Coach Selector

By:

Maaz Ejaz
35507

Muhammad Umar Farooq
36509

Muhammad Hateem Gulzar
37589

Supervised by:
Mr. Muhammad Usman Karim

Faculty of Computing
Riphah International University, Islamabad
Spring 2025

A Dissertation Submitted To

Faculty of Computing,

Riphah International University, Islamabad

As a Partial Fulfillment of the Requirement for the Award of the Degree of

Bachelors of Science in Software Engineering

Faculty of Computing
Riphah International University, Islamabad

Date: May 15, 2025

Final Approval

This is to certify that we have read the report submitted by **Maaz Ejaz (35507) , Muhammad Umar Farooq (36509)** and **Muhammad Hateem Gulzar (37589)** for the partial fulfillment of the requirements for the degree of the Bachelors of Science in Software Engineering (BSSE). It is our judgment that this report is of sufficient standard to warrant its acceptance by Riphah International University, Islamabad for the degree of Bachelors of Science in Software Engineering (BSSE).

Committee:

1 ___________________________
Muhammad Usman Karim
(Supervisor)

2 ___________________________
Dr. Musharraf Ahmed
(Head of Department/chairman)

Declaration

We want to make it clear that our document, **"Coach Selector"** is entirely our own work, and we haven't copied it from anywhere. We've put in our personal efforts to complete this project along with the report. Our teachers, especially our supervisor **Mr. Muhammad Usman Karim**, guided us throughout the process. We promise that none of the content in our system is taken from any other source or copied from another project. If it's proven otherwise, and any part is found to be a reproduction of someone else's work, we're ready to face the consequences. This declaration reflects our commitment to honesty and accountability in our academic endeavors.

____

Maaz Ejaz

35507

____

Muhammad Umar Farooq

36509

____

Muhammad Hateem Gulzar

37589

Dedication

Our heartfelt dedication reaches out to the pillars of our journey – our cherished family, steadfast friends, and dedicated teachers. Their unwavering encouragement and invaluable support have been the bedrock of our success in bringing **"Coach Selector"** to life. A special shout-out goes to our parents for their supreme encouragement, laying the foundation for our endeavors. We extend sincere gratitude to our teachers for their exceptional support, guiding us with wisdom and care on our academic path. In genuine appreciation, we dedicate this work to our esteemed supervisor, **"Mr. Muhamad Usman Karim"**. His unwavering commitment, continuous guidance, and transformative mentorship played a pivotal role in turning our ideas into a tangible reality. With deep thanks, we acknowledge the profound impact of these wonderful individuals on our journey. Their collective influence has not only shaped our success but has also added a rich dimension to our academic pursuits, making this endeavor all the more meaningful and fulfilling.

Acknowledgement

Foremost, we express our gratitude to Allah Almighty, the Merciful and the source of all knowledge, for endowing us with the courage and wisdom to bring this project to fruition. In His infinite mercy, we acknowledge the divine guidance that has been instrumental in our journey. Our heartfelt thanks extend to our family, friends, and teachers for their unwavering support, and to our supervisor, **Muhammad Usman Karim**, for his transformative guidance. Together, these influences have shaped our success and made this endeavor a manifestation of shared dedication and divine blessings.

Maaz Ejaz

35507

Muhammad Umar Farooq

36509

Muhammad Hateem Gulzar

37589

Abstract

Our platform is a next-generation system designed to revolutionize the relationships and management within sports training, connecting sports academies, coaches, players, and parents. Geolocation features make it easy to find specific coaches or academies, and users can explore a wide range of session booking options tailored to different training programs. Advanced tools such as detailed filters and customizable training plans help athletes find coaches based on their area of specialization. A modern booking system provides real-time availability, scheduling, and pricing, allowing users to conveniently plan their training sessions. The platform also includes a GPS-powered tracking system that monitors player participation, performance, injuries, and attendance, with automated notifications sent to parents and coaches. In addition, it supports the organization of coaches, equipment, rehabilitation aids, and nutrition schedules—ensuring comprehensive sports administration.

Table of Contents

Chapter: 01: Introduction

1.1 Introduction……………………………………………………..…………..………3

1.2 Opportunity & Stakeholders 3

Stakeholders 4

Players. 4

Coaches. 4

Academies. 4

Parents/Guardians. 4

Administrators (Platform Admin). 4

1.3 Motivation and Challenges ..…………………………...…………………………3

1.3.1 Motivations of Coach Selector: 4

1.3.2 Challenges of Coach Selector: ………………………..……..…………………4

1.4 Goals and Objectives ………………………………………………………………4

1.4.1 Increase Awareness 5

1.4.2 Generate Leads 5

1.4.3 Build Brand Recognition 5

1.4.4 Increase Bookings...5

1.5 Solution Overview 6

1.5.1 Project Scope 6

1.5.1.1 Admin Module ………………………………..……………………………5

1.5.1.2 Client ……………………………………………………………………..5

1.6 Report Outline...6

1.6.1 Chapter..………………………………………………………………6

1.6.2 Chapter 2 ……..……………………………………..………………6

1.6.3 Chapter 3 ………………………………………………………………6

1.6.4 Chapter 4 ………………………………………………………………6

1.6.5 Chapter 5 ………………………………………………………………6

1.6.6 Chapter 6 ……………………………………………………………..6

Chapter: 02: Literature & Market Survey

2.1 Introduction..9

2.2 Literature Review...9

2.3 Summary 10

Chapter:03: Requirements Analysis

3.1 Introduction……………………..……………………………………………………..11

3.2 Problem Scenarios 12

3.3 Functional Requirements 14

3.3.1 Admin Management………………………………………………………………13

3.3.2 Coach Management..14

3.3.3 Player Management...15

3.3.4 System Management 15

3.3.5 Parent Management...15

3.3.6 Academy Management ……………………………………..…………………14

3.4 Non-Functional Requirements 16

3.5 SQA Activities Defect Detection:..16

3.5.1 Test case Design ……………………………………………………………...16

3.5.1.1 User Registeration 16

3.5.1.2 Assign Equipments..18

3.5.1.3 Academy..2019

3.5.1.4 Attendance...210

3.5.1.5 Check-out Form 221

3.5.1.6 Coaches 242

3.5.1.7 Coach Shedule...253

3.5.1.8 Feedback Form...264

3.5.1.9 Payment 25

3.5.1.10 Player 26

3.5.1.11 Player Parent 27

3.5.1.12 Player score 28

3.5.1.13 Post 29

3.5.1.14 Rating 30

3.5.1.15 Request Equipment 31

3.5.1.16 Return Equipment 32

3.5.1.17 Sports Category 33

3.5.1.18 Videos 34

3.6 Summary ……………………………………………………………………34

Chapter:04: System Design

4.1 Introduction..36

4.2 Model View Controller Diagram (MVC) 36

4.3 Database Design………………………………..……………………………37

4.3.1 ERD Diagram…………………………………………………………..37

4.4 Activity Diagram....……………………..………………………………….38

4.4.1 Admin...38

4.4.2 Coach………………………….;…………………………………………39

4.4.3 Player……………………………………………………………………...40

4.4.4 Parent......…………………………………………………………………41

4.5 Use Case Diagram...4142

4.5.1 Admin…………………..………………………………………………..43

4.5.2 Coach………………………..…………………………………………44

4.5.3 Player……………………………………………………………………..45

4.5.4 Parent……………………………………………………………………..46

4.5.5: FULLY USE-CASE DIAGRAM: 466

4.6 Fully Address Use Cases 47

4.6.1 Admin...47

4.6.2 Coach..499

4.6.3 Player…………………………..……………………………………………54

4.6.4 Parent………………………...……………………………………………57

4.6.5 System…………………………..………………………………………58

4.6.6 Academies…………………...…………………………………………59

Chapter 05: Implementation:

5.1 Endeavour……………………………………………………………………61

5.1.1 Team……………………………………………………………………61

5.1.2 Work Breakdown Structure ………………………………………………61

5.2 Components, Libraries, Web Services& stubs …………………………………66

5.2.1 Back-end …………………………………………………………………...…66

5.2.2 Database……………………………………………………………………..66

5.3 IDE,Tools& Technologies ……………………………………………………67

5.4 Best Practices / Coding Standard..67
5.5 Deployment Diagram...68

Chapter 06: Testing:

6.1 Introduction …………………………………………………………………………..70

6.2 White-Box Testing …………………………..………………………………………70

6.2.1 Role in signup……………………………………………………………………70

6.2.1.1 Control flow…………………………..……………………………………..71

6.2.1.2 Coach Flow…………………………………………………………………72

6.2.1.3 Player Flow…………………………………………………………………72

6.2.2 Attendance, Scoreboard, Equipment’s ……………………………………………….72

6.2.2.1 Attendance……………………………………………………………………72

6.2.2.2 Scoreboard……………………………………………………………………73

6.2.2.3 Equipment’s……………………………………………………………………74

Chapter: 07 Conclusion and Outlook

7.1: Introduction…………………………………………………………………………76

7.2: Achievements and Improvements…………………………………………………….76

7.3: Critical reviews ………………………………………………………………………76

7.4: Future recommendation/Outlook …………………………………………………….77

7.5 Summary…………………………………………………………………………....77

List of Figures:

Figure 4.1: Model View Diagram …..………………………………………………. 36
Figure 4.2: Entity Relationship Diagram ..………………...……………………….. 37
Figure 4.3: Admin Activity Diagram …………………………………………….…. 38
Figure 4.4: Coach Activity Diagram ……………………………………………….. 39
Figure 4.5: Player Activity Diagram ………………………………………………..40
Figure 4.6: Parent Activity Diagram ……………………………………………….. 41
Figure 4.7: Admin Use-case Diagram ……………………………………………… 42
Figure 4.8: Coach Use-case Diagram ……………………………………………….43
Figure 4.9: Player Use-case Diagram ………………………………………………. 44
Figure 4.10: Parent Use-case Diagram …………………………………………….... 45
Figure 4.11: fully Use-case Diagram ……………………………………………….. 46

Chapter 1: Introduction

Chapter 1: Introduction

1.1 Introduction

Coach Selector is an all-inclusive suite of web and mobile-based applications set to effectively manage a sports academy, coaches, players, and parents- simplifying key processes such as looking for the right specialized coach, booking sessions, and tracking player progress, while negating problems with communication gaps and scheduling. Once signed up, academies can track their coaches and players, and parents can track the performance of their children, attendance, and injuries, mainly if the child is an athlete below 18.

The website provides a high search and filter mechanism in order to find a coach based on location and area of expertise; there is an easy booking system for making the arrangements of coaching sessions. Coaches may track the performance, attendance, and nutrition plans of their charge and therefore may be able to produce nice and detailed progress reports and performance analytics. The injury management module sends alert notifications to parents during emergency situations, and players can also reserve coaches for their training at any preferred location. Thus, it makes sure there are real-time updates that provide communication efficiency and this system acts as a complete solution for sports training and management.

1.2 Opportunity & Stakeholders

There are important opportunities through the platform to develop sports training. It makes communication better by providing a single system where players, coaches, and parents can easily talk to each other without any gaps that characteristically happen when tracking developments and planning sessions. The session management feature lets players book coaching sessions based on real-time availability, which assists with better coordination and time saving. Coaches can watch and keep track of how players are doing, their attendance, and what they eat. They give personal feedback and make detailed reports to help players get better. Parents, especially of athletes under 18, can be involved through a special portal. This portal shows them their child’s progress, attendance, and any injuries. It also comes with a strong injury management and emergency support system. It gets in touch with parents right away and ensures that good rehabilitation plans are ready. Specialized and personalized training is also provided through the use of advanced search and filters to find special coaches. Detailed analytics and performance reports provide information regarding where improvements can be made. This platform presents a holistic and functional means of sports training management, helps grow players, and enhances communication among everyone involved.

1.2.1 Stakeholders

1. Players.
2. Coaches
3. Academies.
4. Parents/Guardians.
5. Administrators (Platform Admin).

1.3 Motivation and Challenge

Coach selector aims to make it easier for sports training by connecting players, parents, and coaches. It offers session booking tools, track progress functionality, and injury management. The aim here is to make sure communication and organization among young athletes (Students) improves and that everyone involved receives any support they might need. Challenges include handling data, maintaining privacy with the information kept there, growth, and real-time communication and notifications. Overcoming these is what would propel the development of a comprehensive and easy-to-use sports management system.

1.3.1 Motivations of Coach Selector:

1. Growing demand for organized sports training and management.
2. Need for better communication between coaches, players, and parents.
3. Simplifying the process of finding specialized coaches.
4. Providing transparency for parents, especially for under-18 athletes.
5. Tracking player performance, attendance, and injuries for continuous improvement.
6. Convert Sport Academy Record into Portal to Easily Track our Player record

1.3.2 Challenges of Coach Selector:

1. Managing player data and real-time updates.
2. Ensuring user adoption and ease of use for all stakeholders.
3. Protecting sensitive information and maintaining privacy.
4. Scaling the platform efficiently as the user base grows.
5. Implementing real-time communication and notifications.
6. Accurate injury and emergency tracking with timely notifications.
7. Score Board of Each Player where store daily record of Player

1.4 Goals and Objectives

Coach Selector marketing project goals can be to:

1.4.1 Increase Awareness

The primary objective of the sports management platform is to increase awareness among players, coaches, and parents about the benefits of using the system for managing training, tracking performance, and booking sessions. The platform introduces users to its features like session scheduling, progress tracking, and injury management, aiming to enhance the overall sports experience.

1.4.2 Generate Leads

Another objective is to generate leads for sports academies and individual coaches. The platform includes a call-to-action (CTA) that encourages players and parents to sign up, explore available coaches, and book training sessions, facilitating smooth connections between athletes and coaching professionals.

1.4.3 Build Brand Recognition

The platform helps build brand recognition for coaches and academies, making it easier for them to stand out in the competitive sports training industry. By offering an intuitive, feature-rich system, the platform helps coaches establish lasting relationships with players and their families.

1.4.4 Increase Bookings

Ultimately, the objective is to increase session bookings and overall engagement. By making it easier to find coaches, book training sessions, and track progress, the platform aims to drive more frequent training, leading to improved performance and a steady revenue stream for coaches and academies.

1.5 Solution Overview

A successful implementation of the sports management platform involves identifying the needs of players, parents, and coaches, creating a user-friendly system, offering key features like session management, real-time chat, and injury tracking, and continually updating the platform for better user experience and improved performance.

1.5.1 Project Scope

1.5.1.1 Admin Module:

1. Manage coach and academy registrations.
2. Oversee player performance tracking and reporting.
3. Handle session scheduling and progress management.
4. Manage nutrition and injury tracking modules.
5. Manage payment processing and invoicing.

1.5.1.2 Client (Coaches/Parents):

1. User registration (for coaches, players).
2. Book and manage training sessions.
3. Track player progress, injuries.
4. Monitor attendance and generate detailed reports.
5. Access parent portals to track under-18 player activities.

1.6 Report Outline

Chapter 1 explains the overall workflow of our sports management platform, including its goals and objectives, and how these will be achieved. This chapter details the motivation for developing the system, the problems faced in the existing sports training and management processes, and how our solution meets the expectations of players, coaches, and parents. The scope of the project is clearly outlined, along with the key modules and their respective features. This chapter covers all functional aspects of the platform, including session booking, progress tracking, injury management, and parental monitoring.

1.5.1 Chapter 1

In this chapter, we introduce our system, outlining the needs and technical challenges it addresses. We delve into the goals, objectives, and methodologies employed in developing the proposed solution.

1.6.2 Chapter 2

This chapter delves into the current market landscape, highlighting the unique and practical aspects of our system that cater to user needs.

1.6.3 Chapter 3

Focused on need analysis, this chapter examines the genuine requirements and problem scenarios tackled by the developed system. It also identifies the actual users of the system and those impacted by it.

1.6.4 Chapter 4

Providing insights into the design aspects of the developed system, this chapter details the system's architecture design considerations and presents various diagrams modeling the operational behavior of the system.

1.6.5 Chapter 5

This chapter furnishes information about the environment in which the developed system was deployed and evaluated. It covers aspects such as performance, usability, and other pertinent considerations.

1.6.6 Chapter 6

This chapter focuses on the systematic process of white-box testing, specifically applied to various functionalities within the application such as user signup, role-based workflows (coach and player), and management modules like attendance, scoreboard, and equipment handling.

Chapter 2:

Literature/Market Survey

Chapter 2:

Literature/Market Survey

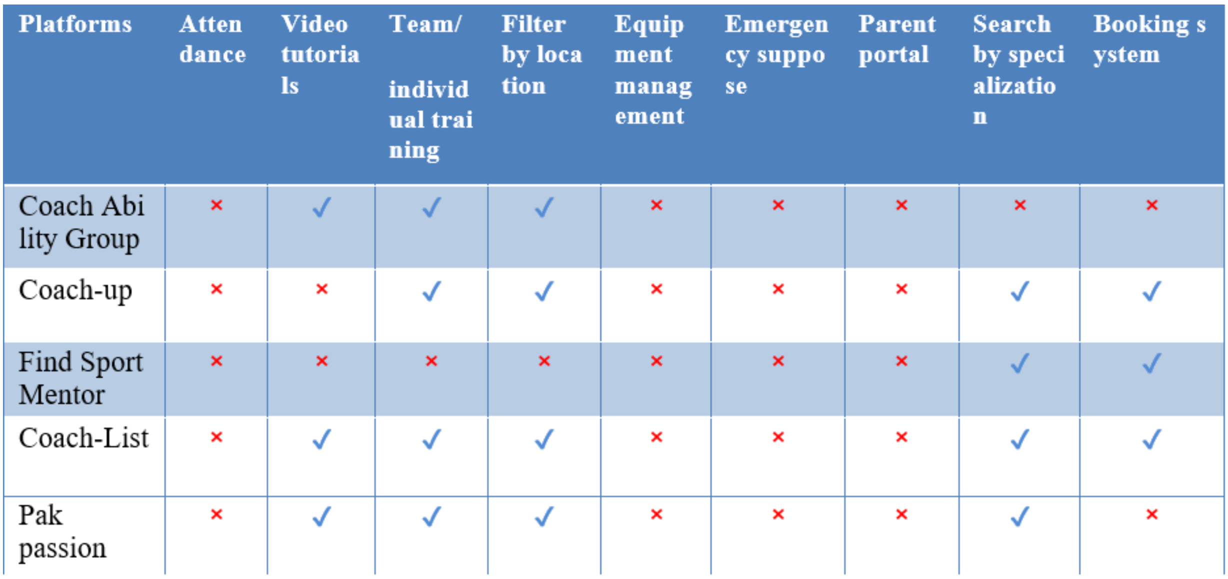
2.1 Introduction

Analyzing existing sports management systems is essential for developing improved solutions, as it provides valuable domain knowledge and insights. To gather relevant data, we conducted a market survey and interacted with coaches, players, and parents involved in sports training and management. Through this process, we identified that current systems lack comprehensive functionality, often failing to provide seamless coordination between coaches, players, and parents. Many systems lack proper tracking of athlete progress, session scheduling, and parental involvement, forcing coaches to rely on manual processes, which are time-consuming and inefficient. This market analysis revealed a gap in the market for a more streamlined and efficient sports management platform. By gathering data, we gained insights into what users need, allowing us to develop a platform that simplifies session management, enhances communication, and tracks player performance and injuries. Our findings highlight the potential to create a solution that not only addresses existing system limitations but also offers a cost-effective and user-friendly platform to meet the evolving needs of athletes, coaches, and parents.

2.2 Literature Review

We performed a thorough search of existing sports management platforms available online. To compare these systems, we evaluated various features such as session scheduling, player performance tracking, communication tools, and parental involvement. Our analysis helped identify common functionalities across all systems, such as basic session management, while also highlighting unique features like injury tracking and real-time chat that are only found in certain platforms. This comparison allowed us to build a solution that combines essential functionalities with innovative features that cater to the needs of all stakeholders. We analyzed following systems:

Table 2.1: Existing Systems



Platforms	Attendance	Video tutorials	Team/ individual training	Filter by location	Equipment management	Emergency suppose	Parent portal	Search by specialization	Booking system
Coach Ability Group	×	✓	✓	✓	×	×	×	×	×
Coach-up	×	×	✓	✓	×	×	×	✓	✓
Find Sport Mentor	×	×	×	×	×	×	×	✓	✓
Coach-List	×	✓	✓	✓	×	×	×	✓	✓
Pak passion	×	✓	✓	✓	×	×	×	✓	×

2.3 Summary

Based on our survey of different sports management platforms and discussions with coaches, academies, players, and parents, we have identified that while existing systems offer some basic functionalities, there are several key features that are either missing or need significant improvement. Areas such as player performance tracking, injury management, real-time communication, and parent involvement require enhancement. This gives us confidence that we can introduce new and innovative features to our platform, making it more comprehensive and better suited to the needs of our users. With feedback and input from stakeholders, we are committed to refining our platform by incorporating these improvements, making it more efficient in managing sports sessions, tracking athlete progress, and ensuring seamless communication.

Chapter 3: Requirement Analysis

Chapter: 03

Requirement Analysis:

3.1 Introduction:

In this chapter we will discuss all functional and nonfunctional requirements of Business to Door. And we will discuss the needs and problem statement due to which we are developing this system.

3.2 Problem Scenarios

Table 3.1: Problem statement 1

Case1:	**Lack of Transparency in Coach-Player Interactions**
Problem	Players and parents often have limited visibility into the coach's training methodology, fees, and session outcomes.
Affects	Miscommunication leads to mistrust, dissatisfaction, and unfulfilled expectations
Result	Coaches face difficulty retaining players, and parents hesitate to invest in sports training.
Benefits	A transparent platform with session logs, fee structures, and progress tracking builds trust and enhances engagement between coaches, players, and parents.

