République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'hamed Bougara
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Licence en informatique

Spécialité : Systèmes Informatiques (SI)

Thème

Conception d'un portail web dédié aux agents de la Division Production SONATRACH

Encadré par :
M. Soukeur Abdelouahed

Membres du jury :
- M. Boudaoud Sidahmed
- Mme. Sabba-Chehbour Fairouz

Réalisé par :
- Benbelkacem Zineddine
- Belounis Mellina Malak
- Selmoune Maria

Projet n° : EXT-012
Année universitaire : 2025 / 2026

Remerciements

Nous tenons à expressément remercier toutes les personnes qui ont contribué au succès de ce projet de fin d'études.

Nous adressons nos sincères remerciements à notre encadrant au sein de SONATRACH, Monsieur **SOUKEUR Abdelouahad**, Ingénieur Systèmes et Exploitation à la direction informatique, pour son encadrement précieux, ses conseils avisés et sa disponibilité tout au long de ce travail.

Nous remercions également les membres du jury, Monsieur **Boudaoud Sidahmed** et Madame **Sabba-Chehbour Fairouz**, pour l'honneur qu'ils nous font en acceptant d'évaluer notre travail.

Enfin, nos remerciements s'adressent à l'ensemble des enseignants du Département d'Informatique qui nous ont guidés durant notre cursus universitaire, ainsi qu'à nos familles et amis pour leur soutien indéfectible.

Résumé

Les systèmes d'information des ressources humaines modernes nécessitent des solutions numériques capables d'améliorer la communication entre les employés et les services administratifs. Cependant, l'application existante RESHUM, utilisée au sein de la Division Production de SONATRACH, reste principalement destinée aux gestionnaires RH et ne permet pas aux agents d'accéder directement à leurs informations ni de gérer leurs demandes de manière autonome.

Ce projet présente la conception et la réalisation d'un portail web dédié aux agents de la Division Production de SONATRACH. Cette plateforme permet aux employés de consulter leurs informations administratives, de soumettre des demandes de congés, de suivre leurs ordres de mission, de recevoir des notifications et d'accéder à leurs documents administratifs générés automatiquement.

Le système repose sur une architecture moderne à trois niveaux, utilisant HTML, CSS, JavaScript, FastAPI et MySQL, avec une authentification sécurisée basée sur les jetons JWT. Cette solution contribue à la dématérialisation des procédures administratives, améliore la communication entre les agents, la hiérarchie et les ressources humaines, et facilite la gestion des processus administratifs au sein de SONATRACH.

Table des matières

Remerciements **1**

Résumé **2**

Table des figures **5**

Introduction Générale **6**

1 ÉTUDE DE L'ENVIRONNEMENT DE TRAVAIL **7**
1.1 Introduction . . . 7
1.2 Présentation de l'organisme d'accueil : SONATRACH . . . 7
1.3 Présentation de la structure d'accueil . . . 7
1.3.1 La Division Production (DP) . . . 7
1.3.2 La Direction Informatique . . . 8
1.4 L'informatique au service des RH : Analyse critique de RESHUM . . . 9
1.5 Présentation de l'application existante (RESHUM) . . . 10
1.5.1 Architecture technique . . . 10
1.5.2 Principales fonctionnalités de RESHUM . . . 10
1.5.3 Limites et motivations de la mise en place du portail web . . . 10
1.6 Conclusion . . . 11

2 MODÉLISATION ET CONCEPTION UML **12**
2.1 Introduction . . . 12
2.2 Démarche de conception . . . 12
2.3 Modélisation fonctionnelle : Le Diagramme de Cas d'Utilisation . . . 12
2.4 Modélisation comportementale : Le Diagramme d'activités . . . 13
2.5 Modélisation dynamique : Le Diagramme de séquences . . . 14
2.6 Modélisation statique : Le Diagramme de classes . . . 15
2.7 Modèle Relationnel de la Base de Données (MRD) . . . 16
2.7.1 Représentation textuelle formelle . . . 17
2.8 Conclusion . . . 17

3 RÉALISATION ET IMPLÉMENTATION **18**
3.1 Introduction . . . 18
3.2 Environnement logiciel . . . 18
3.2.1 Les technologies utilisées . . . 18
3.2.2 Les outils de développement utilisés . . . 19
3.3 Architecture technique globale du système . . . 20

3.4 Présentation des interfaces graphiques et fonctionnalités 20
3.4.1 Module d'authentification (Sécurité JWT) 20
3.4.2 Tableau de bord et Système d'alertes (Vue Agent) 21
3.4.3 Module de gestion des congés (Self-Service) 23
3.4.4 Module de gestion des ordres de mission 24
3.4.5 Module Boîte de Réception (Espace Documents de l'Agent) 25
3.4.6 Module de Génération Dynamique de Documents Administratifs 26
3.5 Conclusion 27

Conclusion Générale **28**
Bibliographie et Webographie 29

Table des figures

1.1 Organigramme global de la Division Production (DP) SONATRACH 8
1.2 Structure interne de la Direction Informatique (DI) 9
1.3 Fonctionnalités de RESHUM 11

2.1 Diagramme de cas d'utilisation global du portail collaborateur 13
2.2 Diagramme d'activités du processus de gestion et validation des demandes de congé 14
2.3 Diagramme de séquences des demandes de congé 15
2.4 Diagramme de classes global du système de gestion des ressources humaines (Vue d'ensemble) 16

3.1 Architecture technique à trois niveaux (3-Tier) du Portail Collaborateur 20
3.2 Interface de la page de connexion (Login) 21
3.3 Interface du Tableau de bord général et de la cloche d'alertes 21
3.4 Menu déroulant du Centre de notifications RH (Détail UI) 22
3.5 Interface de formulation d'une demande de congé (Vue Agent) 23
3.6 Interface d'approbation et de validation des congés (Vue Responsable) 24
3.7 Interface de création et d'affectation d'un ordre de mission (Vue Gestionnaire) . 24
3.8 Interface d'affichage et de suivi des ordres de mission (Vue Agent) 25
3.9 Interface de la Boîte de Réception (Vue Agent) 25
3.10 Interface de l'Attestation de Travail générée dynamiquement 26
3.11 Interface de la Fiche de Paie mensuelle générée 27

Introduction Générale

Dans le cadre de la modernisation de ses infrastructures informatiques, la Division Production de SONATRACH s'est dotée d'un système de gestion des ressources humaines performant nommé RESHUM. Cependant, bien que cet outil soit optimisé pour les gestionnaires RH, les agents et les managers ne disposent pas d'un accès direct en self-service pour soumettre leurs demandes administratives ou consulter leurs informations en temps réel. C'est dans ce contexte que s'inscrit notre projet de fin d'études.

L'objectif principal de ce travail a été de concevoir et de développer une solution web moderne, sécurisée et centralisée sous forme de Portail Collaborateur (SIRH). Ce portail sert de passerelle intelligente, permettant de dématérialiser les processus métiers lourds et répétitifs afin de redonner de l'autonomie aux collaborateurs de l'entreprise.

Afin d'exposer clairement la démarche scientifique, l'analyse des besoins et les choix d'implémentation logicielle qui ont guidé la concrétisation de ce projet, ce mémoire est structuré en trois grands chapitres fondamentaux.

Le premier chapitre est consacré à l'étude détaillée de l'environnement de travail et à l'organisme d'accueil. Cette phase d'immersion permet de présenter la structure de la Division Production de SONATRACH ainsi que sa Direction Informatique. Elle dresse également une analyse critique et fonctionnelle de l'application cliente RESHUM afin de contextualiser les limites actuelles et de motiver le déploiement de notre solution web décentralisée.

Le deuxième chapitre formalise l'étape charnière de la modélisation et de la conception UML du système. À travers l'élaboration des diagrammes comportementaux, dynamiques et statiques, nous définissons l'architecture logique des nouveaux modules métiers. Ce chapitre inclut également la spécification textuelle des règles d'intégrité et la présentation du Modèle Relationnel de Données (MRD) qui régit notre base relationnelle.

Le troisième chapitre expose la phase de réalisation technique et d'implémentation concrète du portail. Il présente l'environnement matériel et l'écosystème logiciel retenu, mettant en évidence les technologies associées à notre architecture à trois niveaux (3-Tier) propulsée par le framework Python FastAPI, la sécurité par jeton JWT et l'asynchronisme du Frontend en JavaScript natif. Ce chapitre dévoile également les interfaces graphiques finales de l'application web, illustrant le parcours utilisateur complet depuis l'authentification jusqu'à la boîte de réception des documents générés.

Enfin, une dernière section non capitulée sera dédiée à la bibliographie et à la webographie, recensant de manière exhaustive l'ensemble des mémoires, des documentations officielles et des ressources numériques ayant constitué le socle théorique indispensable à ce projet.

Chapitre 1

ÉTUDE DE L'ENVIRONNEMENT DE TRAVAIL

1.1 Introduction

Pour automatiser et moderniser efficacement le système d'information d'une organisation, il est crucial de comprendre sa structure et ses flux décisionnels. Ce premier chapitre présente la Division Production de SONATRACH, décrit son organigramme structurel, et dresse une analyse critique de l'outil existant afin de contextualiser les besoins du nouveau portail web.

1.2 Présentation de l'organisme d'accueil : SONATRACH

SONATRACH est une entreprise publique algérienne majeure, créée en 1963. Actrice centrale du secteur de l'énergie en Afrique, elle intervient sur l'ensemble de la chaîne de valeur des hydrocarbures : de l'exploration à la production, jusqu'au transport par canalisation, au raffinage et à la commercialisation internationale.

1.3 Présentation de la structure d'accueil

1.3.1 La Division Production (DP)

Placée sous l'autorité de l'activité Exploration & Production (EP), la Division Production a pour mission l'exploitation, la maintenance et le développement des gisements de pétrole et de gaz naturel. Elle gère un capital humain de plus de 20 000 agents répartis sur l'ensemble des directions régionales et des bases de vie (Hassi Messaoud, Stah, In Amenas, etc.).

