People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
M'Hamed Bougara University of Boumerdès



Faculty of Science
Department of Computer Science

Field: Mathematics and Computer Science
Program: Computer Science
Specialization: Information System

Theme

SmartSwim : web application for Comprehensive Swimmer Monitoring

Presented by :

- Guechetouli Wafaa
- Raissi Hafsa
- Benchouk Cerine
- Meftah Aymene

Supervisor :
Mr.Ali Chaouche

Examiner :
Mr.HABI Idir

Contents

List of Figures **IV**

List of Tables **V**

I State Of the Art

1 Study Of The Existence **1**

Introduction . . . 1

1.1 Methods used by Algerien coaches . . . 1

1.2 Application for swimming management . . . 2

1.2.1 Commit Swimming . . . 2

1.2.2 MySwimPro . . . 2

1.2.3 Swim Coach . . . 3

1.2.4 SwimTopia . . . 3

1.3 Problem Statement and Proposed Solution . . . 4

Conclusion . . . 4

II Conception and Implementation 5

2 Conception **5**

2.1 UML Modeling . . . 5

2.2 Identification of Actors . . . 5

2.3 Use Case Diagram . . . 5

2.3.1 General Use Case Diagram . . . 6

2.4 Detailed Use Case Diagrams . . . 7

2.4.1 Coach Use Case . . . 7

2.4.2 Athlete Use Case . . . 8

2.4.3 Admin Use Case . . . 8

2.5 Sequence Diagram . . . 9

2.5.1 Registration Diagram . . . 9

2.5.2 Validation Diagram . . . 10

2.5.3 Authentication Diagram . . . 11

2.5.4 Search Athlete Diagram . . . 12
2.5.5 Manage athletes Diagram . . . 13
2.5.6 Exercise management Diagram . . . 14
2.5.7 Add Session Diagram . . . 15
2.5.8 Edit Session Diagram . . . 16
2.5.9 Delete Session Diagram . . . 17
2.5.10 Add Schedule Diagram . . . 18
2.5.11 Edit Schedule Diagram . . . 19
2.5.12 Delete Schedule Diagram . . . 20
2.5.13 Ai Load Check Diagram . . . 21
2.5.14 Add performance Diagram . . . 22
2.5.15 Edit performance Diagram . . . 23
2.5.16 Delete Performance Diagram . . . 24
2.6 Class Diagram . . . 25
2.6.1 Class Diagram of the System . . . 25
2.7 Dictionary for Class Diagram Attributes . . . 26
2.8 User Interface desaign . . . 27
2.8.1 Color Palette Specification . . . 27

3 The Implementation 28
3.1 SmartSwim Development Technologies . . . 28
3.1.1 React . . . 28
3.1.2 Node.js . . . 28
3.1.3 MongoDB . . . 29
3.1.4 Tailwind CSS . . . 29
3.2 Graphical User Interfaces (GUIs) . . . 29
3.2.1 Authentication Interface (Login) . . . 29
3.2.2 User Moderation and Account Validation Interface . . . 30
3.2.3 Coach Workspace Interfaces . . . 30
3.2.4 Athlete Workspace Interfaces . . . 35

List of Figures

1.1 Examples of traditional paper-based management forms used by coaches. 1
1.2 Commit Swimming interface overview. 2
1.3 MySwimPro interface overview. 2
1.4 Swim Coach interface overview. 3
1.5 SwimTopia interface overview. 3

2.1 General Use Case Diagram 6
2.2 Coach Use Case Diagram 7
2.3 Athlete Use Case Diagram 8
2.4 Admin Use Case Diagram 8
2.5 Registration Sequence Diagram 9
2.6 Validation Sequence Diagram 10
2.7 Authentication Sequence Diagram 11
2.8 Search Athlete Sequence Diagram 12
2.9 Manage athletes Sequence Diagram 13
2.10 Exercise management Sequence Diagram 14
2.11 Add Session Sequence Diagram 15
2.12 Edit Session Sequence Diagram 16
2.13 Delete Session Sequence Diagram 17
2.14 Add Schedule Sequence Diagram 18
2.15 Edit Schedule Sequence Diagram 19
2.16 Delete Schedule Sequence Diagram 20
2.17 Ai Load Check Sequence Diagram 21
2.18 Add performance Sequence Diagram 22
2.19 Edit performance Sequence Diagram 23
2.20 Delete Performance Sequence Diagram 24
2.21 Class Diagram of the System 25
2.22 Athletic Club Boumerdes Natation Logo. 27

3.1.1 React 28
3.1.2 Node.js 28
3.1.3 MongoDB 29
3.1.4 Tailwind CSS 29

3.3 Authentication Interface 29
3.4 User Moderation and Account Validation Interface 30
3.5 Coach Workspace Interfaces 30
3.6 Athlete Management Interface 31
3.7 Training Squads Configuration Interface 31
3.8 Training Workouts Management Interface 32
3.9 Training Sessions Management Module 32
3.10 Annual Planning and Periodization Interface 33
3.11 Annual Planning and Periodization Interface 33
3.12 Individual Performance Analytics Interface 34
3.13 Group Performance Analytics Interface 34
3.14 Athlete Workspace Interfaces 35

List of Tables

2.1 Registration Description 9
2.2 Account Validation Description 10
2.3 Authentication Description 11
2.4 Search Athlete Description 12
2.5 Manage Athlete Description 13
2.6 Manage Exercise Description 14
2.7 Add Session Description 15
2.8 Edit Session Description 16
2.9 Delete Session Description 17
2.10 Add Schedule Description 18
2.11 Edit Schedule Description 19
2.12 Delete Schedule Description 20
2.13 AI Load Check Description 21
2.14 Add Performance Description 22
2.15 Edit performance Description 23
2.16 Delete Performance Description 24
2.17 Data Dictionary 26
2.18 SmartSwim Color Palette Specification. 27

Acknowledgments

First and foremost, we would like to express our deepest gratitude to Allah Almighty for granting us the strength, health, patience, and determination to undertake and successfully complete this graduation project. His divine guidance, countless blessings, and favor have been our ultimate source of motivation, hope, and perseverance, enabling us to overcome every challenge and obstacle we encountered along this academic journey.

We would like to express our profound and most sincere gratitude to our esteemed supervisor, Mr. Chaouche, for his exceptional guidance, unwavering support, immense patience, and invaluable advice throughout the realization of this project. We are deeply indebted to him for sharing his vast expertise, constructive feedback, and insights, which played a fundamental role in shaping the direction and quality of this work. His constant availability, continuous encouragement, and dedication to pushing us toward excellence have been truly inspiring. We are immensely grateful for the precious time he generously dedicated to supervising us, his high level of professionalism, and his tireless efforts to ensure that we improved step by step to achieve the best possible outcomes. We also extend our sincere thanks and appreciation to all the respected members of the jury who have agreed to evaluate this work. Their valuable time, remarks, and constructive perspectives are highly appreciated and will undoubtedly enrich the quality of our academic endeavor. Finally, our heartfelt thanks go to everyone who contributed, directly or indirectly, with a word of encouragement, a helping hand, or shared knowledge during the completion of this thesis.

Dedication

Dedicated by Wafaa

I would like to thank Allah Almighty for granting me the health and perseverance to reach this day. I dedicate this humble work to my precious parents, who have given me everything and whose encouragement has always been my source of strength. To my beloved brothers , thank you for your kindness, shared moments, and for being my ultimate cheerleaders throughout this academic journey.

Dedicated by Hafsa

Foremost, I thank Allah for His divine guidance and blessings that enabled me to complete this journey. I dedicate this achievement with deep love and gratitude to my exceptional parents, whose sacrifices and belief in me made everything possible. To my wonderful brothers and sisters, thank you for your constant motivation, joyful presence, and unwavering support that kept me going through every difficult step.

Dedicated by Cerine

First of all, I praise Allah Almighty for giving me the patience, health, and willpower to achieve this great academic milestone. I proudly and affectionately dedicate this entire work to my beloved parents, whose endless love, continuous sacrifices, and prayers have been my greatest shield and support throughout my life. Thank you for always believing in me and lighting up my path toward success.

Dedicated by Aymen

All thanks to Allah for giving me the determination and ability to successfully finish this work. I dedicate this project to my dear parents, whose endless support, patience, and sacrifices have guided my path toward success. To my brothers and sisters, I express my sincere gratitude for your continuous encouragement, motivation, and for always standing by me through every single milestone.

General Introduction

Swimming is one of the most demanding sports, requiring discipline, endurance, and continuous performance evaluation. The role of the coach is essential in planning training sessions, monitoring athletes' progress, and preparing swimmers for competitions. However, many coaches still rely on traditional methods such as manual planning and paper-based performance tracking, which makes training management difficult and time-consuming.
In Algeria, these challenges are more significant due to the lack of specialized digital solutions adapted to the needs of local swimming clubs and coaches. Most existing platforms are either expensive, complex, or not designed for the Algerian sports environment. As a result, coaches often face difficulties in organizing training schedules, tracking athlete performance, and managing competitions efficiently.
To address these problems, the idea of developing our platform emerged. The platform aims to provide an accessible and organized solution adapted to the Algerian context, helping coaches and swimmers simplify training management, improve performance monitoring, and modernize swimming activities through digital technologies.

This dissertation is organized into two main parts:

- **Part I: State of the Art**
  - **Chapter 1** studies the current methods, analyzes existing solutions related to swimming management systems, highlights their limitations, and identifies the core needs that motivated our proposed solution.

- **Part II: Conception and Implementation**
  - **Chapter 2 (Conception)** describes the system analysis and modeling process, including the different design approaches and diagrams adopted for our platform.
  - **Chapter 3 (Implementation)** presents the development environment, the technologies and tools used, as well as the practical realization, features, and deployment of the platform's system.

The report concludes with a summary of achievements, main contributions, and future perspectives for improving the platform's scalability and adoption.

Part I

State Of the Art

Chapter 1

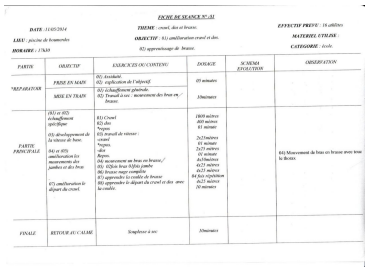
Study Of The Existence

Introduction

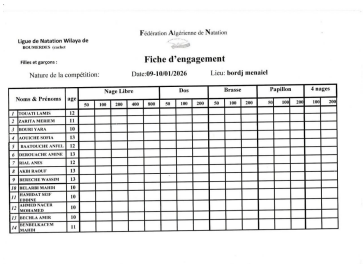
This chapter present existing methodologies in swimming training management. Many Algerian clubs still rely on manual tracking, reducing the efficiency of data management and performance tracking. Furthermore, available sports platforms lacks the adaptation to the Algerian sports framework

1.1 Methods used by Algerien coaches

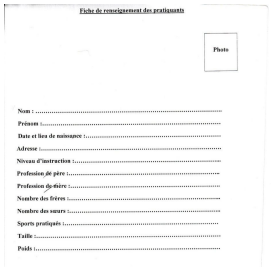
The absence of modern management tools in many Algerian swimming clubs leaves coaches with outdated practices that limit their ability to efficiently oversee swimmers and training programs.. Paper documents and notebooks are commonly used to record swimmers' information, attendance, training schedules, and competition results. Although these methods are simple, they create several practical difficulties in daily management. The use of paper records increases the risk of losing data and makes information retrieval more complicated. Coaches often need to examine multiple documents to monitor swimmers' progress and compare performance results over time. In addition, planning training sessions and preparing weekly exercise programs require considerable time and effort, while manual recording may lead to organizational errors and inaccurate data such as:



a) Training Session Sheet



b) Registration Sheet



c)Athlet Info Sheet

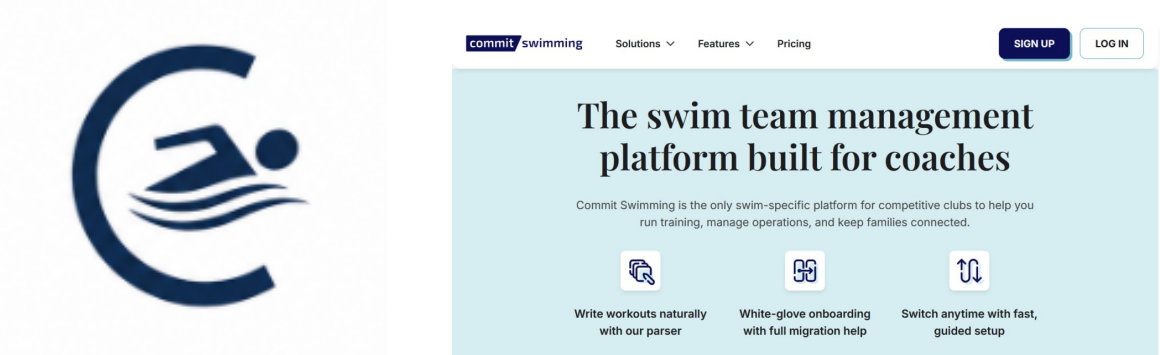
Figure 1.1: Examples of traditional paper-based management forms used by coaches.

1.2 Application for swimming management

Several swimming management applications have been developed to help coaches organize training sessions and monitor swimmers' performance

1.2.1 Commit Swimming

Commit Swimming is a swim training platform designed to help clubs manage workouts, attendance, performance tracking, and team communication to improve organization and coaching efficiency. However, it may not fully fit local coaching needs or some traditional training environments and can be less suitable for small clubs with limited resources.[1].

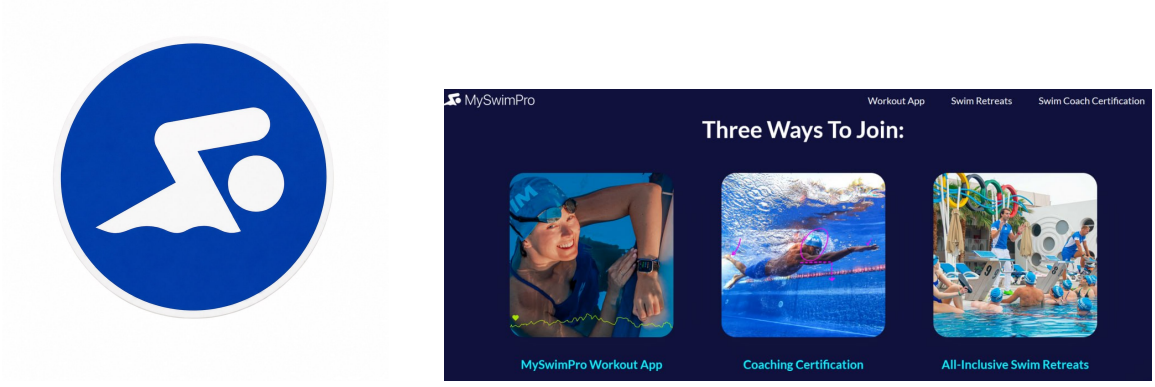


a) Commit Swimming Logo b) Commit Swimming Login Interface

Figure 1.2: Commit Swimming interface overview.

