

Dossier Technique - FitZone

1. Introduction

Ce document présente les choix techniques et l'architecture de l'application web FitZone, plateforme de gestion sportive pour membres, entraîneurs et administrateurs.

2. Architecture et Technologies

2.1 Architecture globale

L'application est construite selon une architecture 3 tiers :

1. Client : Navigateur web (Chrome, Firefox, Edge, Safari)
2. Serveur Web : PHP pour le traitement des requêtes et la logique métier
3. Base de données : MySQL pour le stockage des données (membres, cours, abonnements, paiements)

Schéma simplifié :

2.2 Technologies utilisées

- Frontend : HTML5, CSS3, Bootstrap
- Backend : PHP , gestion des sessions pour le login
- Base de données : MySQL / phpMyAdmin
- Outils : VS Code, Draw.io / Figma pour schémas et maquettes

2.3 Sécurité et performance

- Sécurité : mot de passe hashé, validation des formulaires, HTTPS
 - Performance : interface responsive, chargement rapide des pages
 - Maintenance : structure modulaire pour faciliter les évolutions
-

3. Modèles UML

3.1 Diagramme de classes

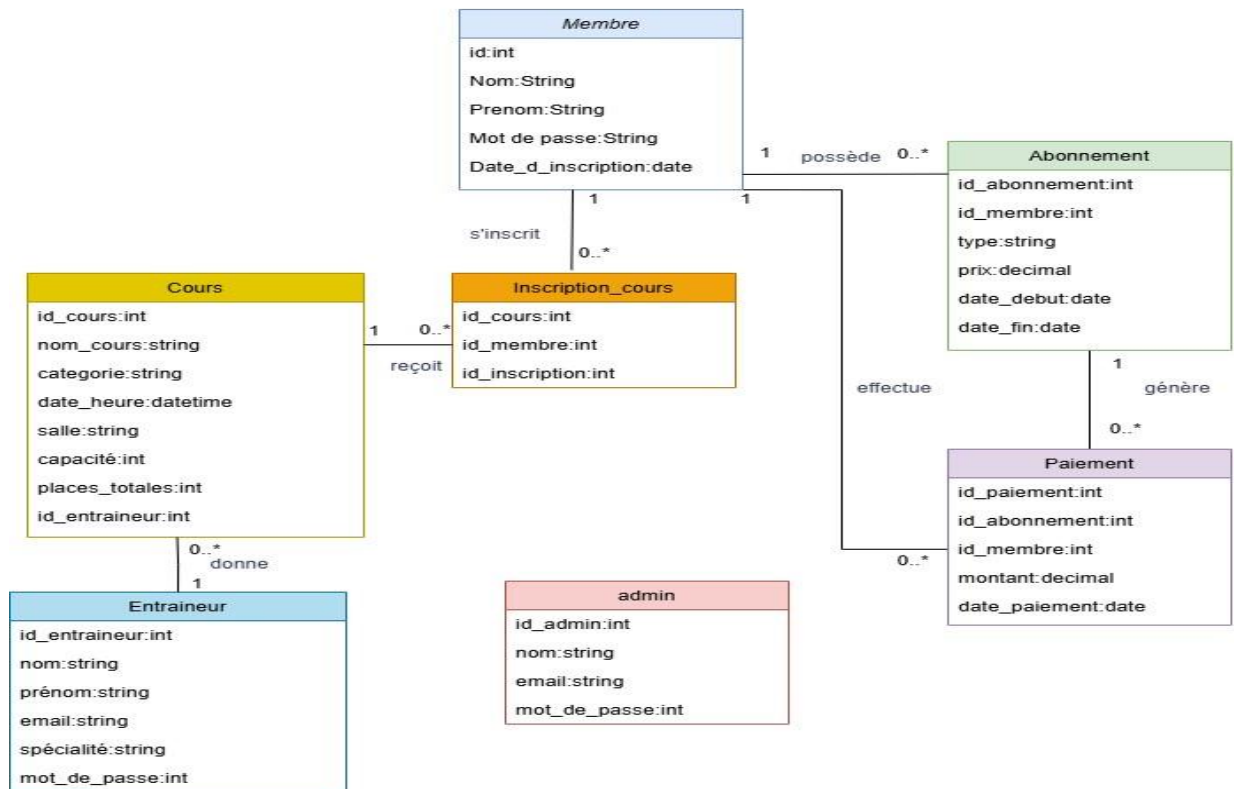
Voici simplement le nom de chaque table et ses attributs :