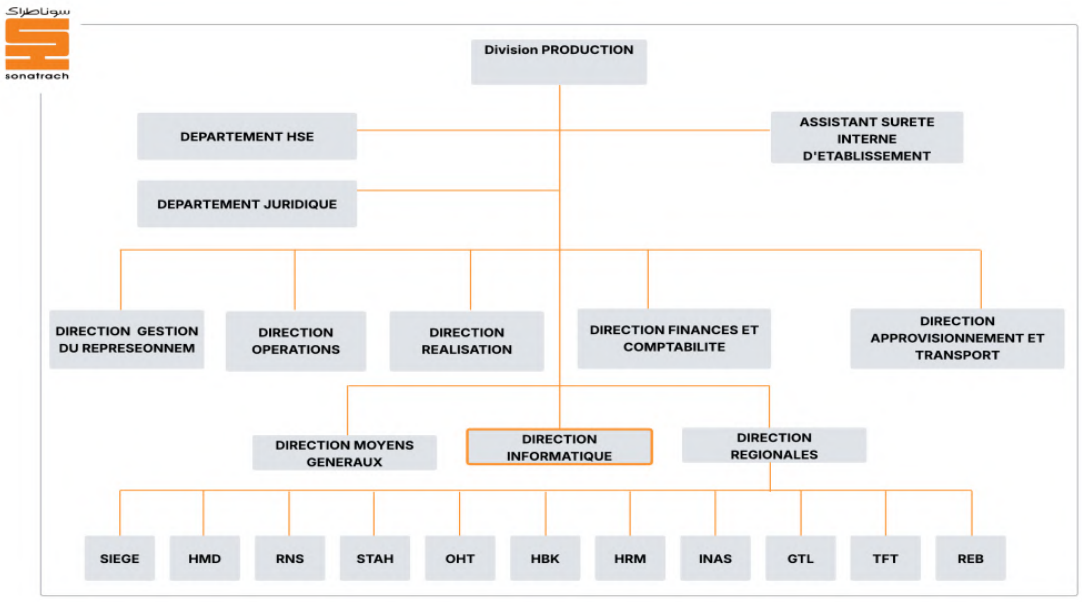


FIGURE 1.1 – Organigramme global de la Division Production (DP) SONATRACH

La figure ci-dessus illustre la répartition hiérarchique et l'organisation fonctionnelle des différentes sous-directions au sein de la Division Production, offrant une vue d'ensemble de la chaîne de commandement.

1.3.2 La Direction Informatique

La Direction Informatique de la Division Production est chargée de l'étude des besoins applicatifs, du déploiement des infrastructures réseaux et du support aux systèmes d'information métiers. C'est au sein de son Département Étude & Développement qu'a été initié ce projet de portail collaborateur.

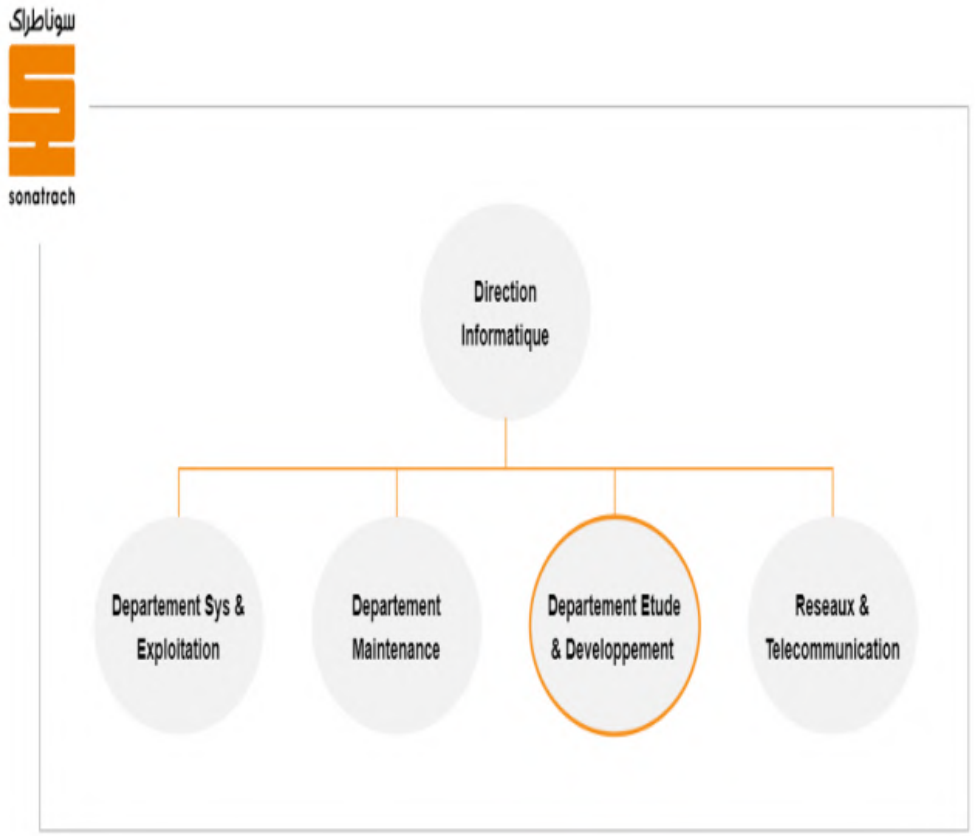


FIGURE 1.2 – Structure interne de la Direction Informatique (DI)

Ce schéma détaille l'organisation interne de la Direction Informatique, mettant en évidence ses départements et services opérationnels clés chargés de maintenir et de développer le système d'information.

1.4 L'informatique au service des RH : Analyse critique de RESHUM

Depuis 1999, l'application interne **RESHUM** centralise l'ensemble des données de gestion administrative, de la paie et du suivi social des employés. Bien que modulaire et robuste, RESHUM souffre d'une architecture client-serveur classique fermée, accessible uniquement aux terminaux des gestionnaires de la direction des ressources humaines.

Cette centralisation stricte génère plusieurs points de friction majeurs :

— **Absence de Self-Service :** Les agents n'ont aucun moyen de consulter en ligne leur profil, leurs compteurs de congés, ou leurs ordres de mission.

— **Circuit d'approbation physique :** Les demandes de congés ou d'attestations nécessitent l'impression de formulaires papier et des signatures manuelles, entraînant des délais de traitement chronophages.

— **Surcharge administrative :** Les équipes RH passent une part importante de leur temps à répondre à des requêtes d'information basiques ou à éditer manuellement des pièces administratives courantes.

1.5 Présentation de l'application existante (RESHUM)

L'application **RESHUM** (Ressources Humaines) est le cœur du système d'information de gestion de la Division Production de SONATRACH. Déployée à la fin des années 90, elle permet de centraliser le cycle de vie administratif de chaque agent.

1.5.1 Architecture technique

RESHUM repose sur une architecture client-serveur classique à deux niveaux (2-Tier) :

— **Le client lourd :** Une interface graphique installée localement sur les postes de travail de la Direction des Ressources Humaines.

— **Le serveur de base de données :** Une base relationnelle centralisée gérant l'ensemble des tables (informations agents, soldes de congés, historiques des carrières).

1.5.2 Principales fonctionnalités de RESHUM

L'outil actuel est structuré autour de plusieurs grands modules métiers :

1. **Gestion des dossiers du personnel :** Saisie et mise à jour des fiches individuelles des agents (coordonnées, situation familiale, affectation, échelon).
2. **Mouvements de carrière :** Suivi des promotions, des mutations entre directions régionales et du calcul de l'ancienneté.
3. **Gestion du temps et des absences :** Suivi réglementaire des congés annuels, des congés de maladie et des récupérations.
4. **Gestion de la paie et des éléments variables :** Calcul des salaires, génération des bulletins de paie et édition des états comptables réglementaires.

1.5.3 Limites et motivations de la mise en place du portail web

Bien que performant pour le traitement de masse de la paie, RESHUM ne répond plus aux exigences modernes d'accessibilité et de décentralisation de l'information :

— **L'accès restreint :** Un agent travaillant sur une base de vie ou un site de production distant n'a aucun moyen de consulter son dossier ou son solde de congés de manière autonome.

— **Flux asynchrones manquants :** Toute action de validation doit être retranscrite manuellement sur papier avant d'être saisie par un gestionnaire dans RESHUM.

C'est pour combler ces lacunes structurelles que notre solution intervient comme une extension web dynamique et sécurisée, exposant les fonctionnalités nécessaires sous forme d'API sans altérer le cœur de la base RESHUM.

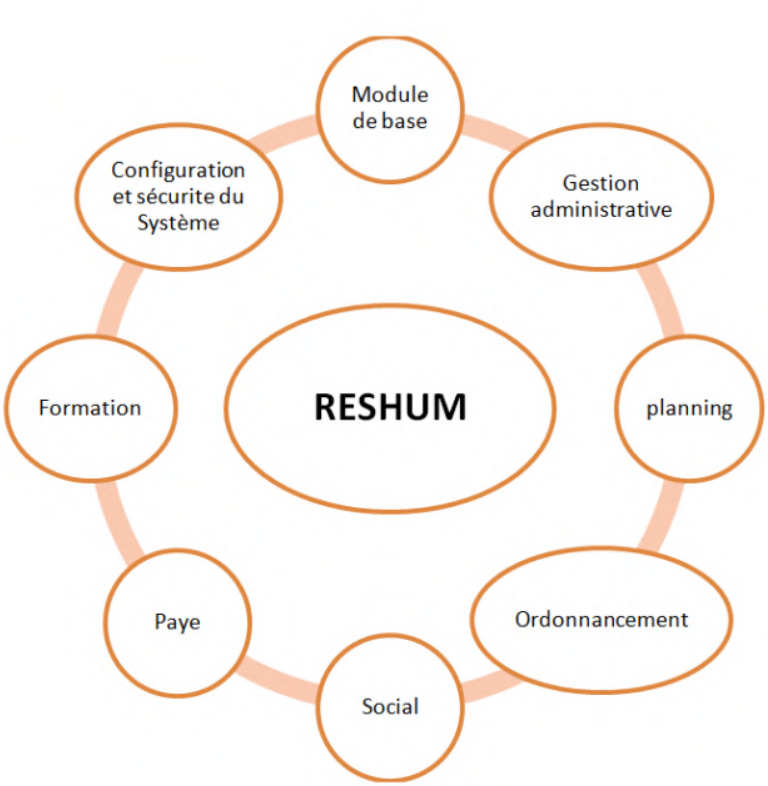


FIGURE 1.3 – Fonctionnalités de RESHUM

Cette représentation schématise les principaux modules fonctionnels couverts par l'application actuelle RESHUM, illustrant la centralisation des données RH.

