

ING 2 GSI
Axel Guesdon
Ilyass Chakhmoun
Matthias Galisson
Mathis Lance Richardot
Clément Premoli
Nassym Ben Abdallah



Rapport JEE et Spring Boot:

RH Éléance

2025-2026
Mohamed Haddache

Rapport JEE et Spring Boot:	0
1. Introduction	2
2. Conception	2
2.1. Objectifs fonctionnels	2
2.2. MCD	2
3. Présentation du site	4
3.1 Site Spring Boot:	4
3.2 Site JEE:	6
4. Technologie utilisée et pourquoi ?	7
4.1 Projet Spring Boot:	7
4.2 Projet JEE:	8
5. TEST:	9
6. Organisation du projet:	10

1. Introduction

Ce rapport présente le développement d'une application complète de gestion des ressources humaines réalisée dans le cadre du module JEE, puis adaptée dans une seconde version sous Spring Boot. L'objectif principal du projet est de concevoir et de déployer une application web permettant la gestion centralisée des employés, de leurs informations professionnelles, de leurs fiches de paie ainsi que des différents services et projets auxquels ils sont rattachés.

Le système vise à fournir une plateforme fiable, sécurisée et structurée, intégrant plusieurs modules essentiels : gestion des employés, des départements, des projets, des congés, génération automatisée de documents (PDF et ZIP), statistiques, et authentification avancée basée sur les rôles.

2. Conception

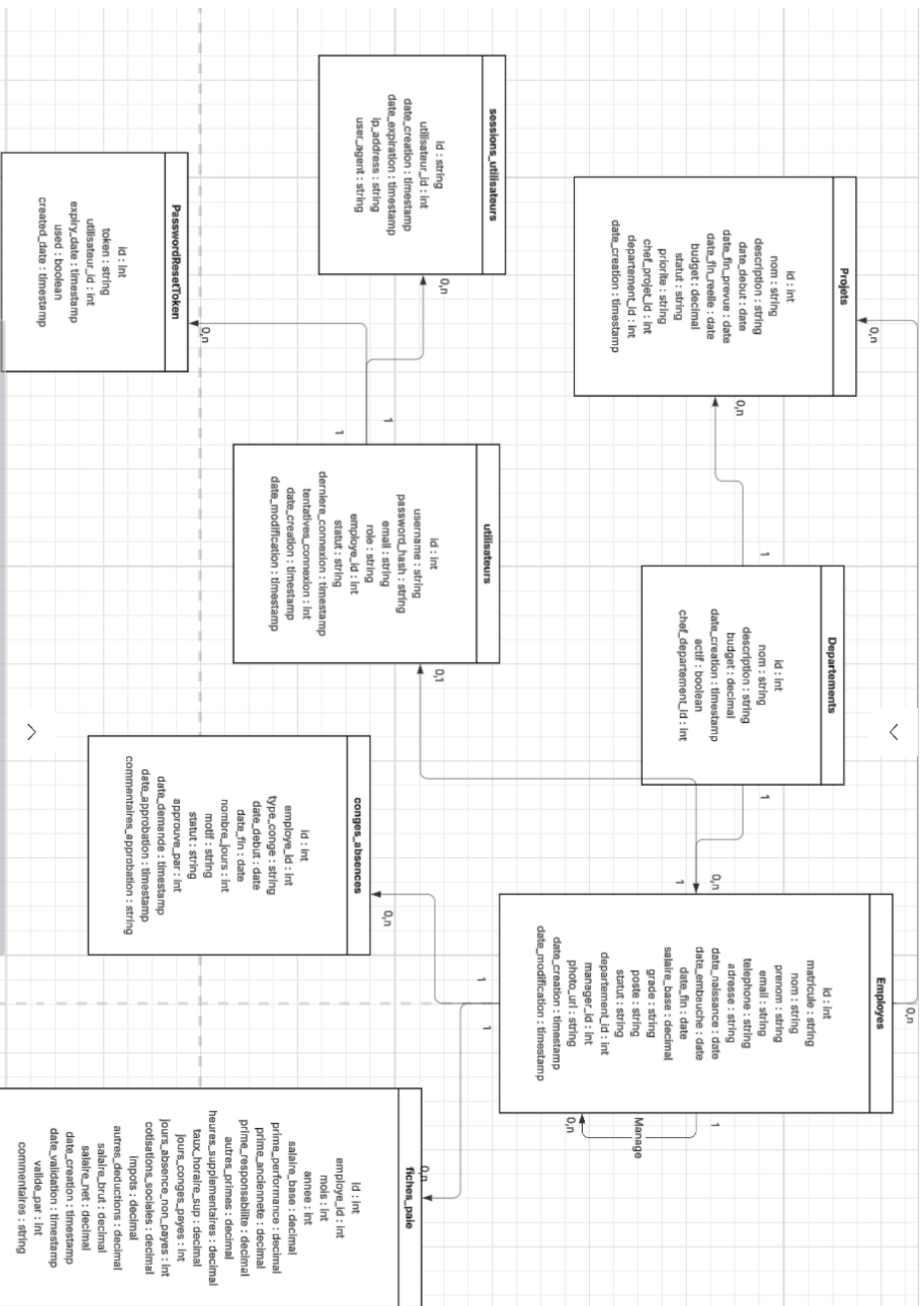
La conception repose sur un MCD permettant d'identifier les différentes entités du système ainsi que leurs relations.