1.2.2 MySwimPro

MySwimPro is a swim training and coaching app that provides structured workouts, personalized training plans, technique videos, and tracking tools to help swimmers improve performance and stay consistent. However, it may not fully support local coaching needs or some training environments. [2].

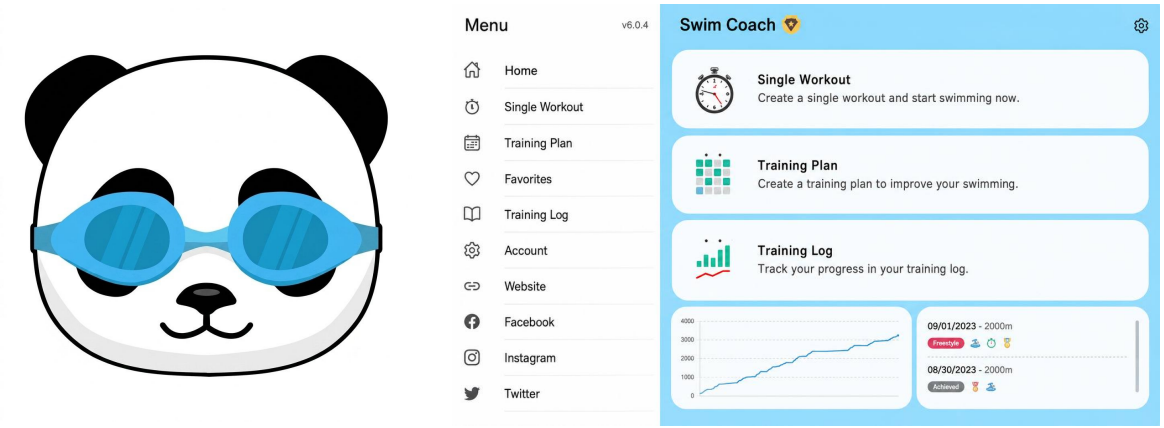


a) MySwimPro Logo b) MySwimPro Login Interface

Figure 1.3: MySwimPro interface overview.

1.2.3 Swim Coach

Swim Coach (Panda) is a swim training app that offers structured workouts, training plans, and guided sessions to help users improve swimming skills and maintain consistent practice. However, it may not fully support local coaching needs or some training environments. [3].



a) Swim Coach Logo b) Swim Coach Login Interface

Figure 1.4: Swim Coach interface overview.

1.2.4 SwimTopia

SwimTopia is swim team management software that helps organize schedules, meet information, results tracking, and communication for coaches, swimmers, and parents, making team operations easier and more efficient. However, it may not fully support local coaching needs or some training environments. [4]



a) SwimTopia Logo b) SwimTopia Login Interface

Figure 1.5: SwimTopia interface overview.

1.3 Problem Statement and Proposed Solution

These applications do not fully meet our needs because they are not adapted to our local working environment. In addition, most of them require paid subscriptions using foreign currencies, which are not always accessible or convenient for local users. Some platforms also do not support local payment methods, making their use more complicated. Moreover, many of these applications are difficult to customize, include unnecessary features, and depend heavily on a stable internet connection. Consequently, coaches and sports managers may face several challenges when trying to use them effectively in their daily activities. For these reasons, we decided to develop our own platform that is better adapted to our local context and real needs.

Conclusion

This chapter has described the general problem of trainers' dependency on paper records, pointed out the limitations of existing approaches, and highlighted the necessity for a creative solution that considers local conditions. As a result, in the next chapter, we present the proposed system architecture, visual identity, and main user interfaces.

Part II

Conception and Implementation

Chapter 2

Conception

Introduction

In this chapter, we present the conception of the SmartSwim project using UML diagrams and the UI design of the application. The next chapter will focus on the implementation process, including the technologies, tools, and development of the application.

2.1 UML Modeling

For the system design, we utilized UML modeling to structure the web application. This architecture is presented using Use Case, Sequence, and Class diagrams.

2.2 Identification of Actors

- **Admin** manages users and groups, validates coaches' content, and ensures the proper functioning of the application.
- **Coach** manages training sessions, monitors athletes' performance, and can send messages to their athletes.
- **Athlete** can view their performance and send messages to the coach.

2.3 Use Case Diagram

We used Use Case Diagram to describe the main interactions between the users and the system.