1.6 Conclusion

En conclusion, l'étude détaillée de l'application RESHUM a mis en évidence la nécessité d'implémenter une solution moderne pour redonner de l'autonomie aux agents. En agissant comme une surcouche web sécurisée par token JWT et propulsée par FastAPI, le portail à développer complétera la robustesse du système existant.

Chapitre 2

MODÉLISATION ET CONCEPTION UML

2.1 Introduction

La phase de modélisation et de conception est une étape charnière dans le cycle de vie d'un projet logiciel. Elle permet de traduire les exigences fonctionnelles identifiées lors de l'analyse de l'existant en une architecture technique claire et structurée. Pour ce faire, nous avons adopté le langage de modélisation unifié **UML** (Unified Modeling Language), standard de l'industrie pour documenter les systèmes orientés objet.

2.2 Démarche de conception

Afin de cadrer rigoureusement les fonctionnalités clés de notre Portail Collaborateur (gestion des congés, ordres de mission et suivi des pointages), notre démarche s'articule autour de trois axes de modélisation :

1. **La modélisation fonctionnelle (Le Diagramme de Cas d'Utilisation) :** Pour identifier les acteurs et les frontières du système.
2. **La modélisation comportementale (Diagramme d'activités) :** Pour représenter la logique métier, l'enchaînement des actions et les flux de décision.
3. **La modélisation dynamique (Diagramme de séquences) :** Pour illustrer l'ordre chronologique des interactions entre les utilisateurs et les serveurs.
4. **La modélisation statique (Diagramme de classes) :** Pour définir la structure de la base de données relationnelle MySQL et l'organisation des entités.

2.3 Modélisation fonctionnelle : Le Diagramme de Cas d'Utilisation

Le diagramme de cas d'utilisation permet de définir les limites du système et les interactions entre les différents acteurs (Agent, Responsable, Gestionnaire RH) et les fonctionnalités offertes par le portail.

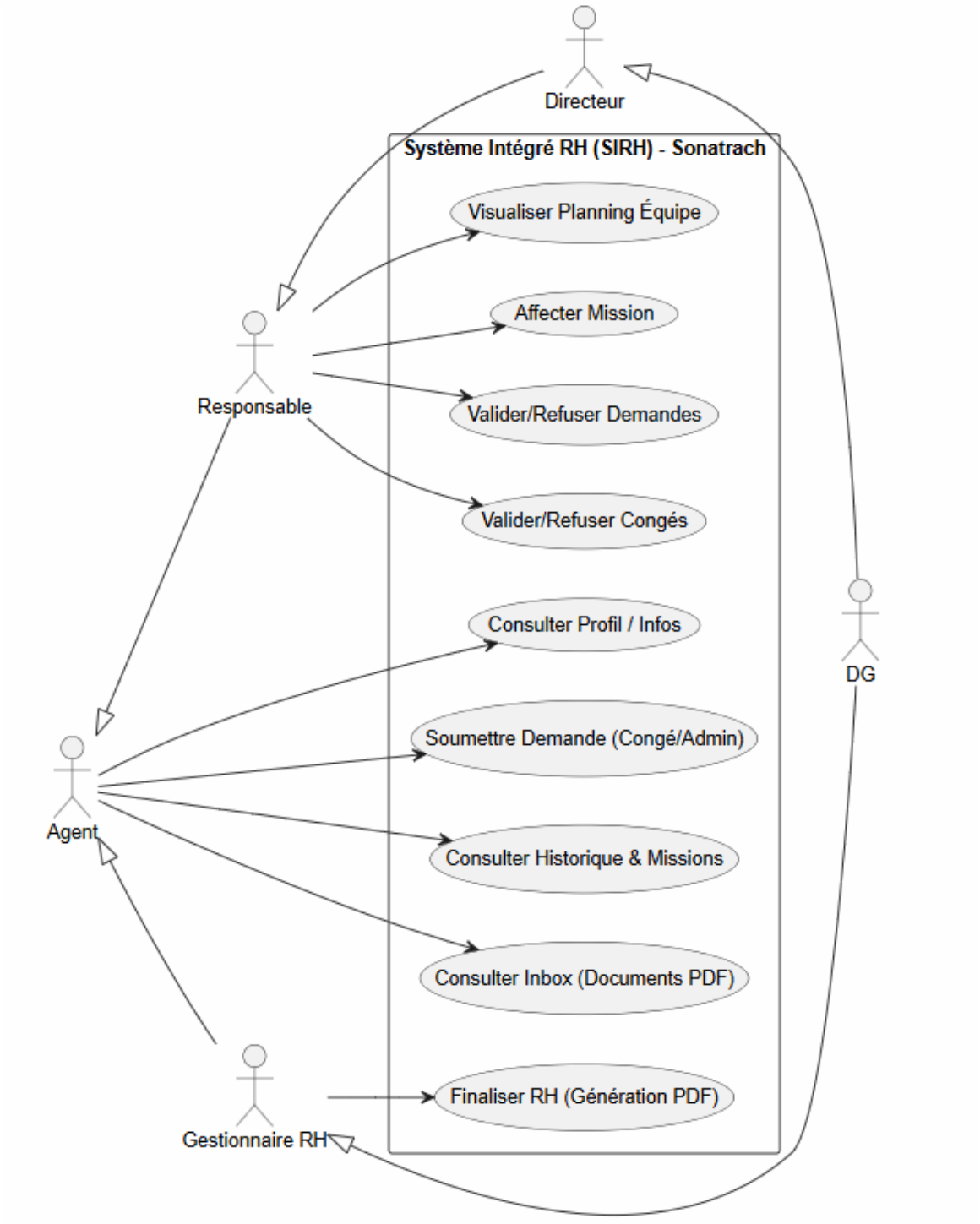


FIGURE 2.1 – Diagramme de cas d'utilisation global du portail collaborateur

Ce diagramme illustre les services globaux offerts par le système, notamment la soumission des demandes, le processus de validation hiérarchique et la consultation des documents générés.

2.4 Modélisation comportementale : Le Diagramme d'activités

Le diagramme d'activités permet de cartographier de manière séquentielle le workflow métier et les flux de décision au sein de l'organisation lors du traitement d'une demande de congé.

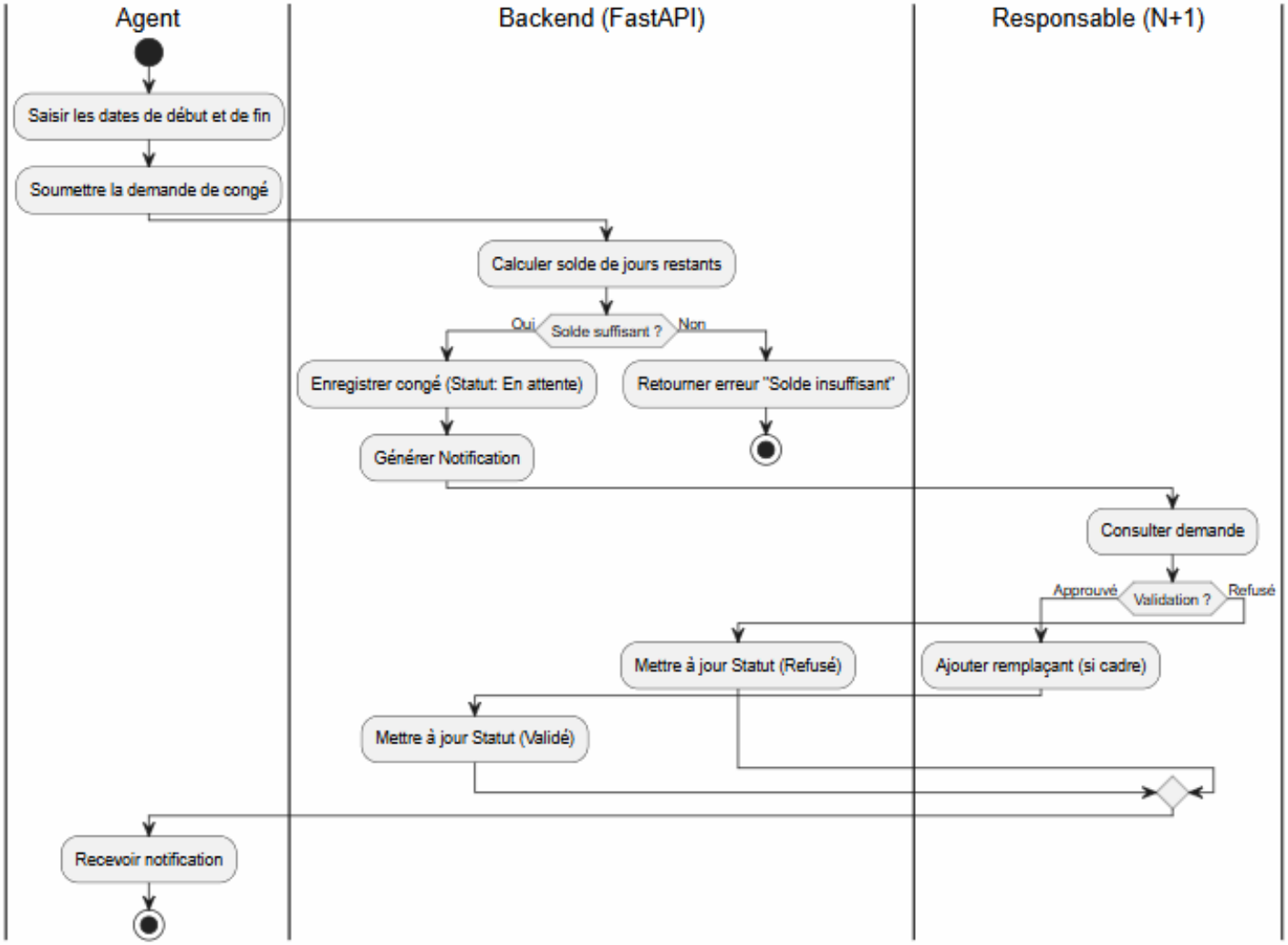


FIGURE 2.2 – Diagramme d'activités du processus de gestion et validation des demandes de congé

Ce diagramme trace le cheminement logique d'une demande de congé, depuis sa création par l'agent jusqu'à sa validation finale par la hiérarchie en passant par la vérification des soldes.

