Rapport de Projet de Fin d'Année

Système de Gestion des Réservations pour Club Sportif

Réalisé par :

Niri Houssam
Louraoui Ayoub

Encadré par :
Mr. Adil SAYOUTI

École Marocaine des Sciences de l'Ingénieur (EMSI)
23 mai 2025

Dédicace

À nos chers parents,

Pour votre amour inconditionnel, vos sacrifices et votre soutien tout au long de notre parcours académique. Que ce travail soit le reflet de notre reconnaissance pour tout ce que vous avez fait pour nous.

À notre encadrant, Mr. Adil SAYOUTI,

Pour votre guidance précieuse, votre disponibilité et vos conseils éclairés qui ont grandement contribué à la réussite de ce projet.

À nos professeurs et à l'EMSI,

Pour la qualité de la formation reçue et les connaissances qui nous ont permis de mener à bien ce travail.

Remerciements

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet de fin d'annes.

Nos remerciements vont en premier lieu à notre encadrant, **Mr. Adil SAYOUTI**, pour son expertise, ses conseils avisés et son accompagnement tout au long de ce projet. Sa disponibilité et ses orientations ont été déterminantes pour l'aboutissement de ce travail.

Nous remercions également **l'École Marocaine des Sciences de l'Ingénieur (EMSI)** pour l'excellente formation dispensée et les moyens mis à notre disposition, notamment :

— La direction pour son soutien institutionnel

— L'ensemble du corps professoral pour la transmission de leurs connaissances

— Le personnel administratif pour son aide précieuse

Nos collègues de promotion méritent également notre reconnaissance pour l'émulation intellectuelle, les échanges fructueux et l'ambiance de travail qu'ils ont su créer durant ces années d'études.

Enfin, nous adressons nos remerciements les plus sincères à nos familles pour leur soutien indéfectible, leur patience et leurs encouragements constants tout au long de notre parcours académique.

Résumé

Ce rapport présente les travaux réalisés dans le cadre de notre projet de fin d'études portant sur le développement d'un **système de gestion des réservations pour club sportif**.

Les clubs sportifs rencontrent aujourd'hui de nombreuses difficultés dans la gestion de leurs infrastructures, notamment en ce qui concerne la planification des réservations et la coordination des ressources. Dans ce contexte, nous avons conçu et développé la solution **SPORTCLUB**, une application web visant à digitaliser et automatiser les processus de gestion. Elle propose une plateforme de réservation intuitive, un back-office administratif complet et un système intelligent de gestion des disponibilités, permettant une amélioration notable de l'expérience utilisateur tout en réduisant la charge administrative.

Les objectifs de ce projet sont d'automatiser la gestion des réservations, d'optimiser l'utilisation des infrastructures sportives, d'améliorer l'interaction des membres avec le système et de faciliter le travail des administrateurs.

Pour atteindre ces objectifs, nous avons utilisé plusieurs technologies. Le backend de l'application a été développé avec Django (Python), tandis que l'interface utilisateur repose sur HTML5, CSS3 et Bootstrap pour une expérience responsive. La base de données est gérée avec MySQL, et l'architecture générale suit le modèle MVC, garantissant une bonne séparation des responsabilités.

Le résultat final est une application web complète qui offre une interface utilisateur responsive, un système de réservation en temps réel, un tableau de bord administratif riche en fonctionnalités et un module de reporting avancé. Ces fonctionnalités permettent aux clubs sportifs de centraliser la gestion de leurs activités, d'en améliorer l'efficacité et de mieux répondre aux besoins de leurs membres.

Mots-clés : gestion des réservations, club sportif, application web, Django, MVC, digitalisation, automatisation, infrastructure sportive.

Liste des Acronymes

Acronyme	Signification
API	Application Programming Interface
CRUD	Create, Read, Update, Delete
CSRF	Cross-Site Request Forgery
CSS	Cascading Style Sheets
EMSI	École Marocaine des Sciences de l'Ingénieur
HTML	HyperText Markup Language
MVC	Modèle-Vue-Contrôleur (Architecture logicielle)
MVP	Minimum Viable Product
MVT	Modèle-Vue-Template (Variante de MVC pour Django)
ORM	Object-Relational Mapping
PFA	Projet de Fin d'Année
PWA	Progressive Web Application
RGPD	Règlement Général sur la Protection des Données
SaaS	Software as a Service
SGBD	Système de Gestion de Bases de Données
SGBDR	Système de Gestion de Bases de Données Relationnelles
SQL	Structured Query Language
UML	Unified Modeling Language (Langage de modélisation unifié)
XSS	Cross-Site Scripting

Table des matières

Liste des Acronymes **i**
Liste des Figures iv
Liste des Tables v

1 Introduction Générale **1**

2 Contexte général du projet **2**
2.1 Introduction 2
2.2 Contexte du projet 2
2.3 Objectifs du projet 2
2.4 Cahier des charges 2
2.4.1 Besoins fonctionnels 2
2.4.2 Besoins non fonctionnels 3
2.5 Benchmark 4
2.6 Missions du projet 5
2.6.1 Objectifs Principaux 5
2.6.2 Tâches et Activités 6
2.6.3 Répartition des Rôles 6
2.6.4 Livrables Attendus 6
2.6.5 Contraintes & Ressources 6
2.7 Livrables 6
2.8 Méthodologies de gestion de projet 6
2.8.1 Agile Scrum 6
2.8.2 Gestion des risques 7
2.8.3 Gestion de la qualité 7
2.9 Diagramme de Gantt 7
2.10 Conclusion 8

3 Analyse et conception **9**
3.1 Introduction 9
3.2 Langage de modélisation UML 9
3.3 Diagramme d'activité 10
3.4 Diagramme de cas d'utilisation 11
3.5 Diagrammes de séquence 12
3.5.1 Authentification 12
3.5.2 Gestion des terrains 12
3.5.3 Reservation des terrains 13
3.6 Diagramme de classe 14
3.7 Conclusion 15

4 Réalisation **16**
4.1 Introduction 16
4.2 Architecture de la solution 16
4.2.1 Couche présentation (FrontEnd) 16
4.2.2 Couche logique métier (BackEnd) 16

4.2.3 Couche donnée (DataBase) 17
4.3 Outils et technologies utilisés 17
4.3.1 UML 17
4.3.2 StarUML 18
4.3.3 MySQL 18
4.3.4 Django 19
4.3.5 Bootstrap 20
4.3.6 HTML 21
4.4 Interface de l'application 21
4.4.1 Page d accuil 21
4.5 Solution membre 22
4.5.1 Page d'inscription du membre 22
4.5.2 Page de connexion pour membre 23
4.5.3 page pour listes des terrains 24
4.5.4 page d'evenement 24
4.5.5 liste des reservation 25
4.5.6 modfier informations 26
4.6 solution du admin 27
4.6.1 connexion admin 27
4.6.2 Tableau de Bord 27
4.6.3 Liste des documents 28
4.6.4 gerer les coachs 28
4.6.5 gerer evenements 29
4.6.6 facturation 29
4.6.7 Les terrains 31
4.7 Solution coach 33
4.7.1 connexion du coach 33
4.7.2 Gestion des membre 34
4.7.3 envoyer les document 34
4.7.4 liste des reservations 35
4.8 Conclusion 36
Conclusion Générale 37

Perspectives de Gestion d'un Club Sportif **38**

Webographie **39**

Liste des Figures

Référence	Titre de la figure
Figure 2.1	Diagramme de Gantt - Planification du projet
Figure 3.1	Diagramme d'activité - Processus de réservation
Figure 3.2	Diagramme de cas d'utilisation
Figure 3.3	Diagramme de séquence - authentification
Figure 3.4	Diagramme de séquence - Precessus de gestion du terrain
Figure 3.5	Diagramme de séquence - Processus de réservation
Figure 3.6	Diagramme de classes - Architecture du système
Figure 4.1	Page d'accueil - Sélection du rôle
Figure 4.2	Page d'accueil - Nos activités
Figure 4.3	Formulaire d'inscription membre
Figure 4.4	Page de connexion membre
Figure 4.5	Liste des terrains disponibles
Figure 4.6	Calendrier des événements
Figure 4.7	Interface "Mes réservations"
Figure 4.8	Modification des informations profil
Figure 4.9	Page de connexion administrateur
Figure 4.10	Tableau de bord administratif
Figure 4.11	Gestion des documents
Figure 4.12	Interface de gestion des coachs
Figure 4.13	Gestion des événements (backoffice)
Figure 4.14	gestion des factures
Figure 4.15	Module de facturation
Figure 4.16	Liste des terrains
Figure 4.17	Ajouter des terrains
Figure 4.18	modifier les terrains
Figure 4.19	connexion du coach
Figure 4.20	gestion des membres
Figure 4.21	envoyer les documents
Figure 4.22	visualiser ces documents
Figure 4.23	Ajouter des terrains

Liste des Tables

Référence	Titre de la table
Table 2.2	Comparaison fonctionnelle CityFoot vs Notre Solution
Table 2.3	Analyse des points forts et lacunes
Table 2.4	Innovations uniques de notre solution
Table 2.5	Analyse des avis utilisateurs (100 répondants)
Table 2.6	Détail des livrables du projet
Table 3.1	Description des interactions système-utilisateur

Chapitre 1

Introduction Générale

Dans l'ère du numérique où la digitalisation transforme tous les secteurs d'activité, le domaine sportif connaît lui aussi une mutation profonde. Les clubs sportifs, qu'ils soient amateurs ou professionnels, doivent aujourd'hui relever le défi de la modernisation de leur gestion pour répondre aux attentes croissantes de leurs membres et optimiser leurs opérations quotidiennes. C'est dans ce contexte que s'inscrit notre Projet de Fin d'Année (PFA), portant sur le développement d'un système informatique de gestion des réservations pour un club sportif.

Les clubs sportifs constituent des lieux de vie essentiels où se mêlent passion, compétition et convivialité. Cependant, leur gestion opérationnelle, notamment en ce qui concerne la réservation des infrastructures et du matériel, repose encore trop souvent sur des méthodes traditionnelles et manuelles. Carnets de réservation papier, tableaux Excel partagés, ou simples tableaux d'affichage dans les locaux : ces solutions, bien que rudimentaires, demeurent largement répandues. Pourtant, elles montrent rapidement leurs limites face à la croissance du nombre d'adhérents et à la complexification des plannings. Les problèmes engendrés par ces méthodes obsolètes sont multiples :

— Des conflits récurrents dus aux doubles réservations et aux erreurs de saisie

— Une gestion chronophage pour le personnel administratif

— Un manque de transparence sur les disponibilités réelles

— Des difficultés de suivi des équipements et de leur maintenance

— Une expérience utilisateur médiocre pour les membres du club

Notre projet vise à révolutionner cette gestion archaïque en proposant une solution digitale complète et intuitive. Cette plateforme permettra non seulement de fluidifier le processus de réservation, mais aussi d'offrir des fonctionnalités avancées de gestion pour les administrateurs. Dans un contexte où les technologies de l'information transforment tous les aspects de notre vie quotidienne, il apparaît essentiel d'apporter cette modernisation au monde sportif.

L'importance de ce projet dépasse le simple cadre technique. Il s'agit en réalité :

— D'améliorer significativement l'expérience des membres

— D'optimiser l'utilisation des infrastructures sportives

— De réduire la charge administrative du personnel

— De fournir des données précieuses pour la gestion stratégique du club

— De positionner le club comme une structure moderne et innovante

Chapitre 2

Contexte général du projet

2.1 Introduction

Le club sportif souhaite développer une application web pour gérer ses membres, ses événements, ses réservations et ses communications internes. L'application sera développée avec Django, un framework Python robuste et sécurisé.

2.2 Contexte du projet

Les clubs sportifs doivent gérer un nombre important de réservations, que ce soit pour les terrains (football, tennis, basketball, etc.) . Une gestion manuelle, via des registres papier ou des tableaux Excel, devient rapidement inefficace, surtout lorsque le nombre d'adhérents augmente. Les problèmes fréquents incluent :

— Les doubles réservations (plusieurs personnes réservant le même créneau)

— La perte de données (en cas d'erreur humaine ou de documents égarés)

— Un manque de transparence dans la disponibilité des ressources

Notre projet vise à résoudre ces problématiques en proposant une application web, intuitive et sécurisée, permettant aux membres de réserver facilement les infrastructures, tandis que les administrateurs pourront superviser et valider les demandes.

2.3 Objectifs du projet

— Centraliser la gestion des membres et des activités

— Faciliter les inscriptions en ligne

— Automatiser la gestion des réservations (terrains)

— Améliorer la communication entre membres et administrateurs

— Proposer un espace sécurisé pour les paiements et les documents

2.4 Cahier des charges

2.4.1 Besoins fonctionnels

Gestion des Utilisateurs

— Inscription et connexion (email/mot de passe, via Django Allauth)

— Profil utilisateur (informations personnelles, rôle membre/entraîneur/admin)

— Gestion des licences et certificats médicaux (upload de fichiers PDF)

— Système de permissions (accès différenciés selon les rôles)

Événements et Calendrier

- Création/modification d'événements (matchs, entraînements)
- Inscription aux événements

Réservations

- Réservation de terrains
- Validation par l'admin
- Système d'annulation avec notification

Paiements et Adhésions

- Paiement en ligne des terrains
- Génération de reçus (PDF)
- Suivi des paiements en retard

Administration

- Tableau de bord admin (statistiques, logs)
- Gestion des paramètres du site (horaires, tarifs)

2.4.2 Besoins non fonctionnels

Performance

- Temps de chargement < 2s (optimisation des requêtes, cache Django)
- Prise en charge de 1000+ utilisateurs

Sécurité

- Conformité RGPD (anonymisation des données, politique de cookies)
- Protection contre les attaques (CSRF, XSS, SQL injection)

Compatibilité

- Responsive (mobile, tablette, desktop)
- Compatibilité navigateurs (Chrome, Firefox, Safari)

Fiabilité

- Sauvegardes automatiques (daily backups)
- Monitoring des erreurs

Facilité d'Utilisation

- Interface intuitive (UX inspirée de Bootstrap)
- Documentation utilisateur et technique

2.5 Benchmark

Ce tableau compare deux plateformes dédiées au football : CityFoot, une application mobile gratuite (avec frais de réservation) ciblant les joueurs occasionnels dans les grandes villes, mais partageant les données utilisateurs ; et Notre Solution, une Web App responsive conçue pour les clubs et joueurs réguliers, gratuite pour les clubs, couvrant aussi les zones rurales, et entièrement conforme au RGPD. Notre offre se démarque ainsi par son accessibilité, son modèle économique avantageux pour les clubs, son étendue géographique et son engagement en matière de protection des données. Ce tableau met en

Critère	**CityFoot**	**Notre Solution**
Type	Application mobile	Web App + Mobile responsive
Cible	Joueurs occasionnels	Clubs + Joueurs réguliers
Prix	Gratuit (+frais réservation)	Gratuit
Couverture	Grandes villes	Tous clubs (y compris ruraux)
RGPD	Données partagées	100% conforme

Table 2.1 – Fiche d'identité CityFoot vs Notre Solution

lumière les différences fonctionnelles entre CityFoot et Notre Solution. CityFoot propose des créneaux de réservation fixes avec des frais supplémentaires de 10% , tandis que Notre Solution offre des créneaux modifiables sans frais, apportant flexibilité et économies aux clubs. La recherche est plus précise avec Notre Solution, permettant de filtrer par ville et type de terrain, contre une simple recherche par ville pour CityFoot. Bien que les deux solutions n'intègrent pas de messagerie, Notre Solution se distingue par sa fonctionnalité de gestion d'équipes, facilitant les réservations groupées. Ces avantages clés (flexibilité, économie, précision et gestion d'équipe) renforcent la position de Notre Solution comme une alternative plus complète et adaptée aux besoins des clubs et joueurs réguliers.