2.3.1 General Use Case Diagram

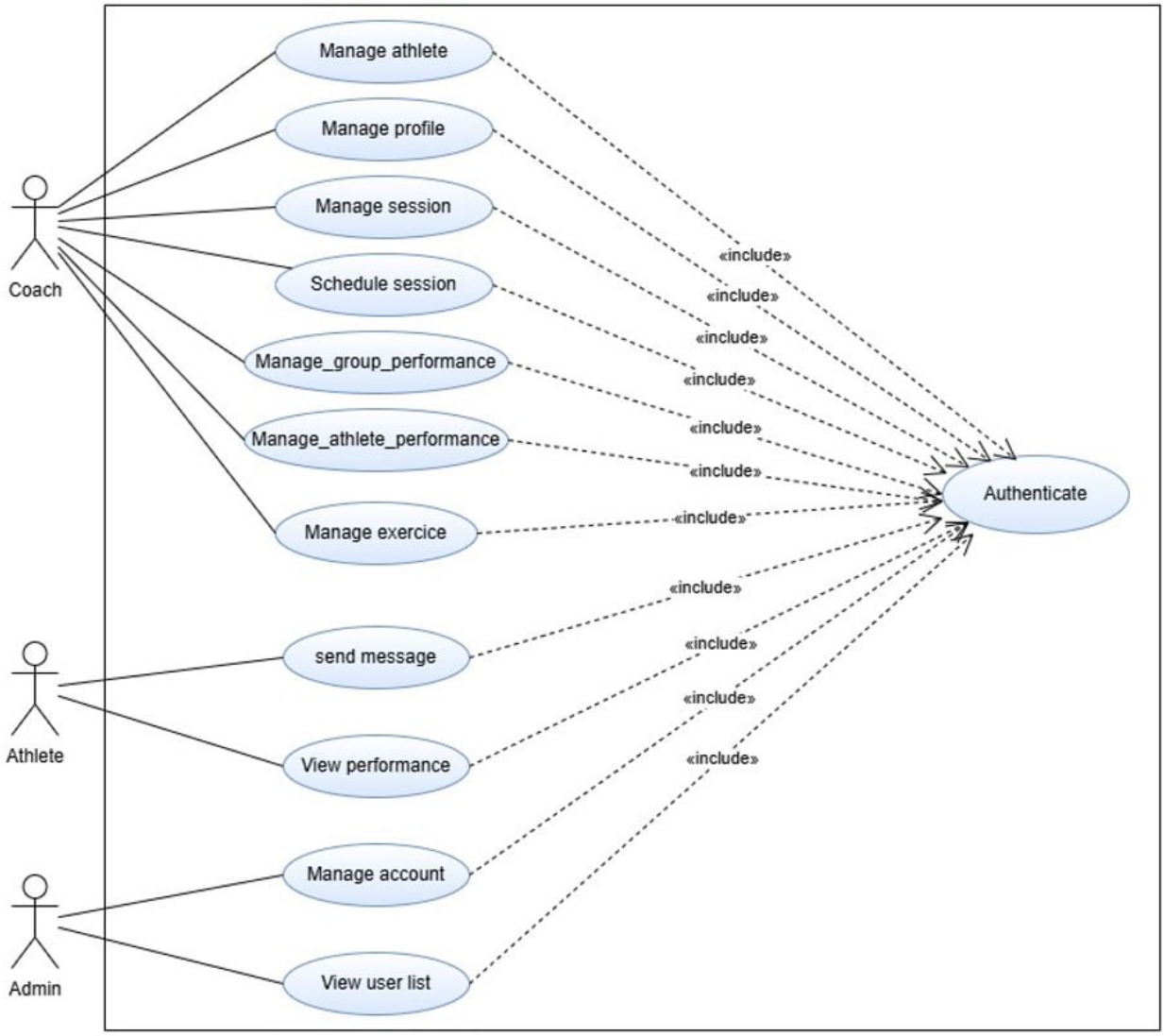


Figure 2.1: General Use Case Diagram

2.4 Detailed Use Case Diagrams

2.4.1 Coach Use Case

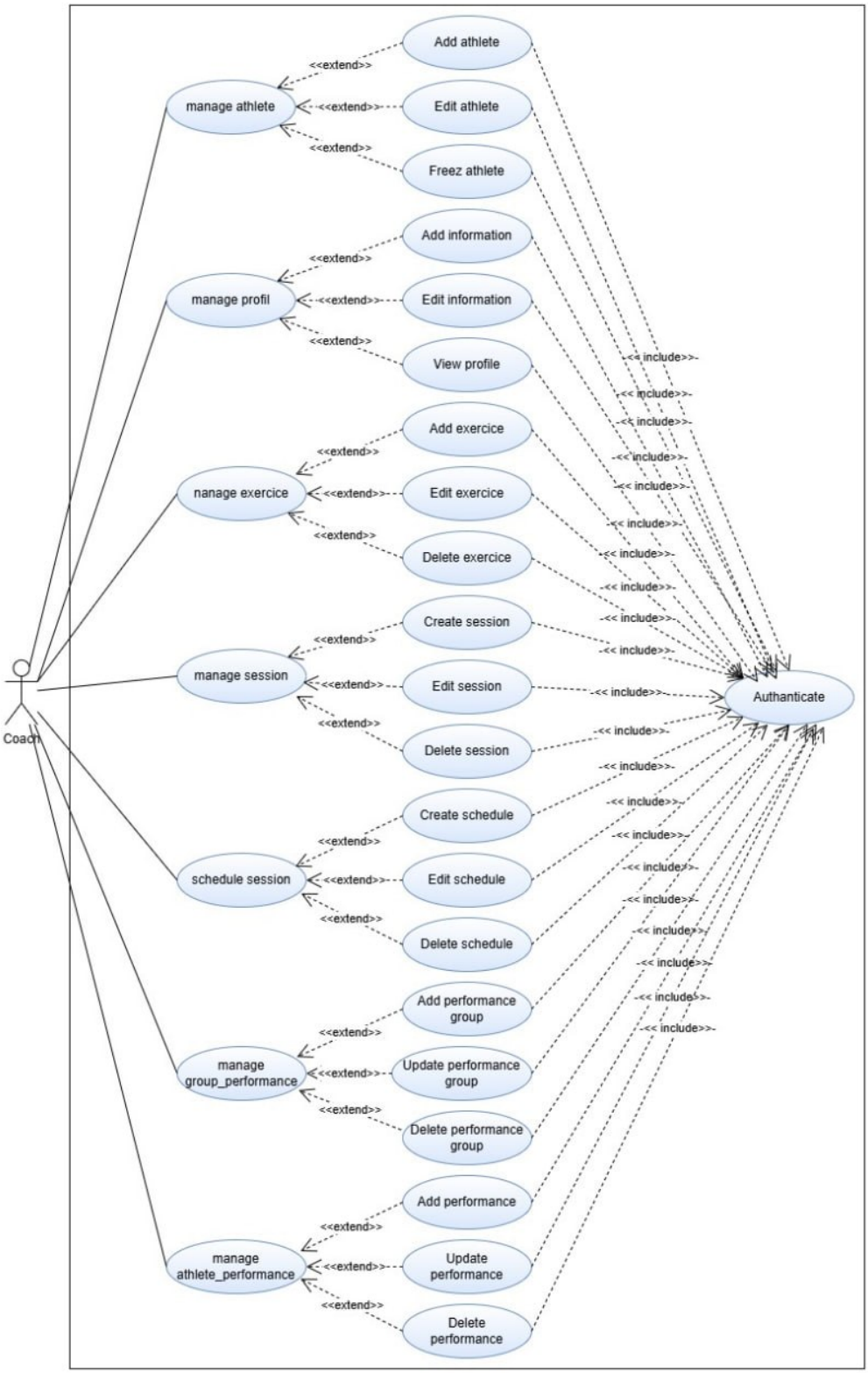


Figure 2.2: Coach Use Case Diagram

2.4.2 Athlete Use Case

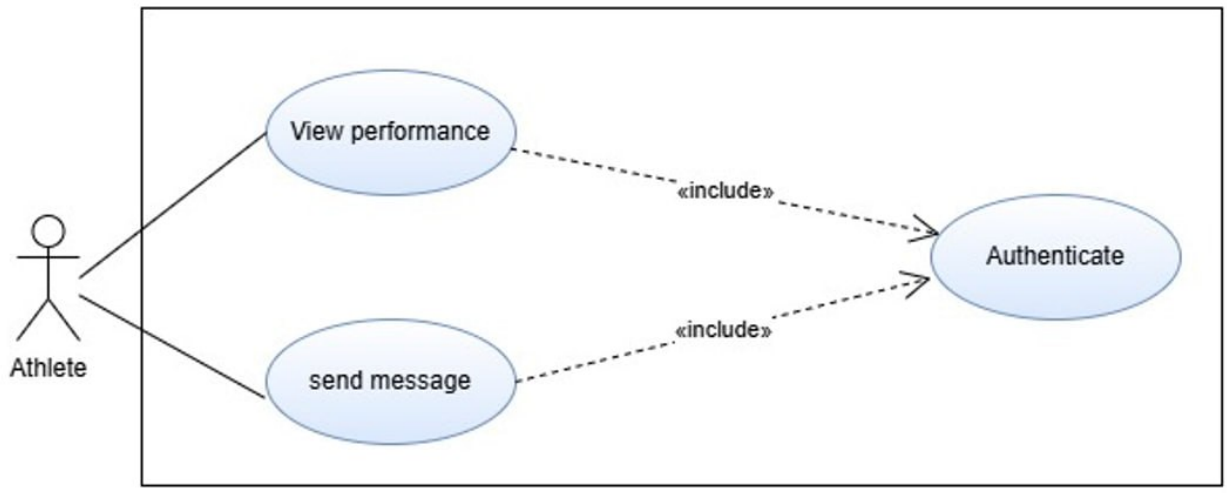


Figure 2.3: Athlete Use Case Diagram

2.4.3 Admin Use Case

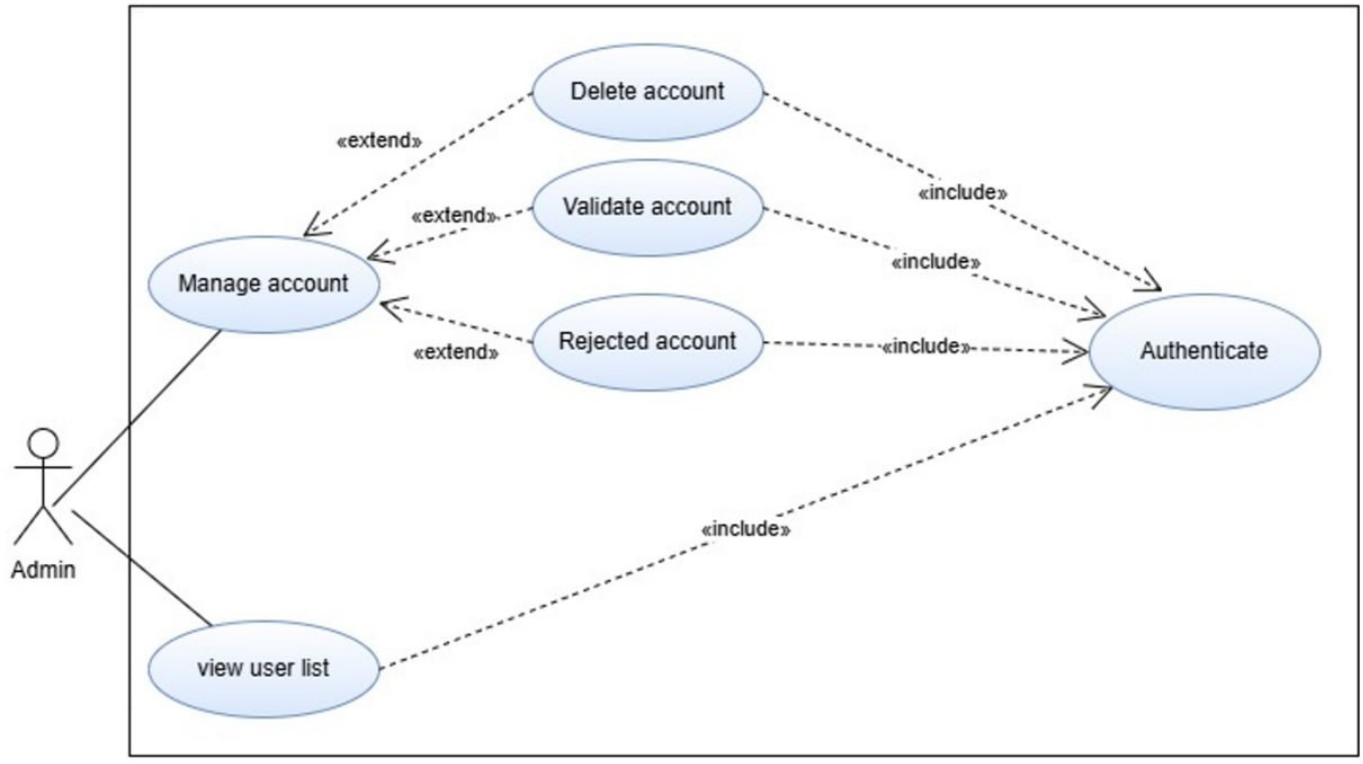


Figure 2.4: Admin Use Case Diagram

2.5 Sequence Diagram

We used sequence diagrams to represent the dynamic behavior of the system and the order of exchanged messages between different actors.

2.5.1 Registration Diagram

This sequence diagram illustrates the user registration process in the system.

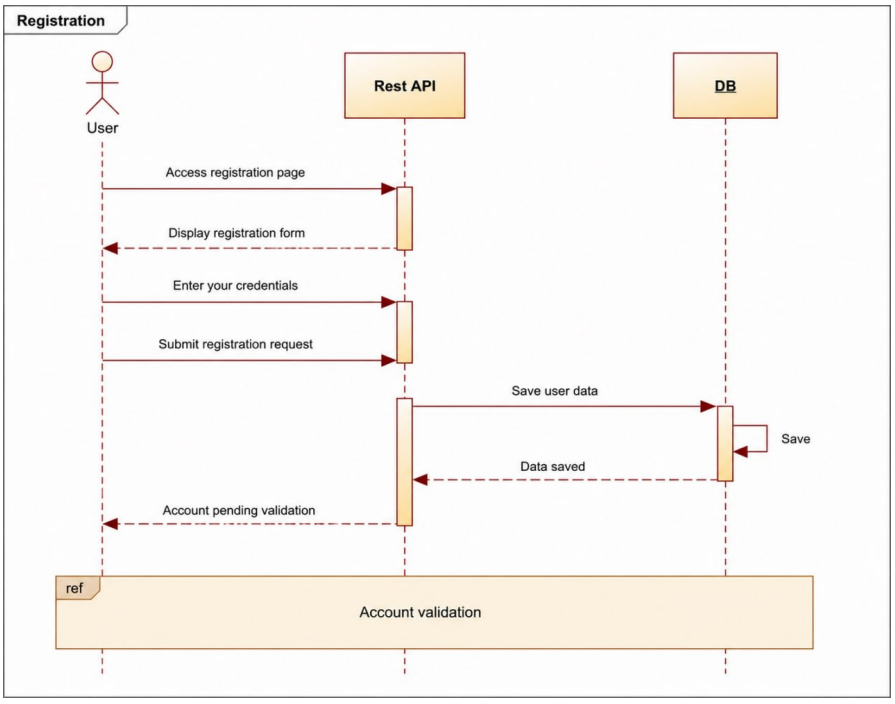


Figure 2.5: Registration Sequence Diagram

Title	Registration
Actor	User
Objective	Create a new account in the system.
Pre-condition	The user accesses the registration page.
Post-condition	The account is created and pending validation.
Main scenario	1- Access the registration page. 2- Fill in the registration form. 3- Submit the registration request. 4- The system saves the user data. 5- The account is pending validation.
Alternative scenario	1- If the entered information is invalid, the system displays an error message.

Table 2.1: Registration Description

2.5.2 Validation Diagram

This sequence diagram describes the account validation process performed by the administrator.

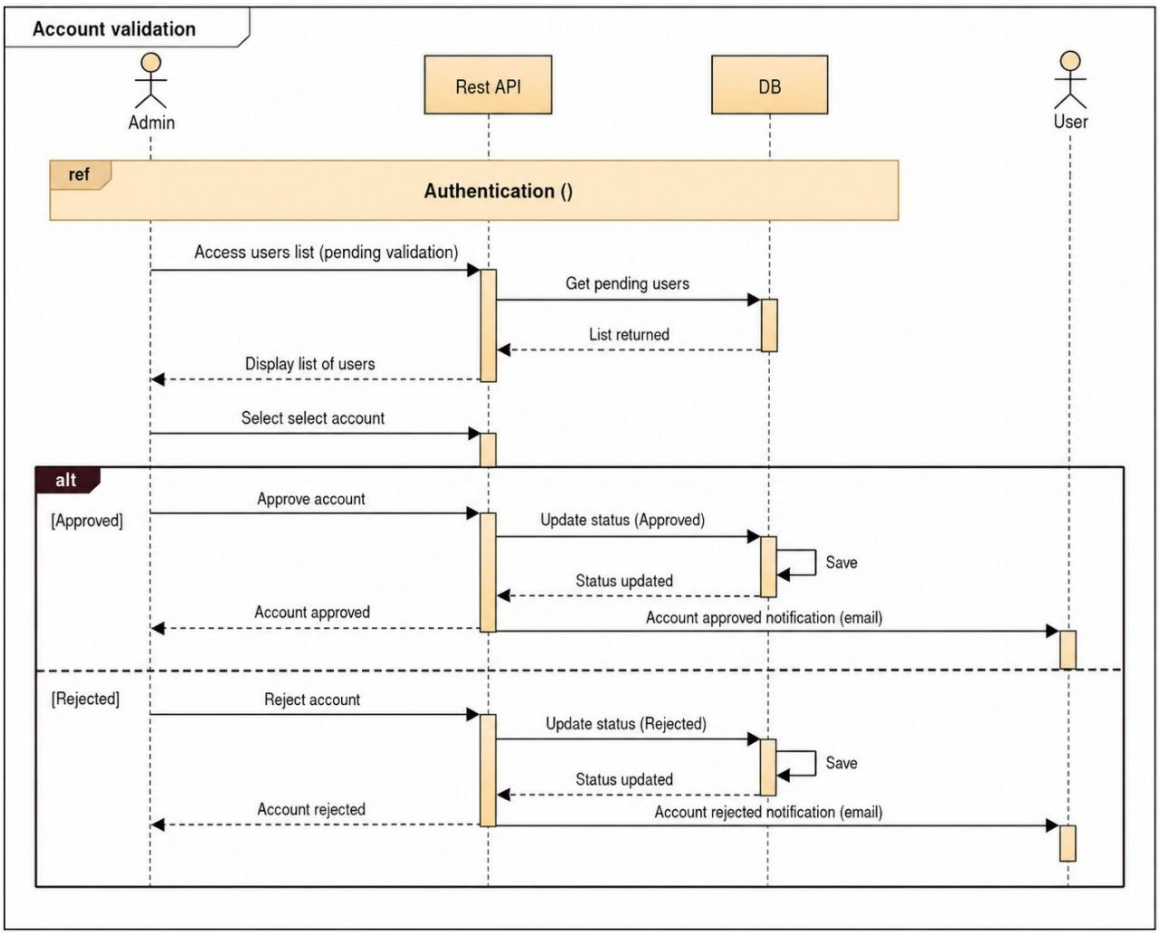


Figure 2.6: Validation Sequence Diagram

Title	Account validation
Actor	Admin
Objective	Validate user accounts.
Pre-condition	Admin authenticated.
Post-condition	Account approved or rejected.
Main scenario	1- Access pending users. 2- Select an account. 3- Approve the account. 4- The system updates the status and sends a notification.
Alternative scenario	1- Reject the account. 2- The system updates the status and sends a rejection notification.

Table 2.2: Account Validation Description

2.5.3 Authentication Diagram

This sequence diagram describes the authentication process in the system.

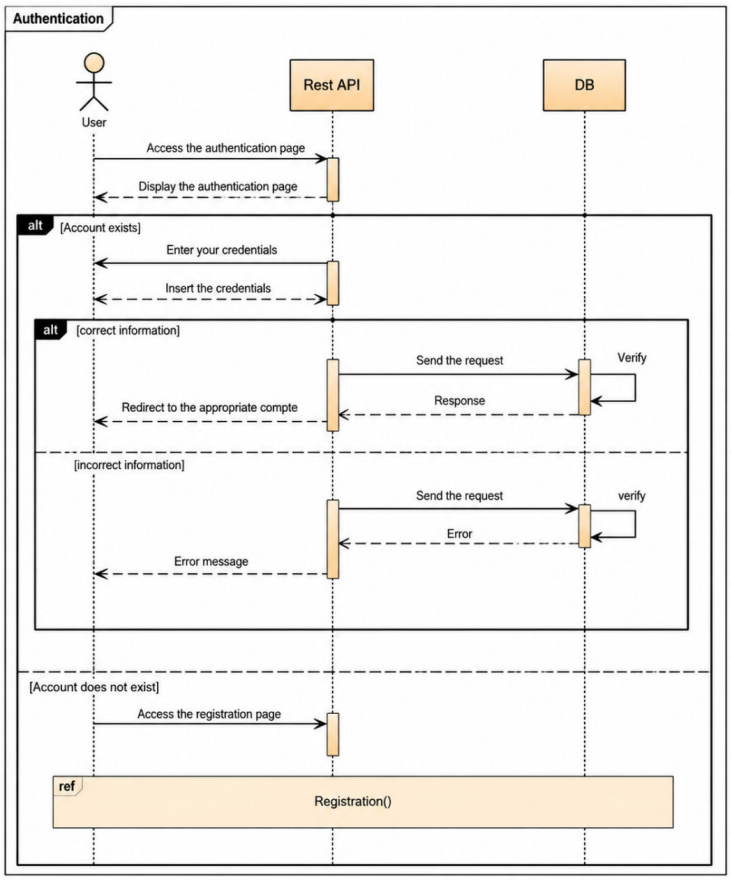


Figure 2.7: Authentication Sequence Diagram

Title	Authentication
Actor	User
Objective	Authenticate into the system.
Pre-condition	The user has an account or accesses the registration page.
Post-condition	Access to the account or display of an error message.
Main scenario	1- Access the authentication page. 2- Enter the credentials. 3- The system verifies the information. 4- Redirect to the appropriate account.
Alternative scenario	1- If the information is incorrect, the system displays an error message. 2- If the account does not exist, the user is redirected to the registration page.

Table 2.3: Authentication Description

2.5.4 Search Athlete Diagram

This sequence diagram describes the athlete search process performed by a coach.

