

# Cahier des charges

## Systeme de gestion d'employés

### 1. Contexte et objectifs du projet

Le projet consiste à développer un système de gestion des employés basé sur une interface graphique écrite en Java et utilisant **Apache NetBeans** comme IDE. Les données sont gérées à l'aide d'une base de données relationnelle **MySQL**. L'objectif principal est de permettre aux utilisateurs (administrateurs ou gestionnaires) d'effectuer des opérations courantes sur les données des employés, telles que l'ajout, la visualisation, la mise à jour et la suppression.

### 2. Technologies utilisées

- **Langage de programmation** : Java
- **Base de données** : MySQL
- **Frameworks/Librairies** :
  - Swing pour l'interface utilisateur
  - JDBC pour l'interaction avec la base de données
  - JDateChooser pour la sélection de dates
- **IDE** : Apache NetBeans

### 3. Fonctionnalités principales

#### 3.1 Authentification

- **Écran de connexion** :
  - Permet à l'utilisateur de s'authentifier via un nom d'utilisateur et un mot de passe.
  - Les identifiants sont validés via une requête SQL sur la base de données MySQL.

#### 3.2 Tableau de bord (Home)

- **Menu principal** :
  - Accès aux fonctionnalités principales :

- Ajouter un employé
- Consulter les employés
- Modifier les informations d'un employé
- Supprimer un employé

### 3.3 Gestion des employés

- **Ajouter un employé :**

- Formulaire permettant d'ajouter les détails d'un employé, tels que :
  - Nom
  - Nom du père
  - Date de naissance
  - Adresse
  - Numéro de téléphone
  - Salaire
  - E-mail
  - Niveau d'éducation
  - Poste
  - Statut matrimonial
  - Identifiant unique généré automatiquement
- Les données sont sauvegardées dans la base de données MySQL.

- **Voir les employés :**

- Affiche la liste des employés enregistrés.
- Les données sont récupérées de la base MySQL.

- **Modifier un employé :**

- Permet de sélectionner un employé existant et de mettre à jour ses informations.

- **Supprimer un employé :**

- Permet de sélectionner un employé par son identifiant et de supprimer ses informations de la base de données.

### 3.4 Splash Screen

- **Écran de démarrage animé :**
  - Affiche un écran d'introduction avant de rediriger l'utilisateur vers l'écran de connexion.

## 4. Structure de la base de données

### Table employe

Colonne	Type	Description
id	INT	Identifiant unique de l'employé
nom	VARCHAR(100)	Nom de l'employé
fname	VARCHAR(100)	Nom du père
date	DATE	Date de naissance
adresse	VARCHAR(255)	Adresse de l'employé
telephone	VARCHAR(15)	Numéro de téléphone
email	VARCHAR(100)	Adresse e-mail
education	VARCHAR(50)	Niveau d'éducation
designation	VARCHAR(100)	Poste occupé
status	VARCHAR(50)	Statut matrimonial
salary	DOUBLE	Salaire

## 5. Contraintes techniques

- La connexion à la base de données MySQL est assurée via le driver JDBC.
- Les interactions avec la base de données doivent inclure la gestion des exceptions (ex : erreurs de connexion ou requêtes incorrectes).
- Les champs du formulaire doivent être validés avant l'envoi des données à la base (ex : vérifier que les champs obligatoires sont remplis).

## 6. Sécurité

- Les mots de passe des utilisateurs doivent être stockés de manière sécurisée
- Validation des données pour prévenir les attaques de type SQL Injection.
- Limiter l'accès au système aux utilisateurs authentifiés.

## 7. Scénarios d'utilisation

### 1. Connexion au système :

- L'utilisateur entre son nom d'utilisateur et son mot de passe pour accéder au tableau de bord.

## 2. **Ajout d'un employé :**

- L'utilisateur remplit le formulaire et enregistre les informations dans la base de données.

## 3. **Modification des informations :**

- L'utilisateur sélectionne un employé, met à jour les informations nécessaires et sauvegarde.

## 4. **Suppression d'un employé :**

- L'utilisateur sélectionne un employé et confirme sa suppression.

## **8. Diagramme d'architecture**

### 1. **Client-Serveur :**

- **Client :** Interface graphique basée sur Java Swing.
- **Serveur :** Base de données MySQL hébergée localement ou sur un serveur dédié.

### 2. **Flux de données :**

- Les données saisies dans l'interface sont envoyées à la base via des requêtes SQL.
- Les résultats des requêtes sont affichés dans l'interface utilisateur.

Merci !