



**Faculté des Sciences et Technologies Département de Mathématiques**

# **Gestion des Réservations de Salles Informatiques pour l'Université**

Réalisé par  
***Carine BREIDY***  
***Dalia MAOUCHE***  
***Mouaad Ibnelaryf***

Année 2023-2024

**Master 2 Ingénierie Statistique et Numérique- Data Sciences**

---

# Table des matières

<b>1</b>	<b>Contexte du Projet</b>	<b>3</b>
<b>2</b>	<b>Modélisation des Données</b>	<b>3</b>
2.1	Modèle Conceptuel de Données . . . . .	3
2.2	Modèle Logique de Données . . . . .	4
2.3	Modèle Physique de Données . . . . .	4
<b>3</b>	<b>Architecture du Projet</b>	<b>5</b>
<b>4</b>	<b>Interface Utilisateur avec JavaFX</b>	<b>5</b>
<b>5</b>	<b>Gestion des Données avec JDBC</b>	<b>7</b>
<b>6</b>	<b>Fonctionnalités Clés de l'Application</b>	<b>7</b>
<b>7</b>	<b>Conclusion</b>	<b>8</b>

---

# 1 Contexte du Projet

Dans le cadre de notre cours de programmation Java et base de données, nous allons développer une application de gestion des salles informatiques pour un campus universitaire. Cette application vise à faciliter la réservation et la gestion des salles, en assurant une utilisation optimale des ressources disponibles.

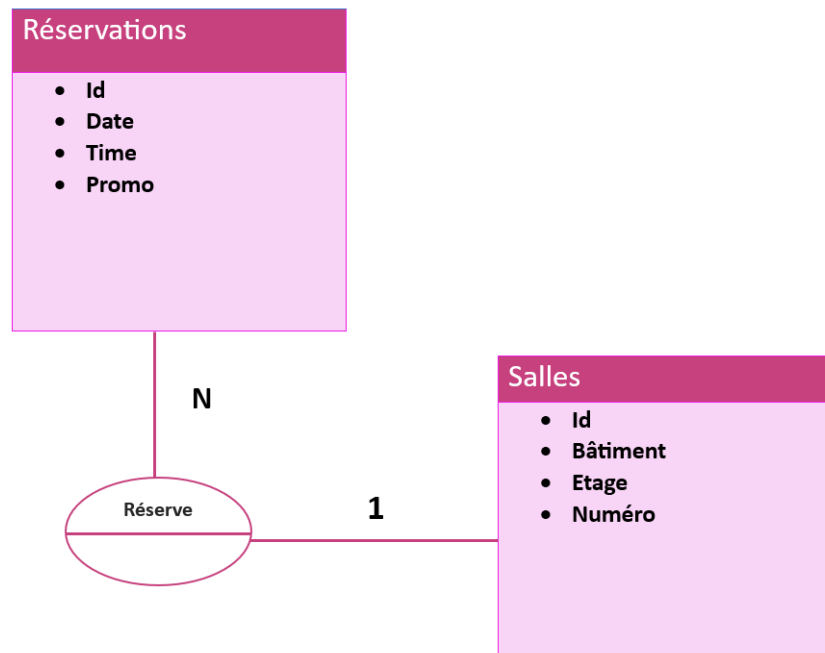
Technologies et Outils Utilisés :

- Java avec JavaFX : Pour le développement de l'interface utilisateur.
- JDBC avec MySQL : Pour la gestion de la base de données.
- Eclipse : En tant qu'environnement de développement intégré pour la programmation Java.
- XAMPP avec phpMyAdmin : Pour la mise en place et la gestion de la base de données MySQL.

