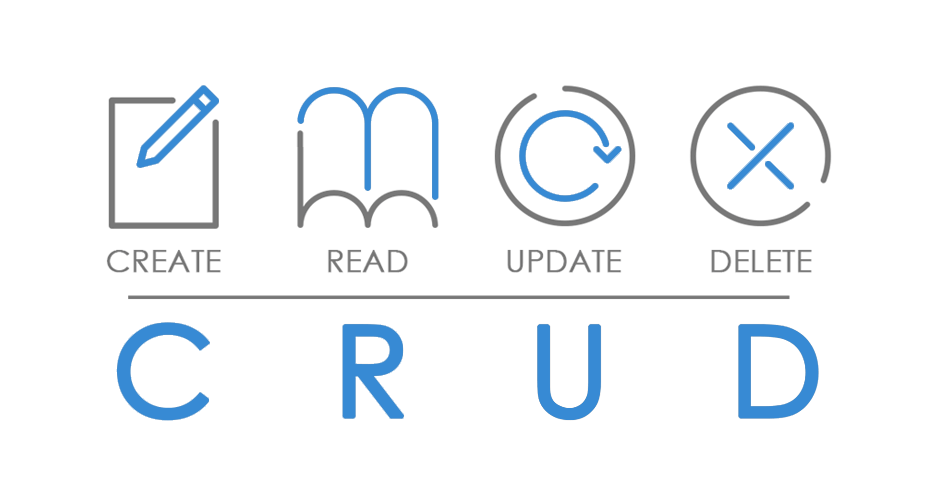
**Rapport de l'application CRUD de gestion des étudiants avec JavaFX et base de données**



**Réalisé par :**

* Fatima Zahra Hajaj.
* El Fahssi Hanan.

**Introduction :**

Ce rapport présente le développement d'une application CRUD de gestion des étudiants en utilisant JavaFX pour l'interface utilisateur et une base de données pour stocker les informations des étudiants. L'objectif principal de cette application est de permettre aux utilisateurs de créer, lire, mettre à jour et supprimer des informations sur les étudiants de manière conviviale et de permet à l’utilisateur d’imprimer les détails d’un étudiant ,ce qui peut être utile pour générer des rapports ou des documents à partir des données du tableau.

**Conception de l'application:**

Spécification des fonctionnalités: Les fonctionnalités de base comprennent l'ajout d'un nouvel étudiant avec des détails tels que le nom, le prénom, l’email, etc., l'affichage de la liste des étudiants existants, la mise à jour des informations d'un étudiant sélectionné , la suppression d'un étudiant de la base de données et l’impression des données qu’imprimer les détails d’un étudiant sélectionné à partir du tableau.

**Conception de l'interface utilisateur:**

La conception de l'interface utilisateur est réalisée en utilisant le code JavaFX directement, sans utiliser Scene Builder. Cela implique la création des différents éléments de l'interface tels que les fenêtres, les boutons, les champs de texte, les tableaux, etc., en utilisant les classes et les méthodes fournies par JavaFX.

**Mise en place de la base de données:**

Choix du système de gestion de base de données (SGBD): Par exemple, MySQL, PostgreSQL, SQLite, etc.

Création de la structure de la base de données: Définition d'une table "étudiant" avec des colonnes pour stocker les informations telles que l'ID, le nom, le prénom, l’email, etc.

**Connexion à la base de données:**

Utilisation de JDBC: JDBC (Java Database Connectivity) pour établir une connexion entre l'application JavaFX et la base de données.

Configuration des paramètres de connexion: Configuration des informations de connexion telles que le nom d'utilisateur, le mot de passe, l'URL de la base de données, etc.

**Implémentation des fonctionnalités CRUD:**

* **Création d'un nouvel étudiant:**

La création d'un nouvel étudiant est réalisée en utilisant des champs de texte pour permettre à l'utilisateur d'entrer les détails, puis enregistrer ces informations dans la base de données en utilisant des requêtes SQL appropriées.

* **Lecture des informations des étudiants:**

Les informations des étudiants sont récupérées à partir de la base de données en utilisant des requêtes SQL et sont affichées dans un tableau ou une liste pour permettre à l'utilisateur de visualiser les données des étudiants existants.

* **Mise à jour des informations des étudiants:**

Lorsqu'un étudiant est sélectionné dans la liste, ses détails sont affichés dans des champs de texte préremplis. L'utilisateur peut alors modifier les informations et les enregistrer dans la base de données en utilisant des requêtes SQL de mise à jour.

* **Suppression d'un étudiant:**

Lorsqu'un étudiant est sélectionné dans la liste, l'utilisateur peut choisir de le supprimer. Cette action est réalisée en utilisant des requêtes SQL de suppression pour supprimer les données correspondantes de la base de données.

* **Impression des données :**

L’impression des données ce fait avec le bouton « **print** » qu’est utilisé pour imprimer les détails d’un étudiant sélectionné dans le tableau . Lorsque le bouton est cliqué, la méthode « **printSummary**() » est appelée. Cette méthode récupère l’étudiant sélectionné dans le tableau, crée un objet « **PrinterJob** » et affiche la boite de dialogue d’impression .

**Conclusion:**

L'application CRUD de gestion des étudiants développée avec JavaFX et une base de données offre une solution pratique et conviviale pour gérer les informations des étudiants. En utilisant JavaFX pour la conception de l'interface utilisateur et en se connectant à une base de données via JDBC, cette application permet aux utilisateurs d'effectuer facilement des opérations de création, de lecture, de mise à jour et de suppression sur les données des étudiants. En suivant les étapes de conception, de mise en place de la base de données, de connexion, d'implémentation des fonctionnalités CRUD, de gestion des erreurs et de tests, l'application peut être développée de manière efficace et fiable, même sans utiliser Scene Builder. Des tests rigoureux et un processus de débogage approprié contribuent à garantir le bon fonctionnement de l'application. L'application peut être améliorée en ajoutant des fonctionnalités supplémentaires telles que la recherche d'étudiants, des filtres, des statistiques, etc., en fonction des besoins spécifiques de l'utilisateur.