- **Membre** : id, nom, prénom, email, mot_de_passe, date_inscription
- **Cours** : id, nom, date, heure, id_entraineur, categorie, salle, capacité
- **Abonnement** : id, type, date_debut, date_fin, id_membre, prix
- **Paielement** : id, montant, date, id_abonnement, id_membre
- **Entraîneur** : id, nom, prénom, email, spécialité
- **Inscription_cours** : id, id_membre, id_cours

Relations :

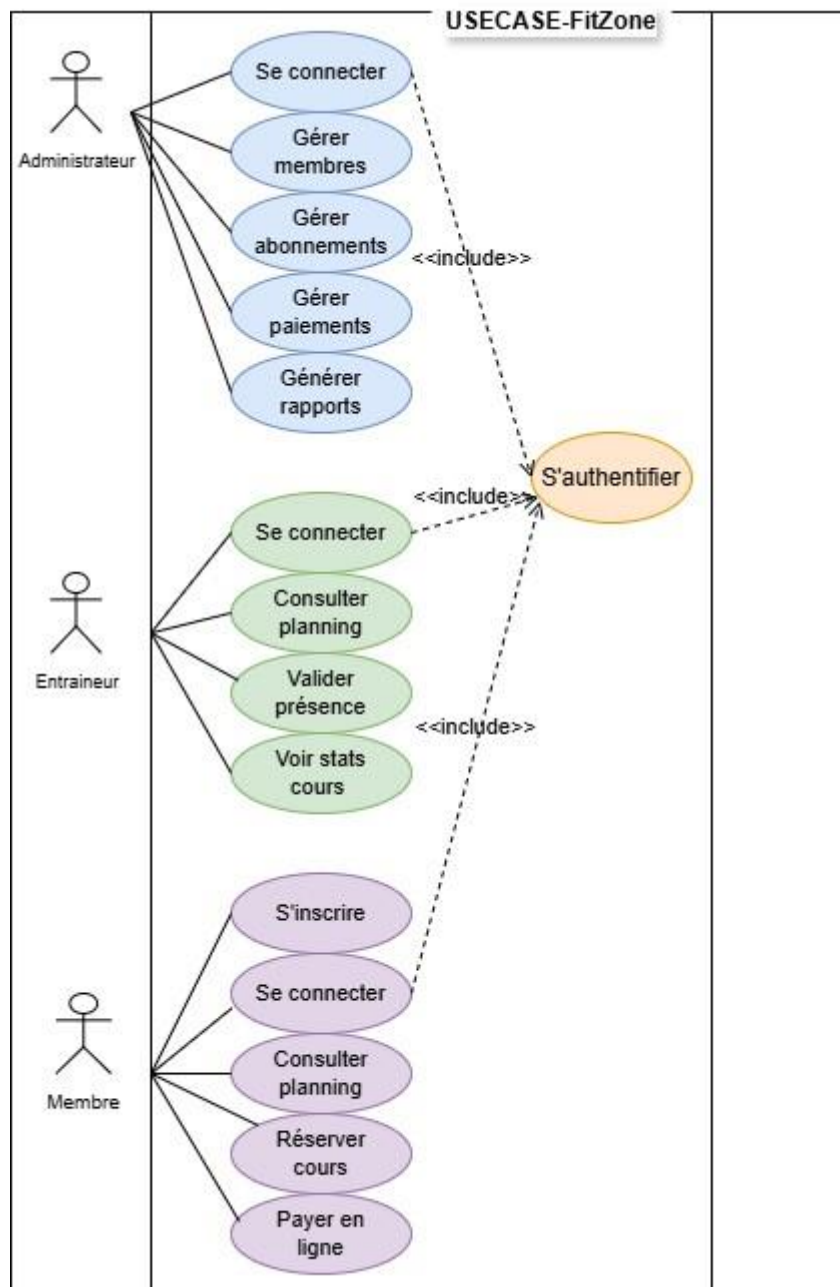
- Un **Membre** peut avoir **de 0 à plusieurs Abonnements**.
 - Un **Membre** peut effectuer **de 0 à plusieurs Paiements**.
 - Un **Abonnement** peut générer **de 0 à plusieurs Paiements**.
 - Un **Membre** peut s'inscrire **à 0 à plusieurs Cours** via **Inscription_Cours**.
 - Un **Cours** peut recevoir **0 à plusieurs inscriptions** via **Inscription_Cours**.
 - Un **Entraîneur** peut donner **de 0 à plusieurs Cours**.
-

