

Universite Djillali Liabes de Sidi Bel Abbes  
Faculte des Sciences Exactes  
Departementd informatique

## Mémoire de fin d'études Pour l'obtention du diplôme de Licence en Informatique

*Spécialité : Ingénierie des Systèmes d'Information et Logiciel (ISIL)*

**Thème :**

**Conception et realisation d'un site web de gestion des  
espaces de fitness et des salles de sport**

**Présente par :**

Mr OURMASSI ALAA EDDINE

Mr SEBTI MOHAMED ELAMINE

Soutenu le : 03/06/2024

Devant le jury composé de :

Jury	Nom et Prénom	Grade	Université
Encadrant	M. Nemmich Mohamed Amine	MCB	UDL SBA
Examinateur M. enseignant1		MCB	UDL SBA

# REMERCIEMENT

*En témoignage de notre gratitude et de notre profond respect nous adressons nos vifs et sincères remerciements à :*

*Dieu miséricordieux, pour son aide et sa bénédiction.*

*A notre encadreur Monsieur **NEMMICH MOHAMED AMINE** pour ses conseils, sa disponibilité et son encouragement qui nous ont permis de réaliser ce travail dans les meilleures conditions.*

*Les jurys pour leurs efforts et leur soin apporté à notre travail.*

*Aux enseignants de notre université et département informatique.*

*Nous ne pouvons nommer ici toutes les personnes qui de près ou de loin nous ont aidé et encouragé mais nous les en remercions vivement.*

*Enfin, nous tenons à dire combien le soutien quotidien de notre famille a été important tout au long de ces quelques années, nous leur devons beaucoup.*

*Merci pour tous*

## *Dédicace*

Je dédie ce mémoire

A mes chers parents pour leur amour inestimable, leur confiance,  
leur soutien, leurs sacrifices et toutes les valeurs qu'ils ont su  
m'inculquer.

A mes frères « Bagdade, Faycale et Omar » pour leurs appui et  
leur encouragement.

A ma famille « SEBTI » chacun à son nom.

A Tous nos ami(e)s et collègues de la spécialité ISIL.

Que ce travail soit l'accomplissement de vos vœux tant allégués,  
et le fruit de votre soutien infaillible.

Merci d'être toujours là pour moi.

***MOHAMED ELAMINE***

## *Dédicace*

A ceux qui ont contribué à l'élaboration de ce travail

Et ceux à qui nous devons tant

Nos plus grande gratitude et tout notre amour à nos parents ,qui  
ont su me faire confiance ,mesouterir et nous encourager au  
cours de nos vie.

A tous qui nous aiment

A tous nos amies en qui nous avons toujourstrouvé le souten et le  
réconfort.

***ALAA EDDINE***

# SOMMAIRE

**SOMMAIRE****LISTE DES FIGURES****INTRODUCTION GENERALE**

<b>Chapter 1: Contexte et présentation du projet</b>	<b>9</b>
1.1 Introduction . . . . .	9
1.2 Contexte générale du projet : . . . . .	9
1.2.1 Présentation des espaces de fitness et des salles de sport . . . . .	9
1.2.2 Plateformes de gestion des salles de sport . . . . .	10
1.2.3 Problématique . . . . .	12
1.2.4 Objectifs du projet . . . . .	12
1.2.5 Conclusion . . . . .	13
<b>Chapter 2: Analyse des besoins et conception</b>	<b>14</b>
2.1 Introduction . . . . .	14
2.2 Analyse et spécification des Besoins . . . . .	14
2.2.1 Identification des acteurs . . . . .	14
2.2.2 Les besoins fonctionnels . . . . .	15
2.2.3 Les besoins non fonctionnels . . . . .	15
2.3 Présentation d'UML . . . . .	15
2.4 Types des diagrammes d'UML . . . . .	16
2.4.1 Diagramme de cas d'utilisation . . . . .	16
2.4.2 Diagramme de séquence . . . . .	18
2.4.3 Diagramme de classe . . . . .	23
2.5 Conception de la base de données . . . . .	24
2.5.1 Description du modèle relationnel . . . . .	25
2.6 Conclusion . . . . .	25
<b>Chapter 3: Réalisation de l'application</b>	<b>26</b>
3.1 Introduction . . . . .	26

3.2 Application web . . . . .	26
3.3 Environnement de travail . . . . .	27
3.3.1 L'environnement matériel . . . . .	27
3.3.2 L'environnement logiciel . . . . .	27
3.3.3 Langages de programmation . . . . .	30
3.4 Présentation des interfaces de notre application . . . . .	30
3.4.1 Coté Administrateur . . . . .	31
3.4.2 Coté Utilisateur(member) . . . . .	35
3.5 Conclusion . . . . .	41

## **CONCLUSION GÉNÉRALE**

## **BIBLIOGRAPHIE**

# LISTE DES FIGURES

1.1	Interface de fresha.	10
1.2	Interface de liber.fit.	11
1.3	Interface de virtuagym.	11
2.1	Diagramme de cas d'utilisation « Admin ».	16
2.2	Diagramme de cas d'utilisation « utilisateur/client ».	17
2.3	Diagramme de cas d'utilisation « Coach ».	18
2.4	Diagramme de séquence « Inscription ».	19
2.5	Diagramme de séquence « s'inscrire à un class ».	20
2.6	Diagramme de séquence « reserver du materiel ».	21
2.7	Diagramme de séquence « gestion des classes ».	22
2.8	Diagramme de séquence « réservation du materiel ».	23
2.9	Diagramme de classe.	24
3.1	MySQL.	28
3.2	Node.js.	28
3.3	Visual Studio Code.	29
3.4	Interface d'authentification.	31
3.5	Interface de contrôle utilisateur.	31
3.6	Interface de statistiques générales du club.	32
3.7	Interface des réservations.	32
3.8	Interface pour ajouter des équipements.	33
3.9	Interface de paiements.	33
3.10	Interface pour ajouter une transaction.	34
3.11	Interface pour ajouter des classes.	34

3.12 Interface du tableau de bord . . . . .	35
3.13 Interface pour communiquer avec les utilisateurs. . . . .	35
3.14 Interface d'inscription (Utilisateur). . . . .	36
3.15 Interface de sélection d'équipement (Utilisateur) . . . . .	36
3.16 Ajouter du matériel à la liste de souhaits (Utilisateur). . . . .	37
3.17 Interface de sélection des entraîneurs (Utilisateur). . . . .	37
3.18 L'interface du compte personnel du coach. . . . .	38
3.19 Interface de sélection de classe (Utilisateur). . . . .	38
3.20 Interface du tableau de bord (Utilisateur) . . . . .	39
3.21 Une interface spéciale pour tous les choix des utilisateurs . . . . .	39
3.22 Interface client . . . . .	40
3.23 Interface pour communiquer avec les membres. . . . .	40

# INTRODUCTION GENERALE

Dans notre société moderne, axée sur le bien-être et la santé, les espaces de fitness et les salles de sport occupent une place centrale. Ces établissements ne se limitent pas à être de simples centres d'entraînement, mais sont plutôt des bastions dynamiques où la promotion de la forme physique, de la santé et du bien-être est au cœur de leur mission.

En effet, ils offrent bien plus que de simples équipements d'entraînement, ils représentent des environnements où les individus peuvent s'engager dans une multitude d'activités physiques, bénéficier de conseils d'experts et trouver la motivation et le soutien nécessaires pour atteindre leurs objectifs de remise en forme.

Cependant, dans le contexte du monde moderne, la gestion efficace de ces établissements est cruciale pour assurer leur bon fonctionnement et pour offrir une expérience client optimale, c'est dans cette optique que nous avons développé une application web dédiée à la gestion des espaces de fitness et des salles de sport, cette plateforme offre aux responsables d'établissements les outils nécessaires pour garantir un fonctionnement fluide et une gestion efficace, tout en favorisant une interaction transparente avec leur clientèle.

Notre projet repose sur la fusion habile de la conception web, de l'expérience utilisateur et de la gestion de données spécifiques au secteur du fitness, l'objectif principal est d'offrir aux utilisateurs une expérience intuitive et conviviale, tout en permettant aux propriétaires d'établissements de gérer efficacement leurs activités et d'interagir avec leur clientèle de manière transparente.

En mettant l'accent sur l'optimisation de l'interaction client et la simplification de la gestion des salles de sport, notre application vise à répondre aux besoins changeants du secteur et à accompagner les établissements dans leur quête d'excellence en matière de service et de satisfaction client.

Ce rapport comprendra trois chapitres :

- **Chapitre I : Contexte et présentation du projet**

Contenant une introduction, problématiques, présentation des espaces de fitness et des salles de sport et plateformes de gestion des salles de sport.

- **Chapitre II : Analyse des besoins et conception**

Nous l'avons chargé de la partie conception, où nous avons présenté le langage de modélisation UML, après ça, nous avons identifié les acteurs de l'application. Ensuite, nous avons cité les cas d'utilisations pour chaque acteur afin qu'on modélise les différents diagrammes de cas d'utilisations et de séquences, et vers la fin le diagramme de classe.

- **Chapitre III : Réalisation de l'application**

Présente la partie réalisation de notre application. Nous avons exposé les technologies que nous avons utilisées pour la réalisation et le développement de cette application, et présenté les différentes interfaces de l'application.

# Chapter 1

## Contexte et présentation du projet

### 1.1 Introduction

Ce chapitre présente le contexte et les objectifs de notre projet de fin d'études, nous allons commencer par une présentation des espaces de fitness et des salles de sport et introduction sur les plateformes de gestion des salles de sport et applications existantes , ensuite nous aborderons la problématique et les raisons qui nous ont amenées à nous lancer dans ce travail.

### 1.2 Contexte générale du projet :

#### 1.2.1 Présentation des espaces de fitness et des salles de sport

Les espaces de fitness et les salles de sport sont des piliers incontournables de notre société moderne axée sur le bien-être et la santé, Bien plus que de simples centres d'entraînement, ces établissements jouent un rôle vital dans la promotion de la forme physique, de la santé et du bien-être, En offrant une gamme variée d'équipements d'entraînement et de programmes, ils créent des environnements dynamiques où les individus peuvent s'engager dans une multitude d'activités physiques adaptées à leurs besoins et à leurs objectifs.

les espaces de fitness et les salles de sport représentent également des communautés dynamiques où les membres peuvent trouver motivation, inspiration et soutien pour atteindre leurs objectifs de remise en forme.

Les conseils d'experts dispensés par les instructeurs et les entraîneurs ajoutent une valeur significative en aidant les membres à optimiser leurs séances d'entraînement et à adopter des modes de vie sains.

Dans un monde où la santé et le bien-être sont devenus des priorités majeures, les espaces de fitness et les salles de sport jouent un rôle essentiel en fournissant des ressources et des environnements favorables à l'adoption de modes de vie actifs et sains. Ils constituent des sanctuaires où les individus peuvent investir dans leur santé physique et mentale, et où les communautés se rassemblent pour promouvoir le bien-être dans son ensemble.

### 1.2.2 Plateformes de gestion des salles de sport

Les plateformes de gestion des salles de sport sont des outils numériques conçus pour simplifier et améliorer la gestion quotidienne des salles de sport et des centres de fitness, ces plateformes offrent une gamme de fonctionnalités visant à rationaliser les opérations, à améliorer l'expérience client et à optimiser les performances commerciales, de la planification des horaires et des cours à la gestion des adhésions et des paiements, ces solutions technologiques fournissent aux propriétaires de salles de sport les outils nécessaires pour gérer efficacement leur entreprise dans un environnement de plus en plus concurrentiel.

Voici trois applications web populaires pour gestion des salles de sport en ligne :

- Fresha** : Fresha est une plateforme de gestion de salle de sport qui simplifie la gestion d'une salle de sport avec une suite complète de fonctionnalités, notamment la gestion des adhésions, la planification des cours, les paiements en ligne, la communication, etc. Il vise à améliorer l'efficacité et à offrir une expérience personnalisée aux gestionnaires et aux membres.

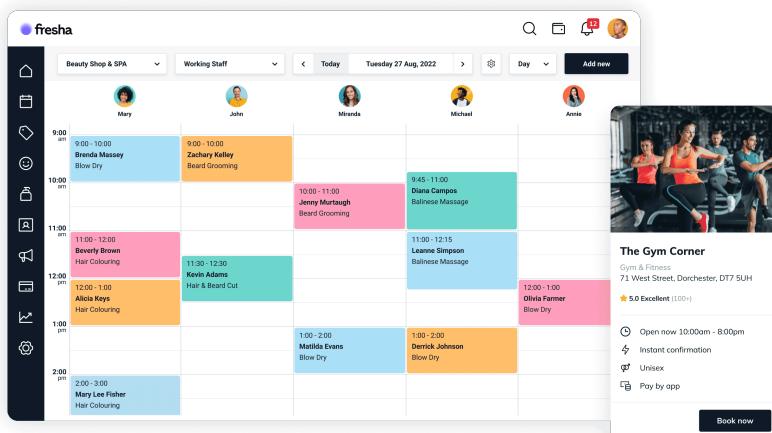


Figure 1.1: Interface de fresha.

- Liber.fit** : Liber.fit est une plateforme web pour la gestion des salles de sport, offrant une gamme complète d'outils pour faciliter les opérations, elle permet la gestion des membres,

la planification des cours, la gestion des réservations et des paiements en ligne, ainsi que la communication avec les membres, dans le but de simplifier les opérations quotidiennes et d'améliorer l'expérience des utilisateurs.



Figure 1.2: Interface de liber.fit.

**3. Virtuagym** : Virtuagym est une application web de fitness offrant une plateforme pour la gestion et le suivi de la condition physique, elle permet de planifier des entraînements, de suivre les progrès, d'accéder à des programmes personnalisés, et de se connecter avec des coachs en ligne, elle comprend également le suivi de l'alimentation, des outils de motivation, et des communautés en ligne pour favoriser l'engagement des utilisateurs.