2.1. Objectifs fonctionnels

- - CRUD Employés, Départements, Projets, Congés
- - Affectation Employés ↔ Projets, département
- - Génération et export PDF/ZIP (fiches de paie, identifiants)
- - Statistiques et dashboard
- - Authentification / rôles (contrôle d'accès)

2.2. MCD

D'après l'analyse du code fourni, les principales entités identifiées sont :



3. Présentation du site

3.1 Site Spring Boot:

RH Éléance
SYSTÈME DE GESTION

IDENTIFIANTS DE DÉMO

Admin: admin / admin123
RH: rh / rh123
Manager: manager / manager123

NOM D'UTILISATEUR

Entrez votre nom d'utilisateur

MOT DE PASSE

Entrez votre mot de passe

SE CONNECTER

Mot de passe oublié ?

© 2025 RH Éléance. Tous droits réservés.

RH Éléance
GESTION DES RESSOURCES HUMAINES

Réinitialisation de votre mot de passe

Bonjour,

Nous avons reçu une demande de réinitialisation de mot de passe pour votre compte RH Éléance. Pour créer un nouveau mot de passe, veuillez cliquer sur le bouton ci-dessous :

RÉINITIALISER MON MOT DE PASSE

Lien valide pendant 1 heure
Ce lien expirera automatiquement après 60 minutes pour des raisons de sécurité.

Si le bouton ne fonctionne pas, copiez et collez ce lien dans votre navigateur :

<http://localhost:8080/gestion-rh/reset-password?token=92cccc3f-b588-4979-a29b-ba9e729afde6>

Pour accéder au site, nous avons mis en place un système d'authentification comprenant des profils de démonstration. Chaque employé dispose également d'un identifiant personnel, généré directement depuis l'application. Un système de récupération de mot de passe ainsi qu'une demande de modification sont disponibles, permettant à l'utilisateur de changer son mot de passe sans intervention de l'administrateur et de recevoir un mail pour valider la réinitialisation du mot de passe

Tableau de bord
Vue d'ensemble de votre système de gestion des ressources humaines

admin (ROLE_ADMIN)

Tableau de bord

Employés

Départements

Projets

Fiches de Paie

Congés

Statistiques

Identifiants

Déconnexion

14 EMPLOYÉS

6 DÉPARTEMENTS

4 PROJETS ACTIFS

2 CONGÉS EN ATTENTE

Notifications récentes

Nouvelle demande de congé
Une demande de congé est en attente d'approbation
IL Y A 2H

Projet terminé
Un projet a été marqué comme terminé
HIER

Actions rapides

AJOUTER UN EMPLOYÉ

+ CRÉER UN PROJET

DEMANDER DES CONGÉS

VOIR LES STATISTIQUES

Une fois connecté, l'utilisateur peut accéder aux différents modules de gestion tels que les départements, les projets, les employés, les fiches de paie et les congés. L'accès aux fonctionnalités dépend des droits associés à son rôle. Ainsi, certaines options peuvent être visibles ou masquées selon le niveau d'autorisation accordé.