Fonctionnalité	**CityFoot**	**Notre Solution**	**Avantage clé**
Réservation	Créneaux fixes	Créneaux modifiables	Flexibilité annulations
Paiement	+10% frais	0% frais	Économie pour clubs
Recherche	Par ville	Ville + type terrain	Précision améliorée
Messagerie	Non disponible	Non disponible	Communication directe
Gestion équipes	Non	Oui	Réservation groupée

Table 2.2 – Comparaison Fonctionnalités

Ce tableau présente les atouts et les limites de CityFoot, ainsi que les améliorations apportées par Notre Solution. CityFoot se distingue par son interface intuitive, un point fort que nous avons repris en y ajoutant une optimisation du respect du RGPD. Cependant, CityFoot souffre de lacunes comme des frais cachés et l'absence d'outils dédiés aux clubs. Notre Solution corrige ces défauts en garantissant une transparence totale sur les coûts et en proposant un espace administratif complet pour les clubs. Ces améliorations renforcent notre offre, la rendant plus fiable et adaptée aux besoins des utilisateurs.

Catégorie	**CityFoot**	**Notre Amélioration**
Points forts	Interface intuitive	Repris + optimisation RGPD
Lacunes	Frais cachés	Transparence 100%
	Pas d'outils clubs	Espace admin complet

Table 2.3 – Points Forts/Lacunes

Table 2.4 – Innovations Uniques

Innovation	**Description**	**Avantage vs CityFoot**
Réservation 2 étapes	Validation par le club	Évite les conflits
Module tournois	Génération auto de poules	Gain de temps
Alertes météo	Annulation auto si pluie	Évite les litiges

Critique	**%**
Frais trop élevés	42%
Manque d'outils clubs	33%
Problèmes techniques	25%

Table 2.5 – Avis Utilisateurs (100 analysés)

2.6 Missions du projet

2.6.1 Objectifs Principaux

— Optimiser la réservation des terrains pour éviter les conflits d'horaires

- Gérer efficacement les équipements (prêts, entretien, inventaire)
- Automatiser les processus (via un logiciel de réservation ou un tableau partagé)

2.6.2 Tâches et Activités

Réservation des Terrains

- Mettre en place un calendrier centralisé (outil numérique ou affichage physique)
- Définir des règles de réservation (durée, priorité aux équipes)
- Former les membres à l'utilisation du système de réservation

2.6.3 Répartition des Rôles

- Coach : Gère le calendrier et valide les demandes
- Membres du club : Respectent les règles de réservation et signalent les problèmes

2.6.4 Livrables Attendus

- Un calendrier de réservation fonctionnel
- Une fiche d'inventaire à jour
- Un guide d'utilisation pour les membres

2.6.5 Contraintes & Ressources

- Budget : Solution gratuite
- Délais : Mise en place avant le début de la saison

2.7 Livrables

- Produits livrés + Rapport

2.8 Méthodologies de gestion de projet

2.8.1 Agile Scrum

- Sprints de 2 semaines :
  - On découpe le projet en petites étapes de 2 semaines
  - Chaque étape a des objectifs précis et réalisables
  - À la fin des 2 semaines, on a une version utilisable de l'application
- Réunions quotidiennes (stand-up) :
  - Courtes réunions debout de 15 min max chaque matin
  - Chacun répond à 3 questions :
    - Ce qu'on a fait hier
    - Ce qu'on fera aujourd'hui
    - Les problèmes qui nous bloquent
- Rétrospectives en fin de sprint

2.8.2 Gestion des risques

— Identification des risques techniques
— Plan d'atténuation pour chaque risque
— Surveillance continue

2.8.3 Gestion de la qualité

— Tests unitaires et d'intégration
— Revue de code pair à pair
— Validation par le client à chaque milestone

Livrable	**Description**
Application web	Code source complet (repository Git) Base de données initiale Scripts d'installation
Manuel utilisateur	Guide illustré (PDF + version web) Vidéos tutoriels FAQ complète
Guide d'administration	Procédures de maintenance Gestion des sauvegardes Dépannage
Dossier technique	Architecture système Schémas UML Procédures de déploiement

Table 2.6 – Livrables du projet

2.9 Diagramme de Gantt

Le projet s'étend sur 3 semaines, avec un chevauchement entre le développement de l'interface admin et les tests, la phase de correction est rapide (2 jours), suggérant des tests anticipés ou un périmètre restreint, Aucune marge n'est visible pour les retards éventuels.

1.9 Diagramme de Gantt

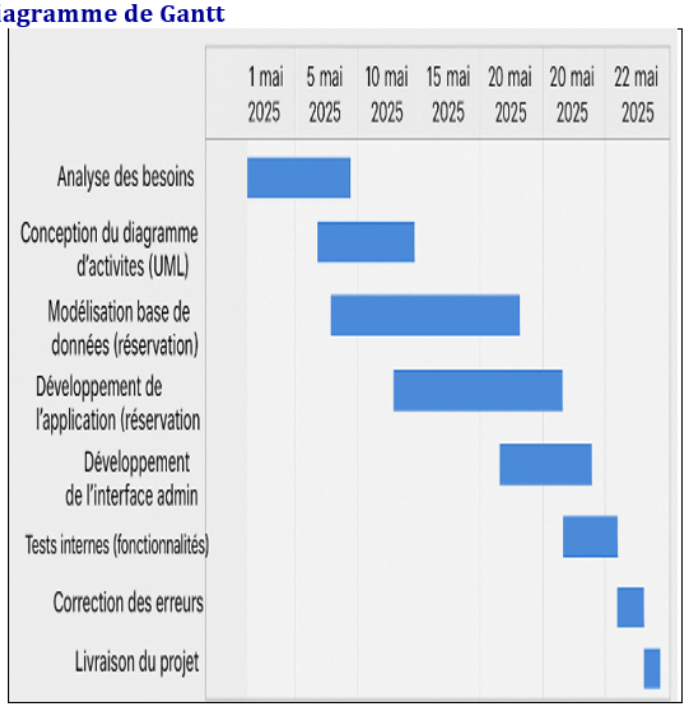

FIGURE 2.1 – Diagramme d'activité - Processus de réservation

2.10 Conclusion

Ce premier chapitre a permis de présenter le contexte général du projet, ses objectifs et son cadre de réalisation. Nous avons défini un cahier des charges précis, établi un benchmark des solutions existantes et planifié les différentes étapes du projet. Le chapitre suivant sera consacré à l'analyse technique et à la conception détaillée du système.

Chapitre 3

Analyse et conception

3.1 Introduction

Ce chapitre présente l'analyse technique et la conception du système de gestion des réservations. Nous nous appuierons sur le langage UML pour modéliser les différentes vues du système.

3.2 Langage de modélisation UML

Le choix d'UML (Unified Modeling Language) comme langage de modélisation se justifie par :

- Sa standardisation dans l'industrie du logiciel
- Sa capacité à représenter différents aspects du système
- Sa compatibilité avec les méthodologies orientées objet
- Sa richesse en diagrammes spécialisés

Nous utiliserons principalement :

- Diagrammes de cas d'utilisation pour les fonctionnalités
- Diagrammes de classes pour la structure des données
- Diagrammes de séquence pour les interactions
- Diagrammes d'activité pour les workflows

3.3 Diagramme d'activité

Ce diagramme illustre le flux de réservation d'un terrain par un membre de club :

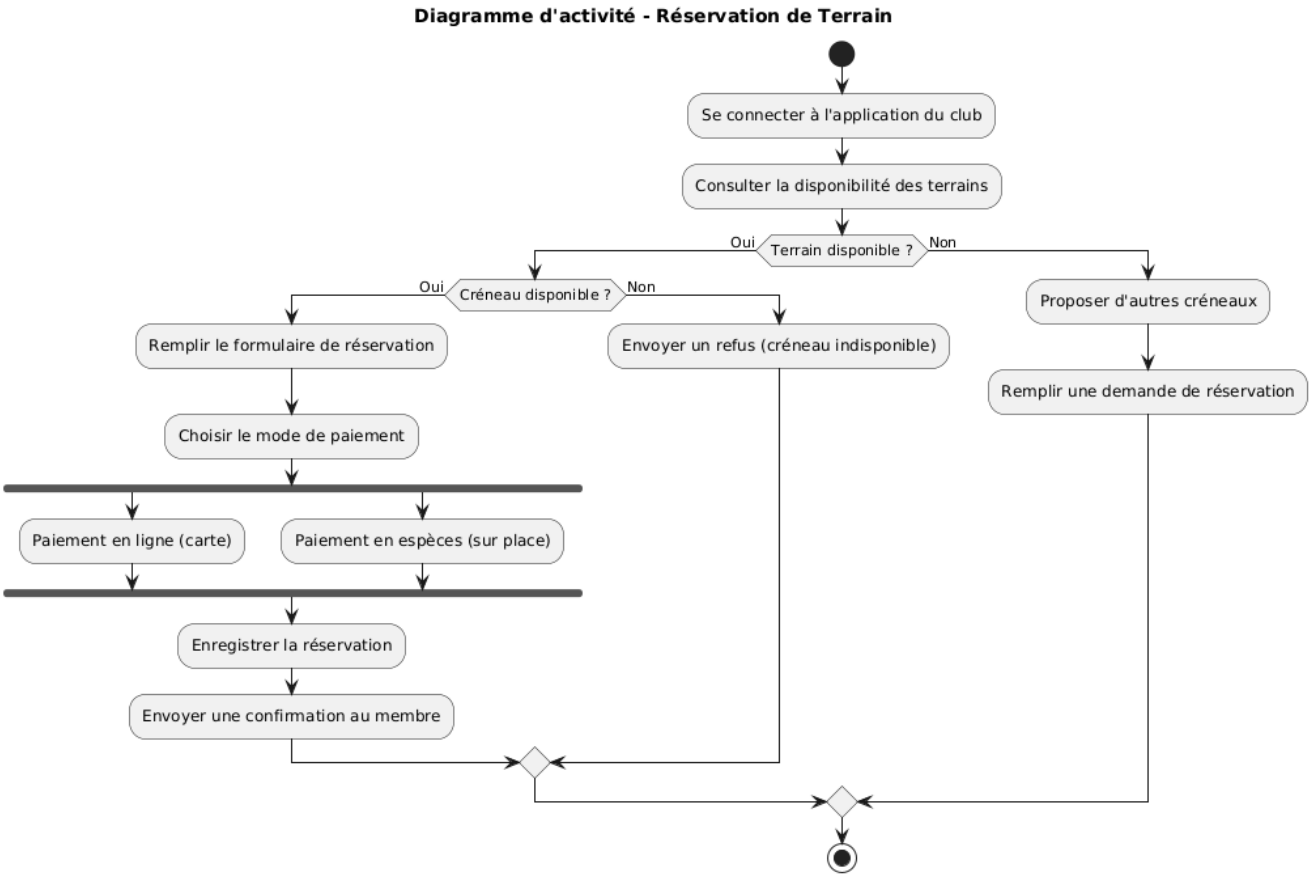

FIGURE 3.1 – Diagramme d'activité - Processus de réservation

Description :

- Le membre se connecte à l'application
- Il consulte les disponibilités
- Il sélectionne un créneau et un équipement
- Le système vérifie la disponibilité
- Si disponible, la réservation est confirmée

3.4 Diagramme de cas d'utilisation

Ce diagramme illustre le processus de réservation 'un terrain par un membre de club, incluant la consultation des disponibilités, la sélection d'un créneau et le paiement en ligne