Diagramme de classes-FitZone



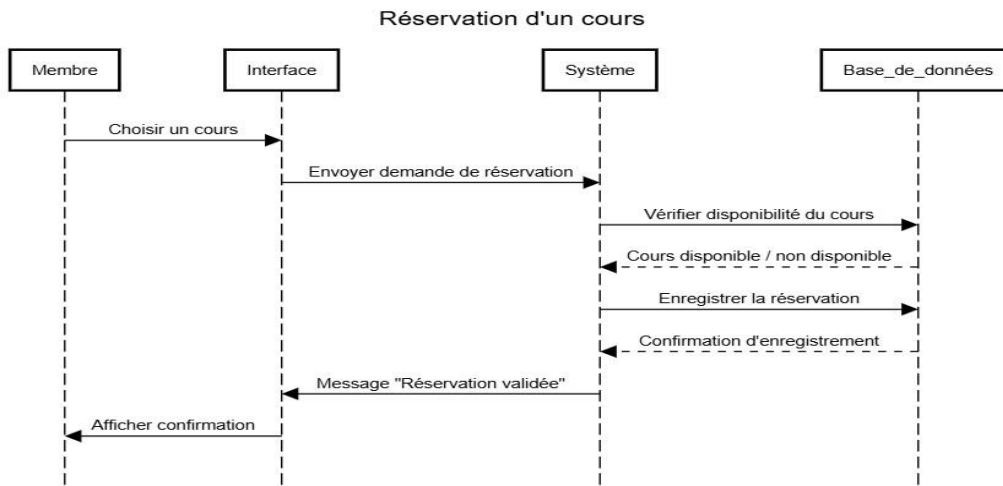
3.2 Diagrammes Use Case

- Administrateur : gérer membres, abonnements, paiements, cours, rapports
- Entraîneur : consulter planning, valider présences, consulter statistiques simples
- Membre : s'inscrire, se connecter, réserver un cours, consulter planning, payer abonnement



3.3 Diagramme de séquence (exemple réservation)

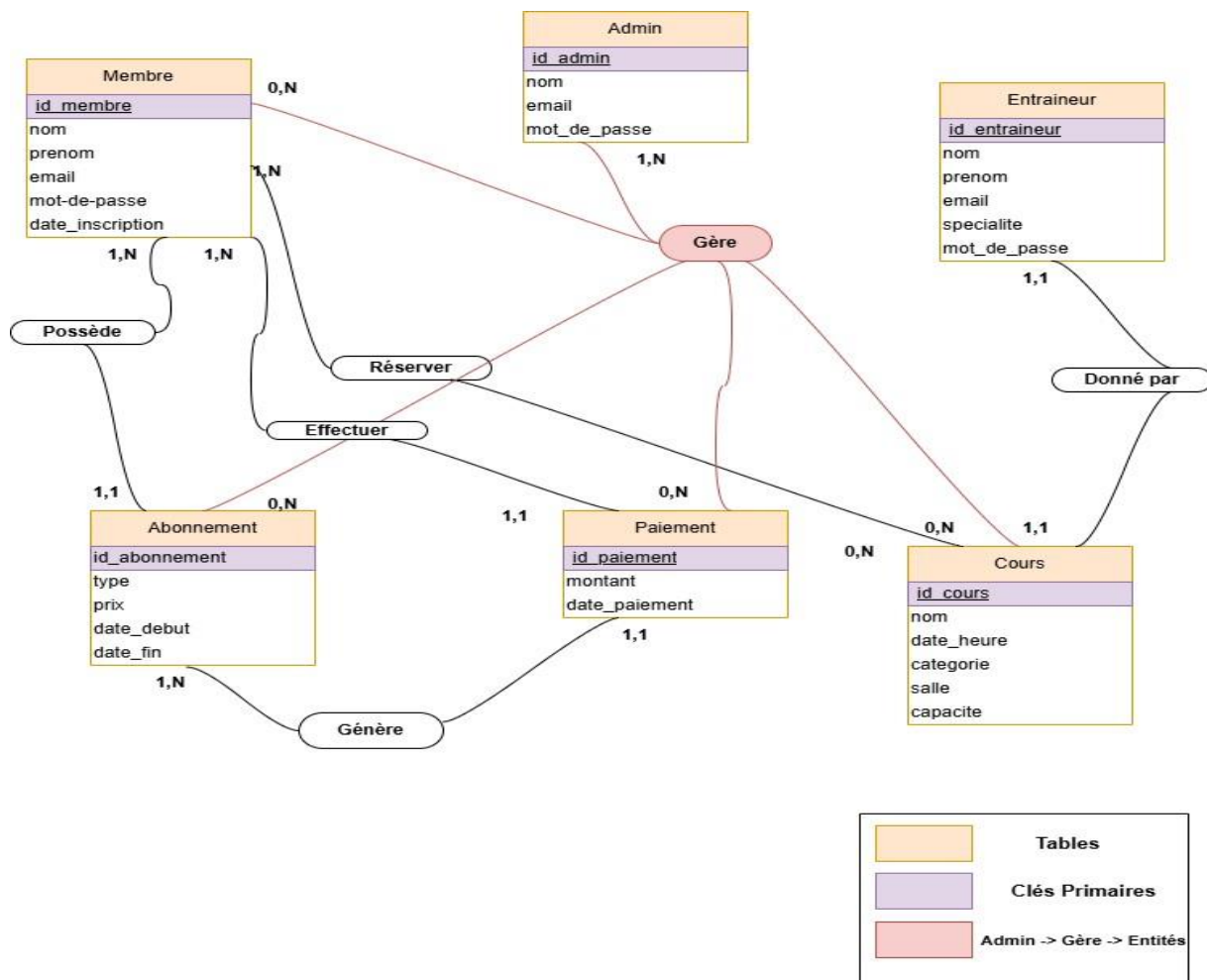
1. Membre choisit un cours
2. Interface envoie la requête au Système
3. Système vérifie connexion et disponibilité
4. Système enregistre réservation dans DB
5. DB confirme l'enregistrement
6. Système renvoie confirmation
7. Interface affiche "Réservation réussie"



