**Table des matières**

[**Table des figures** 2](#_Toc156235892)

[**1.** **Introduction :** 3](#_Toc156235893)

[**2.** **Cadre générale :** 3](#_Toc156235894)

[**2.1.** **Entreprise : Primatec Engineering** 3](#_Toc156235895)

[**2.2.** **Description de projet :** 5](#_Toc156235896)

[2.2.1. Objectifs Principaux : 5](#_Toc156235897)

[2.2.2. Fonctionnalités Clés : 6](#_Toc156235898)

[**3.** **Objectifs et Installation :** 6](#_Toc156235899)

[**3.1.** **Analyse des exigences :** 6](#_Toc156235900)

[**3.2.** **Cahier de Charges : Développement Backend pour le Système de Gestion des Évaluations des Employés** 8](#_Toc156235901)

[3.2.1. Objectifs du Système : 8](#_Toc156235902)

[3.2.2. Fonctionnalités Principales : 9](#_Toc156235903)

[3.2.3. Tests : 10](#_Toc156235904)

[3.2.4. Livrables Attendus : 10](#_Toc156235905)

[**3.3.** **Technologies utilisées :** 10](#_Toc156235906)

[3.3.1. Framework Backend - Spring Boot : 11](#_Toc156235907)

[3.3.2. Base de Données - MySQL : 11](#_Toc156235908)

[3.3.3. Outil de Test API - Postman : 12](#_Toc156235909)

[3.3.4. Documentation API - Open API (Swagger) : 12](#_Toc156235910)

[**4.** **Développement et Structuration :** 12](#_Toc156235911)

[**4.1.** **Conception et architecture :** 12](#_Toc156235912)

[4.1.1. Entités du Système : 13](#_Toc156235913)

[4.1.2. Architecture du Projet : 14](#_Toc156235914)

[**4.2.** **Maquette des interfaces :** 15](#_Toc156235915)

[**4.3.** **Implémentation :** 18](#_Toc156235916)

[4.3.1. Patron de Conception Adapter : 19](#_Toc156235917)

[4.3.2. Patron de Conception Builder : 20](#_Toc156235918)

[**5.** **Conclusion :** 21](#_Toc156235919)

[**Bibliographie :** 22](#_Toc156235920)

# **Table des figures**

[Figure 1: Primatec logo 3](#_Toc156235933)

[Figure 2: Entités du Système 13](#_Toc156235934)

[Figure 3: Architecture générale du Projet 14](#_Toc156235935)

[Figure 4: Relation entre service et controller 14](#_Toc156235936)

[Figure 5: Documentation des API REST 16](#_Toc156235937)

[Figure 6: Transformation des données d'un fichier Excel vers la base de données 16](#_Toc156235938)

[Figure 7: Récupérer les employés 17](#_Toc156235939)

[Figure 8: Récupérer les détails d'un employé spécifique 17](#_Toc156235940)

[Figure 9: Met à jour les informations d'un employé existant 18](#_Toc156235941)

[Figure 10: Patron de Conception Adapter 19](#_Toc156235942)

[Figure 11: Patron de Conception Builder 20](#_Toc156235943)

# **Introduction :**

Au cours de mon stage d'été, en tant qu'étudiant en génie informatique au sein de l'International Institute of Technologies (IIT) de Sfax, j'ai eu l'opportunité stimulante de participer à un projet de grande envergure axé sur la gestion des évaluations des employés. Ce projet ambitieux visait à moderniser et automatiser le processus d'évaluation annuelle, fournissant une solution intégrée pour suivre la performance individuelle, évaluer la satisfaction et la stabilité des employés, et faciliter la gestion des objectifs et de la proactivité au sein de l'organisation. Mon rôle actif dans le développement de ce projet a permis d'appliquer les connaissances acquises au cours de mon cursus d'ingénierie, tout en explorant des concepts avancés et des pratiques de développement logiciel. Cette expérience a été une immersion enrichissante dans le monde professionnel, combinant les défis stimulants du génie informatique avec la mise en pratique concrète des compétences acquises au sein de l'IIT.Top of Form

# **Cadre générale :**

## **Entreprise : Primatec Engineering**



Figure 1: Primatec logo

**Description :** Primatec Engineering, fondée en 2010, s'est imposée comme une entreprise leader dans le domaine de l'externalisation offshore, spécialisée dans le test et le développement de contrôleurs électroniques. Basée à Sfax, en Tunisie, Primatec s'est positionnée comme un acteur clé en fournissant des solutions informatiques sur mesure, innovantes et rentables, spécialement adaptées à l'industrie automobile.

**Compétences Clés :** Primatec Engineering se concentre sur la fourniture de solutions de pointe dans le domaine des tests logiciels et matériels, ainsi que de la validation. Son expertise réside dans sa capacité à répondre à la demande croissante de certaines des entreprises automobiles les plus renommées au monde. Un aspect notable de la compétence de Primatec est son équipe, dont la majorité détient des certifications ISTQB, garantissant ainsi un haut niveau de qualité et de précision dans ses services.