Les droits et fonctionnalités accessibles dépendent du type de profil connecté. L'administrateur bénéficie d'un ensemble complet de permissions lui permettant de gérer les départements, les projets, les identifiants et la création de nouveaux employés. À l'inverse, les employés ne disposent que d'un accès limité : ils peuvent principalement consulter les informations les concernant, effectuer des demandes de congés ou accéder à leurs fiches de paie. Leur capacité à modifier les données reste fortement restreinte afin de garantir la cohérence et la sécurité des informations du système.

The screenshot displays a dashboard with three summary cards at the top: 'EMPLOYÉS TOTAL' (14), 'Avec compte' (11), and 'SANS COMPTE' (3). Below these, a navigation bar shows 'Sans compte (3)' as the active section. The main content area is titled 'Employés sans compte utilisateur' and includes a 'TOUT GÉNÉRER' button. A table lists three employees with their details and a 'GÉNÉRER' button for each.


MATRICULE	NOM ET PRÉNOM	POSTE	DÉPARTEMENT	EMAIL	ACTIONS
MGR002	Sophie Bernard	Responsable Finance	Finance	sophie.bernard@entreprise.com	GÉNÉRER
DIR002	Marie Dubois	Directrice RH	Ressources Humaines	marie.dubois@entreprise.com	GÉNÉRER
DIR001	Jean Martin	Directeur Général	Informatique	jean.martin@entreprise.com	GÉNÉRER

Un système de gestion des mots de passe a été mis en place afin que l'administrateur puisse générer un mot de passe lors de la création d'un employé. Cette opération produit automatiquement un fichier PDF que l'administrateur doit transmettre à l'employé pour lui permettre d'accéder à son compte.


Nous avons également implémenté la génération de fiches de paie, permettant de produire soit un fichier PDF individuel, soit une archive ZIP contenant plusieurs fiches de paie selon les fichiers demandé

RH ÉLÉGANCE

SYSTÈME DE GESTION




NOM D'UTILISATEUR



MOT DE PASSE

➔

SE CONNECTER



COMPTES DE DÉMONSTRATION

ADMINISTRATEUR

admin / admin123

RESPONSABLE RH

marie.dubois / marie123

CHEF DE PROJET


pierre.leroy / pierre123

EMPLOYÉ STANDARD

claire.durand / claire123

La génération de mots de passe est également disponible dans cette version, permettant à l'administrateur de créer les identifiants des employés. Toutefois, contrairement à la version Spring Boot, il n'existe pas de fonctionnalité permettant aux utilisateurs de demander une modification de leur mot de passe ni d'en recevoir un nouveau par e-mail. La gestion des droits reste néanmoins pleinement opérationnelle et conforme aux spécifications initiales.

Vue d'ensemble de vos notifications récentes

<div><div></div><div>Mes notifications</div></div>	
<hr/>	
<div><div>INFO</div><div>Le projet "Refonte" a été créé par admin.</div><div>Crée le : 30/11/2025 20:30 • Non lue</div></div>	<div>MARKER COMME LUE</div>
<hr/>	
<div><div>INFO</div><div>Le département "Services" a été créé par marie.dubois.</div><div>Crée le : 30/11/2025 20:29 • Non lue</div></div>	<div>MARKER COMME LUE</div>
<hr/>	
<div><div>INFO</div><div>Test notification pour admin</div><div>Crée le : 30/11/2025 20:06 • Lu le : 30/11/2025 20:06</div></div>	
<hr/>	
<div><div>ALERTE</div><div>Le département "jdv b" a été supprimé.</div><div>Crée le : 30/11/2025 19:48 • Lu le : 30/11/2025 19:48</div></div>	
<hr/>	
<div><div>INFO</div><div>Le projet "test" a été modifié par admin.</div><div>Crée le : 30/11/2025 19:47 • Lu le : 30/11/2025 19:47</div></div>	

6

4. Technologie utilisée et pourquoi ?

4.1 Projet Spring Boot:

Java + Spring Boot :

L'utilisation de Java combinée à Spring Boot offre une base robuste et éprouvée pour le développement d'applications d'entreprise. Spring Boot simplifie considérablement la configuration initiale grâce à son principe de "convention over configuration" et fournit un conteneur d'injection de dépendances puissant, facilitant la modularité et la testabilité du code.