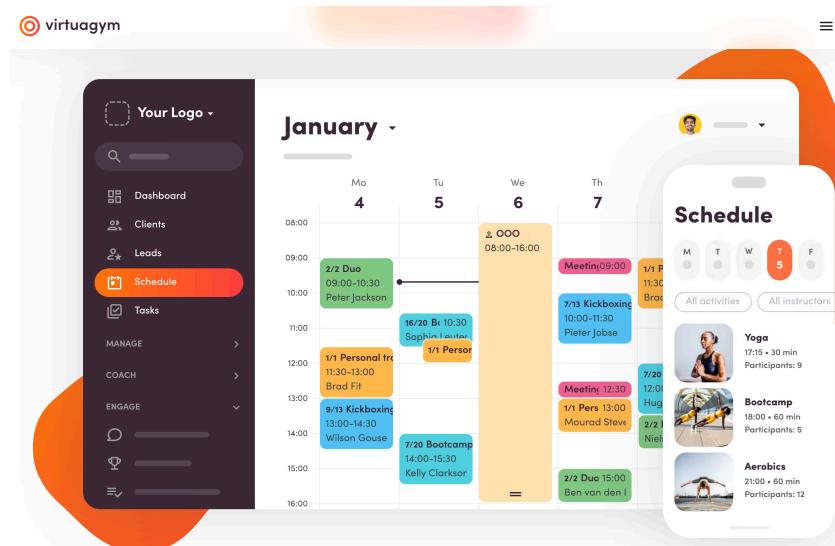


Figure 1.3: Interface de virtuagym.

### 1.2.3 Problématique

La problématique des salles de sport est un sujet complexe et multidimensionnel qui attire un intérêt croissant dans notre société contemporaine, alors que la prise de conscience de l'importance d'un mode de vie sain augmente, les salles de sport deviennent des lieux incontournables pour de nombreux individus en quête de bien-être physique et mental.

Cependant, malgré leur popularité, ces établissements font face à une série de défis allant de l'accessibilité et de l'équité à la qualité des services.

N'oublions pas que la pandémie de Covid-19 a exacerbé ces défis, les obligeant à s'adapter rapidement aux restrictions sanitaires. En plus des défis traditionnels, ils doivent désormais jongler avec facilité entre distanciation sociale, les entraînements à distance et communication avec leurs clients, Coordonner efficacement toutes ces étapes peut être difficile sans un processus automatisé.

### 1.2.4 Objectifs du projet

L'objectif essentiel de ce travail se résume à « la conception et la réalisation d'un site web de gestion des espaces de fitness et des salles de sport », Le projet vise à créer une plateforme numérique innovante destinée à simplifier et à optimiser la gestion des installations sportives, Les objectifs primordiaux incluent la mise en place d'une interface utilisateur conviviale permettant aux membres de réserver des créneaux horaires pour les équipements et les cours, ainsi que la gestion efficace des abonnements et des paiements en ligne.

En outre, le site sera conçu pour offrir une expérience mobile optimisée, garantissant ainsi la disponibilité et l'accessibilité à tout moment et en tout lieu.

Pour cela notre site web doit répondre aux fonctionnalités suivantes :

**Côté client :**

- Inscription et Connexion.
- Réservation de Séances.
- Gestion d'Adhésion.
- Paiement des adhésions et génération de factures.

**Côté admin :**

- Administrer les comptes utilisateurs.
- Contrôler les réservations.
- Créer et modifier les plans d'adhésion.
- Analyser les données et générer des rapports.

### **1.2.5 Conclusion**

En conclusion de ce chapitre, nous avons exposé le contexte général du projet, en présentant des espaces de fitness et des salles de sport, en soulignant la problématique liée à leur gestion. Nous avons également introduit la notion de plateforme de salles de sport en ligne. Les objectifs du projet sont de créer une solution numérique complète qui simplifie la gestion des adhésions, des réservations, de la facturation, et de la communication pour les salles de sport. Notre but est de fournir une interface conviviale tant pour les membres que pour les administrateurs, améliorant ainsi l'expérience globale des utilisateurs tout en optimisant les opérations internes des salles de sport.

# Chapter 2

## Analyse des besoins et conception

### 2.1 Introduction

La conception est une phase essentielle du projet et clé de la réalisation de l'application. Ce chapitre sera consacré à la conception, nous analyserons et identifierons les besoins (identifier les acteurs), la méthodologie et l'approche UML adoptées, développerons des modèles et des diagrammes UML : diagrammes de classes, cas d'utilisation, séquences, pour le développement de bases de données

### 2.2 Analyse et spécification des Besoins

Dans cette partie, nous analyserons les besoins et les spécifications,y compris :

#### 2.2.1 Identification des acteurs

Un acteur représente un rôle joué par une entité externe qui interagit directement avec le système étudié. Dans notre cas, nous avons trois acteurs :

**Admin** : La personne responsable de la gestion du système de l'application Web.

**Member** : La personne inscrite dans la base de données et qui a accès à tous les moyens (classes, équipement,...etc) de l'application web.

**Coach** : La personne inscrite dans la base de données qui communique avec les utilisateurs pour les former.

**Visiteur** : c'est une personne non enregistrée dans la BDD (qui accède au site pour la première fois).

### **2.2.2 Les besoins fonctionnels**

Le système comportera des différentes fonctionnalités nécessaires pour une meilleure gestion. Le site doit accomplir les traitements suivants :

Gestion des utilisateurs : inscription, connexion, authentification pour différents rôles.

Système de réservation : vérification de la disponibilité, création, annulation, modification des réservations.

Gestion des adhésions : types d'adhésion, prix, durée, suivi de l'état de l'adhésion.

Facturation et paiement : génération de factures, gestion des méthodes de paiement, suivi du statut de paiement.

Planification des cours : gestion des horaires, des instructeurs, et de la capacité.

### **2.2.3 Les besoins non fonctionnels**

Une fois les besoins fonctionnels sont bien définis, les besoins non fonctionnels doivent être pris en compte tout au long du processus de développement du site à savoir.

Utilisabilité : interface utilisateur intuitive et accessible.

Sécurité : cryptage des données sensibles, authentification à deux facteurs.

Scalabilité : capacité à évoluer avec l'augmentation des utilisateurs et des réservations.

Performance : temps de réponse rapide, gestion des requêtes simultanées.

Maintenance : Facilité de mise à jour et de déploiement de nouvelles fonctionnalités.

## **2.3 Présentation d'UML**

Le langage UML (Unified Modeling Language) est constitué de diagrammes intégrés utilisés par les développeurs informatiques pour la représentation visuelle des objets, des états et des processus dans un logiciel ou un système. Le langage de modélisation peut servir de modèle pour un projet et garantir une architecture d'information structurée, il peut également aider les développeurs à présenter leur description d'un système d'une manière compréhensible pour les spécialistes externes.

## 2.4 Types des diagrammes d'UML

Dans notre projet nous allons utiliser seulement les diagrammes suivants :

- **Diagramme de cas d'utilisation.**
- **Diagramme de séquence.**
- **Diagramme de classe.**

### 2.4.1 Diagramme de cas d'utilisation

Le Diagramme de cas d'utilisation représente la structure des fonctionnalités nécessaires aux utilisateurs du système. Il est utilisé dans les deux étapes de capture des besoins fonctionnels et techniques.

**Diagramme de cas d'utilisation « Admin » :**

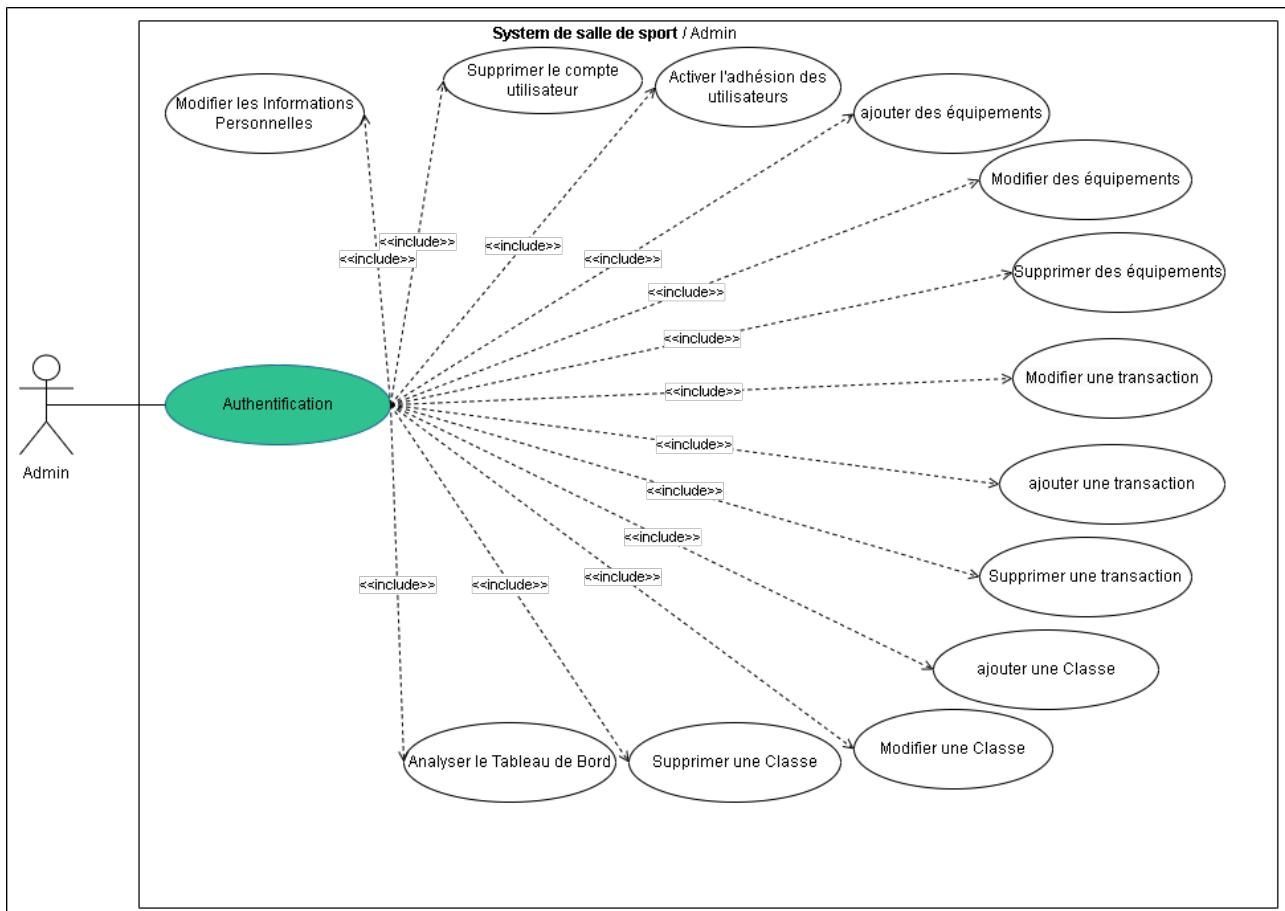


Figure 2.1: Diagramme de cas d'utilisation « Admin » .

**Diagramme de cas d'utilisation « utilisateur/client » :**

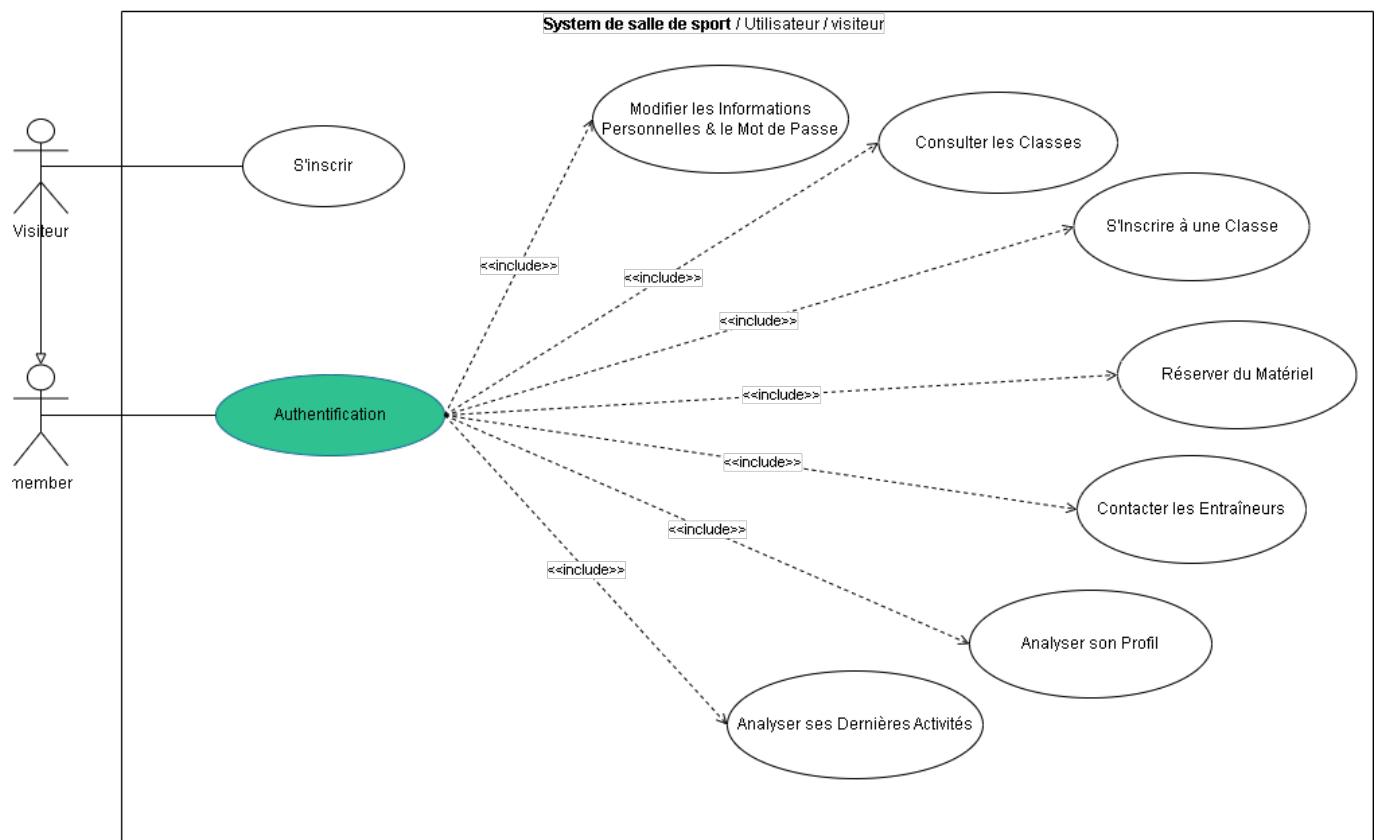


Figure 2.2: Diagramme de cas d'utilisation « utilisateur/client ».