**Présence Mondiale et Partenariats Stratégiques :** Opérant depuis son siège social à Sfax, Primatec a réalisé des progrès significatifs en établissant une empreinte mondiale. À travers des partenariats stratégiques, notamment avec Technica Engineering GmbH à Munich, Primatec a forgé des relations solides avec des géants automobiles allemands et internationaux tels que BMW, MAN, Lear Corporation, Valeo, entre autres. Cette collaboration a non seulement renforcé les capacités techniques de Primatec, mais a également permis à l'entreprise de contribuer aux avancées technologiques de l'industrie automobile.

**Répondre de Manière Responsable aux Exigences de l'Industrie :** Primatec Engineering répond activement aux besoins évolutifs du secteur automobile en fournissant des solutions économiquement viables, innovantes et efficaces pour les tests et la validation logiciels et matériels. La société est consciente de la demande croissante de processus de test robustes et, à travers son équipe dévouée, s'efforce de répondre à ces exigences de manière responsable et durable.

**Le Capital Humain comme Force Motrice :** Primatec attribue son succès à ses ressources humaines motivées et dévouées, les considérant comme le plus grand atout. La société met fortement l'accent sur le développement des talents, favorisant un environnement de travail qui encourage l'innovation, la collaboration et l'apprentissage continu. En investissant dans ses collaborateurs, Primatec veille à rester à la pointe des avancées technologiques et à maintenir les normes élevées attendues par ses clients.

En résumé, Primatec Engineering se démarque comme un pionnier de l'externalisation offshore, apportant une contribution significative au paysage technologique de l'industrie automobile. Grâce à une combinaison de savoir-faire technique, de partenariats stratégiques et d'un engagement envers ses ressources humaines, Primatec continue de jouer un rôle essentiel pour relever les défis dynamiques des tests et du développement de contrôleurs électroniques.

## **Description de projet :**

Le projet de gestion des évaluations des employés a été conçu dans le but de rationaliser et d'améliorer le processus d'évaluation des performances au sein de l'organisation. En réponse à la nécessité croissante de gestion efficace des ressources humaines, ce système vise à centraliser et à automatiser la collecte, l'analyse et la présentation des évaluations périodiques des employés.

### Objectifs Principaux :

**Centralisation des Évaluations :** Le projet vise à centraliser toutes les données relatives aux évaluations des employés, couvrant divers aspects tels que la performance, la stabilité, la satisfaction et les compétences techniques.

**Automatisation des Processus :** L'automatisation des processus d'évaluation permet de réduire les tâches administratives manuelles, améliorant ainsi l'efficacité et réduisant les erreurs potentielles.

**Facilitation des Prises de Décision :** En consolidant les évaluations dans un système unique, les gestionnaires ont accès à des données centralisées pour prendre des décisions éclairées sur le développement professionnel et les ajustements au sein de l'équipe.

**Amélioration de la Transparence :** Le système vise à accroître la transparence en fournissant aux employés un accès facile à leurs évaluations et en facilitant la communication entre les gestionnaires et les membres de l'équipe.

**Suivi Continu des Performances :** La mise en œuvre du projet permettra un suivi continu des performances des employés au fil du temps, fournissant une base solide pour les évaluations annuelles.

### Fonctionnalités Clés :

**Gestion des Employés :** Enregistrement et suivi des détails individuels de chaque employé, y compris les données personnelles, le département et la position occupée.

**Évaluations Techniques :** Collecte et analyse des compétences techniques des employés pour orienter les plans de développement professionnel.

**Évaluations Annuelles :** Un module dédié pour les évaluations annuelles, regroupant les performances sur une période donnée avec des commentaires et des suggestions.

**Satisfaction et Stabilité :** Suivi de la satisfaction et de la stabilité des employés au sein de l'organisation pour identifier les tendances et prendre des mesures proactives.

**Automatisation des Rapports :** Génération automatisée de rapports personnalisés pour une analyse approfondie des données d'évaluation.

Ce projet est conçu pour avoir un impact significatif sur la gestion des ressources humaines en créant un système robuste et intégré. Les avantages anticipés comprennent une prise de décision plus éclairée, une gestion proactive des performances, une amélioration de la communication entre les équipes et une facilité d'accès aux évaluations pour les employés.

# **Objectifs et Installation :**

## **Analyse des exigences :**

***Collecte Complète des Informations Employé :***

L'exigence fondamentale du projet était de collecter de manière exhaustive les informations relatives à chaque employé. Cela inclut les détails personnels, les informations de carrière, les évaluations techniques, et d'autres éléments essentiels pour avoir une vue holistique de chaque membre de l'équipe.