Build : Maven (pom.xml)

Maven a été choisi pour la gestion des dépendances et l'automatisation du cycle de vie de construction (build). Sa structure standardisée (via le `pom.xml`) permet une gestion claire des bibliothèques externes et assure la reproductibilité du build sur différents environnements.

BDD : Spring Data JPA (repositories)

Pour la couche de persistance, Spring Data JPA est utilisé afin d'abstraire la complexité des interactions avec la base de données. Il réduit drastiquement le code "boilerplate" nécessaire pour les opérations CRUD et permet une intégration fluide avec Hibernate, tout en offrant des fonctionnalités avancées comme la pagination et le tri automatique.

Templates/UI : Thymeleaf

Thymeleaf est retenu comme moteur de template pour le rendu côté serveur. Il s'intègre naturellement avec l'écosystème Spring et permet de créer des pages HTML dynamiques tout en maintenant un code lisible et maintenable. Sa capacité à fonctionner avec des fragments favorise la réutilisation des composants d'interface.

Sécurité : Spring Security

La sécurité de l'application est assurée par **Spring Security**, le standard de fait pour les applications Spring. Il offre une gestion complète et hautement configurable de l'authentification et des autorisations, protégeant l'application contre les vulnérabilités courantes et permettant une gestion fine des rôles utilisateurs.

Génération PDF : iText (PdfService, utilitaires)

La bibliothèque iText est utilisée pour la génération de documents PDF. Elle offre une API riche et performante pour créer des documents complexes de manière programmatique, répondant ainsi aux besoins d'exportation et de reporting de l'application.

Tests : JUnit / SpringBootTest

La qualité du code est garantie par l'utilisation de JUnit couplé à SpringBootTest. Cette combinaison permet d'écrire des tests unitaires et d'intégration efficaces, assurant le bon fonctionnement des composants isolés ainsi que de l'application dans son ensemble (contexte Spring chargé).

Script BDD MySQL: database_setup_springboot.sql

Le script SQL `database_setup_springboot.sql` est utilisé pour initialiser la structure de la base de données et insérer des données de référence. Il permet de configurer rapidement un environnement de développement ou de production cohérent, en s'assurant que toutes les tables et contraintes nécessaires sont présentes avant le démarrage de l'application.

Spring-boot-starter-mail

L'extension Spring Boot Starter Mail est intégrée pour gérer l'envoi d'emails depuis l'application. Elle simplifie la configuration des serveurs SMTP et fournit une abstraction facile à utiliser pour envoyer des notifications, des confirmations d'inscription ou des réinitialisations de mot de passe, améliorant ainsi l'interactivité avec les utilisateurs.

4.2 Projet JEE:

Java + Jakarta EE (Servlets / JSP)

L'application est basée sur Jakarta EE avec des servlets comme contrôleurs et des JSP pour les vues. Le pattern MVC est géré côté serveur via des servlets et des filtres.

Build : Maven (pom.xml)

Maven gère les dépendances et le cycle de build. Le fichier de configuration principal est pom.xml et inclut des dépendances pour iText, Hibernate, JSTL, JUnit, Mockito, Apache POI, MySQL, etc.

BDD : Hibernate + DAO

La persistance utilise Hibernate avec un utilitaire de configuration et un pattern DAO pour les opérations CRUD. La configuration Hibernate est fournie et les DAO encapsulent l'accès aux entités.

Templates / UI : JSP + JSTL

Les pages sont rendues avec JSP et JSTL. Les vues contiennent les listes, formulaires et fragments d'interface réutilisables.

