*Submitted in partial fulfillment* *****of the requirements of the degree of*

**BACHELOR OF TECHNOLOG**

**SPORTS ACADEMY**

*Submitted in partial fulfilment of the requirements of the degree of*

**BACHELOR OF TECHNOLOGY**



**Session:- Jan-June 2023**





Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur

**Certificate**

This is to certify that this Major Project report entitled “**Sports Academy Management System”** by Dharmishtha Ajmera, Divyanshi Thakurani, Vaibhavraj Nath Chauhan has completed the work under my supervision and guidance, hence approved for submission in partial fulfilment for the award of the degree of Bachelor of Technology in Computer Science and Engineering to the Department of Computer Science and Engineering, Techno India NJR Institute of Technology, Udaipur during the academic session 2022-2023.

Mr Aaditya Maheshwari Dr Rimpy Bishnoi

Head of Industry Project Head of Department

Dept. of CSE, TINJRIT, Udaipur Dept. of CSE, TINJRIT, Udaipur

Date: —----------------- Date: —----------------



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur

**Examiner Certificate**

This is to certify that the following students **Vaibhavraj Nath Chauhan, Dharmishtha Ajmera, Divyanshi Thakurani** of final year B.Tech (Computer Science and Engineering), were examined for the project work entitled **“Sports Academy”** during the academic year 2019-2023 at Techno India NJR Institute of Technology, Udaipur.

**Remarks:**

**Date:**

Signature Signature

(**Internal Examiner**) (**External Examiner**)

ABSTRACT

1. Purpose

## *1.1. Introduction*

The Sports Academy Management System is a software solution designed to streamline and automate the operations of sports academies. It aims to enhance efficiency by simplifying athlete registrations, coaching sessions, facility bookings, scheduling, and administration. The system provides a centralized platform for athletes, coaches, and administrative staff to manage training schedules, track progress, communicate effectively, and handle financial transactions.

## *1.2. Scope*

Scope of this project is very broad regarding other manual bookings of court.

A few of them are:-

* Multi-usage: The system is designed to serve multiple sports academies simultaneously.
* Anytime, Anywhere Access: Being a web-based application, users can access it from any location and at any time.
* Reduced Human Involvement: The system automates processes, reducing the need for constant human assistance. Users can independently book courts and manage their reservations.

## 2. Document Overview

The first provides an introduction to the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter provides testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns the bibliography.

ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide Mr **Aaditya Maheshwari**, TINJRIT for their valuable guidance, encouragement and help in completing this work. Their useful suggestions for this whole work and cooperative behaviour are sincerely acknowledged.

I would like to express my sincere thanks to **Dr Rimpy Bishnoi, Head of Department, Department of Computer Science and Engineering (TINJRIT)** for giving me this opportunity to undertake this project.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead.

In the end, I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.