***Automatisation de l'Évaluation Annuelle :***

L'entreprise aspirait à automatiser le processus d'évaluation annuelle des employés. Cela impliquait la création d'une fonctionnalité permettant aux gestionnaires de saisir et d'évaluer les performances de chaque employé, avec la possibilité de fournir des commentaires détaillés.

***Gestion des Objectifs et de la Proactivité :***

Un aspect clé du projet était de mettre en place un mécanisme pour gérer les objectifs individuels des employés et de mesurer leur proactivité. Cela devait faciliter le suivi des objectifs assignés et encourager une culture proactive au sein de l'entreprise.

***Suivi de la Satisfaction et de la Stabilité :***

L'entreprise visait à évaluer la satisfaction et la stabilité de chaque employé au sein de l'organisation. Des mécanismes ont été intégrés pour recueillir des feedbacks, mesurer le niveau de satisfaction, et évaluer la stabilité afin de favoriser un environnement de travail positif et stable.

***Centralisation des Données pour une Vue Globale :***

Une exigence majeure était la centralisation des données. Le projet devait fournir une plateforme unique où toutes les données liées aux employés étaient regroupées, facilitant ainsi une vue globale des performances individuelles et de l'ensemble de l'équipe.

***Utilisation de Patrons de Conception pour la Maintenabilité :***

La considération des patrons de conception, tels que l'Adapter et le Builder, était fondamentale dans le processus de développement pour garantir la maintenabilité du code. Ces choix de conception ont permis une séparation claire des préoccupations, facilitant ainsi l'extension et l'évolution future du système.

***Facilité d'Importation des Données :***

Une fonctionnalité a été développée pour faciliter l'importation des données à partir de fichiers Excel, assurant ainsi une transition fluide vers le nouveau système sans nécessiter une saisie manuelle laborieuse.

***Documentation Complète des API :***

Pour assurer la facilité d'utilisation et de compréhension, une exigence était de documenter de manière complète toutes les API exposées par le système. L'intégration de Open API (Swagger) a été mise en place pour garantir une documentation précise et accessible.

L'ensemble de ces exigences a été méticuleusement pris en compte et mis en œuvre dans le projet afin de répondre aux ambitions de l'entreprise pour une gestion efficace des évaluations des employés, favorisant une culture d'amélioration continue et de bien-être au sein de l'organisation.

## **Cahier de Charges : Développement Backend pour le Système de Gestion des Évaluations des Employés**

Le présent cahier des charges vise à définir en détail les spécifications et les exigences pour le développement du backend d'un site web destiné à consolider et présenter les statistiques issues des rapports d'évaluations des employés. Cette solution sera développée en utilisant le framework Spring Boot, avec MySQL comme base de données, Postman pour les tests API, et Open API (Swagger) pour la documentation.

### Objectifs du Système :

Le système a pour objectif de rationaliser la gestion des évaluations des employés en transformant les données statistiques actuellement stockées sous forme de fichiers Excel en base données relationnels après en une interface web conviviale et dynamique. Les fonctionnalités clés incluent la gestion des employés, l'importation des rapports d'évaluations, le calcul automatique des statistiques, l'exposition d'APIs REST pour le frontend, et la documentation détaillée de ces APIs.

### Fonctionnalités Principales :

**Gestion des Employés :**

Ajout d’Employés : Les administrateurs auront la capacité d'ajouter de nouveaux employés via une interface dédiée, saisissant des informations telles que le nom, le prénom, le département, etc.

Consultation des Employés : Des APIs seront mises à disposition pour permettre la récupération des détails de chaque employé, facilitant ainsi l'affichage sur le frontend.

**Importation des Évaluations :**

L'application supportera l'importation des rapports d'évaluations au format Excel. Les fichiers Excel seront analysés, les données seront extraites et stockées dans la base de données MySQL.

**Mise à jour des Statistiques :**

Les statistiques seront mises à jour automatiquement à chaque ajout ou modification de rapport d'évaluation, assurant ainsi des informations en temps réel.

**API pour le Frontend :**

Un ensemble complet d'APIs REST sera exposé pour permettre au frontend d'accéder aux données. Ces APIs seront documentées via Open API (Swagger).

**Documentation :**

L'intégration de Open API (Swagger) permettra une documentation claire et détaillée des différentes API disponibles. Les descriptions, les paramètres et les réponses seront soigneusement documentés.

### Tests :

**Tests Unitaires :** Des tests unitaires seront développés pour garantir que chaque composant du backend fonctionne comme prévu.

**Tests d'Intégration :** Des tests d'intégration seront réalisés pour s'assurer de la cohérence entre les différentes parties du système. Les interactions entre le backend et la base de données seront particulièrement testées.