Table 3.2: Problem statement 2

Case2:	Inefficiency in Talent Identification and Development
Problem	Sports talent at the grassroots level goes unnoticed due to a lack of structured scouting or coaching programs.
Affects	Players with potential remain untrained, limiting their career prospects
Result	Stunted growth of the national sports ecosystem and missed opportunities for global recognition.
Benefits	A transparent platform with session logs, fee structures, and progress tracking builds trust and enhances engagement between coaches, players, and parents.

Table 3.3: Problem statement 3

Case3:	High Dropout Rates in Sports Training
Problem	Many players quit training due to demotivation, lack of progress, or misaligned training styles.
Affects	Players lose confidence, and coaches face declining enrolments
Result	A cycle of reduced player retention and wasted potential..
Benefits	Regular progress updates, personalized training plans, and feedback mechanisms keep players motivated, fostering long-term commitment.

Table 3.4: Problem statement 4

Case4:	Difficulty in Monetizing Coaching Expertise
Problem	Many skilled coaches fail to monetize their knowledge effectively due to poor marketing and lack of access to clients.
Affects	Coaches face financial instability, and their expertise remains underutilized.
Result	Talented coaches exit the profession, further reducing quality coaching availability
Benefits	The platform connects coaches to players, offers tools for pricing strategies, and facilitates secure payments, enabling sustainable careers for coaches.

3.3 Functional Requirements

3.3.1 Admin (platform Admin)

1. Admin should be able to verify the coach.
2. Admin should be able to add, update, delete the sports categories.
3. Admin should be able to see the review of the coaches.
4. Admin should be able to block the profile of Coach on the basis of Reviews.

3.3.2 Coach Management

1. Coach should be able to create our profile.
2. Coach Should be able to create an academy.
3. Coach should be able to create posts.
4. Coach should be able to accept & decline the player booking request.
5. Coach should be able to chat with the player.
6. Coach should be able to verify the payment

7. Coach should be able to mark the attendance of the player.
8. Coach should be able to manage the score board.
9. Coach should be able to manage the emergency record.
10. Coach should be able to assign the sport equipment to players

3.3.3 Player Management

1. Players should be able to create our profile.
2. Players should be able to view the posts of available coaches.
3. Players should be able to search and filter the coach by location, name.
4. Players should be able to book the coach.
5. Players should be able to pay the payment to the coach.
6. Players should be able to chat with coaches.
7. Players should be able to see our attendance record.
8. Players should be able to see our weekly/Daily progress.
9. Players should be able to add review of the coach after the session.
10. Players should be able to request the coach for the equipment of available stock.

3.3.4 System Management

1. System should be able to connect the player and coach.
2. System should be able to assign the player guardian portal.
3. System should be able to generate the certificate to the player.

3.3.5 Parent Management

1. Parents should be able to see the player's attendance.
2. Parent should be able to see the progress of the player
3. Parents should be able to see the Injury record of the player.

3.3.6 Academy management

1. Coach should be able to maintain the players.
2. Coach should be able to maintain the progress of the player.

3. Coach should be able to share the progress report with parents.

3.4 Nonfunctional requirements

There is no specific NFR in our system.

3.5 SQA Activities Defect Detection:

We applied black box testing techniques for defect detection and designed our test cases.

3.5.1 Test Case Design:

Functional Requirement Test Cases

3.5.1.1 User Registration

Table 3.5.: User Registration

Input	Valid Class	Invalid Class

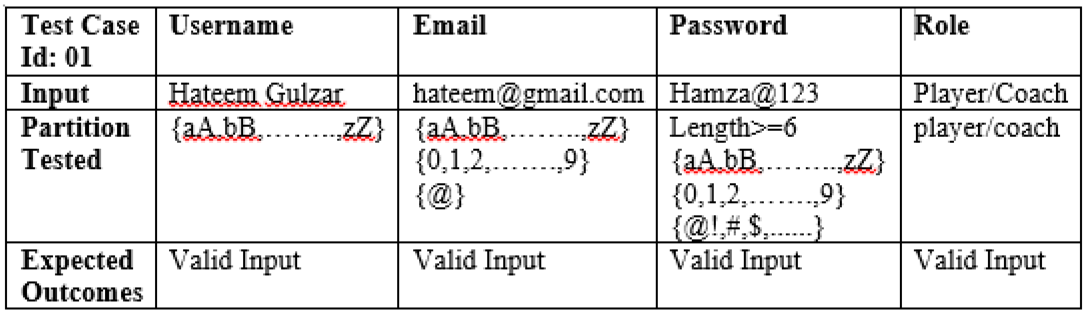
Username	{aA,bB,…..,zZ}
	{0,1,2,….,9}
		Outside valid class

Email	{aA,bB,…..,zZ}
	{0,1,2,….,9} {@}
		Outside valid class

Password	Length>=8
	{aA,bB,…..,zZ}
	{0,1,2,….,9}
	{!,@,#,…}
		Outside valid class

Confirm Password	Length>=8
	{aA,bB,…..,zZ}
	{0,1,2,….,9}
	{!,@,#,…}
		Outside valid class

Table 3.6: User Registration Valid

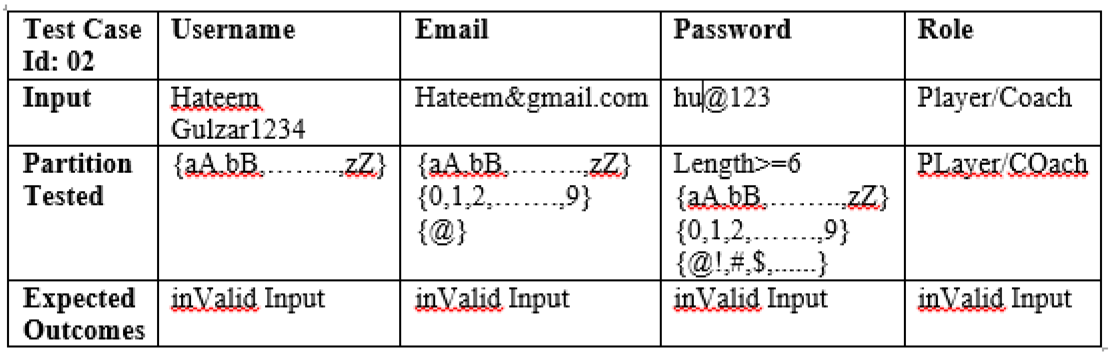


Test Case Id: 01	Username	Email	Password	Role
Input	Hateem Gulzar	hateem@gmail.com	Hamza@123	Player/Coach

Partition Tested	{aA,bB,………,zZ}	{aA,bB,………,zZ}
		{0,1,2,……..,9}
		{@}
			Length>=6
			{aA,bB,………,zZ}
			{0,1,2,……..,9}
			{@!,#,$,......}
				player/coach

Expected Outcomes	Valid Input	Valid Input	Valid Input	Valid Input

Table 3.7: User Registration In-Valid

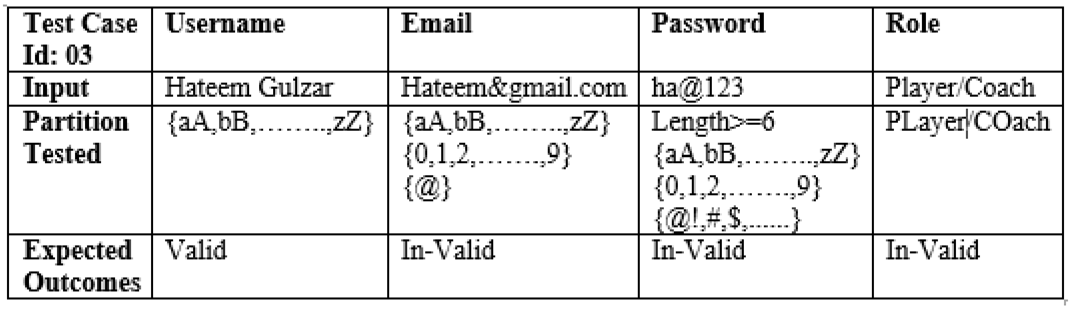


Test Case Id: 02	Username	Email	Password	Role
Input	Hateem Gulzar1234	Hateem&gmail.com	hu@123	Player/Coach

Partition Tested	{aA,bB,………,zZ}	{aA,bB,………,zZ}
		{0,1,2,……..,9}
		{@}
			Length>=6
			{aA,bB,………,zZ}
			{0,1,2,……..,9}
			{@!,#,$,......}
				PLayer/COach

Expected Outcomes	inValid Input	inValid Input	inValid Input	inValid Input

Table 3.8: User Registration In-Valid



Test Case Id: 03	Username	Email	Password	Role
Input	Hateem Gulzar	Hateem&gmail.com	ha@123	Player/Coach

Partition Tested	{aA,bB,………,zZ}	{aA,bB,………,zZ}
		{0,1,2,……..,9}
		{@}
			Length>=6
			{aA,bB,………,zZ}
			{0,1,2,……..,9}
			{@!,#,$,......}
				PLayer/COach

Expected Outcomes	Valid	In-Valid	In-Valid	In-Valid

3.5.1.2 Assign Equipment:

Table 3.9: Valid Assign Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
04	Insert Equipment	equipment_name='Treadmill', equipment_quantity=5,	New Equipment created successfully	Pass

Table 3.10: In-Valid assign Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
05	Insert equipment	equipment_name=Tennis Ball equipment_quantity=,	Error: `equipment_quantity` cannot be null	Fail

Table 3.11: In-Valid Player assign Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
06	Insert equipment	equipment_quantity=2, equipment_name = ‘’	Error: equipment_name cannot be null	Fail

3.5.1.3 Academy

Table 3.12: Valid Academy

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
07	Insert academy data	All required fields provided	Record save successfully	Pass

Table 3.13: Invalid Academy

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
08	Insert academy data	All required fields provided but the coach id can’t be added	Record cannot be save successfully	Fail

3.5.1.4 Attendance:

Table 3.14: Valid Attendance

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
09	Insert attendance record	date = '2024-05-01', start_time = '09:00:00', end_time = '10:00:00', attendance_status = 'present'	Record save successfully	Pass

Table 3.15: Invalid Attendance

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
10	Insert attendance record	date = '2024-05-01', start_time = NULL, end_time = NULL, attendance_status = 'present',	Record cannot be save because time is not null	Fail

3.5.1.5 Checkout Form:

Table 3.16: Valid Check-out form

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
11	Insert booking details with all required fields	player_name = 'John', player_email = hateem@gmail.com', player_phone_number = 9998887777, player_address = 'NYC', coach_name = 'Umer', start_time = '10:00:00', end_time = '11:00:00', from_date = '2024-06-01', to_date = '2024-06-01', payment_type = 'stripe'	Data Save Successfully	Pass

Table 3.17: Invalid Post

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
12	Insert booking details with required fields	player_name = 'John', player_email = 'john@gmail.com', player_phone_number = 9998887777, player_address = 'NYC', coach_name = 'Coach A', start_time = '10:00:00', end_time = '11:00:00', from_date = '2024-06-01', to_date = '2024-06-01', payment_type = ' '	Error: Payment_type cannot be null	Fail

3.5.1.6 Coaches

Table 3.18: Valid coaches

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
13	Insert coach profile with required fields (name, category_id, experience, level, phone_number, certificate, image,	name = 'John Doe', category_id = 1, experience = '5 years', level = 'Intermediate', phone_number = '9876543210', coach_location = 'Islamabad' per_hour_charges = 1000 certificate = 'cert.pdf', image = 'img.jpg',	Data can be saved success fully	Pass

Table 3.19: Invalid coaches

Test case ID	Function to Be tested	Input	Expected Output	Test Case Status
14	Insert coach profile with required fields (name, category_id, experience, level, phone_number, etc.	name = ' ', category_id = 1, experience = '5 years', level = 'Intermediate', phone_number = '9876543210', coach_location = 'Islamabad' per_hour_charges = 1000 certificate = 'cert.pdf', image = 'img.jpg',	Data cannot be save successfully b/c the name is not null	Fail

3.5.1.7 Coach Schedule:

Table 3.20: Valid Coach schedule

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
15	Insert valid booking event with all required fields	start_time = '10:00:00', end_time = '11:00:00', from_date = '2024-06-01', to_date = '2024-06-01', playwith = 'individual', event_name = 'Morning Session'	Data save successfully	Pass

Table 3.21: In-Valid Coach Schedule

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
16	Insert valid booking event with all required fields	start_time = '10:00:00', end_time = '11:00:00', from_date = '2024-06-01', to_date = '2024-06-01', booking_count = 1, playwith = ' ', event_name = 'Morning Session'	Data cannot be saved successfully b/c playwith not be null	Fail

3.5.1.8 Feedback Form:

Table 3.23: Valid Feedback form

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
17	Insert feedback entry with all required fields	name = 'John Doe', email = 'johndoe@gmail.com', message = 'Great service!',	Data save successfully	Pass

Table 3.24: In-Valid Sports Category

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
18	Insert feedback entry with all required fields	name = 'Jane Doe', email = 'janedoe@gmail.com',	Error: message is required	Fail

3.5.1.9 Payment:

Table 3.25: Valid payment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
19	Insert Payment Record	email = 'test@gmail.com', payment_id = 'pay123', amount = 100.50	Payment record insert successfully	Pass

Table 3.26: In-Valid payment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
20	Insert Payment Record	email = 'haateem@gmail.com', payment_id = 'pay123', amount =	Error amount cannot be null	Fail

3.5.1.10 Player:

Table 3.27: Valid player

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
21	Insert Player Record	player_name = 'John Doe', cat_id = 1, playwith = individual, player_gender = 'Male', player_phonenumber = '1234567890', player_dob = '1990-01-01' player_image =image.png, player Address = Islamabad	Data can successfully add on the database	Pass

Table 3.28: In-Valid Map

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
22	Insert Player Record	player_name = 'John Doe', cat_id = 1, playwith = individual, player_gender = 'Male', player_phonenumber = '1234567890', player_dob = '1990-01-01' player_image =, player Address =	Error: data cannot be sent b/c player image and player address cannot be null	Fail

3.5.1.11 Player Parent:

Table 3.29: Valid player parent

Test case ID	Function to Be tested	Input	Expected Output	Test Case Status
23	Insert parent Record	cnic = '1234567890', name = 'Hateem', email = ali@gmail.com', address = '123 Main St', phone_number = '1234567890', location = 'New York'	Data can save successfully	Pass

Table 3.30: In-Valid player parent

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
24	Insert parent Record	cnic = '1234567890', name = 'Hateem', email = ali&gmail.com', address = '123 Main St', phone_number = '1234567890', location = 'New York'	Please include an '@' in the email address. 'abcgmail.com' is missing an '@'.	Fail

3.5.1.12 Player Score:

Table 3.31: Valid player score

Test case ID	Function to Be tested	Input	Expected Output	Test Case Status
25	Insert score	date = '2025-05-01', player_type = 'Batsman', played_over = 10, today_give_wickets = 2, through_over = 8, today_taken_wickets = 3,	Data save successfully	Pass

Table 3.32: In-Valid player score

Test case ID	Function to Be tested	Input	Expected Output	Test Case Status
26	Insert score	date = '2025-05-01', player_type = ‘’, played_over = 10, today_give_wickets = 2, through_over = 8, today_taken_wickets = 3,	Player type cannot be null	Fail

3.5.1.13 Post:

Table 3.33: Valid Post

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
27	Add post	post_title = 'New Post', post_name = 'new-post', post_description = 'This is a new post.', post_image = 'image.jpg', post_location = Islamabad,	Post added successfully	Pass

Table 3.34: In-Valid Map

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
28	Add post	post_title = '', post_name = 'new-post', post_description = 'This is a new post.', post_image = 'image.jpg', post_location = Islamabad,	Error post title cannot be null	Fail

3.5.1.14 Rating:

Table 3.35: Valid rating

Test case ID	**Function to Be tested**	**Input**	**Expected Output**	**Test Case Status**
29	Add rating	rating = 5, reviews = 'Excellent coach!'	Rating can add successfully	Pass

Table 3.36: In-Valid rating

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
30	Add rating	rating = , reviews = 'Excellent coach!'	Error plz add the rating	Fail

3.5.1.15 Request Equipment:

Table 3.37: Valid Request Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status

31	Request for equipment	equipment_name_id = 1,
		equipment_quantity = 10
		return_time = 12-5 2025
			Record inserted successfully.	Pass

Table 3.38: In-Valid Request Equipment

Test case ID	Function to Be tested	Input	Expected Output	Test Case Status

32	Request for equipment	equipment_name_id = 1,
		equipment_quantity = 10
		return_time =
			Error plz add the return time	Fail

3.5.1.16 Return Equipment:

Table 3.39: Valid Return Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
33	Return Equipment’s	equipment_name = hocky, quantity = 5, return_date_time = '2025-05-15 10:00:00', return_note = 'Returned in good condition'!'	Successful update	pass

Table 3.40: In-Valid Return Equipment

Test case ID	Function To Be tested	Input	Expected Output	Test Case Status
34	Return Equipment’s	equipment_name = hocky, quantity = 150, return_date_time = '2025-05-15 10:00:00', return_note = 'Returned in good condition'!'	Error: there was an error that the return quantity cannot be exceed to the assign quantity	Fail

3.5.1.17 Sport Category:

Table 3.41: Valid Sport Category

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
35	Add Sports	name = 'cricket',	Data can save successfully	Pass

Table 3.42: In-Valid sport Category

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
36	Add sports	name = ‘’,	Error plz add the name	Fail

3.5.1.18 Videos:

Table 3.43: Valid videos

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**

43	Add video	Video_title = session
		Video = video.mp4
			Successfully add video	pass

Table 3.44: In-Valid videos

Test case ID	**Function To Be tested**	**Input**	**Expected Output**	**Test Case Status**
43	Add video	Video_title = session Video =	Error plz add the video file as well	Fail

3.6. Summary

As shown above, we have written all the functional requirements using techniques like interviewing and brainstorming. Moreover, we don't have Non – Functional requirements of our system. while non-functional considerations such as performance, security, and usability were implicit in the discussions, they were not explicitly documented.

Chapter 4:

System Design

Chapter:04
System Design

4.1 Introduction

Design is a significant engineering representation of something that is intended to be constructed. It is derived from the customer's requirements. Without proper design, there is a risk of developing an unstable system that may fail when even minor changes are made. It could also be challenging to test such a system, and its quality may only be assessed late in the software development process, when time is limited, and a considerable amount of money has already been invested. In this chapter, we will explore the importance of design, the paradigm employed in this project, and the architecture chosen for the development of our marketing project.