**Diagramme de cas d'utilisation « Coach » :**

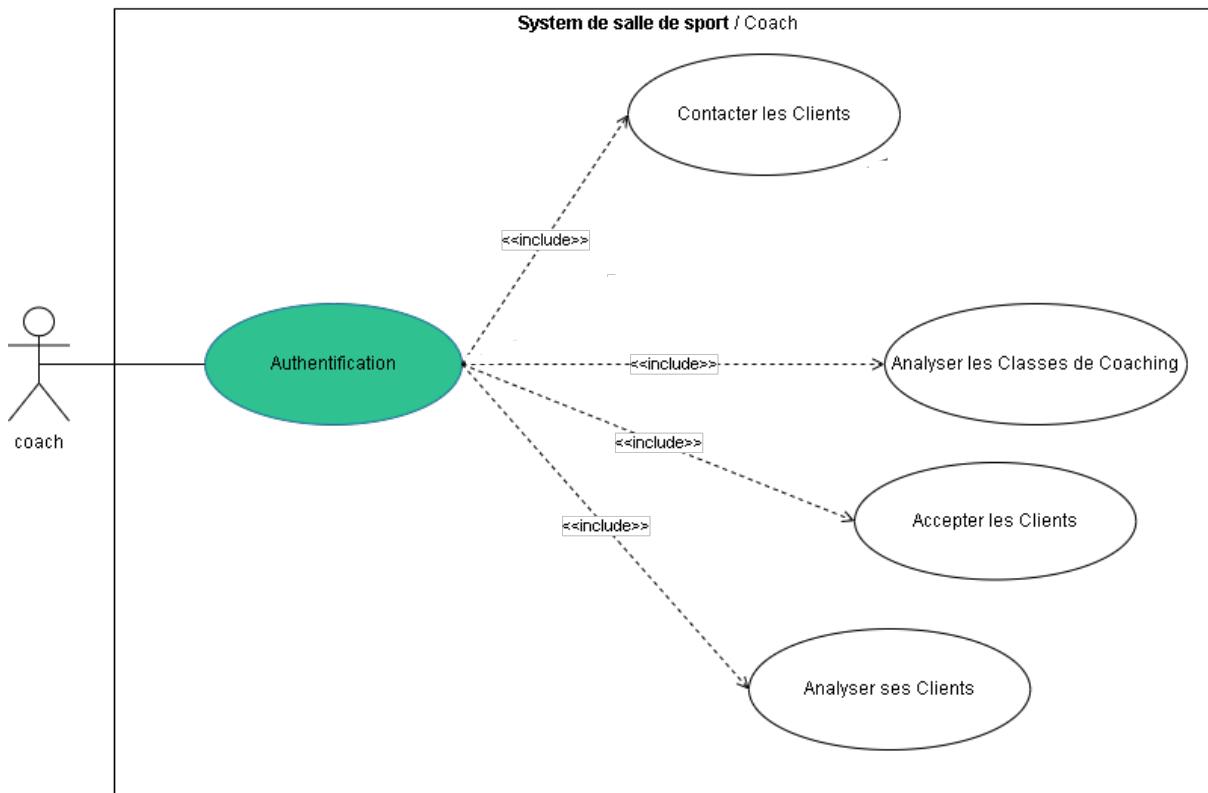


Figure 2.3: Diagramme de cas d'utilisation « Coach ».

#### 2.4.2 Diagramme de séquence

Un diagramme de séquence est un type de diagramme d’interaction. Il représente les échanges de messages entre objets dans le cadre d’un fonctionnement particulier du système. Les diagrammes de séquence sont utilisés pour développer les scénarios d’utilisation du système lors de l’analyse. Voici les composants principaux d’un diagramme de séquence :

**Ligne de vie :** Représente l’ensemble des opérations exécutées par un objet.

**Message synchrone (flèche avec extrémité pleine) :** L’émetteur reste en attente de la réponse à son message avant de poursuivre ses actions.

**Message asynchrone (flèche avec une extrémité non pleine) :** L’émetteur n’attend pas la réponse à son message et poursuit l’exécution de ses opérations.

Pour chaque cas d’utilisation, nous avons développé un diagramme de séquence modélisant l’ensemble des interactions :

**Diagramme de séquence pour le cas « inscrire » :**

Le diagramme de séquence suivant représente le cas d'utilisation d'inscrire pour les nouveaux utilisateurs :

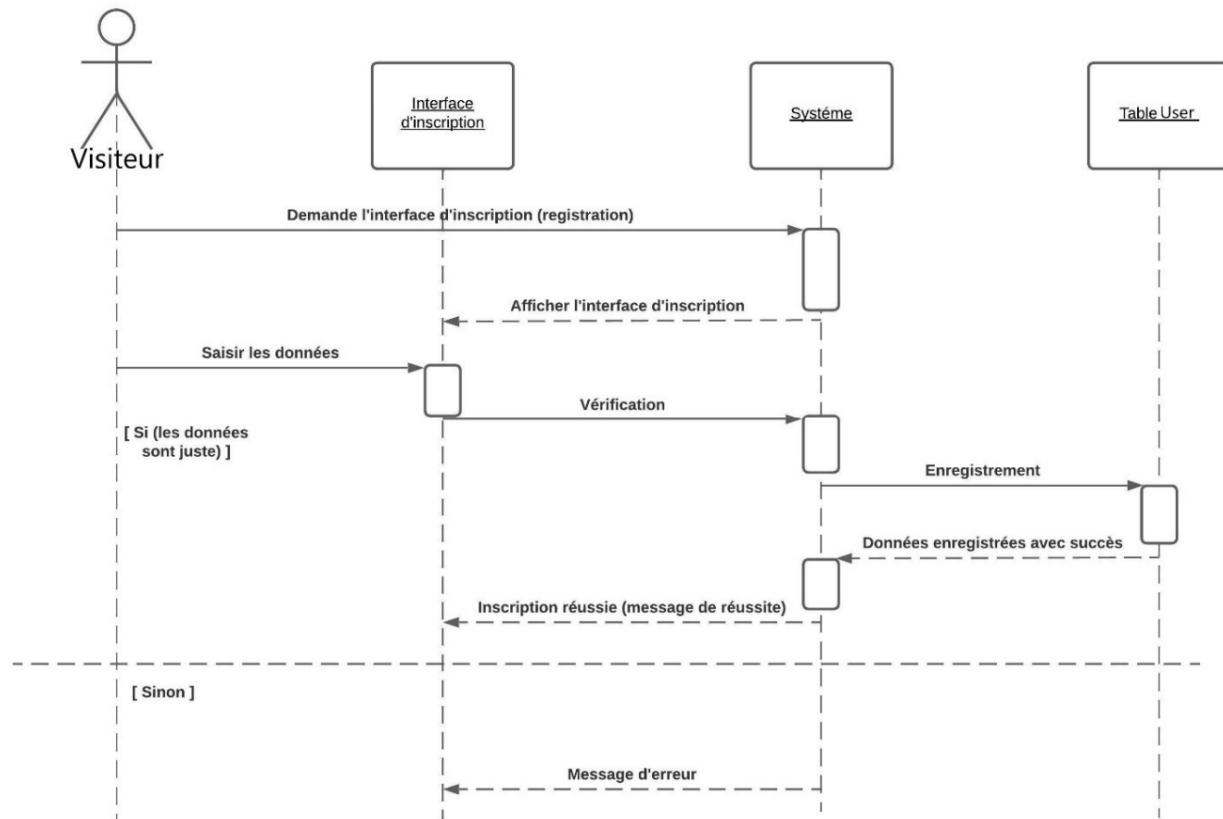


Figure 2.4: Diagramme de séquence « Inscription ».

**Description de l'état d'Inscription :**

Pour profiter des priviléges utilisateurs, chaque visiteur doit s'inscrire avec succès en suivant ces séquences :

- 1. Demande du formulaire d'inscription :** L'utilisateur demande le formulaire.
- 2. Affichage du formulaire :** Le système affiche le formulaire.
- 3. Remplissage du formulaire :** L'utilisateur remplit le formulaire.
- 4. Vérification des données :** Le système vérifie les données.
- 5. Inscription réussie :** L'utilisateur est inscrit avec succès.
- 6. Retour en cas d'erreur :** Si la vérification échoue, retour à l'étape 3.

**Diagramme de séquence pour le cas « réserver » :**

Le diagramme de séquence suivant représente le cas d'utilisation des réserves, qui est le processus suivi pour effectuer des réservations de class.

**Diagramme de séquence pour le cas « s'inscrire à un class » :**

Le diagramme de séquence suivant représente le cas d'utilisation de S'inscrire à un class.

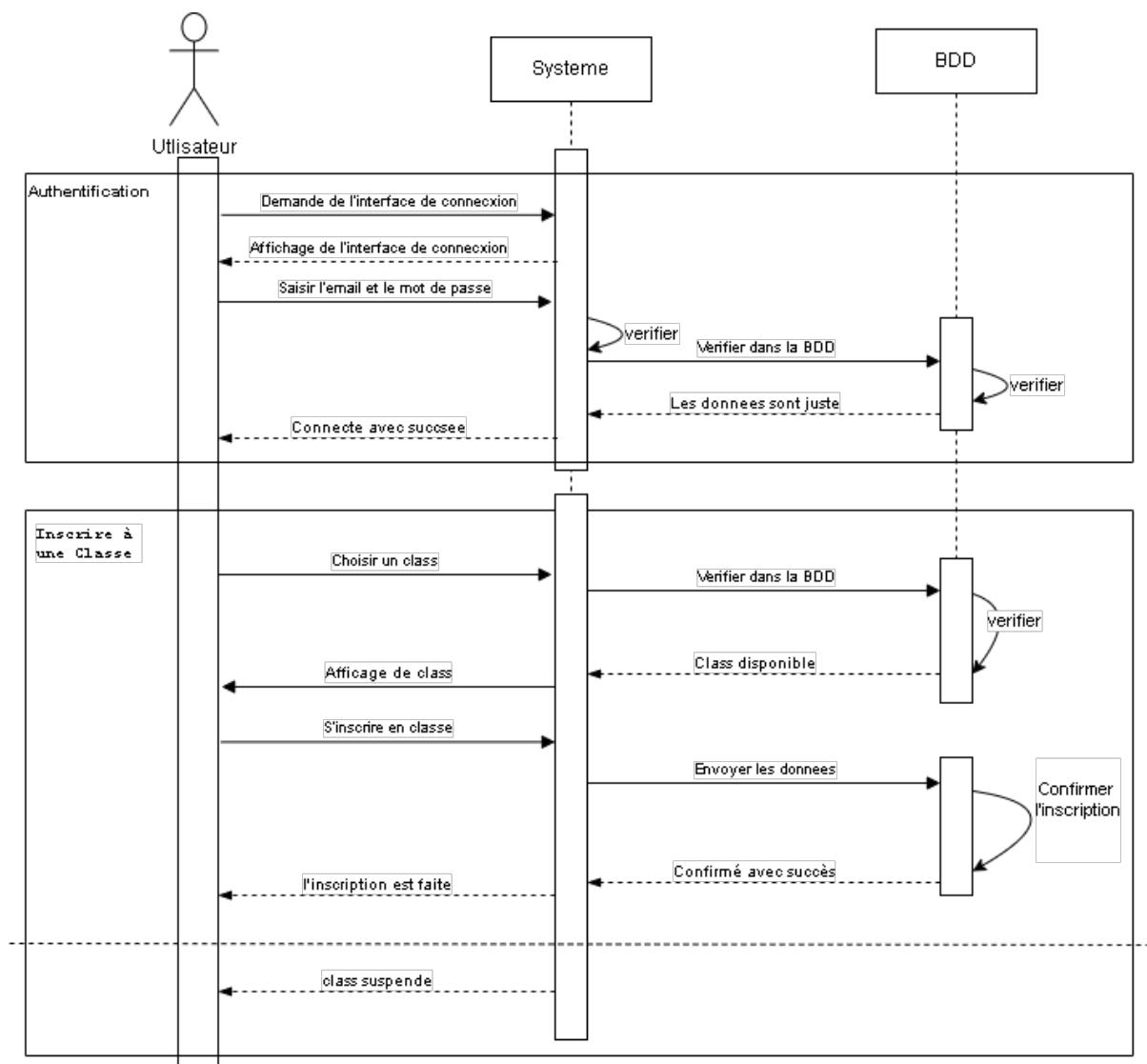


Figure 2.5: Diagramme de séquence « s'inscrire à un class ».

**Diagramme de séquence pour le cas « reserver du materiel » :**

Le diagramme de séquence suivant représente le cas d'utilisation de reserver du materiel.

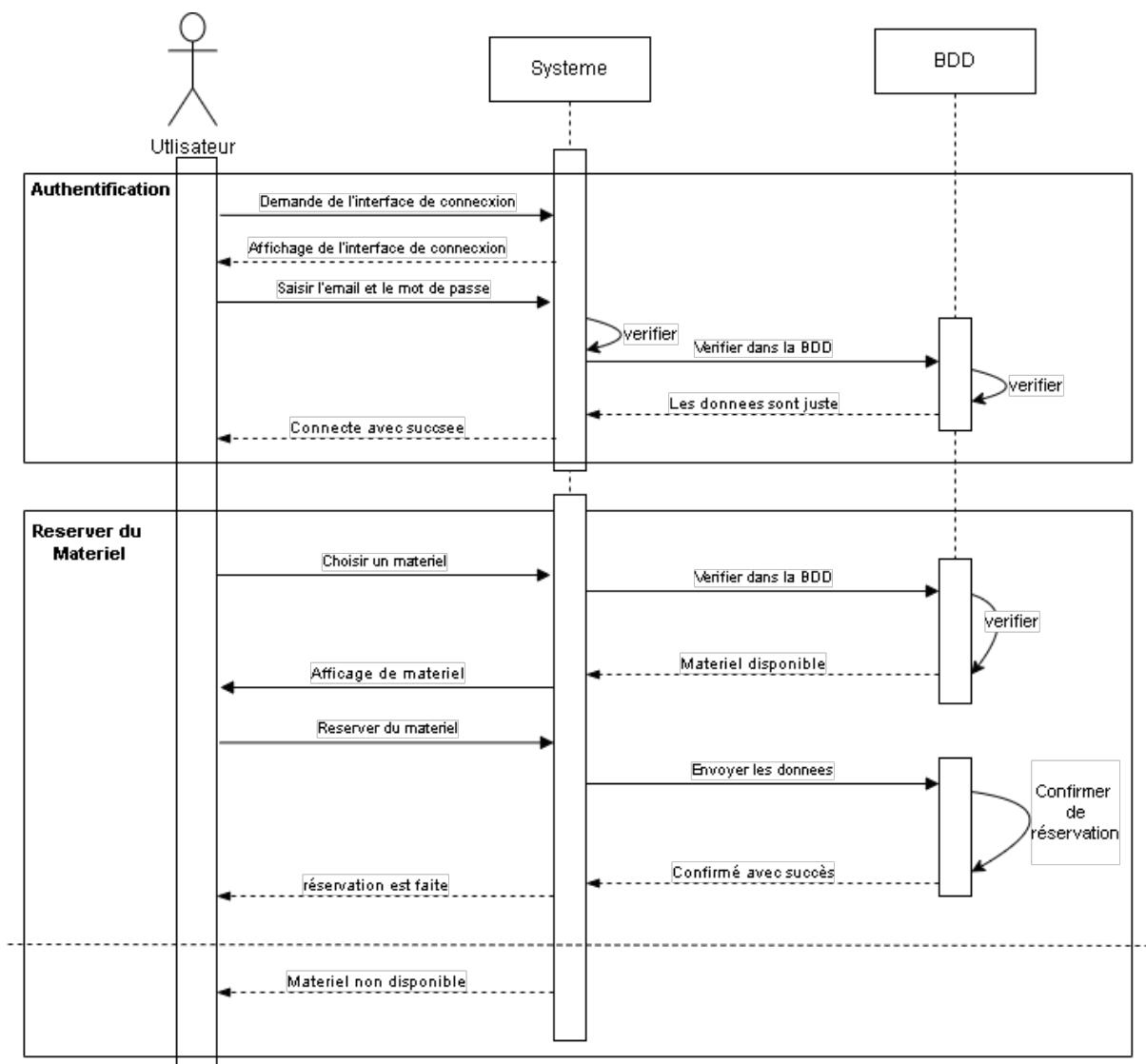


Figure 2.6: Diagramme de séquence « reserver du materiel ».