### Livrables Attendus :

1. Code source du backend développé en utilisant Spring Boot.

2. Documentation exhaustive des API à l'aide de Open API (Swagger).

3. Rapport de tests comprenant des tests unitaires et d'intégration.

4. Guide d'installation et de déploiement détaillé, couvrant l'environnement de développement et de production.

## **Technologies utilisées :**

Le développement du backend pour le système de gestion des évaluations des employés implique l'utilisation de technologies modernes et robustes pour garantir la performance, la sécurité et la flexibilité du système. Voici une description détaillée des technologies clés qui seront employées dans ce projet :

### Framework Backend - Spring Boot :

*Spring Boot :* est un Framework Java basé sur Spring, conçu spécifiquement pour simplifier le développement des applications Java. Ses caractéristiques clés incluent l'inversion de contrôle, la gestion des dépendances, et l'abstraction de la configuration, ce qui permet aux développeurs de se concentrer davantage sur la logique métier sans être trop préoccupés par les détails techniques.

*Inversion de Contrôle (IoC) :* Spring Boot utilise l'IoC pour faciliter la gestion des composants et la configuration, permettant une structure de code plus claire et plus modulaire.

*Gestion des Dépendances :* Avec Maven, Spring Boot simplifie la gestion des dépendances, facilitant ainsi le processus de développement et de maintenance.

### Base de Données - MySQL :

*MySQL :* est un système de gestion de base de données relationnelle open-source qui offre une performance élevée, une stabilité et une facilité d'utilisation. Il est bien adapté pour les applications nécessitant une gestion efficace des données.

*Structure Relationnelle :* MySQL utilise une structure relationnelle qui est bien adaptée à la gestion de données tabulaires, ce qui est essentiel pour stocker les informations des employés et des évaluations.

*Open Source :* En tant que solution open-source, MySQL offre un coût total de possession réduit et une communauté active.

### Outil de Test API - Postman :

*Postman :* est un outil de collaboration pour le développement d'API, permettant aux développeurs de tester, déboguer et documenter les API de manière efficace.

*Collaboration :* Il facilite la collaboration entre les membres de l'équipe de développement en partageant facilement des collections d'API et des environnements.

*Interface Conviviale :* L'interface utilisateur de Postman est conviviale, facilitant la création, la modification et le test d'API.

### Documentation API - Open API (Swagger) :

*Swagger/Open API* : est un ensemble de spécifications open-source pour la documentation des API RESTful.

*Documentation Automatisée :* Swagger/Open API génère automatiquement la documentation des API à partir du code source, offrant ainsi une documentation précise et à jour.

*Interface Interactive :* Les utilisateurs peuvent interagir avec l'API directement à partir de la documentation, simplifiant le processus de test.

*Standardisé :* En tant que norme de facto, Swagger/Open API assure la cohérence et la compatibilité avec d'autres outils et bibliothèques.