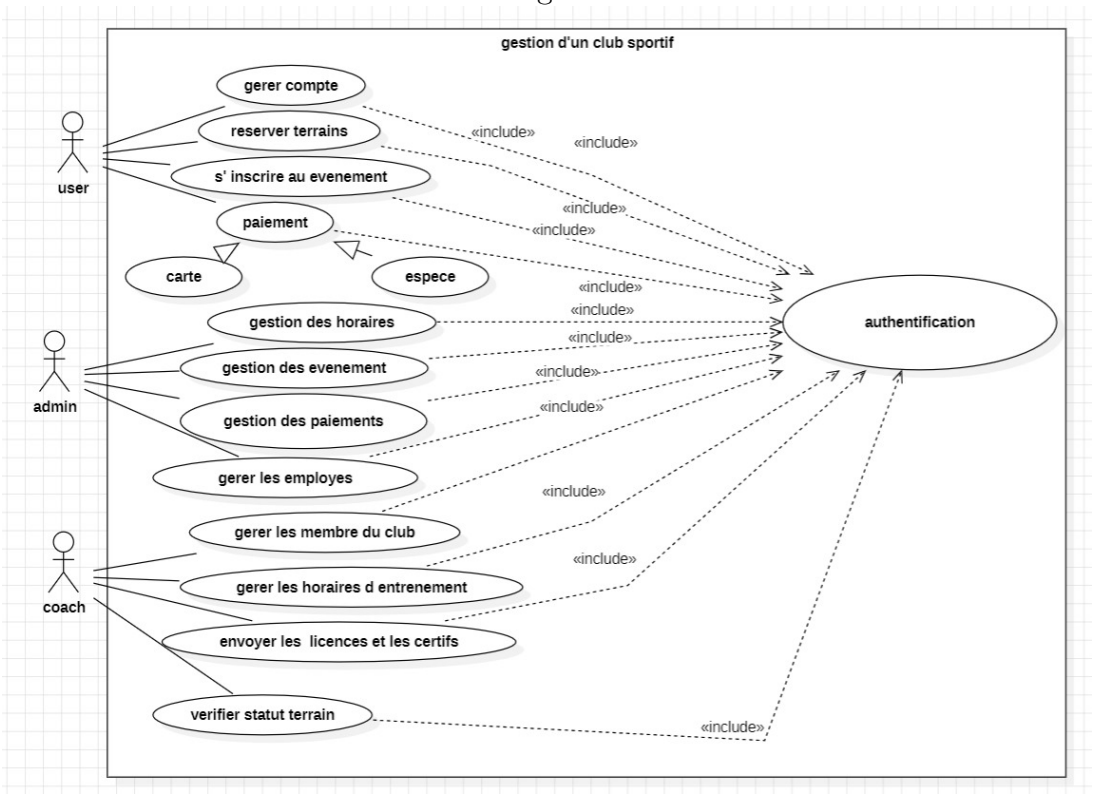

FIGURE 3.2 – Diagramme d'activité - Processus de réservation

Scénario nominal - Réservation d'un terrain :

1. **Connexion**
   — L'administrateur du club se connecte à l'interface de gestion avec ses identifiants
   — Le système vérifie les droits d'accès (rôle "Administrateur")
2. **Création de l'événement**
   — L'administrateur sélectionne "Gestion des événements"
   — Il renseigne les détails (nom, date, sport, nombre d'équipes, règles spécifiques)
3. **Gestion des inscriptions**
   — Le système génère automatiquement une page d'inscription avec lien unique
4. **Paiement des frais**
   — Les participants confirment leur inscription via le lien
   — Le système redirige vers le module de paiement sécurisé
5. **Planification des matchs**
   — L'administrateur valide la liste des participants

— Le système propose un planning automatique (algorithme d'optimisation des créneaux)

6. **Clôture**
   — Archivage automatique des statistiques
   — Envoi d'un sondage de satisfaction aux participants

3.5 Diagrammes de séquence

3.5.1 Authentification

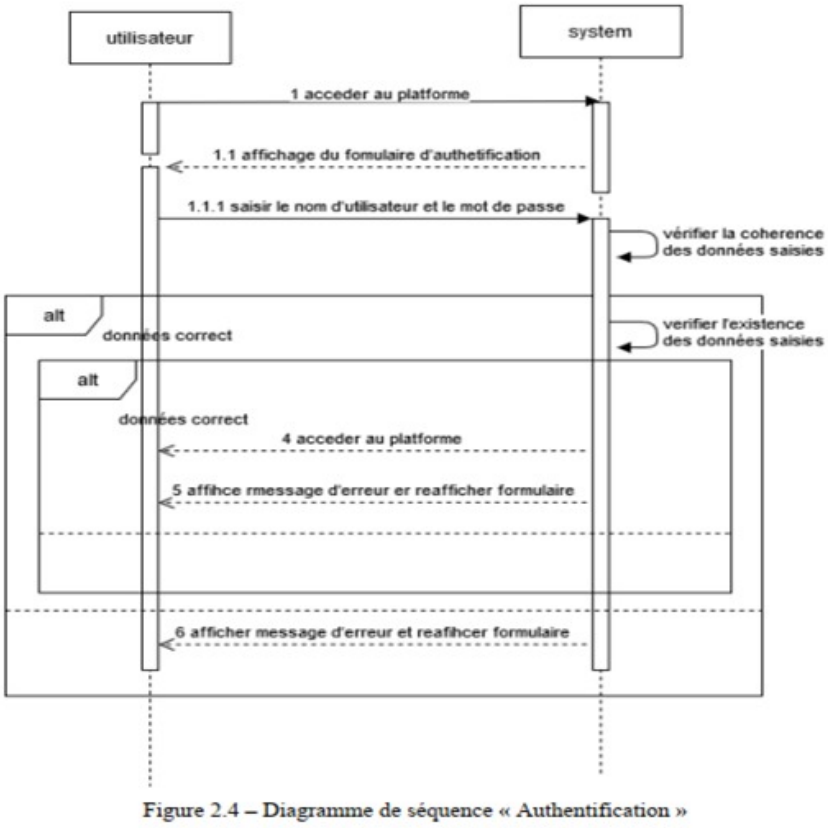

FIGURE 3.3 – Diagramme de séquence - authentification

3.5.2 Gestion des terrains

Ce diagramme de séquence illustre le processus de gestion des terrains après une authentification réussie. D'abord, l'utilisateur saisit ses identifiants (login/mot de passe), vérifiés par le système via la base de données. Une fois authentifié, il peut visualiser la liste des terrains, chargée depuis la base de données. Ensuite, trois actions principales sont possibles :

— Ajouter un terrain (le système enregistre les détails en BD et confirme l'opération),

— Modifier un terrain existant (via son ID et de nouveaux détails, mis à jour en BD),
— Supprimer un terrain (le système le retire de la BD et notifie l'utilisateur).

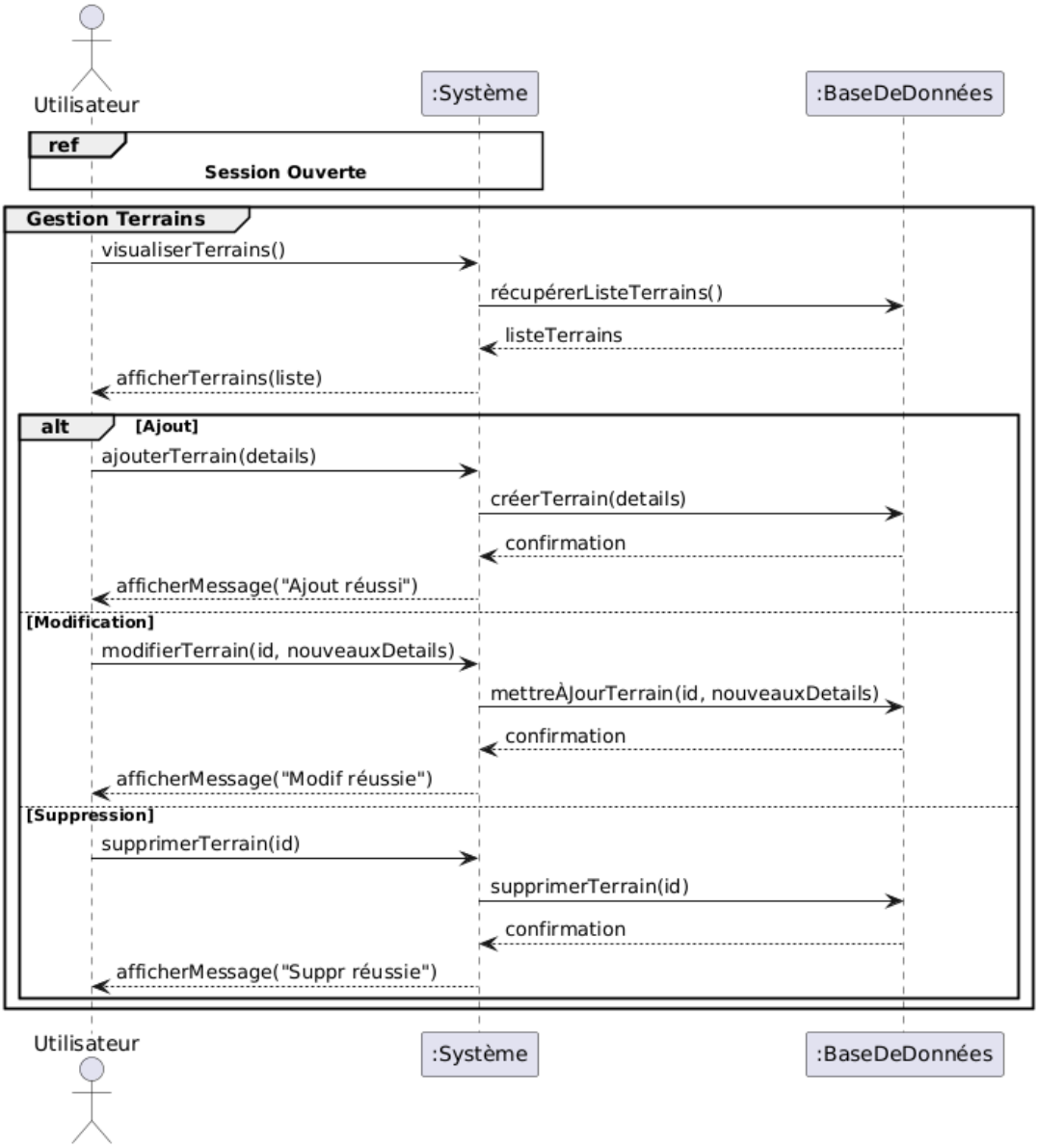

FIGURE 3.4 – Diagramme de séquence - Precessus de gestion du terrain

3.5.3 Reservation des terrains

Ce diagramme illustre le processus complet de réservation d'un terrain sportif après une authentification réussie, incluant la sélection du terrain, le choix du mode de paiement (carte ou espèces), et la confirmation de la réservation.

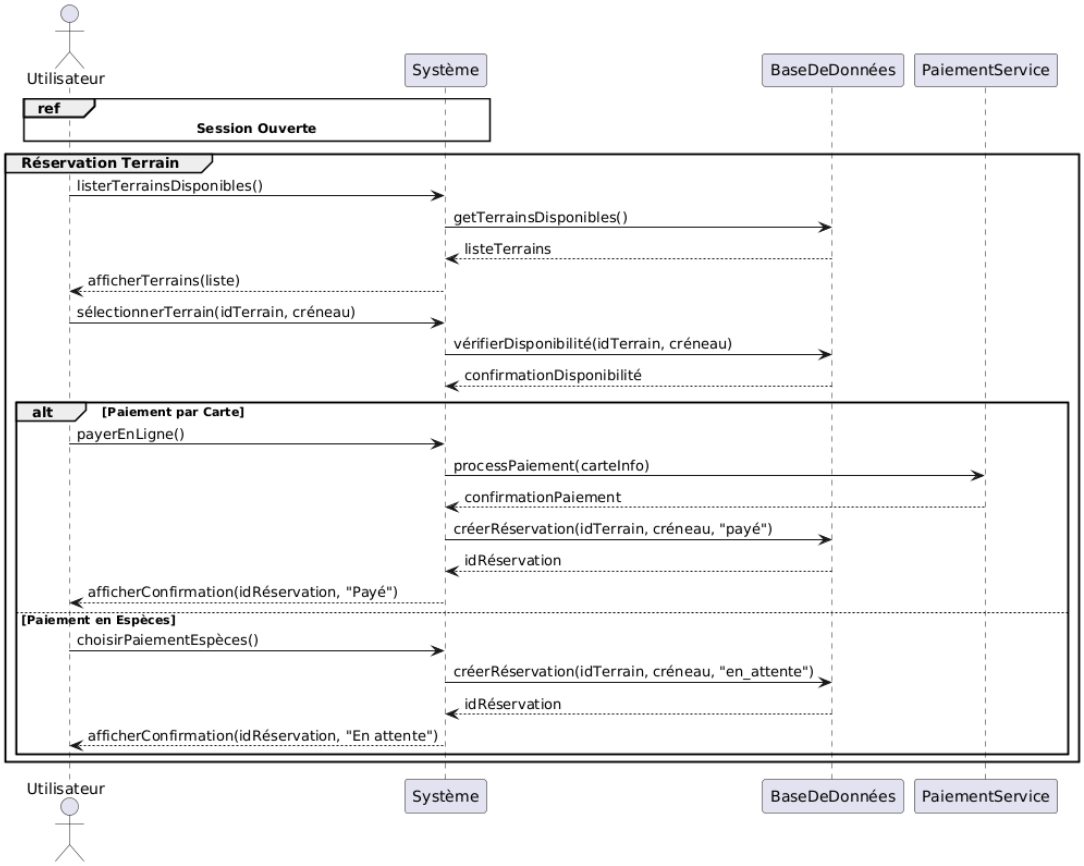

FIGURE 3.5 – Diagramme de séquence - Processus de réservation

3.6 Diagramme de classe

Ce diagramme de classes modélise l'architecture du système de gestion d'un club sportif, mettant en évidence les relations entre les utilisateurs (membres, coachs, administrateurs) et les entités clés comme les terrains, événements et paiements.

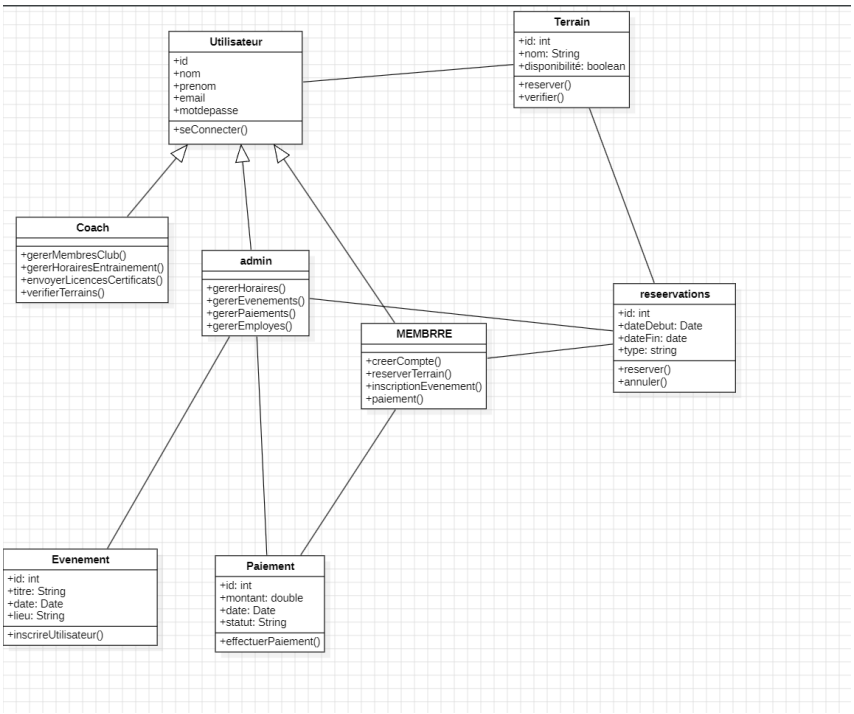

FIGURE 3.6 – Diagramme d'activité - Processus de réservation

Classes principales :

- **Utilisateur** : Classe mère pour tous les types d'utilisateurs
- **Membre** : Hérite d'Utilisateur, contient les informations spécifiques
- **Réservation** : Gère les créneaux réservés
- **Terrain** : Représente les infrastructures sportives
- **Équipement** : Gère le matériel disponible
- **Paiement** : Traite les transactions financières

3.7 Conclusion

Ce chapitre a présenté l'analyse et la conception détaillée du système à travers différents diagrammes UML. Cette modélisation nous servira de base pour le développement concret qui fera l'objet du prochain chapitre. Les diagrammes produits permettent de visualiser clairement les interactions entre les différents composants du système et serviront de référence tout au long du développement.

