif).filter(Motif.libelle == libelle).first()
13
14     def create(self, db: Session, *, obj_in: MotifCreate) -> Motif:
15         db_obj = Motif(libelle=obj_in.libelle, description=obj_in.description)
16         db.add(db_obj)
17         db.commit()
18         db.refresh(db_obj)
19         return db_obj
20
21     def update(
22         self, db: Session, *, db_obj: Motif, obj_in: Union[MotifUpdate, Dict[str, Any]]
23     ) -> Motif:
24         if isinstance(obj_in, dict):
25             update_data = obj_in
26         else:
27             update_data = obj_in.dict(exclude_unset=True)
28
29         return super().update(db, db_obj=db_obj, obj_in=update_data)
30
31 motif = CRUDMotif(Motif)
```

Flo_Ayt, a month ago • Started endpoints implementation

FIGURE 22: EXEMPLE DE CLASSE ENCAPSULANT LES OPERATIONS CRUD D'UN MODELE SQLALCHEMY

PARTIE 4 : GUIDE D'EXPLOITATION

4.1 Configuration logicielle et matérielle

4.1.1 Configuration logicielle

PARTIE 4 : GUIDE D'EXPLOITATION

TABLEAU 9 CONFIGURATION LOGICIELLE

Appareil	Configuration minimale requise
Serveur hébergeant l'API	<u>Type d'hébergement</u> : Cloud (Okteto) <u>Support logiciel</u> : Kubernetes, Docker
Serveur hébergeant la base de données PostgreSQL	<u>Type d'hébergement</u> : Cloud (Heroku)
Serveur hébergeant PgAdmin	<u>Type d'hébergement</u> : Cloud (Okteto) <u>Support logiciel</u> : Kubernetes
Serveur hébergeant l'application web	<u>Type d'hébergement</u> : Cloud (Firebase)
Poste de l'utilisateur	<u>Système d'exploitation</u> : Linux, Windows ou MacOS <u>Support Logiciel</u> : Navigateur avec Javascript activé

4.1.2 Configuration matérielle

PARTIE 4 : GUIDE D'EXPLOITATION

TABLEAU 10 CONFIGURATION MATERIELLE

	Configuration minimale	Configuration recommandée
Serveur	Processeur : 1.7 GHz Nombre de cœur : 2 Disque dur : 500 Go SSD Mémoire RAM : 4 Go DDR5 Système d'exploitation : Ubuntu Server 20.04.2 LTS Bande passante : 1 Gbit/s	Processeur : 3.1 GHz Nombre de cœur : 16 Disque dur : 2 To SSD Mémoire RAM : 64 Go DDR5 Système d'exploitation : Ubuntu Server 20.04.2 LTS Bande passante : 10 Gbit/s
Clients	Mémoire RAM : 2 Go Bande passante : 10 Mbit/s	Mémoire RAM : 2 Go Bande passante : 10 Mbit/s

4.2 Déploiement et suivi

4.2.1 Déploiement de l'API

Pour le déploiement de l'API, nous utilisons Okteto présenté précédemment dans la partie **Mise en œuvre**. Nous mettrons en place un fichier de configuration nommé `okteto-stack.yaml` et un autre fichier de configuration Docker tous les deux situés à la racine de notre projet.

Les fichiers sus-cités contiennent des tâches à exécuter durant le déploiement. Parmi ces travaux on a notamment :

- L'installation de l'image de base (**tiangolo/uvicorn-gunicorn-fastapi**⁵⁴) ;
- L'installation de python, et des dépendances nécessaires pour l'exécution du projet ;
- La construction du projet du projet docker ;
- Le déploiement de l'API sur la plateforme Okteto.

✓ Le fichier docker

```

Dockerfile > ...
You, 3 weeks ago | 1 author (You)
1 FROM tiangolo/uvicorn-gunicorn-fastapi:python3.7
2
3 WORKDIR /app
4 #Installing dependencies with pipenv
5 RUN pip install pipenv
6 COPY Pipfile Pipfile.lock /app/
7 RUN pipenv install --system --dev
8
9 EXPOSE 80
10
11 COPY ./app /app
12 ENV PYTHONPATH=/app
13 CMD ["uvicorn", "app.main:app", "--host", "0.0.0.0", "--port", "80"]
14

```

FIGURE 23 : FICHIER DOCKER

⁵⁴ <https://hub.docker.com/r/tiangolo/uvicorn-gunicorn-fastapi>

✓ Le fichier okteto-stack.yaml

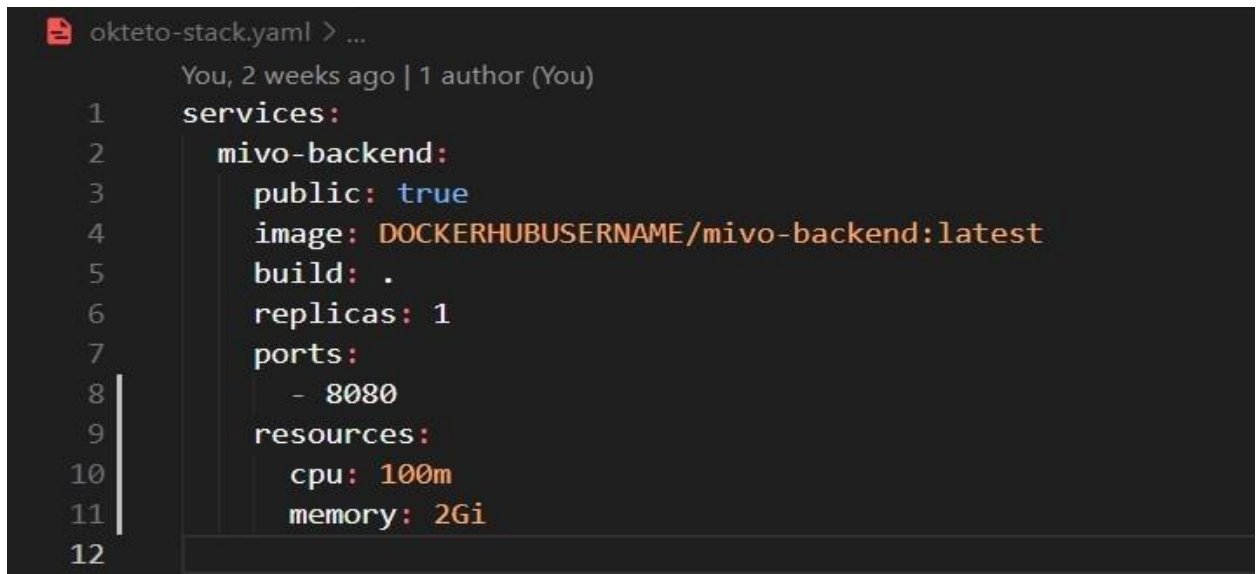


FIGURE 24 : FICHIER DE CONFIGURATION OKTETO-STACK.YAML

Après cette étape il faudra se connecter à la plateforme Okteto et à DockerHub pour pouvoir finaliser le déploiement.

Nous aurons à utiliser les commandes suivantes :

- `scoop install okteto` (pour installer le cli d'Okteto) ;
- `okteto login` (pour la connexion à la plateforme okteto) ;
- `docker login` (pour la connexion à DockerHub) ;
- `okteto namespace` espace de nom sous lequel nous allons déployer notre api ;
- `okteto stack deploy --build`, qui permet de déployer notre application à partir d'un container

Après le déploiement il nous faudra nous connecter à la plateforme Okteto pour obtenir le lien sécurisé nous permettant d'accéder notre API.