# **Développement et Structuration :**

## **Conception et architecture :**

### Entités du Système :

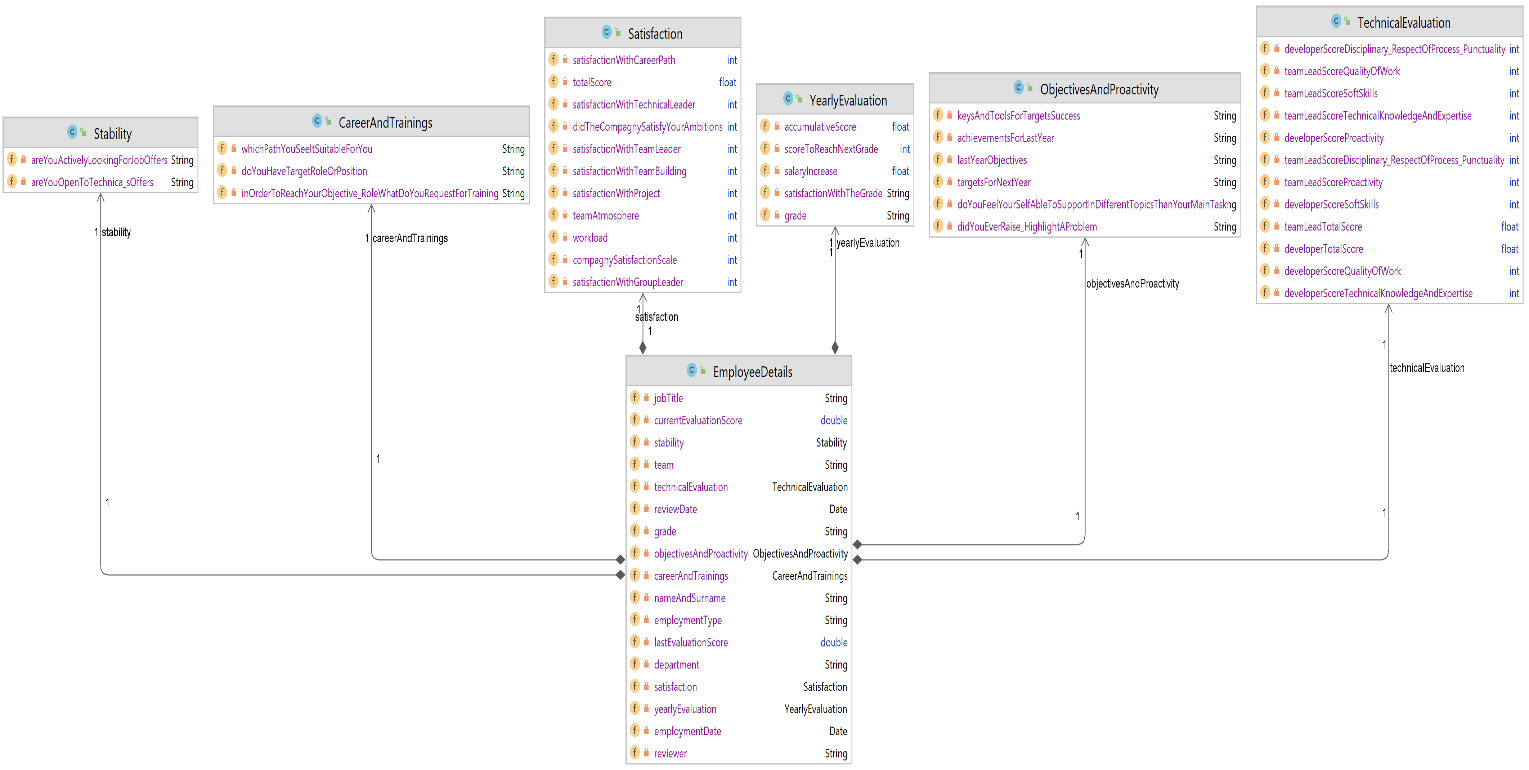
****

Figure 2: Entités du Système

**EmployeeDetail :** L'entité `EmployeeDetail` contient des informations détaillées sur un employé.

**CareerAndTrainings :** L'entité `CareerAndTrainings` représente les détails de la carrière et des formations d'un employé.

**ObjectivesAndProactivity :** L'entité `ObjectivesAndProactivity` représente les objectifs et la proactivité d'un employé.