2.5 Modélisation dynamique : Le Diagramme de séquences

Le diagramme de séquences décrit la dimension temporelle du système en montrant pas à pas comment les objets collaborent à travers l'échange de messages. Il est particulièrement crucial dans notre architecture découplée pour valider le comportement asynchrone des services.

Ce diagramme illustre le scénario d'authentification et de soumission d'une requête :

— L'acteur envoie ses identifiants via le formulaire web (Frontend).

— Le contrôleur JavaScript intercepte l'action et effectue un appel asynchrone via la **Fetch API** vers le endpoint correspondant du Backend.

— L'API **FastAPI** traite la demande, vérifie la validité du Token **JWT**, interroge la base de données **MySQL** via l'ORM, puis retourne une réponse structurée en JSON permettant

au Frontend de rafraîchir dynamiquement l'affichage du DOM.

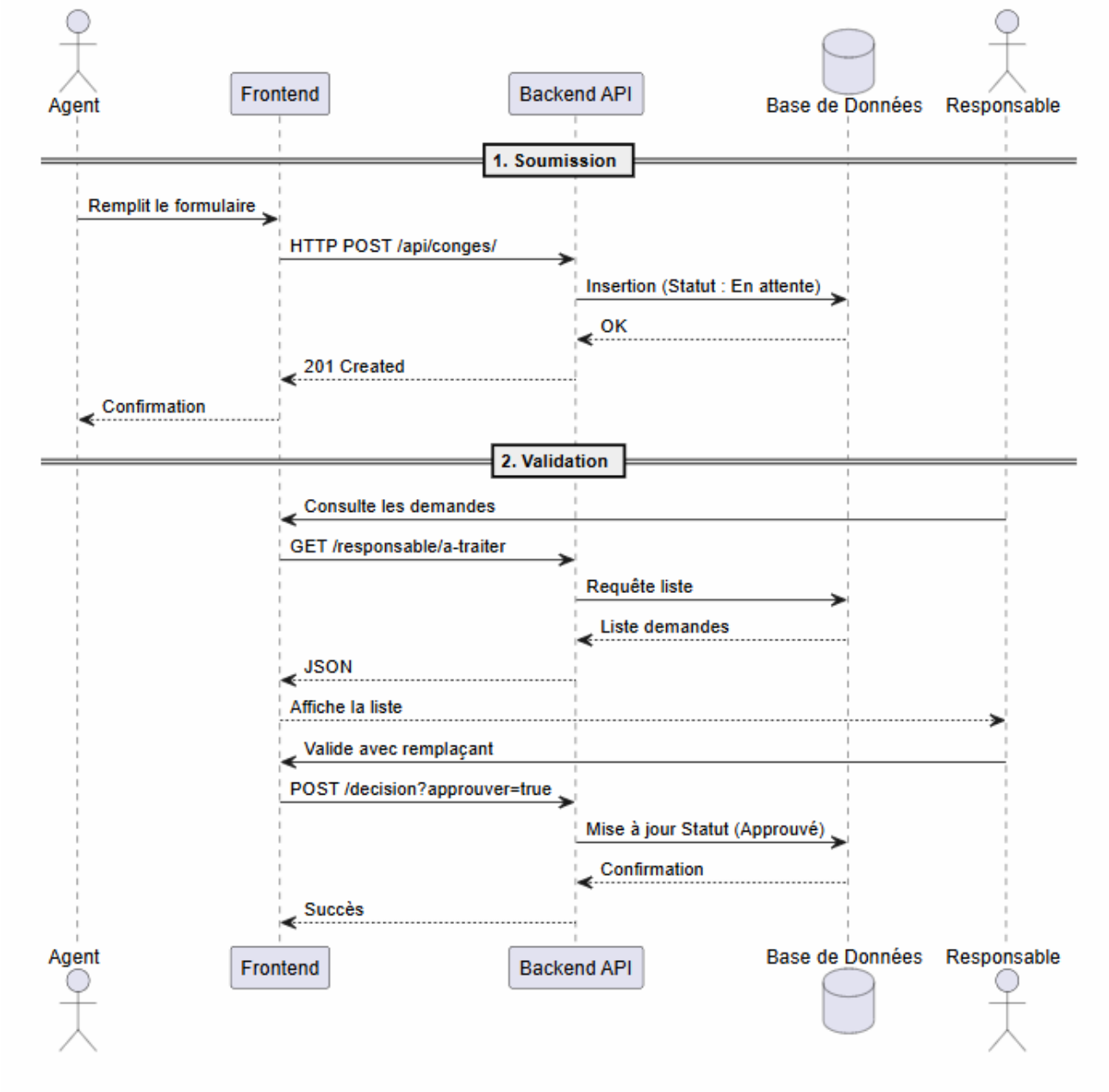


FIGURE 2.3 – Diagramme de séquences des demandes de congé

Ce schéma met en évidence les échanges de requêtes HTTP et de réponses JSON asynchrones entre l'interface utilisateur, l'API Backend et la base de données.

2.6 Modélisation statique : Le Diagramme de classes

Le diagramme de classes représente la structure stable et permanente du système. Il détaille les attributs, les types de données et les méthodes des classes, servant de fondation directe pour générer nos modèles de données avec SQLAlchemy.

Ce diagramme regroupe les entités fondamentales du domaine des ressources humaines de la Division Production, notamment les classes `Agent` (matricule, nom, rôle, structure), `Conge`,

`Mission`, `Demande`, et `Pointage`. Les liaisons de type un-à-plusieurs (1..∗) y sont explicitées pour garantir l'intégrité référentielle des données.

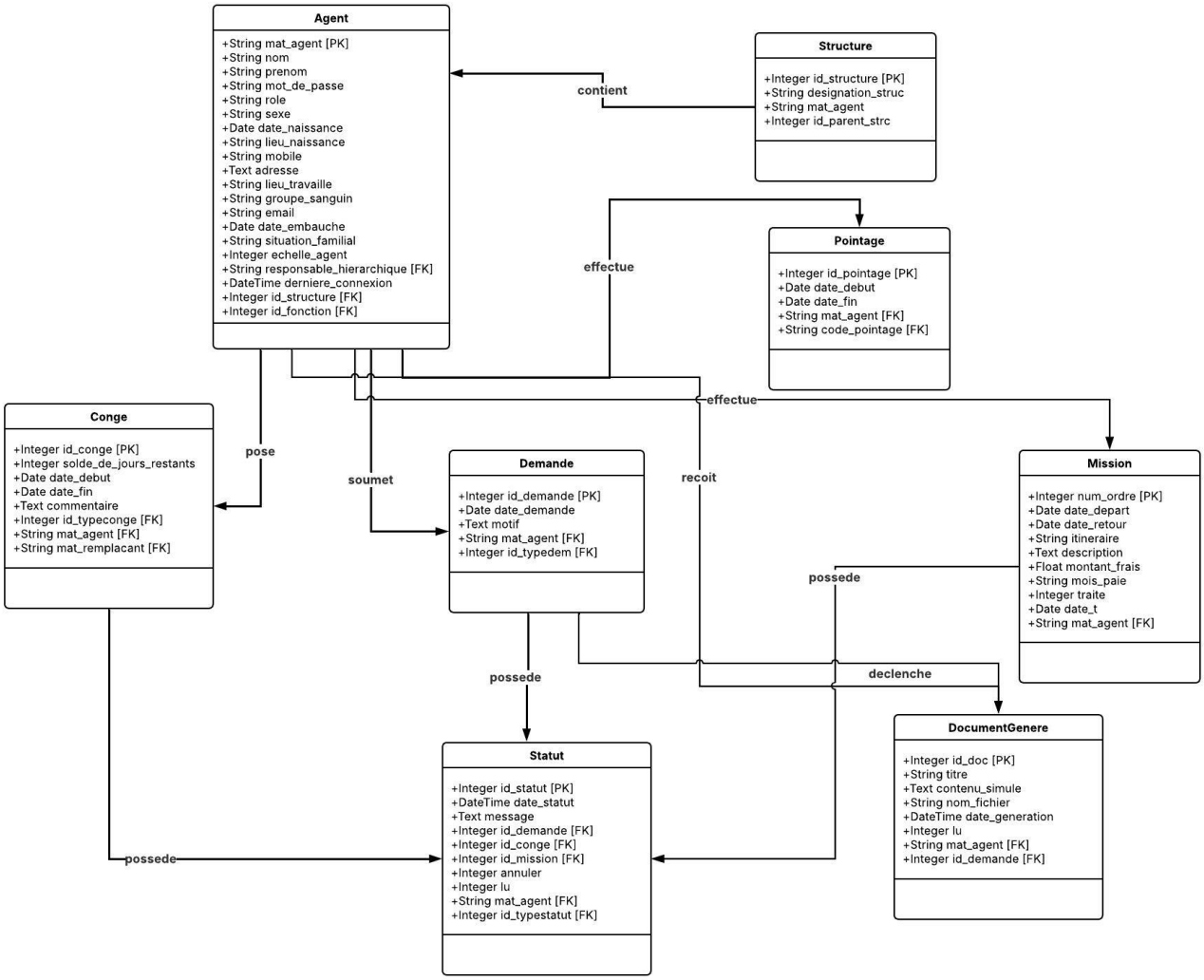


FIGURE 2.4 – Diagramme de classes global du système de gestion des ressources humaines (Vue d'ensemble)

Ce modèle conceptuel expose les entités principales de notre système, leurs attributs techniques, ainsi que les relations de cardinalité qui garantissent la cohérence des données.