PARTIE 4 : GUIDE D'EXPLOITATION

```
Windows PowerShell
PS C:\Users\devar\Desktop\mivo-back-end> okteto login
What is the URL of your Okteto instance? [https://cloud.okteto.com]:
Authentication will continue in your default browser
You can also open a browser and navigate to the following address:
https://cloud.okteto.com/auth/authorization-code?redirect=http%3A%2F%2F127.0.0.1%3A53455%2Fauthorization-code%2Fcallback%3Fstate%3Dc7g0%2BsZwSPYcQZSo%2BLinTgQiiHqhtbSDPprsGdvrfjA%3D
gQiiHqhtbSDPprsGdvrfjA%3D&state=c7g0%2BsZwSPYcQZSo%2BLinTgQiiHqhtbSDPprsGdvrfjA%3D
+ Logged in as FlorianAyt
  Run 'okteto namespace' to switch your context and download your Kubernetes credentials.
PS C:\Users\devar\Desktop\mivo-back-end> docker login
Authenticating with existing credentials...
Login Succeeded
PS C:\Users\devar\Desktop\mivo-back-end> okteto namespace mivo-florianayt
+ Updated context 'cloud_okteto_com' in 'C:\Users\devar\.kube\config'
PS C:\Users\devar\Desktop\mivo-back-end> okteto stack deploy --build
i Running your build in tcp://buildkit.cloud.okteto.net:1234...
i Building image for service 'mivo-backend'...
[+] Building 20.9s (11/11) FINISHED
=> [internal] load build definition from buildkit-1242538426 1.2s
=> => transferring dockerfile: 355B 0.9s
=> [internal] load .dockerignore 1.1s
=> => transferring context: 46B 0.9s
=> [internal] load metadata for docker.io/tiangolo/uvicorn-gunicorn-fastapi:python3.7 1.2s
=> [1/6] FROM docker.io/tiangolo/uvicorn-gunicorn-fastapi:python3.7@sha256:8efb7b7c2419090159955d01c9978b2d59ef0 0.1s
=> => resolve docker.io/tiangolo/uvicorn-gunicorn-fastapi:python3.7@sha256:8efb7b7c2419090159955d01c9978b2d59ef0 0.1s
=> [internal] load build context 3.7s
=> => transferring context: 503.62kB 3.6s
=> CACHED [2/6] WORKDIR /app 0.0s
=> CACHED [3/6] RUN pip install pipenv 0.0s
=> CACHED [4/6] COPY Pipfile Pipfile.lock /app/ 0.0s
=> CACHED [5/6] RUN pipenv install --system --dev 0.0s
=> [6/6] COPY ./app /app 0.2s
=> exporting to image 14.3s
=> => exporting layers 11.4s
=> => exporting manifest sha256:51b2ac12c65358f9283009480dcc12c3d29c85e1653bc50db865e2d9c4bfcc66 0.1s
=> => exporting config sha256:7e1a27dbbe88d1ffb6729c5a85ba2648645c976f12be7b696369cfa75bd246ee 0.1s
=> => pushing layers 1.8s
=> => pushing manifest for registry.cloud.okteto.net/mivo-florianayt/mivo-back-end-mivo-backend:okteto 0.5s
+ Image for service 'mivo-backend' successfully pushed
+ Deployed service 'mivo-backend'
+ Stack 'mivo-back-end' successfully deployed
```

FIGURE 25 : DEPLOIEMENT DE L'API AVEC LE CLI D'OKTETO

PARTIE 4 : GUIDE D'EXPLOITATION

Namespace: mivo-florianayt

Deploy | Share | Delete

Pods: 1 / 10 | Storage: 0B / 5GB

mivo-back-end Stack | 1 Deployed | **mivo-backend** Deployment | Running

mivo-backend Deployment | Start Development | Restart | Destroy

Endpoints: <https://mivo-backend-mivo-florianayt.cloud.okteto.net> | Replicas: 1 | Created: 8 days ago | Last updated: 11 hrs ago | Pods: 1 | Memory: 75.6MB / 2GB | CPU: 0.0 / 0.1

Logs | YAML | Timestamps | Pod: All pods | Container: All containers

```
2021-09-11 22:50:05.04 UTC mivo-backend-84dfc76b5f-mr922 mivo-backend INFO: Started server process [1]
2021-09-11 22:50:05.04 UTC mivo-backend-84dfc76b5f-mr922 mivo-backend INFO: Waiting for application startup.
2021-09-11 22:50:08.05 UTC mivo-backend-84dfc76b5f-mr922 mivo-backend INFO: Application startup complete.
2021-09-11 22:50:08.05 UTC mivo-backend-84dfc76b5f-mr922 mivo-backend INFO: Uvicorn running on http://0.0.0.0:80 (Press CTRL+C to quit)
```

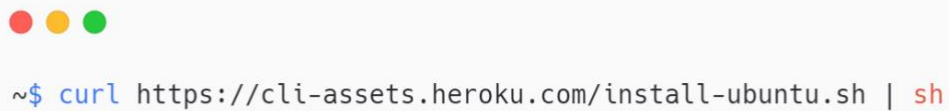
FIGURE : 26 DEPLOIEMENT DE L'API (ENDPOINTS)

4.2.2 Déploiement de la base de données

Notre base de données est hébergée sur Heroku. Nous allons présenter toutes les étapes de son déploiement.

Il faudra d'abord disposer d'un compte (<https://signup.heroku.com/login>) et ensuite installer le cli d'Heroku.

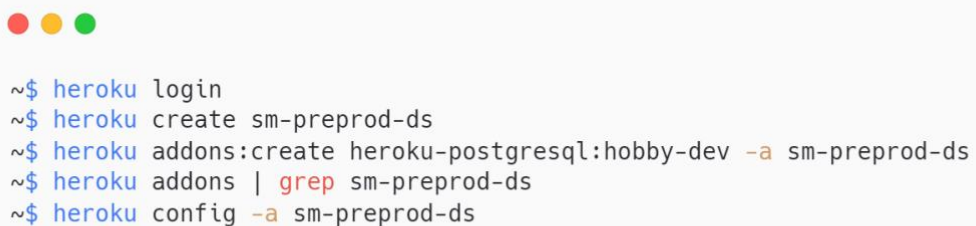
❖ Installation Heroku CLI



```
~$ curl https://cli-assets.heroku.com/install-ubuntu.sh | sh
```

FIGURE 27: INSTALLATION DE HEROKU CLI SOUS UBUNTU 20.4

❖ Authentification et création d'une nouvelle base de données



```
~$ heroku login
~$ heroku create sm-preprod-ds
~$ heroku addons:create heroku-postgresql:hobby-dev -a sm-preprod-ds
~$ heroku addons | grep sm-preprod-ds
~$ heroku config -a sm-preprod-ds
```


FIGURE 28: CREATION D'UNE BASE DE DONNEES AVEC HEROKU CLI

4.2.3 Déploiement de l'application web

PARTIE 4 : GUIDE D'EXPLOITATION

Pour déployer l'application web nous créons un projet en utilisant la plateforme de Google : **Firebase** avec l'url <https://console.firebase.google.com/>.

Après la création le projet, il faudra installer les outils de ligne de commande de firebase et ensuite procéder au déploiement :

A terminal window with a white background and three colored window control buttons (red, yellow, green) in the top-left corner. It displays five lines of terminal commands in a monospaced font, each preceded by a tilde and a dollar sign (~\$).

```
~$ npm run build
~$ npm install -g firebase-tools
~$ firebase login
~$ firebase init
~$ firebase deploy
```

FIGURE 29: DEPLOIEMENT DE L'APPLICATION WEB AVEC FIREBASE

4.3 **Maintenance**

La maintenance consiste à documenter les erreurs pouvant survenir lors de l'utilisation de l'application et les différentes actions à mener lorsque celles-ci se produisent.