#### Diagramme de séquence pour le cas « gestion des classes » (Côté Admin) :

Le diagramme de séquence suivant représente le cas d'utilisation de gérer les classes ainsi que la démarche suivie par l'admin pour authentifier.

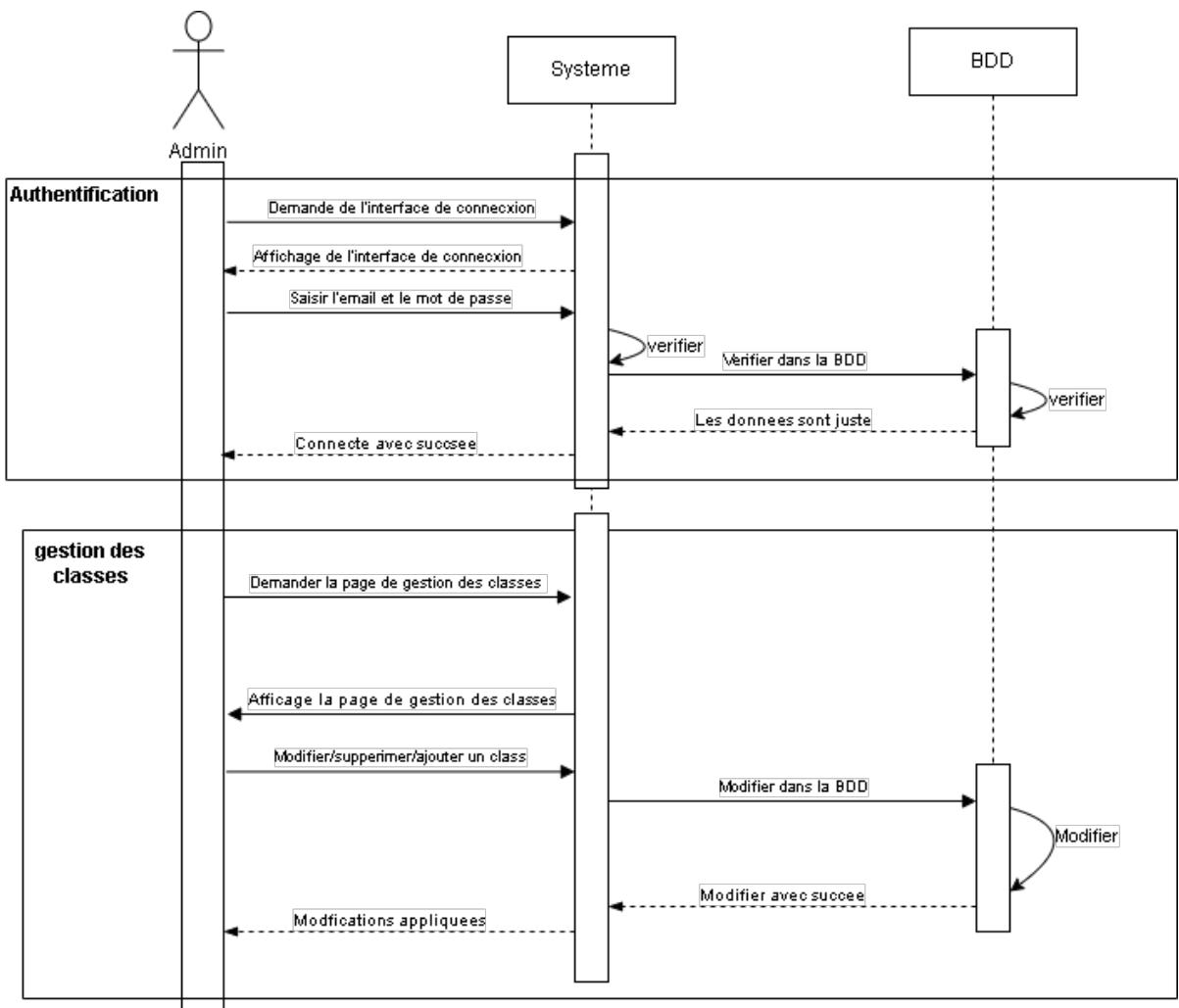


Figure 2.7: Diagramme de séquence « gestion des classes ».

**Diagramme de séquence pour le cas « réservation du matériel » (Côté Admin) :**

Le diagramme de séquence suivant représente le cas d'utilisation de gérer les réservation du matériel.

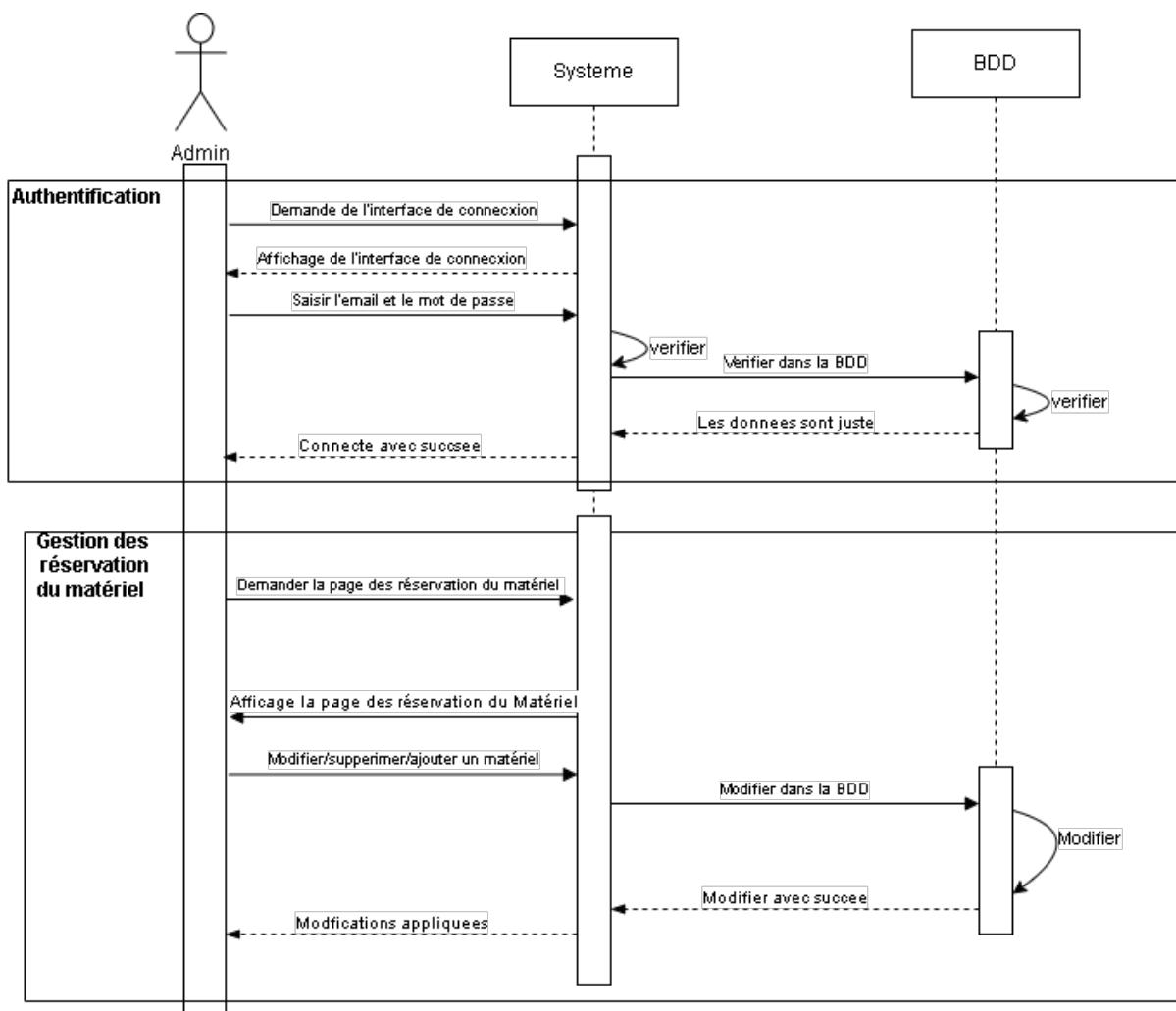


Figure 2.8: Diagramme de séquence « réservation du matériel ».

### 2.4.3 Diagramme de classe

Diagramme de classe a toujours été le plus important dans toutes les méthodes orientés objet. C'est également celui qui contient la plus grande gamme de notations et de variantes centralise l'organisation des classes de conception, c'est lui qui se transforme le plus aisément en code.

La figure suivante représente le diagramme de classe de notre projet :

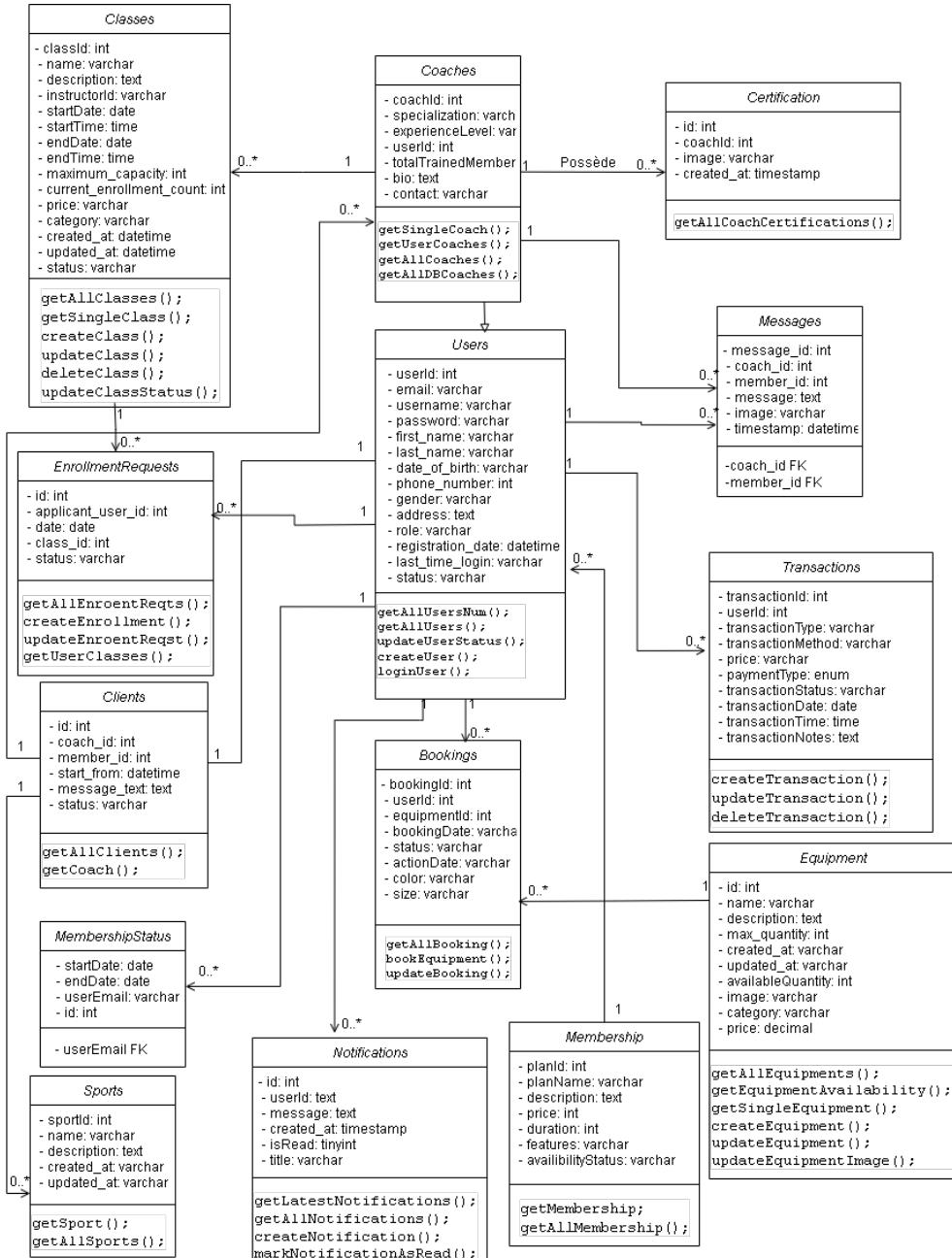


Figure 2.9: Diagramme de classe.