Sécurité : filtre d'authentification et contrôles côté servlet

L'authentification et l'autorisation sont mises en œuvre via un filtre personnalisé et des vérifications de rôle dans les servlets.

Génération PDF : iText

La génération de PDF est réalisée avec iText depuis des servlets/utilitaires. Le projet supporte l'exportation individuelle de PDF et la création de ZIP contenant plusieurs PDF.

Tests : JUnit / Mockito

La configuration de tests unitaires et d'intégration utilise JUnit et Mockito, déclarés dans pom.xml.

Script BDD MySQL

Un script SQL d'initialisation de la base est fourni pour créer les tables et insérer des données de référence.

5. TEST:

Pour la phase de tests, nous avons utilisé quelques tests JUnit, mais la majorité des vérifications a été réalisée directement sur le site et sur l'ensemble de ses fonctionnalités. L'organisation du travail reposait sur une répartition claire des tâches : une partie du groupe était chargée d'identifier les bugs et anomalies, tandis que l'autre se consacrait à leur correction.

Nous avons mis en place un salon textuel sur Discord afin de centraliser la liste des bugs détectés et de suivre leur résolution au fur et à mesure. Après chaque commit, une nouvelle série de vérifications était effectuée pour prévenir l'apparition de bug ou de nouveaux dysfonctionnements, garantissant ainsi une stabilité progressive du système.

Connexion:

.Rh et manager pas dispo à enlever dans identifiant de démo ou à ajouter MOYEN
.ADMIN coder en dur à enlever

Dans fiche de Paie:

.Salaire négatif si salaire < impôts + Cotisation IMPORTANT
.Filtres Rechercher par nom,département,mois marche pas IMPORTANT

Départements:

.On ne peut pas supprimer un départements (Erreur) IMPORTANT
.Affecter un employé pas possible (bouton non fonctionnel) IMPORTANT
.Nombre employé pas mis à jour MOYEN
.PAS de filtre pour les départements Pas obligatoire

Erreur page:

Bouton retour accueil ne fonctionne pas 404 not found IMPORTANT

Liste employé:

.Recherche pas fonctionnel les filtres ne marche pas IMPORTANT

Création employé:

.enlever photo profil FAIBLE
.Statut: démissionnaire, en congé => erreur dans la modification IMPORTANT
Licencié + retraité pas affiché dans le Statut du Profil FAIBLE
.Grade: Principal + Confirmé => erreur dans la modification IMPORTANT

6. Organisation du projet:

Pour le projet, nous avons commencé par développer la version JEE. Dans un premier temps, nous nous sommes concertés afin d'organiser l'architecture du projet, de définir les fonctionnalités à implémenter et de répartir les tâches entre les membres du groupe. En parallèle de cette répartition, nous avons également structuré notre communication interne.

Un serveur Discord a été créé, comprenant plusieurs salons dédiés : un pour le suivi des tests et des bugs, un autre pour les tâches à réaliser, ainsi qu'un salon destiné à l'organisation générale du projet. Nous avons également mis en place des réunions hebdomadaires afin de faire le point sur l'avancement des travaux et d'assurer un retour d'information constant.

La répartition des tâches initiale a été organisée de la manière suivante :

- **1 personne** : gestion des projets et des départements
- **1 personne** : gestion des fiches de paie et des congés
- **1-2 personnes** : gestion des comptes utilisateurs, des droits d'accès et des droits de modification
- **1-2 personnes** : gestion du système de mots de passe et des employés

Pour les autres fonctionnalités, nous nous sommes concertés régulièrement afin d'établir un plan clair regroupant les éléments restants à développer, ainsi que les fonctionnalités supplémentaires que nous souhaitions intégrer. Progressivement, nous avons ajouté des tests et enrichi le système avec de nouvelles améliorations.

Concernant la version Spring Boot, il n'a pas été nécessaire de revoir l'architecture ni de modifier la répartition des tâches, puisque les bases du projet restaient identiques. Le seul changement notable a été l'arrivée d'une **sixième personne** dans le groupe, arrivée en milieu d'année. Ne pouvant participer au développement en raison de son retard, elle a néanmoins contribué à la phase de tests et à l'analyse du code.