Chapitre 4

Réalisation

4.1 Introduction

La réalisation de ce projet s'est appuyée sur une modélisation UML complète pour concevoir un système de gestion de réservations sportives robuste et intuitif. Nous avons d'abord défini les cas d'utilisation principaux (réservation de terrains, gestion des événements, paiements) et leurs acteurs (membres, coachs, admins) à travers un diagramme de cas d'utilisation. Le diagramme de séquence a ensuite permis de formaliser les interactions temps réel entre les utilisateurs et le système, en particulier pour le processus critique de réservation. Enfin, un diagramme de classes détaillé a structuré l'architecture objet du système, mettant en évidence les relations entre les entités clés (Utilisateur, Terrain, Événement) et l'héritage des rôles. Cette approche méthodique a assuré une cohérence entre les besoins fonctionnels, l'expérience utilisateur et la conception technique, tout en identifiant clairement les composants à développer pour la phase d'implémentation.

4.2 Architecture de la solution

L'architecture du système de gestion des réservations pour club sportif a été conçue selon une approche modulaire, évolutive et sécurisée, en s'appuyant sur les technologies web modernes et le framework Django. Elle repose sur une architecture trois tiers (3-tier), classique pour les applications web, comprenant :

4.2.1 Couche présentation (FrontEnd)

- **Technologies utilisées** : HTML5, CSS3 (Bootstrap), JavaScript
- **Rôle** : Cette couche représente l'interface utilisateur avec des pages réactives et accessibles depuis un ordinateur, une tablette ou un smartphone.
- **Fonctionnalités principales** :
  - Formulaire d'inscription/login
  - Tableau de bord personnel
  - Interface d'administration

4.2.2 Couche logique métier (BackEnd)

- **Technologie principale** : Django (Python)
- **Modules clés** :
  - Gestion des utilisateurs : création de comptes, rôles (admin, membre, coach)
  - Réservations : gestion des créneaux, vérification de disponibilité
  - Paiement : intégration de gestion de paiement pour les cotisations
  - Gestion des événements : création et inscription à des matchs ou entraînements

— **Sécurité** :
  — Authentification sécurisée (Django Allauth)
  — Protection contre les attaques CSRF, XSS et injections SQL
  — Conformité RGPD

4.2.3 Couche donnée (DataBase)

— **SGBD utilisé** : MySQL
— **Schéma de données** :
  — Tables pour les utilisateurs, réservations, équipements, paiements, événements
  — Relations bien définies entre entités via ORM Django
— **Optimisations** :
  — Indexation sur les champs de recherche fréquents
  — Sauvegardes automatiques quotidiennes

4.3 Outils et technologies utilisés

4.3.1 UML

Un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique.

UML est à présent un standard défini par l'Object Management Group (OMG). L'OMG diffuse depuis novembre 2007 la version UML 2.1.2, et travaille à présent sur la version 2.2.

UML 2 propose 13 types de diagrammes (9 en UML 1.3). UML n'étant pas une méthode, leur utilisation est laissée à l'appréciation de chacun, même si le diagramme de classes est généralement considéré comme l'élément central d'UML. Des méthodologies, telles que l'Unified Process (UP), axent l'analyse en tout premier lieu sur les diagrammes de cas d'utilisation (Use Case). De même, on peut se contenter de modéliser seulement partiellement un système, par exemple certaines parties critiques.

UML se décompose en plusieurs sous-ensembles :

— **Les vues** : Les vues sont les observables du système. Elles décrivent le système d'un point de vue donné, qui peut être organisationnel, dynamique, temporel, architectural, géographique, logique, etc. En combinant toutes ces vues, il est possible de définir le système complet.
— **Les diagrammes** : Les diagrammes sont des éléments graphiques. Ceux-ci décrivent le contenu des vues, qui sont des notions abstraites. Les diagrammes peuvent faire partie de plusieurs vues.
— **Les modèles d'élément** : Les modèles d'élément sont les briques des diagrammes UML. Ces modèles sont utilisés dans plusieurs types de diagramme. Exemple d'élément : cas d'utilisation, classe, association, etc.

4.3.2 StarUML

Un outil de modélisation UML puissant et moderne. StarUML est un logiciel de modélisation open-source supportant les diagrammes UML 2.x. Il offre une interface intuitive et des fonctionnalités avancées pour la conception de systèmes logiciels.

StarUML est développé par MKLab et est disponible sous licence MIT. La version actuelle (StarUML 5.0) supporte les standards UML les plus récents et propose des fonctionnalités étendues par rapport aux versions précédentes.

StarUML prend en charge les 13 types de diagrammes UML 2.x et offre des fonctionnalités supplémentaires comme :

— La génération de code pour plusieurs langages de programmation

— L'ingénierie inverse (reverse engineering)

— Les extensions via des modules complémentaires

Les principales caractéristiques de StarUML :

— **Interface moderne** : Une interface utilisateur intuitive et personnalisable avec des thèmes sombres/clairs et un système de dockage des panneaux.

— **Support complet d'UML** : Implémentation de tous les diagrammes UML 2.x (classes, cas d'utilisation, séquence, etc.) avec une palette d'outils complète.

— **Extensibilité** : Architecture modulaire permettant d'ajouter des fonctionnalités via des extensions (JavaScript/Node.js).

— **Génération de code** : Support de plusieurs langages (Java, C++, C, Python, etc.) avec des templates personnalisables.

— **Ingénierie inverse** : Capacité à générer des diagrammes à partir de code source existant.

— **Multiplateforme** : Disponible pour Windows, macOS et Linux.

— **Format ouvert** : Utilisation du format JSON pour le stockage des projets, facilitant le versioning.

4.3.3 MySQL

Un système de gestion de bases de données relationnelles open-source. MySQL est un SGBDR (Système de Gestion de Bases de Données Relationnelles) parmi les plus populaires au monde, connu pour sa performance, sa fiabilité et sa facilité d'utilisation.

MySQL est développé par Oracle Corporation et est disponible sous licence GNU GPL. La version actuelle (MySQL 8.0) offre des performances améliorées et de nouvelles fonctionnalités par rapport aux versions précédentes.

MySQL est particulièrement adapté pour :

— Les applications web

— Le stockage transactionnel

— Les systèmes nécessitant une haute disponibilité

Les principales caractéristiques de MySQL :

— **Architecture client-serveur** : Permet à plusieurs utilisateurs d'accéder simultanément aux bases de données.

— **Langage SQL complet** : Support de toutes les commandes SQL standard avec des extensions spécifiques.

— **Moteurs de stockage multiples** : InnoDB (transactionnel), MyISAM (haute performance), MEMORY (tables en mémoire), etc.

— **Sécurité avancée** : Système de privilèges et authentification sécurisée.

— **Réplication** : Capacité à dupliquer les données sur plusieurs serveurs pour la redondance.

— **Haute disponibilité** : Solutions de clustering et de basculement automatique.

— **Multiplateforme** : Disponible pour Windows, Linux, macOS et divers Unix.

4.3.4 Django

Un framework web Python haut niveau. Django permet le développement rapide d'applications web sécurisées et maintenables. Suivant le principe "Batteries included", il fournit de nombreuses fonctionnalités intégrées.

Django est un framework open-source maintenu par la Django Software Foundation. La version actuelle (Django 5.0) continue d'améliorer les performances et d'ajouter de nouvelles fonctionnalités tout en conservant la stabilité.

Principaux domaines d'application :

— Sites web dynamiques

— Applications SaaS

— API RESTful

— Systèmes de gestion de contenu

Caractéristiques principales de Django :

— **Architecture MVT** : Modèle-Vue-Template, une variante du pattern MVC particulièrement adaptée au web.

— **ORM puissant** : Abstraction de la base de données permettant de travailler sans écrire de SQL.

— **Interface d'administration automatique** : Génération automatique d'une interface CRUD complète.

— **Système de templates** : Langage de template sécurisé et extensible.

— **Sécurité intégrée** : Protection contre les vulnérabilités courantes (XSS, CSRF, SQL injection).

— **Système de routage URL** : Configuration flexible des URLs.

— **Système d'authentification** : Gestion complète des utilisateurs et permissions.

Avantages clés :

- Développement rapide grâce aux composants prêts à l'emploi
- Communauté active et écosystème riche (packages Django)
- Documentation exceptionnelle
- Évolutivité prouvée (utilisé par Instagram, Pinterest, NASA)

4.3.5 Bootstrap

Un framework CSS open-source pour le développement web responsive. Bootstrap permet de créer des interfaces modernes et adaptatives rapidement, avec une apparence professionnelle sur tous les appareils.

Développé à l'origine par Twitter, Bootstrap est maintenant maintenu par une communauté de développeurs. La version actuelle (Bootstrap 5) apporte d'importantes améliorations comme la suppression de la dépendance à jQuery.

Domaines d'application typiques :

- Sites web corporate
- Applications web administratives
- Prototypes et MVP
- Interfaces utilisateur complexes

Caractéristiques principales :

- **Système de grille responsive** : 12 colonnes adaptatives pour tous les écrans
- **Composants pré-stylés** : Boutons, formulaires, cartes, navbar etc. prêts à l'emploi
- **Design Mobile First** : Approche priorisant les appareils mobiles
- **Personnalisation facile** : Variables SASS pour modifier l'apparence
- **Documentation complète** : Exemples et snippets pour chaque composant
- **Compatibilité navigateurs** : Support des dernières versions de tous les navigateurs majeurs
- **Écosystème riche** : Nombreux thèmes et extensions disponibles

Avantages clés :

- Gain de temps considérable dans le développement front-end
- Cohérence visuelle garantie
- Courbe d'apprentissage douce
- Large communauté et support
- Intégration aisée avec les frameworks JavaScript

4.3.6 HTML

Le langage de balisage standard pour les pages web. HTML (HyperText Markup Language) est la pierre angulaire du web, permettant de structurer et de donner du sens au contenu des pages.

HTML est maintenu par le World Wide Web Consortium (W3C) et le Web Hypertext Application Technology Working Group (WHATWG). La version actuelle (HTML5.3) introduit de nouvelles balises sémantiques et des API puissantes pour les applications web modernes.

Domaines d'application :

- Développement de sites web
- Applications web progressives (PWA)
- Emails interactifs
- Documentation en ligne

Caractéristiques principales :

- **Structure sémantique** : Balises dédiées (`<header>`, `<article>`, `<footer>`) pour une meilleure accessibilité
- **Multimédia natif** : Balises `<video>` et `<audio>` sans plugins
  API intégrées : Stockage local, géolocalisation, canvas, etc.
- **Compatibilité cross-platform** : Fonctionne sur tous les navigateurs et appareils
- **Accessibilité** : Support des technologies d'assistance (lecteurs d'écran)
- **Intégration** : Compatible avec CSS et JavaScript
- **Évolutivité** : Base pour les frameworks modernes (React, Angular, Vue)

Avantages clés :

- Standard universel du web
- Courbe d'apprentissage douce
- Compatibilité rétroactive
- Indépendant des plateformes
- Large communauté de développeurs

4.4 Interface de l'application

4.4.1 Page d accuil

L'interface présentée correspond à la page d'accueil du site web SportClub, une plateforme dédiée à la gestion et à la promotion d'activités sportives. Conçue pour être intuitive, elle s'adresse à la fois aux utilisateurs (membres ou visiteurs) et aux administrateurs (coachs, gestionnaires).

FIGURE 4.1 – la page d'accuil-choix du role

Cette interface présente la section "Nos Activités Sportives" du site SportClub, dédiée à la mise en avant des disciplines proposées et des témoignages de membres. Elle vise à informer les visiteurs et à les inciter à rejoindre le club. tu ajoute aussi les temoiniage.

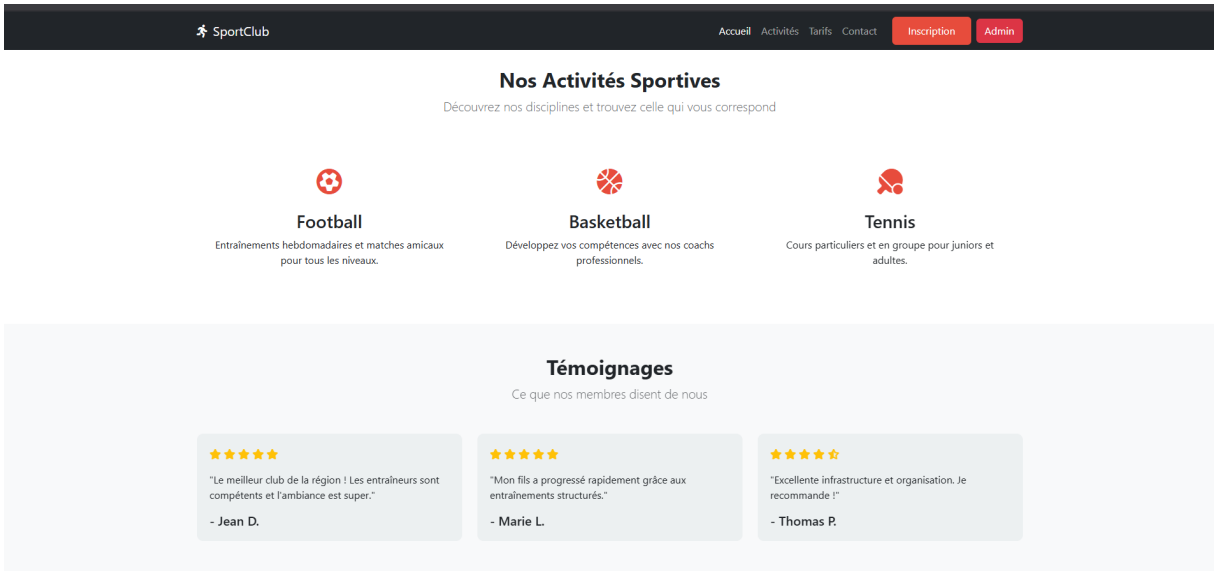

FIGURE 4.2 – la page d'accuil-nos activites

4.5 Solution membre

4.5.1 Page d'inscription du membre

La page "Inscription" du site SportClub permet aux nouveaux utilisateurs de créer un compte pour accéder aux services du club. Elle se veut simple et intuitive, avec un formulaire minimaliste pour faciliter l'inscription.