Nous avons pris le soin d'afficher dans notre application des détails pour la plupart des erreurs rencontrés à l'aide des notifications Toastr pour permettre aux utilisateurs de savoir précisément ce pourquoi ils ont ces messages d'erreurs.

On distingue notamment quelques erreurs récurrentes :

- **400** : La syntaxe de la requête utilisée pour accéder à la ressource est invalide. La requête doit être corrigée ;
- **401** : L'utilisateur doit s'authentifier pour accéder aux ressources ;
- **403** : L'utilisateur ne dispose des droits d'accès aux ressources. Il peut contacter l'administrateur pour plus d'explications ;
- **404** : La ressource voulue n'est pas trouvée ;
- **422** : Les données dans la requête envoyée au serveur sont incomplètes ou incompréhensible. L'utilisateur doit corriger la requête avant de la renvoyer ;
- **500** : Erreur interne du serveur. Pour ce cas d'erreur, l'utilisateur doit rafraîchir la page. Si l'erreur persiste il devra contacter un administrateur.

PARTIE 5 : GUIDE D'UTILISATION

5.1 **Description textuelle du logiciel**

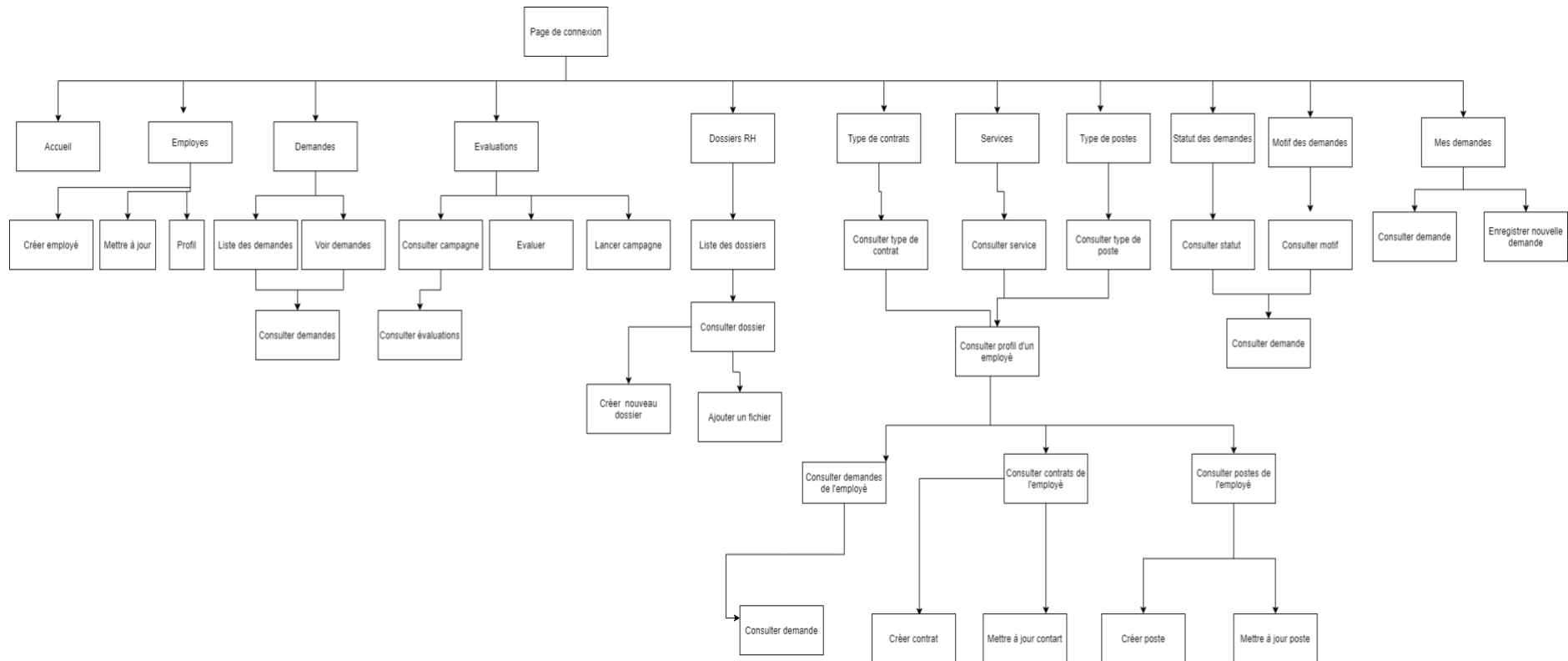
Notre application a pour but de permettre au personnel de l'ONG Mivo Energie de pouvoir procéder à des demandes de congés, aux supérieurs hiérarchiques d'évaluer ses subordonnées et aux chargés des ressources humaines d'avoir et de gérer plus facilement les dossiers RH du personnel.

Ses objectifs sont notamment :

- de permettre d'évaluer les employés sur une période donnée ;
- de faciliter la dématérialisation des fichiers et des documents et de les sauvegarder ;
- de permettre à l'ONG de disposer des informations en temps réel sur l'ensemble du personnel ;
- de faciliter le suivi des demandes avec l'envoi de mails notifiant l'état des demandes.

5.2 Plan de navigation

Le plan de navigation permet entre autres aux utilisateurs de savoir comment il faudra naviguer à travers les différentes pages de notre application.



PARTIE 5 : GUIDE D'UTILISATION

FIGURE 30:PLAN DE NAVIGATION DE L'APPLICATION

5.3 **Présentation des différentes interfaces**

Dans cette partie, nous faisons la présente de quelques pages de notre application, à savoir :

❖ **La page de connexion**

❖ **La page de consultation de la liste des employés**

Cette page est visible uniquement pour le chargé des ressources humaines, elle lui permet de consulter, modifier ou désactivé le profil d'un employé.

❖ **La page de validation d'une demande de congés**

Cette comporte les informations sur la demande de congés et le demandeur et offre la possibilité au supérieur hiérarchique ou au chargé des RH de refuser ou accepter une demande de congés.

❖ **La page d'enregistrement d'une demande de congés**

Elle comporte un formulaire qui permet d'enregistrer une demande de congés et d'ajouter des pièces justificatives.

❖ **La page d'évaluations**

Cette page permet d'enregistrer les notes, observations de l'évaluation d'un employé. Elle comporte un formulaire affichant les objectifs du poste actuel de l'employé.

❖ **La page de lancement des campagnes d'évaluations**

Elle offre la possibilité à un chargé des RH de lancer une campagne d'évaluations.

❖ **La page d'attribution des objectifs**

Cette page contient un formulaire qui donne la possibilité d'ajouter des objectifs à un type de poste.