2.7 Modèle Relationnel de la Base de Données (MRD)

Le schéma relationnel suivant formalise la structure de notre base de données relationnelle `MySQL`. Les clés primaires sont soulignées et les clés étrangères sont suivies d'un astérisque (*).

1. **Structure** (id_structure, designation_struc, id_parent_strc*)
2. **Fonction** (id_fonction, designation_fonction)
3. **Agent** (mat_agent, nom, prenom, role, email, mot_de_passe, echelle_agent, responsable_hierarchique, id_structure*, id_fonction*)
4. **TypeConge** (id_typeconge, designation)
5. **TypeDemande** (id_typedem, designation)

6. **TypeStatut** (id_typestatut, designation)
7. **CodePointage** (code_pointage, designation)
8. **Demande** (id_demande, date_demande, motif, mat_agent*, id_typedem*)
9. **Conge** (id_conge, date_debut, date_fin, solde_jours_restants, commentaire, mat_remplacant, id_demande*, id_typeconge*)
10. **Mission** (num_ordre, date_depart, date_retour, itineraire, montant_frais, traite, id_demande*)
11. **Pointage** (id_pointage, date_debut, date_fin, mat_agent*, code_pointage*)
12. **Statut** (id_statut, date_statut, message, annuler, lu, id_demande*, id_conge*, id_mission*, id_typestatut*, mat_agent*)
13. **DocumentGenere** (id_doc, titre, contenu_simule, nom_fichier, date_generation, lu, mat_agent*, id_demande*)

2.7.1 Représentation textuelle formelle

Pour une intégration rigoureuse dans le manuscrit, voici la définition formelle des clés et des contraintes d'intégrité référentielle :

— **Agent** : La clé étrangère *id_structure* fait référence à `Structure(id_structure)`. La clé étrangère *id_fonction* fait référence à `Fonction(id_fonction)`.
— **Demande** : La clé étrangère *mat_agent* fait référence à `Agent(mat_agent)` pour identifier l'émetteur.
— **Statut** : Modélise l'historique des états. Elle lie une demande, un congé ou une mission (clés étrangères *id_demande*, *id_conge*, *id_mission*) à un type de statut.
— **DocumentGenere** : Entité finale de génération. La clé étrangère *id_demande* fait référence à `Demande(id_demande)` pour assurer la traçabilité entre la requête initiale de l'agent et le fichier PDF produit par les RH.

2.8 Conclusion

En somme, ce chapitre a exposé de façon exhaustive les différentes étapes de modélisation UML, définissant ainsi les aspects comportementaux, dynamiques et statiques du futur système. Cette base conceptuelle robuste constitue le fondement nécessaire avant d'entamer la phase de développement logiciel.

Chapitre 3

RÉALISATION ET IMPLÉMENTATION

3.1 Introduction

Après avoir défini l'architecture logique et comportementale du système, ce troisième et dernier chapitre est dédié à la concrétisation technique du projet. Nous y détaillerons l'environnement technologique retenu, ainsi que l'implémentation concrète des fonctionnalités et des interfaces graphiques finales du Portail Collaborateur.

3.2 Environnement logiciel

La concrétisation technique du Portail Collaborateur repose sur le choix d'un écosystème logiciel moderne, garantissant les performances, la sécurité et la maintenabilité exigées par la Division Production de SONATRACH.

3.2.1 Les technologies utilisées

1. **HTML :**

   HTML5 (HyperText Markup Language) est le langage de balisage standard utilisé pour créer des pages Web. Il définit la structure et le contenu d'une page Web à l'aide d'une série d'éléments et de balises.

2. **CSS :**

   CSS3 (Cascading Style Sheets) est un langage informatique utilisé pour styliser les éléments écrits dans un langage de balisage comme HTML. Il sépare le contenu de la représentation visuelle du site.

3. **JavaScript :**



JavaScript est un langage de programmation interprété de haut niveau, orienté objet, utilisé pour créer des sites Web interactifs et dynamiques. Il permet aux développeurs de manipuler les éléments d'une page de manière asynchrone.

4. **FastAPI :**



Framework Python de dernière génération utilisé pour bâtir notre API. Il a été privilégié pour ses performances exceptionnelles, sa gestion native de l'asynchronisme et la génération automatique de la documentation interactive via Swagger UI.

5. **MySQL :**



MySQL est un SGBD relationnel open source qui utilise SQL pour gérer et organiser les données efficacement. Il est réputé pour sa rapidité, fiabilité et compatibilité avec plusieurs langages.

3.2.2 Les outils de développement utilisés

1. **Visual Studio Code :**



Éditeur de code gratuit et open source développé par Microsoft. Il fournit un environnement complet pour le développement, avec des outils d'édition et de débogage.

2. **XAMPP :**



Environnement de développement local permettant d'instancier et de manager notre serveur de base de données relationnelle via l'interface phpMyAdmin.

3. **Lucidchart :**



Logiciel de création de diagrammes basé sur le web. Il offre une interface intuitive pour concevoir nos diagrammes UML (classes, séquences, activités).

4. **Overleaf / CoCalc :**



Éditeur LaTeX en ligne facilitant la création, la structuration et le formatage de documents académiques professionnels comme ce manuscrit.

3.3 Architecture technique globale du système

L'interaction entre les différents langages et technologies présentés ci-dessus suit le modèle d'architecture découplée (3-Tier) représenté dans le schéma suivant :

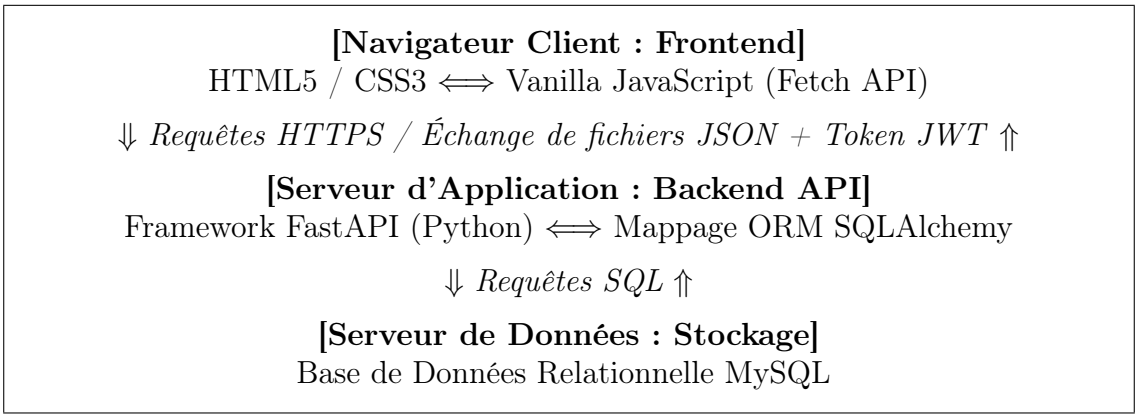


FIGURE 3.1 – Architecture technique à trois niveaux (3-Tier) du Portail Collaborateur

Cette figure résume la séparation claire en trois couches de notre application : la présentation (Frontend), la logique métier (Backend) et le stockage (Base de données).

3.4 Présentation des interfaces graphiques et fonctionnalités

Cette section présente l'environnement visuel et fonctionnel du Portail Collaborateur. Pour chaque module critique, nous mettons en correspondance l'interface utilisateur destinée à l'agent et celle dédiée au traitement par la structure hiérarchique de la Division Production de SONATRACH.

3.4.1 Module d'authentification (Sécurité JWT)

L'accès au portail est strictement sécurisé et nécessite que l'agent renseigne ses identifiants uniques (Matricule et Mot de passe).

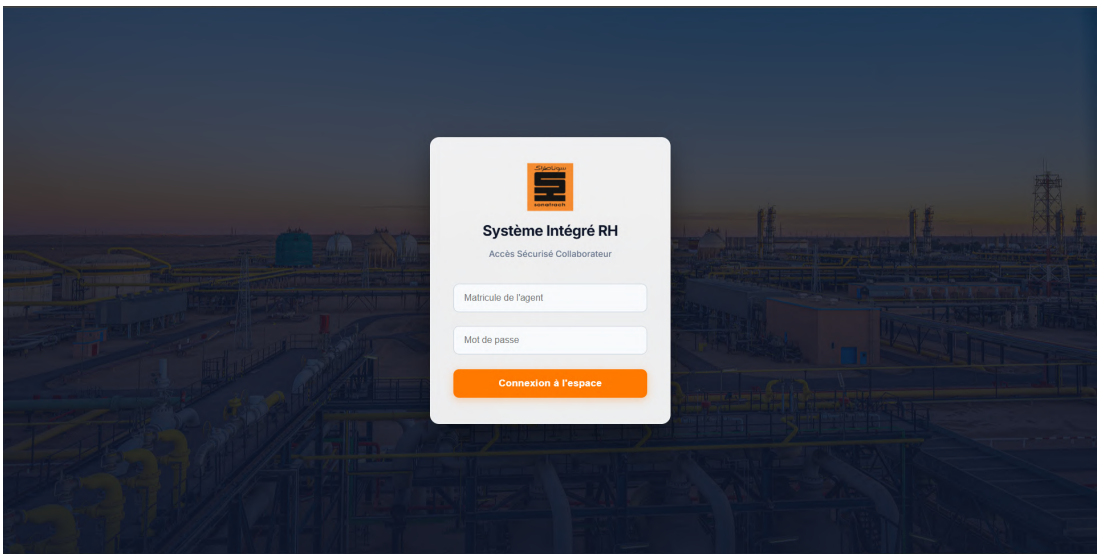


FIGURE 3.2 – Interface de la page de connexion (Login)

Cette capture d'écran montre le formulaire de connexion sécurisé permettant aux utilisateurs de s'authentifier via leurs identifiants uniques.