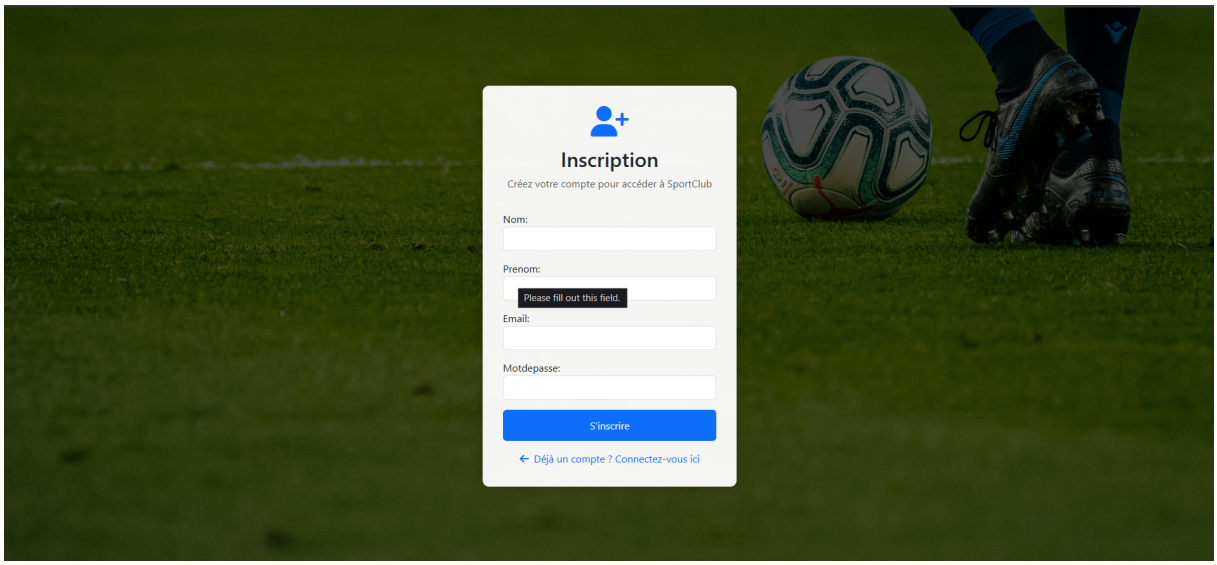

FIGURE 4.3 – la page d'inscription

4.5.2 Page de connexion pour membre

L interface montre la page de connexion du site SportClub, qui permet aux utilisateurs existants d'accéder à leur espace personnel. La conception épurée de cette interface met l'accent sur les éléments essentiels, à savoir les champs pour saisir l'adresse e-mail et le mot de passe. Un bouton "Se connecter" invite clairement l'utilisateur à s'authentifier. En bas de la fenêtre, un lien "Pas encore de compte ? Inscrivez-vous ici" offre une voie d'accès aux nouveaux membres souhaitant créer un compte.

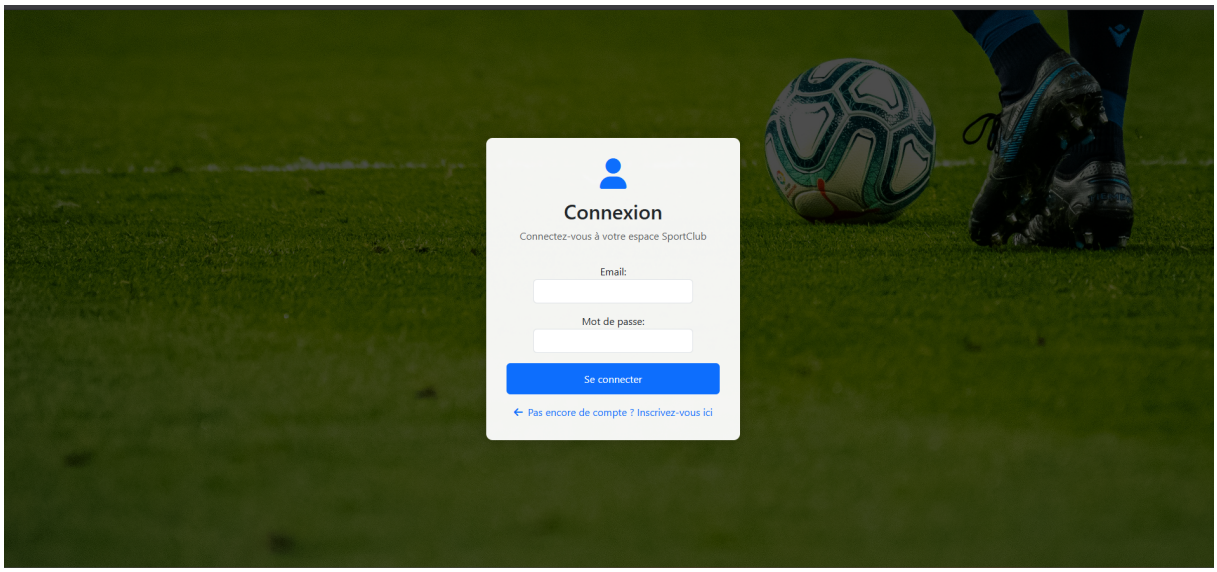

FIGURE 4.4 – la page de connexion

4.5.3 page pour listes des terrains

L'image présente la page "Liste des terrains" du site SportClub, qui répertorie les différents espaces sportifs disponibles pour la réservation. On y trouve trois types de terrains . Chaque terrain est accompagné d'une image, de son état de disponibilité et de son tarif de location. Un bouton "Réserver" permet aux membre de procéder à la réservation de l'espace de leur choix. Cette page offre une vue d'ensemble claire et concise des infrastructures sportives proposées par le club, facilitant ainsi la sélection et la réservation par les membres.

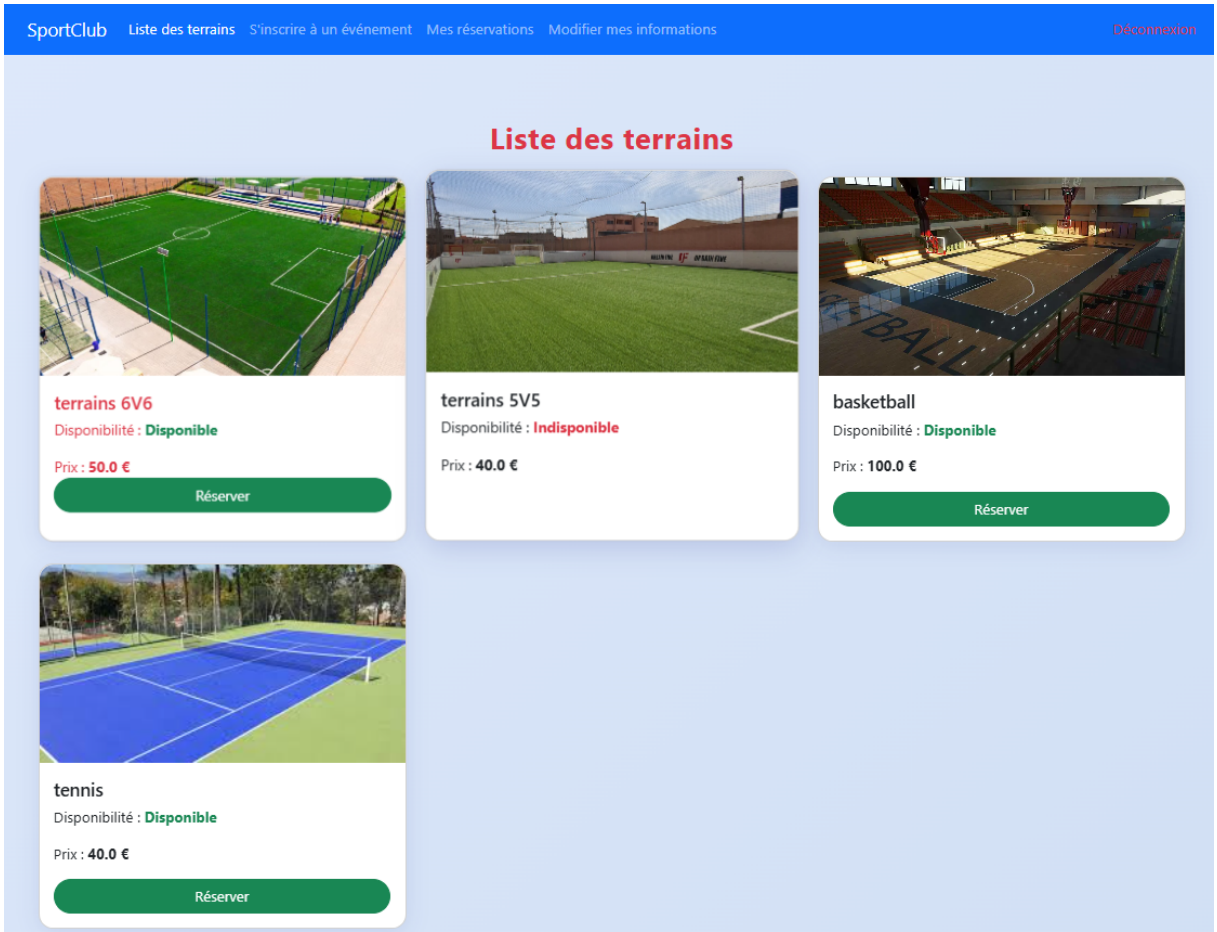

FIGURE 4.5 – liste des terrains

4.5.4 page d'evenement

La page "Liste des événements" présente les différentes activités et manifestations organisées par le club SportClub.

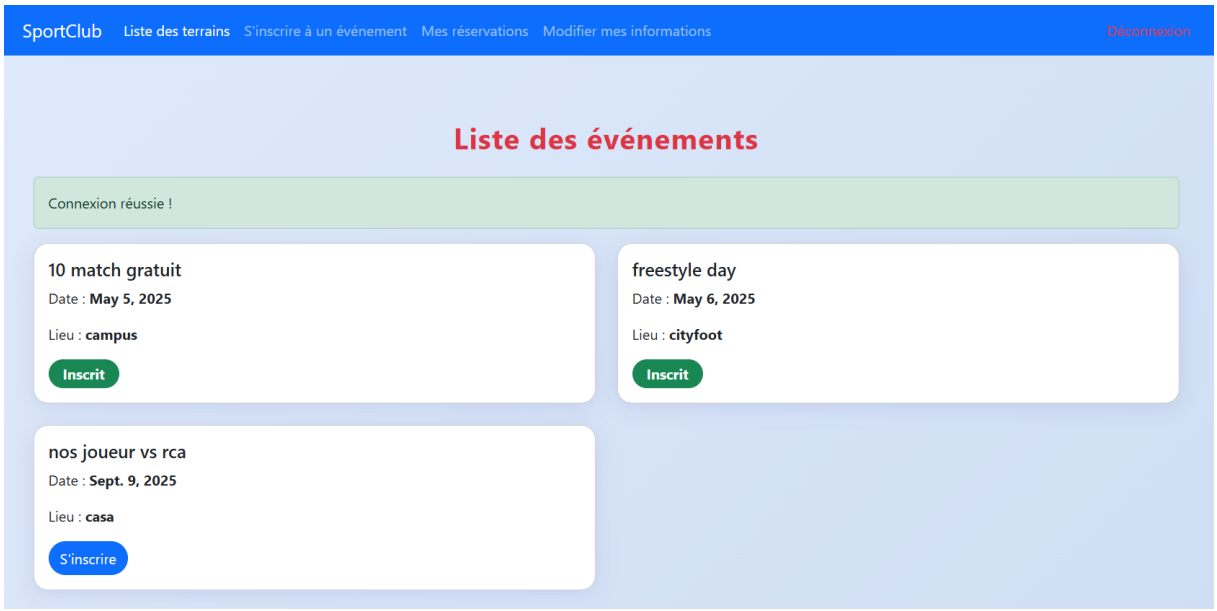

FIGURE 4.6 – la page d evenement

4.5.5 liste des reservation

Le tableau récapitule les différentes réservations de terrains effectuées par l'utilisateur. Chaque ligne correspond à une réservation avec les informations suivantes :

- Date de début et de fin de la réservation
- Type de terrain réservé (terrain)
- Montant de la réservation
- Statut de la réservation (Validée, Annulée, Rattrapage)
- Actions possibles (Rattrapage)

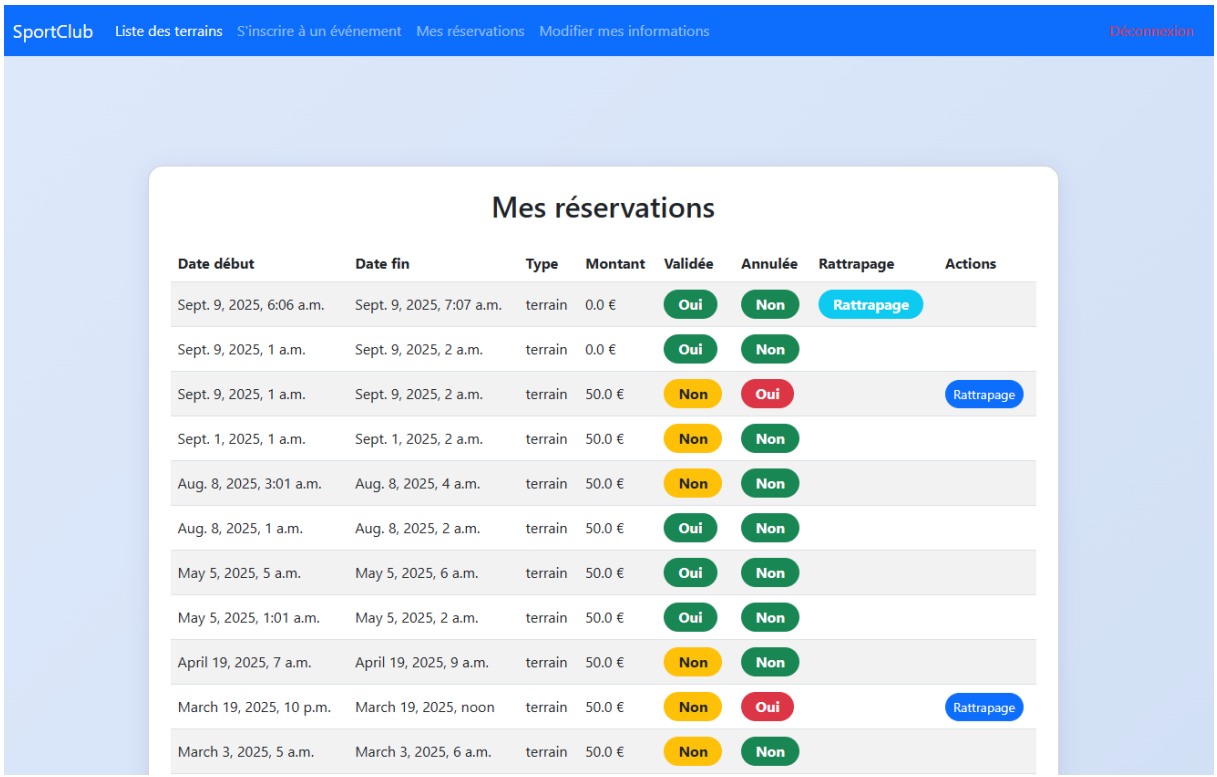

FIGURE 4.7 – mes reservations

4.5.6 modfier informations

Cette page permet à l'utilisateur de visualiser et de modifier ses informations de profil, telles que son nom, prénom et adresse email.