5.3.1 Images de quelques pages de notre application

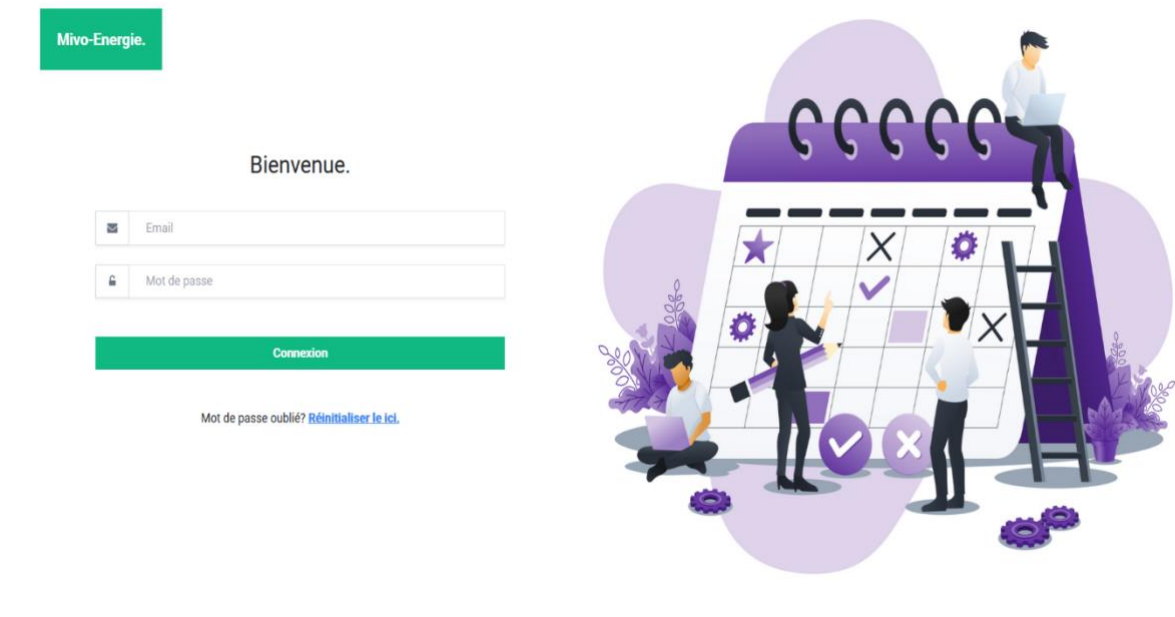


FIGURE 31:PAGE DE CONNEXION

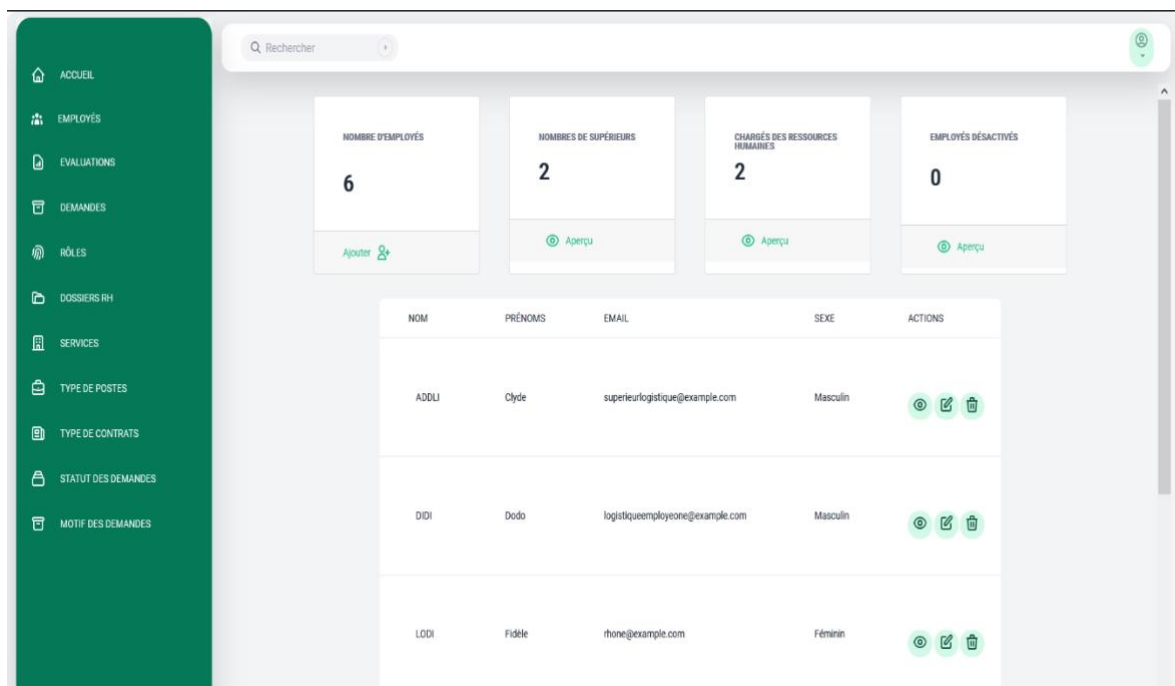


FIGURE 32: PAGE MONTRANT LA LISTE DES EMPLOYES

PARTIE 5 : GUIDE D'UTILISATION

The screenshot shows a web application interface for a CRH user. On the left is a green sidebar with navigation links: ACCUEIL, SUBORDONNÉS, EVALUATIONS, and DEMANDES. The main content area has a search bar at the top. Below it is a form titled 'Demande de congés' with the following fields: Nom (DOSI), Prénoms (Arthur), Date de la demande (19 / 09 / 2021), Motif de la demande (Congé maladie), Date de départ en congés (24 / 09 / 2021), Date de retour (01 / 10 / 2021), Ancienneté (Mois) (93), and Jours de congés restant (Année en cours) (14). There are also fields for 'Décision' and 'Observation'. At the bottom right of the form are buttons for 'Enregistrer' and 'Annuler'. Below the form is a section titled 'Pièces justificatives' containing a table with one row: 'PROCESSUS DE DÉVELOPPEMENT.pdf' and a 'Voir' button.

FIGURE 33:PAGE DE VALIDATION D'UNE DEMANDE DE CONGES PAR UN CRH

FIGURE 34:PAGE DE VALIDATION D'UNE DEMANDE DE CONGES PAR UN SUPERIEUR HIERARCHIQUE

The screenshot shows the same 'Demande de congés' form, but for a superior hierarchy user. The sidebar now includes additional links: EMPLOYÉS, RÔLES, DOSSIERS RH, SERVICES, TYPE DE POSTES, TYPE DE CONTRATS, STATUT DES DEMANDES, and MOTIF DES DEMANDES. The form fields are identical to Figure 33, but it includes an additional field for 'Nom du supérieur hiérarchique' (KOFI) and 'Prénoms du supérieur hiérarchique' (Adj). The 'Pièces justificatives' section remains the same.

FIGURE 35:PAGE D'EVALUATION

PARTIE 5 : GUIDE D'UTILISATION

The screenshot displays the 'Créer des objectifs' (Create Objectives) page. On the left is a green sidebar with the following menu items: ACCUEIL, EMPLOYÉS, EVALUATIONS, DEMANDES, RÔLES, DOSSIERS RH, SERVICES, TYPE DE POSTES, TYPE DE CONTRATS, STATUT DES DEMANDES, and MOTIF DES DEMANDES. The main content area has a search bar labeled 'Rechercher' and a user profile icon. The central form is titled 'Créer des objectifs' and includes a blue button '+Ajouter nouvel objectif'. Below this, there are four rows of input fields, each with a 'libelle' (label) field and a 'description' field, accompanied by a trash icon for deletion. At the bottom of the form are two buttons: 'Enregistrer' (green) and 'Annuler' (orange).

