

Documentation du Projet Gestion du Personnel EFREI

Ce document fournit une vue d'ensemble complète du projet de gestion du personnel de l'EFREI, une application développée en Java selon l'architecture MVC. Il détaille la structure du projet, ses fonctionnalités principales, la configuration technique requise, ainsi que les procédures d'installation et de déploiement. Le document aborde également les aspects de sécurité, de validation des données, et de maintenance, offrant ainsi un guide complet pour les développeurs et les équipes techniques de l'EFREI.

Présentation Générale du Projet

Le projet de gestion du personnel de l'EFREI est une application conçue pour simplifier et centraliser la gestion des étudiants et des promotions de l'école. Développée en Java, l'application adopte une architecture MVC (Modèle-Vue-Contrôleur), ce qui favorise une séparation claire des préoccupations et facilite la maintenance et l'évolution du code. L'objectif principal de ce projet est de fournir une plateforme intuitive et efficace pour le personnel administratif de l'EFREI, leur permettant de gérer les informations relatives aux étudiants, aux promotions, et aux utilisateurs du système.

Cette application offre une gamme complète de fonctionnalités, allant de l'inscription des étudiants à la gestion des promotions, en passant par l'attribution des salles et des diplômes. Elle intègre également un système de recherche avancé, permettant de localiser rapidement les informations spécifiques. L'architecture MVC assure une flexibilité et une modularité optimales, facilitant l'ajout de nouvelles fonctionnalités et l'adaptation aux besoins évolutifs de l'EFREI. En résumé, ce projet vise à améliorer l'efficacité et la précision de la gestion du personnel de l'EFREI, tout en offrant une expérience utilisateur conviviale et intuitive.

La date de dernière mise à jour de cette documentation est le 13 Mars 2025. Cette date indique que le document est à jour et reflète l'état actuel du projet. Il est recommandé de consulter régulièrement cette documentation pour rester informé des dernières modifications et des nouvelles fonctionnalités implémentées.

Structure du Projet

Le projet est structuré de manière à séparer clairement les différentes composantes de l'application, en suivant les principes de l'architecture MVC. La racine du projet, nommée **EfreiScolarite**, contient plusieurs dossiers clés, chacun ayant une responsabilité spécifique dans l'ensemble de l'application. Voici une description détaillée de chaque dossier :

- **src/** : Ce dossier contient le code source de l'application. Il est subdivisé en plusieurs sous-dossiers, chacun correspondant à une couche de l'architecture MVC :
 - **controleur/** : Ce dossier contient les classes de contrôle, qui gèrent les interactions entre le modèle et la vue. Il inclut également les classes métier, telles que **Etudiant.java**, **Promotion.java**, et **User.java**, qui représentent les entités principales de l'application.
 - **modele/** : Ce dossier contient les classes de modèle, qui gèrent l'accès à la base de données et les requêtes SQL. Il inclut les classes **Connexion.java**, qui gère la connexion à la base de données, et **Modele.java**, qui implémente les requêtes SQL et les méthodes CRUD (Create, Read, Update, Delete).
 - **vue/** : Ce dossier contient les classes de vue, qui définissent les interfaces utilisateur Java Swing. Il inclut également la gestion des événements utilisateur, tels que les clics de bouton et les saisies de texte.
 - **images/** : Ce dossier contient les ressources graphiques de l'application, telles que les icônes et les images utilisées dans les interfaces utilisateur.
 - **sql/** : Ce dossier contient les scripts de base de données, qui permettent de créer et de configurer la base de données MySQL utilisée par l'application.
- **bin/** : Ce dossier contient les fichiers compilés de l'application, c'est-à-dire les fichiers **.class** générés à partir du code source Java.
- **.settings/** : Ce dossier contient la configuration du projet Eclipse, notamment les paramètres de compilation, les dépendances, et les préférences de l'environnement de développement.

Cette structure claire et organisée facilite la navigation dans le code source, la compréhension des différentes composantes de l'application, et la collaboration entre les développeurs.

Architecture MVC

L'application de gestion du personnel de l'EFREI repose sur l'architecture MVC (Modèle-Vue-Contrôleur), un patron de conception largement utilisé dans le développement d'applications interactives. Cette architecture vise à séparer les différentes responsabilités de l'application en trois composantes distinctes : le modèle, la vue, et le contrôleur. Voici une description détaillée de chaque composante :