**Satisfaction :** L'entité `Satisfaction` représente les données sur la satisfaction d'un employé.

**Stability :** L'entité `Stability` représente la stabilité d'un employé au sein de l'organisation.

**TechnicalEvaluation :** L'entité `TechnicalEvaluation` contient les résultats de l'évaluation technique d'un employé.

**YearlyEvaluation :** L'entité `YearlyEvaluation` représente l'évaluation annuelle d'un employé.

### Architecture du Projet :

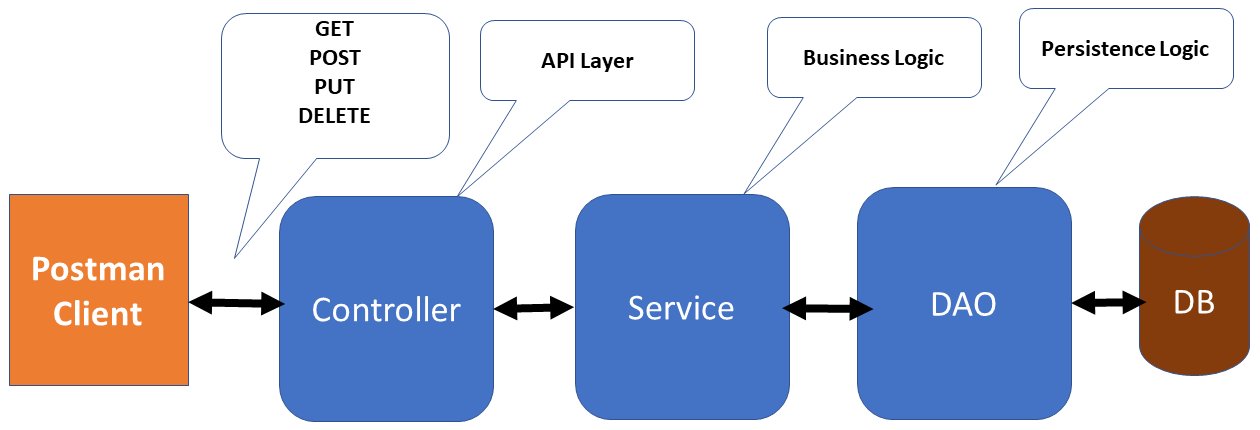
****

Figure 3: Architecture générale du Projet

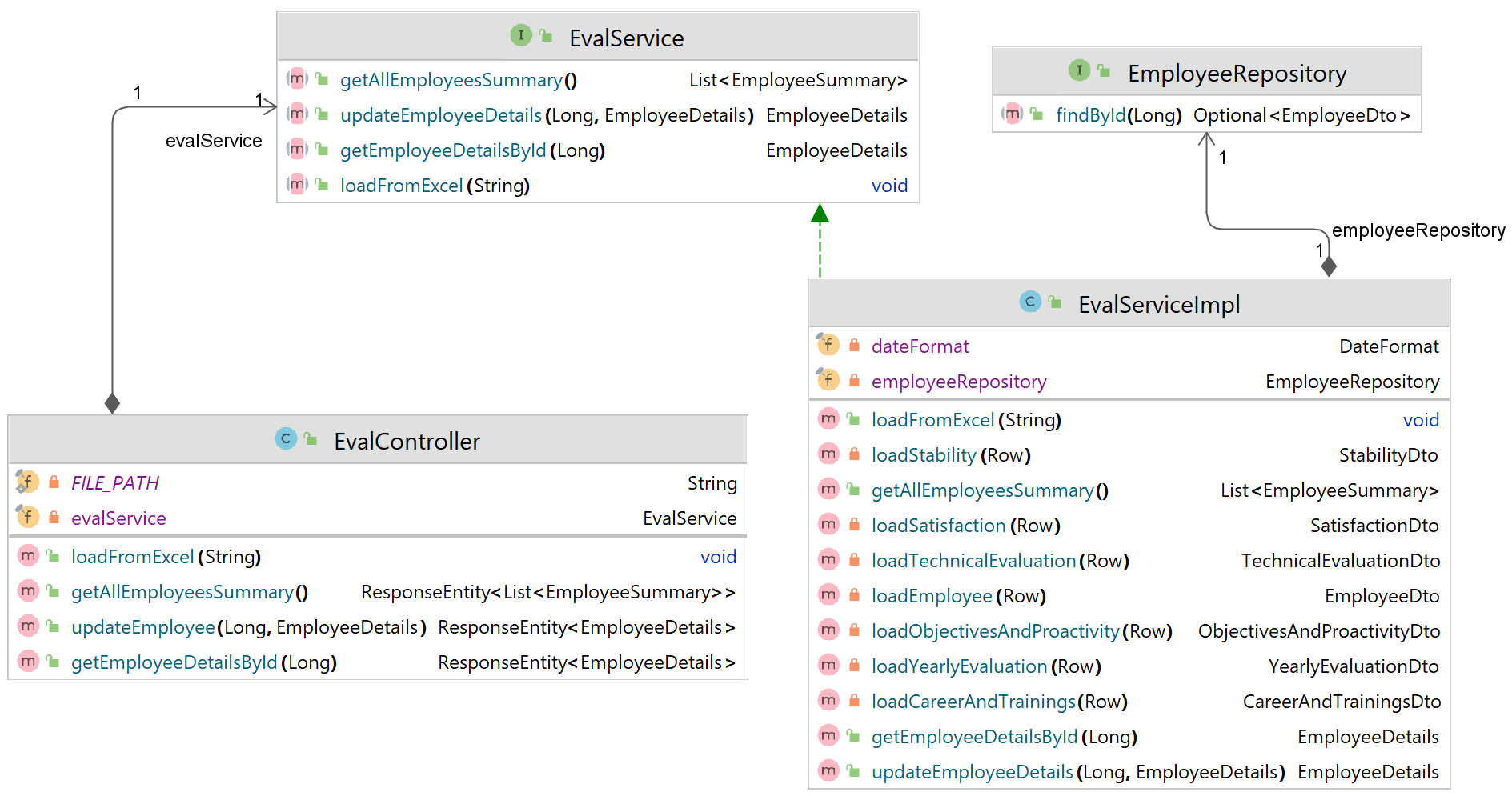
****

Figure 4: Relation entre service et controller

**Modèle Relationnel de Données :** Le modèle relationnel sera conçu en utilisant des tables pour chaque entité avec des clés étrangères reliant les entités associées.

**Architecture Backend - Spring Boot :** L'architecture backend suivra un modèle MVC (Modèle-Vue-Contrôleur) avec Spring Boot.

- Contrôleurs : Géreront les requêtes HTTP, invoqueront les services, et renverront les réponses appropriées.

- Services : Implémenteront la logique métier en interagissant avec les repositories.

- Repositories : Géreront l'accès à la base de données et effectueront les opérations CRUD.

- Modèles (Entités) : Représenteront les objets métier et seront mappés aux tables de la base de données.

**Communication avec la Base de Données – MySQL :** Spring Data JPA sera utilisé pour interagir avec la base de données MySQL. Les entités seront annotées avec des annotations JPA pour définir la relation entre les objets Java et les tables MySQL.

**Documentation et Tests :** Open API (Swagger) sera intégrée pour documenter les API REST, offrant ainsi une référence claire et compréhensible pour les développeurs. Les tests unitaires seront mis en place avec JUnit pour garantir la fiabilité des différentes couches du backend.