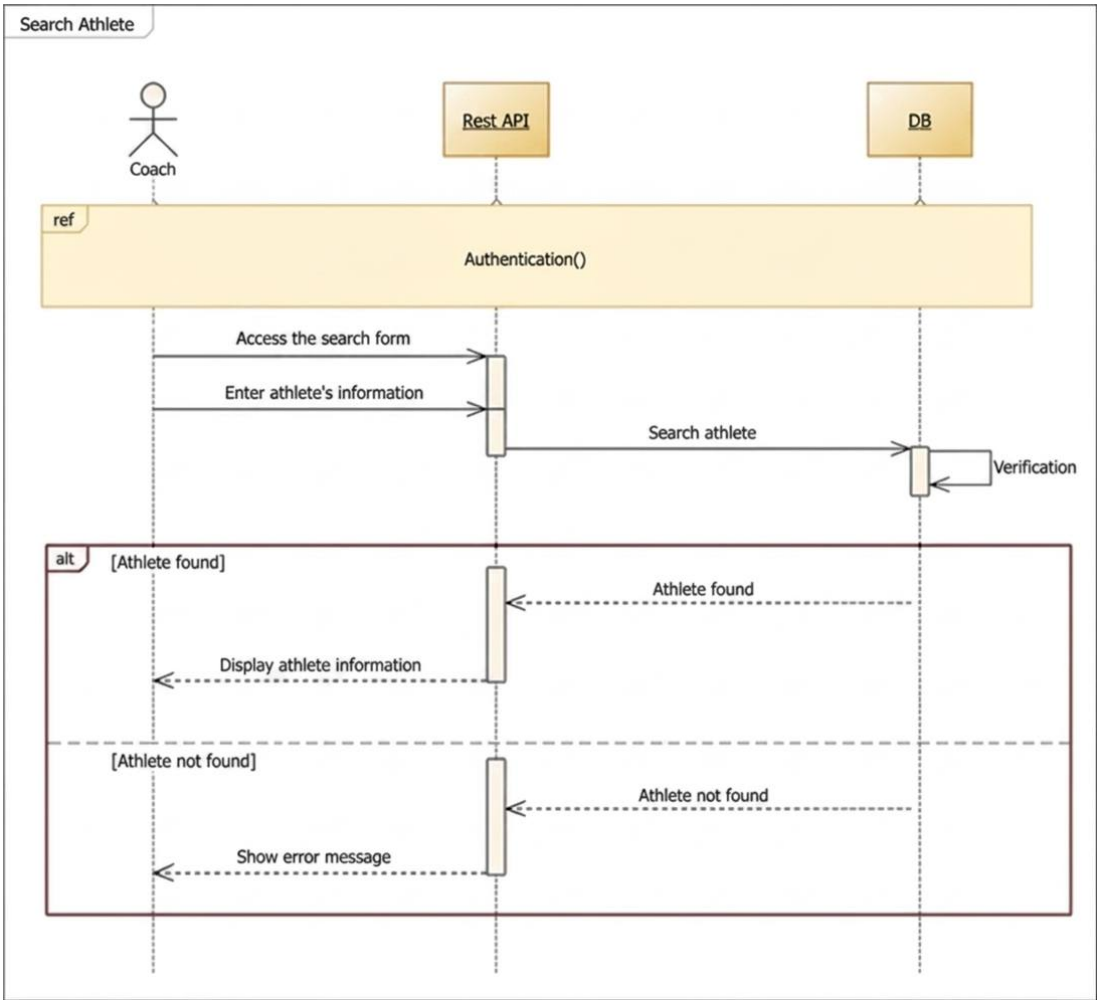


Figure 2.8: Search Athlete Sequence Diagram

Title	Search Athlete
Actor	Coach
Objective	Search for an athlete.
Pre-condition	Coach authenticated.
Post-condition	Athlete information displayed or error message shown.
Main scenario	1- Access the search form. 2- Enter athlete information. 3- The system searches for the athlete. 4- Display athlete information.
Alternative scenario	1- If the athlete is not found, the system displays an error message.

Table 2.4: Search Athlete Description

2.5.5 Manage athletes Diagram

This sequence diagram describes the athlete management process performed by a coach.

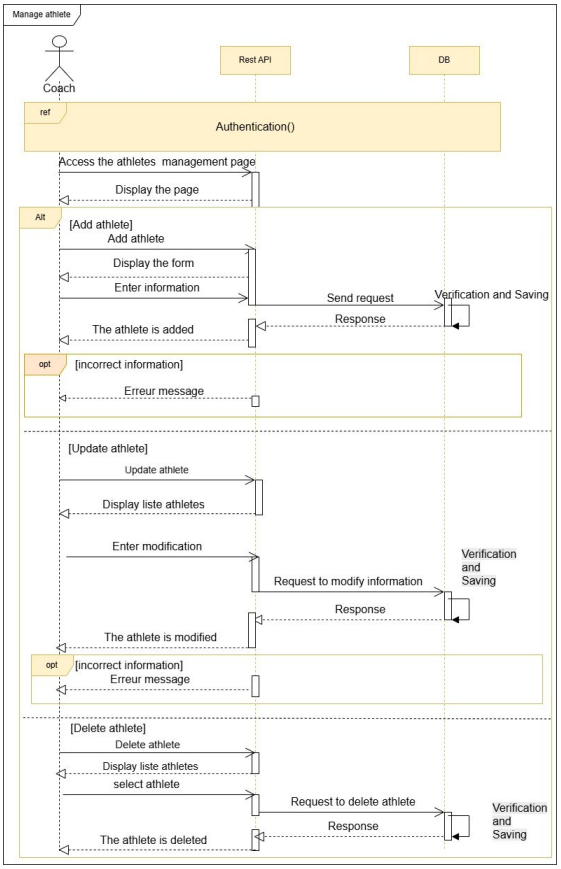


Figure 2.9: Manage athletes Sequence Diagram

Title	Manage Athlete
Actor	Coach
Objective	Manage athletes in the system.
Pre-condition	The coach is authenticated and accesses the athlete management page.
Post-condition	The athlete is added, modified, or deleted successfully.
Main scenario	1- Access the athlete management page. 2- Select an operation (Add, Update, or Delete athlete). 3- The system verifies the request. 4- The system saves the changes.
Alternative scenario	1- If the entered information is incorrect, the system displays an error message.

Table 2.5: Manage Athlete Description

2.5.6 Exercise management Diagram

This sequence diagram describes the exercise management process performed by a coach.

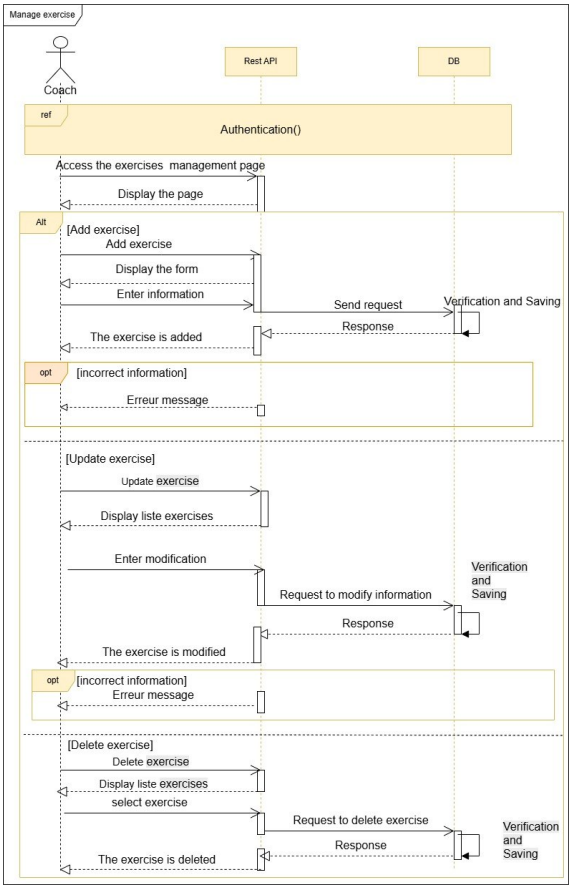


Figure 2.10: Exercise management Sequence Diagram

Title	Manage Exercise
Actor	Coach
Objective	Manage exercises in the system.
Pre-condition	The coach is authenticated and accesses the exercise management page.
Post-condition	The exercise is added, modified, or deleted successfully.
Main scenario	1- Access the exercise management page. 2- Select an operation (Add, Update, or Delete exercise). 3- The system verifies the request. 4- The system saves the changes.
Alternative scenario	1- If the entered information is incorrect, the system displays an error message.

Table 2.6: Manage Exercise Description

2.5.7 Add Session Diagram

This sequence diagram shows how a Coach adds a session after authentication. The Coach enters session information such as duration or distance and can also select an already created exercise, then the information is verified before the session is created.

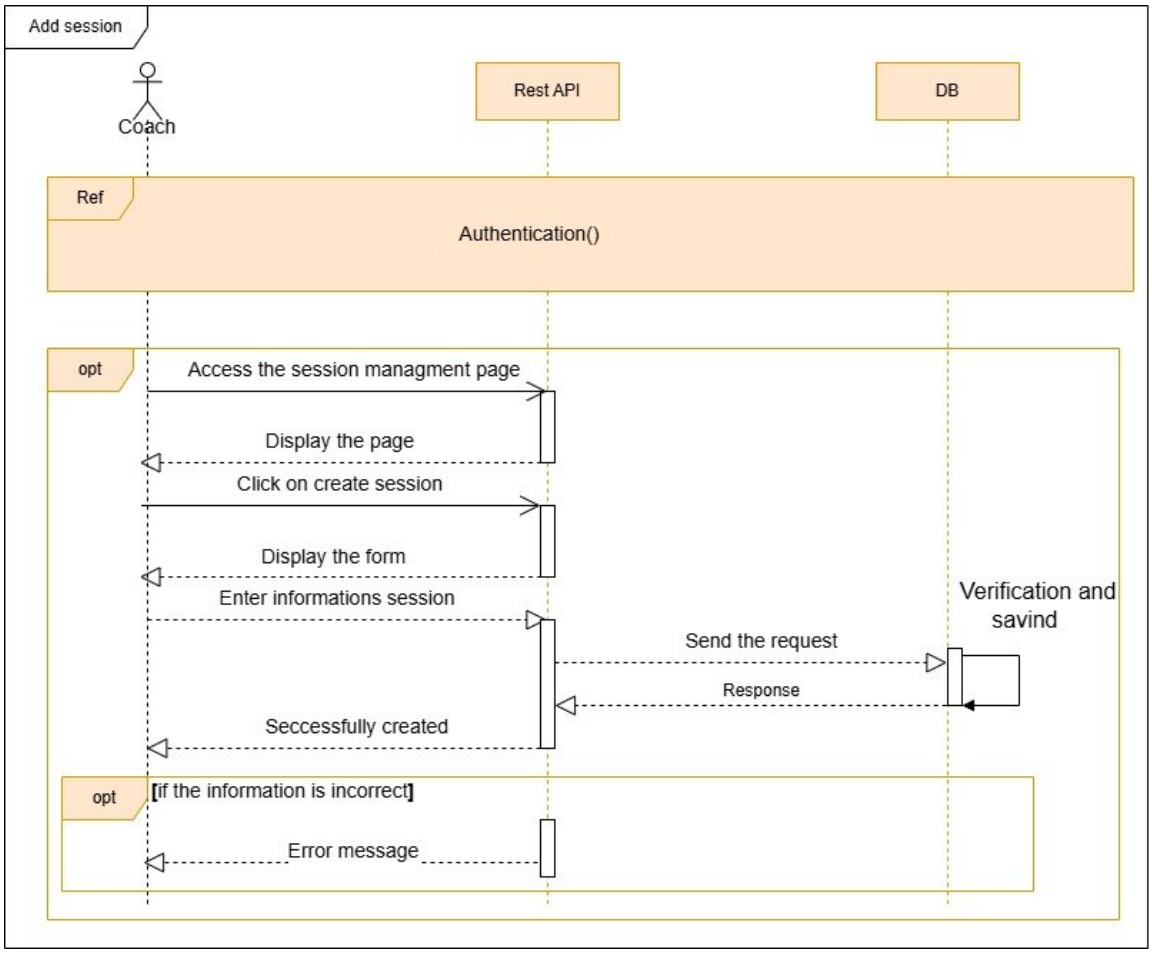


Figure 2.11: Add Session Sequence Diagram

Title	Add Session
Actor	Coach
Objective	Add a new session.
Pre-condition	Coach authenticated.
Post-condition	Session created or error message displayed.
Main scenario	1- Access the session management page. 2- Click on create session. 3- Enter session information. 4- The system verifies and saves the session. 5- Session successfully created.
Alternative scenario	1- If the information is incorrect, the system displays an error message.

Table 2.7: Add Session Description

2.5.8 Edit Session Diagram

This sequence diagram illustrates how a Coach edits a session after authentication. The Coach can modify session information such as duration or distance, and can also add or remove exercises from the session. Then, the information is verified and the modifications are saved, or an error message is displayed if the information is incorrect.

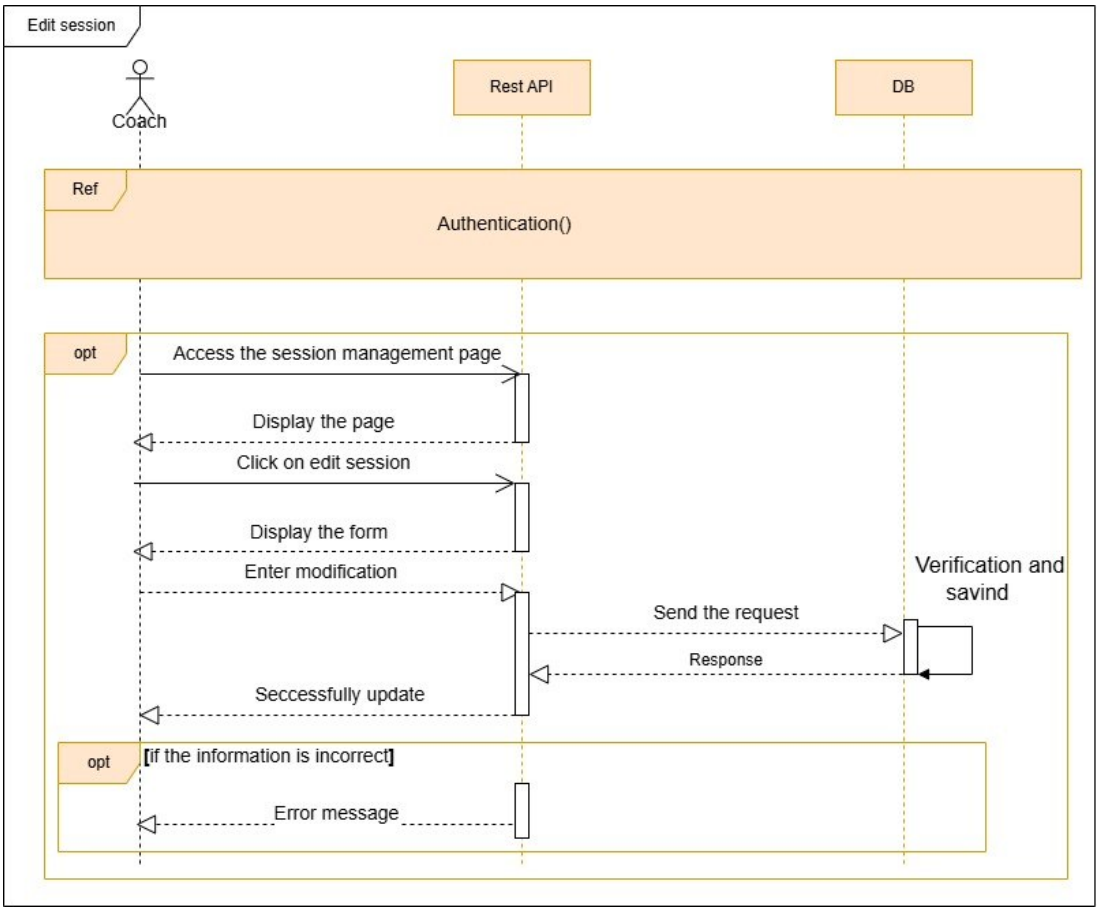


Figure 2.12: Edit Session Sequence Diagram

Title	Edit Session
Actor	Coach
Objective	Edit an existing session.
Pre-condition	Coach authenticated.
Post-condition	Session updated or error message displayed.
Main scenario	1- Access the session management page. 2- Click on edit session. 3- Modify session information. 4- The system verifies and saves the modifications. 5- Session successfully updated.
Alternative scenario	1- If the information is incorrect, the system displays an error message.