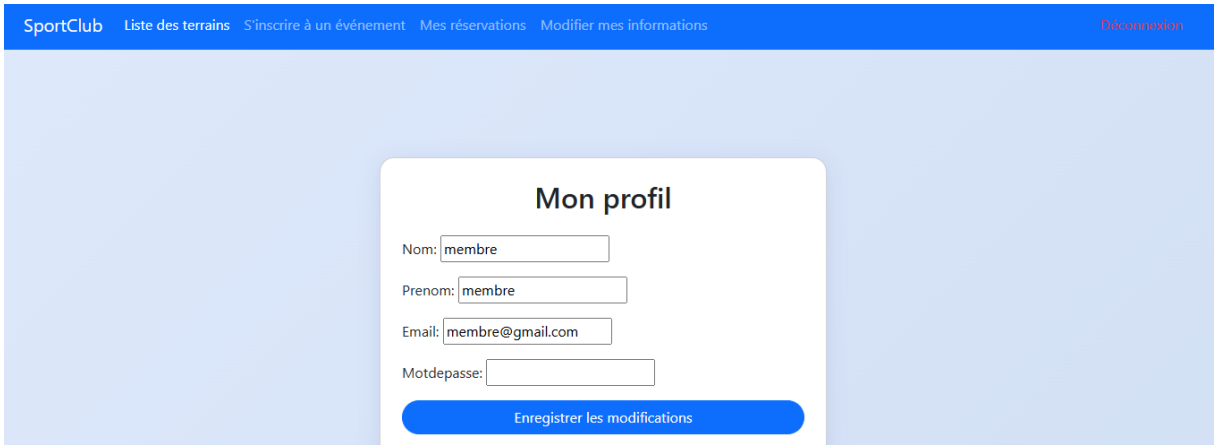

FIGURE 4.8 – modfier les informations

4.6 solution du admin

4.6.1 connexion admin

La page affichée est la page de connexion pour l'espace administrateur du site SportClub.

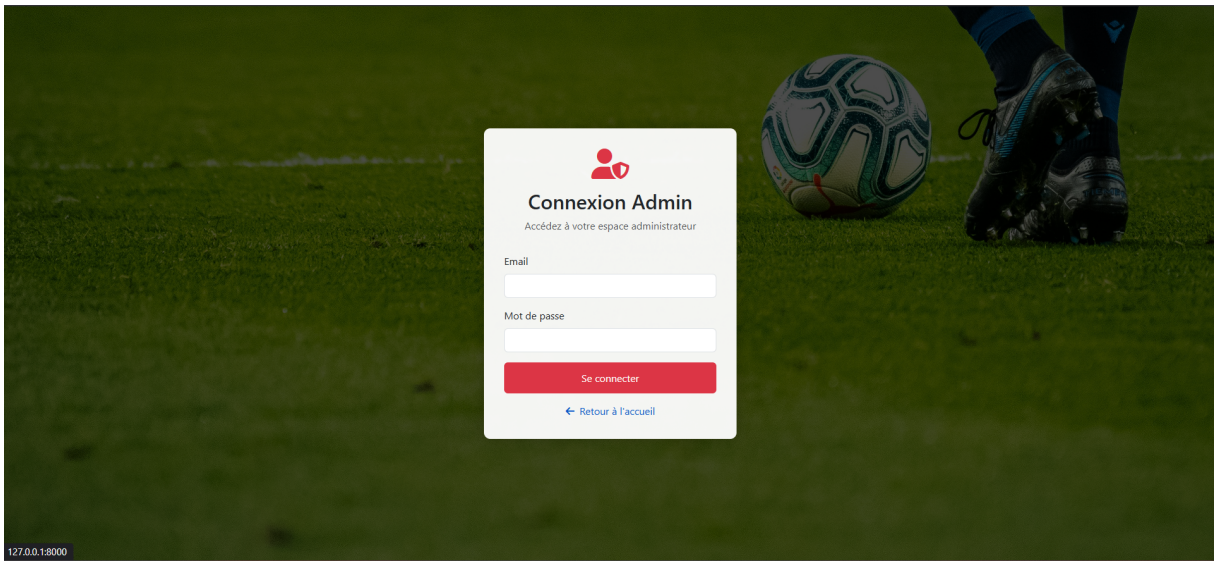

FIGURE 4.9 – connexion admin

4.6.2 Tableau de Bord

L'interface "SportClub Admin" est un tableau de bord administratif conçu pour la gestion interne du club sportif. Il offre une vue d'ensemble des principales métriques (membres, coachs, factures) et permet d'accéder rapidement aux différentes sections de gestion.

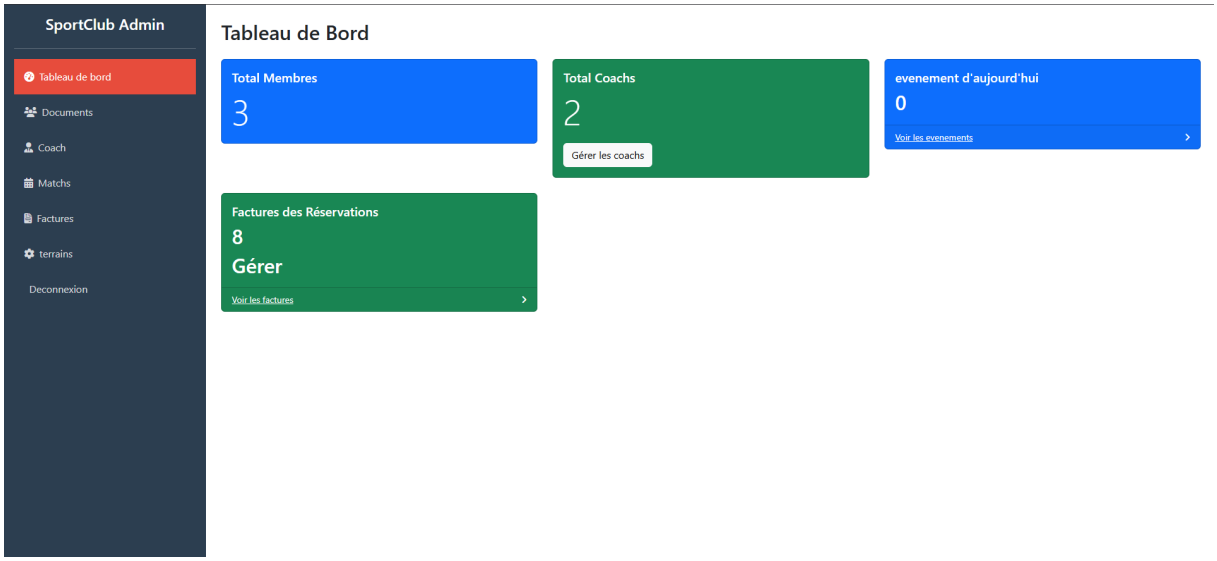

FIGURE 4.10 – Tableau de Bord

4.6.3 Liste des documents

L'interface analysée correspond à la section "Documents" du back-office SportClub Admin, actuellement sélectionnée dans le menu de navigation. Elle affiche une liste de fichiers envoyer par les coachs dans le système, avec des informations basiques et des options de gestion.

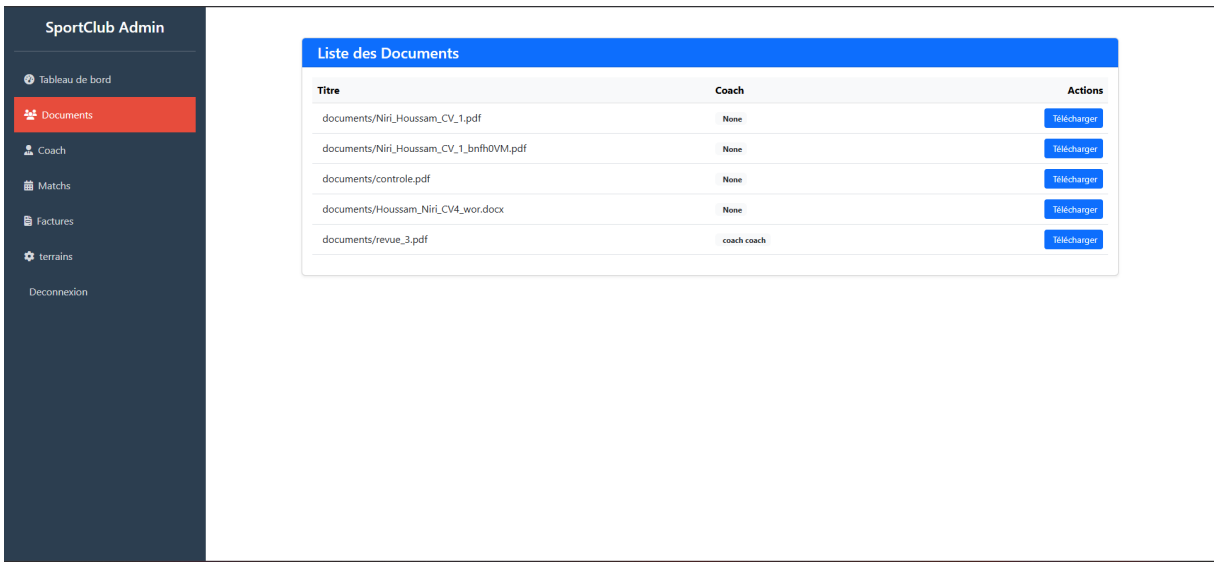

FIGURE 4.11 – liste des documents

4.6.4 gerer les coachs

Cette interface représente la section de gestion des coachs dans le back-office SportClub Admin. Elle permet aux administrateurs de visualiser, rechercher et gérer les coachs affiliés au club.

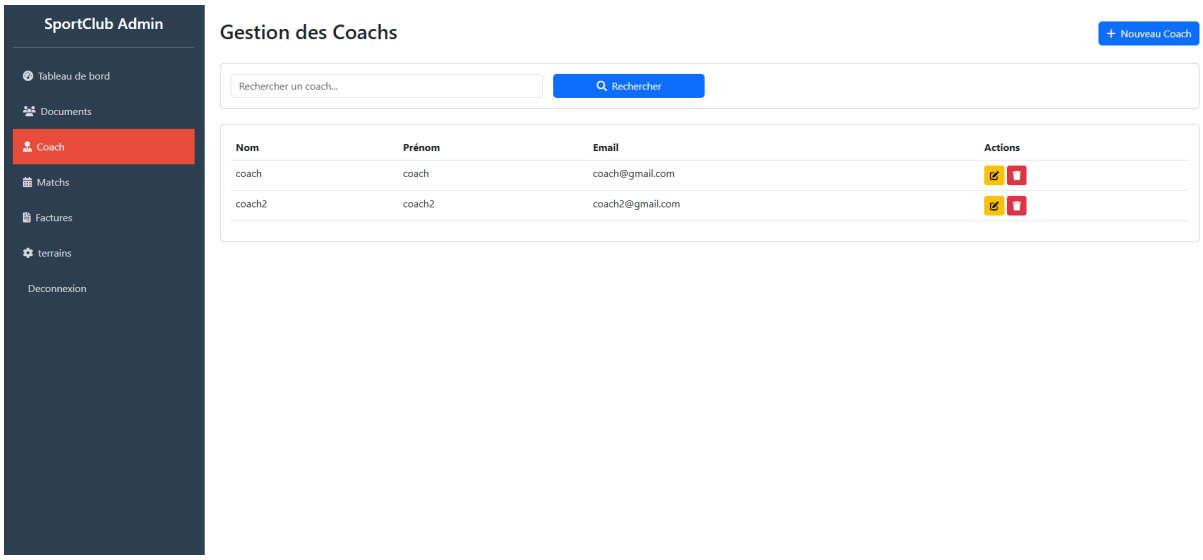

FIGURE 4.12 – gerer les coachs

4.6.5 gerer evenements

Cette interface représente la section de gestion des événements dans le back-office SportClub Admin. Elle permet aux administrateurs de visualiser, rechercher et gérer les différents événements organisés par le club, tels que les matchs, les entraînements ou les activités spéciales. Les administrateurs peuvent consulter les détails de chaque événement, notamment le titre, la date, le lieu, et effectuer diverses actions comme la modification, la suppression ou la publication des événements. Un bouton "+ Nouvel Événement" en haut à droite permet également d'ajouter de nouveaux événements au système. Cette interface offre ainsi aux gestionnaires du club un outil complet pour organiser et contrôler l'ensemble de la programmation événementielle.

