



المدرسة العليا
للتكنولوجيا-أسفي
ECOLE SUPÉRIEURE
DE TECHNOLOGIE -SAFI



Université Cadi Ayyad
École Supérieure De Technologie-Safi
Département : Informatique
Filière : GI

Rapport Du TP : Gestion des employés
GI 2ème année

Développement d'une application de gestion des employés (MVC,DAO et
Java Swing)

Réalisé par :

BAIROUK Ikram

Enseigné par :

M.EL ABDELLAOUI Said

DATE DE LA SÉANCE : 6 DÉCEMBRE

ANNÉE UNIVERSITAIRE : 2024/2025

Introduction

Dans le cadre de ce projet, nous avons développé une application de bureau pour la gestion des employés, en appliquant les principes de l'architecture MVC (Modèle-Vue-Contrôleur) et en utilisant le design pattern DAO (Data Access Object).

Cette application, conçue avec Java Swing et MySQL, offre une interface intuitive permettant aux administrateurs de gérer efficacement les informations des employés. Elle met en œuvre des concepts clés du génie logiciel, tels que la séparation des responsabilités et l'abstraction de l'accès aux données, pour garantir une solution robuste, maintenable et adaptée aux besoins des utilisateurs.

Ce rapport détaille les étapes de conception et d'implémentation de l'application, en soulignant les défis rencontrés et les solutions apportées.

Table des matières

Introduction	6
1 Contexte du TP	7
1.1 Présentation du TP	7
1.1.1 Problématique	7
1.1.2 Solution proposée	7
1.2 Processus de développement	8
2 Conception	10
2.1 Base de données	10
2.1.1 Conception de la base de données et de la table <code>employes</code>	10
2.2 Environnement de travail	11
2.2.1 Outils utilisés	11
2.2.2 Approches et concepts appliqués	12
3 Réalisation	14
3.1 Présentation des Fonctionnalités de l'Interface	14
3.2 Page d'Accueil	14
3.3 Gestion des Employés	15
3.3.1 Ajouter un Employé	15
3.3.2 Lister les Employés	16
3.3.3 Modifier un Employé	17
3.3.4 Supprimer un Employé	18
3.4 Conclusion	19

Conclusion	19
Webographie	21

Liste des figures

2.1	Schéma de la base de données	11
2.2	Visual Studio Code	11
2.3	XAMPP	12
2.4	phpMyAdmin	12
2.5	Data Access Object (DAO)	12
2.6	Swing - Framework Java pour l'interface graphique	13
2.7	Architecture MVC (Model-View-Controller)	13
3.1	Fenêtre de l'application	15
3.2	Ajout d'un employé	16
3.3	Affichage des employés	17
3.4	Modification d'un employé	18
3.5	Suppression d'un employé	19

Introduction

La gestion des employés est un élément clé pour toute organisation souhaitant améliorer son efficacité et sa gestion interne. Dans le cadre de ce projet, nous avons développé une application de bureau dédiée à la gestion des employés, en appliquant les principes de l'architecture MVC (Modèle-Vue-Contrôleur) et le design pattern DAO (Data Access Object).

Cette application, réalisée en Java avec l'interface Swing et une base de données MySQL, permet aux administrateurs de gérer de manière centralisée les informations des employés, tout en assurant une interface intuitive et facile à utiliser. L'objectif principal de l'application est d'automatiser les tâches courantes de gestion des employés telles que l'ajout, la modification, la suppression et la consultation des données des employés.

L'application repose sur l'architecture MVC pour garantir une séparation claire des responsabilités, avec un **Modèle** qui gère les données des employés, une **Vue** qui affiche l'interface utilisateur, et un **Contrôleur** qui gère les interactions. Le DAO permet de séparer la logique d'accès aux données, facilitant ainsi la maintenance et l'extensibilité de l'application.

Les principales fonctionnalités comprennent la gestion complète des employés, avec des options telles que la création, la mise à jour et la suppression d'employés, ainsi que la consultation des informations. L'interface est simple et fluide, permettant aux administrateurs de gérer efficacement les informations des employés tout en garantissant la sécurité et la fiabilité des données.