Table 2.8: Edit Session Description

2.5.9 Delete Session Diagram

This sequence diagram describes the session deletion process performed by a coach.

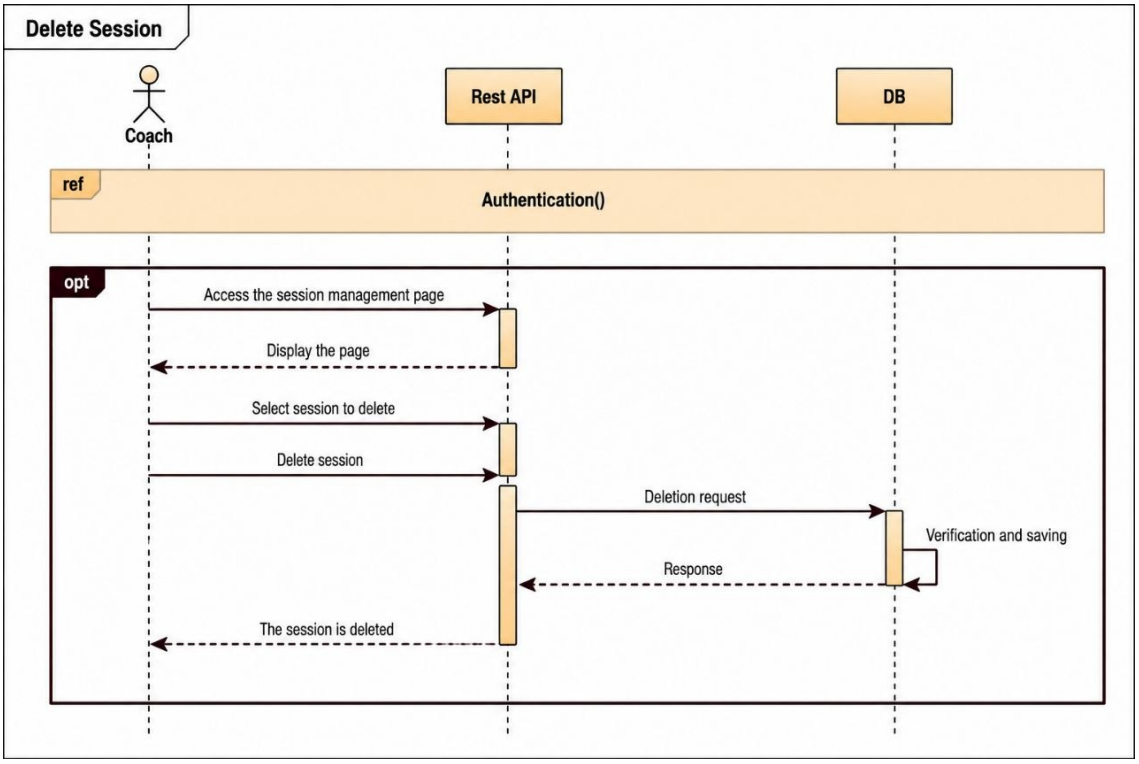


Figure 2.13: Delete Session Sequence Diagram

Title	Delete Session
Actor	Coach
Objective	Delete a session.
Pre-condition	Coach authenticated.
Post-condition	Session deleted successfully.
Main scenario	1- Access the session management page. 2- Select the session to delete. 3- Send the deletion request. 4- The system verifies and deletes the session. 5- Confirmation message displayed.
Alternative scenario	1- If the session cannot be deleted, the system displays an error message.

Table 2.9: Delete Session Description

2.5.10 Add Schedule Diagram

This sequence diagram illustrates how a Coach adds a schedule after authentication. The Coach selects sessions that have already been created and adds them to the schedule, then the information is verified before the schedule is successfully created or an error message is displayed if the information is incorrect.

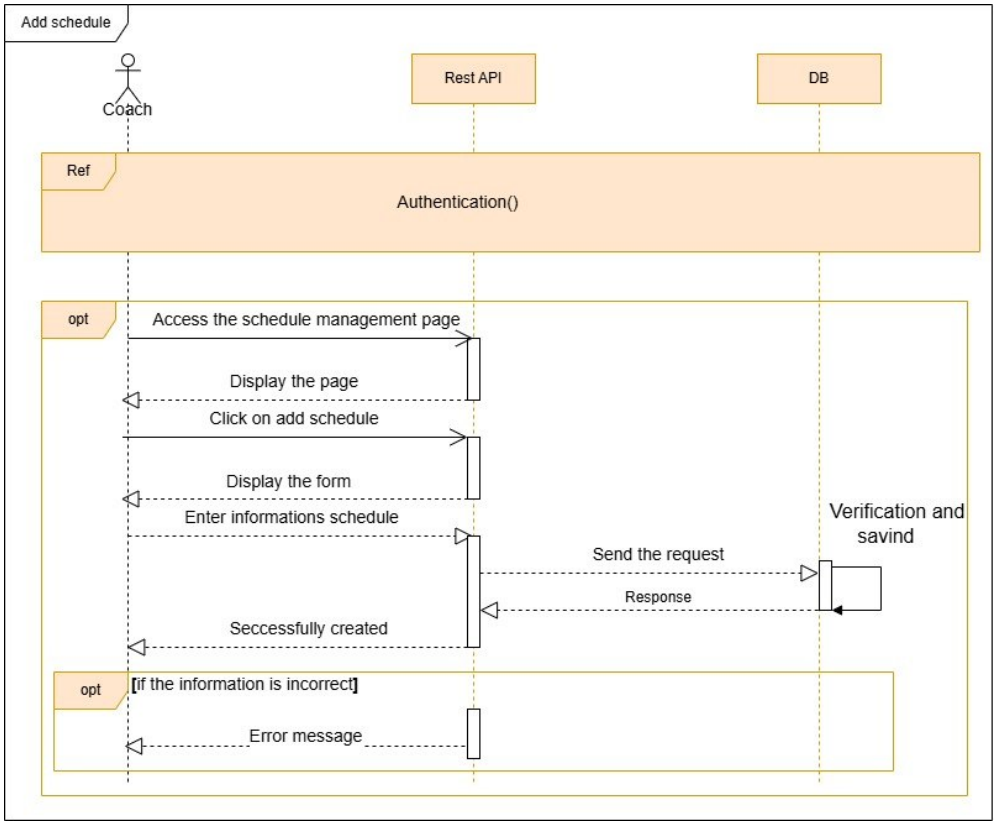


Figure 2.14: Add Schedule Sequence Diagram

Title	Add Schedule
Actor	Coach
Objective	Add a new schedule.
Pre-condition	Coach authenticated.
Post-condition	Schedule created or error message displayed.
Main scenario	1- Access the schedule management page. 2- Click on add schedule. 3- Enter schedule information. 4- The system verifies and saves the schedule. 5- Schedule successfully created.
Alternative scenario	1- If the information is incorrect, the system displays an error message.

Table 2.10: Add Schedule Description

2.5.11 Edit Schedule Diagram

This sequence diagram illustrates how a Coach edits a Schedule after authentication. The Coach can add a new Session to the Schedule or remove an existing one, then the modifications are verified and saved, or an error message is displayed if the information is incorrect.

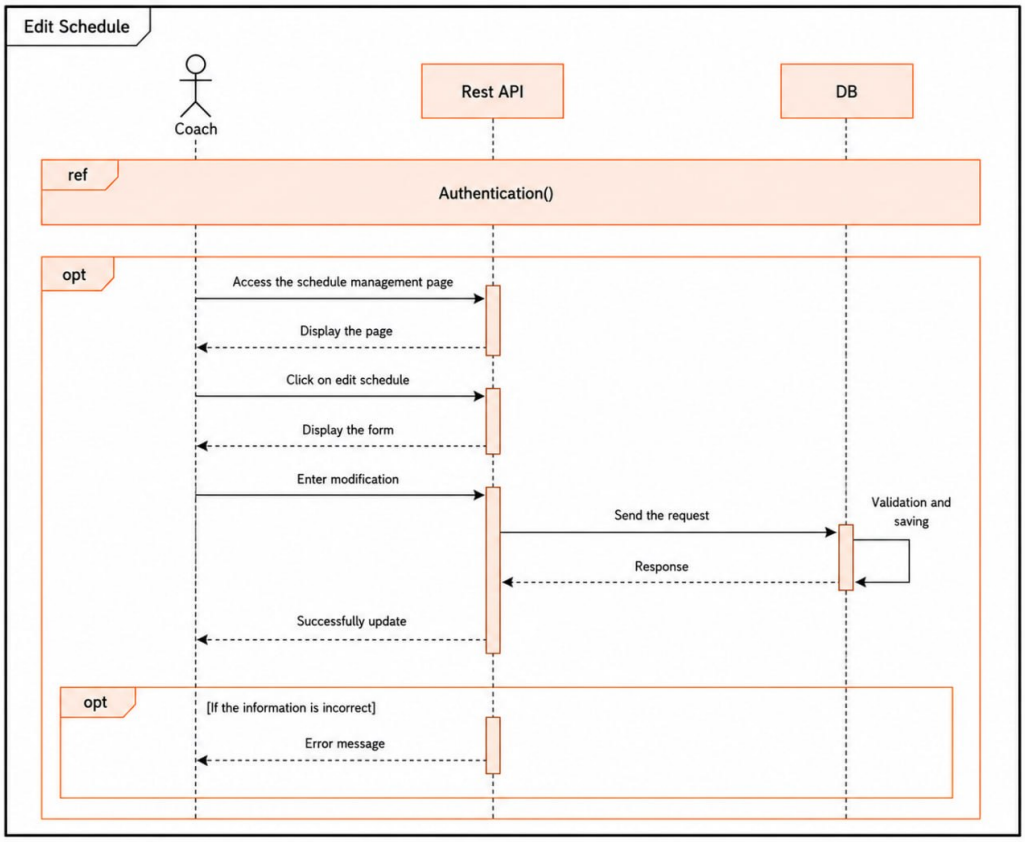


Figure 2.15: Edit Schedule Sequence Diagram

Title	Edit Schedule
Actor	Coach
Objective	Edit an existing schedule.
Pre-condition	Coach authenticated.
Post-condition	Schedule updated or error message displayed.
Main scenario	1- Access the schedule management page. 2- Click on edit schedule. 3- Modify the schedule. 4- The system verifies and saves the modifications. 5- Schedule successfully updated.
Alternative scenario	1- If the information is incorrect, the system displays an error message.

Table 2.11: Edit Schedule Description

2.5.12 Delete Schedule Diagram

This sequence diagram describes the schedule deletion process performed by a coach.

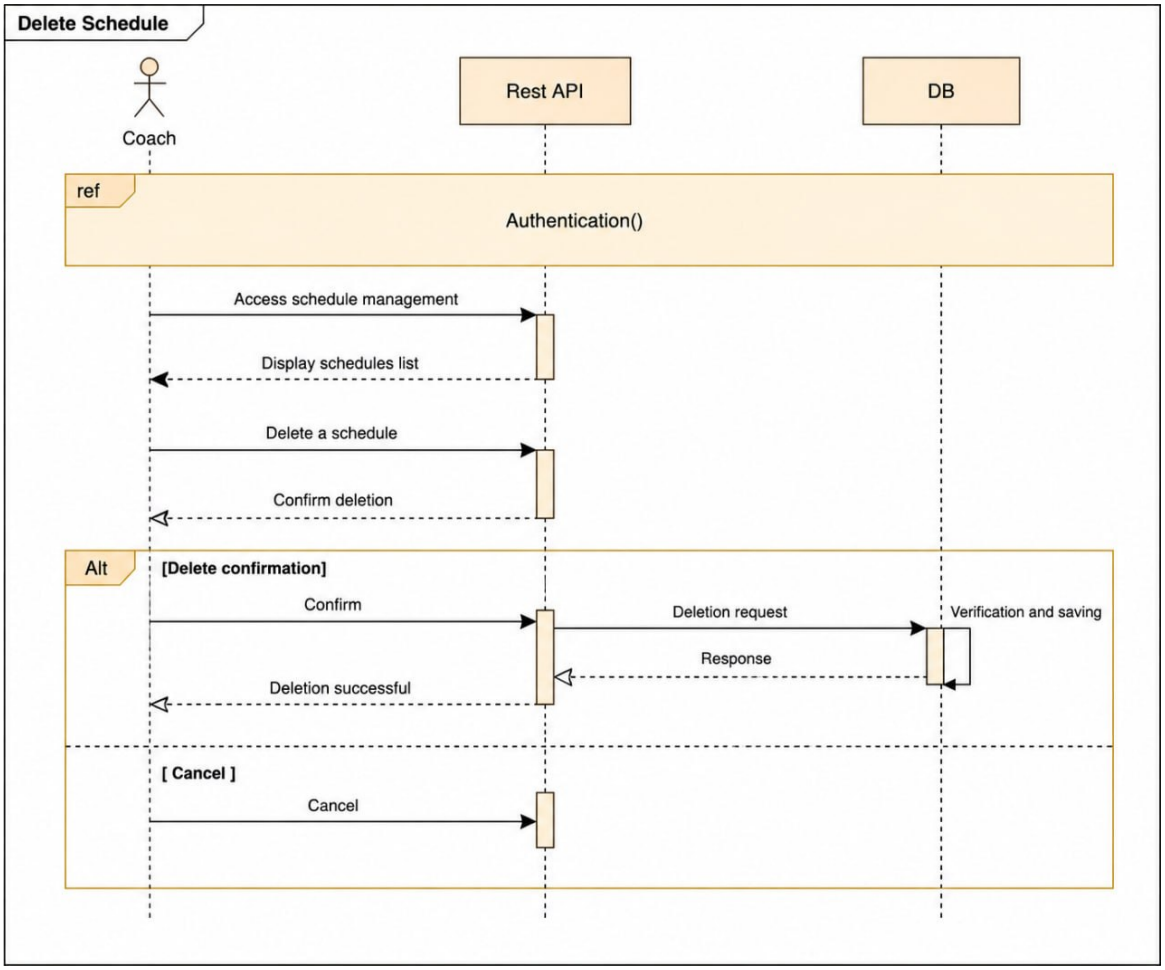


Figure 2.16: Delete Schedule Sequence Diagram

Title	Delete Schedule
Actor	Coach
Objective	Delete a schedule.
Pre-condition	Coach authenticated.
Post-condition	Schedule deleted successfully or cancellation confirmed.
Main scenario	1- Access the schedule management page. 2- Select a schedule to delete. 3- Confirm deletion. 4- The system verifies and deletes the schedule. 5- Deletion successful message displayed.
Alternative scenario	1- The coach cancels the deletion process. 2- The deletion operation is aborted.