4.2 Model View Controller Diagram (MVC)

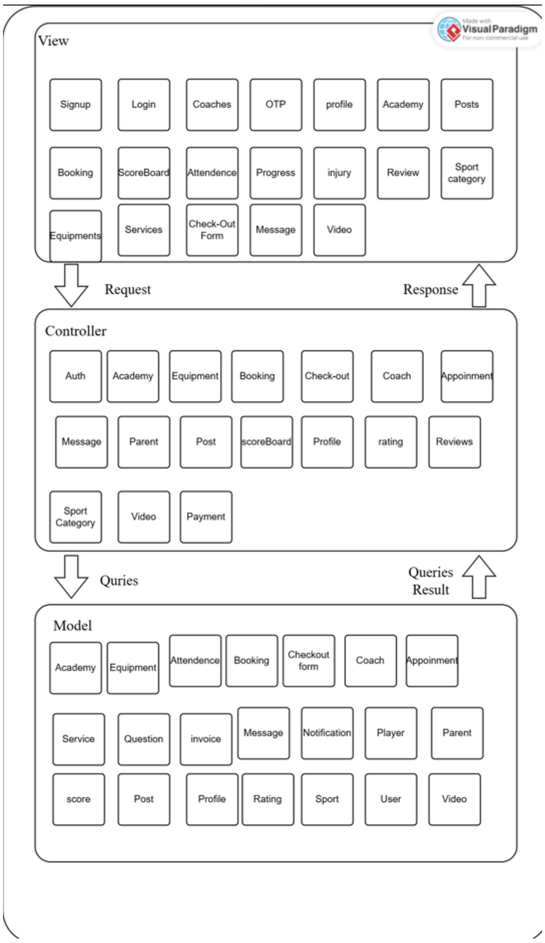


Figure 4.1: MVC Diagram

4.3 Database Design

4.3.1: Entity Relationship Diagram (ERD):