À travers ce projet, l'objectif est de fournir une solution fonctionnelle, modulaire et facile à maintenir, permettant à l'organisation de gérer ses employés de manière optimale. Nous explorerons également les possibilités d'amélioration et d'adaptation de cette application pour répondre aux futurs besoins de l'organisation.

Chapitre 1

Contexte du TP

Introduction

Ce travail pratique vise à développer une application de gestion des employés en s'appuyant sur les concepts de l'architecture MVC (Modèle-Vue-Contrôleur), du modèle DAO (Data Access Object) et de Java Swing pour l'interface utilisateur. L'objectif principal est de proposer une solution permettant de gérer efficacement les informations relatives aux employés d'une entreprise, telles que leur ajout, modification ou suppression, tout en garantissant une interface simple et intuitive pour les utilisateurs.

1.1 Présentation du TP

1.1.1 Problématique

Dans le cadre de la gestion des ressources humaines, disposer d'un système efficace pour organiser, stocker et accéder aux données des employés est primordial. Une architecture non structurée peut entraîner des erreurs, des pertes de temps et une mauvaise gestion globale. Ce projet pratique a pour ambition de développer une application permettant de centraliser la gestion des employés, en adoptant l'architecture MVC et en utilisant le modèle DAO pour dissocier la logique métier de la manipulation de la base de données.

1.1.2 Solution proposée

L'application repose sur les principes suivants :

- **Le Modèle (Model)** : Représente les données des employés à travers une classe Java conçue pour gérer leurs informations.

- **La Vue (View)** : Offre une interface utilisateur basée sur Java Swing, permettant de consulter, modifier ou supprimer les informations des employés.
- **Le Contrôleur (Controller)** : Interagit avec l'utilisateur, accède aux données via le DAO et met à jour la vue en fonction des actions effectuées.
- **DAO (Data Access Object)** : Fournit une abstraction pour l'accès aux données, simplifiant les opérations CRUD (Création, Lecture, Mise à jour, Suppression) sur les employés.

1.2 Processus de développement

Analyse des besoins :

- Identification des principales fonctionnalités de l'application.
- Définition des interactions utilisateur, notamment pour l'ajout, la modification, la suppression et l'affichage des données.

Conception de l'architecture :

- Adoption de l'architecture MVC pour une séparation claire entre la logique métier et l'interface graphique.
- Modélisation de la base de données et des classes Java représentant les entités des employés.

Développement :

- Implémentation des classes Java permettant de manipuler les données des employés.
- Création de l'interface graphique en utilisant Java Swing.
- Intégration du modèle DAO pour gérer les opérations sur la base de données.

Tests et validation :

- Validation des fonctionnalités CRUD (Créer, Lire, Mettre à jour, Supprimer).
- Test de l'intégration entre la base de données et l'interface utilisateur.

Conclusion

Ce projet pratique a permis la réalisation d'une application de gestion des employés exploitant les concepts avancés de Java, notamment l'architecture MVC et le modèle DAO. L'application résultante est pleinement fonctionnelle, avec une interface utilisateur conviviale qui facilite la gestion des données des employés. Ce travail a contribué à approfondir ma compréhension des architectures logicielles et des bonnes pratiques en développement Java.

Chapitre 2

Conception

2.1 Base de données

Le développement de l'application a commencé par la conception de la base de données et de la table `employees`, élément central qui permet de gérer efficacement les informations concernant les employés du système. Cette base de données garantit un stockage organisé et simplifie les interactions avec les différentes fonctionnalités de l'application.

2.1.1 Conception de la base de données et de la table `employees`

La table `employees`, implémentée dans une base de données MySQL, regroupe des informations essentielles sur les employés : leur identifiant unique, leur nom, prénom, email, numéro de téléphone, poste, rôle et salaire. La structure de cette table a été conçue pour répondre aux besoins fonctionnels de l'application.

- **Table Staff** : Cette table contient les champs suivants :
 - `id` : Identifiant unique de l'employé (clé primaire).
 - `nom` : Nom de l'employé.
 - `prenom` : Prénom de l'employé.
 - `email` : Adresse e-mail de l'employé.
 - `phone` : Numéro de téléphone de l'employé.
 - `salaire` : Salaire de l'employé.
 - `role` : Rôle de l'employé (ADMIN ou EMPLOYEE).

