

## FICHE EXPLICATIVE

Projet : Système de gestion d'une salle de sport

### 1. Présentation du projet

Ce projet a pour objectif de concevoir et exploiter une base de données permettant la gestion complète d'une salle de sport. La base de données doit permettre de gérer les membres, les adhésions, les coachs, les cours, les salles, les séances ainsi que les réservations, tout en respectant des règles de sécurité et d'intégrité des données.

L'ensemble est exploité via une application développée en C#, permettant aux utilisateurs de se connecter et d'interagir avec la base selon leurs droits.

### 2. Choix techniques

Le SGBD choisi pour ce projet est MySql Workbench, car il est fiable, largement utilisé et compatible avec les applications C#. La communication entre l'application et la base de données est réalisée à l'aide de la bibliothèque MySql.Data.MySqlClient.

L'application C# repose sur une classe dédiée (CommandeManager) qui centralise l'exécution des requêtes SQL :

- lecture des données (SELECT)
- modification des données (INSERT, UPDATE, DELETE)
- calculs et statistiques (fonctions d'agrégation)

Ce choix permet de séparer clairement la logique applicative de l'accès à la base de données, ce qui rend le projet plus lisible et plus maintenable.

### 3. Structure et organisation de la base de données

La base de données est organisée autour de plusieurs entités principales :

- Utilisateur : stocke les informations des membres, des coachs et des administrateurs.
- Rôle : permet de distinguer les différents types d'utilisateurs.
- Coach : contient les informations spécifiques aux coachs.
- Salle : représente les différentes salles disponibles.
- TypeCours et Seance : décrivent les cours proposés et leur planification.
- Reservation : gère les inscriptions des membres aux séances.
- TypeAdhesion et Souscription : permettent de suivre les adhésions et leur statut.

Des contraintes ont été mises en place afin d'assurer la cohérence des données (capacités positives, statuts valides, unicité de certaines informations comme les adresses e-mail).

### 4. Gestion des utilisateurs et des priviléges

Plusieurs utilisateurs MySQL ont été créés avec des priviléges distincts :

- un administrateur principal, disposant de tous les droits sur la base
- un administrateur secondaire, avec des droits limités à la gestion des données courantes

- un utilisateur applicatif, celui qui sert à l'authentification initiale avant que le programme ne bascule sur le compte Gérant ou Staff utilisé par l'application C# pour exécuter les requêtes

Cette gestion des privilèges permet de sécuriser l'accès à la base de données et de respecter les contraintes du sujet.

## 5. Requêtes SQL et fonctionnalités

Un fichier SQL dédié regroupe l'ensemble des requêtes utilisées ou utilisables par l'application. Ces requêtes couvrent :

- la gestion des utilisateurs, des coachs et des réservations
- l'affichage des séances et des salles
- la génération de statistiques

Le fichier contient notamment :

- des jointures (JOIN, LEFT JOIN, RIGHT JOIN),
- des sous-requêtes
- une requête ensembliste (UNION)
- les 6 fonctions d'agrégation vues en cours (COUNT, SUM, AVG, MIN, MAX, GROUP CONCAT)

Ces requêtes répondent aux exigences fonctionnelles et statistiques demandées dans le sujet.

## 6. Répartition du travail

Le projet a été réalisé en trinôme :

- Hubert Ferrand : conception de la base de données, scripts SQL (création et peuplement).
- Clément Doat : développement de l'application C# et gestion de la connexion à la base.
- Cyrille Balamba : rédaction des requêtes SQL, gestion des utilisateurs et documentation.