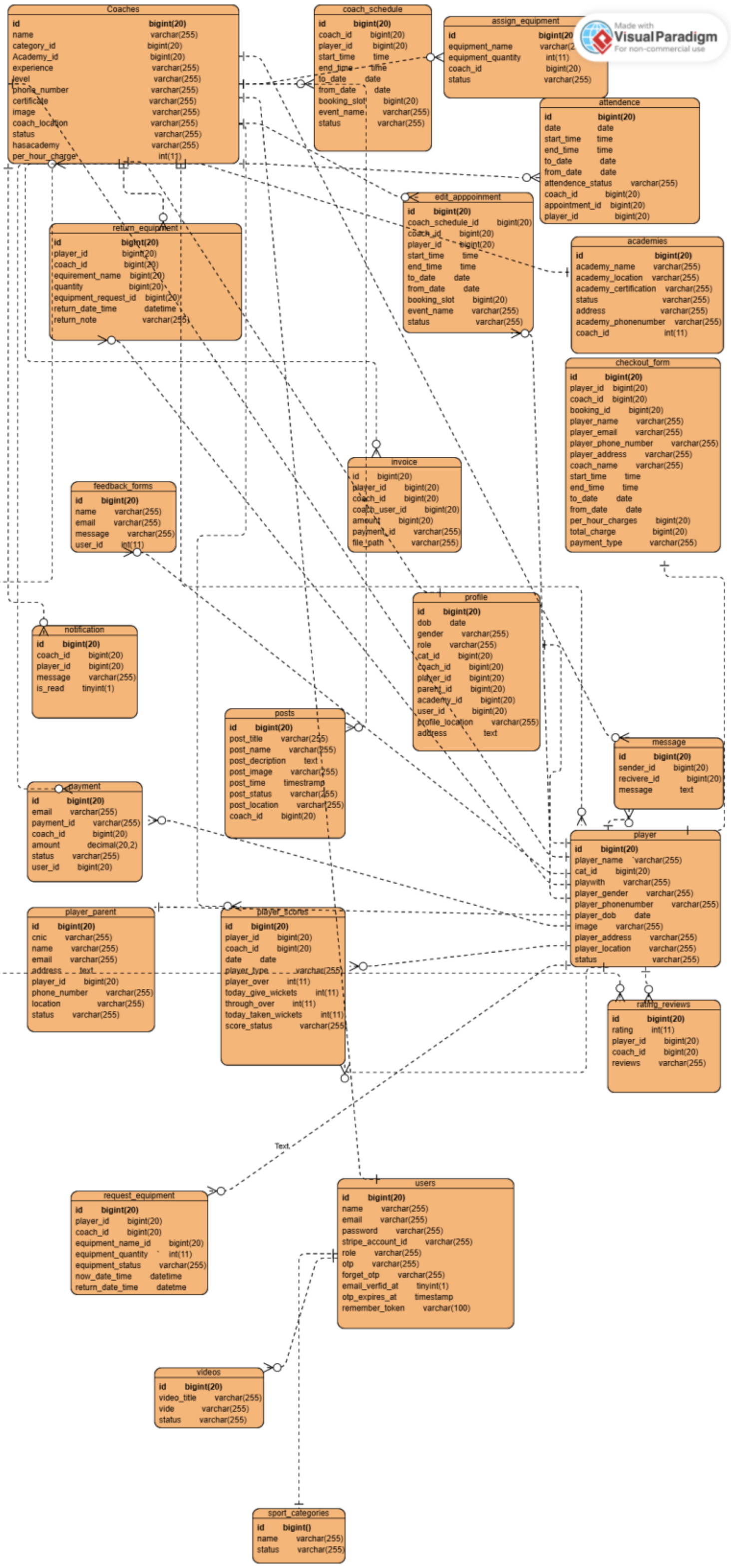


Fig 4.2 | Database ERD

4.4 Activity Diagrams:

4.4.1 Admin

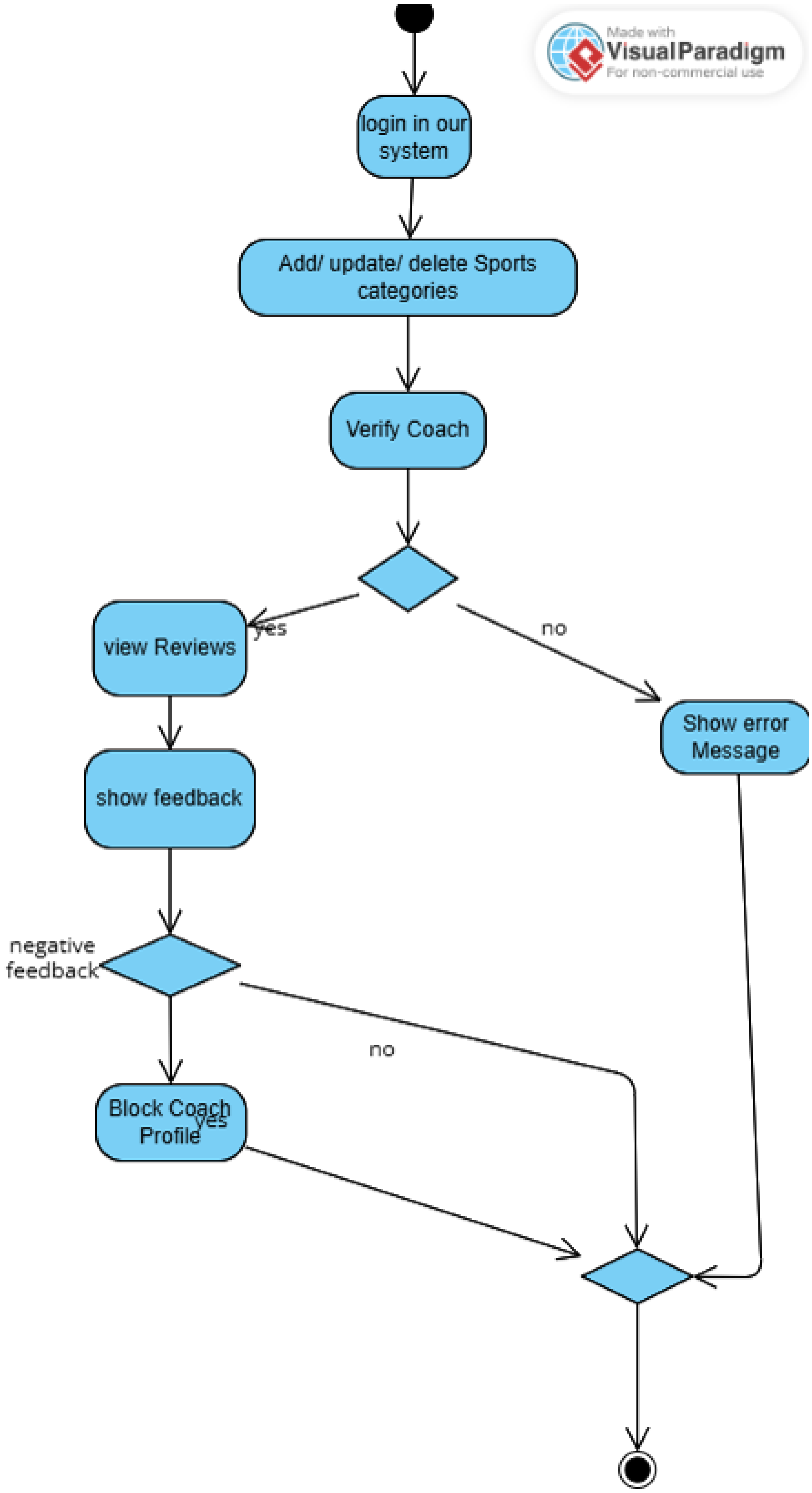


Figure 4.3: Admin Activity Diagram

4.4.2 Coach

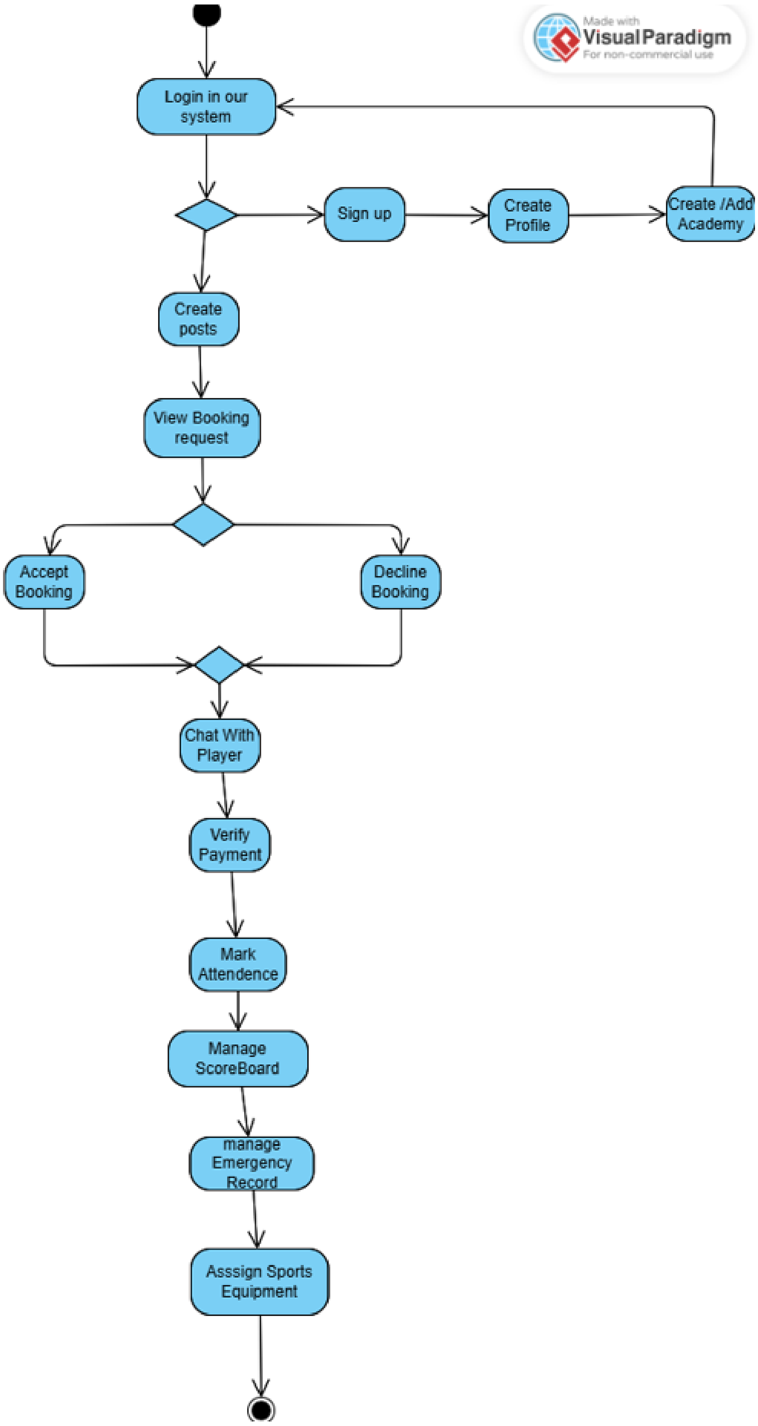


Figure 4.4: Coach Activity Diagram

4.4.3 Player

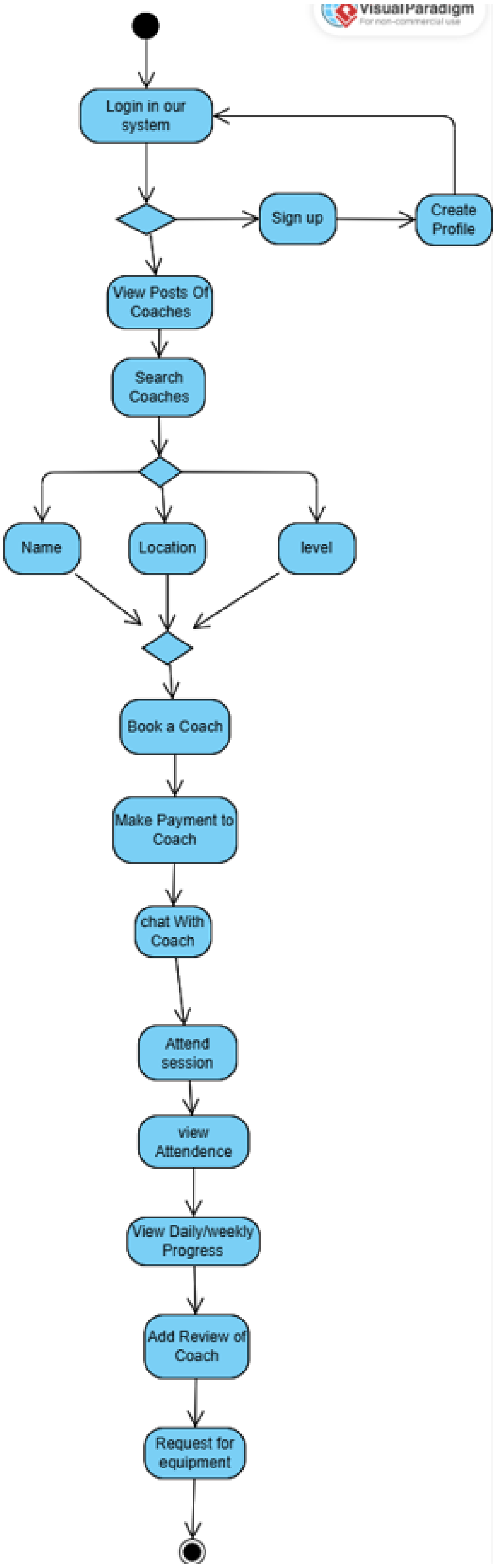


Figure 4.5: Player Activity Diagram

4.4.4 Parent

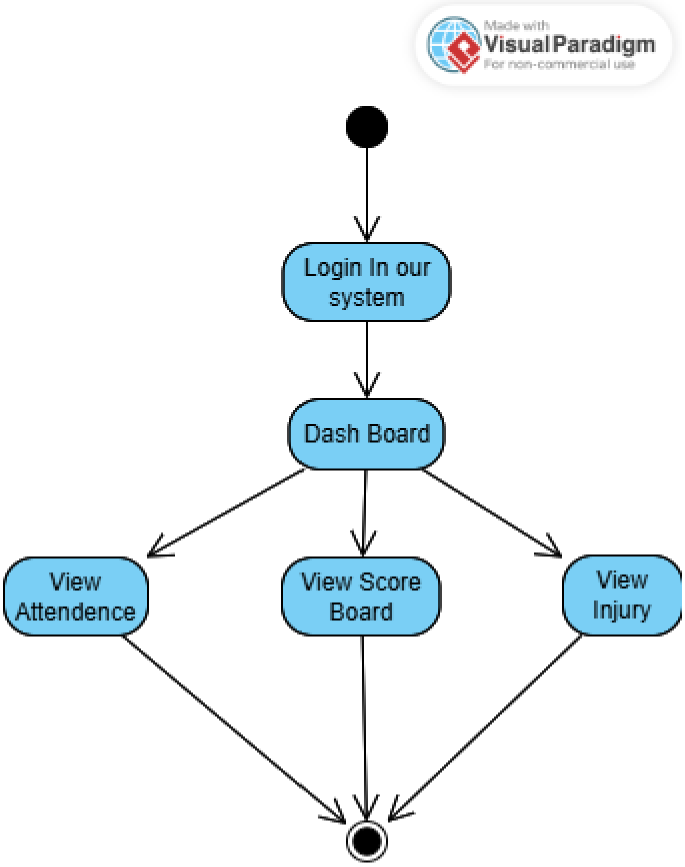


Figure 4.6: Parent Activity Diagram

4.5 Use Case Diagram

4.5.1 Admin:

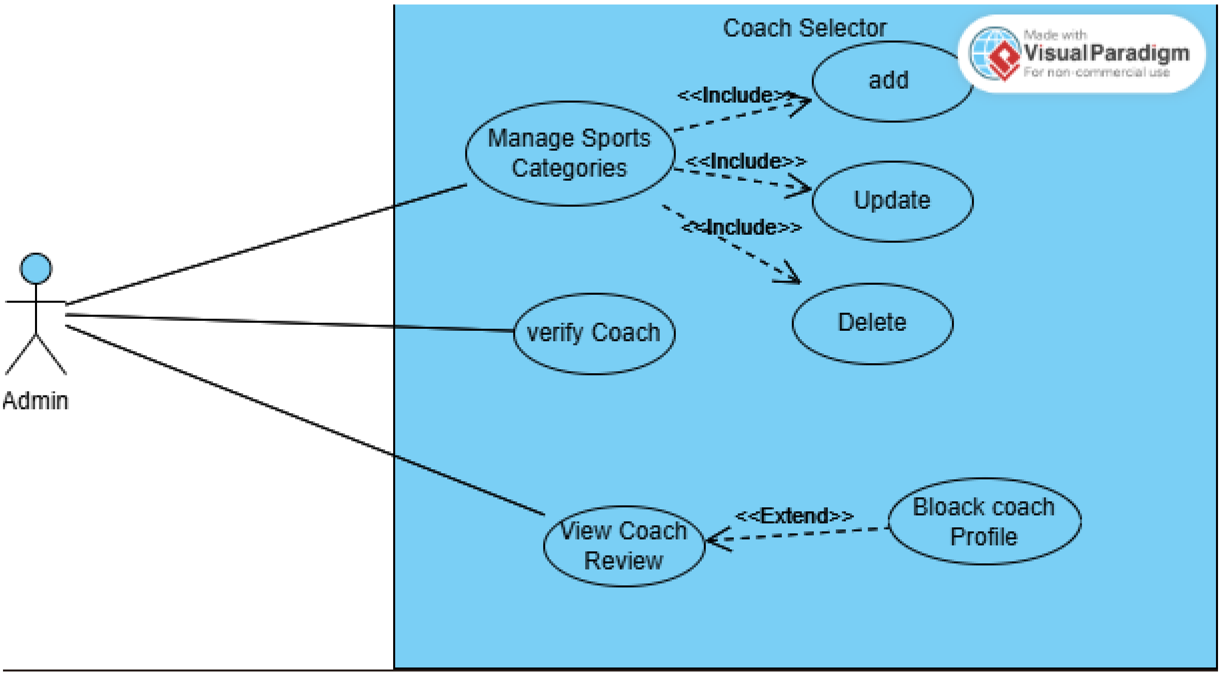


Figure 4.7: Admin Use Case Diagram

4.5.2 Coach:

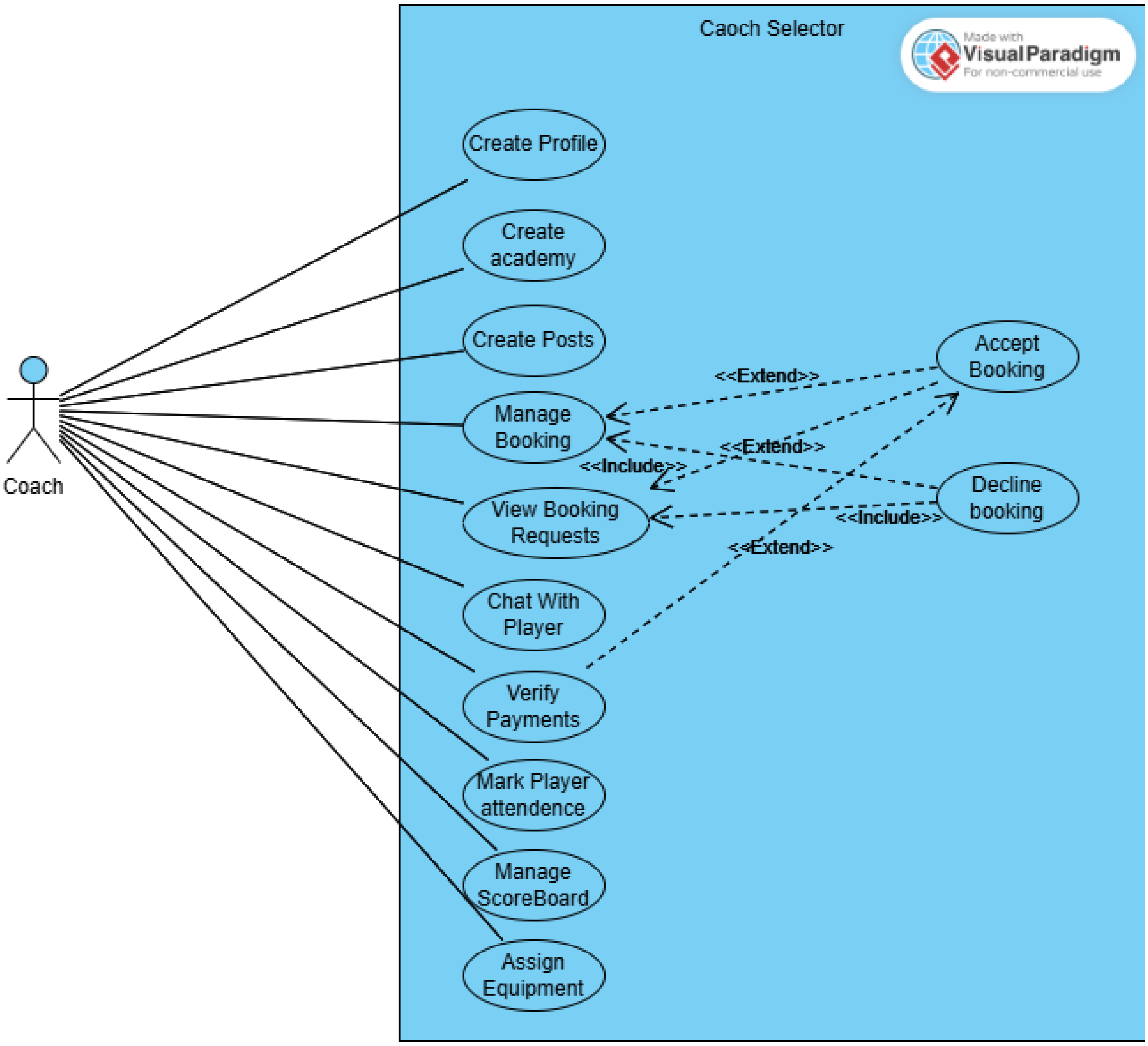


Figure 4.8: Coach Use Case Diagram

4.5.3 Player:

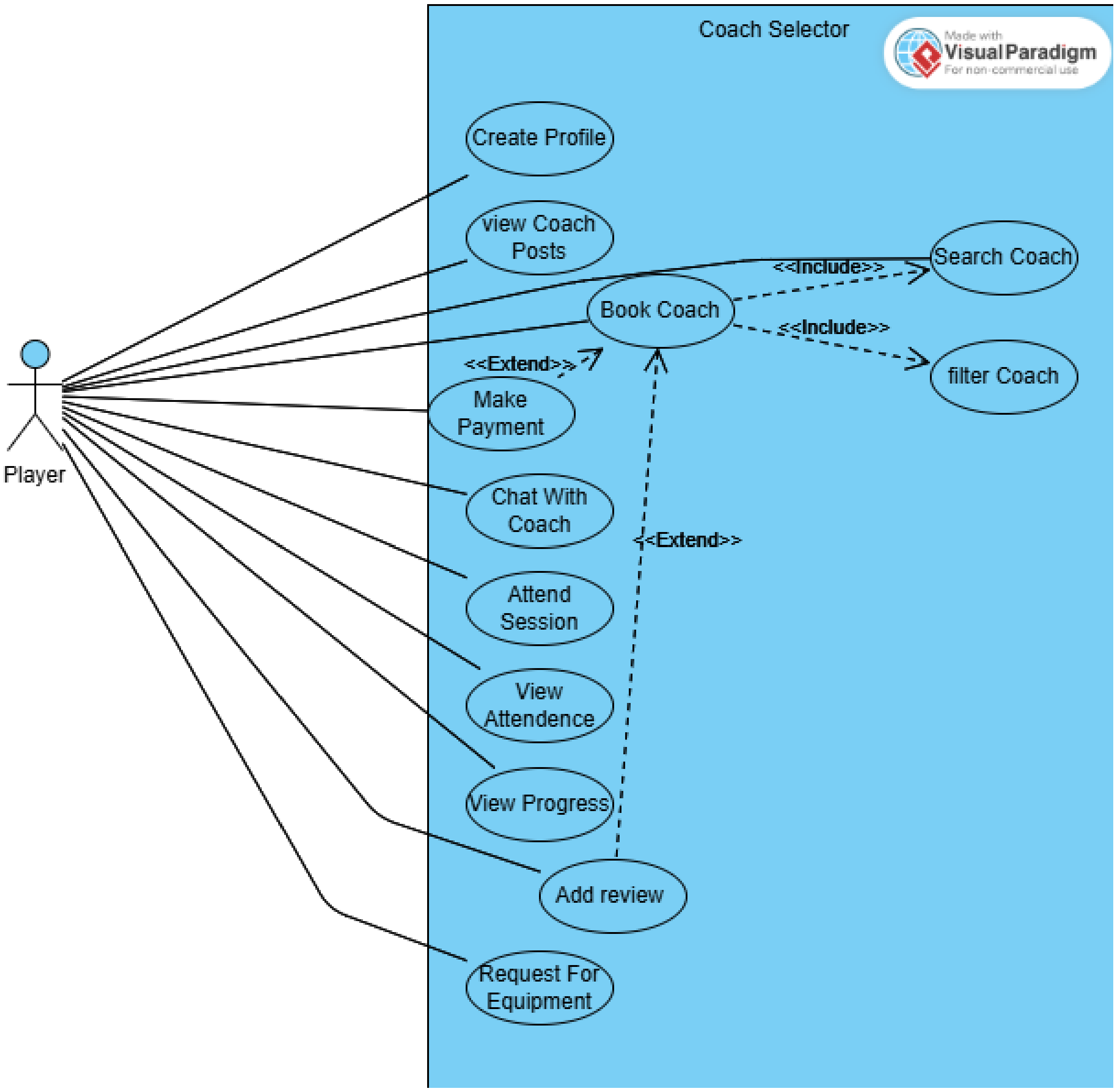


Figure 4.9: Player Use Case Diagram

4.5.4 Parent:

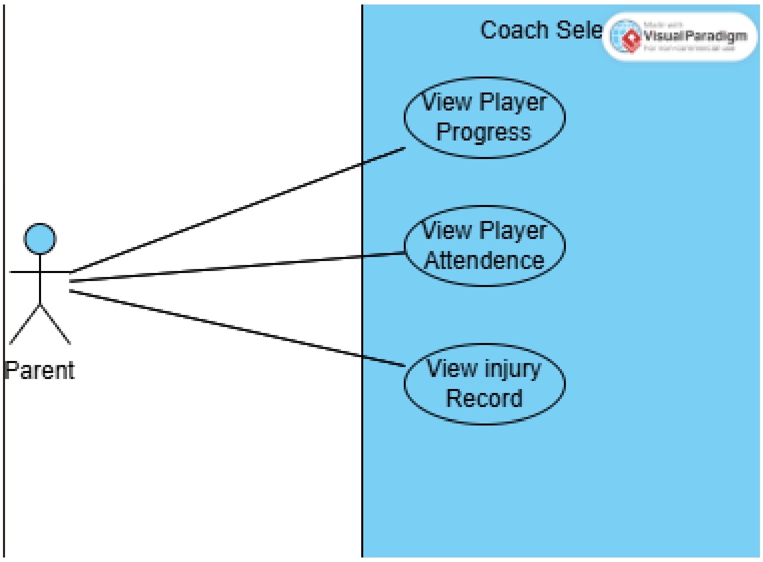


Figure 4.10: Parent Use Case Diagram

4.5.5: FULLY USE-CASE DIAGRAM:

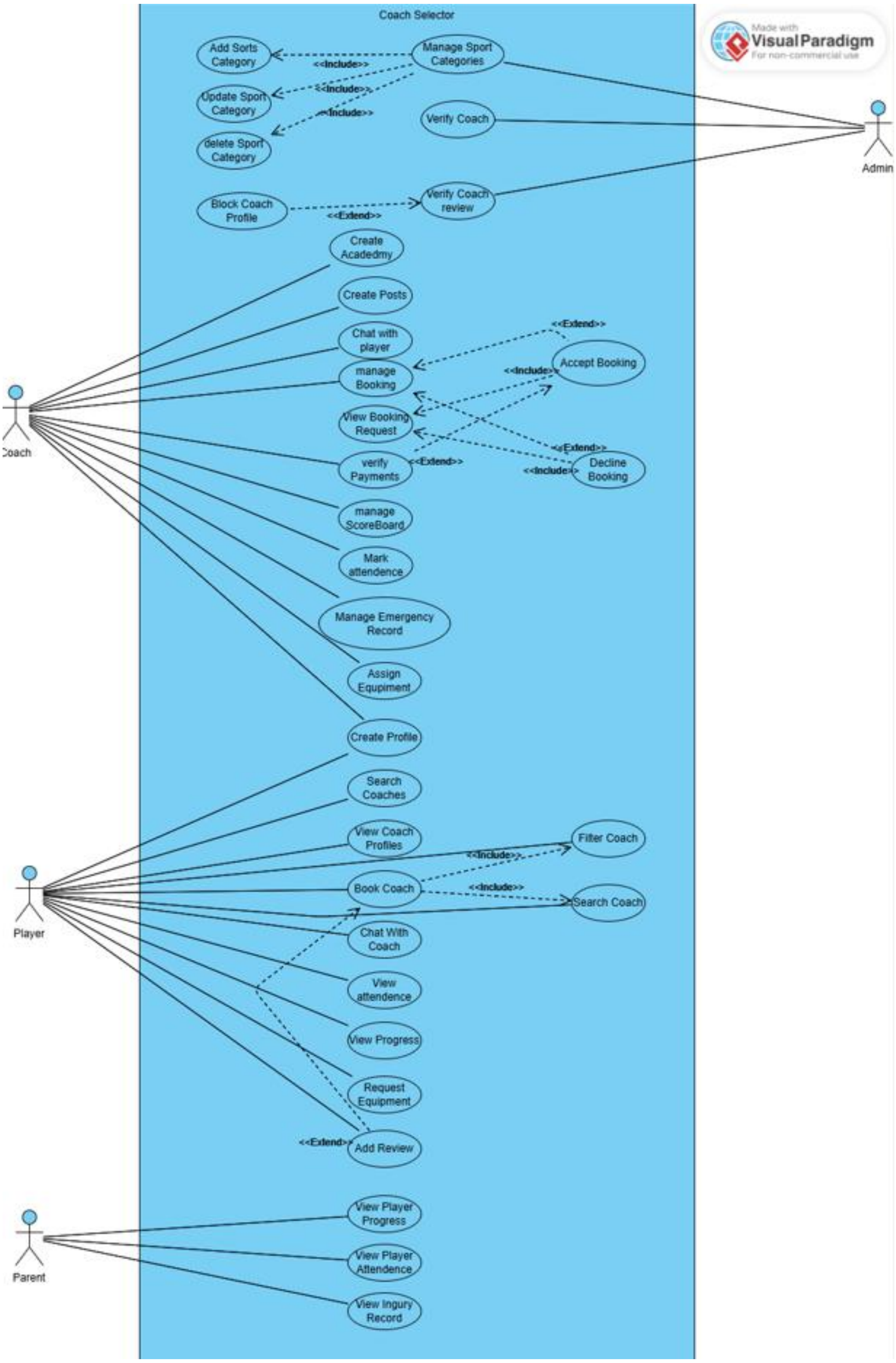
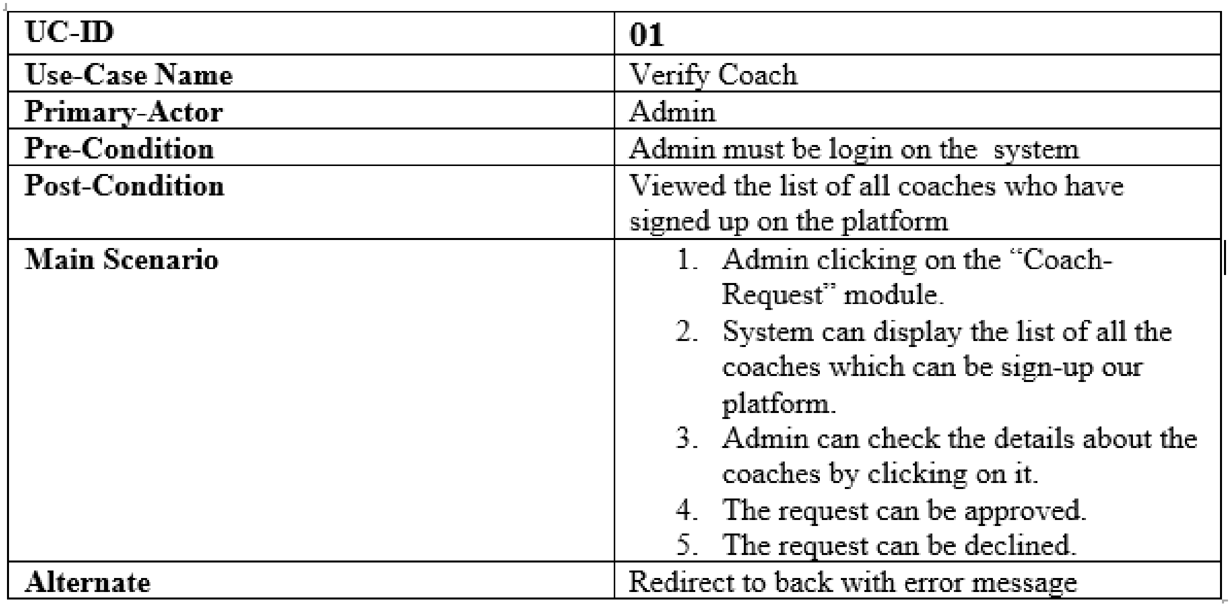


Figure 4.11: Full Use Case Diagram

4.6 Fully Address Use Cases

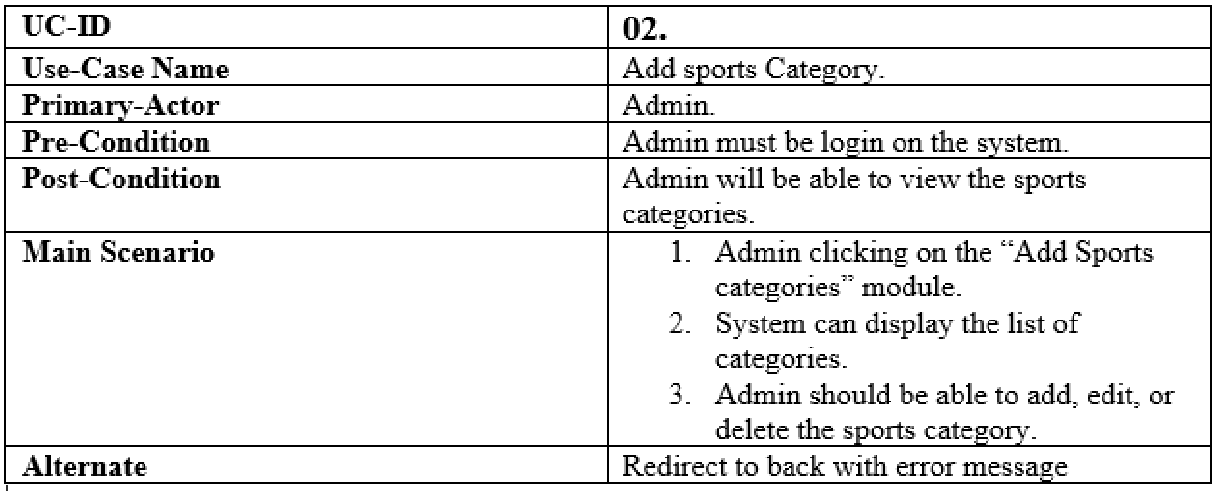
4.6.1 Admin

Table 4.1: Verify Coaches



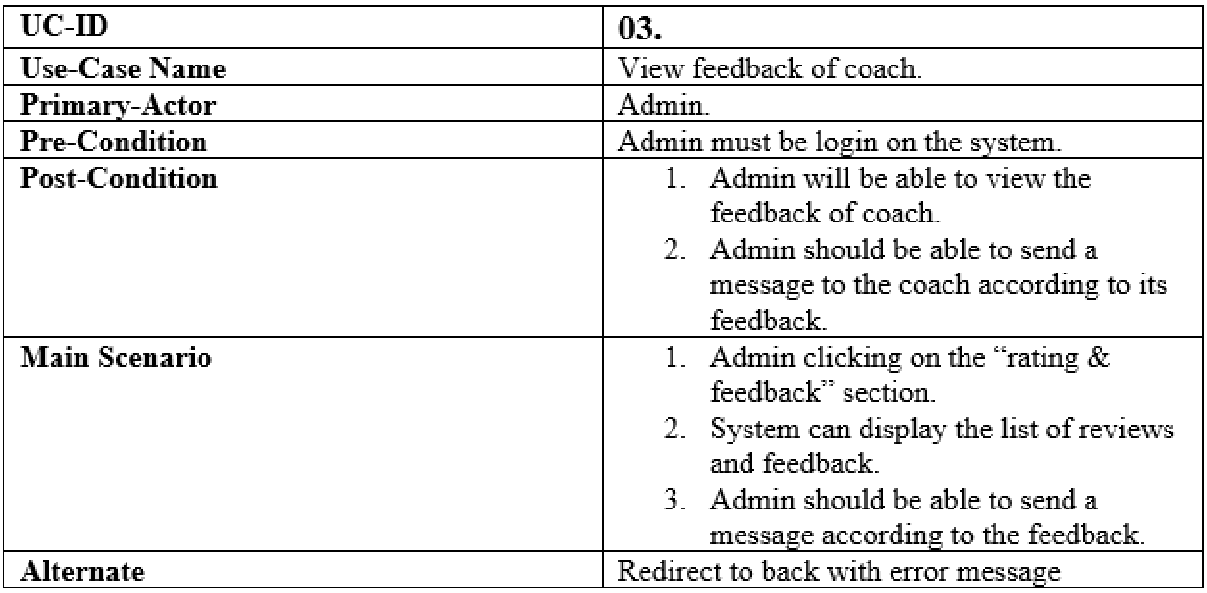
UC-ID	**01**
Use-Case Name	Verify Coach
Primary-Actor	Admin
Pre-Condition	Admin must be login on the system
Post-Condition	Viewed the list of all coaches who have signed up on the platform
Main Scenario	1. Admin clicking on the "Coach-Request" module. 2. System can display the list of all the coaches which can be sign-up our platform. 3. Admin can check the details about the coaches by clicking on it. 4. The request can be approved. 5. The request can be declined.
Alternate	Redirect to back with error message

Table 4.2: Add Sports Category



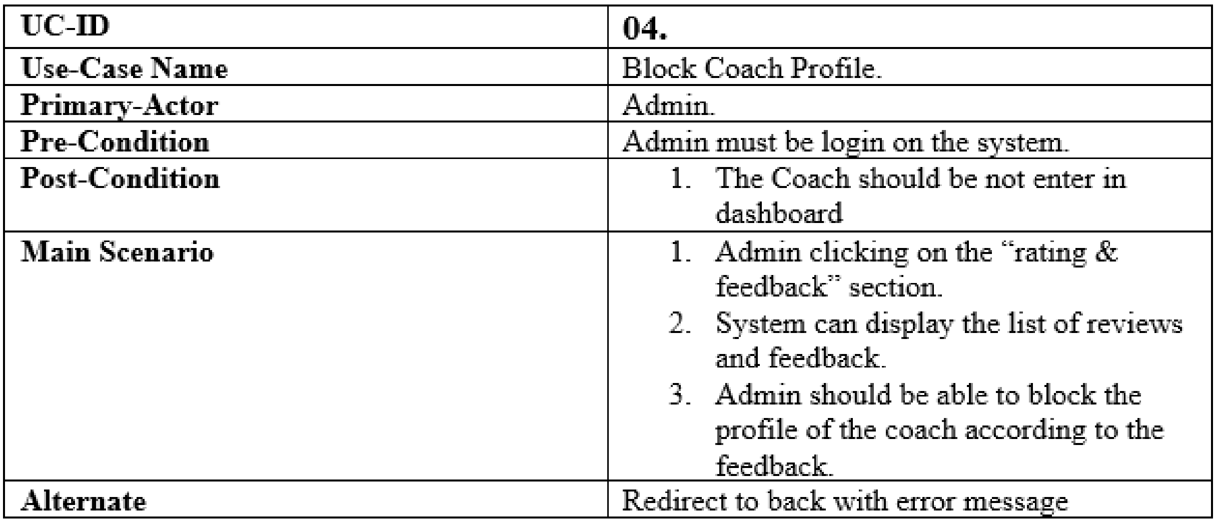
UC-ID	**02.**
Use-Case Name	Add sports Category.
Primary-Actor	Admin.
Pre-Condition	Admin must be login on the system.
Post-Condition	Admin will be able to view the sports categories.
Main Scenario	1. Admin clicking on the "Add Sports categories" module. 2. System can display the list of categories. 3. Admin should be able to add, edit, or delete the sports category.
Alternate	Redirect to back with error message

Table 4.3: View Feedback of Coach



UC-ID	**03.**
Use-Case Name	View feedback of coach.
Primary-Actor	Admin.
Pre-Condition	Admin must be login on the system.
Post-Condition	1. Admin will be able to view the feedback of coach. 2. Admin should be able to send a message to the coach according to its feedback.
Main Scenario	1. Admin clicking on the "rating & feedback" section. 2. System can display the list of reviews and feedback. 3. Admin should be able to send a message according to the feedback.
Alternate	Redirect to back with error message

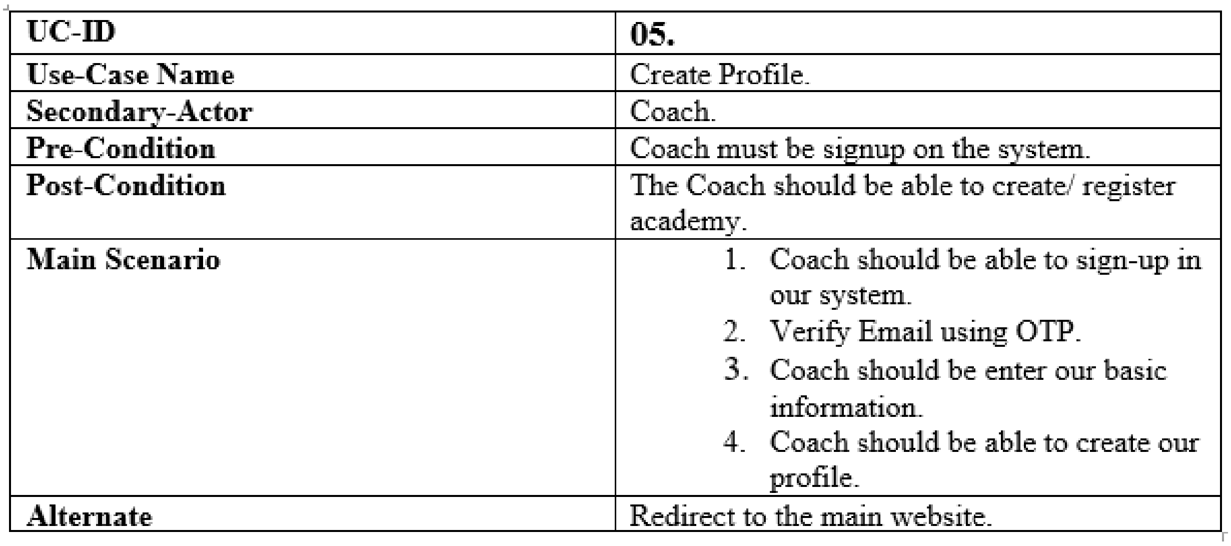
Table 4.4: Block Coach Profile



UC-ID	**04.**
Use-Case Name	Block Coach Profile.
Primary-Actor	Admin.
Pre-Condition	Admin must be login on the system.
Post-Condition	1. The Coach should be not enter in dashboard
Main Scenario	1. Admin clicking on the "rating & feedback" section. 2. System can display the list of reviews and feedback. 3. Admin should be able to block the profile of the coach according to the feedback.
Alternate	Redirect to back with error message