Utilité fonctionnelle :

— **Sécurisation de l'accès :** Elle fait office de barrière de sécurité. Lors de la soumission du formulaire, le JavaScript intercepte les données et effectue une requête `POST` asynchrone vers l'API FastAPI.

— **Génération de session :** Si les identifiants correspondent, le serveur génère et renvoie un jeton **JWT (JSON Web Token)** stocké localement dans le navigateur pour authentifier toutes les requêtes futures.

3.4.2 Tableau de bord et Système d'alertes (Vue Agent)

Une fois connecté, l'utilisateur est redirigé vers son espace d'accueil personnalisé.

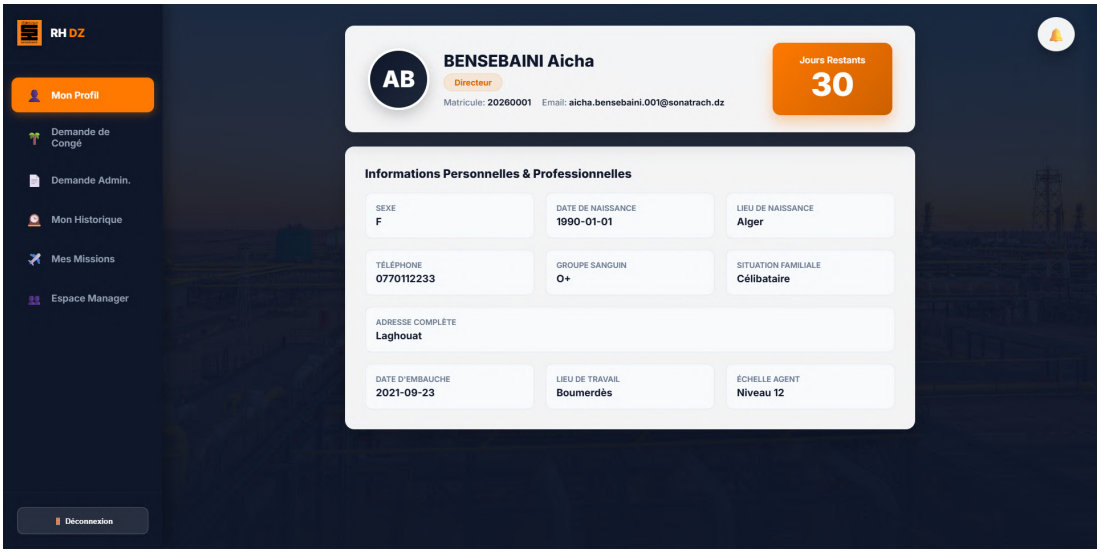


FIGURE 3.3 – Interface du Tableau de bord général et de la cloche d'alertes

L'interface principale offre une vue d'ensemble des services de self-service disponibles et intègre, dans sa barre supérieure, un système d'alerte en temps réel.

Utilité de la cloche de notifications :

— **Centre d'alertes instantané :** Située dans la barre de navigation supérieure, une cloche animée en JavaScript interroge régulièrement l'API (mécanisme de Polling). Elle alerte visuellement l'agent dès qu'un responsable modifie le statut d'une de ses requêtes en cours.

Focus sur le Centre de Notifications RH (Menu Déroulant)

L'interaction avec la cloche de la barre de navigation supérieure déploie un volet interactif listant les derniers événements liés aux dossiers de l'agent.

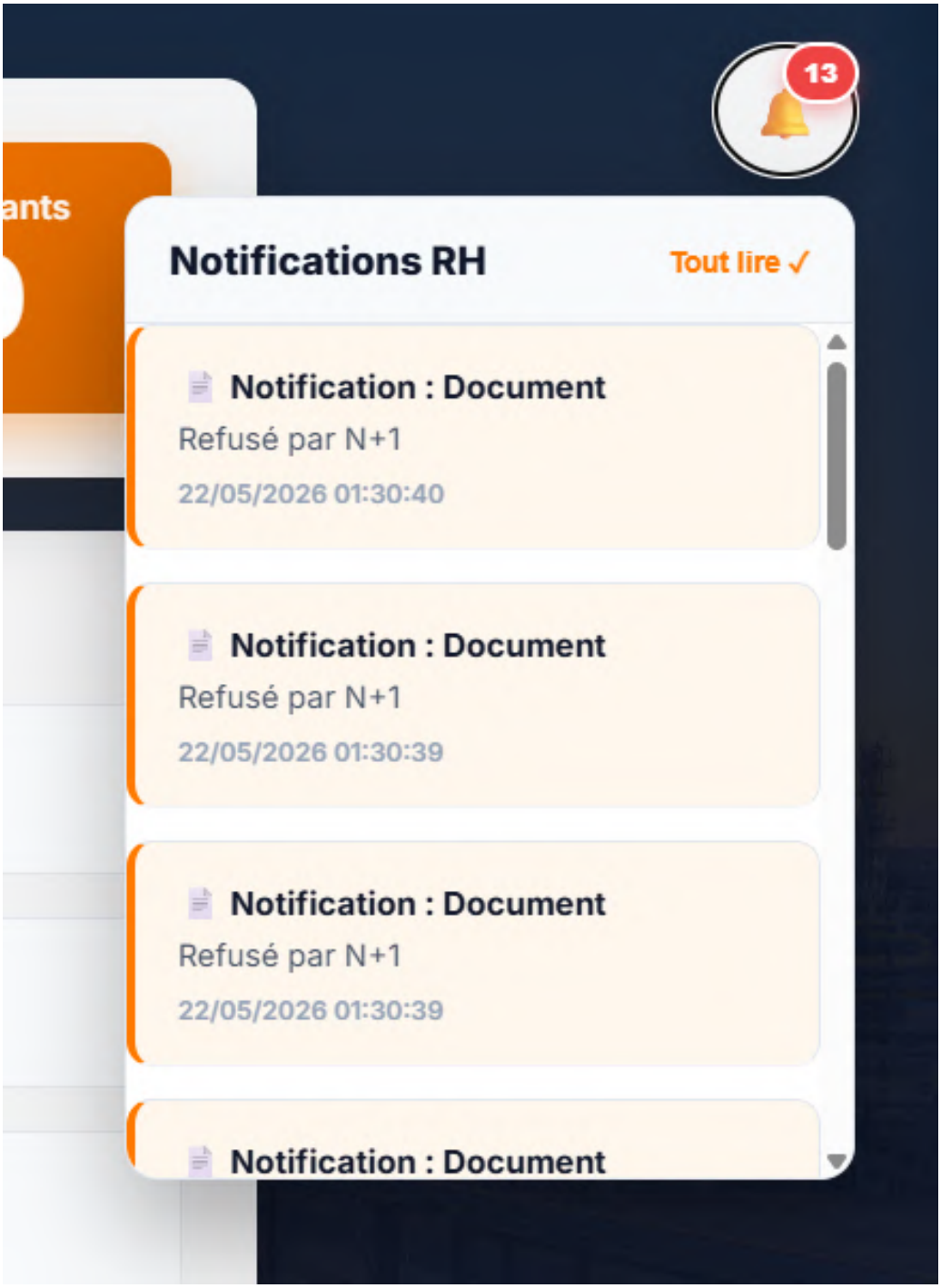


FIGURE 3.4 – Menu déroulant du Centre de notifications RH (Détail UI)

Ce volet interactif liste de façon dynamique l'historique chronologique des événements et des validations de la hiérarchie impactant les demandes de l'agent.

Utilité fonctionnelle et logique technique :

— **Alerte visuelle et contextualisation :** La cloche affiche en permanence une pastille numérique rouge. En cliquant dessus, l'agent déroule un panneau présentant le détail des décisions de la hiérarchie.

— **Mécanisme de Polling JavaScript :** Côté client, un script Vanilla JavaScript s'exécute en arrière-plan et utilise la `Fetch API` pour récupérer les nouvelles statuts dans MySQL.

— **Action de mise à jour (Tout lire) :** L'interface propose un bouton interactif. Son activation fait basculer l'état de toutes les notifications à `lu = 1`, mettant à jour la base de données de manière asynchrone.

3.4.3 Module de gestion des congés (Self-Service)

Le processus de gestion des congés a été entièrement dématérialisé afin de supprimer le circuit d'approbation physique.

Interface de demande de congé (Vue Agent)

L'agent utilise cette interface pour formuler ses demandes d'absence de manière autonome.

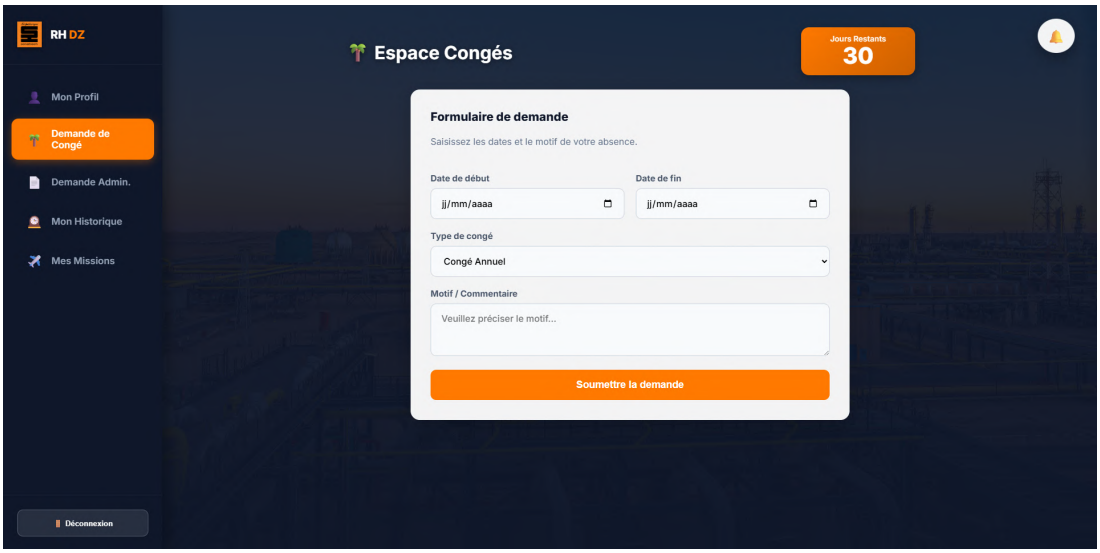


FIGURE 3.5 – Interface de formulation d'une demande de congé (Vue Agent)

Ce formulaire dynamique permet à l'employé de saisir les dates d'absence souhaitées tout en consultant instantanément son solde de jours restants.