Table 2.12: Delete Schedule Description

2.5.13 Ai Load Check Diagram

This sequence diagram describes the AI-based training load verification process.

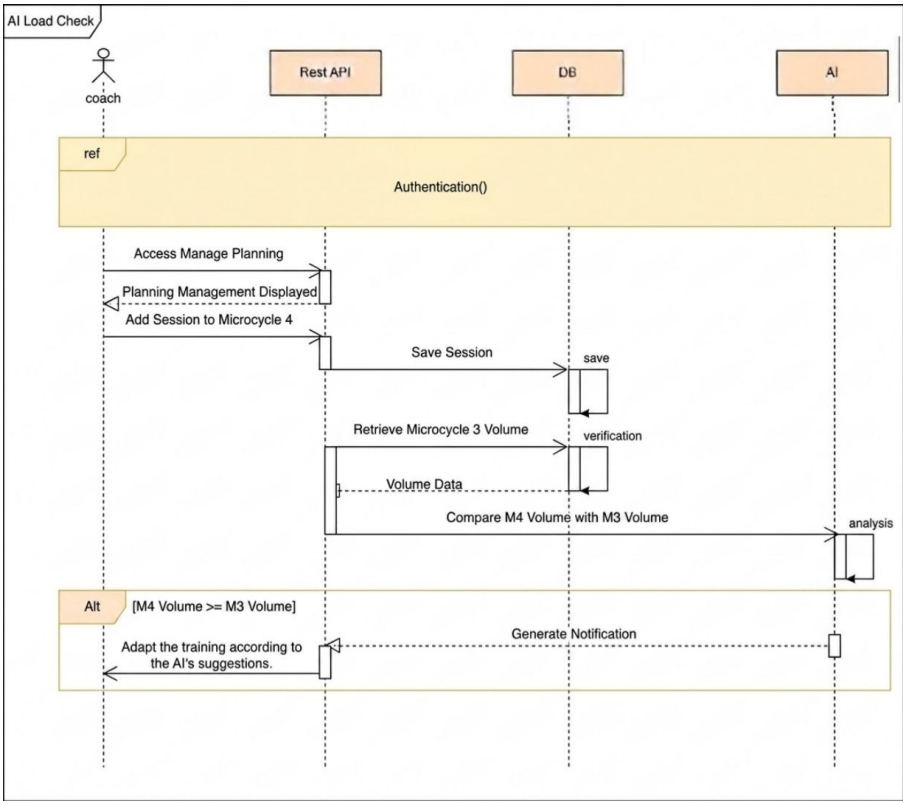


Figure 2.17: Ai Load Check Sequence Diagram

Titre	AI Load Check
Acteur	Coach
Objectif	Vérifier la charge d'entraînement à l'aide de l'IA.
Pré-condition	Coach authentifié.
Post-condition	Recommandation générée par l'IA.
Scénario nominal	1- Le coach accède à la gestion du planning. 2- Il ajoute une séance au microcycle 4. 3- Le système enregistre la séance. 4- Le système récupère le volume du microcycle 3. 5- L'IA compare les volumes des deux microcycles.
Scénario alternatif	**ALT1 :** Si M4 $\geq$ M3 : 1- L'IA génère une notification. 2- Le coach adapte l'entraînement selon les suggestions de l'IA.

Table 2.13: AI Load Check Description

2.5.14 Add performance Diagram

This sequence diagram describes the process of adding an athlete's performance.

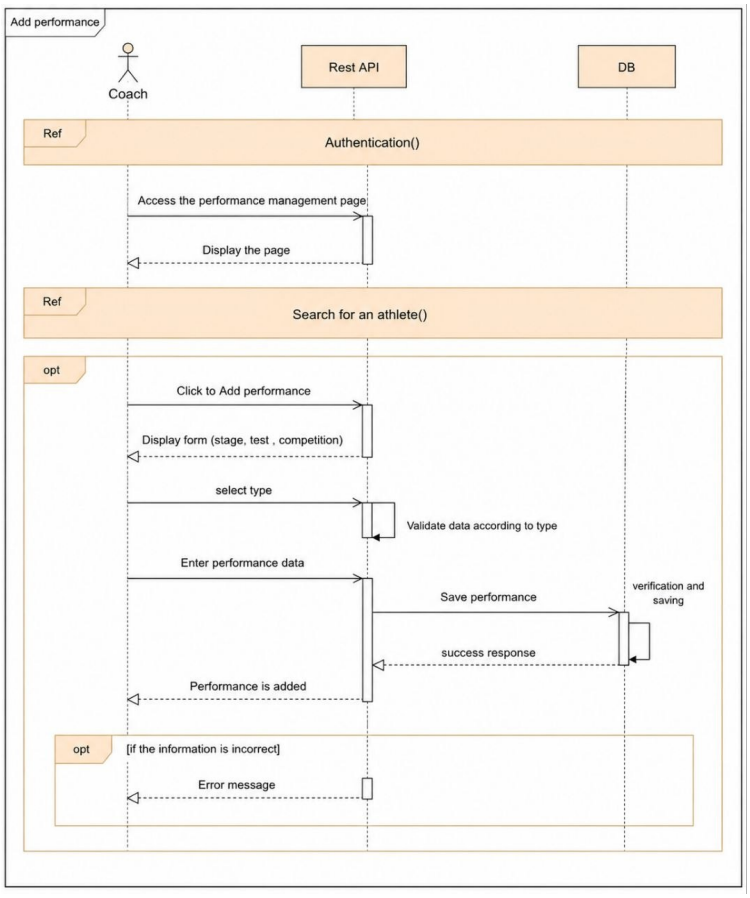


Figure 2.18: Add performance Sequence Diagram

Titre	Add Performance
Acteur	Coach
Objectif	Ajouter une performance d'un athlète
Pré-condition	Coach authentifié et athlète recherché
Post-condition	Performance enregistrée
Scénario nominal	1- Accéder à la page de gestion des performances. 2- Rechercher un athlète. 3- Cliquer sur Ajouter une performance. 4- Sélectionner le type de performance. 5- Entrer les données de performance. 6- Sauvegarder les données. 7- Affichage du message de succès.
Scénario alternatif	Afficher un message d'erreur si les informations sont incorrectes.

Table 2.14: Add Performance Description

2.5.15 Edit performance Diagram

This sequence diagram describes the process of editing an athlete's performance.

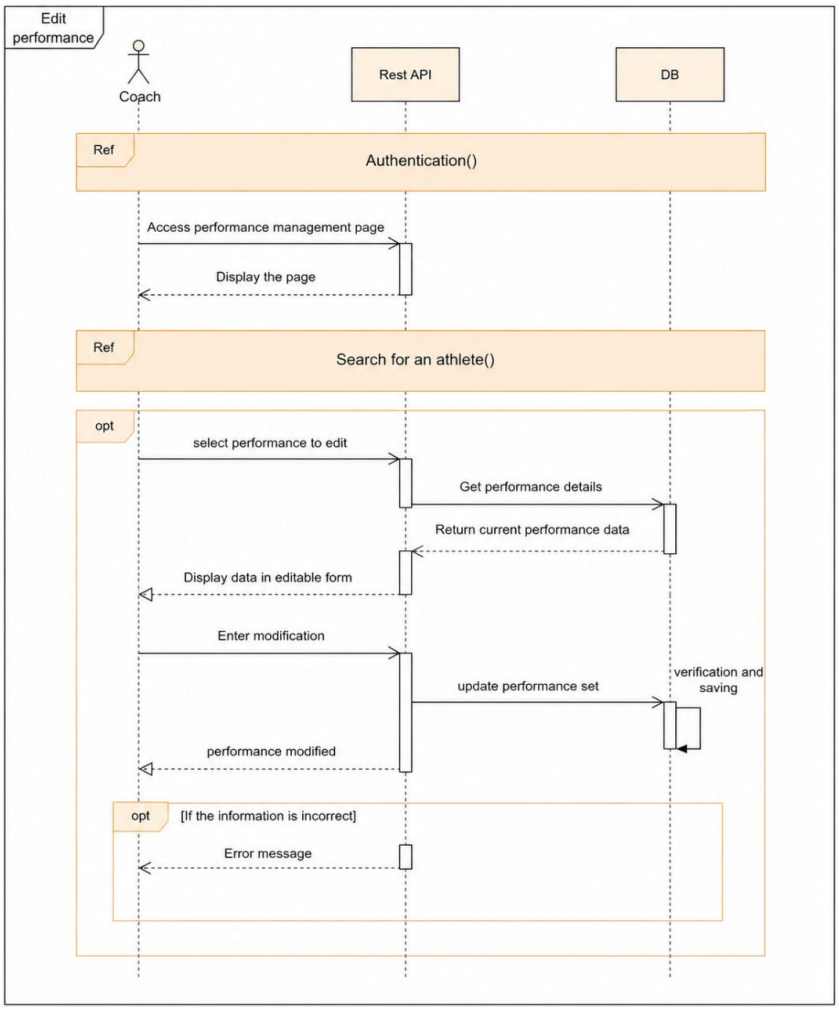


Figure 2.19: Edit performance Sequence Diagram

Titre	Edit Performance
Acteur	Coach
Objectif	Modifier la performance d'un athlète.
Pré-condition	Coach authentifié et athlète sélectionné.
Post-condition	Performance mise à jour ou erreur affichée.
Scénario nominal	1- Sélectionner la performance à modifier. 2- Le système affiche les détails actuels. 3- Saisir les modifications et valider. 4- Le système enregistre et valide les changements.
Scénario alternatif	Si les données sont incorrectes, un message d'erreur s'affiche.

Table 2.15: Edit performance Description

2.5.16 Delete Performance Diagram

This sequence diagram describes the process of deleting an athlete's performance.

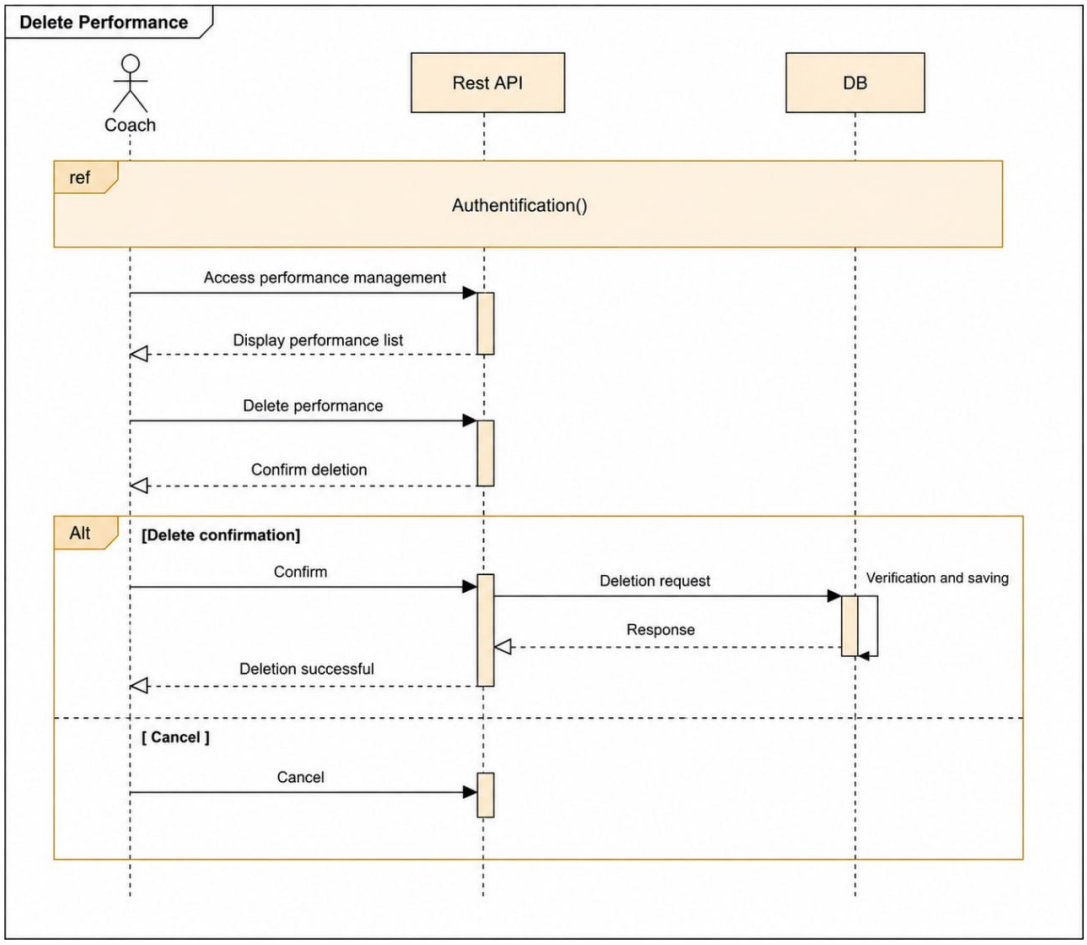


Figure 2.20: Delete Performance Sequence Diagram

Titre	Delete Performance
Acteur	Coach
Objectif	Supprimer une performance d'un athlète.
Pré-condition	Coach authentifié.
Post-condition	Performance supprimée avec succès ou opération annulée.
Scénario nominal	1- Accéder à la gestion des performances. 2- Demander la suppression d'une performance. 3- Confirmer la suppression. 4- Le système valide, supprime la donnée et affiche un message de succès.
Scénario alternatif	Si le coach clique sur "Cancel", la suppression est annulée.

Table 2.16: Delete Performance Description

2.6 Class Diagram

We used class diagrams to represent the static structure of the system, including classes, attributes, and relationships between them.

