



École Supérieure de Technologie - Safi

Rapport de TP1

Application de Gestion des Employés avec Swing et
Modèle MVC

Filière : Génie Informatique

Réalisé par : Omar Akalay

Année Universitaire : 2024/2025

Sous la supervision de : [Nom du superviseur, optionnel]

Contents

Introduction	2
1 Couche Modèle	3
1.1 Définition des Classes et Énumérations	3
1.2 Exemple de Classe <code>Employee</code>	3
2 Couche DAO	4
2.1 Connexion à la Base de Données	4
2.1.1 Exemple de Script SQL	4
2.1.2 Classe <code>DBConnection</code>	4
3 Interface Graphique	5
3.1 Conception de la Vue	5
4 Contrôleur	6
4.1 Gestion des Événements	6
Conclusion	7

Introduction

L'objectif principal de ce projet est de développer une application de gestion des employés pour l'entreprise SEA. Cette application, basée sur l'architecture MVC (Modèle-Vue-Contrôleur), permet une gestion centralisée et efficace des données des employés, des rôles et des postes. Ce rapport détaille les étapes de conception, développement, et implémentation du projet.

Chapter 1

Couche Modèle

1.1 Définition des Classes et Énumérations

Les éléments suivants sont définis pour représenter les données de l'application :

- **Classe Employee** : représente les employés (nom, rôle, poste, etc.).
- **Énumérations Poste et Role** : spécifient les postes et rôles disponibles.

1.2 Exemple de Classe Employee

```
public class Employee {
    private int id;
    private String name;
    private String role;
    private String poste;

    public Employee(int id, String name, String role, String poste) {
        this.id = id;
        this.name = name;
        this.role = role;
        this.poste = poste;
    }

    // Getters et setters
    public int getId() { return id; }
    public void setId(int id) { this.id = id; }
    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
    public String getRole() { return role; }
    public void setRole(String role) { this.role = role; }
    public String getPoste() { return poste; }
    public void setPoste(String poste) { this.poste = poste; }
}
```

Chapter 2

Couche DAO

2.1 Connexion à la Base de Données

Les tables nécessaires pour stocker les données sont créées dans une base MySQL.

2.1.1 Exemple de Script SQL

```
CREATE TABLE Employee (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100),  
    role VARCHAR(50),  
    poste VARCHAR(50)  
);
```

```
CREATE TABLE Poste (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(50)  
);
```

```
CREATE TABLE Role (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(50)  
);
```

2.1.2 Classe DBConnection

```
public class DBConnection {  
    private static final String URL = "jdbc:mysql://localhost:3306/sea";  
    private static final String USER = "root";  
    private static final String PASSWORD = "password";  
    public static Connection getConnection() throws SQLException {  
        return DriverManager.getConnection(URL, USER, PASSWORD);  
    }  
}
```

Chapter 3

Interface Graphique

3.1 Conception de la Vue

L'interface utilisateur utilise Java Swing. Elle permet d'ajouter, afficher, modifier ou supprimer des employés via une interface graphique intuitive.

The screenshot shows a Java Swing application window titled "App". The window contains a form for adding or editing employee data. The form has fields for Nom, Prenom, Email, Telephone, Salaire, Role, and Poste. Below the form is a table displaying the current data. At the bottom of the window are four buttons: Ajouter, Modifier, Supprimer, and Afficher.

ID	Nom	Prenom	Tel	Email	Salaire	Poste	Role
13	Akalay	Omar	0652147485	akalayomar28@gmail.com	33333.0	INGENIEUR_ETUDE_ET_DEVELOPPEMENT	EMPLOYEE

Chapter 4

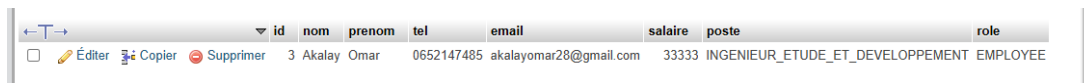
Contrôleur

4.1 Gestion des Événements

La classe `EmployeeController` coordonne les actions entre la Vue et le Modèle.

Exemple de Fonctionnalités

- **Ajouter un Employé** : Ajoute de nouvelles données à la base.
- **Afficher les Employés** : Présente les données sous forme de tableau.
- **Modifier un Employé** : Permet de mettre à jour les informations existantes.
- **Supprimer un Employé** : Supprime un employé sélectionné.



	id	nom	prenom	tel	email	salaire	poste	role
<input type="checkbox"/> Éditer Copier Supprimer	3	Akalay	Omar	0652147485	akalayomar28@gmail.com	33333	INGENIEUR_ETUDE_ET_DEVELOPPEMENT	EMPLOYEE

Conclusion

Ce projet a permis de construire une application complète pour une gestion optimisée des employés. À l'avenir, des améliorations pourraient inclure la gestion des horaires ou des performances.