FIGURE 36: PAGE D'ATTRIBUTION DES OBJECTIFS

PARTIE 5 : GUIDE D'UTILISATION

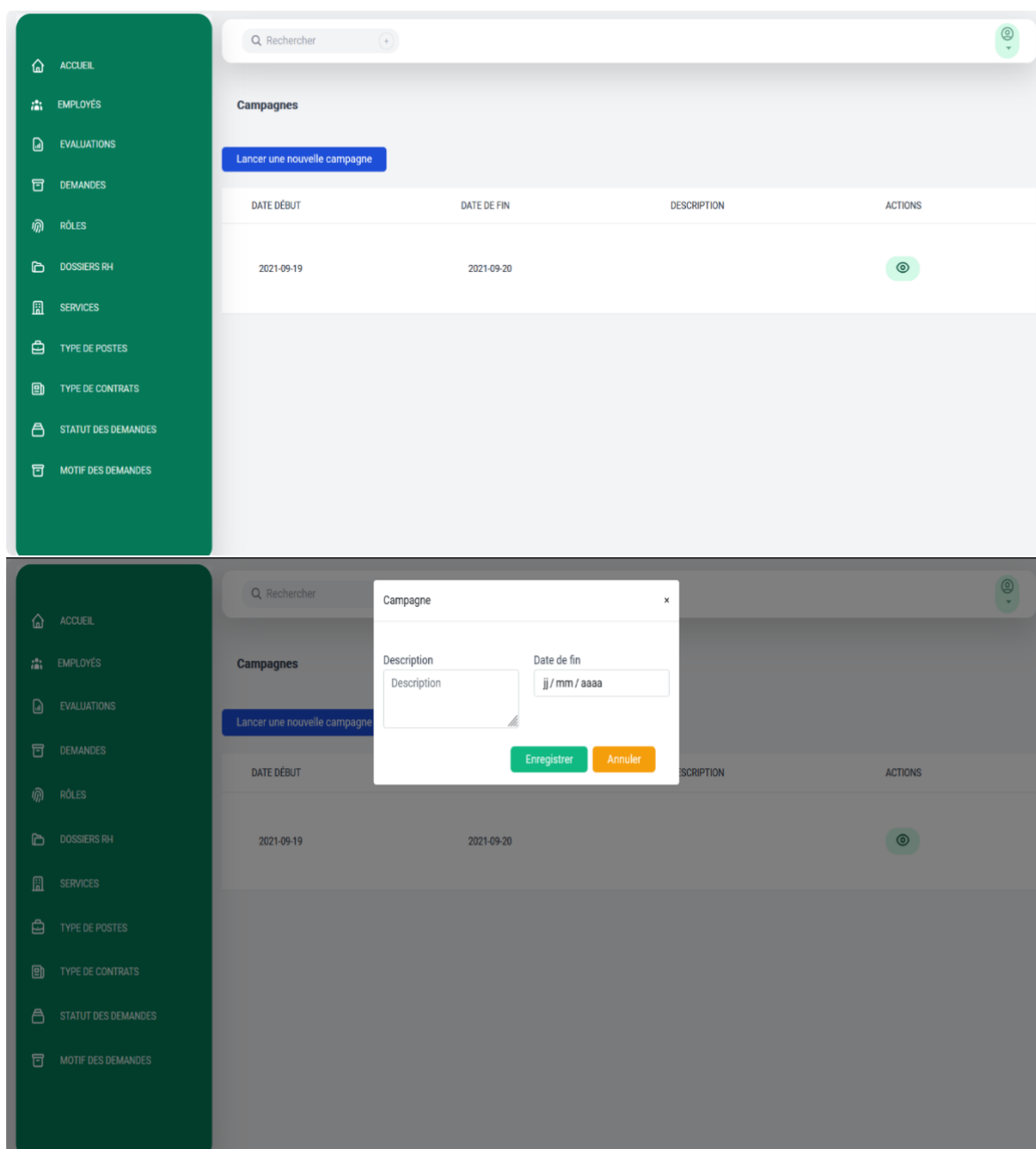


FIGURE 37: PAGE D'AJOUT ET DE CONSULTATION CAMPAGNES D'EVALUATIONS

5.4 Présentation des états

5.4.1 Page des statistiques pour le GRH

Sur la figure ci-dessous nous pouvons apercevoir les graphiques qui permettent au Chargé des ressources humaines de connaître le nombre d'utilisateurs par rôles et par service et le nombre de demandes de congés effectuées par motif de demandes.

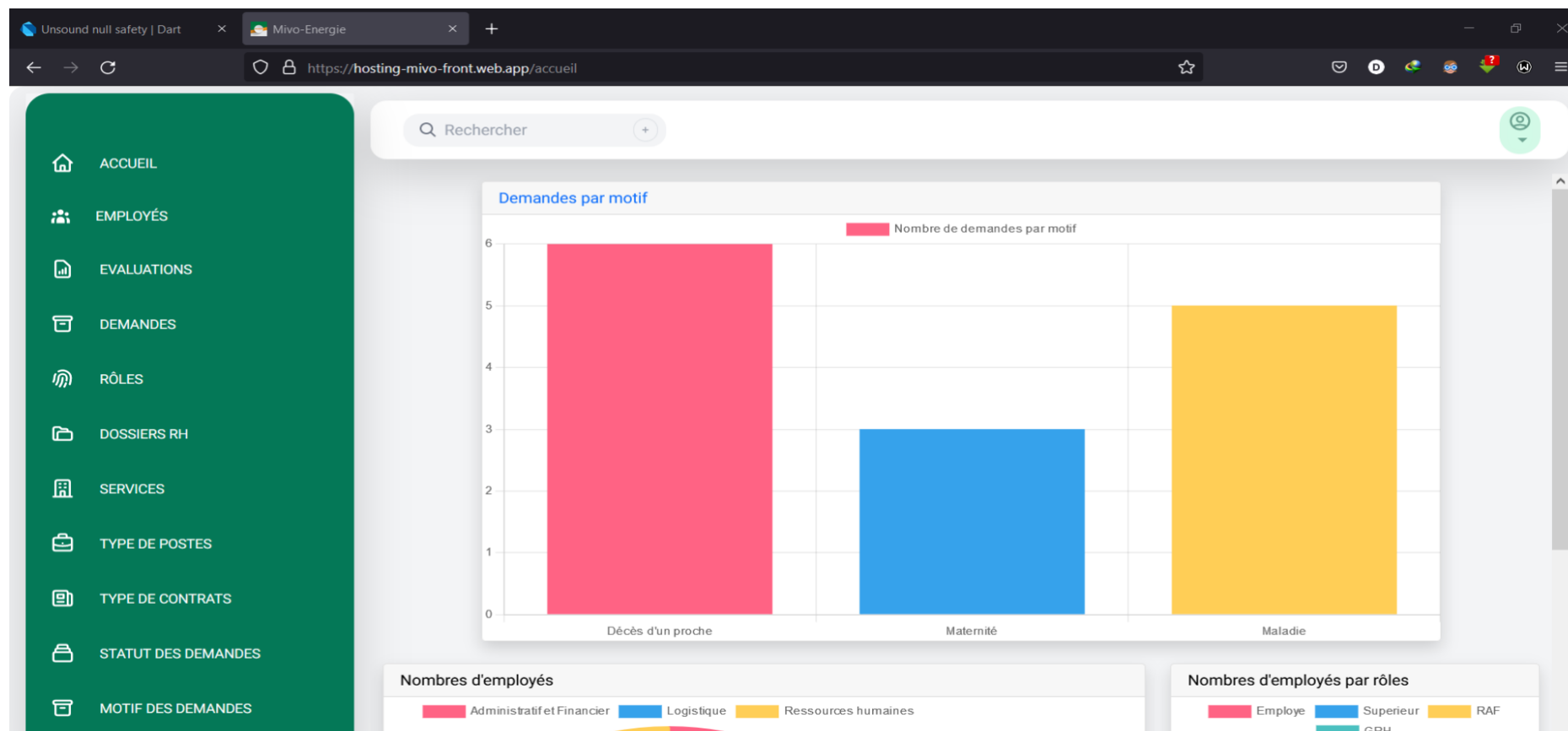


FIGURE 38: PAGE DES STATISTIQUES DU GRH (1)

PARTIE 5 : GUIDE D'UTILISATION

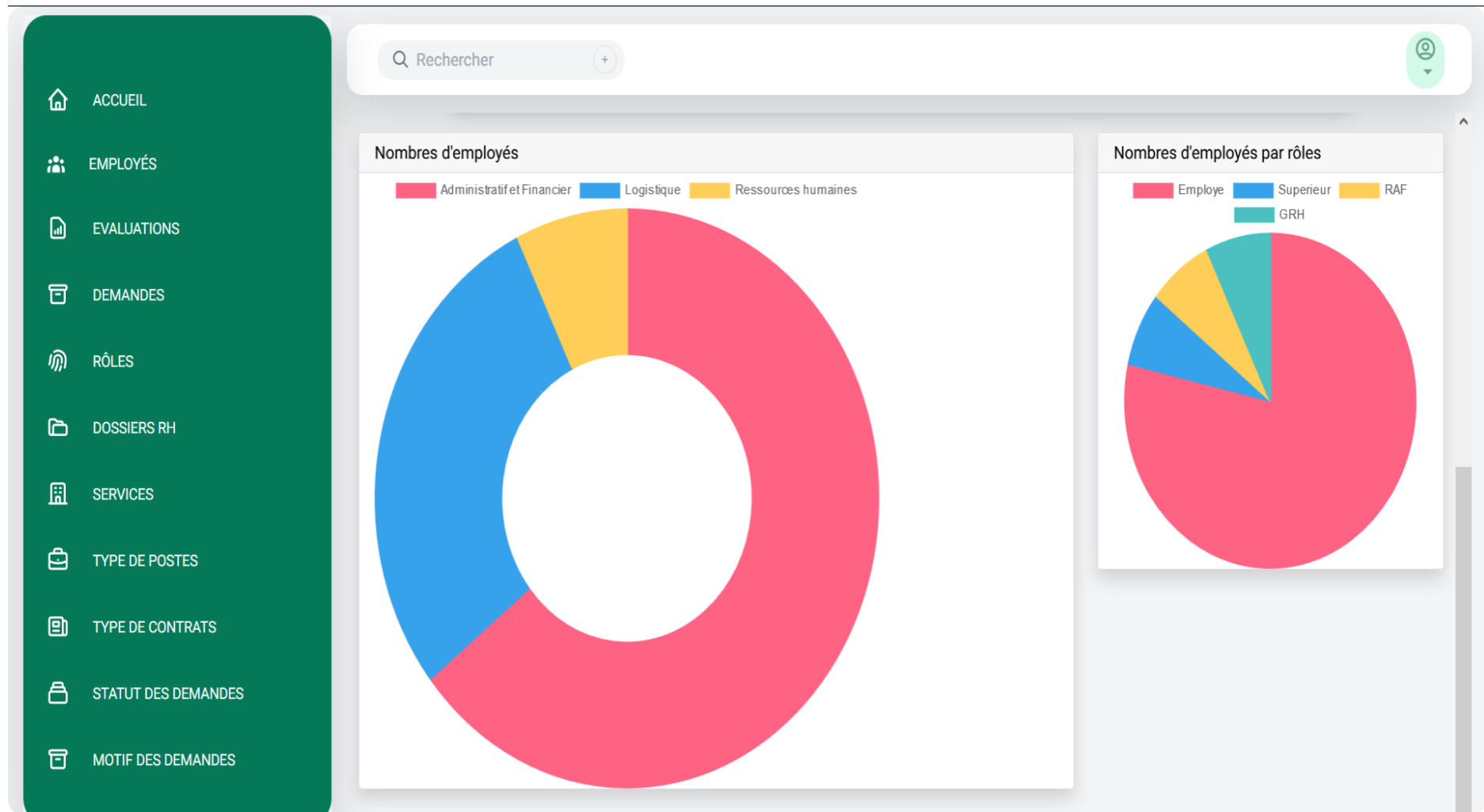


FIGURE 39: PAGE DE STATISTIQUES DU GRH (2)

5.4.2 Page de statistiques des congés pour les utilisateurs

Cette page permet aux utilisateurs d'avoir un récapitulatif du nombre de jours de congés restant par année, en plus d'un calendrier retraçant les congés accordés.

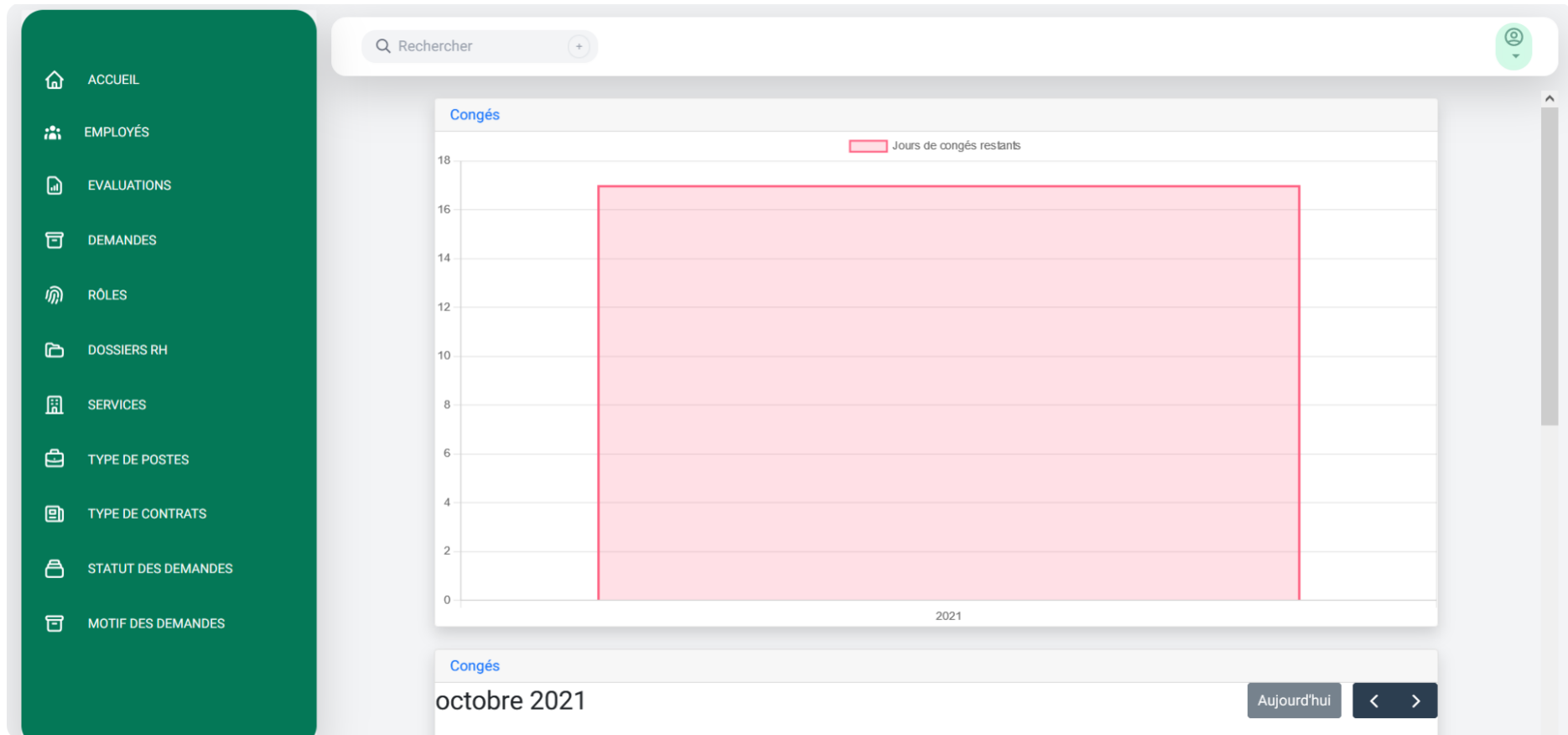


FIGURE 40: PAGE DES STATISTIQUES DES CONGES POUR LES UTILISATEURS (1)

PARTIE 5 : GUIDE D'UTILISATION

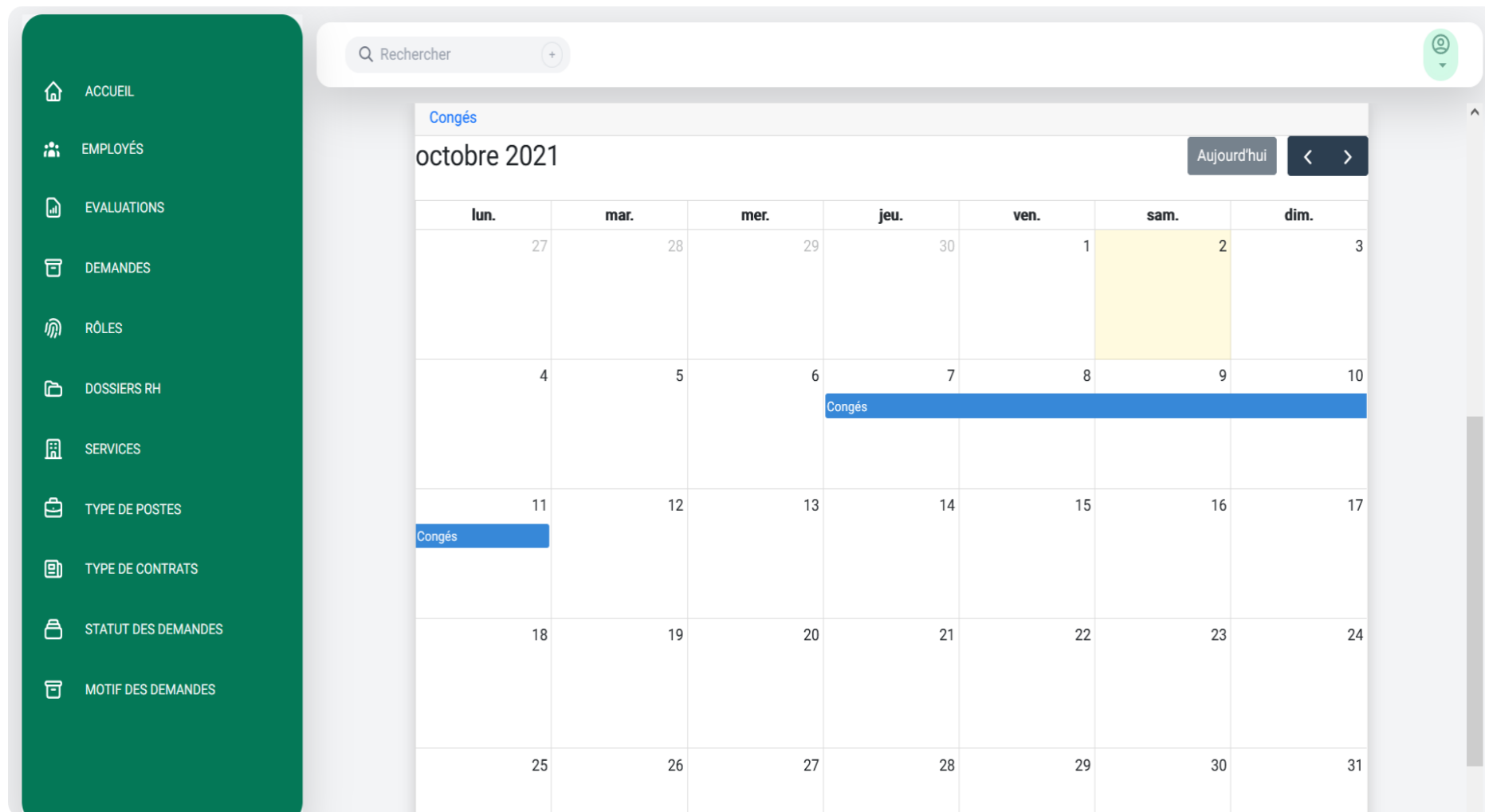


FIGURE 41: PAGE DES STATISTIQUES DES CONGES POUR LES UTILISATEURS (2)

CONCLUSION

Nous avons eu l'occasion d'effectuer un stage d'une durée de trois mois au sein de l'ONG **Mivo Energie** qui souhaite disposer d'une application permettant de faciliter et d'automatiser les demandes de congés et les évaluations de son personnel.

Nous avons conçu notre service et notre application à l'aide des framework **FastAPI** basé sur le langage de programmation **Python** et **Angular** basé sur **TypeScript**, **Html** et **Css**.

Ces trois mois ont été pour nous, l'occasion de nous confronter au monde professionnel et à ses réalités.

Cette opportunité nous a aussi permis de mettre en pratique les connaissances et les enseignements acquis au cours de notre cursus au sein de **l'Institut Africain d'Informatique**. Ils nous ont aussi permis d'apprendre à travailler et à communiquer avec des clients qui ne sont pas forcément de notre domaine et ainsi consolider nos compétences en relations humaines.

En perspective, notre service pourrait s'étendre par l'ajout de nouvelles fonctionnalités et proposé à d'autres entreprises désireuses de l'utiliser.

BIBLIOGRAPHIE INDICATIVE

❖ **Ouvrages**