**Place: Udaipur**

**Date: —-----------**

**TABLE OF CONTENTS**

**Table of Contents**

Abstract

Acknowledgements

Table of Contents

List of Tables

List of Figures

List of Symbols

**CHAPTER – 1 Introduction**

Introduction

* 1. Purpose
  2. Overall Description

**CHAPTER – 2 Software Requirement Specificatio**

2.1. Purpose

2.2. Overall Description

2.2.1. Functional Requirements definitions

2.2.2. Use Cases

2.2.3. Non – Functional Requirements

2.3. Requirement Specification

2.3.1. External interface specifications

2.3.2. Functional Requirements

2.4. Hardware Specification

2.5. Software Specification

2.6. Hardware and Software Requirements in detail

**CHAPTER – 3 System Analysis and Design**21-39

3.1. Study & Weaknesses of current system

3.2 Requirements of new system

3.3. Feasibility Study

3.4. Features of new System

3.5. Data flow diagram

3.6. UML Modelling

3.6.1. Activity Diagram

3.6.2. Class Diagram

3.6.3. Context Diagram

3.6.4. Sequence Diagram

**CHAPTER – 4 Data Dictionary**

4.1. Table Details

4.2. E-R Diagram

**CHAPTER - 5 Screen Shots**

5.1. Admin Master

5.2. User Master

**CHAPTER – 6 Testing**

**CHAPTER – 7 Conclusion and Future Enhancements**

7.1. Limitations

7.2. Future Enhancement

7.3. Conclusion71

**CHAPTER – 8 Bibliography**

**List of Tables**

| **Table No.** | **Table Name** | **Page No.** |
| --- | --- | --- |
| Table 2.1 | Glossary | 7 |
|  | **Use Cases** |  |
| Table 2.2 | Access Home page | 15 |
| Table 2.3 | User login or signup | 15 |
| Table 2.4 | Create category/subcategory/sub/topic/question | 16 |
| Table 2.5 | Student chooses topic, gives test and views result | 18 |
|  | **Database Design** |  |
| Table 4.1 | Admin Login | 41 |
| Table 4.2 | Advertisement | 42 |
| Table 4.3 | Answer | 42 |
| Table 4.4 | Category | 42 |
| Table 4.5 | Login | 42 |
| Table 4.6 | Member Details | 43 |
| Table 4.7 | Question | 43 |
| Table 4.8 | Sub Category | 44 |
| Table 4.9 | Subject | 44 |
| Table 4.10 | Topic | 44 |
| Table 6.1 | Test report with test data | 67 |

**List of Figures**

| **Figure No.** | **Figure Name** | **Page No.** |
| --- | --- | --- |
|  | **Use Cases** |  |
| Figure 2.2.2a | Access Home page | 10 |
| Figure 2.2.2b | User login or signup | 10 |
| Figure 2.2.2c | Admin selects, create a new category/subcategory/subject/topic/question | 12 |
| Figure 2.2.2d | Student chooses topic, gives test and views result | 13 |
|  | **Data Flow Diagrams** |  |
| Figure 3.1 | DFD Level 0 | 25 |
| Figure 3.2 | Registrations(DFD Level 1) | 26 |
| Figure 3.3 | Question Insert(DFD Level 1) | 27 |
| Figure 3.4 | Examination(DFD Level 1) | 28 |
| Figure 3.5 | Registrations(DFD Level 2) | 29 |
| Figure 3.6 | Question Insert(DFD Level 2) | 30 |
| Figure 3.7 | Examination(DFD Level 2) | 31 |
| Figure 3.8 | User Authentication(DFD Level 3) | 32 |
| Figure 3.9 | User Authentication(DFD Level 4) | 33 |
|  | **UML Modelling** |  |
|  | *Activity Diagrams* |  |
| Figure 3.10 | For Student | 34 |
| Figure 3.11 | For Admin | 34 |
| Figure 3.12 | Class Diagram | 35 |
| Figure 3.13 | Context Diagram | 36 |
|  | *Sequence Diagram* |  |
| Figure 3.14 | Login | 37 |
| Figure 3.15 | Student Registration | 37 |
| Figure 3.16 | Manage Test | 38 |
| Figure 3.17 | Appear for test | 39 |
| Figure 4.1 | E-R Diagram | 45 |

**List of Symbols**

| **Term** | **Definition** |
| --- | --- |
| Admin | The only user who has the permission to insert or update category etc. in the database. |
| Entry | Admin stored in the Database |
| Html | Hyper text markup language |
| IEEE | Institute of Electrical and Electronic Engineers |
| QA | Quality assurance |
| SCMP | Software Configuration Management Plan |
| SDD | Software Design Document |
| SQAP | Software Quality Assurance Plan |
| SRS | Software Requirements Specification |
| Web Site | A place on the world wide web |

***CHAPTER – I***

***INTRODUCTION***

# Introduction

# 1.1. Purpose

## *1.1.1. Introduction*

The Sports Academy Management System is a software solution designed to streamline and automate the operations of sports academies. It aims to enhance efficiency by simplifying athlete registrations, coaching sessions, facility bookings, scheduling, and administration. The system provides a centralized platform for athletes, coaches, and administrative staff to manage training schedules, track progress, communicate effectively, and handle financial transactions.

## *1.1.2. Scope*

Scope of this project is very broad regarding other manual sports academies,.A few of them are:-

* Multi-usage: The system is designed to serve multiple sports academies simultaneously.
* Anytime, Anywhere Access: Being a web-based application, users can access it from any location and at any time.
* Reduced Human Involvement: The system automates processes, reducing the need for constant human assistance. Users can independently book courts and manage their reservations.