## 2.5 Conception de la base de données

Pour établir la base de données d'un système, nous avons d'abord identifié les entités potentiellement impliquées dans l'application. À partir de ces entités, en respectant les principes du modèle relationnel, nous avons créé les tables nécessaires. Une base de données, quant à elle, est un ensemble structuré de données organisées de manière logique et pérenne pour faciliter la gestion, l'accès et la manipulation des informations. Elle est conçue pour stocker et extraire les données de manière efficace, permettant ainsi une interaction fiable et sécurisée entre les utilisateurs, les applications et les informations.

### **2.5.1 Description du modèle relationnel**

Le modèle relationnel, un pilier fondamental des bases de données, organise les données en tables bidimensionnelles, communément appelées relations ou tables.

Chaque table représente une entité du système, composée de tuples (lignes) contenant des enregistrements spécifiques et de colonnes représentant les attributs de ces enregistrements et pour garantir la validité et la cohérence des données, le modèle relationnel utilise des contraintes d'intégrité telles que les clés primaires, les clés étrangères et les contraintes d'unicité.

Les bases de données relationnelles, administrées par des systèmes de gestion de bases de données relationnelles (SGBDR) tels qu'Oracle, MySQL, Microsoft SQL Server et PostgreSQL, exploitent ce modèle. Le langage universel pour interagir avec ces bases de données est le SQL (Structured Query Language). Grâce à l'algèbre relationnelle, les opérations comme la sélection, la projection et la jointure permettent d'interroger efficacement les données stockées dans ces bases de données relationnelles.

## **2.6 Conclusion**

La conception de l'application est la partie la plus importante de l'application à produire.

Dans ce chapitre Nous sommes concentrés sur les parties analytiques et conceptuelles de notre système, différents schémas de modélisation qui sont des cas de l'utilisation, les diagrammes de séquence ainsi que le diagrammes de classes pour notre application.

Le chapitre suivant sera consacré à réalisation et implémentation de l'application.

# Chapter 3

## Réalisation de l'application

### 3.1 Introduction

Tout développement de projet informatique nécessite le choix des technologies adéquates à son implémentation. Et c'est en définissant les outils de développement que nous débutons ce dernier chapitre. Par la suite, nous allons présenter les interfaces de notre site web afin de mettre en évidence leurs aspect pratique et intuitive qui nous ont été l'un de nos principaux objectifs.

### 3.2 Application web

Les applications web sont des plateformes logicielles accessibles via un navigateur web, offrant une variété de fonctionnalités sans nécessiter de téléchargement ou d'installation. Elles sont devenues essentielles dans notre monde numérique en offrant une accessibilité depuis n'importe quel appareil connecté à Internet.

Dans ce contexte, leur atout réside dans l'accessibilité universelle qu'elles offrent. Les utilisateurs peuvent ainsi profiter des fonctionnalités de l'application sans se soucier de la compatibilité du système d'exploitation ou de l'espace de stockage disponible sur leur appareil. De plus, les applications web sont souvent plus simples à maintenir et à mettre à jour, car les mises à jour sont déployées côté serveur et sont immédiatement disponibles pour tous les utilisateurs.

Le développement d'une application web implique l'utilisation de technologies telles que HTML, CSS et JavaScript pour la création de l'interface utilisateur et l'interaction avec le navigateur. Les frameworks et les bibliothèques populaires comme React, Angular et Vue.js simplifient le processus de développement en fournissant des outils puissants et des modèles de conception réutilisables. Les

développeurs peuvent également utiliser des langages côté serveur comme Node.js ou des frameworks comme Django et Ruby on Rails pour gérer la logique métier et l'accès aux données.

## 3.3 Environnement de travail

### 3.3.1 L'environnement matériel

Afin de pouvoir réaliser notre projet, nous avons utilisé comme environnement machine un ordinateur HP T4PDER4 sous Windows 10 Pro avec un processeur i7-6600U, un disque dur SSD de 250 Go et une mémoire vive de 8 Go.

### 3.3.2 L'environnement logiciel

#### Xampp :

XAMPP est un ensemble de logiciels libres. Le nom est un acronyme venant des initiales de tous les composants de cette suite. Ce dernier réunit le serveur Web Apache, la base de données relationnelles et système d'exploitation My SQL (ou Maria DB) ainsi que les langages scripts Perl et PHP. L'initiale X représente tous les systèmes d'exploitation possibles, à savoir Linux, Windows et Mac OSX .

#### MySQL :

MySQL est un système de gestion de base de données relationnelle open-source largement utilisé. Il permet de stocker et de gérer efficacement des données structurées dans des bases de données. MySQL utilise le langage SQL (Structured Query Language) pour exécuter des requêtes, modifier des données et gérer la structure des bases de données. Il prend en charge de nombreuses fonctionnalités telles que les transactions ACID (Atomicité, Cohérence, Isolation, Durabilité), les index, les vues, les procédures stockées, les déclencheurs, la réPLICATION, la sécurité des données et bien d'autres encore. En raison de sa fiabilité, de sa performance et de sa facilité d'utilisation, MySQL est largement utilisé dans diverses applications web, des sites web simples aux grandes applications d'entreprise.



Figure 3.1: MySQL.

### **Node.js :**

Node.js est une plateforme logicielle open-source permettant d'exécuter du code JavaScript côté serveur, offrant une gestion efficace des requêtes clients et l'accès aux ressources du serveur comme les bases de données et les fichiers système. Son modèle d'entrées-sorties non bloquantes et asynchrones lui confère une grande extensibilité, idéale pour les applications web en temps réel telles que les chats en ligne ou les jeux multi-joueurs. Souvent utilisé avec des frameworks comme Express.js, Node.js facilite le développement web et la construction d'API, de serveurs de microservices, d'outils de développement et bien d'autres applications, bénéficiant d'un riche écosystème de modules grâce au gestionnaire de paquets npm.



Figure 3.2: Node.js.

### **Visual Studio Code :**

Visual Studio Code est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'autocomplétions, la coloration

syntaxique, le débogage, et les commandes git. Il peut être utilisé avec une variété de langages de programmation, notamment Java, JavaScript, Go, Node.js et C++. Il est basé sur le cadre Electron, qui est utilisé pour développer des applications Web.

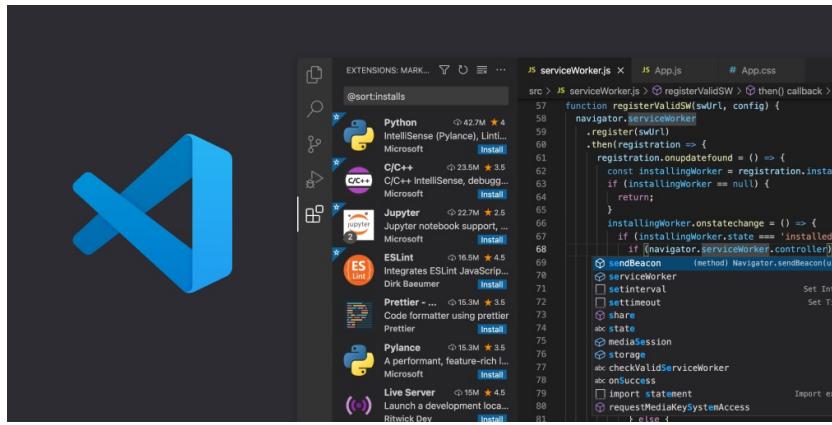


Figure 3.3: Visual Studio Code.

## React.js :

React.js, ou simplement React, est une bibliothèque JavaScript populaire conçue pour créer des interfaces utilisateur interactives et dynamiques. Développée par Facebook, elle offre une approche déclarative pour la construction de composants réutilisables, simplifiant ainsi le développement d'applications web modernes. React permet aux développeurs de gérer efficacement les mises à jour d'interface utilisateur en réponse aux changements d'état, ce qui améliore les performances et la maintenabilité du code. Avec sa large adoption et sa communauté active, React est devenu un choix privilégié pour le développement d'applications web évolutives.

## Postman :

Postman est un outil essentiel pour les développeurs d'API, offrant une interface conviviale pour tester, déboguer et documenter les API. Il permet d'envoyer des requêtes HTTP, de gérer les en-têtes et les authentifications, et d'analyser les réponses des serveurs. Avec des fonctionnalités avancées comme la création de collections d'API et l'automatisation des tests, Postman est devenu indispensable pour le développement d'applications web et mobiles.

## Git :

Git est un système de contrôle de version distribué largement utilisé pour suivre les modifications dans les fichiers et coordonner le travail sur ces fichiers parmi plusieurs personnes.

## Tailwind :

Tailwind CSS est un framework CSS utilitaire qui permet de construire des interfaces utilisateur

modernes et réactives de manière rapide et efficace. Contrairement aux frameworks CSS traditionnels comme Bootstrap ou Foundation, qui fournissent des composants prêts à l'emploi .

### 3.3.3 Langages de programmation

#### HTML :

Le langage HTML, abréviation de HyperText Markup Language, est le fondement de la création de pages web. Utilisant des balises pour définir la structure et le contenu, il permet de créer des éléments tels que des titres, des paragraphes, des images et des liens. Chaque balise a une fonction spécifique, indiquant au navigateur comment afficher le contenu. Associé à CSS pour la mise en forme et à JavaScript pour l'interactivité, HTML constitue la base de la conception web moderne.

#### CSS :

Le CSS, ou Cascading Style Sheets, est un langage de feuille de style utilisé en tandem avec HTML pour styliser les éléments d'une page web. En définissant des règles de style, il contrôle des aspects tels que la couleur, la police, la taille et la disposition des éléments HTML. Cette séparation entre le contenu et la présentation visuelle permet aux développeurs de créer des designs attrayants et cohérents pour leurs sites web, offrant ainsi une plus grande flexibilité dans la conception et la maintenance des pages web.

#### JavaScript :

JavaScript est un langage de programmation essentiel pour le développement web, permettant d'ajouter des fonctionnalités interactives et dynamiques aux pages web. Intégré directement dans les navigateurs, il offre la possibilité de créer des animations, des formulaires dynamiques et des interactions en temps réel avec les utilisateurs. Sa polyvalence s'étend également au développement d'applications mobiles, de jeux et de logiciels côté client, ce qui en fait l'un des langages les plus populaires et influents dans le domaine du développement logiciel.

## 3.4 Présentation des interfaces de notre application

Dans les paragraphes qui suivent, nous allons exposer les différentes situations que peuvent renconter le client et l'administrateur de notre projet.

### 3.4.1 Coté Administrateur

#### Interface d'authentification :

Sur cette interface, l'administrateur doit saisir son email, et un son mot de passe, puis cliquer sur le bouton (Login) pour authentifier.

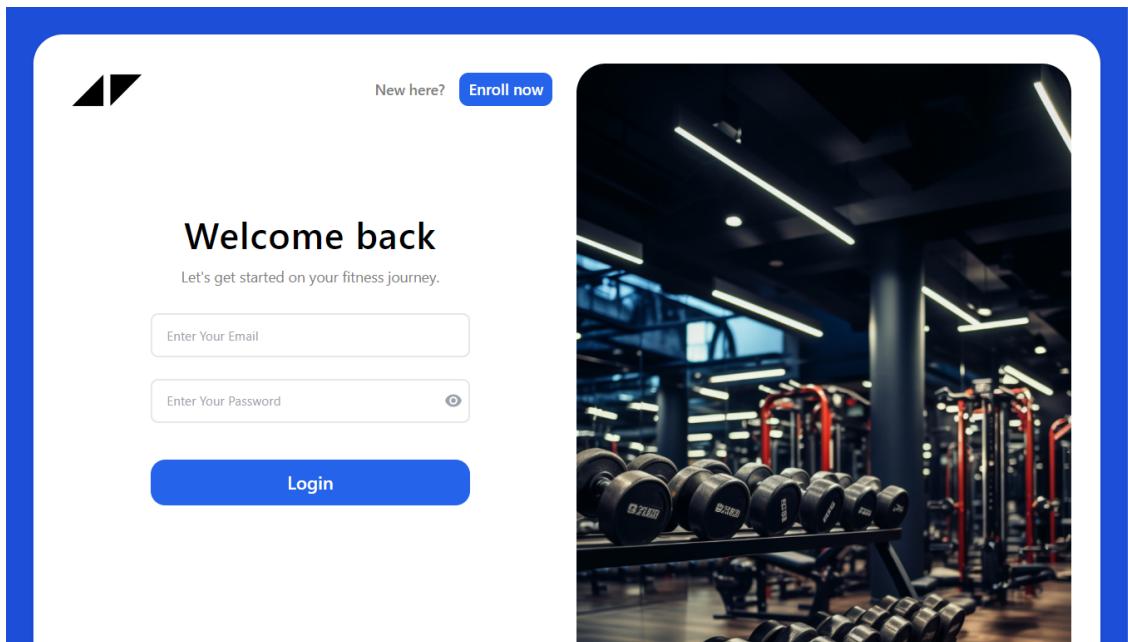


Figure 3.4: Interface d'authentification .

#### Interface de contrôle utilisateur :

L'administrateur peut contrôler les utilisateurs en les supprimant ou en activant leur adhésion.

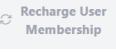
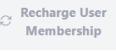
NAME	ROLE	STATUS	ACTIONS
 ieBreffsdrldn@gmail.com	admin	active	
 alaa123@gmail.com	admin	active	
 zarhoun zarhouni zarhouni@gmail.com	member	active	
 Alaa dfsd adsa@gmail.com	member	active	

Figure 3.5: Interface de contrôle utilisateur .

### Interface de statistiques générales du club :

L'administrateur peut effectuer une analyse complète des données en connaissant les statistiques générales du club.