## **Maquette des interfaces :**

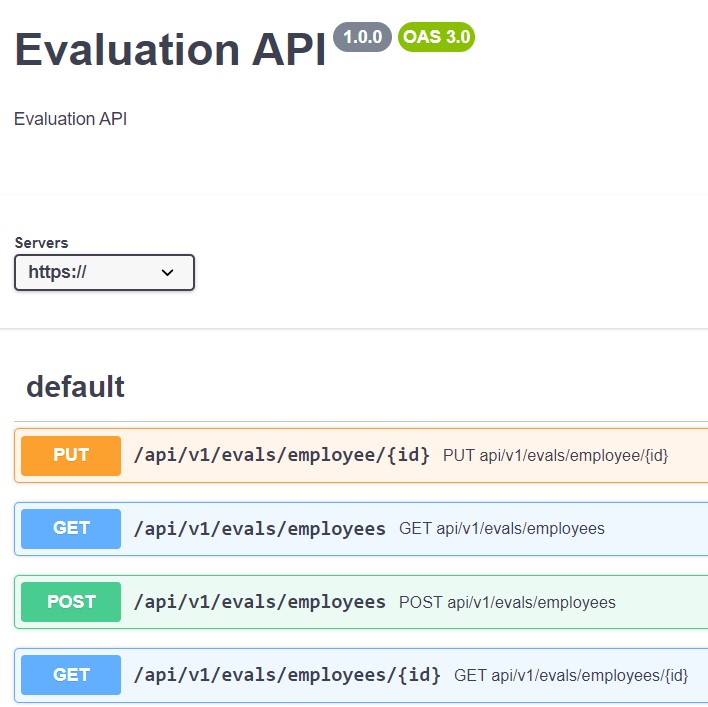


Figure 5: Documentation des API REST

**A screenshot of a computer

Description automatically generated**

Figure 6: Transformation des données d'un fichier Excel vers la base de données

**A screenshot of a computer

Description automatically generated**

Figure 7: Récupérer les employés

**A screenshot of a computer

Description automatically generated**

Figure 8: Récupérer les détails d'un employé spécifique

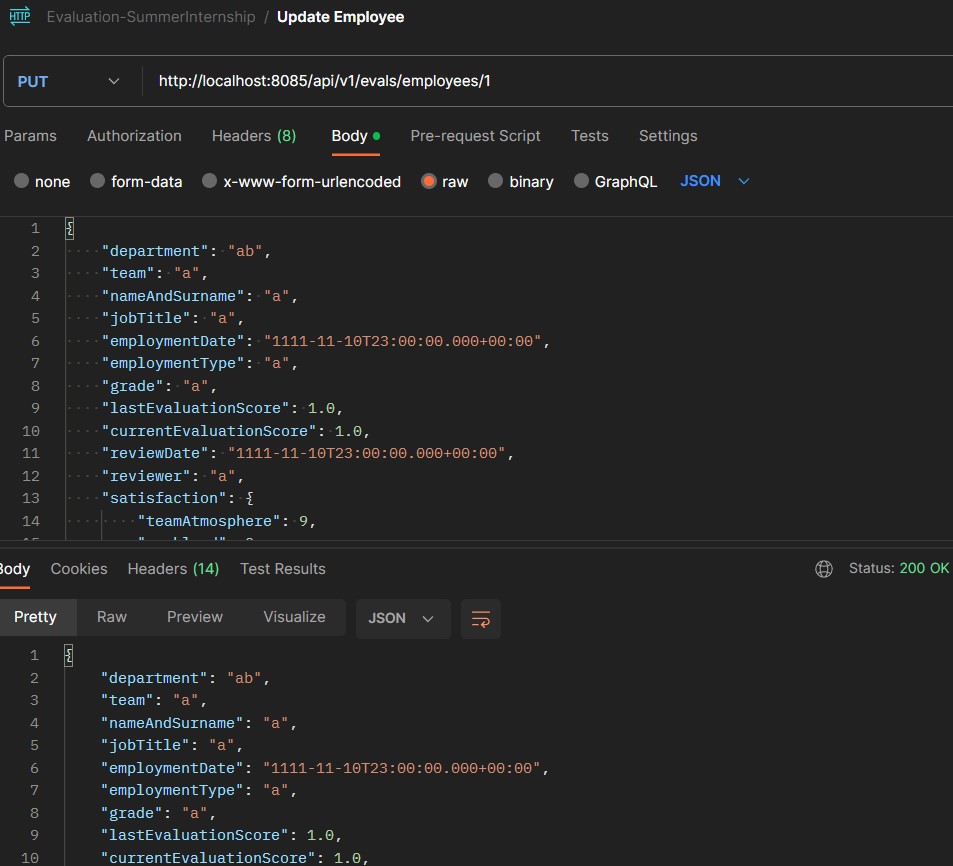
****

Figure 9: Met à jour les informations d'un employé existant

Ces captures d'écran et exemples d'interface illustrent l'utilisation des « endpoints » API pour les opérations de création, mise à jour et récupération des données clés liées aux employés dans le système.

## **Implémentation :**

La phase d'implémentation du projet de gestion des évaluations des employés a été guidée par une approche systématique, mettant en œuvre des bonnes pratiques de développement logiciel. Dans cette section, nous détaillerons l'implémentation en mettant en avant l'utilisation des patrons de conception Adapter et Builder.