4. Conception de la base de données

4.1 MCD (Modèle Conceptuel de Données)

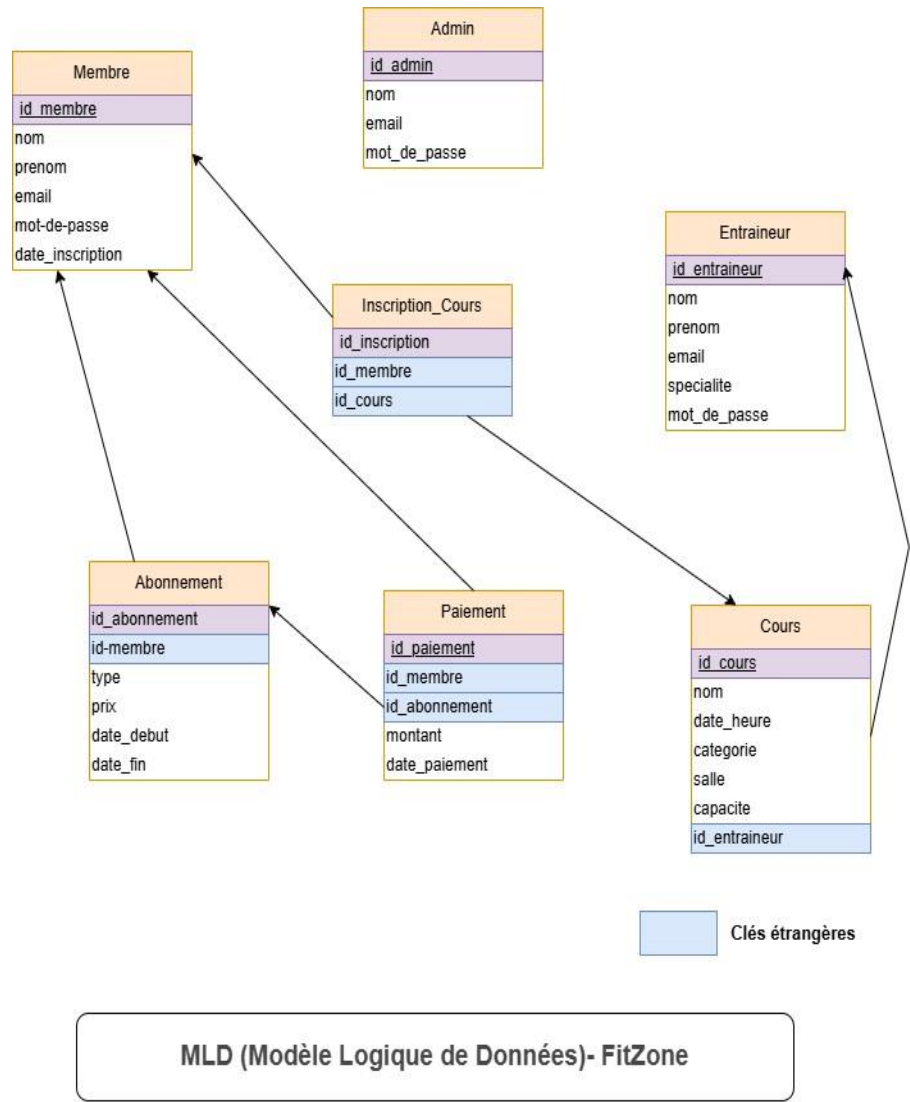
- Entités : Membre, Entraîneur, Cours, Abonnement, Paiement
- Relations principales : voir UML diagramme de classes