- UML 2 en action : De l'analyse des besoins à la conception. 4^e édition de **Pascal Roques et Franck Vallée** ;
- UML 2 Modéliser une application web 4^e édition de **Pascal Roques** ;

❖ **Notes de cours**

- Droit du Travail, de **M. AGANO** Enseignant à l'IAI-TOGO,
- Méthodes et processus de développement, de **M. SEWAVI** Enseignant à l'IAI-TOGO
- Plateforme et outils de développement, de **M. GBODUI**

❖ **Anciens mémoires consultés**

- « APPLICATION DE MONITORING DU SYSTEME D'APPROVISIONNEMENT DES BASES DE PRODUCTION : CAS DE MOOV TOGO » de **EGBAKOU Kodjo Laurent** (2017 - 2018) ;
- « GESTION INFORMATISEE D'APPROVISIONNEMENT DE SOCIETE TELECOM : VALIDATION DE DEMANDES D'ACHATS » de **ADANTO Améto Cornelia** (2019 - 2020) ;
- « MISE EN PLACE D'UNE PLATEFORME DE COLLABORATION ENTRE VOYAGEURS ET EXPEDITEURS DE COLIS » de **ANANI Komivi Gédéon** (2017 - 2018) ;
- « MISE EN PLACE D'UNE PLATEFORME POUR LA GESTION INFORMATISÉE DU TRANSPORT EN COMMUN AU TOGO » de **MABANTEY Siegfried Dametote** (2019 - 2020)

WEBOGRAPHIE INDICATIVE

LIENS	DATES DE CONSULTATION
https://fastapi.tiangolo.com/	Tout au long du projet
https://stackoverflow.com/	Tout au long du projet
https://tailwindcss.com/	Tout au long du projet
https://www.youtube.com/	Tout au long du projet
https://angular.io/	Tout au long du projet
https://okteto.com/	Tout au long du projet
https://firebase.google.com/	Tout au long du projet
https://www.heroku.com	Tout au long du projet
https://www.positronx.io/	Tout au long du projet
https://www.sqlalchemy.org/	Tout au long du projet

TABLES DES MATIERES

DEDICACES.....	i
REMERCIEMENTS.....	ii
SOMMAIRE.....	iii
AVANT-PROPOS.....	iv
RESUME.....	v
ABSTRACT.....	vi
GLOSSAIRE.....	vii
LISTE DES FIGURES.....	viii
LISTE DES TABLEAUX.....	ix
LISTE DES PARTICIPANTS AU PROJET.....	x
INTRODUCTION.....	1
PARTIE 1 : CAHIER DES CHARGES.....	2
1.1 Présentations.....	3
1.1.1 Brève présentation de l'IAI-TOGO.....	3
1.1.1.1 Formation à l'IAI-TOGO.....	3
1.1.1.2 Plan de localisation.....	4
1.1.2 Présentation du cadre de stage.....	5
1.1.2.1 Statut.....	5
1.1.2.2 Mission.....	5
1.1.2.3 Activités.....	5
1.1.2.4 Quelques réalisations.....	5
1.1.2.5 Organigramme.....	6
1.1.2.6 Service d'accueil.....	7
1.1.2.7 Plan de localisation.....	7
1.2 Thème de stage.....	8
1.2.1 Présentation du sujet.....	8
1.2.2 Problématique du sujet.....	8
1.2.3 Intérêt du sujet.....	9
1.2.3.1 Objectifs.....	9