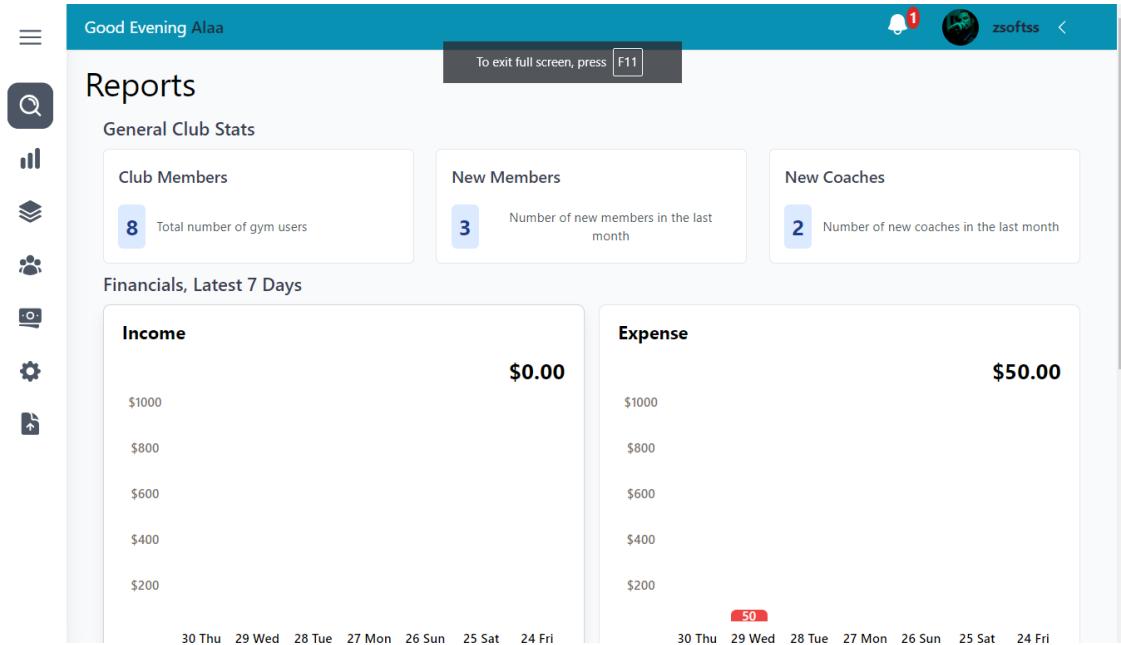


Figure 3.6: Interface de statistiques générales du club .

### Interface des réservations :

L'administrateur peut voir toutes les réservations.

The screenshot shows the 'All Reservations' page with a title 'All Reservations (24)'. A 'Filter' button is located in the top right corner. The main content is a table with columns: BOOKING ID, BOOKING DATE, TIME, STAFF, USER TYPE, STATUS, and PRICE. The table lists 24 entries, each with a unique booking ID, date, time, staff member, user type, status (e.g., rejected, pending, approved, reserved), and price. The interface includes a sidebar with icons and a navigation bar at the bottom.

BOOKING ID	BOOKING DATE	TIME	STAFF	USER TYPE	STATUS	PRICE
31	May 28, 2024	1:23:00 PM	member	member	rejected	\$89.11
30	May 28, 2024	11:46:00 AM	coach	coach	PENDING	\$121.61
29	May 26, 2024	2:53:00 PM	member	member	approved	\$90.74
26	May 25, 2024	4:57:00 PM	member	member	approved	\$90.74
25	May 25, 2024	4:55:00 PM	member	member	reserved	\$72.07
24	May 25, 2024	10:59:00 AM	member	member	rejected	\$89.11
23	May 25, 2024	10:57:00 AM	admin	admin	rejected	\$89.11

Figure 3.7: Interface des réservations .

### **Interface pour ajouter des équipements :**

L'administrateur peut ajouter des équipements et connaître la date de sa réservation.

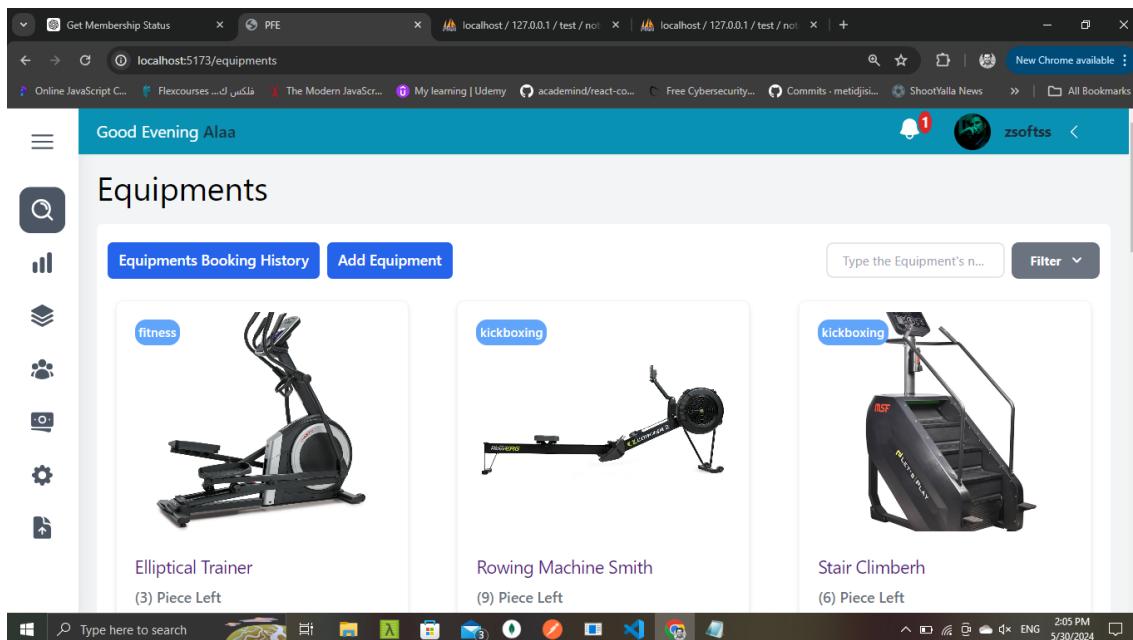


Figure 3.8: Interface pour ajouter des équipements .

### **Interface de paiements :**

L'administrateur peut consulter la liste des paiements et modifier, supprimer ou ajouter une transaction.

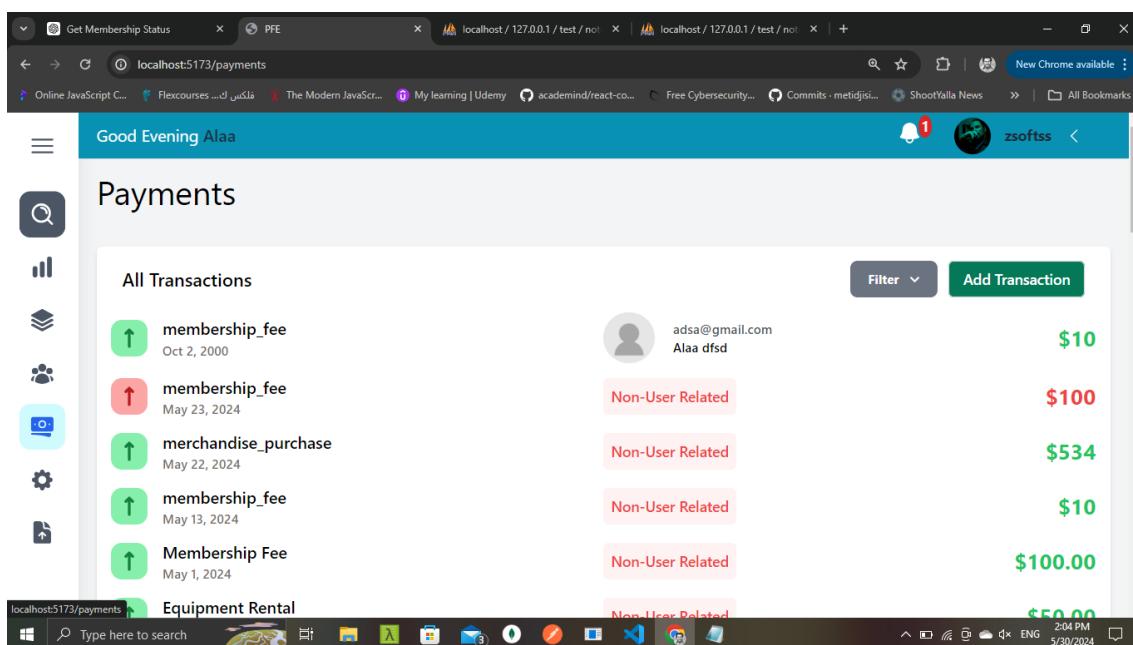


Figure 3.9: Interface de paiements.

### Interface pour ajouter une transaction :

L'administrateur peut ajouter et contrôler une transaction.

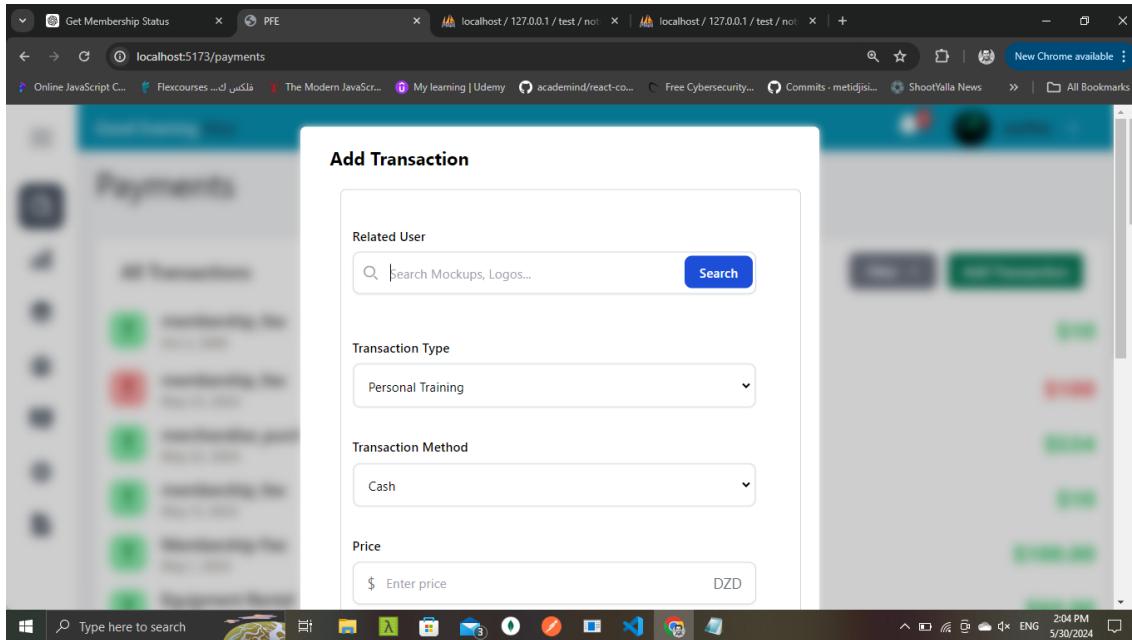


Figure 3.10: Interface pour ajouter une transaction .

### Interface pour ajouter des classes :

L'administrateur peut également ajouter et contrôler des classes.

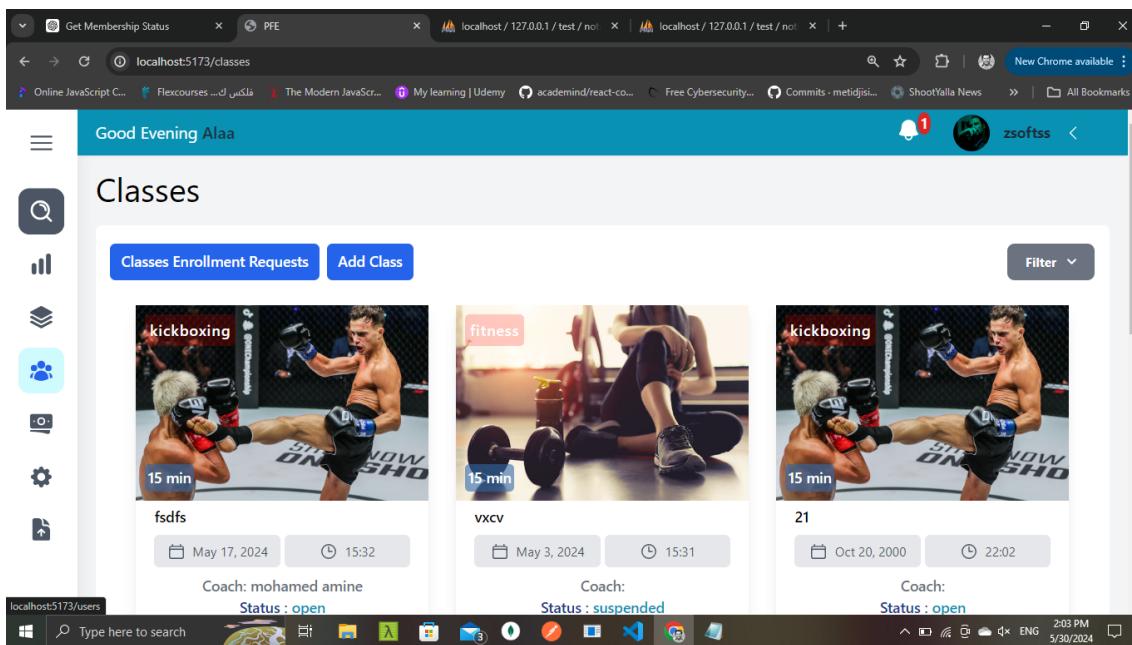


Figure 3.11: Interface pour ajouter des classes .

### Interface du tableau de bord :

L'administrateur peut surveiller et analyser les paiements des adhésions, la location d'équipement et d'autres transactions financières sur une base mensuelle.

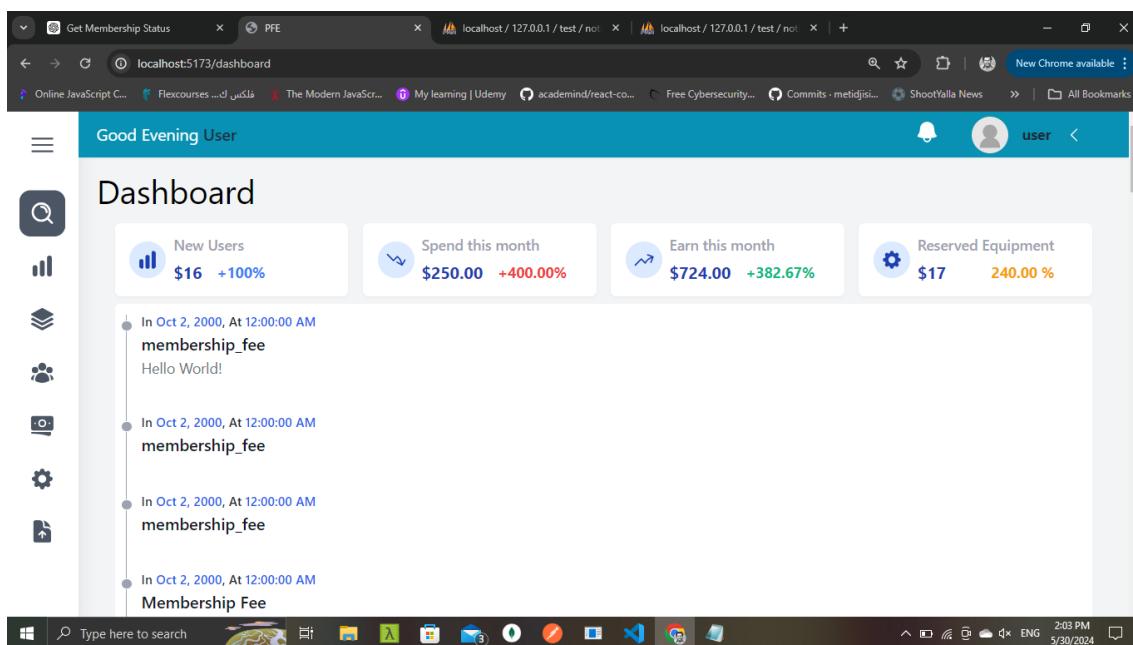


Figure 3.12: Interface du tableau de bord .