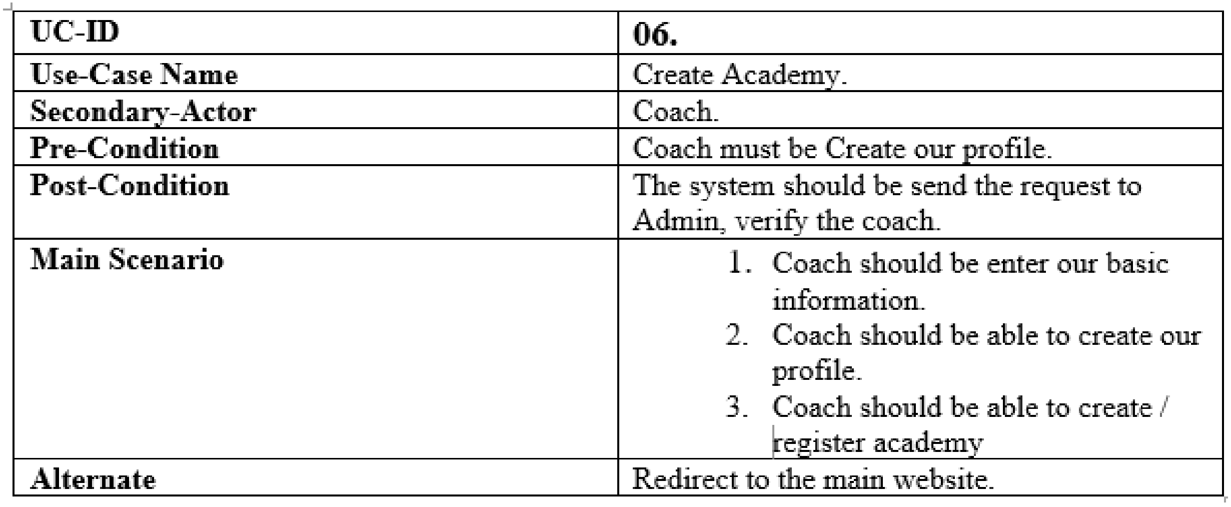
4.6.2 Coach

Table 4.5: Create Profile



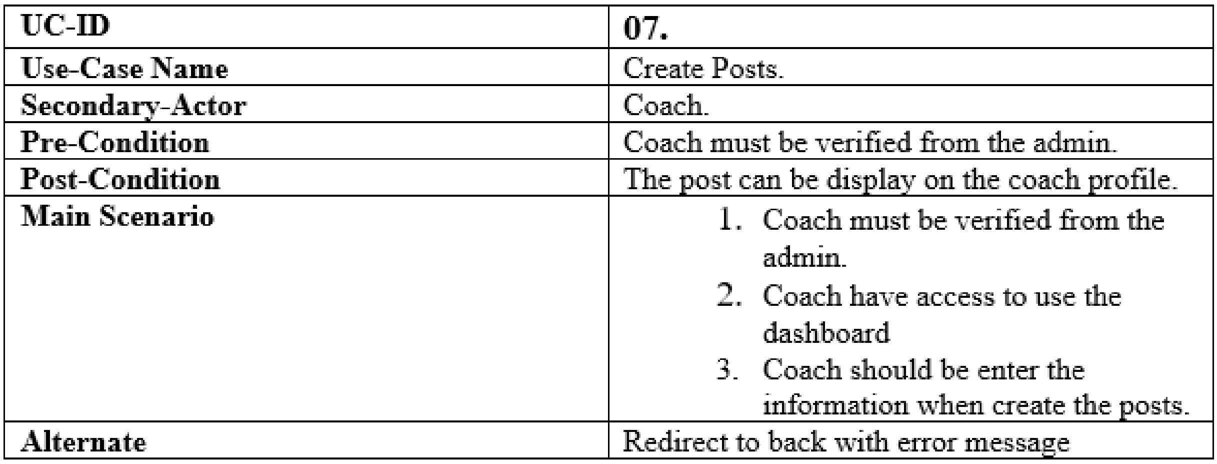
UC-ID	**05.**
Use-Case Name	Create Profile.
Secondary-Actor	Coach.
Pre-Condition	Coach must be signup on the system.
Post-Condition	The Coach should be able to create/ register academy.
Main Scenario	1. Coach should be able to sign-up in our system. 2. Verify Email using OTP. 3. Coach should be enter our basic information. 4. Coach should be able to create our profile.
Alternate	Redirect to the main website.

Table 4.6: Create Academy



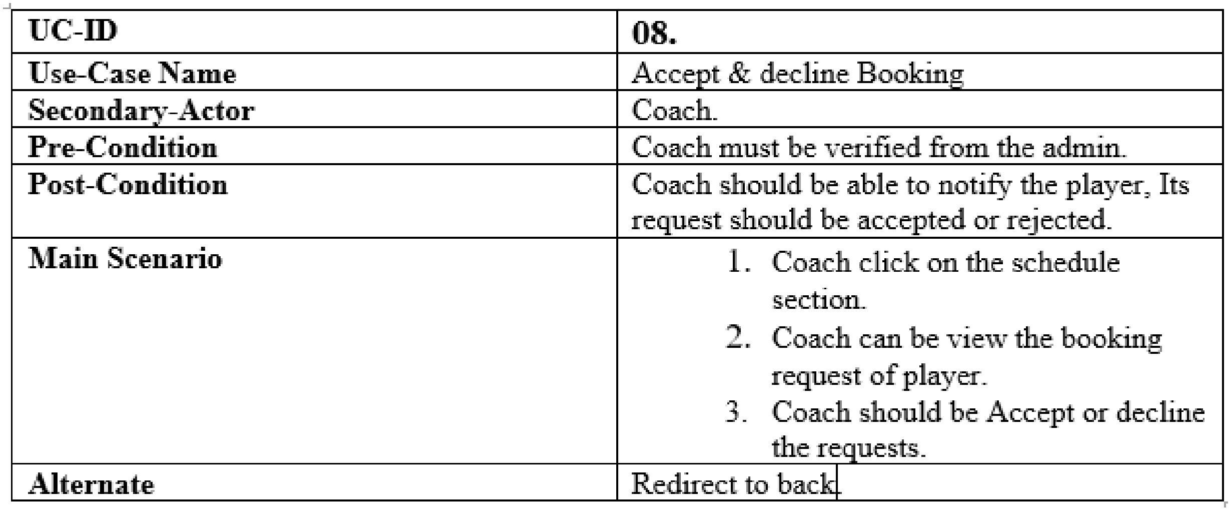
UC-ID	**06.**
Use-Case Name	Create Academy.
Secondary-Actor	Coach.
Pre-Condition	Coach must be Create our profile.
Post-Condition	The system should be send the request to Admin, verify the coach.
Main Scenario	1. Coach should be enter our basic information. 2. Coach should be able to create our profile. 3. Coach should be able to create / register academy
Alternate	Redirect to the main website.

Table 4.6.7: Create Post



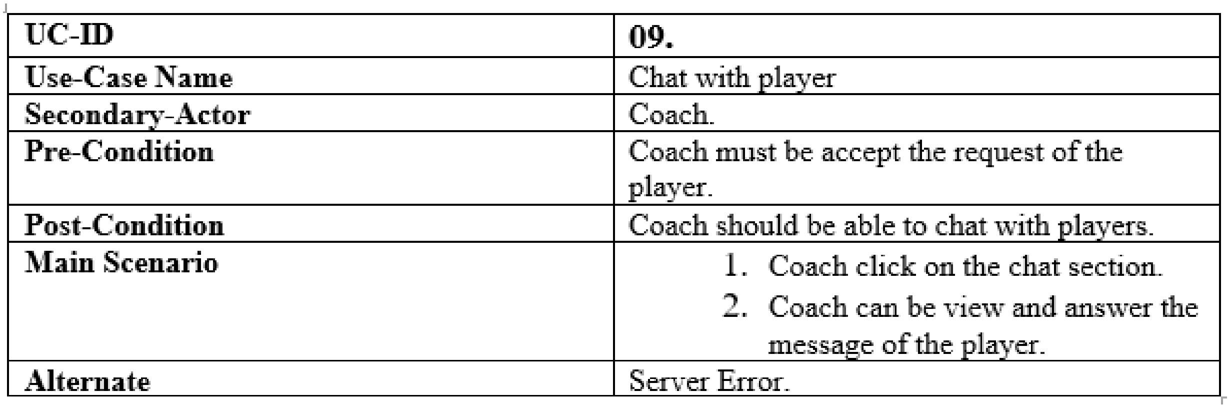
UC-ID	**07.**
Use-Case Name	Create Posts.
Secondary-Actor	Coach.
Pre-Condition	Coach must be verified from the admin.
Post-Condition	The post can be display on the coach profile.
Main Scenario	1. Coach must be verified from the admin. 2. Coach have access to use the dashboard 3. Coach should be enter the information when create the posts.
Alternate	Redirect to back with error message

Table 4.8: Accept & Decline Booking



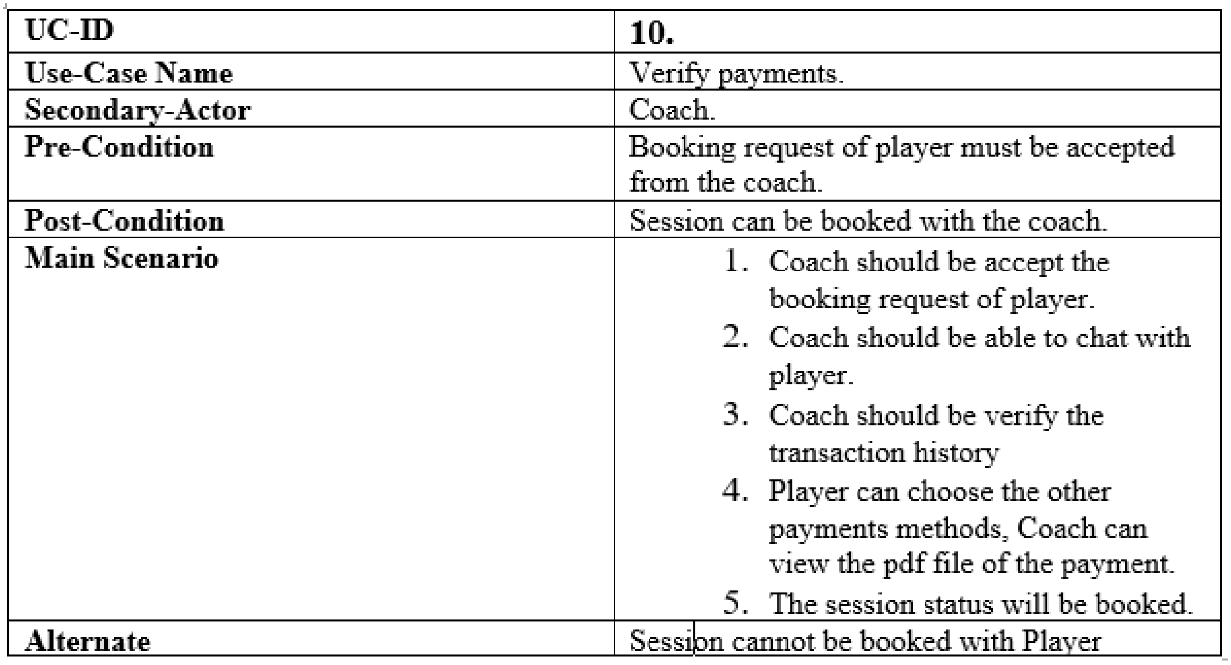
UC-ID	**08.**
Use-Case Name	Accept & decline Booking
Secondary-Actor	Coach.
Pre-Condition	Coach must be verified from the admin.
Post-Condition	Coach should be able to notify the player, Its request should be accepted or rejected.
Main Scenario	1. Coach click on the schedule section. 2. Coach can be view the booking request of player. 3. Coach should be Accept or decline the requests.
Alternate	Redirect to back.

Table 4.9: Chat with Player



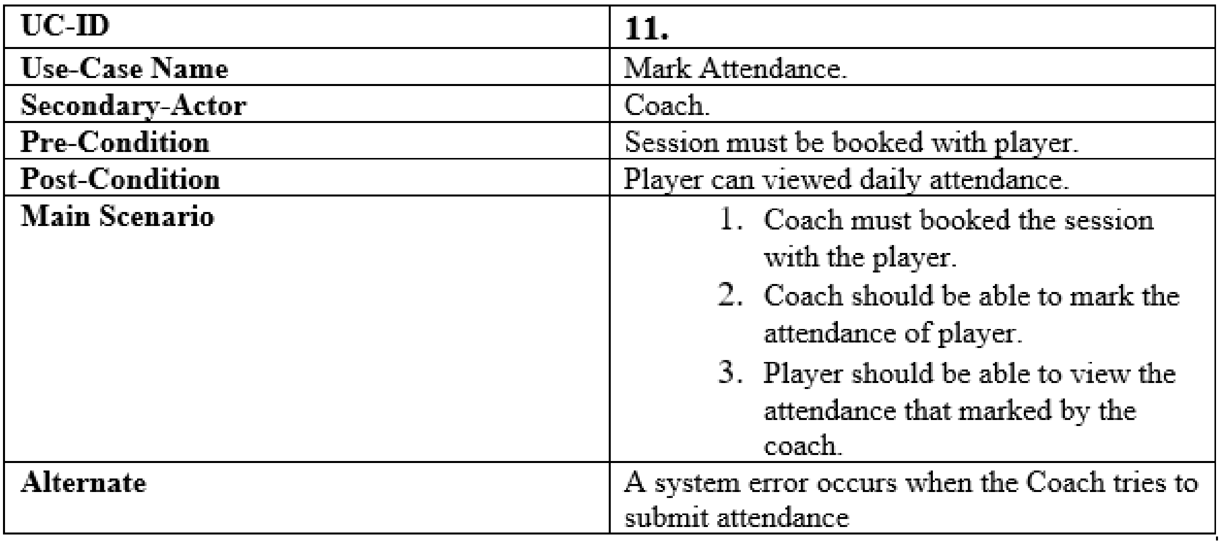
UC-ID	**09.**
Use-Case Name	Chat with player
Secondary-Actor	Coach.
Pre-Condition	Coach must be accept the request of the player.
Post-Condition	Coach should be able to chat with players.
Main Scenario	1. Coach click on the chat section. 2. Coach can be view and answer the message of the player.
Alternate	Server Error.

Table 4.10: Verify Payments



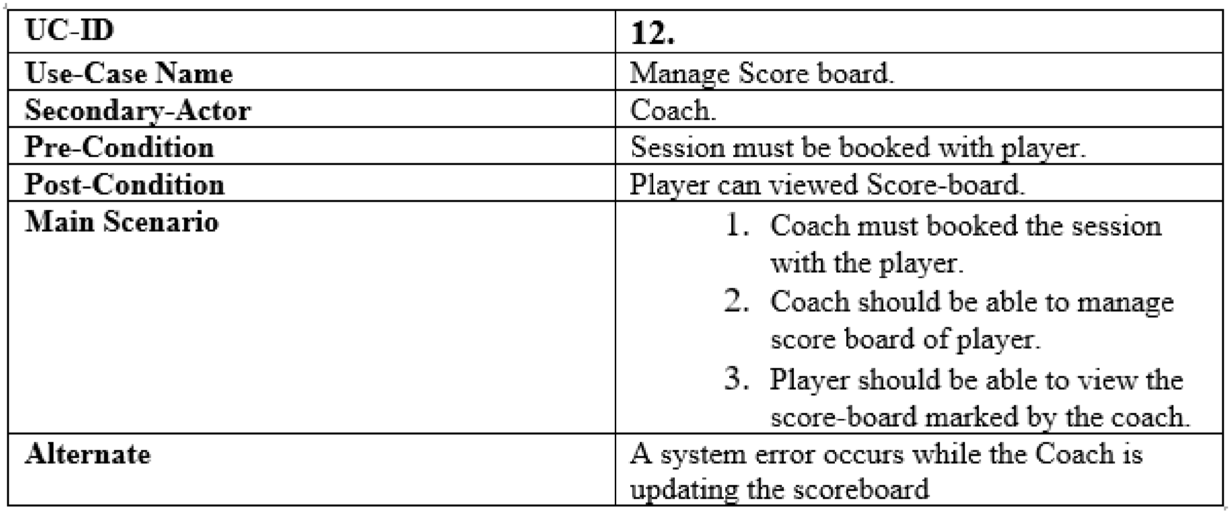
UC-ID	**10.**
Use-Case Name	Verify payments.
Secondary-Actor	Coach.
Pre-Condition	Booking request of player must be accepted from the coach.
Post-Condition	Session can be booked with the coach.
Main Scenario	1. Coach should be accept the booking request of player. 2. Coach should be able to chat with player. 3. Coach should be verify the transaction history 4. Player can choose the other payments methods, Coach can view the pdf file of the payment. 5. The session status will be booked.
Alternate	Session cannot be booked with Player

Table 4.11: Mark Attendance



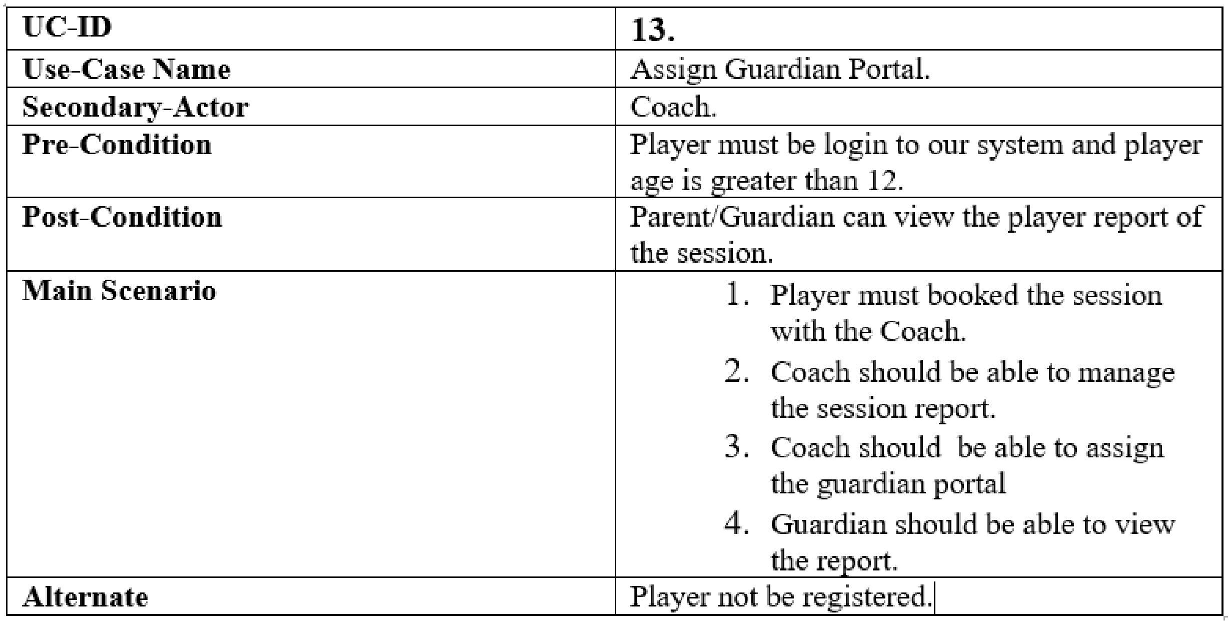
UC-ID	**11.**
Use-Case Name	Mark Attendance.
Secondary-Actor	Coach.
Pre-Condition	Session must be booked with player.
Post-Condition	Player can viewed daily attendance.
Main Scenario	1. Coach must booked the session with the player. 2. Coach should be able to mark the attendance of player. 3. Player should be able to view the attendance that marked by the coach.
Alternate	A system error occurs when the Coach tries to submit attendance

Table 4.12: Manage Score-Board



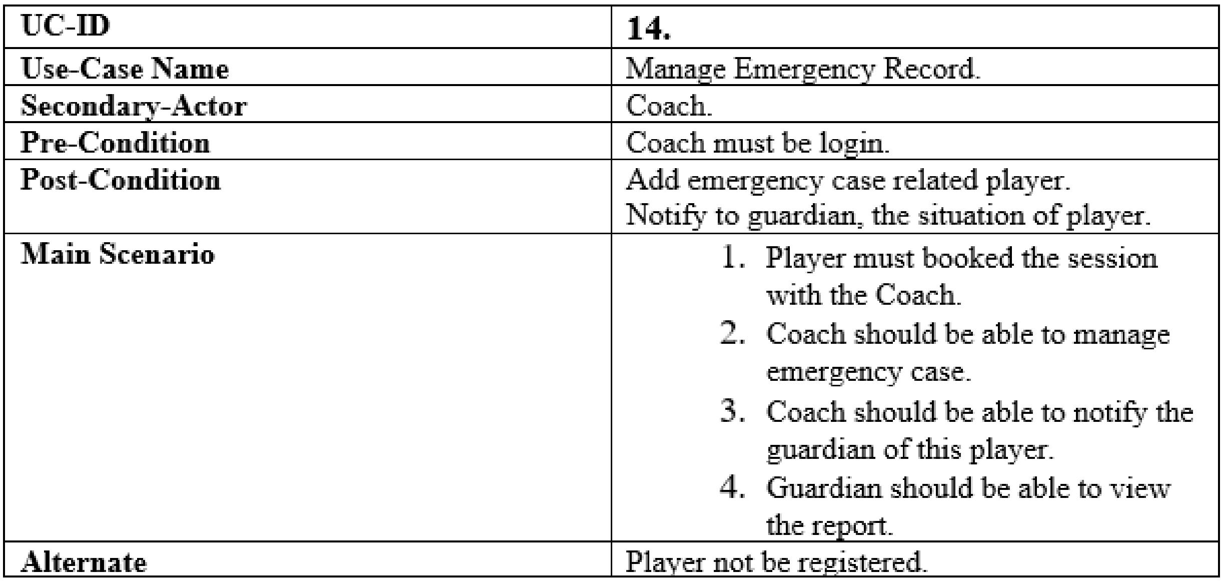
UC-ID	**12.**
Use-Case Name	Manage Score board.
Secondary-Actor	Coach.
Pre-Condition	Session must be booked with player.
Post-Condition	Player can viewed Score-board.
Main Scenario	1. Coach must booked the session with the player. 2. Coach should be able to manage score board of player. 3. Player should be able to view the score-board marked by the coach.
Alternate	A system error occurs while the Coach is updating the scoreboard