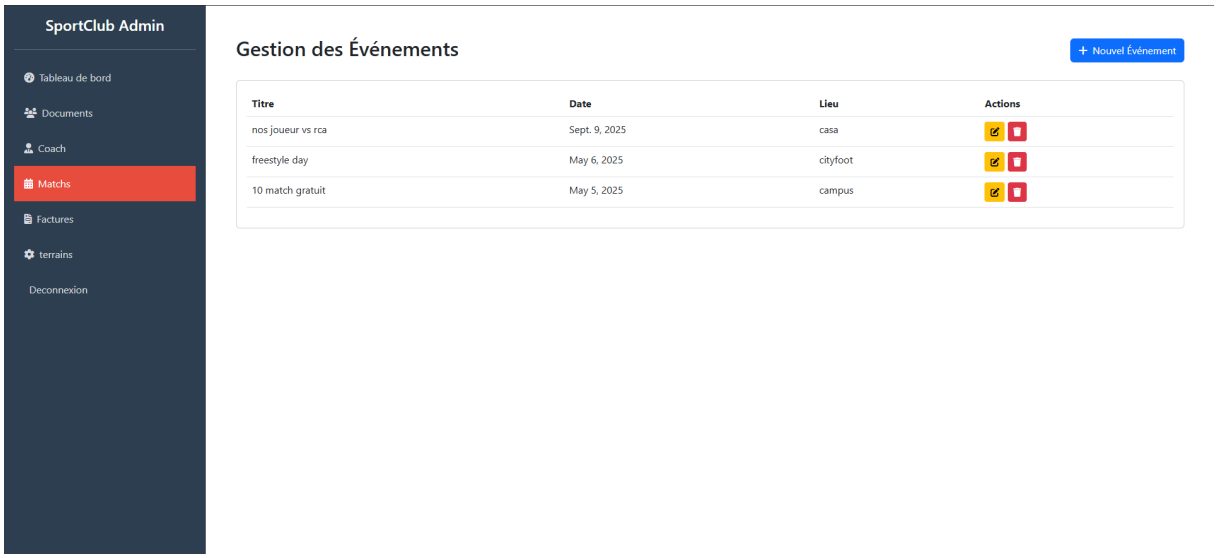

FIGURE 4.13 – gerer les evenements

4.6.6 facturation

Cette interface permet aux administrateurs de SportClub de visualiser et gérer les factures liées aux réservations des terrains de jeu. Les informations affichées incluent le numéro de facture, le membre concerné, le terrain réservé, la date, le montant et le statut de la facture. Des actions telles que le téléchargement ou la validation des factures sont également disponibles.

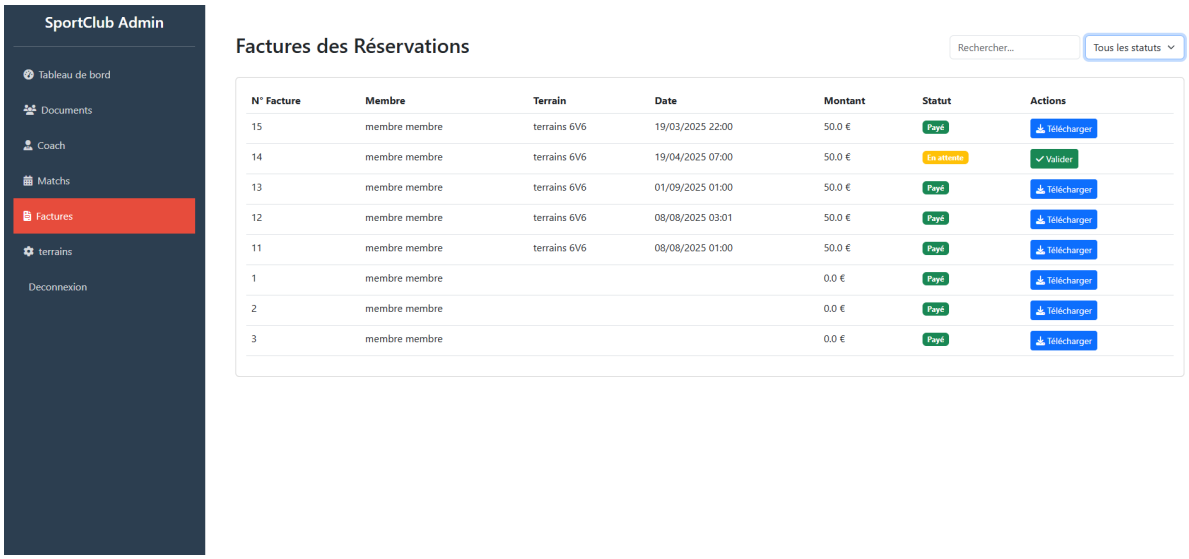

FIGURE 4.14 – facturation

Cette facture possède une base fonctionnelle mais gagnerait à être professionnalisée. Les améliorations proposées en feraient un outil complet à la fois utile administrativement et valorisant pour l'image du club. La version améliorée pourrait servir de modèle pour tous les documents financiers du système.

FACTURE

13

SportClub

140 lot adarissa Sidi Maarouf
20500 casablanca
Tél: 06 06 87 30 74
Email: contact@sportclub.com

Client

membre membre
membre@gmail.com

Description	Date	Durée	Montant
Réservation de terrain - terrains 6V6	01/09/2025 01:00	1 hour	50.0 €

Total TTC : 50.0 €

Facture émise le 17/05/2025

Merci de votre confiance !

FIGURE 4.15 – La facture

4.6.7 Les terrains

Cette interface permet la gestion des terrains sportifs du club. Elle liste les différents terrains disponibles avec leurs caractéristiques principales et options de gestion. L'interface souffre cependant de plusieurs problèmes d'organisation et de terminologie.

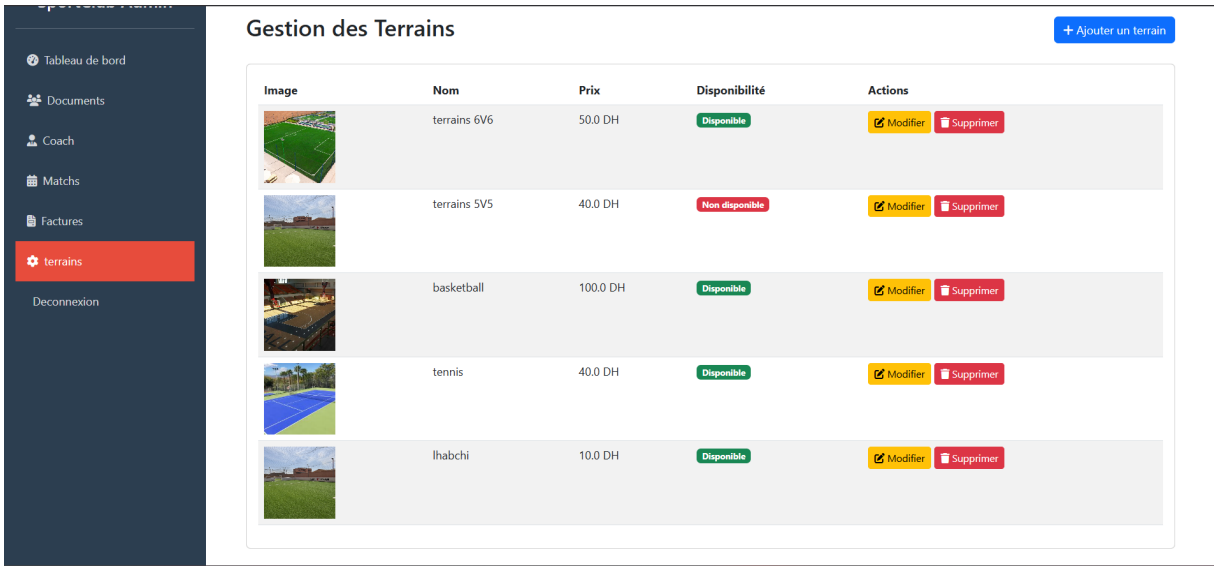

FIGURE 4.16 – Liste des terrains

Cette interface nécessite une refonte urgente pour être fonctionnelle. Les corrections proposées permettraient une gestion efficace des terrains tout en évitant les confusions. La version améliorée serait à la fois plus professionnelle et plus intuitive pour les administrateurs.

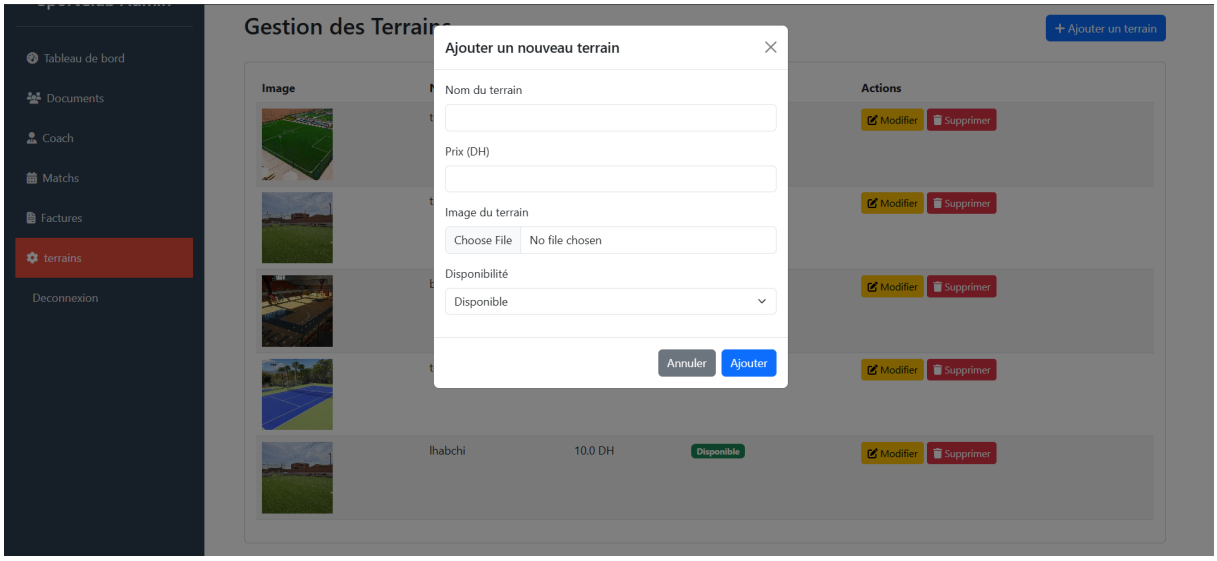

FIGURE 4.17 – Ajouter des terrains

L'interface "Modifier le terrain" permet aux administrateurs du SportClub de mettre à jour les informations d'un terrain existant. Ils peuvent modifier le nom, l'image, le prix en dirhams et l'état de disponibilité du terrain. L'image actuelle du terrain est affichée pour référence, et l'utilisateur peut choisir un nouveau fichier pour la remplacer. Des boutons d'action sont également présents pour enregistrer les modifications ou supprimer complètement le terrain du système. Cette interface offre ainsi une fonctionnalité complète de gestion des terrains, permettant aux responsables du club de maintenir à jour les détails

de leurs installations sportives.

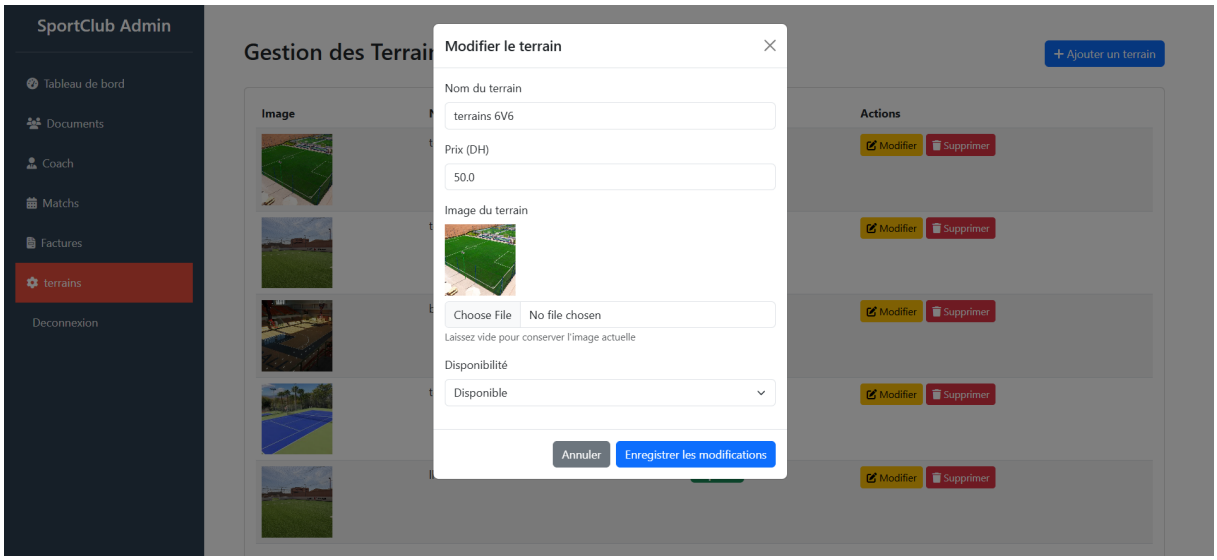

FIGURE 4.18 – Modifier les terrains

4.7 Solution coach

4.7.1 connexion du coach

cette page de connexion offre une expérience utilisateur fluide et efficace pour permettre aux coachs de s'authentifier et d'accéder à leur espace personnel au sein de l'application.