### Interface pour communiquer avec les utilisateurs :

L'administrateur peut communiquer avec les utilisateurs.

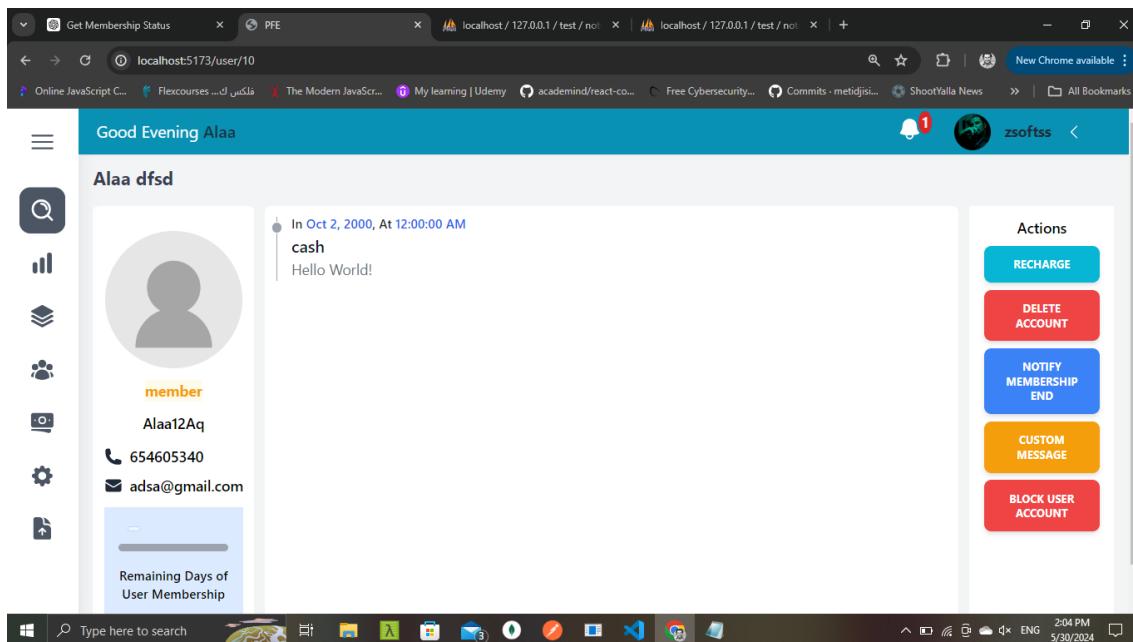


Figure 3.13: Interface pour communiquer avec les utilisateurs.

### 3.4.2 Coté Utilisateur(member)

#### Interface d'inscription :

L'utilisateur doit remplir le formulaire d'inscription.

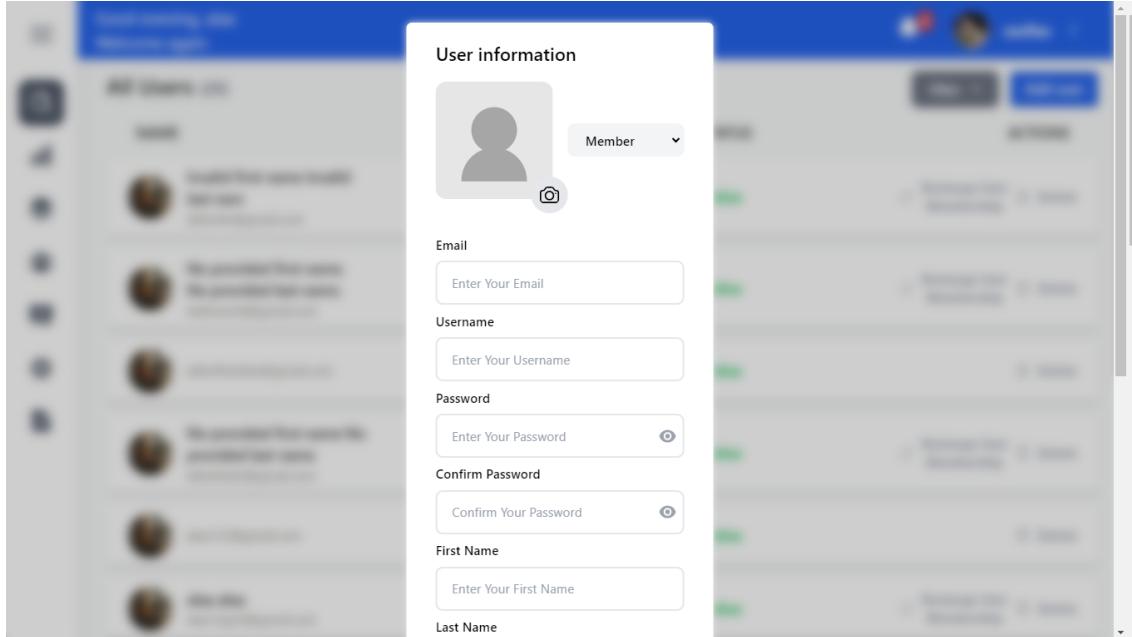


Figure 3.14: Interface d'inscription (Utilisateur).

### Interface de sélection d'équipement :

L'utilisateur peut choisir le matériel qui lui convient et l'ajouter à sa liste de souhaits.

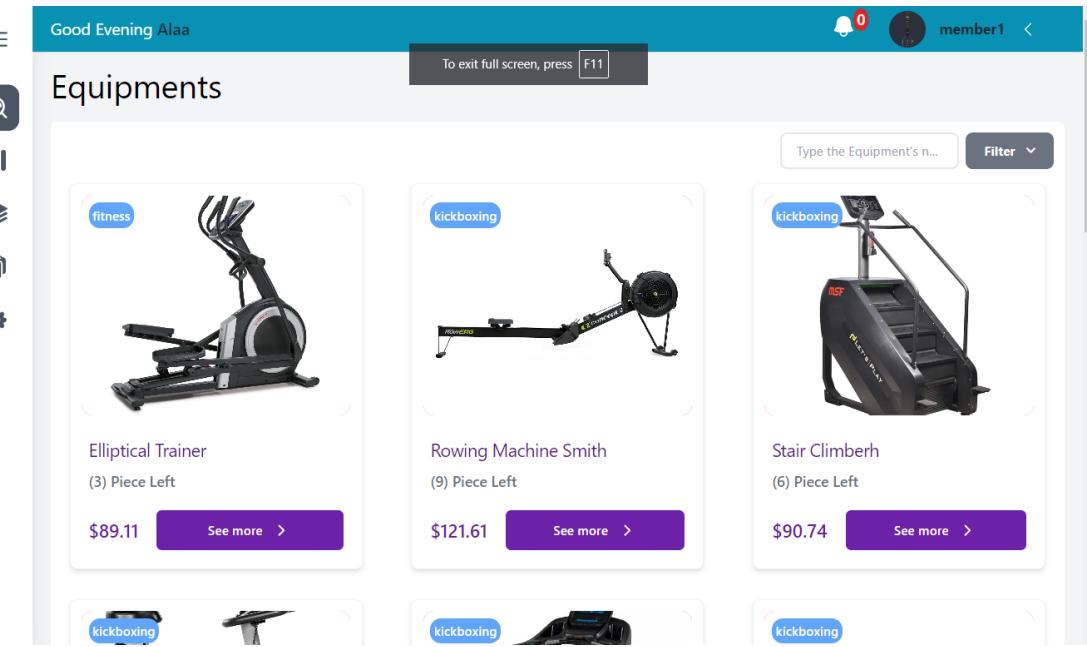


Figure 3.15: Interface de sélection d'équipement (Utilisateur) .

### Interface de sélection des entraîneurs :

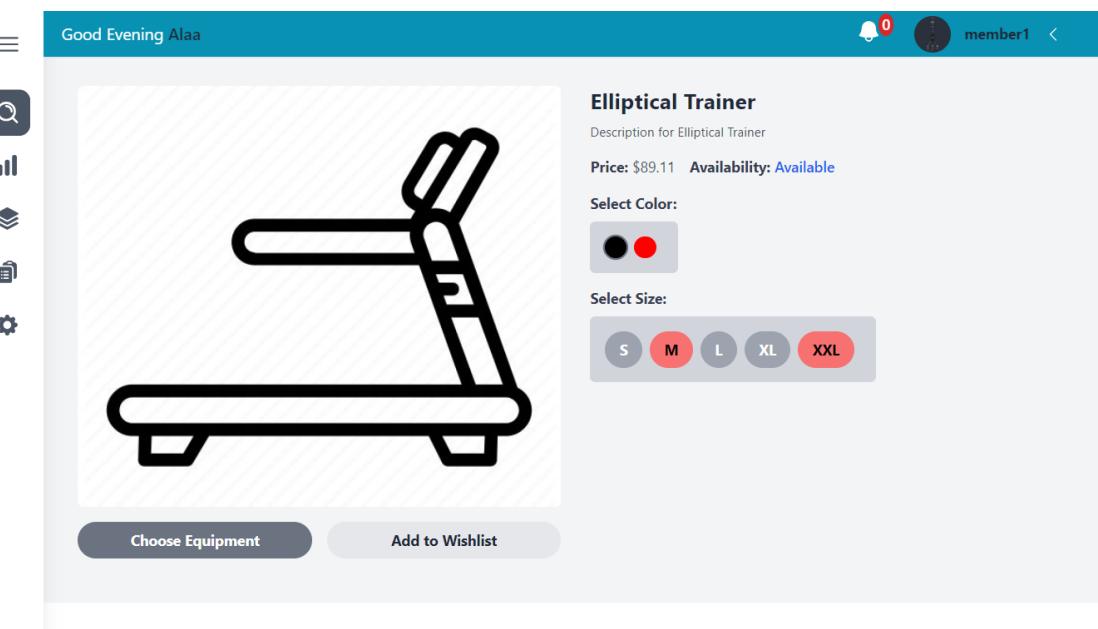


Figure 3.16: Ajouter du matériel à la liste de souhaits (Utilisateur).

L'utilisateur peut communiquer avec les coachs et parcourir leur profil.

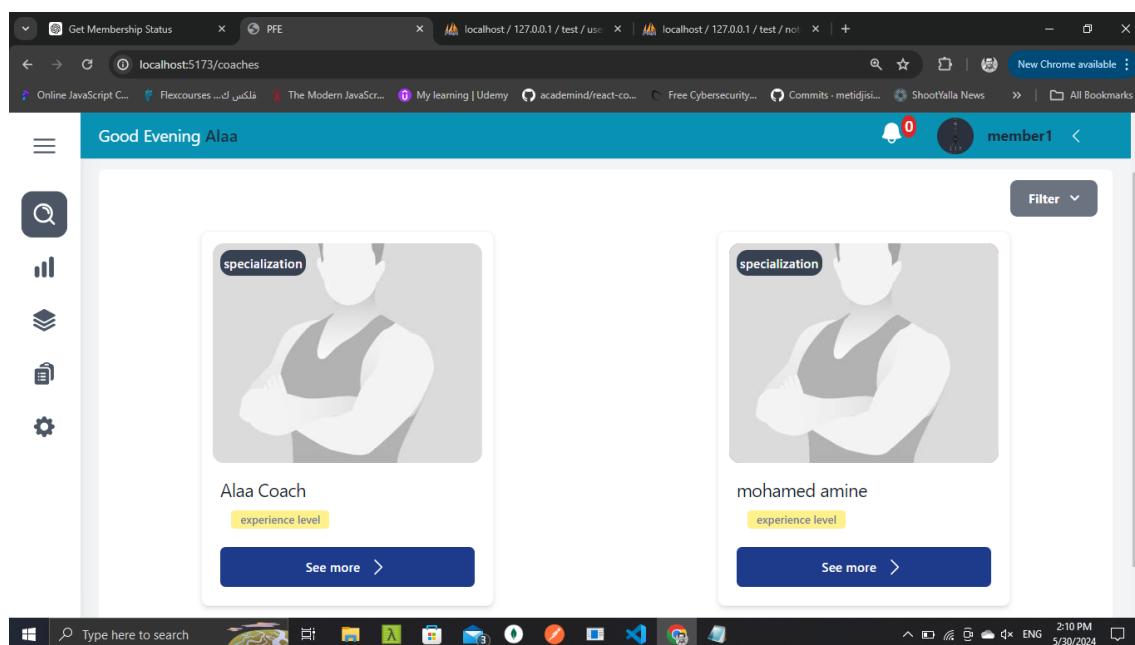


Figure 3.17: Interface de sélection des entraîneurs (Utilisateur).