Table 4.13: Assign Guardian Portal



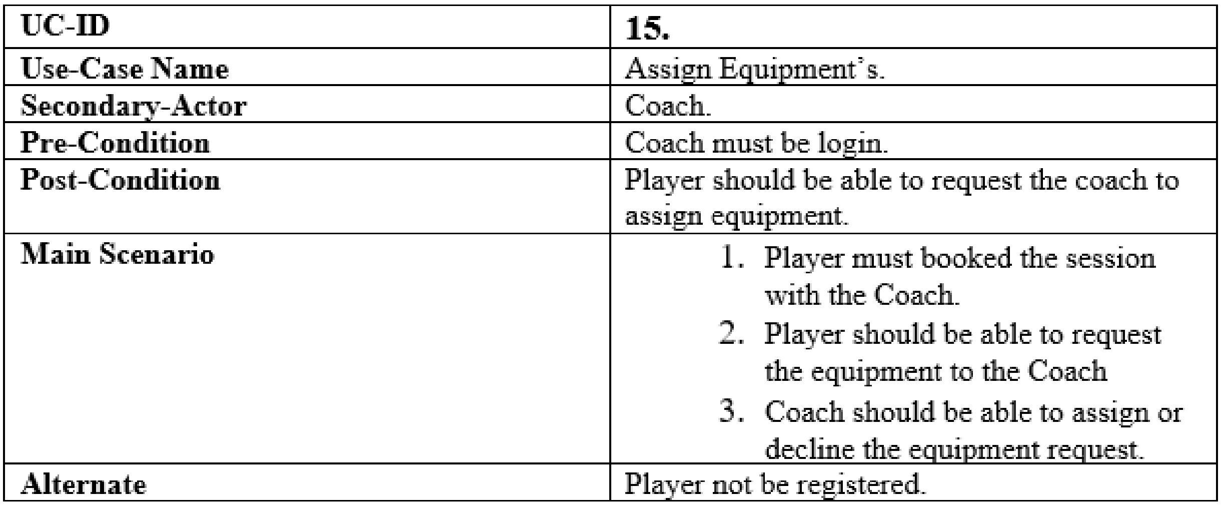
UC-ID	**13.**
Use-Case Name	Assign Guardian Portal.
Secondary-Actor	Coach.
Pre-Condition	Player must be login to our system and player age is greater than 12.
Post-Condition	Parent/Guardian can view the player report of the session.
Main Scenario	1. Player must booked the session with the Coach. 2. Coach should be able to manage the session report. 3. Coach should be able to assign the guardian portal 4. Guardian should be able to view the report.
Alternate	Player not be registered.

Table 4.14: Manage Emergency Record



UC-ID	**14.**
Use-Case Name	Manage Emergency Record.
Secondary-Actor	Coach.
Pre-Condition	Coach must be login.
Post-Condition	Add emergency case related player. Notify to guardian, the situation of player.
Main Scenario	1. Player must booked the session with the Coach. 2. Coach should be able to manage emergency case. 3. Coach should be able to notify the guardian of this player. 4. Guardian should be able to view the report.
Alternate	Player not be registered.

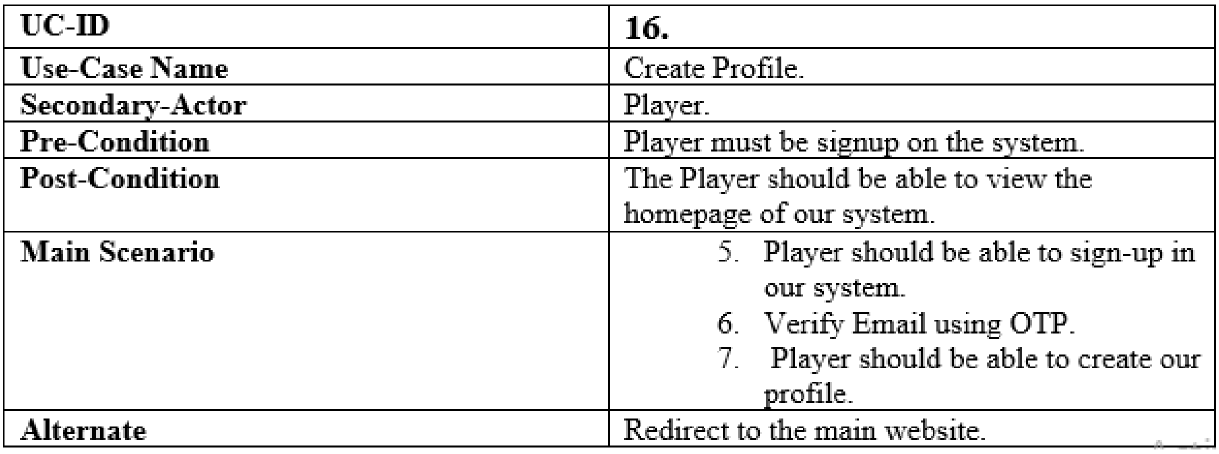
Table 4.15: Assign Equipment’s



UC-ID	**15.**
Use-Case Name	Assign Equipment’s.
Secondary-Actor	Coach.
Pre-Condition	Coach must be login.
Post-Condition	Player should be able to request the coach to assign equipment.
Main Scenario	1. Player must booked the session with the Coach. 2. Player should be able to request the equipment to the Coach 3. Coach should be able to assign or decline the equipment request.
Alternate	Player not be registered.

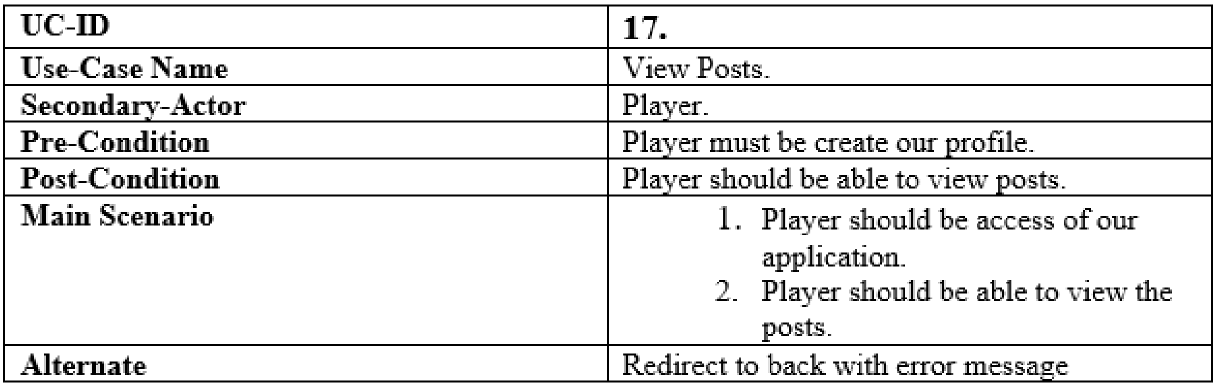
4.6.3Player

Table 4.16: Create Profile



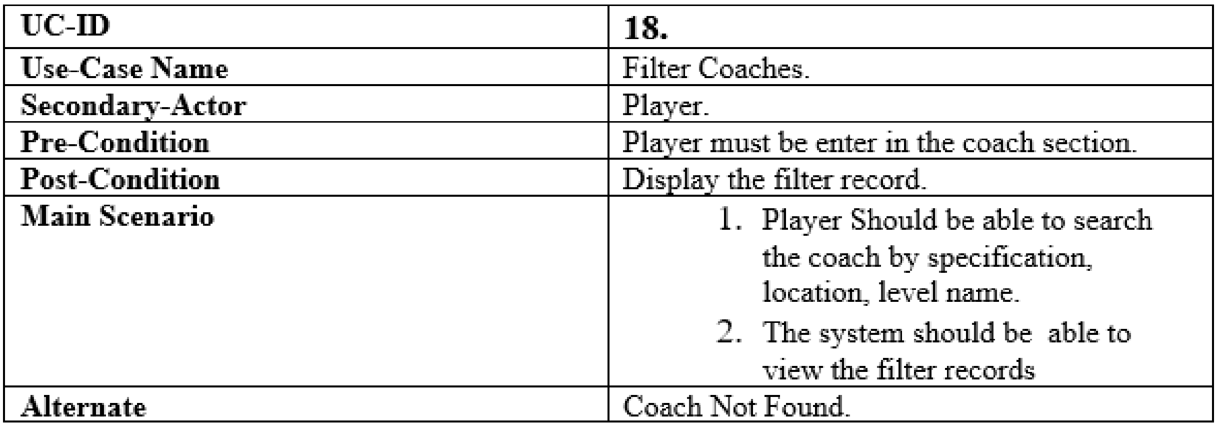
UC-ID	**16.**
Use-Case Name	Create Profile.
Secondary-Actor	Player.
Pre-Condition	Player must be signup on the system.
Post-Condition	The Player should be able to view the homepage of our system.
Main Scenario	5. Player should be able to sign-up in our system. 6. Verify Email using OTP. 7. Player should be able to create our profile.
Alternate	Redirect to the main website.

Table 4.17: View Posts



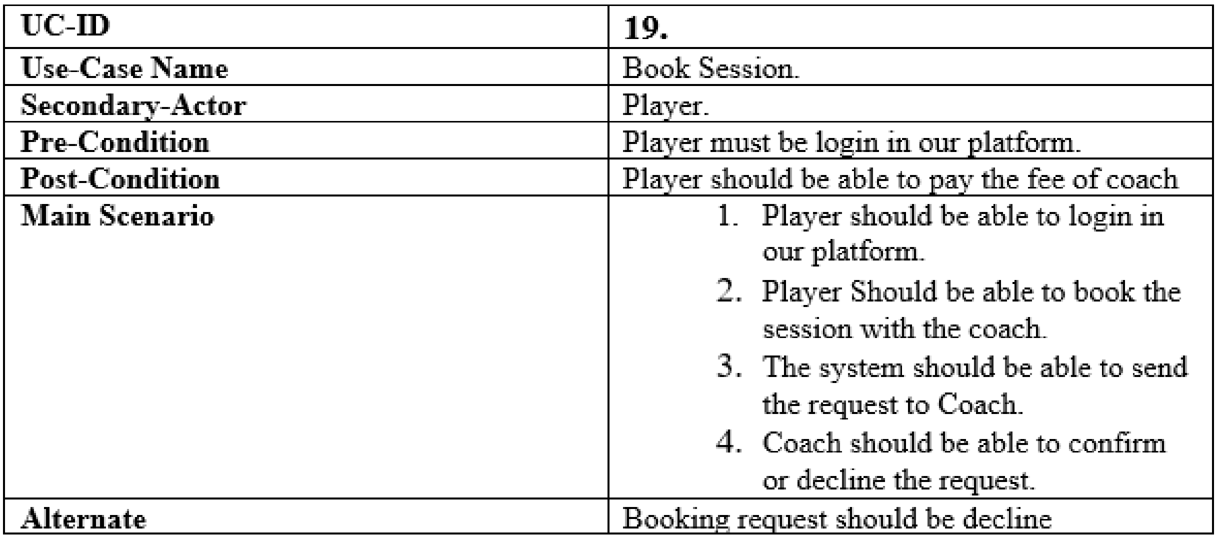
UC-ID	**17.**
Use-Case Name	View Posts.
Secondary-Actor	Player.
Pre-Condition	Player must be create our profile.
Post-Condition	Player should be able to view posts.
Main Scenario	1. Player should be access of our application. 2. Player should be able to view the posts.
Alternate	Redirect to back with error message

Table 4.18: Filter Coaches



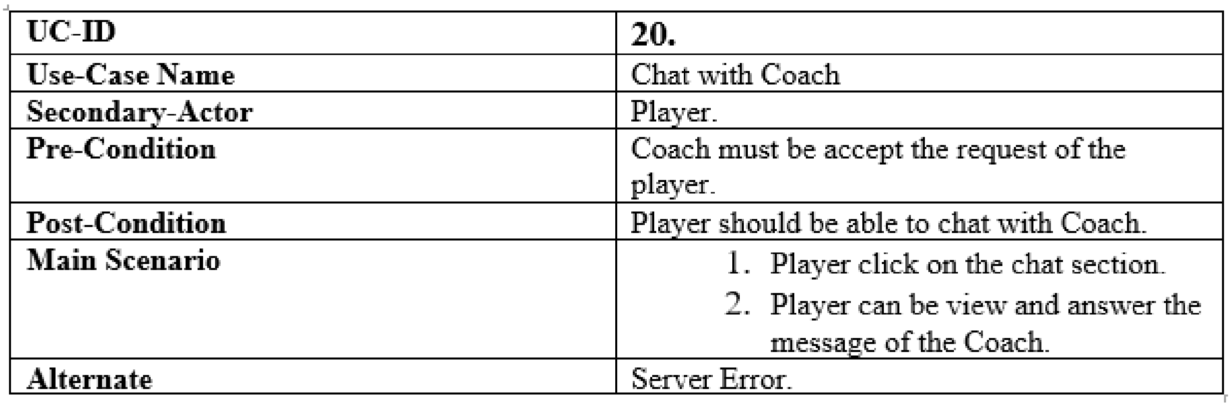
UC-ID	**18.**
Use-Case Name	Filter Coaches.
Secondary-Actor	Player.
Pre-Condition	Player must be enter in the coach section.
Post-Condition	Display the filter record.
Main Scenario	1. Player Should be able to search the coach by specification, location, level name. 2. The system should be able to view the filter records
Alternate	Coach Not Found.

Table 4.19: Book session



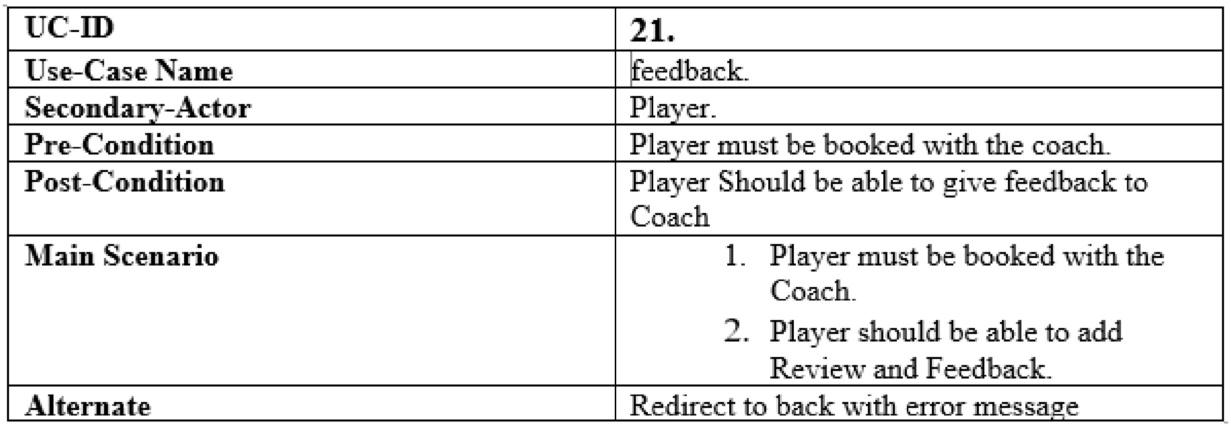
UC-ID	19.
Use-Case Name	Book Session.
Secondary-Actor	Player.
Pre-Condition	Player must be login in our platform.
Post-Condition	Player should be able to pay the fee of coach
Main Scenario	1. Player should be able to login in our platform. 2. Player Should be able to book the session with the coach. 3. The system should be able to send the request to Coach. 4. Coach should be able to confirm or decline the request.
Alternate	Booking request should be decline

Table 4.20: Chat with Coach



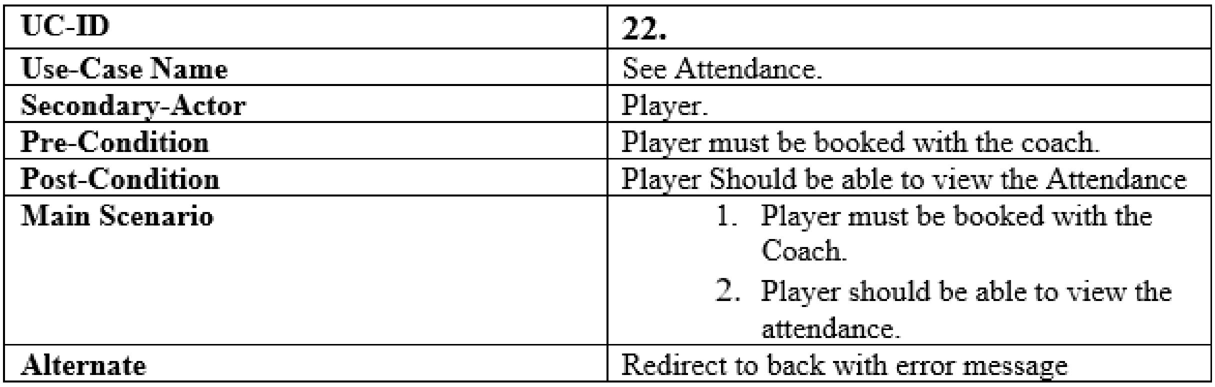
UC-ID	20.
Use-Case Name	Chat with Coach
Secondary-Actor	Player.
Pre-Condition	Coach must be accept the request of the player.
Post-Condition	Player should be able to chat with Coach.
Main Scenario	1. Player click on the chat section. 2. Player can be view and answer the message of the Coach.
Alternate	Server Error.

Table 4.21: Feedback



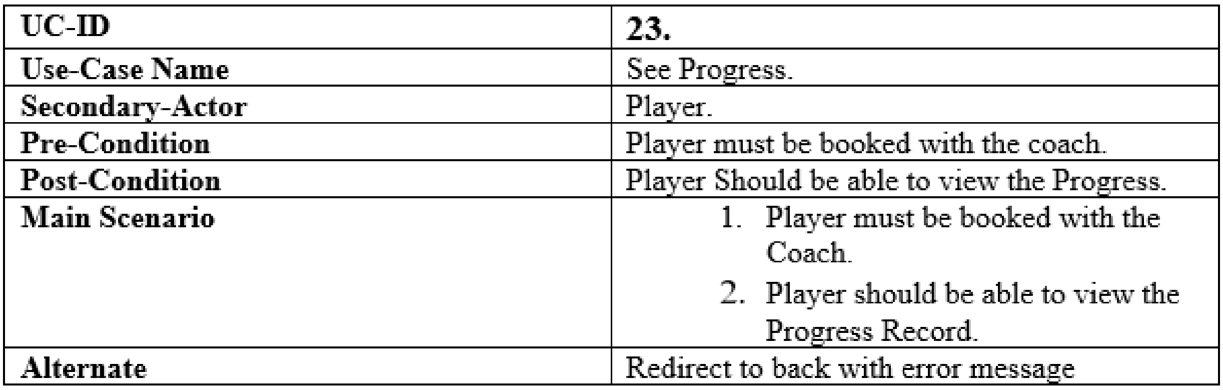
UC-ID	21.
Use-Case Name	feedback.
Secondary-Actor	Player.
Pre-Condition	Player must be booked with the coach.
Post-Condition	Player Should be able to give feedback to Coach
Main Scenario	1. Player must be booked with the Coach. 2. Player should be able to add Review and Feedback.
Alternate	Redirect to back with error message

Table 4.22: See Attendance



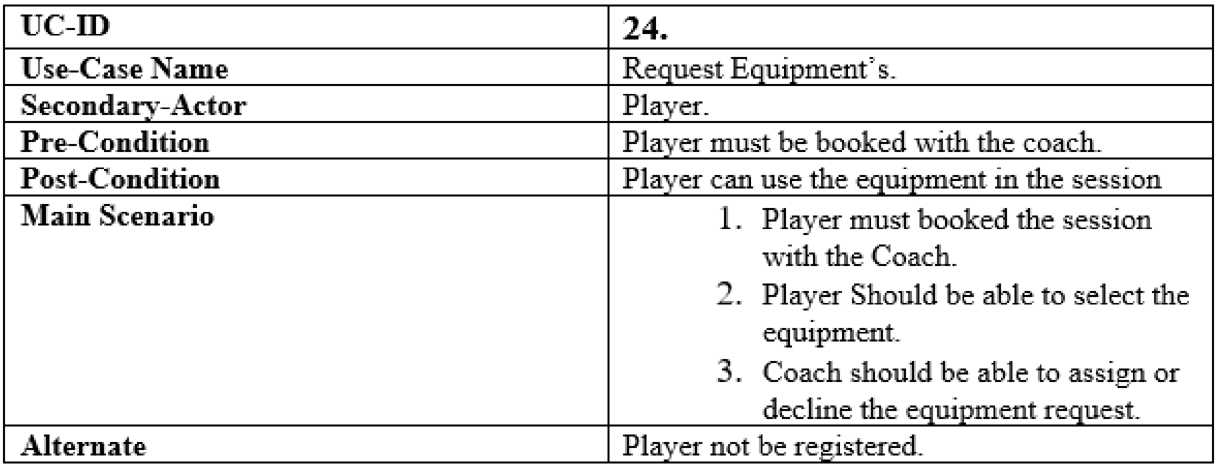
UC-ID	**22.**
Use-Case Name	See Attendance.
Secondary-Actor	Player.
Pre-Condition	Player must be booked with the coach.
Post-Condition	Player Should be able to view the Attendance
Main Scenario	1. Player must be booked with the Coach. 2. Player should be able to view the attendance.
Alternate	Redirect to back with error message

Table 4.23: See Progress Report



UC-ID	**23.**
Use-Case Name	See Progress.
Secondary-Actor	Player.
Pre-Condition	Player must be booked with the coach.
Post-Condition	Player Should be able to view the Progress.
Main Scenario	1. Player must be booked with the Coach. 2. Player should be able to view the Progress Record.
Alternate	Redirect to back with error message

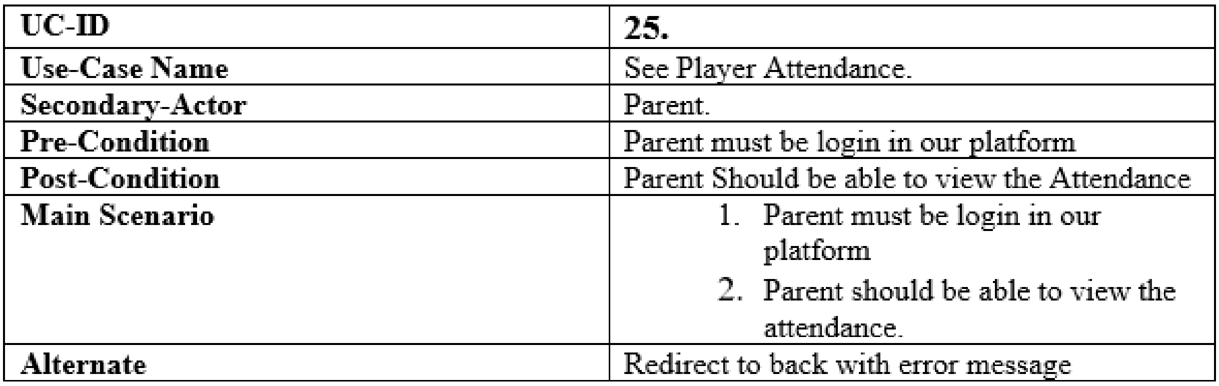
Table 4.24: Request Equipment’s



UC-ID	**24.**
Use-Case Name	Request Equipment’s.
Secondary-Actor	Player.
Pre-Condition	Player must be booked with the coach.
Post-Condition	Player can use the equipment in the session
Main Scenario	1. Player must booked the session with the Coach. 2. Player Should be able to select the equipment. 3. Coach should be able to assign or decline the equipment request.
Alternate	Player not be registered.