2.6.1 Class Diagram of the System

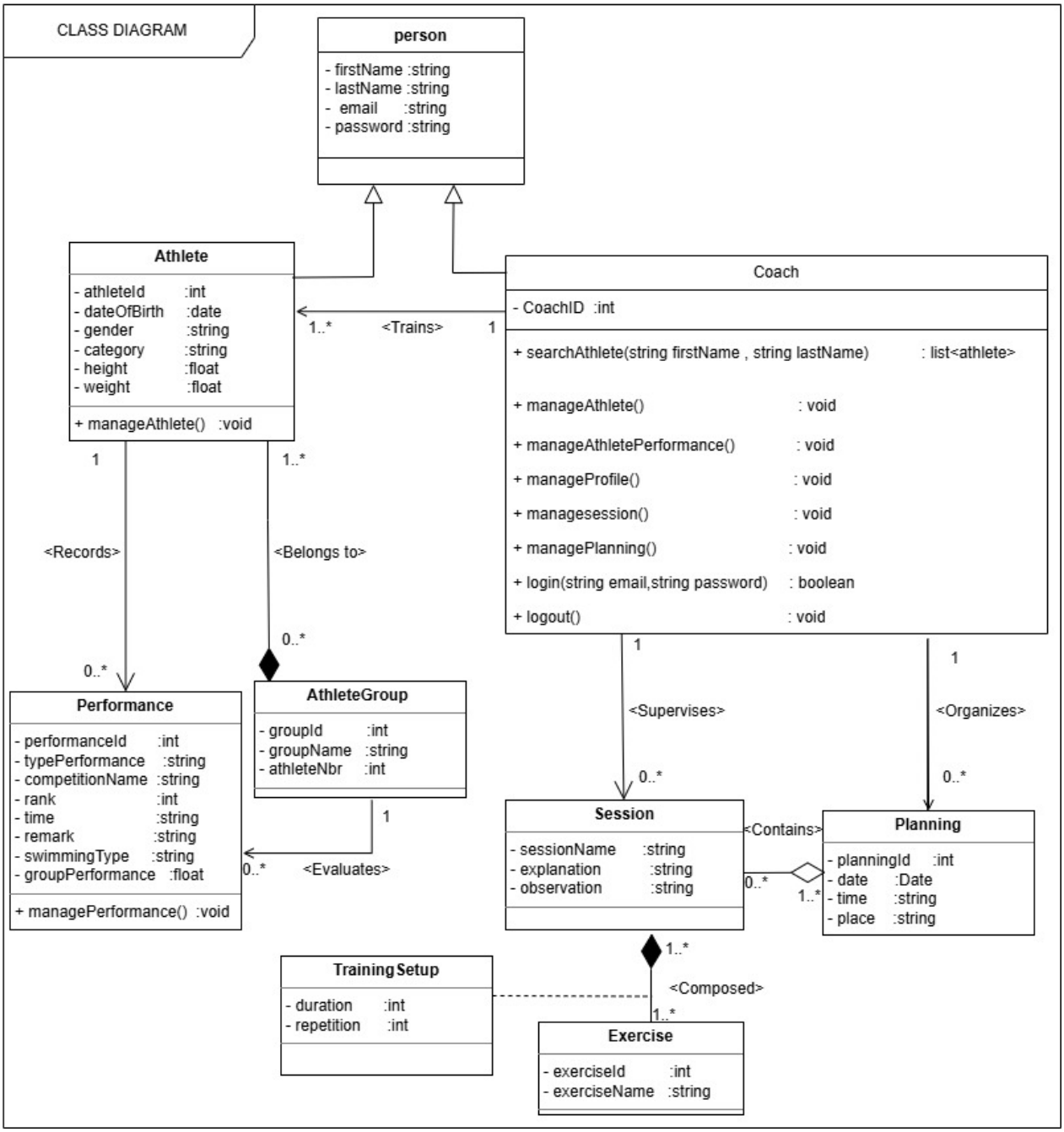


Figure 2.21: Class Diagram of the System

2.7 Dictionary for Class Diagram Attributes

The following table presents the data dictionary of the class diagram. It describes the attributes of each class, their data types, sizes, and identifiers, providing a clear reference for the implementation of the SmartSwim system.

Class	Designation	Code	Type	Size	Comment
Person	First name	firstName	A	50	
	Last name	lastName	A	50	
	Email	email	AN	100	
	Password	password	AN	200	
Athlete	Athlete identifier	athleteId	N	10	Primary key
	Date of birth	dateOfBirth	D	10	
	Gender	gender	A	20	
	Category	category	A	50	
	Height	height	N	5	
	Weight	weight	N	5	
Coach	Coach identifier	CoachID	N	10	Primary key
AthleteGroup	Group identifier	groupId	N	10	Primary key
	Group name	groupName	AN	100	
	Number of athletes	athleteNbr	N	5	
Performance	Performance identifier	performanceId	N	10	Primary key
	Performance type	typePerformance	A	50	
	Competition name	competitionName	AN	100	
	Rank	rank	N	5	
	Time	time	AN	20	
	Remark	remark	AN	200	
	Swimming type	swimmingType	A	50	
	Group performance	groupPerformance	N	10	
Training Setup	Duration	duration	N	5	
	Repetition	repetition	N	5	
session	Session name	sessionName	A	100	
	Explanation	explanation	AN	500	
	Observation	observation	AN	500	
Exercise	Exercise identifier	exerciseId	N	10	Primary key
	Exercise name	exerciseName	A	100	
Planning	Planning identifier	planningId	N	10	Primary key
	Date	date	D	10	
	Time	time	AN	10	
	Place	place	AN	100	

Table 2.17: Data Dictionary

2.8 User Interface desaign

SmartSwim is a web-based platform developed to help swimming clubs manage athletes, training sessions, performance tracking, and communication efficiently through a modular role-based architecture. Its visual identity is inspired by the ACB Natation club (Figure 2.22), from which we adopted the application's color palette and aquatic design style to create a modern and professional interface. The platform uses a clean light-themed design with consistent visual elements that ensure readability, accessibility, and an intuitive user experience through well-organized interfaces adapted to different user roles.



Figure 2.22: Athletic Club Boumerdes Natation Logo.

2.8.1 Color Palette Specification

The following table maps the systematic color applications consistently across the user interface:

Usage	**Hex Code**	**Application Description**
Primary Background	#E1F1FF	Used for the main page background, providing a clean light blue canvas throughout the interface.
Sidebar Background	#CCE5FF	Used as the sidebar background to create a clean, calm, and modern interface.
Selected Menu Item	#CCE5FF	Applied to selected menu items and active navigation elements.
Card Background	#C2E2FF	Used as the background color for cards and information containers.

Table 2.18: SmartSwim Color Palette Specification.

Conclusion

In this chapter, we presented the UML diagrams and the UI design of the SmartSwim project. These elements helped illustrate the system's structure and functionality. In the next chapter, we will present the implementation, technologies, and development of the application.

Chapter 3

The Implementation

Introduction

This chapter presents the implementation of the SmartSwim project and its application interfaces. It also introduces the main technologies and tools used during development. implementation process.

3.1 SmartSwim Development Technologies

To develop SmartSwim, several programming languages, frameworks, and software tools were used. The following section presents these technologies and their contribution to the project.

3.1.1 React

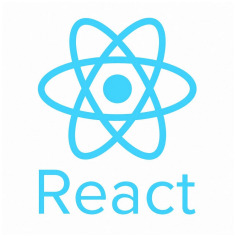


FIGURE 3.1.1 – React platform layout.

React is a JavaScript library for building user interfaces, allowing developers to create complex and interactive UIs out of small and isolated pieces of code called components. [5].

3.1.2 Node.js



FIGURE 3.1.2 – Node.js platform layout.

Node.js is an asynchronous event-driven JavaScript runtime environment designed to build scalable network applications on Google Chrome's V8 engine. [6].

3.1.3 MongoDB



MongoDB is a document-oriented, scalable database built for modern applications, storing data in flexible, JSON-like documents instead of traditional tables. [7].

FIGURE 3.1.3 – MongoDB platform layout.

3.1.4 Tailwind CSS



Tailwind CSS is a utility-first CSS framework packed with classes that can be composed to build any design, directly in your HTML markup. [8].

FIGURE 3.1.4 – Tailwind CSS platform layout.

3.2 Graphical User Interfaces (GUIs)

3.2.1 Authentication Interface (Login)

The safe entry point to the SmartSwim program is represented by this interface. By providing their email addresses and passwords, registered users—administrators, coaches, and athletes—can safely authenticate. Using Role-Based Access Control (RBAC), the system automatically directs the user to their designated area after verifying these credentials against the database.

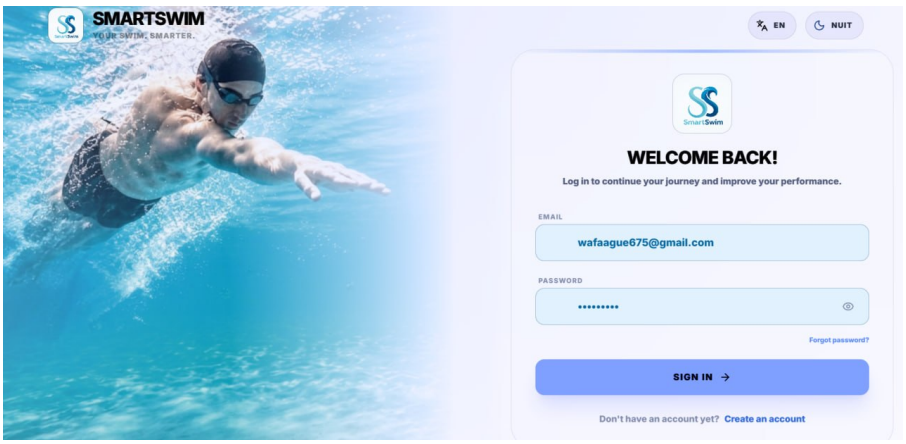


Figure 3.3: Authentication Interface

3.2.2 User Moderation and Account Validation Interface

This interface, which is only available to the system administrator, makes it easier to moderate and manage new registrations. A detailed list of all registered accounts and their current status are displayed. To make sure that only verified coach haS access to the platform, the administrator is given clear action controls to either approve (Valider) or reject (Rejeter) current requests.

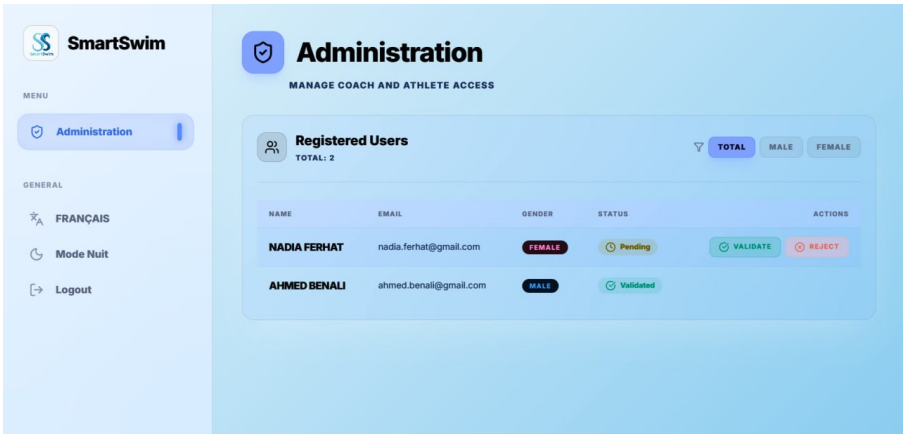


Figure 3.4: User Moderation and Account Validation Interface

3.2.3 Coach Workspace Interfaces

This workspace encompasses all administrative, statistical, and methodological functionalities required by a coach to plan and monitor training sessions.

3.2.3.1 Central Dashboard

The dashboard displays a real-time statistics analysis of the system's active components. For high-level structure management, it analyses monthly growth indicators and compiles the total number of managed athletes, planned training sessions, and routine exercise.

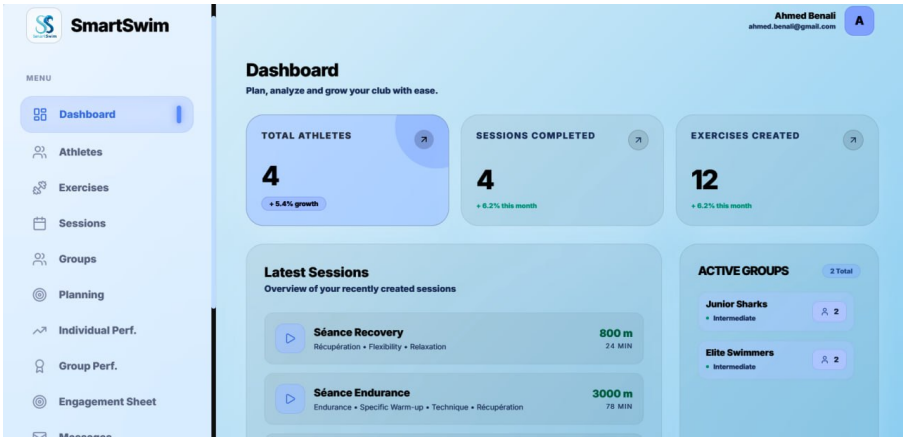


Figure 3.5: Coach Workspace Interfaces

3.2.3.2 Athlete Management Interface

This interface displays a dynamic list that tracks organisations, age groups, and athlete profiles. A new athlete's system account is automatically generated when a coach adds them, and navigating is made easier by built-in filtering, search, and speed functions.

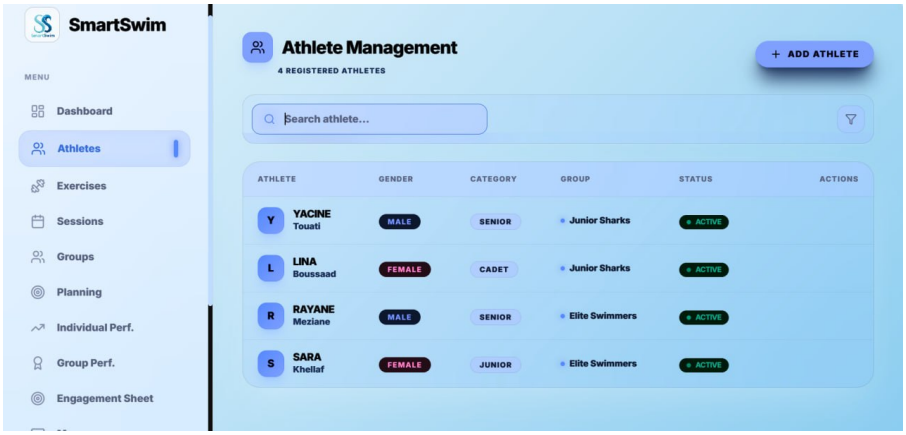


Figure 3.6: Athlete Management Interface

3.2.3.3 Training Squads Configuration Interface

The group management system, which divides swimmers into similar groups according to performance or ability levels (e.g., Junior Sharks, Elite Swimmers), is visualised via this interface. The coach can access the group's member management module to assign new swimmers, change group parameters, and verify the total number of participants via this view.