**Interface de sélection de classe :**

## CHAPTER 3. RÉALISATION DE L'APPLICATION

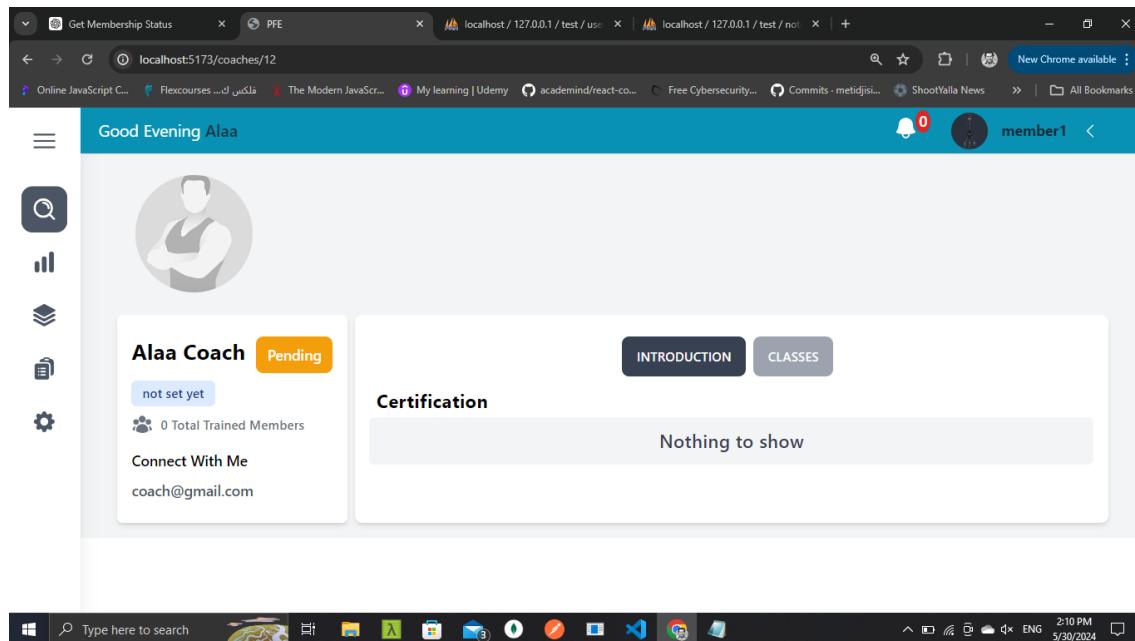


Figure 3.18: L'interface du compte personnel du coach.

L'utilisateur peut choisir class qui lui convient et connaître tout ce que contiennent les classes.

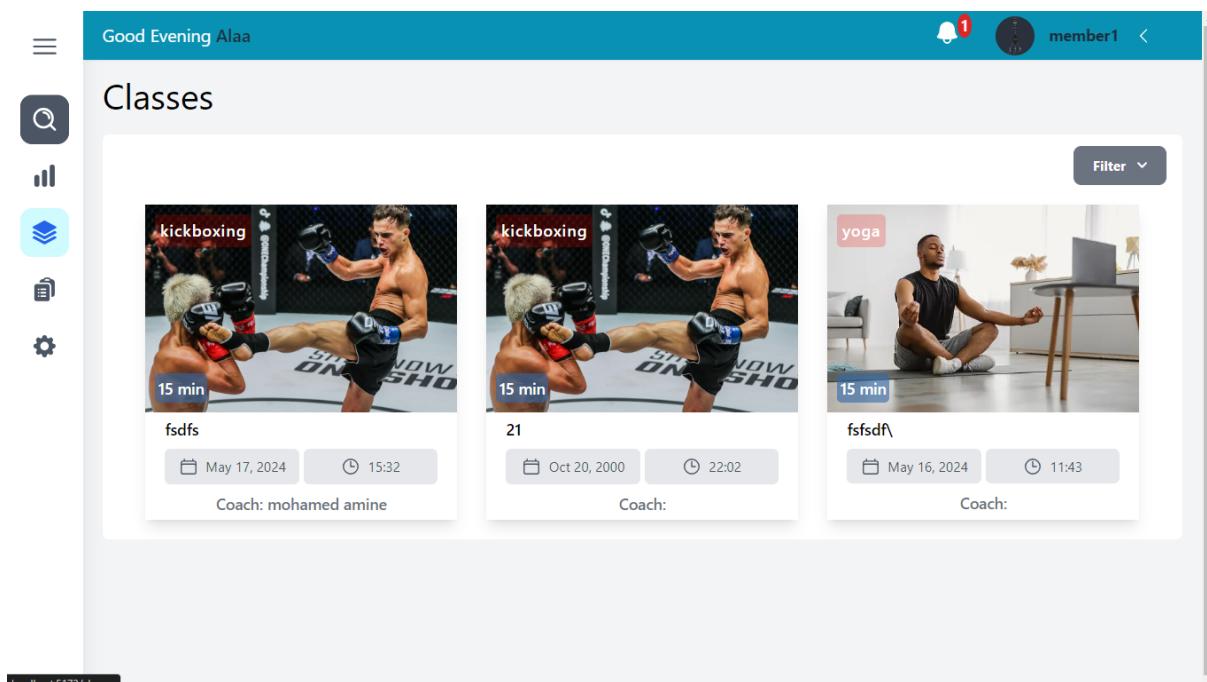


Figure 3.19: Interface de sélection de classe (Utilisateur).

### **Interface du tableau de bord :**

L'utilisateur peut visualiser ses paiements et transactions financières.

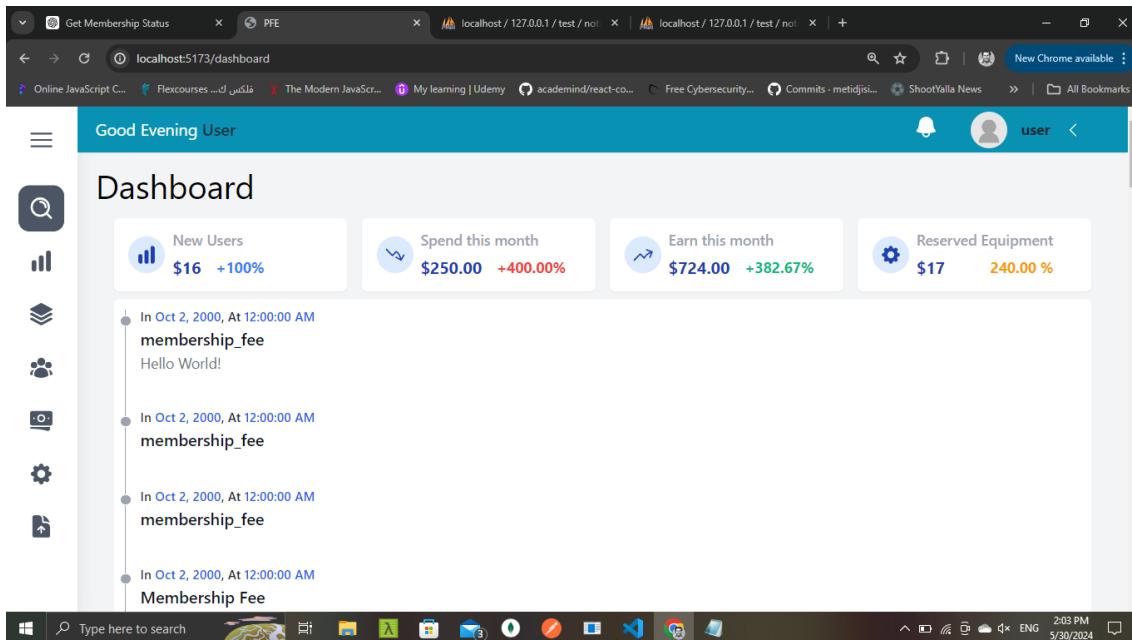


Figure 3.20: Interface du tableau de bord (Utilisateur).

### **Une interface spéciale pour tous les choix des utilisateurs :**

Enfin, l'utilisateur peut visualiser l'ensemble de ses choix mentionnés précédemment.

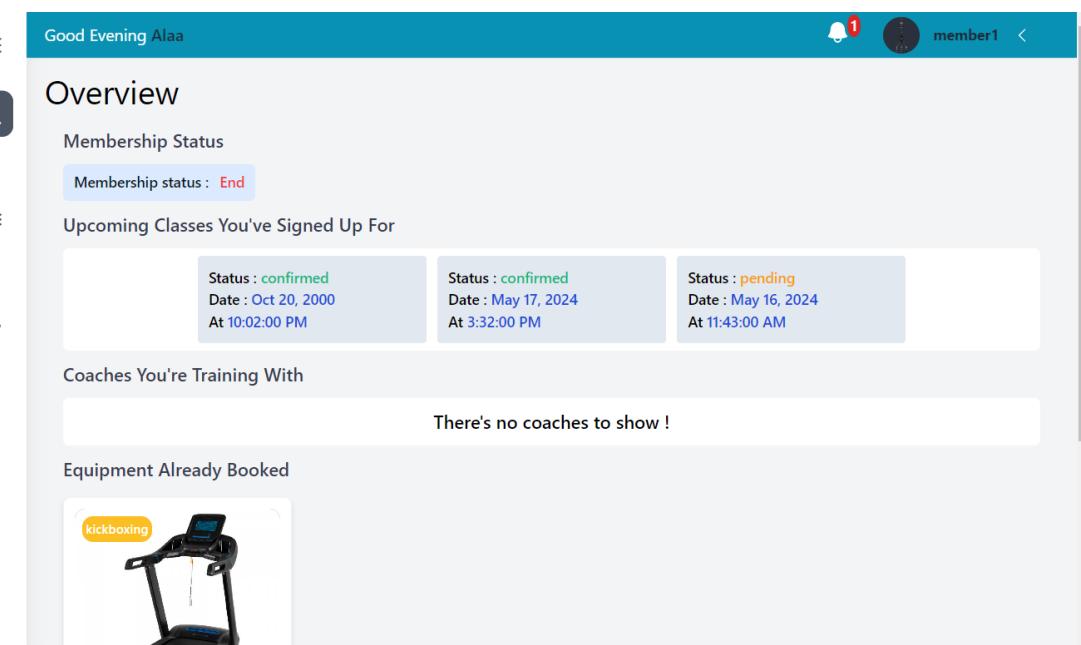


Figure 3.21: Une interface spéciale pour tous les choix des utilisateurs .

## Interfaces du coach

### Interface client :

Le formateur peut voir quels clients ont choisi son cours.

The screenshot shows a mobile application interface for a coach named 'mohamedAmine'. The top bar displays 'Good Evening Mohamed'. On the right, there is a notification icon with a red '1' and the user's name 'mohamedAmine'. The main screen is titled 'Clients' and lists one client entry:

- Profile Picture:** A placeholder gray user icon.
- Name:** alaa alaa
- Email:** member1@gmail.com
- Action:** A blue 'Contact >' button.

The URL 'localhost:5173/clients' is visible at the bottom of the browser window.

Figure 3.22: Interface client .

### Interface pour communiquer avec les membres :

Le formateur peut répondre aux messages des membres.

The screenshot shows a mobile application interface for a coach named 'mohamedAmine'. The top bar displays 'Good Evening Mohamed'. On the right, there is a notification icon with a red '1' and the user's name 'mohamedAmine'. The main screen lists messages from members:

- Rafik Coach:** 2:08:18 PM - Lorem ipsum dolor sit amet consectetur adipisicing elit. Sit a quam est reiciendis, soluta dolor repellat asperiores totam, corporis tenetur nihil aliquam rerum facilis iusto? Nesciunt consectetur facere quos illo.
- Bonnie Green:** 11:46 - Flowbite Terms & Conditions  
12 Pages 18 MB PDF
- Rafik Coach:** 2:08:18 PM - Lorem ipsum dolor sit amet consectetur adipisicing elit. Sit a quam est reiciendis, soluta dolor repellat asperiores totam, corporis tenetur nihil aliquam rerum facilis iusto? Nesciunt consectetur facere quos illo.

A text input field 'Type me anything...' and a send button are located at the bottom right of the message list.

Figure 3.23: Interface pour communiquer avec les membres.

## 3.5 Conclusion

Ce dernier chapitre commence par une description des outils de développement que nous avons employés pour mener à bien notre projet. Par la suite, nous présentons les interfaces les plus cruciales de notre site web. Nous croyons que ce travail satisfait aux besoins définis dans le premier chapitre ainsi qu'aux principales exigences des utilisateurs, à savoir la création d'un site web pour la gestion des espaces de fitness et des salles de sport.

## CONCLUSION GENERALE

Le travail présenté au cours de ce projet a permis de réaliser un site web dynamique dédié à l'accès aux différentes informations et commodités offertes par les espaces de fitness et les salles de sport, tout en facilitant la gestion des services à distance.

Pour atteindre cet objectif, les exigences du système ont été déterminées et étudiées. Il a fallu rechercher, exprimer, analyser et perfectionner la plupart des cas d'utilisation à l'aide de diagrammes UML. La conception a ensuite commencé par la mise en œuvre du diagramme de classes. La réalisation a été effectuée en employant les outils d'implémentation appropriés, intégrant le contenu et le style avec HTML5, JavaScript, React.js et CSS, ainsi que la gestion de la base de données en utilisant MySQL.

Ce projet a été extrêmement bénéfique pour nous, car il nous a permis de gagner une expérience précieuse et de mettre en pratique les connaissances acquises au cours de nos études. Nous avons pu approfondir notre compréhension de l'analyse des besoins et de la conception, tout en découvrant et en développant de nouvelles compétences en programmation et en gestion de bases de données.

Nous espérons que ce travail, bien que modeste, pourra servir d'outil d'aide, de documentation et de base de travail expérimental pour les débutants dans ce domaine.

# BIBLIOGRAPHIE

- [1] **cincontournable de la salle de sport,**[https://www.plusdesport.com/article/fitness park](https://www.plusdesport.com/article/fitness-park),consulté le 06 avril 2024.
- [2] **Salle de sport, Wikipedia, l'encyclopédie libre.** Consulté le 15 avril 2024, à l'adresse [https://fr.wikipedia.org/wiki/Salle\\_de\\_sport](https://fr.wikipedia.org/wiki/Salle_de_sport).
- [3] **Logiciel de gestion de salle de sport,**<https://www.gym-connect.com/>,consulté le 08 avril 2024.
- [4] **Liberfit,** <https://liberfit.fr>, consulté le 25 mai 2024.
- [5] **Article Fitness Today, numéro de mai 2024, pages 12-15**
- [6] **UML : un langage de modélisation de type graphique,**<https://www.ionos.fr/digitalguide/sites-internet/developpement-web/uml-un-langage-de-modelisation-pour-la-programmation-orientee-objet/>, consulté le 20 avril 2024.
- [7] **XAMPP,** <https://fr.wikipedia.org/wiki/XAMPP>, consulté le 01 mai 2024.
- [8] **HTML, CSS, JavaScript, MySQL : petites explications des technologies Web,**<https://www.xp-internet.com/xp-infos/HTML-CSS-JavaScript-PHP-MySQL-petitesexplications-des-technologies-Web.php>, consulté le 03 mai 2024.
- [9] **E.F. Codd ,A Relational Model of Data for Large Shared Data Banks ,**Communications of the ACM, Vol. 13, Juin 1970.
- [10] **Jelly Vincint, Développement d'une application de fitness,** Haute École de Gestion de Genève (HEG-GE) ,Genève, le 15 Septembre 2017.