MCD (Modèle Conceptuel de Données)- FitZone

4.2 MLD (Modèle Logique de Données)

- Conversion du MCD en tables MySQL avec clés primaires et étrangères

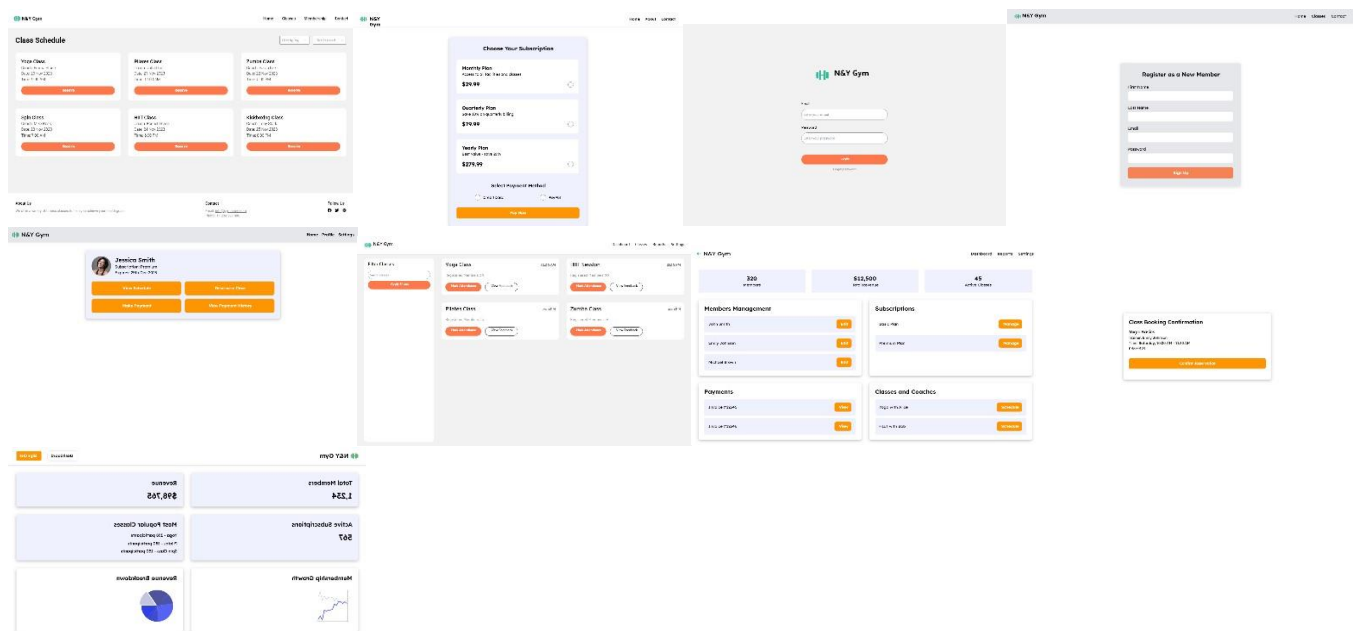


Exemple de tables :

Table	Champs principaux
Membre	id, nom, prénom, email, mot_de_passe ,date_inscription
Cours	id, nom, date, heure, id_entraineur,categorie,salle,capacité
Abonnement	id, type, date_debut, date_fin, id_membre,prix
Paieement	id, montant, date, id_abonnement,id_membre
Entraîneur	id, nom,prénom,email, spécialité
Insciprion_cours	Id,id_membre,id_cours

5. Maquettes / Prototypes

- Page de connexion : email + mot de passe + bouton “Se connecter”
- Tableau de bord membre : profil, abonnement, réservation, paiement
- Planning des cours : liste des cours, boutons réservation
- Tableau de bord entraîneur : planning et suivi des participants
- Tableau de bord administrateur : gestion membres, cours, paiements



6. Plan de tests (essentiel)

Fonctionnalité	Test	Résultat attendu
Création compte membre	Remplir formulaire inscription	Compte créé avec succès
Connexion	Email + mot de passe correct	Accès au tableau de bord
Réservation d'un cours	Cliquer "Réserver"	Confirmation affichée
Paieiment abonnement	Choisir abonnement + payer	Transaction réussie
Gestion membres (admin)	Ajouter / modifier / supprimer	Opérations réalisées

7. Conclusion

Ce dossier technique résume les choix techniques et l'architecture de FitZone, permettant aux développeurs de comprendre la structure, les interactions et les fonctionnalités principales avant le développement.