## 2 Modélisation des Données

### 2.1 Modèle Conceptuel de Données

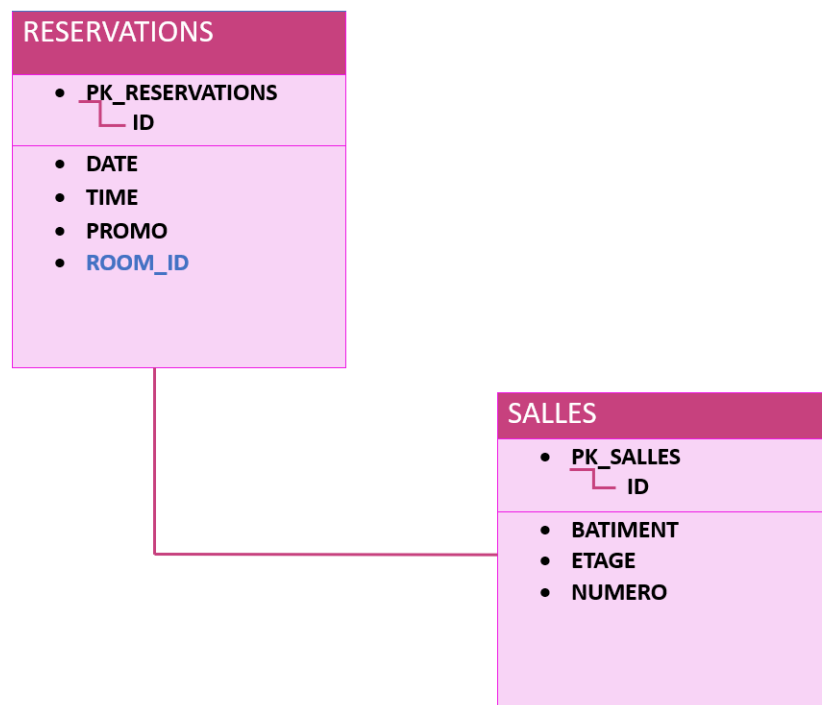
La modélisation de notre base de données débute avec le Modèle Conceptuel de Données (MCD), qui sert à définir de manière abstraite les entités et leurs relations au sein du système de réservation. Le MCD montre les entités Réservations et Salles, et illustre une association Réserve indiquant qu'une salle peut être réservée plusieurs fois, tandis qu'une réservation ne concerne qu'une seule salle à la fois. Ce modèle se concentre sur la représentation des entités et de leurs relations sans s'attarder sur les détails d'implémentation ou les contraintes techniques.



---

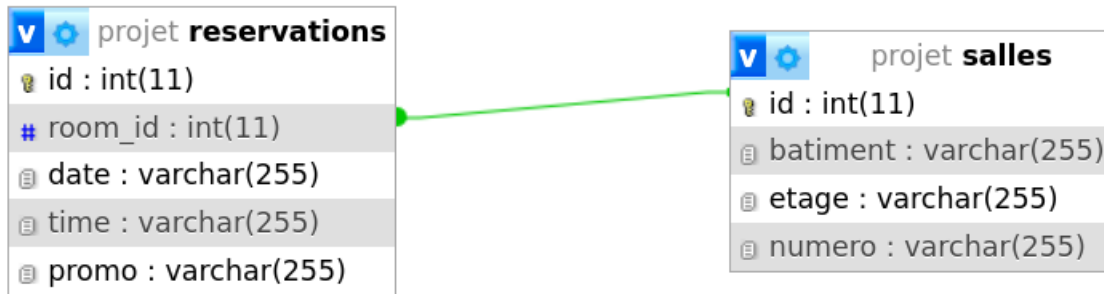
## 2.2 Modèle Logique de Données

Le passage au Modèle Logique de Données (MLD) transforme cette représentation abstraite en une structure plus concrète adaptée aux systèmes de gestion de base de données relationnels. Dans le MLD, les entités deviennent des tables avec des attributs, et les relations sont exprimées à travers des clés étrangères. C'est à ce niveau que nous définissons les clés primaires pour chaque table, comme ID pour les entités Réservations et Salles, et que nous établissons la connexion entre elles via la clé étrangère ROOM\_ID.