Fonctionnalités clés : Le système interroge le Backend pour afficher en temps réel le compteur de jours de congés restants.

Interface d'approbation des congés (Vue Responsable)

Le responsable hiérarchique accède à un espace de validation centralisé pour manager les absences de son équipe.

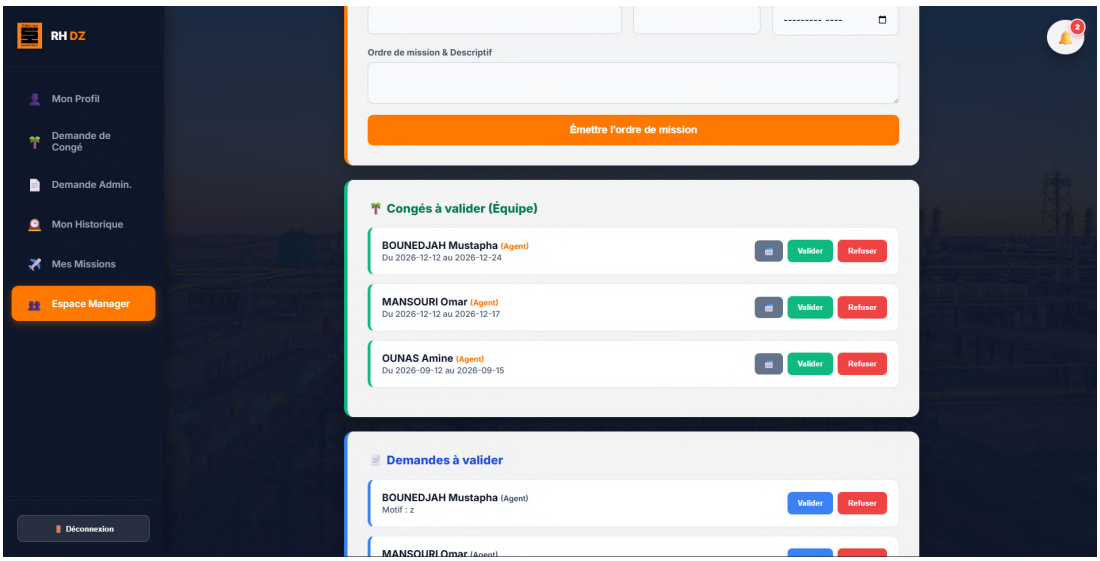


FIGURE 3.6 – Interface d'approbation et de validation des congés (Vue Responsable)

Cet écran offre au manager une vue centralisée des requêtes de son équipe, facilitant la prise de décision rapide grâce aux boutons d'action d'approbation ou de refus.

Fonctionnalités clés : Le manager visualise l'ensemble des demandes en attente de sa structure. Il dispose d'actions directes pour approuver ou refuser la demande.

3.4.4 Module de gestion des ordres de mission

Ce module prend en charge la planification et le suivi des déplacements professionnels.

Interface d'affectation et de création de mission (Vue Responsable)

L'initialisation d'un ordre de mission est orchestrée par le gestionnaire ou le responsable de structure.

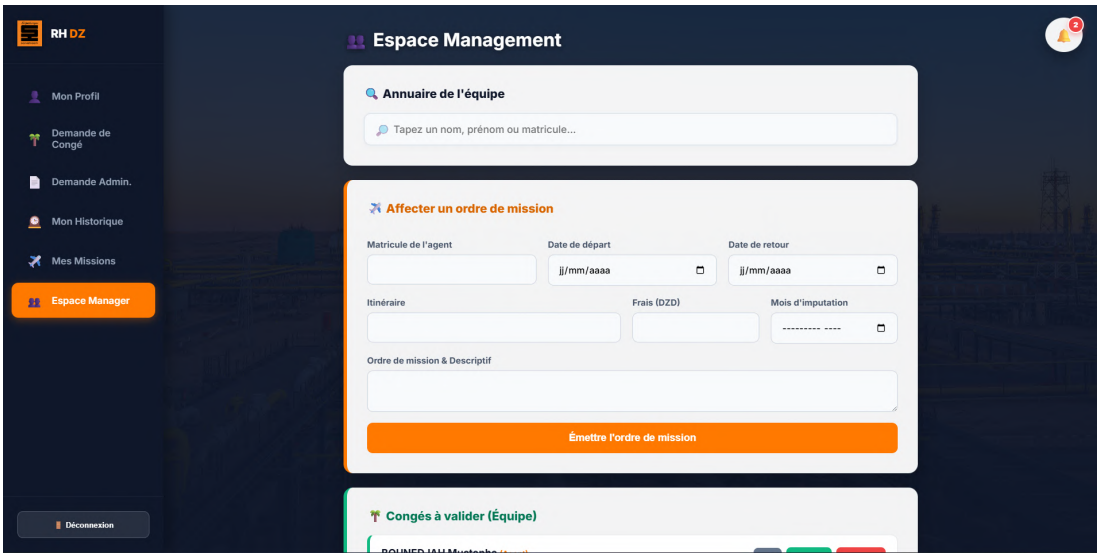


FIGURE 3.7 – Interface de création et d'affectation d'un ordre de mission (Vue Gestionnaire)

Cette interface permet à la hiérarchie de planifier un déplacement en définissant l'itinéraire, les dates précises et le budget alloué à l'agent.

Fonctionnalités clés : Permet de définir l'itinéraire, les dates et d'imputer le montant prévisionnel des frais de mission.

Interface d'affichage et de suivi des missions (Vue Agent)

L'agent dispose d'un accès en lecture seule pour prendre connaissance des détails de ses déplacements validés.

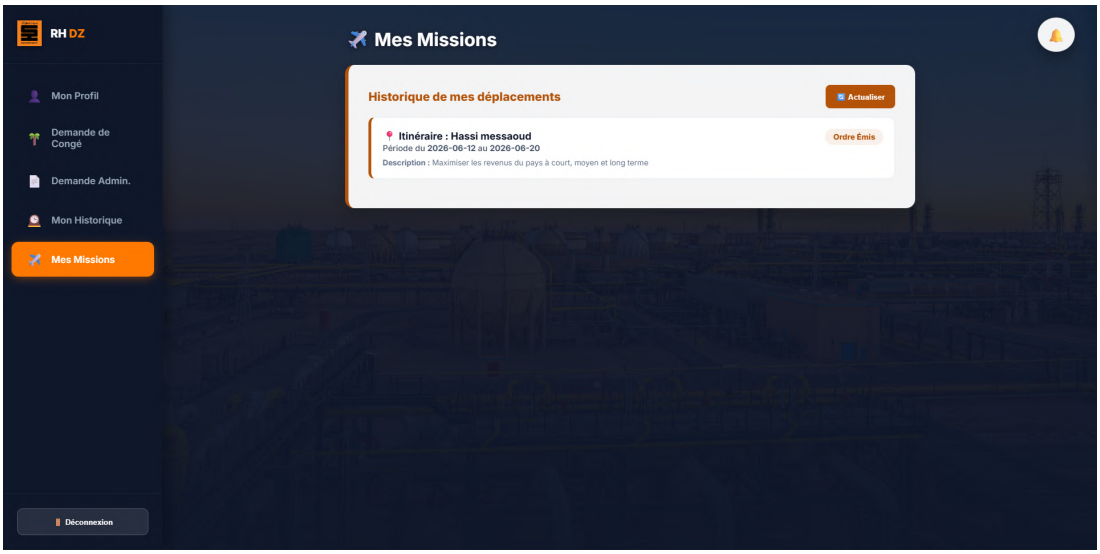


FIGURE 3.8 – Interface d'affichage et de suivi des ordres de mission (Vue Agent)

L'agent peut consulter ici le détail exhaustif des ordres de mission qui lui ont été assignés, accompagné des informations logistiques correspondantes.

3.4.5 Module Boîte de Réception (Espace Documents de l'Agent)

La **Boîte de Réception** constitue un espace de stockage et de réception applicatif structuré pour l'agent.

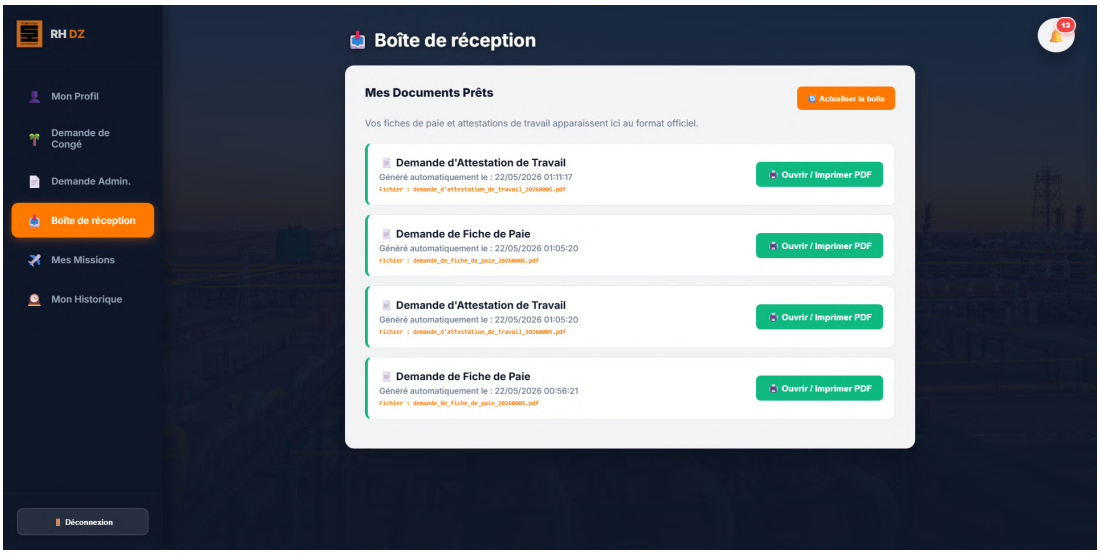


FIGURE 3.9 – Interface de la Boîte de Réception (Vue Agent)

Cet espace personnel agit comme un véritable coffre-fort numérique où l'agent peut retrouver, consulter et télécharger l'ensemble de ses documents administratifs validés.