- **poste** : Poste de l'employé (INGENIEURE_ETUDE_ET_DEVELOPPEMENT, TEAM_LEADER, ou PILOTE).

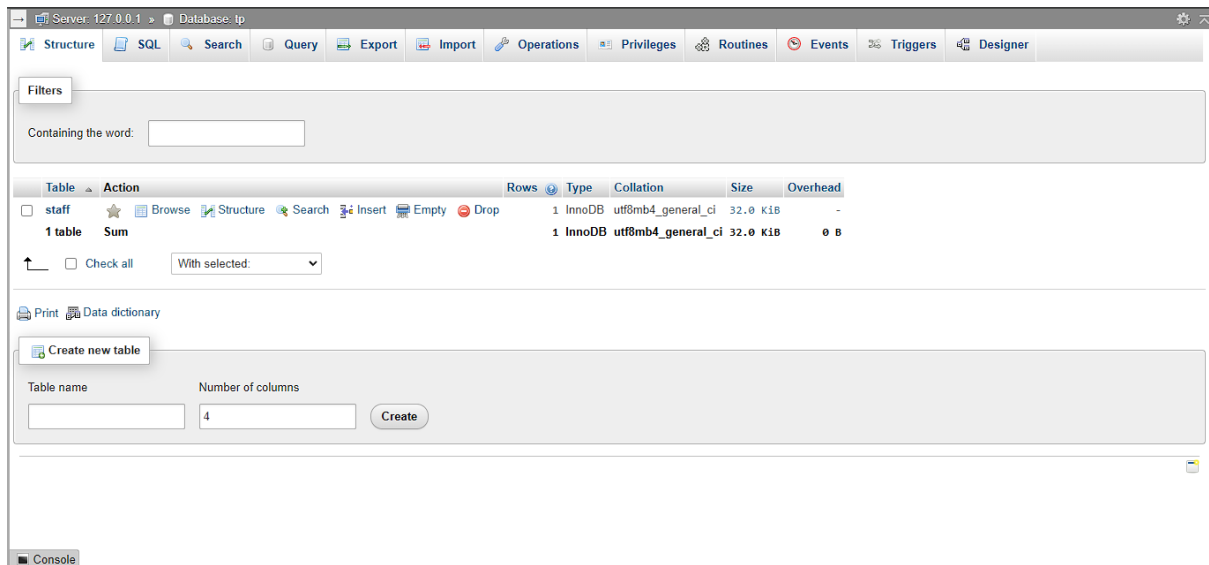


FIGURE 2.1 – Schéma de la base de données

2.2 Environnement de travail

2.2.1 Outils utilisés

Pour assurer une gestion efficace du projet et un développement optimal, plusieurs outils ont été mobilisés. Voici les principaux :

- **Visual Studio Code** : Utilisé comme éditeur principal pour écrire et gérer le code Java, grâce à ses fonctionnalités avancées et ses extensions.



FIGURE 2.2 – Visual Studio Code

- **XAMPP** : Employé pour la configuration d'un environnement serveur local.



FIGURE 2.3 – XAMPP

- **phpMyAdmin** : Utilisé pour gérer la base de données MySQL via une interface graphique web.



FIGURE 2.4 – phpMyAdmin

2.2.2 Approches et concepts appliqués

Le développement a suivi une structure basée sur l'architecture **MVC** (Modèle-Vue-Contrôleur), permettant de séparer les responsabilités entre la gestion des données, l'interface utilisateur et la logique métier. Les concepts et outils suivants ont été utilisés :

- **Model** : Gestion des données et de la logique métier, utilisant le design pattern **DAO** (Data Access Object) pour une interaction optimisée avec la base de données.



FIGURE 2.5 – Data Access Object (DAO)

- **View** : Création de l'interface utilisateur via **Swing**, qui offre une gamme de composants graphiques pour concevoir une interface interactive et intuitive.

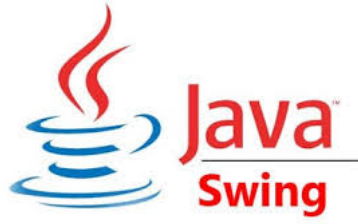


FIGURE 2.6 – Swing - Framework Java pour l'interface graphique

- **Controller** : Gestion des interactions entre l'utilisateur et les données, assurant une communication fluide entre la vue et le modèle.