## *1.1.3. References*

[IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition.

[Bruade] The principal source of textbook material is “Software Engineering: An Object-Oriented Perspective” by Eric J. Bruade (Wiley 2001).

[Wiley] ASP.NET Bible by MridulaParihar.

## *1.1.4. Document Overview*

1. Introduction

* Overview of the Sports Academy Management System
* Purpose and objectives of the project
* Scope of the system

1. System Analysis and Requirements

* Description of the current system (if any) and its limitations
* Elicitation of requirements through interviews and surveys
* Use case diagram depicting the interaction between admin and users

1. System Design

* Architectural overview of the system
* Class diagram illustrating the key classes and their relationships
* Description of each class and its attributes
* Sequence diagrams showing the flow of activities during court creation and booking processes

1. Database Design

* Entity-Relationship (ER) diagram representing the database structure
* Description of tables, fields, and relationships
* Schema definition and normalization techniques employed

1. User Interface Design

* Wireframes or screenshots of the user interface
* Description of the interface components and their functionality
* Usability considerations and design principles followed

1. System Implementation

* Choice of programming language, frameworks, and technologies
* Overview of the development process and methodology
* Details of major system modules and their implementation

1. System Testing and Quality Assurance

* Test plan outlining different types of tests conducted
* Description of test cases and expected results
* Defect tracking and resolution process
* Quality assurance techniques applied

1. System Deployment

* Hardware and software requirements for deployment
* Deployment architecture and environment setup
* Installation and configuration steps

9.User Manual

* Instructions for admin: Court creation, managing slots, and user management
* Instructions for users: Court booking, slot availability, and cancellation

10.Conclusion

* Summary of the project
* Achievements and challenges faced during development
* Future enhancements and recommendations

# 1.2. Overall description

Online court booking is designed for sports academies where users book their court on a regular basis. The system handles all the operations and updates immediately which saves the precious time of users as well as sports academy owners.

## *1.2.1. Functional requirements definitions*

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

## *1.2.2. Use cases*

This system will be used in two User Modules which are Administrator and User. As both of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of both User Modules have been described below.

[1] Admin can do the following functions in the Administrator Module

* Add new courts, providing relevant details.
* Modify court information as needed.
* Access user profiles and view their details.
* Retrieve user bookings based on specified filters such as date and court.

[2] Users can do the following functions in the User Module

* Reserve an available court slot by booking it.
* View their own bookings to keep track of scheduled courts.
* Access details about specific courts, including information and features.
* Cancel a booking if necessary.
* Login and logout from their user accounts.
* Generate a new password or reset their password if needed

## *1.2.3. User characteristics*

The administrative role encompasses several key responsibilities within the court management system. Firstly, the admin holds the capability to create, update, and delete court information. This entails managing the court database, ensuring accurate and up-to-date details are available for users.

Furthermore, the admin has the privilege of accessing and viewing all user profiles. This allows them to have a comprehensive overview of the system's users, their preferences, and any relevant information associated with their accounts.

In terms of bookings, the admin is also granted full visibility into the booking records. They can review all bookings made within the system, enabling them to monitor court occupancy, track usage patterns, and identify any potential conflicts or issues that may arise. Additionally, the admin possesses the ability to apply filters to the bookings, enabling them to sort and organize the data based on specific criteria such as date, time, or court type.

On the user side, individuals utilizing the court management system have the ability to book courts according to their preferences and availability. They can choose suitable time slots, select desired courts, and proceed with the booking process. Moreover, users are also empowered to update their existing bookings if changes or modifications are needed. This flexibility allows them to adapt to any schedule adjustments or unforeseen circumstances that may arise.

Lastly, users retain the option to delete their own bookings if they no longer require the court reservation. This ensures that the system remains accurate and up-to-date, as unneeded bookings can be promptly removed, making the courts available for other users.

Overall, the administrative role encompasses various tasks related to court management, while users are provided with the necessary functionality to book, modify, and delete their own court reservations as needed.

## *1.2.4. Constraints*

There is no maintainability of back up so availability will get affected.

Limited to HTTP/HTTPS.