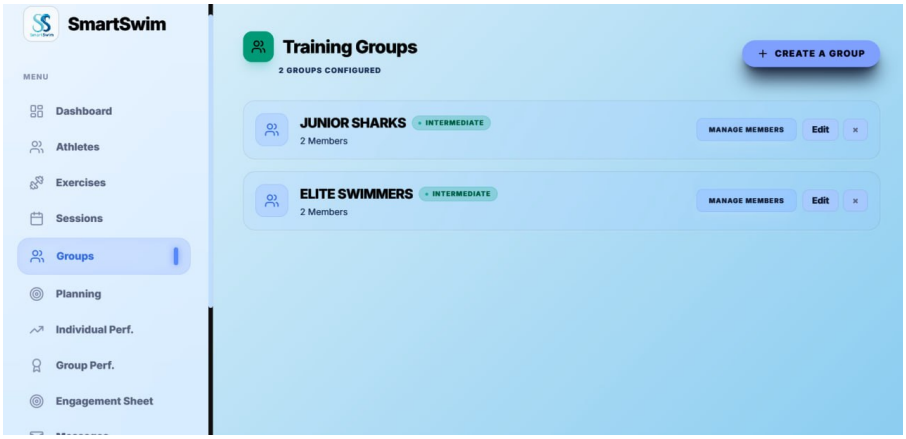


Figure 3.7: Training Squads Configuration Interface

3.2.3.4 Training Workouts Management Interface

This interface shows the central library where coaches manage particular swim techniques and exercises. To guarantee organised athletic advancement, it offers an "Add Exercise" action button in addition to unique categorisation tags for strokes and sorts, such as length and distance limitations.

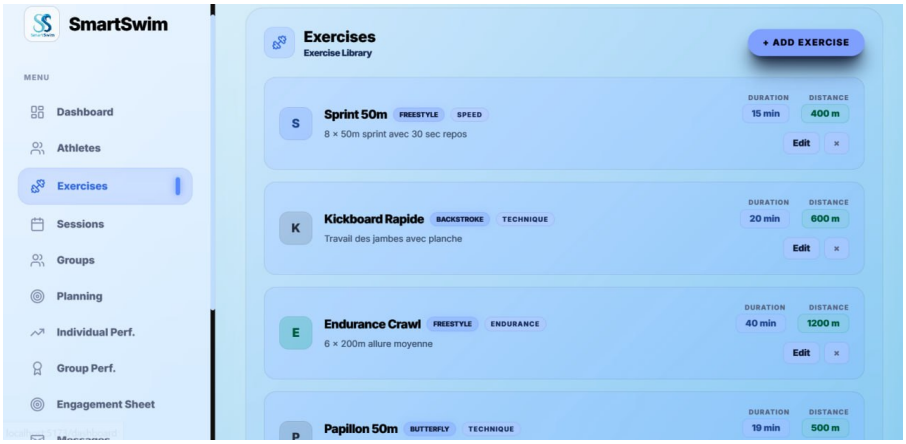


Figure 3.8: Training Workouts Management Interface

3.2.3.5 Training Sessions Management Module

This interface allows coaches to dynamically construct training sessions by aggregating individual drills from the exercise library. The system automatically computes and displays the collective workout distance and total duration, streamlining the verification of planned daily training volume.

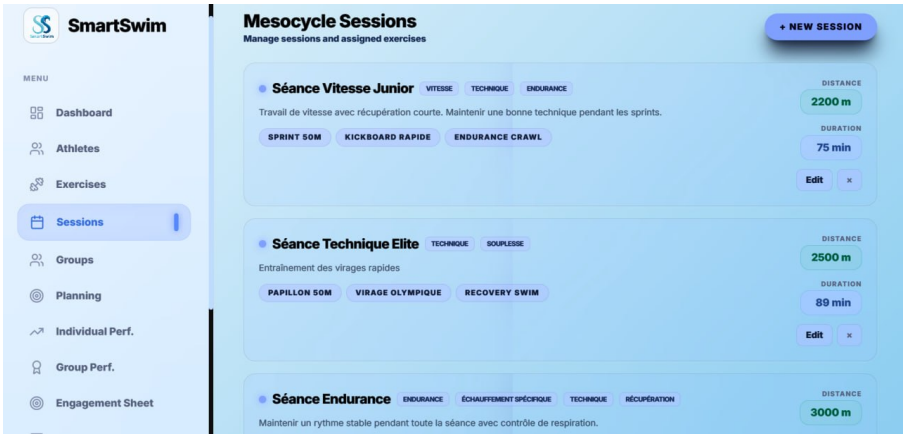


Figure 3.9: Training Sessions Management Module

3.2.3.6 Annual Planning and Periodization Interface

This view highlights the specialized athletic periodization module embedded within the system. It enables coaches to structure a sports season into comprehensive Macrocycles which are further divided into monthly Mesocycles. The system automatically aggregates and displays the total swimming volumes (e.g., 15,700m programmed for the month of May), allowing coaches to properly manage cumulative training workloads.

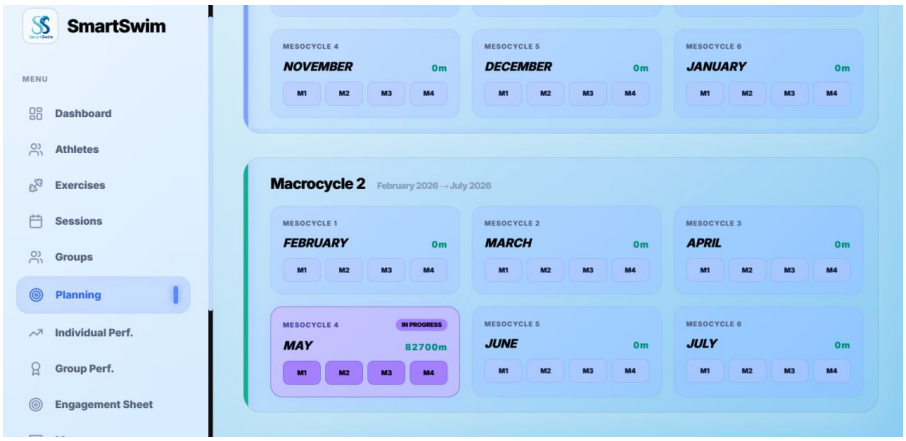


Figure 3.10: Annual Planning and Periodization Interface

An integrated validation bot automatically monitors the progression of training volumes between microcycles. It compares the total swimming volume of Microcycle 4 with that of Microcycle 3 and generates a notification whenever the programmed load exceeds the recommended progression limits, advising the coach to reduce the workload and maintain an appropriate periodization structure.

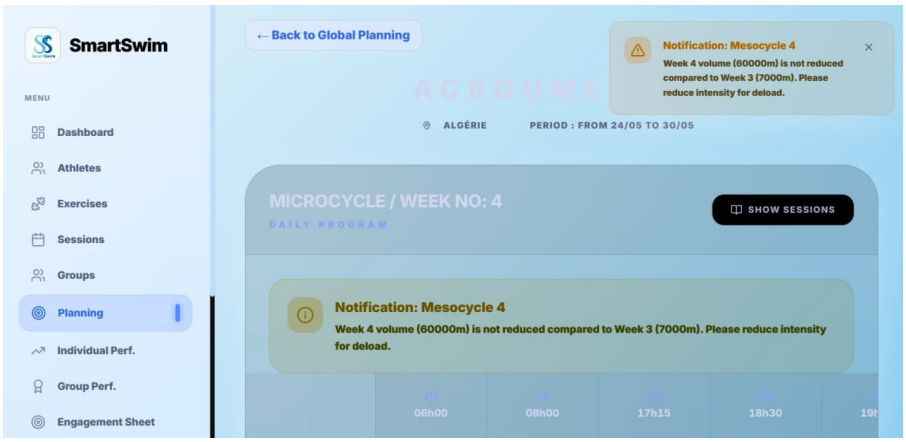


Figure 3.11: Annual Planning and Periodization Interface

3.2.3.7 Individual Performance Analytics Interface

This interface features a data analytics dashboard that tracks individual swimmer progression over time using interactive line graphs and bar charts. It highlights critical athletic metrics such as personal best times, recent trial data, and calculated improvement percentages.

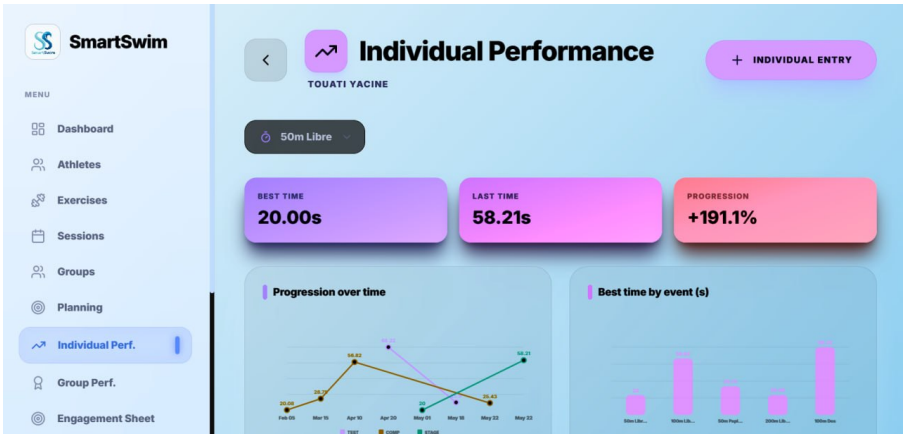


Figure 3.12: Individual Performance Analytics Interface

3.2.3.8 Group Performance Analytics Interface

This screen provides coaches with a comparative visualization platform to analyze overall squad capabilities against target baselines. It displays statistical performance distributions and group averages for specific events, enabling strategic training adjustments.

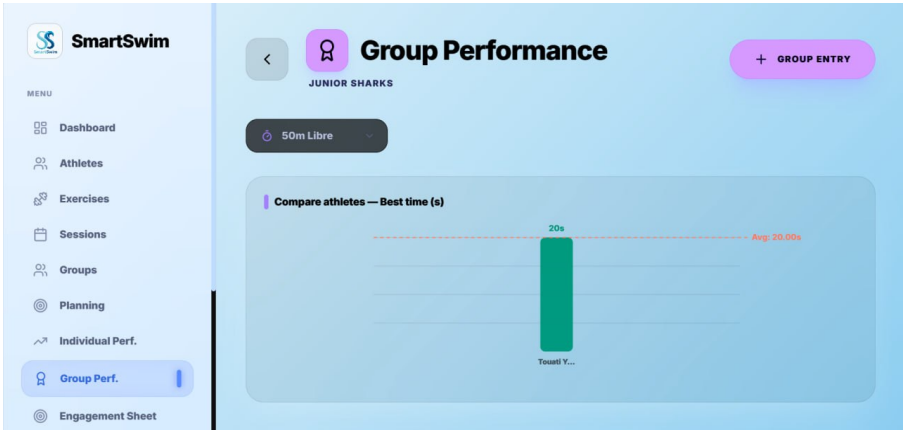


Figure 3.13: Group Performance Analytics Interface

3.2.4 Athlete Workspace Interfaces

This interface represents the personal workspace where athletes can view their biometric data, tracked training performances, and squad assignments. It also features integrated communication tools enabling swimmers to send direct messages to their coach.

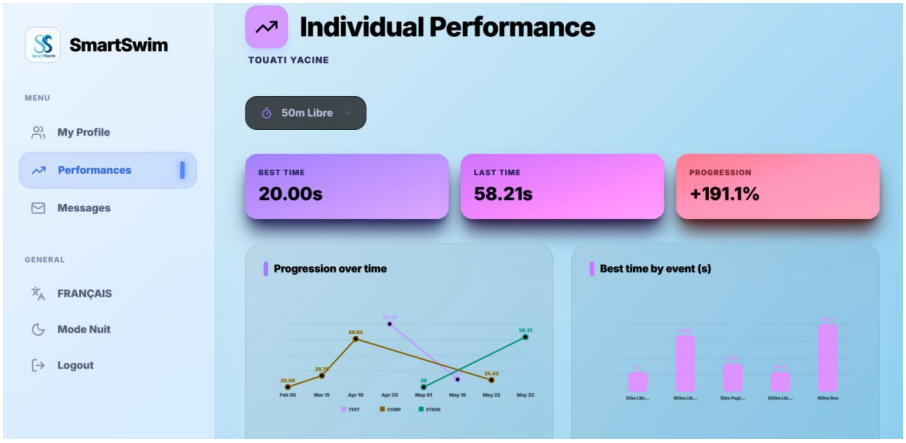


Figure 3.14: Athlete Workspace Interfaces

Conclusion

In this chapter, we presented the implementation of the SmartSwim project, its interfaces, and the technologies used in its development. This stage allowed us to transform the project concept into a functional application.

General Conclusion

In summary, the SmartSwim app is a modern digital solution created to improve swimming team management and training in a clever and efficient way. By offering features to help plan training sessions, analyse swimmers' performance, and precisely assess their results, the app was created to make the jobs of coaches and athletes easier. Additionally, the project has made it possible to combine current programming ideas with the needs of the sports sector, helping to develop a practical framework that facilitates better training choices. Additionally, user-friendly interfaces and an easy user experience that facilitates simple app participation have been addressed. Following athletes' progress, analyzing the volume and intensity of training sessions, assisting coaches in the intelligent planning of training cycles like the Mesocycle, and spotting possible mistakes in the distribution of training load are some of the main features provided by SmartSwim.

This project was more than just a technical application; it was a possibility to hone our analytical, design, and programming abilities while working on a practical project with room to grow. Advanced technologies may be added in the future, such as artificial intelligence, performance prediction, and connecting the app to sports devices to get more precise and real-time data.

Finally, we believe that our study will be a step to the digital transformation of the athletics sector and the creation of modern training methods that benefit players and coaches, improve performance, and provide better results.

In the future, SmartSwim can be extended by developing a mobile application to improve accessibility for coaches and athletes. Other enhancements may include real-time notifications, integration with wearable devices, advanced performance analytics, and more intelligent AI features to support decision-making and optimize training programs.

Bibliography

[1] Commit Swimming. (2026). *Commit Swimming: Swim Team Management Software*. https://www.commitswimming.com/

[2] MySwimPro. *Swim Team Management Software*. URL: `https://www.myswimpro.com/` (Accessed: May 2026).

[3] Swim Coach. *Swim Team Management Software*. URL: `https://swimcoachapp.com/` (Accessed: May 2026).

[4] SwimTopia. *Swim Team Management Software*. URL: `https://www.swimtopia.com/` (Accessed: May 2026).

[5] React. *A JavaScript library for building user interfaces*. URL: `https://react.dev` (Accessed: May 2026).

[6] Node.js. *Node.js JavaScript runtime environment*. URL: `https://nodejs.org` (Accessed: May 2026).

[7] MongoDB. *The database for modern applications*. URL: `https://mongodb.com` (Accessed: May 2026).

[8] Tailwind CSS. *A utility-first CSS framework*. URL: `https://tailwindcss.com` (Accessed: May 2026).