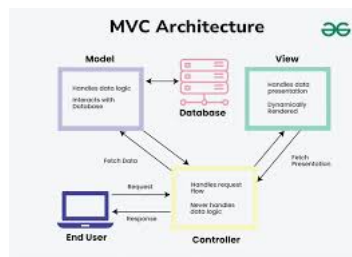


FIGURE 2.7 – Architecture MVC (Model-View-Controller)

L'utilisation du **DAO** a permis une nette séparation entre la logique métier et l'accès aux données, renforçant ainsi la maintenabilité et la modularité du code. Parallèlement, **Swing** a facilité la création d'une interface utilisateur ergonomique tout en respectant les principes de l'architecture **MVC**.

Conclusion

Cette phase initiale du projet a permis de poser les bases essentielles, tant au niveau de la structure de la base de données qu'au niveau des outils et des méthodologies employés. L'architecture claire et les outils performants ont été des éléments clés pour garantir un développement fluide et organisé.

Chapitre 3

Réalisation

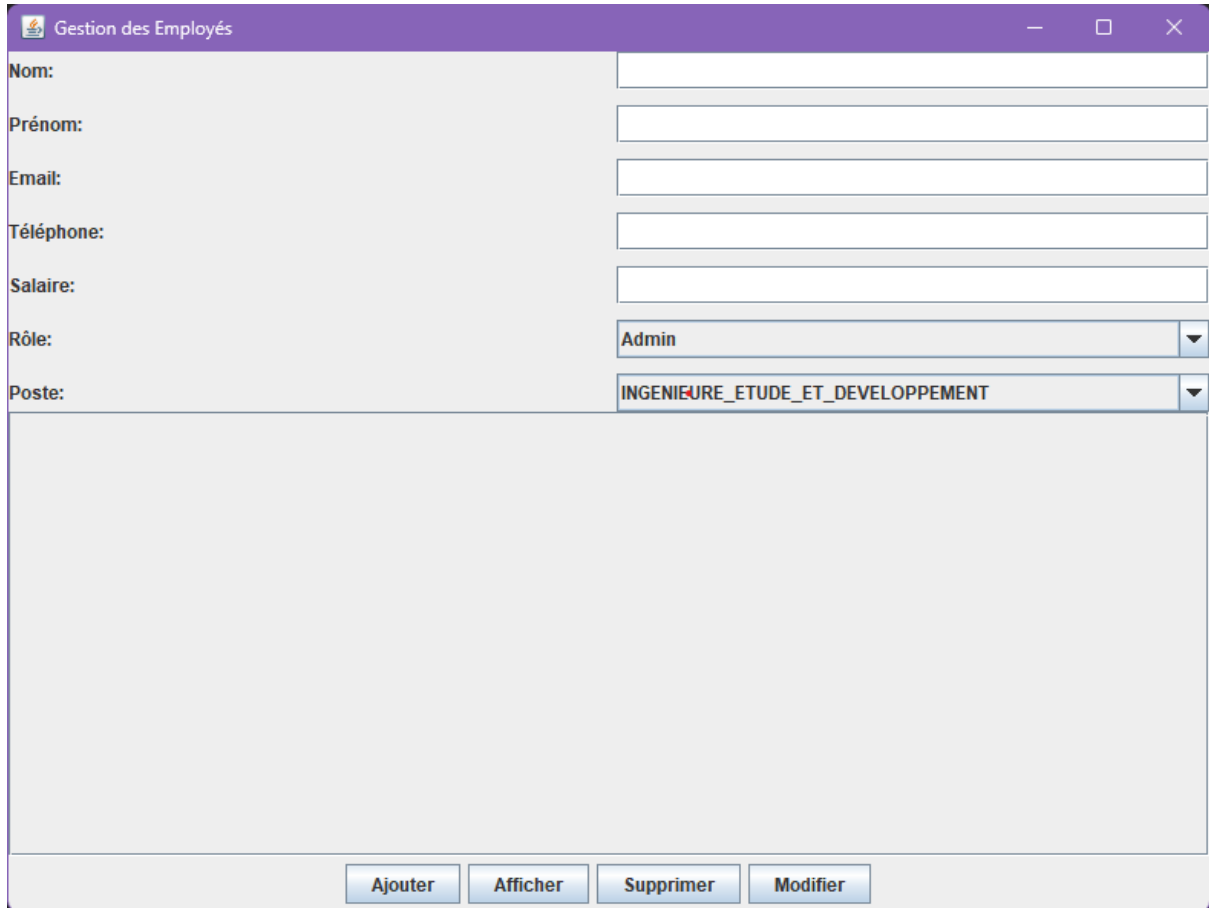
3.1 Présentation des Fonctionnalités de l'Interface