- **Modèle (modele/)** :
 - **Connexion.java** : Cette classe est responsable de la gestion de la connexion à la base de données MySQL. Elle établit la connexion, fournit les méthodes pour exécuter les requêtes SQL, et gère la fermeture de la connexion.
 - **Modele.java** : Cette classe implémente les requêtes SQL et les méthodes CRUD (Create, Read, Update, Delete) pour interagir avec la base de données. Elle fournit les méthodes pour récupérer, insérer, mettre à jour, et supprimer les données relatives aux étudiants, aux promotions, et aux utilisateurs.
- **Vue (vue/)** :
 - Les vues sont les interfaces graphiques Java Swing qui présentent les données à l'utilisateur et permettent l'interaction avec l'application. Elles sont responsables de l'affichage des informations et de la capture des événements utilisateur.
 - La gestion des événements utilisateur est assurée par les vues, qui réagissent aux actions de l'utilisateur, telles que les clics de bouton, les saisies de texte, et les sélections dans les listes déroulantes. Ces événements sont ensuite transmis aux contrôleurs pour traitement.
- **Contrôleur (controleur/)** :
 - Les classes métier, telles que **Etudiant.java**, **Promotion.java**, et **User.java**, représentent les entités principales de l'application. Elles contiennent les attributs et les méthodes nécessaires pour gérer les informations relatives à chaque entité.
 - Le contrôleur gère la logique métier de l'application, c'est-à-dire les règles et les opérations qui régissent le comportement de l'application. Il reçoit les événements utilisateur des vues, effectue les traitements nécessaires, et met à jour le modèle en conséquence.

L'architecture MVC assure une séparation claire des préoccupations, ce qui facilite la maintenance, l'évolution, et la testabilité de l'application. Elle permet également de développer des interfaces utilisateur riches et interactives, tout en garantissant la cohérence et la robustesse de l'application.

Fonctionnalités Principales

L'application de gestion du personnel de l'EFREI offre une gamme complète de fonctionnalités pour simplifier et centraliser la gestion des étudiants et des promotions. Voici une description détaillée des principales fonctionnalités :

- **Gestion des Promotions :**
 - Création, modification et suppression de promotions : Permet d'ajouter, de modifier, ou de supprimer des promotions dans le système.
 - Recherche et filtrage des promotions : Permet de rechercher et de filtrer les promotions en fonction de différents critères, tels que le nom, le diplôme, ou la salle.
 - Attribution des salles et diplômes : Permet d'attribuer des salles et des diplômes aux promotions.
- **Gestion des Étudiants :**
 - Inscription et gestion des informations personnelles : Permet d'inscrire de nouveaux étudiants et de gérer leurs informations personnelles, telles que le nom, le prénom, l'adresse, et l'email.
 - Attribution à une promotion : Permet d'attribuer un étudiant à une promotion.
 - Gestion du statut étudiant : Permet de gérer le statut étudiant, tel que étudiant inscrit, étudiant diplômé, ou étudiant en suspension.
 - Système de recherche avancé : Permet de rechercher des étudiants en fonction de différents critères, tels que le nom, le prénom, l'email, ou la promotion.

Ces fonctionnalités permettent au personnel administratif de l'EFREI de gérer efficacement les informations relatives aux étudiants et aux promotions, et de simplifier les tâches administratives courantes. L'application offre une interface utilisateur intuitive et conviviale, ce qui facilite l'utilisation et réduit le temps de formation pour les nouveaux utilisateurs.

Configuration Technique

L'application de gestion du personnel de l'EFREI nécessite une configuration technique spécifique pour fonctionner correctement. Voici les détails de la configuration requise :

- **Langage** : Java
- **Interface graphique** : Java Swing
- **Base de données** : MySQL
 - Serveur : localhost
 - Port : 8889
 - Base de données : `scolarite_efrei`
 - Utilisateur : root

Pour installer et configurer l'application, vous devez disposer d'un environnement de développement Java (JDK) installé sur votre machine. Vous devez également installer un serveur MySQL et configurer une base de données nommée **scolarite_efrei**. Les scripts SQL nécessaires pour créer et initialiser la base de données sont fournis dans le dossier **sql/** du projet.

Vous devez également vérifier les paramètres de connexion à la base de données dans la classe **Connexion.java** et les adapter à votre configuration locale. Assurez-vous que le serveur MySQL est en cours d'exécution et que l'utilisateur **root** a les droits d'accès nécessaires à la base de données **scolarite_efrei**.

Une fois la configuration technique terminée, vous pouvez compiler et exécuter l'application. L'interface graphique Java Swing s'affichera et vous pourrez commencer à utiliser les fonctionnalités de gestion du personnel de l'EFREI.

Installation et Déploiement

Pour installer et déployer l'application de gestion du personnel de l'EFREI, suivez les étapes suivantes :

1. Cloner le projet : Utilisez un client Git pour cloner le projet depuis le dépôt distant vers votre machine locale. Par exemple, vous pouvez utiliser la commande **git clone [URL du dépôt]**.
2. Importer dans Eclipse : Ouvrez l'environnement de développement Eclipse et importez le projet existant depuis le dossier local où vous avez cloné le projet.
3. Configurer MySQL :
 - Créer une base de données 'scolarite_efrei' : Utilisez un client MySQL pour créer une base de données nommée **scolarite_efrei** sur votre serveur MySQL.
 - Importer les scripts SQL du dossier sql/ : Importez les scripts SQL fournis dans le dossier **sql/** du projet pour créer les tables et initialiser les données de la base de données.
 - Vérifier les paramètres de connexion dans Connexion.java : Ouvrez la classe **Connexion.java** et vérifiez les paramètres de connexion à la base de données. Adaptez