## 2.3 Modèle Physique de Données

Enfin, le Modèle Physique de Données (MPD) concrétise le MLD en spécifiant les détails techniques tels que les types de champs exacts et les contraintes d'intégrité pour la base de données. Alors que le MLD énonce la structure et les relations, le MPD se charge de la création effective des tables et de l'implémentation des contraintes comme les clés étrangères, les index et les politiques de sécurité.



La cohérence entre le MCD, le MLD et le MPD assure que notre conception de la base de données répond aux besoins fonctionnels tout en étant techniquement viable, posant ainsi une fondation solide pour le développement et l'expansion futures du système de réservation. Les diagrammes qui accompagnent chaque modèle fournissent une visualisation claire de la structure à chaque étape, facilitant ainsi la compréhension et l'implémentation de la base de données pour le système de réservation.

### 3 Architecture du Projet

Notre application est structurée en quatre classes principales :

- Main : Le point d'entrée de l'application qui gère l'interface utilisateur. (fichier Main.java)
- Room et Reservation : Classes modèles représentant les salles et les réservations. (Reservation.java et Room.java)
- DatabaseManager : Classe responsable de la gestion des opérations de base de données. (DatabaseManager.java)

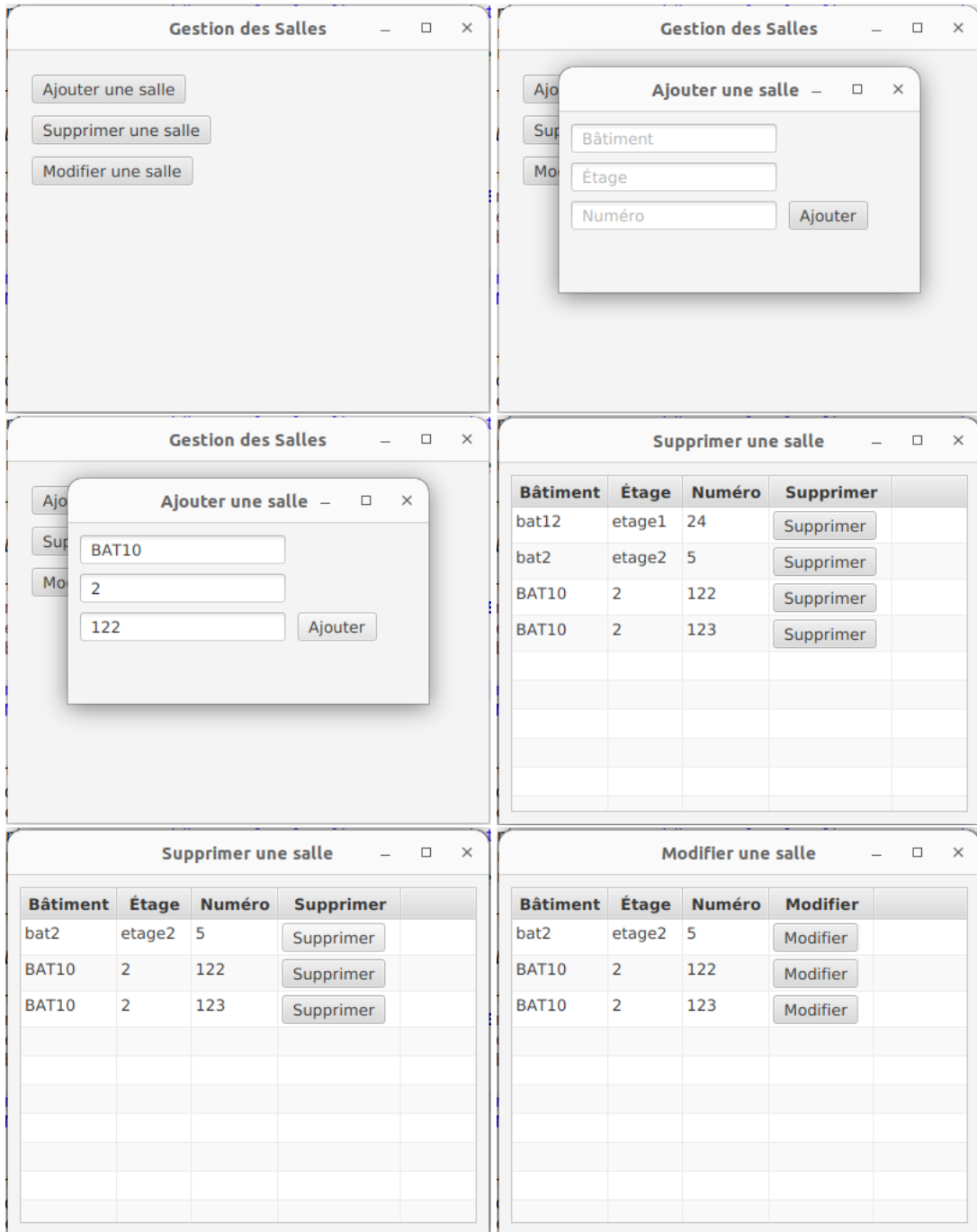
Cette architecture facilite la maintenance et l'évolution future de l'application. Nous avons choisi cette architecture pour sa simplicité et sa clarté. Elle permet de séparer la logique de l'interface utilisateur de celle de la base de données, ce qui est crucial pour la maintenabilité du code.