Utilité fonctionnelle de l'interface : C'est l'emplacement officiel où l'agent retrouve l'historique des documents administratifs validés. Une fois qu'une demande d'attestation est finalisée, le document PDF généré par le serveur y est déposé.

3.4.6 Module de Génération Dynamique de Documents Administratifs

Ce module prend en charge la compilation de fichiers officiels au format PDF via le Backend FastAPI à partir des données extraites de la base de données.

Interface de l'Attestation de Travail (Vue Document Généré)

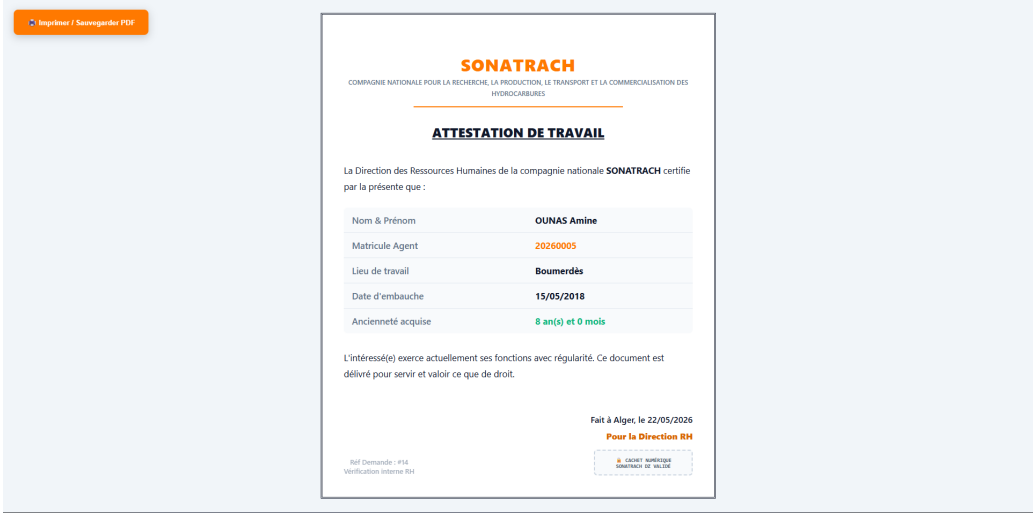


FIGURE 3.10 – Interface de l'Attestation de Travail générée dynamiquement

Ce rendu visuel montre le fichier PDF de l'attestation de travail, généré automatiquement par le serveur avec les données exactes et actualisées de l'agent.

Interface de la Fiche de Paie Électronique (Vue Document Généré)

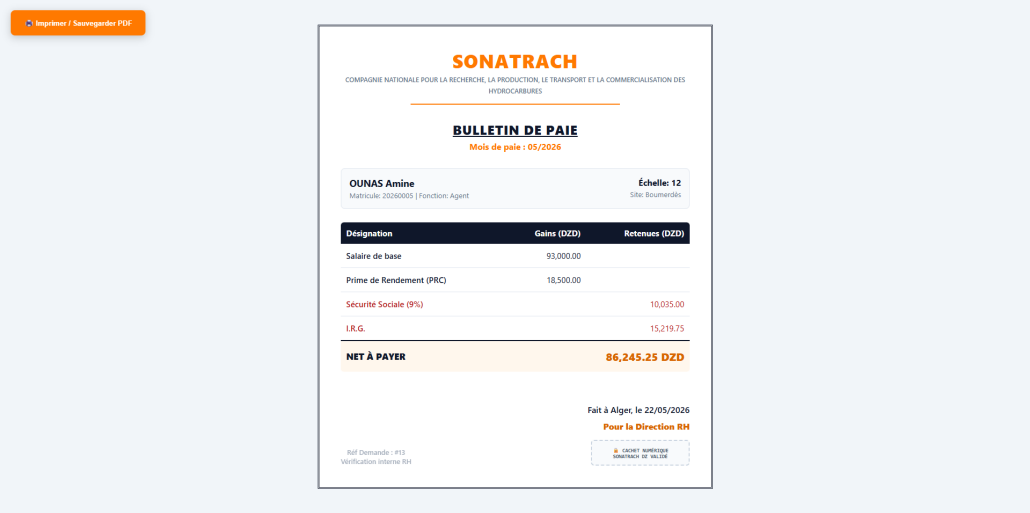


FIGURE 3.11 – Interface de la Fiche de Paie mensuelle générée

Cette capture illustre le bulletin de paie dématérialisé, calculé dynamiquement par le système et prêt à être imprimé ou archivé par l'employé.

Détails techniques du document : Le document arbore le titre clair de *« FICHE DE PAIE »* et est obligatoirement indexé sur un mois d'exercice précis. Le salaire net calculé s'affiche dynamiquement selon l'échelon et la fonction de l'employé.

3.5 Conclusion

En conclusion, ce chapitre a permis de présenter le fruit de notre travail de développement. L'implémentation de cette architecture moderne et la réalisation des interfaces graphiques asynchrones répondent pleinement aux objectifs fixés. Le portail web développé constitue un outil robuste, sécurisé et ergonomique, prêt à être déployé pour faciliter le quotidien administratif des agents de la Division Production.

Conclusion Générale

Ce projet de fin d'études a porté sur la conception et la réalisation d'un portail web dédié aux agents de la Division Production de SONATRACH. Il s'inscrit dans un contexte de modernisation du système d'information des ressources humaines, marqué par les limites de l'application existante RESHUM, principalement orientée vers les gestionnaires RH et peu accessible aux agents. Le portail proposé vise ainsi à compléter le système existant par une solution web moderne, centralisée et sécurisée, permettant aux collaborateurs de consulter leurs informations, de suivre leurs demandes administratives et d'interagir plus efficacement avec leur hiérarchie.

Les principaux objectifs fixés au début du projet ont été atteints. L'étude de l'existant a permis d'identifier les besoins réels des utilisateurs et les insuffisances du fonctionnement actuel. La phase de conception a abouti à une modélisation claire du système à travers les diagrammes UML, le modèle relationnel de données et la définition des principaux acteurs. Sur le plan technique, l'application réalisée repose sur une architecture à trois niveaux, associant une interface web dynamique, une API développée avec FastAPI et une base de données relationnelle MySQL. Les modules de gestion des congés, des missions, des notifications et des documents administratifs générés illustrent concrètement la dématérialisation des processus visés.

Cependant, malgré les résultats obtenus, certaines limitations doivent être soulignées. Le portail réalisé constitue une base fonctionnelle, mais son intégration complète avec l'environnement réel de RESHUM nécessiterait des validations supplémentaires, notamment au niveau de la synchronisation des données, de la sécurité applicative et de la gestion fine des droits d'accès. De plus, certains aspects comme les tests de charge, l'audit de sécurité, la traçabilité complète des actions utilisateurs, la sauvegarde des données et la conformité aux règles internes de l'entreprise n'ont pas été approfondis dans le cadre de ce travail. Ces limites n'enlèvent pas la valeur du projet, mais elles montrent que son passage vers une exploitation réelle demanderait une phase complémentaire de validation et de durcissement.

Comme perspectives d'amélioration, plusieurs évolutions peuvent être envisagées. Le portail pourrait être enrichi par une intégration directe et sécurisée avec RESHUM, par l'ajout d'un système de signature électronique, par une gestion plus avancée des rôles et des permissions, ainsi que par un tableau de bord statistique destiné aux responsables RH. Il serait également pertinent d'améliorer le système de notifications en adoptant des mécanismes temps réel, de renforcer la sécurité par une authentification multifactorielle, et de développer une version responsive ou mobile afin de faciliter l'accès aux agents travaillant sur des sites distants. Ces améliorations permettraient de transformer le portail en une véritable plateforme SIRH évolutive, adaptée aux besoins opérationnels de la Division Production.

Bibliographie et Webographie

Ce document recense de manière exhaustive l'ensemble des ressources académiques, mémoires de recherche, documentations techniques officielles et plateformes numériques ayant alimenté notre démarche d'étude et de développement pour la Division Production de SONATRACH.

Mémoires et Rapports d'études universitaires

- **Zellag, I., Drid, A., & Dridi, A. A. (2014).** *Conception et réalisation d'un portail web pour les ressources humaines.* Rapport de stage d'été, École Nationale Supérieure d'Informatique (ESI), Alger.
- **Remadna, M. R. (2025).** *Conception et réalisation d'une application web pour la gestion des demandes d'attestations de travail.* Rapport de stage, SONATRACH Division Production (DP), Alger, Document N°1663.
- **Kemoum, M., & Deghbar, S. (2020).** *Gestion des Ressources Humaines.* Mémoire de Licence, Université des Sciences et de la Technologie Houari Boumediene (USTHB), Faculté d'Informatique.
- **Saou, S., & Araibia, Z. M. (2022).** *Mise en place d'une plateforme de suivi des permis de travail.* Mémoire de Licence, SONATRACH Division Production (DP).

Documentations de développement et ressources web

- **Site Officiel de SONATRACH.** *Portail institutionnel et présentation des activités.* URL : `https://sonatrach.com/`
- **W3Schools Online Web Tutorials.** *Tutoriels sur les standards web (HTML5, CSS3, JavaScript).* URL : `https://www.w3schools.com/`
- **Lucidchart Documentation.** *Modélisation et création de diagrammes UML.* URL : `https://www.lucidchart.com/`
- **XAMPP Open-Source Distribution.** *Environnement de développement local (Apache, MariaDB).* URL : `https://www.apachefriends.org/`