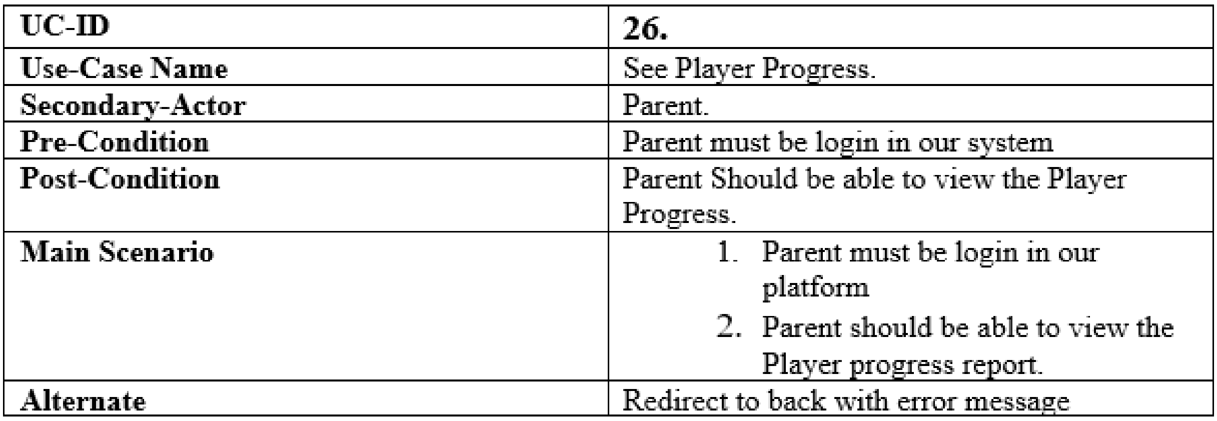
4.6.4 Parent

Table 4.25: See Players Attendance



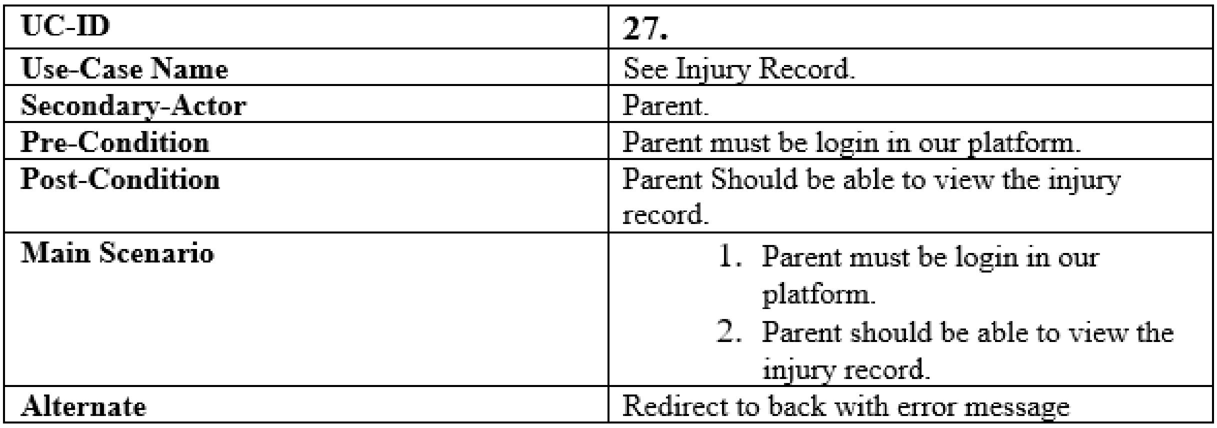
UC-ID	**25.**
Use-Case Name	See Player Attendance.
Secondary-Actor	Parent.
Pre-Condition	Parent must be login in our platform
Post-Condition	Parent Should be able to view the Attendance
Main Scenario	1. Parent must be login in our platform 2. Parent should be able to view the attendance.
Alternate	Redirect to back with error message

Table 4.26: See Player Progress



UC-ID	**26.**
Use-Case Name	See Player Progress.
Secondary-Actor	Parent.
Pre-Condition	Parent must be login in our system
Post-Condition	Parent Should be able to view the Player Progress.
Main Scenario	1. Parent must be login in our platform 2. Parent should be able to view the Player progress report.
Alternate	Redirect to back with error message

Table 4.27: See Injury Record



UC-ID	**27.**
Use-Case Name	See Injury Record.
Secondary-Actor	Parent.
Pre-Condition	Parent must be login in our platform.
Post-Condition	Parent Should be able to view the injury record.
Main Scenario	1. Parent must be login in our platform. 2. Parent should be able to view the injury record.
Alternate	Redirect to back with error message

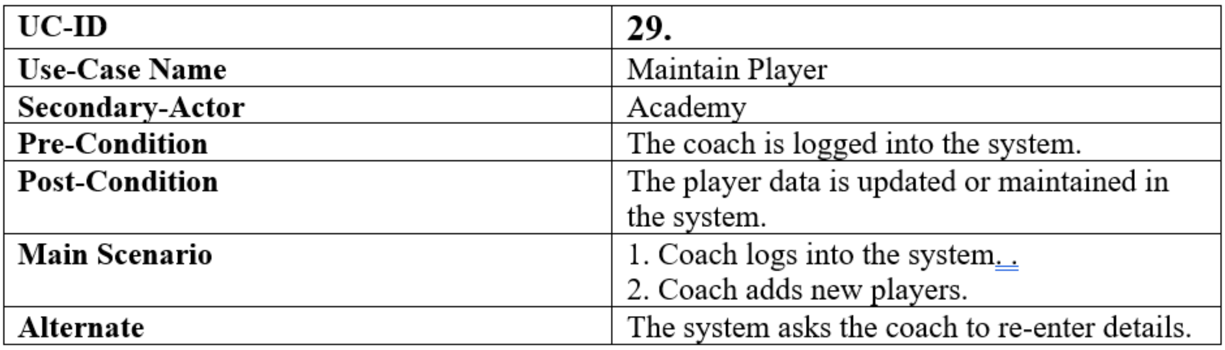
4.6.5 System:

Table 4.28: Assign Guardian Portal

UC-ID	**28.**
Use-Case Name	Assign Guardian Portal.
Secondary-Actor	System.
Pre-Condition	Player must be login to our system and player age is greater than 12.
Post-Condition	Parent/Guardian can view the player report of the session.
Main Scenario	1. Player must book the session with the Coach. 2. Coach should be able to manage the session report. 3. Coach should be able to assign the guardian portal 4. Guardian should be able to view the report.
Alternate	Player not be registered.

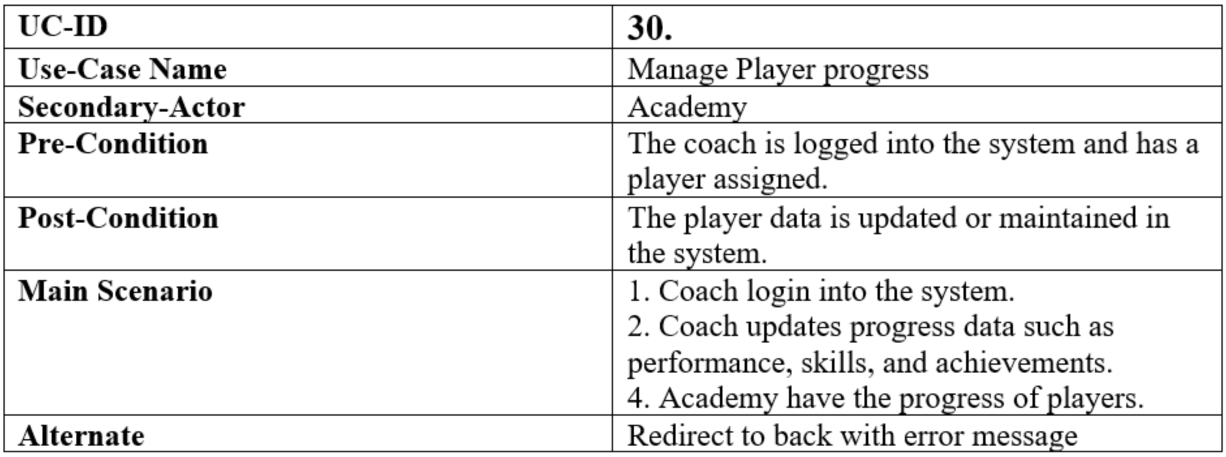
Academies:

Table 4.29: Maintain Player



UC-ID	**29.**
Use-Case Name	Maintain Player
Secondary-Actor	Academy
Pre-Condition	The coach is logged into the system.
Post-Condition	The player data is updated or maintained in the system.
Main Scenario	1. Coach logs into the system. . 2. Coach adds new players.
Alternate	The system asks the coach to re-enter details.

Table 4.30: Manage Player Progress



UC-ID	**30.**
Use-Case Name	Manage Player progress
Secondary-Actor	Academy
Pre-Condition	The coach is logged into the system and has a player assigned.
Post-Condition	The player data is updated or maintained in the system.
Main Scenario	1. Coach login into the system. 2. Coach updates progress data such as performance, skills, and achievements. 4. Academy have the progress of players.
Alternate	Redirect to back with error message

Chapter 5:

Implementation

Chapter: 05

Implementation

5.1. Endeavour (Team + Work + Way of Working)

5.1.1. Team

- Maaz Ejaz
- Muhammad Umer Farooq
- Muhammad Hateem Gulzar

5.1.2. Work Breakdown Structure

WBS #	**WBS Deliverable**	**Activity #**	**Activity to complete the deliverable**	**Duration (# of Days)**	**Responsible Team Member(s) & Role(s)**
2	Documentation	2.1	Team Members and Project Proposal	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Documentation	2.2	Project Proposal Document	4	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq Farooq
2.2	Project Proposal Document	2.2.1	Opportunity & Stakeholders	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.2	Existing Systems	2	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq

2.2	Project Proposal Document	2.2.3	Problem Statement	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.4	Proposed Solution	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5	Project Scope	4	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.1	Session Scheduling & Management	20	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.2	Real Time Communication	15	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.3	Progress Tracking and Reporting	15	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.4	Parent Portal	20	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.5	Emergency Module	15	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq

2.2	Project Proposal Document	2.2.5.6	Payment Gateway	15	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.2	Project Proposal Document	2.2.5.7	Coach Rating & Reviews	15	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.3	Proposal Plan	2.3.1	Proposed changes	2	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.3	Proposal Plan	2.3.2	Work Breakdown Structure	3	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.4	Planning Document	2.4.1	Problem the Software will solve	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.4	Planning Document	2.4.2	The use of the system	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.4	Planning Document	2.4.3	The Primary Function the Software	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.4	Planning Document	2.4.4	The Order of Development	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.4	Planning Document	2.4.5	Leadership Roles of the Project	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq

2.4	Planning Document	2.4.6	Each Team Member's Responsibilities	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.5	Documentation	2.5	Final Documentation Introduction	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.6	Documentation	2.6	Market Survey	10	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.6	Market Survey	2.6.1	Survey	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.6	Market Survey	2.6.2	Interviews	10	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.6	Market Survey	2.6.3	Brainstorming	6	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq

2.7	Documentation	2.7	Requirement Analysis	5	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq

2.7	Requirement Analysis	2.7.1	Elicited Requirements	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.7	Requirement Analysis	2.7.2	Functional Requirements	4	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.7	Requirement Analysis	2.7.3	Non-Functional Requirements	3	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.7	Requirement Analysis	2.7.4	Stakeholder Requirements	1	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.7	Documentation	2.8	System Design	8	Hateem Gulzar
2.8	System Design	2.8.1	Interface Design	8	Maaz Ejaz
2.8	System Design	2.8.2	Architectural Design	3	Maaz Ejaz
2.8	System Design	2.8.3	Use Cases	3	Maaz Ejaz
2.8	System Design	2.8.4	Activity Diagrams	5	Hateem Gulzar Maaz Ejaz Muhammad Umer Farooq
2.9	Implementation	2.9	Development	30	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
2.10	Testing & Performance	2.10	Testing & Performance Evaluation	20	Hateem Gulzar

	Evaluation				Muhammad Umer Frooq
3	System	3.1	Development Environment	10	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
3.1	System	3.2	Web Application	40	Muhammad Umer Farooq Maaz Ejaz
3.2	Web Application	4.2.1	Front End	20	Maaz Ejaz Hateem Gulzar Muhammad Umer Farooq
3.3	Web Application	3.2.1	Back End	20	Muhammad Umer Farooq

5.2. Components, Libraries, Web Services and stubs

- HTML
- CSS
- Tailwind CSS
- Material UI
- File Pond
- React
- React icons.

5.2.1 Back-End:

- Laravel.

5.2.2 Database:

- MySQL

5.3. IDE, Tools and Technologies

- Visual Studio Code
- Canva
- Visual Paradigm
- Git & GitHub
- Figma
- XAMPP

5.4 Best Practices / Coding Standard

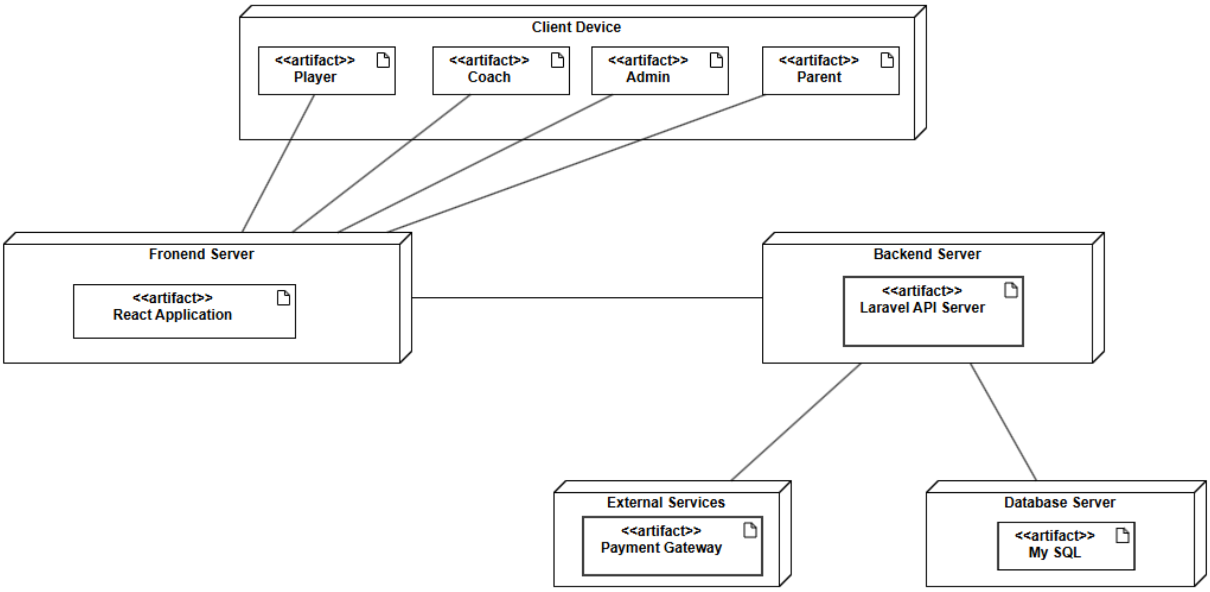
5.4.1 Software Engineering Methodologies.

- **Agile Methodology:** Interactive development, regular Feedback, and adaptive planning
- **Scrum Framework:** Regular sprints, daily stand–ups, and spirit reviews to manage tasks and progress.

5.4.2 Coding Standards

- **Naming Conventions:** Consistent naming for variables, methods, and classes.
- **Code Documentation:** Inline comments and documentation for better code readability.
- **Error Handling:** Proper error handling and logging mechanisms.
- **Code Reviews:** Regular code reviews to maintain code quality.

5.5 Deployment Diagram



CHAPTER: 06
Testing

Chapter: 06

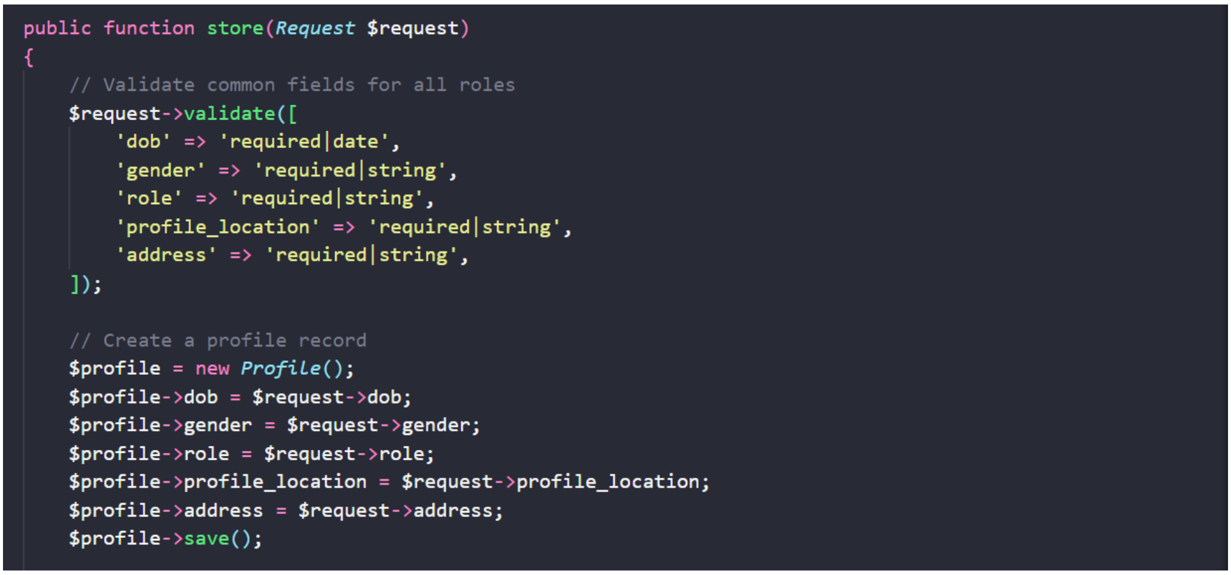
Testing

6.1: Introduction:

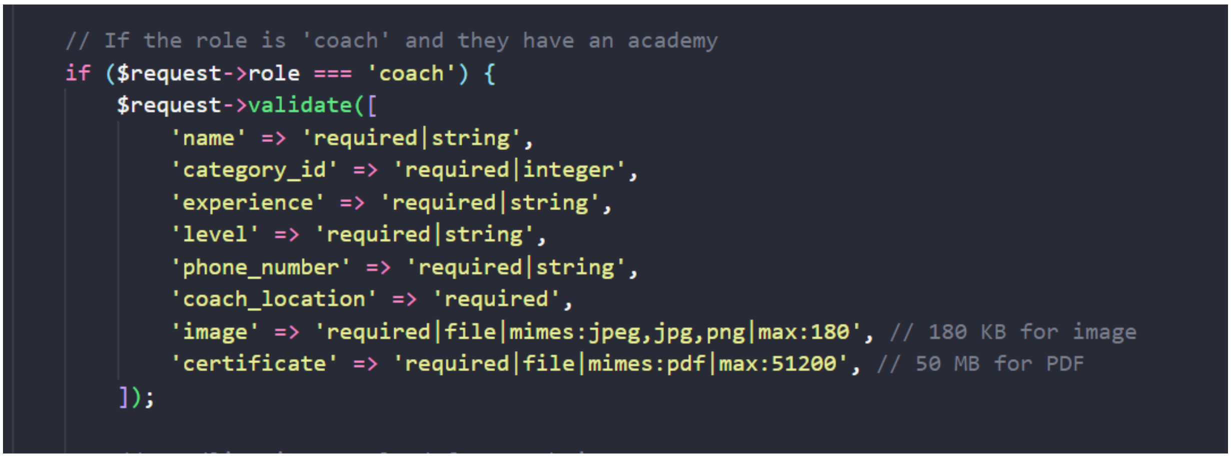
This chapter focuses on the systematic process of white-box testing, specifically applied to various functionalities within the application such as user signup, role-based workflows (coach and player), and management modules like attendance, scoreboard, and equipment handling. White-box testing, which involves testing the internal logic and structure of the code, ensures that each function performs as intended and meets the system's requirements. Through control flow graphs and detailed scenario testing, we validate the proper implementation of features and enhance the overall reliability and maintainability of the software.