### Patron de Conception Adapter :

****

Figure 10: Patron de Conception Adapter

**Objectif :** Le patron de conception Adapter a été intégré pour simplifier et découpler la logique de transformation entre les entités du système et les DTOs (Data Transfer Objects). L'objectif était de rendre le code plus modulaire, facile à comprendre, et d'isoler la logique de transformation des données de la logique métier principale.

**Mise en Œuvre :** L'Adapter a été utilisé pour créer des classes intermédiaires responsables de la conversion entre les entités et les DTOs. Ces adaptateurs ont permis de maintenir une séparation claire entre la structure des données utilisées dans la logique métier et celles destinées à être transférées entre les couches du système.

**Avantages :**

- Isolation de la Logique de Transformation : L'utilisation de l'Adapter a permis d'isoler la logique de transformation dans des classes dédiées, évitant ainsi d'encombrer la logique métier principale.

- Facilité de Maintenance : En centralisant la logique de transformation, les adaptations futures des entités ou des DTOs peuvent être effectuées sans perturber la logique métier existante.

- Clarté du Code : Le code est devenu plus clair et plus lisible, avec des adaptateurs dédiés responsables de la transformation spécifique entre les entités et les DTOs.

### Patron de Conception Builder :

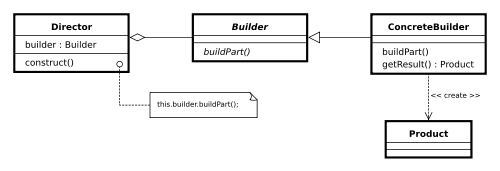
****

Figure 11: Patron de Conception Builder

**Objectif :** Le patron de conception Builder a été choisi pour construire des objets complexes étape par étape, offrant une flexibilité accrue dans la création d'objets avec diverses configurations possibles. L'objectif était de simplifier la création d'objets tout en maintenant une séparation nette entre la logique de construction et la logique métier.

**Mise en Œuvre :** Le Builder a été implémenté pour les objets complexes, tels que les entités représentant les détails de carrière, les évaluations techniques, etc. Chaque Builder était responsable de créer un type spécifique d'objet en suivant des étapes définies.

**Avantages :**

- Flexibilité dans la Construction : Le Builder a permis de créer des objets avec différentes configurations sans surcharger les constructeurs, offrant une flexibilité précieuse.

- Logique de Construction Séparée : La logique de construction était clairement séparée de la logique métier principale, simplifiant la gestion des configurations complexes.

- Facilité d'Extension : L'introduction de nouveaux paramètres ou configurations dans les objets était facile à gérer grâce au Builder, évitant la surcharge des constructeurs.

L'implémentation du projet de gestion des évaluations des employés a été guidée par une approche méthodique, intégrant les patrons de conception Adapter et Builder pour assurer une structure de code solide et modulaire. L'utilisation de l'Adapter a simplifié la transformation des entités en DTOs, tandis que le Builder a facilité la création d'objets complexes avec une variété de configurations possibles. Ces choix de conception ont contribué à un code plus propre, plus maintenable et ont préparé le terrain pour une évolution future du système avec une meilleure extensibilité.

# **Conclusion :**

En concluant ce stage passionnant au sein du Primatec engineering, le projet de gestion des évaluations des employés représente une étape significative dans mon parcours d'ingénieur en génie informatique. L'implémentation réussie de cette solution a permis d'atteindre plusieurs objectifs majeurs. Nous avons établi une plateforme centralisée qui capture de manière exhaustive les détails des employés, automatisé le processus d'évaluation annuelle, et fourni des mécanismes pour gérer les objectifs, la proactivité, la satisfaction, et la stabilité des employés. L'utilisation judicieuse de patrons de conception tels que l'Adapter et le Builder a renforcé la maintenabilité du code, tandis que la documentation exhaustive des API a facilité l'interopérabilité et la compréhension du système. Ce projet a été une opportunité riche d'apprentissage et de mise en pratique, combinant les compétences théoriques avec une expérience pratique dans le développement de solutions informatiques complexes. Il a également jeté les bases d'une culture d'amélioration continue au sein de l'entreprise, alignée sur les ambitions de l'IIT en matière d'excellence et d'innovation. Cette expérience de stage a consolidé ma compréhension des processus de développement logiciel, élargi mes compétences techniques, et m'a permis de contribuer activement à la réalisation d'objectifs organisationnels significatifs.

# **Bibliographie :**

<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>

<https://springjava.com/spring-boot/use-lombok-in-spring-boot>

<https://www.codeflow.site/fr/article/spring-vs-spring-boot>

<https://stackoverflow.com>

<https://www.geeksforgeeks.org/difference-between-json-and-xml/>

<https://fr.wikipedia.org/wiki/Portail:Accueil>

<https://medium.com>

<https://www.w3schools.com>