Dans ce chapitre, les fonctionnalités principales de l'application sont décrites. Elle est dédiée à la gestion des employés via une interface intuitive et facile à utiliser. Les actions principales incluent : l'ajout, la modification, la suppression, et la consultation des employés.

3.2 Page d'Accueil

La page d'accueil de l'application propose une interface épurée et fonctionnelle, avec des boutons d'action facilitant la gestion des employés. Les fonctionnalités disponibles sont :

- **Ajouter** : Ajout de nouveaux employés en renseignant leurs informations de base.
- **Lister** : Affichage de la liste complète des employés enregistrés.
- **Modifier** : Modification des informations d'un employé existant.
- **Supprimer** : Suppression d'un employé du système.



The screenshot shows a software window titled "Gestion des Employés". It features a form with the following fields and values:

- Nom: [Empty text box]
- Prénom: [Empty text box]
- Email: [Empty text box]
- Téléphone: [Empty text box]
- Salaire: [Empty text box]
- Rôle: Admin (dropdown menu)
- Poste: INGENIEURE_ETUDE_ET_DEVELOPPEMENT (dropdown menu)

Below the form is a large, empty rectangular area. At the bottom of the window, there are four buttons: "Ajouter", "Afficher", "Supprimer", and "Modifier".

FIGURE 3.1 – Fenêtre de l'application

Ces boutons, clairement identifiables, assurent une navigation intuitive pour l'administrateur.

3.3 Gestion des Employés

L'interface de gestion des employés permet à l'administrateur d'effectuer diverses actions liées à la gestion des informations des employés.

3.3.1 Ajouter un Employé

Le bouton **Ajouter** ouvre un formulaire permettant de saisir les informations essentielles d'un nouvel employé, comme le nom, prénom, email, téléphone, salaire, rôle et poste.

The screenshot displays a web application window titled "Gestion des Employés". It features a form with the following fields: "Nom:" (filled with "baillade"), "Prénom:" (filled with "hiba"), "Email:" (filled with "hibabd2112@gmail.com"), "Téléphone:" (filled with "0635425673"), "Salaire:" (filled with "100000000"), "Rôle:" (a dropdown menu showing "Employe"), and "Poste:" (a dropdown menu showing "T_DEVELOPPEMENT"). A modal dialog box titled "Message" is centered on the screen, displaying an information icon, the text "Employé ajouté avec succès.", and an "OK" button. At the bottom of the application window, there are four buttons: "Ajouter", "Afficher", "Supprimer", and "Modifier".

FIGURE 3.2 – Ajout d'un employé

3.3.2 Lister les Employés

Le bouton **Lister** affiche la liste des employés enregistrés sous forme de tableau. Ce tableau inclut des options **Modifier** et **Supprimer** pour chaque entrée.

Gestion des Employés

Nom:

Prénom:

Email:

Téléphone:

Salaire:

Rôle:

Poste:

ID	Nom	Prénom	Email	Téléphone	Salaire	Rôle	Poste
18	Bairouk	Ikram	ikrambairouk3...	0767748465	1.0E11	ADMIN	INGENIEURE_...
19	baillade	hiba	hibabd2112@g...	0635425673	1.0E8	EMPLOYE	INGENIEURE_...

Ajouter Afficher Supprimer Modifier

FIGURE 3.3 – Affichage des employés

3.3.3 Modifier un Employé

Après la sélection d'un employé (via un événement souris), l'option **Modifier** permet de mettre à jour ses informations (nom, email, téléphone, salaire, rôle, etc.). Une fois les modifications effectuées, les données sont enregistrées dans la base de données.