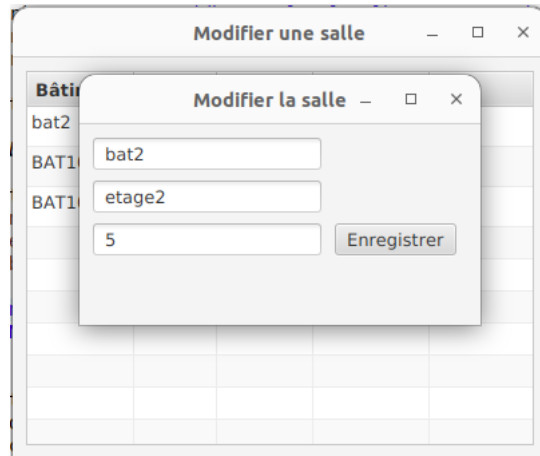
### 4 Interface Utilisateur avec JavaFX

Éléments de notre Interface :

Notre interface utilisateur, conçue avec JavaFX, comprend plusieurs composants interactifs :

- Boutons pour ajouter, supprimer et modifier les salles.
- Champs de texte pour entrer les détails des salles et des réservations.
- Tableaux pour afficher les salles disponibles et les réservations existantes.





## 5 Gestion des Données avec JDBC

La classe DatabaseManager implémente toutes les opérations CRUD nécessaires :

- insertRoom et deleteRoom pour gérer les salles.
- insertReservation et deleteReservation pour les réservations.

Structure de la Base de Données :

Nous avons deux tables principales : salles et reservations, avec des relations clairement définies entre elles.

Exemples de notre Requêtes SQL :

Insertion d'une salle : `INSERT INTO salles (batiment, etage, numero) VALUES (?, ?, ?)`

Recherche de réservations : `SELECT * FROM reservations WHERE room_id = ?`

## 6 Fonctionnalités Clés de l'Application

Gestion des Salles :

Les utilisateurs peuvent facilement ajouter, supprimer et modifier les salles. Chaque action est immédiatement reflétée dans la base de données.

Réservation des Salles :

La réservation des salles est intuitive, permettant aux utilisateurs de choisir la salle, la date, l'heure et promo. Gestion des Contraintes

---

## 7 Conclusion

Ce projet a été une opportunité pour développer nos compétences en Java et en gestion de bases de données. Il a également renforcé notre compréhension de la conception d'interfaces utilisateur, et pour l'avenir, nous envisageons d'améliorer l'interface utilisateur et d'intégrer des fonctionnalités avancées comme un calendrier de réservation.