6.2: White-Box Testing:

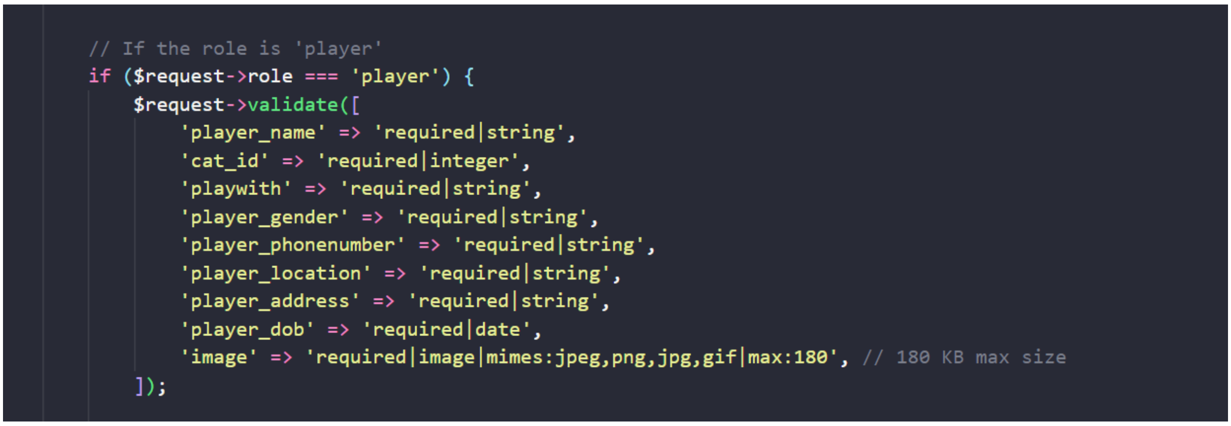
6.2.1: Role in Signup:



```php
public function store(Request $request)
{
    // Validate common fields for all roles
    $request->validate([
        'dob' => 'required|date',
        'gender' => 'required|string',
        'role' => 'required|string',
        'profile_location' => 'required|string',
        'address' => 'required|string',
    ]);

    // Create a profile record
    $profile = new Profile();
    $profile->dob = $request->dob;
    $profile->gender = $request->gender;
    $profile->role = $request->role;
    $profile->profile_location = $request->profile_location;
    $profile->address = $request->address;
    $profile->save();
```



```php
    // If the role is 'coach' and they have an academy
    if ($request->role === 'coach') {
        $request->validate([
            'name' => 'required|string',
            'category_id' => 'required|integer',
            'experience' => 'required|string',
            'level' => 'required|string',
            'phone_number' => 'required|string',
            'coach_location' => 'required',
            'image' => 'required|file|mimes:jpeg,jpg,png|max:180', // 180 KB for image
            'certificate' => 'required|file|mimes:pdf|max:51200', // 50 MB for PDF
        ]);
```



```
// If the role is 'player'
if ($request->role === 'player') {
    $request->validate([
        'player_name' => 'required|string',
        'cat_id' => 'required|integer',
        'playwith' => 'required|string',
        'player_gender' => 'required|string',
        'player_phonenumber' => 'required|string',
        'player_location' => 'required|string',
        'player_address' => 'required|string',
        'player_dob' => 'required|date',
        'image' => 'required|image|mimes:jpeg,png,jpg,gif|max:180', // 180 KB max size
    ]);
```

6.1.2.1Control Flow Graph:

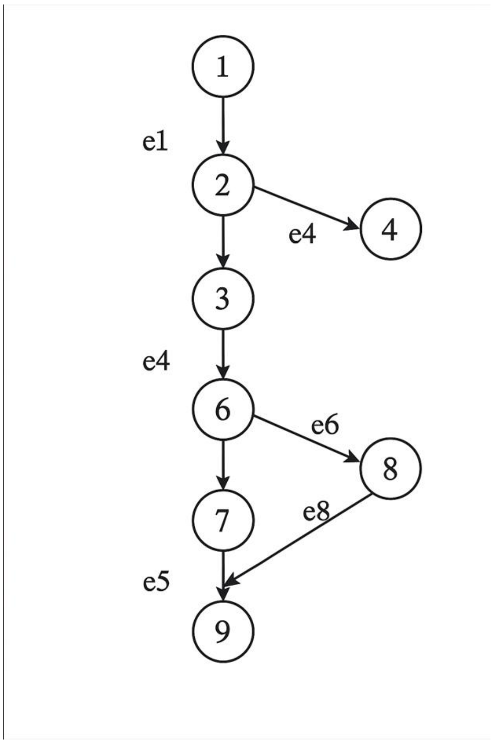


Fig: role base profile Create

6.2.1.2: Coach Flow:

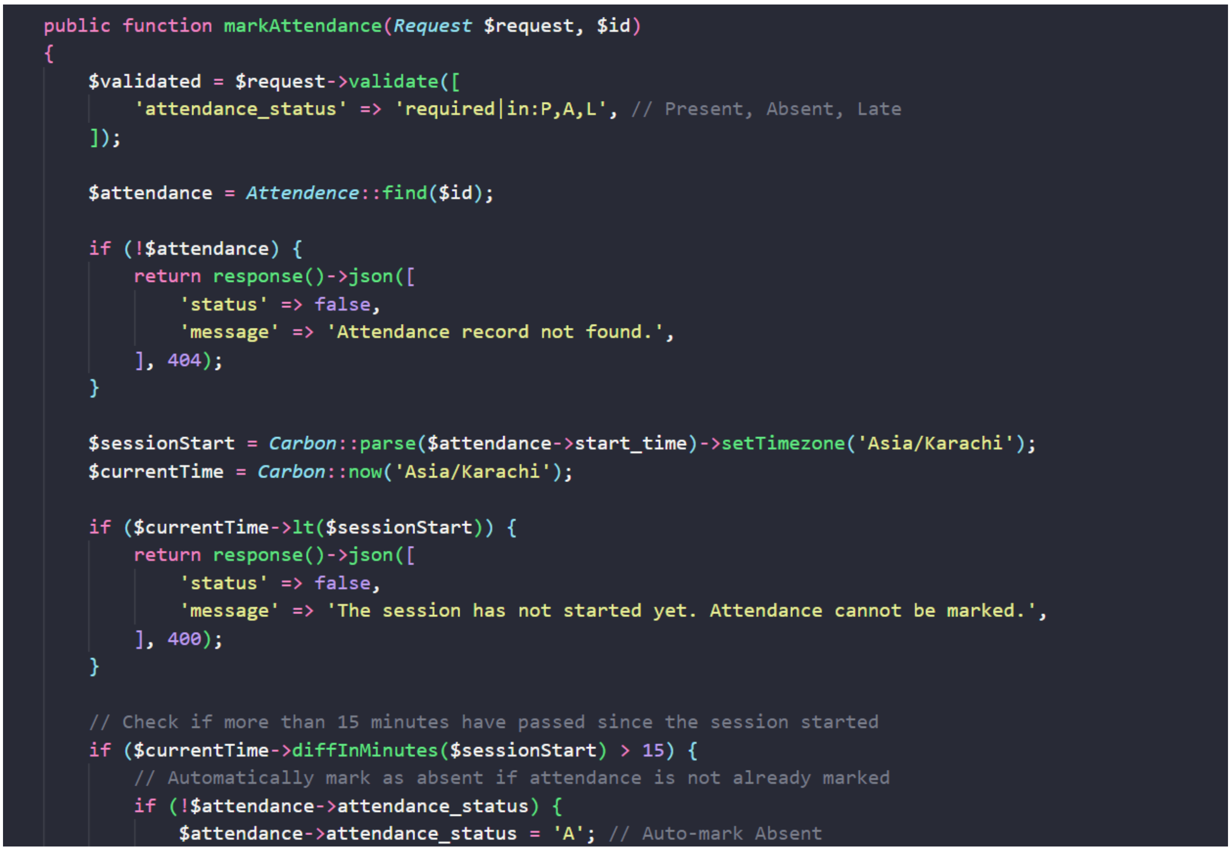
2 → 3 → 6 → 7 → 9

6.2.1.3: Player Flow:

2 → 4 → 5 → 8 → 7 → 9

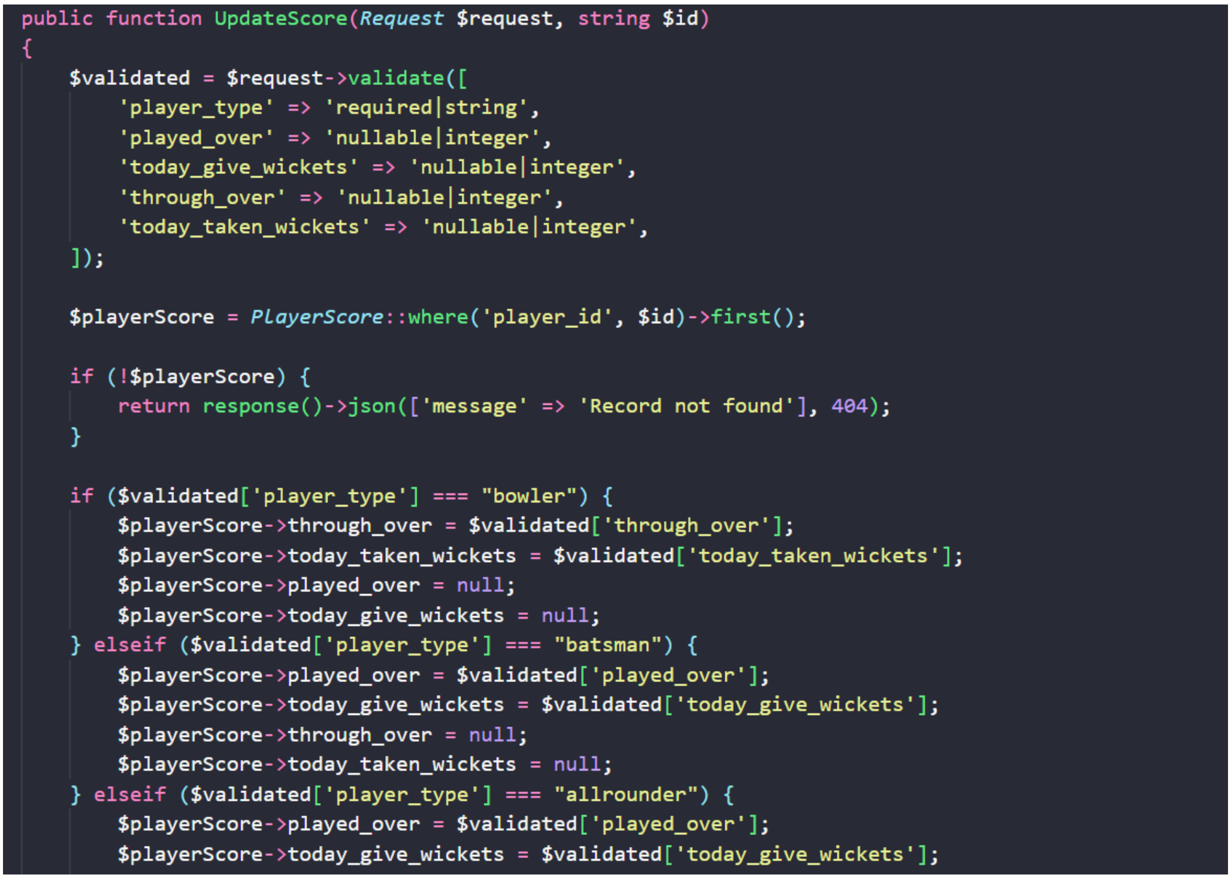
6.2.2: Attendance, Scoreboard, equipment’s:

6.2.2.1: Attendance:



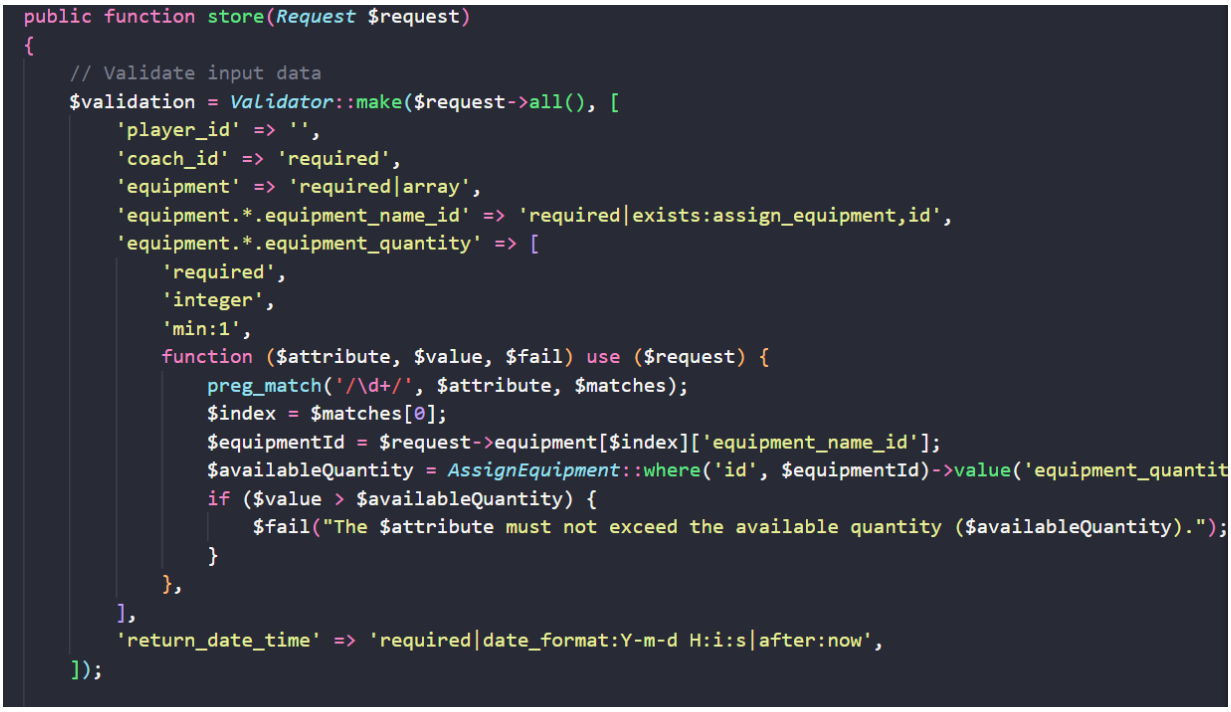
```
public function markAttendance(Request $request, $id)
{
    $validated = $request->validate([
        'attendance_status' => 'required|in:P,A,L', // Present, Absent, Late
    ]);

    $attendance = Attendence::find($id);

    if (!$attendance) {
        return response()->json([
            'status' => false,
            'message' => 'Attendance record not found.',
        ], 404);
    }

    $sessionStart = Carbon::parse($attendance->start_time)->setTimezone('Asia/Karachi');
    $currentTime = Carbon::now('Asia/Karachi');

    if ($currentTime->lt($sessionStart)) {
        return response()->json([
            'status' => false,
            'message' => 'The session has not started yet. Attendance cannot be marked.',
        ], 400);
    }

    // Check if more than 15 minutes have passed since the session started
    if ($currentTime->diffInMinutes($sessionStart) > 15) {
        // Automatically mark as absent if attendance is not already marked
        if (!$attendance->attendance_status) {
            $attendance->attendance_status = 'A'; // Auto-mark Absent
```

6.2.2.2: Scoreboard:



```
public function UpdateScore(Request $request, string $id)
{
    $validated = $request->validate([
        'player_type' => 'required|string',
        'played_over' => 'nullable|integer',
        'today_give_wickets' => 'nullable|integer',
        'through_over' => 'nullable|integer',
        'today_taken_wickets' => 'nullable|integer',
    ]);

    $playerScore = PlayerScore::where('player_id', $id)->first();

    if (!$playerScore) {
        return response()->json(['message' => 'Record not found'], 404);
    }

    if ($validated['player_type'] === "bowler") {
        $playerScore->through_over = $validated['through_over'];
        $playerScore->today_taken_wickets = $validated['today_taken_wickets'];
        $playerScore->played_over = null;
        $playerScore->today_give_wickets = null;
    } elseif ($validated['player_type'] === "batsman") {
        $playerScore->played_over = $validated['played_over'];
        $playerScore->today_give_wickets = $validated['today_give_wickets'];
        $playerScore->through_over = null;
        $playerScore->today_taken_wickets = null;
    } elseif ($validated['player_type'] === "allrounder") {
        $playerScore->played_over = $validated['played_over'];
        $playerScore->today_give_wickets = $validated['today_give_wickets'];
```

6.2.2.3: Equipment’s:



```
public function store(Request $request)
{
    // Validate input data
    $validation = Validator::make($request->all(), [
        'player_id' => '',
        'coach_id' => 'required',
        'equipment' => 'required|array',
        'equipment.*.equipment_name_id' => 'required|exists:assign_equipment,id',
        'equipment.*.equipment_quantity' => [
            'required',
            'integer',
            'min:1',
            function ($attribute, $value, $fail) use ($request) {
                preg_match('/\d+/', $attribute, $matches);
                $index = $matches[0];
                $equipmentId = $request->equipment[$index]['equipment_name_id'];
                $availableQuantity = AssignEquipment::where('id', $equipmentId)->value('equipment_quantit
                if ($value > $availableQuantity) {
                    $fail("The $attribute must not exceed the available quantity ($availableQuantity).");
                }
            },
        ],
        'return_date_time' => 'required|date_format:Y-m-d H:i:s|after:now',
    ]);
```

Chapter 07:
Conclusion and Outlook

Chapter 07: Conclusion and Outlook

7.1: Introduction:

Coach Selector is a user-friendly sports management platform designed to bring together players, coaches, and parents in one seamless experience. It addresses common issues in sports training such as scheduling problems, communication barriers, and a lack of progress tracking. The platform streamlines the booking process, enables real-time chat, tracks performance, and keeps everyone connected, ensuring a more efficient and transparent training environment for everyone involved.

7.2 Achievements and Improvements

Achievements:

1. We've successfully implemented a role-based registration system, allowing players, coaches, and parents to have tailored experiences.
2. The platform allows players to easily book sessions with coaches, chat in real-time, and leave reviews after their sessions.
3. Coaches can now manage their availability, accept or decline requests, and monitor player progress all in one place.
4. Parents have access to the parent portal, where they can track their child's attendance, progress, and any injuries.
5. We've integrated a secure payment system to ensure smooth financial transactions for both players and coaches.

Improvements:

1. We're looking to improve equipment management by adding features like tracking equipment availability, usage time, and return status.
2. We want to offer more detailed progress reports for players, including weekly and daily performance tracking.
3. We're planning to make it easier for players to leave feedback for coaches, helping both sides improve the experience.

7.3 Critical Review

Coach Selector does a great job of solving many common issues faced by sports academies and players. It's an all-in-one platform that connects players with coaches, helps track player progress, and ensures real-time communication. The role-based registration system keeps things organized and easy to use for everyone. We've also considered real-world sports management issues like equipment handling, attendance, and injuries, which makes the platform truly comprehensive. However, as we continue to grow, we'll need to test the system further and refine some features to make sure we're always meeting user expectations.

7.4 Future Recommendations/Outlook

1. We could introduce live video sessions for remote coaching, helping to reach players who can't always be present.
2. We plan to expand the platform to include more sports.
3. We'll also work on pushing notifications and alerts to keep users updated in real-time, so they never miss important information.

7.5 Summary

Coach Selector is a comprehensive platform that's designed to make sports training easier and more efficient for players, coaches, and parents. By offering role-based access and a wide range of features, we've created a system where players can book coaches, track their progress, and communicate easily. Coaches can manage their sessions, players, and equipment, while parents can stay updated on their child's performance and well-being. As the platform continues to evolve, we're excited to add more features and expand its capabilities to meet the needs of a growing and diverse user base.