Gestion des Employés

Nom:

Prénom:

Email:

Téléphone:

Salaire:

Rôle:

Poste:

ID	Nom	Prénom	Email	Téléphone	Salaire	Rôle	Poste
18	Bairouk	Ikram	ikrambairouk3...	0767748465	1.0E11	ADMIN	INGENIEURE_...
19	baillade	hiba	hibabd2112@g...	0635425673	1.0E8	EMPLOYE	INGENIEURE_...

Ajouter Afficher Supprimer Modifier

FIGURE 3.4 – Modification d'un employé

3.3.4 Supprimer un Employé

Le bouton **Supprimer** retire un employé de la liste. L'administrateur sélectionne l'ID de l'employé concerné et confirme la suppression avant de finaliser l'action.

The screenshot shows a web application titled "Gestion des Employés". It features a form for adding new employees with fields for Nom, Prénom, Email, Téléphone, Salaire, Rôle, and Poste. Below the form is a table listing existing employees. A modal dialog titled "Input" is open, prompting the user to enter the ID of the employee to be deleted. The dialog shows the ID "19" entered in the input field. At the bottom of the application, there are four buttons: "Ajouter", "Afficher", "Supprimer", and "Modifier".

ID	Nom	Prénom	Email	Téléphone	Salaire	Rôle	Poste
18	Bairouk	Ikram	ikrambairouk3...	0767748465	1.0E11	ADMIN	INGENIEURE_...
19	baillade	hiba	hibabd2112@g...	0635425673	1.0E8	EMPLOYEE	INGENIEURE_...

FIGURE 3.5 – Suppression d'un employé

3.4 Conclusion

En résumé, l'application propose une interface ergonomique dédiée à la gestion des employés. Elle permet aux administrateurs d'ajouter, consulter, modifier, et supprimer les employés de manière efficace. Cette gestion simplifiée assure une base de données toujours à jour et garantit une administration rapide et fluide.

Conclusion Générale

La réalisation de ce travail pratique a été une expérience significative qui m'a permis de renforcer mes compétences en développement d'applications et de comprendre les différentes étapes nécessaires pour concevoir un système de gestion performant. Travailler sur ce projet a été l'occasion d'explorer les concepts théoriques étudiés en cours tout en les appliquant à un cas concret et structuré.

Le TP avait pour objectif de développer une application permettant de gérer les informations des employés de manière efficace à travers les fonctionnalités d'ajout, de modification, de suppression et de consultation. Grâce à ce projet, j'ai pu approfondir mes connaissances en Java, notamment en mettant en œuvre des principes essentiels tels que la gestion des bases de données, la manipulation des interfaces graphiques et l'intégration d'éléments interactifs.

En concevant l'application, j'ai utilisé une architecture claire et structurée qui facilite l'organisation des données et leur interaction avec l'interface utilisateur. La gestion des employés, comme l'ajout ou la modification, a été implémentée de manière intuitive, ce qui reflète une attention particulière portée à l'ergonomie et à la simplicité d'utilisation. Ce projet m'a également donné l'opportunité de maîtriser l'utilisation d'outils tels que MySQL pour la gestion de la base de données et Visual Studio Code pour le développement de l'application.

Les défis rencontrés tout au long de ce TP, notamment la synchronisation des données entre l'interface et la base de données, ou encore la validation des entrées utilisateur, m'ont permis de développer des stratégies efficaces pour résoudre des problèmes complexes. Ces expériences m'ont appris l'importance de la rigueur dans le développement logiciel, ainsi

que l'importance de tester et de valider chaque étape du processus.

Ce TP a donc été un véritable tremplin pour renforcer mes compétences techniques et mes capacités d'analyse. Il a également amélioré ma capacité à structurer un projet, à collaborer avec des outils modernes et à respecter les bonnes pratiques de développement.

En guise de conclusion, cette expérience m'a permis de mettre en œuvre des compétences précieuses en programmation et en gestion de projet informatique. Je suis convaincu que les acquis de ce TP seront des atouts majeurs pour mes projets futurs dans le domaine du développement logiciel. Ce travail pratique m'a donné les outils nécessaires pour aborder avec confiance et méthode des problématiques similaires dans des contextes professionnels.

Webographie

[1] <https://github.com/zdrues/TP7.git>