1.2.3.2 Résultats attendus.....	10
1.3 Etude de l'existant.....	11
1.3.1 Gestion des congés.....	11
1.3.2 Evaluation des performances des employés.....	12
1.4 Critique de l'existant.....	13
1.5 Propositions et choix de solutions.....	14
1.5.1 Evaluations techniques des solutions.....	14
1.5.1.1 Factorial.....	14
1.5.1.2 Développement de la solution.....	15
1.5.2 Evaluations financières des solutions.....	16
1.5.2.1 Coût matériel.....	17
1.5.2.2 Coût des solutions.....	18
1.5.2.2.1 Evaluations financières de Factorial.....	18
1.5.2.2.2 Développement de la solution.....	20
1.5.3 Choix de la solution.....	22

Mise en place d'un service de gestion de congés et d'évaluation des performances du personnel d'une entreprise

1.6	Planning prévisionnel de réalisation	23
PARTIE 2 : ANALYSE ET CONCEPTION		25
2.1	Choix de la méthode d'analyse et justification	26
2.1.1	Comparaison des méthodes de modélisation.....	26
2.1.2	Le langage de modélisation	29
2.1.3	Le processus de développement	30
2.2	Choix de l'outil de modélisation et justification	31
2.3	Etude détaillée de la solution	32
2.3.1	Etude préliminaire	32
2.3.1.1	Les acteurs.....	32
2.3.2	Capture des besoins fonctionnels.....	33
2.3.2.1	Diagrammes de cas d'utilisation	33
2.3.2.1.1	Les cas d'utilisation	33
2.3.2.1.2	Représentation des diagrammes de cas d'utilisation	36
2.3.2.1.3	Description textuelle de quelques cas d'utilisation	38
2.3.2.1.3.1	Cas d'utilisation « S'authentifier »	38
2.3.2.1.3.2	Cas d'utilisation « Evaluer »	39
2.3.2.1.3.3	Cas d'utilisation « Faire une demande de congés »	40
2.3.2.2	Les diagrammes d'activité.....	41
2.3.2.2.1	Diagrammes d'activité « S'authentifier »	41
2.3.2.2.2	Diagrammes d'activités « Faire une demande de congés »	42
2.3.2.2.3	Diagramme d'activité « Evaluer un employé »	43
2.3.2.2.4	Diagrammes d'activité « Lancer une campagne d'évaluation »	44
2.3.2.3	Les diagrammes de séquences	45
2.3.2.3.1	Diagrammes de séquences « S'authentifier »	46
2.3.2.3.2	Diagramme de séquences « Faire une demande de congés »	47
2.3.2.3.3	Diagramme de séquence « Valider une demande de congés »	48
2.3.2.3.4	Diagramme de séquence « Lancer une campagne d'évaluation »	49
2.3.2.3.5	Diagramme de séquence « Evaluer un employé »	50
2.3.3	Capture des besoins techniques.....	51
2.3.4	Conception	52
2.3.4.1	Diagramme de déploiement	52
2.3.4.1	Diagramme de classes.....	52
PARTIE 3 : REALISATION ET MISE EN ŒUVRE		55