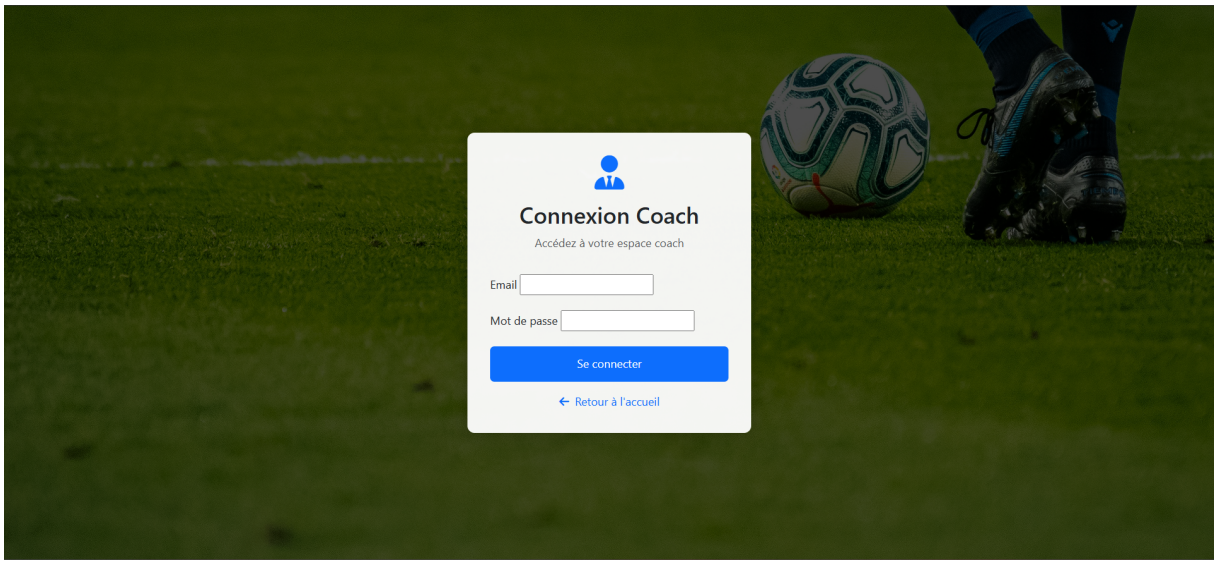

FIGURE 4.19 – connexion du coach

4.7.2 Gestion des membre

Cette interface permet aux administrateurs de gérer les membres du club SportClub. Elle offre une vue tabulaire des adhérents avec des fonctionnalités de recherche et d'actions basiques.

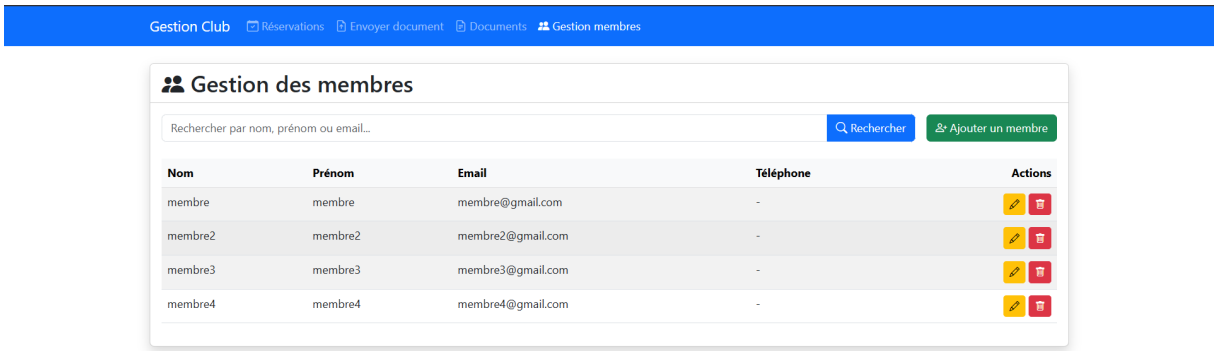

FIGURE 4.20 – gestion des membres

4.7.3 envoyer les document

Cette interface fait partie du menu "Gestion Club" et permet aux utilisateurs d'envoyer une licence ou un certificat. Elle comporte les éléments suivants :

— Un champ "Fichier" avec un bouton "Choose File" pour sélectionner le fichier à envoyer.

— Un menu déroulant "Type" avec deux options : "Envoyer" et "Annuler"

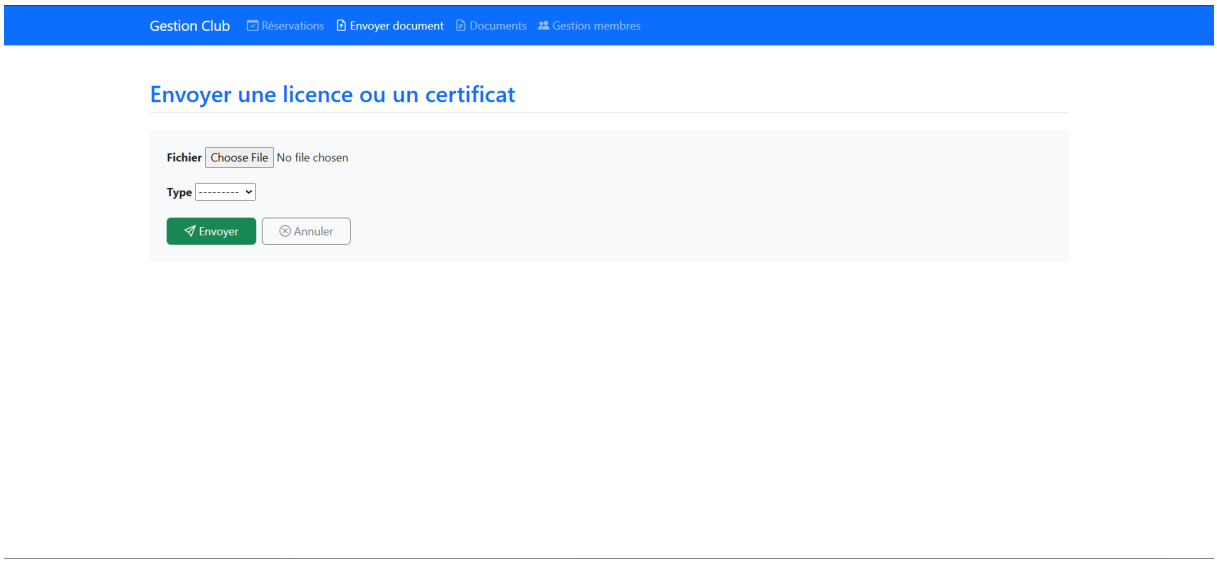

FIGURE 4.21 – envoyer des documents

Cette interface permet aux coach de visualiser la liste des documents disponibles dans le système, avec des informations sur le titre, le type (licence ou certificat) et la possibilité de télécharger chaque document.

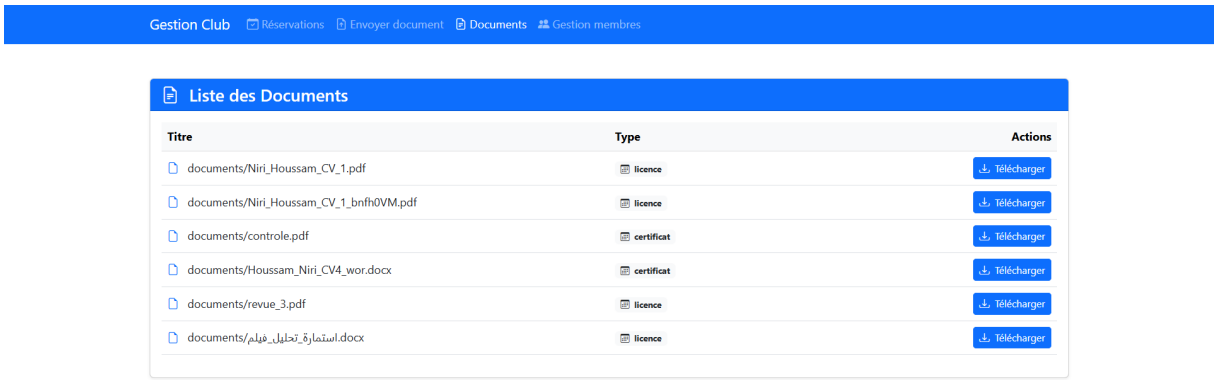

FIGURE 4.22 – visualiser ces documents

4.7.4 liste des reservations

Cette interface permet de visualiser les réservations effectuées pour chaque jour, avec des informations détaillées sur chaque réservation (membre, dates, type, validité) ainsi que des actions possibles (modifier, supprimer, valider).

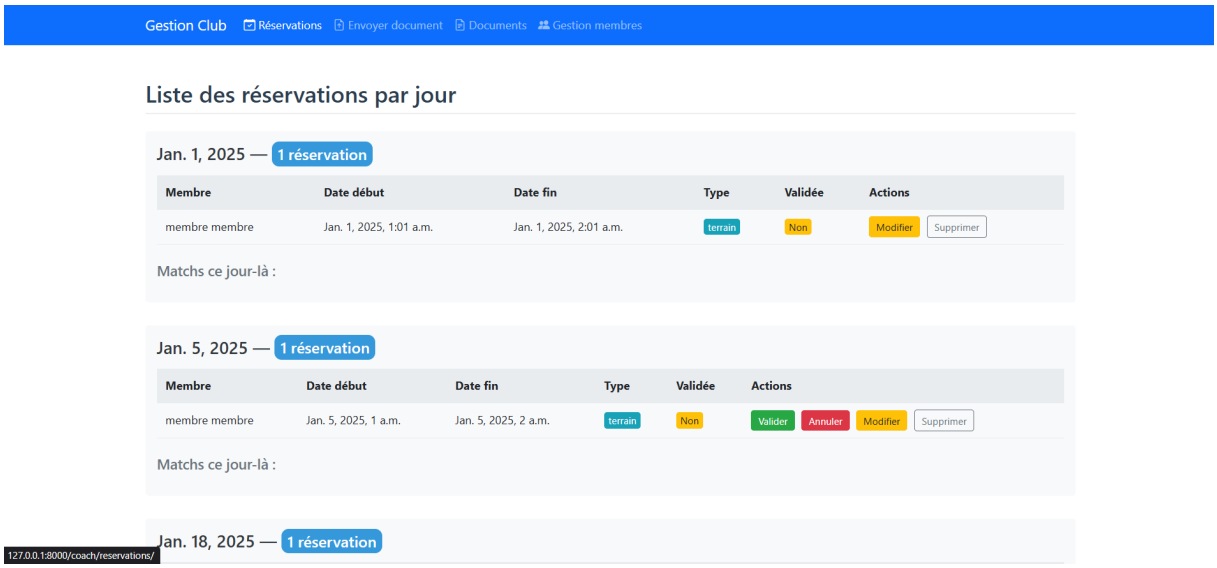

FIGURE 4.23 – liste des reservations

4.8 Conclusion

La phase de réalisation a permis de transformer la conception théorique en une application web fonctionnelle, répondant aux besoins du club sportif. Grâce à Django et une architecture bien structurée, nous avons implémenté les principales fonctionnalités : gestion des utilisateurs, réservations, paiements et événements. Cette étape a consolidé le travail de modélisation et validé nos choix techniques à travers un développement concret et opérationnel.

Conclusion Générale et perspective

Conclusion Générale

Ce projet de fin d'année a constitué une opportunité précieuse pour concevoir et développer un **système complet de gestion des réservations pour un club sportif**, répondant à des besoins concrets de modernisation, d'efficacité et de centralisation dans un secteur souvent encore marqué par des méthodes manuelles.

À travers une démarche rigoureuse et structurée, nous avons mené à bien les étapes suivantes :

— Analyse du **besoin fonctionnel** du club et élaboration d'un **cahier des charges précis**.

— Conception d'une **modélisation UML complète** (cas d'utilisation, diagrammes de classes, de séquence et d'activités), assurant la cohérence entre la solution et les attentes métier.

— Développement d'une **application web responsive, performante et sécurisée**, en s'appuyant sur le framework **Django** et le système de gestion de base de données **MySql**.

— Intégration de fonctionnalités avancées : gestion des utilisateurs et rôles, réservation en ligne, paiements, gestion des événements, calendrier dynamique, tableau de bord administratif, etc.

— Application d'une **méthodologie agile (Scrum)**, facilitant l'avancement itératif, la réactivité face aux imprévus, et l'amélioration continue de la solution.

Ce projet nous a permis de mobiliser l'ensemble des compétences techniques acquises au cours de notre formation : **développement web, modélisation orientée objet, sécurité informatique, gestion de projet**, ainsi que des **compétences transversales** essentielles telles que la communication, le travail en équipe, l'autonomie et l'esprit d'initiative.

En conclusion, cette expérience a non seulement consolidé nos acquis, mais elle nous a également préparés à aborder le monde professionnel avec une vision claire des enjeux

technologiques et organisationnels. Le système développé constitue une base évolutive pouvant être enrichie et déployée dans un véritable contexte d'exploitation.

Perspectives

Bien que les objectifs initiaux aient été atteints, plusieurs améliorations et extensions peuvent être envisagées pour enrichir davantage l'application :

- **Module de notifications** :
  - Notification par e-mail ou SMS pour confirmer les réservations
  - Alertes en cas de modification ou d'annulation de créneau
  - Rappels automatiques avant les événements
- **Application mobile** :
  - Développement d'une version native pour iOS et Android
  - Fonctionnalités hors-ligne pour consulter ses réservations
  - Notifications push intégrées
- **Internationalisation** :
  - Interface multilingue (français, anglais, arabe)
  - Adaptation aux spécificités locales (devises, fuseaux horaires)
  - Traduction dynamique du contenu

Webographie

— https://www.djangoproject.com/ – Site officiel de Django

— https://www.w3schools.com/django/index.php – Tutoriels de base sur Django

— https://docs.djangoproject.com/en/stable/ – Documentation officielle de Django

— https://www.w3schools.com/bootstrap5/index.php – Tutoriels sur Bootstrap 5

— https://getbootstrap.com/ – Site officiel de Bootstrap

— https://www.w3schools.com/html/default.asp – Cours HTML de W3Schools

— https://www.w3schools.com/css/default.asp – Cours CSS de W3Schools

— https://www.mysql.com/ – Site officiel de MySQL

— https://www.sqltutorial.org/ – Tutoriels SQL clairs et progressifs

— https://www.tutorialspoint.com/uml/index.htm – Introduction au UML

— https://lucidchart.com/pages/uml-diagram – Outil et guide pour les diagrammes UML

— https://developer.mozilla.org/fr/ – Documentation web complète (HTML, CSS, JS)

— https://realpython.com/tutorials/django/ – Tutoriels avancés Django

— https://www.geeksforgeeks.org/django-tutorial/ – Ressources Django sur GeeksforGeeks

— https://owasp.org/www-project-top-ten/ – Principales vulnérabilités de sécurité web (OWASP)