3.1	Matériels et logiciels utilisés.....	56
3.1.1	Matériels utilisés.....	56
3.1.2	Logiciels utilisés	57
3.1.2.1	Backend	57
3.1.2.1.1	Python	57
3.1.2.1.2	FastAPI.....	58
3.1.2.1.3	Pydantic.....	58
3.1.2.1.4	SQLAlchemy.....	59
3.1.2.1.5	Pipenv.....	59
3.1.2.1.6	Postman	60
3.1.2.1.7	Swagger UI.....	60
3.1.2.1.8	Uvicorn	61
3.1.2.1.9	Google Cloud Storage for Firebase	61
3.1.2.1.10	Docker	61
3.1.2.1.11	PostgreSQL	62
3.1.2.1.12	PgAdmin	63
3.1.2.1.13	SQLite.....	63
3.1.2.1.14	Mailtrap.io	63
3.1.2.1.15	Gmail	64
3.1.2.1.16	Visual Studio Code	64

Mise en place d'un service de gestion de congés et d'évaluation des performances du personnel d'une entreprise

3.1.2.2	Frontend	64
3.1.2.2.1	Angular	64
3.1.2.2.2	TypeScript	65
3.1.2.2.3	Npm	65
3.1.2.2.4	Sass	66
3.1.2.2.5	TailwindCSS	66
3.1.2.2.6	Node.JS	67
3.1.2.2.7	WebStorm	67
3.1.2.3	Gestion de version	68
3.1.2.3.1	Git	68
3.1.2.3.2	Github	68
3.1.2.4	Choix des hébergeurs	69
3.1.2.4.1	Okteto (API & PGADMIN)	69
3.1.2.4.2	Firebase (Frontend)	69
3.1.2.4.3	Heroku (Bases de données)	69
3.2	Architectures matérielle et logicielle de l'application	70
3.2.1	Architecture matérielle	70
3.2.2	Architecture logicielle	71
3.2.2.1	API	71
3.2.2.2	Application web	72
3.3	Sécurité de l'application	73
3.3.1	Authentification et autorisation	73
3.3.2	Cross-Origin Resource Sharing (CORS)	74
3.3.3	Hypertext Transfer Protocol Secure (HTTPS)	74
3.4	Mise en place de la base de données	75
3.4.1	Le fichier session.py	75
3.4.2	Les modèles SQLAlchemy	75
3.4.3	Les opérations de CRUD et les requêtes	75
3.4.4	Quelques exemples montrant la mise en place de notre base de données	76
PARTIE 4 : GUIDE D'EXPLOITATION		80
4.1	Configuration logicielle et matérielle	80
4.1.1	Configuration logicielle	80
4.1.2	Configuration matérielle	81
4.2	Déploiement et suivi	83
4.2.1	Déploiement de l'API	83
4.2.2	Déploiement de la base de données	87
4.2.3	Déploiement de l'application web	87
4.3	Maintenance	89
PARTIE 5 : GUIDE D'UTILISATION		90
5.1	Description textuelle du logiciel	91
5.2	Plan de navigation	92
5.3	Présentation des différentes interfaces	94
5.3.1	Images de quelques pages de notre application	95
5.4	Présentation des états	99
5.4.1	Page des statistiques pour le GRH	99
5.4.2	Page de statistiques des congés pour les utilisateurs	101
CONCLUSION		103
BIBLIOGRAPHIE INDICATIVE		I
WEBOGRAPHIE INDICATIVE		II
TABLES DES MATIERES		III