Real-life credit card validation and Banking system is not implemented.

No multilingual support

# 

# 

***CHAPTER – II***

***SOFTWARE REQUIREMENT SPECIFICATION***

**Software Requirement Specification**

## *2.1. Glossary*

**Table 2.1**

| **Term** | **Definition** |
| --- | --- |
| Admin | The only user who has permission to insert or update court etc. in the database. |
| Entry | Admin stored in the Database |
| Html | Hypertext markup language |
| IEEE | Institute of Electrical and Electronic Engineers |
| QA | Quality assurance |
| SCMP | Software Configuration Management Plan |
| SDD | Software Design Document |
| SQAP | Software Quality Assurance Plan |
| SRS | Software Requirements Specification |
| Web Site | A place on the world wide web |

# 2.2. Overall description

Online court booking is designed for sports academies where users book their court on a regular basis. The system handles all the operations and updates immediately which saves the precious time of users as well as sports academy owners.

***2.2.1. Functional requirements definitions***

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

## *2.2.2. Use cases*

This system will be used in two User Modules which are Administrator and User. As both of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of both User Modules have been described below.

[1] Admin can do the following functions in the Administrator Module

* Add new courts, providing relevant details.
* Modify court information as needed.
* Access user profiles and view their details.
* Retrieve user bookings based on specified filters such as date and court.

[2] Users can do the following functions in the User Module

* Reserve an available court slot by booking it.
* View their own bookings to keep track of scheduled courts.
* Access details about specific courts, including information and features.
* Cancel a booking if necessary.
* Login and logout from their user accounts.
* Generate a new password or reset their password if needed

### 2.2.2a Use Case: Access Home Page

Brief Description:

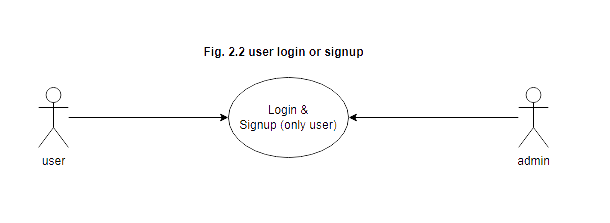
User uses the online sports academy system to access the home page.

Initial step-by-step description:

For this use case to be initiated, the user can use the sports academy system..

1. The user connects to the system using a web browser.
2. The user selects the Home link on the website home page.
3. The system passes the user to the website Home Page.

### 2.2.2b. Use Case: User Login or Signup



Brief Description:

The user either logs in or signs up.

Initial step-by-step description:

For this use case to be initiated the user must be on the website Home Page.

1. The User selects the “Login” link.
2. If the user is already registered, he successfully gets logged in.
3. Else user has to sign up.
4. The online examination system returns the signup form.
5. The User fills in the form.
6. The User clicks submit.
7. The System retains information in the database.
8. The System returns the user to the website Home Page.

### 2.2.2c. Use Case: Admin create court

### 

### 

### 

### 

### 

### 

### 

### Fig. 2.3 Admin Selects Create a New court

Brief Description:

The Admin chooses to create a new court.

Initial step-by-step description:

For this use case to be initiated, the Admin must be on the website’s Home page.

1. The Admin selects the create court link.
2. The website returns the corresponding page.
3. The Admin fills in the entries.
4. The Admin clicks submit.
5. The system adds the data to the Database.
6. The system returns the Admin to the Admin Home Page.

### 2.2.2d. Use Case: User create booking

### Fig. 2.4 User create booking

Brief Description:

The user can create booking for any court according to available slots.

Initial step-by-step description:

For this use case to be initiated, the student must be logged in on “Court” page.

1. The Student selects the “Court” link on tab.
2. If the User is not logged in, he’s directed to login page. If he’s not a member, he has to sign up. Once User gets logged in, he may book the slot.
3. The user chooses a court.
4. The user chooses the slot which is a available for that particular court.
5. And clicks on “create booking”.
6. This way slot is booked for user for it’s chosen time.

## *2.2.3. Non-functional requirements*

There are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

The web site must be compatible with Internet Explorer web browser.

# 2.3. Requirement specifications

## *2.3.1. External interface specifications*

None

***2.3.2. Functional Requirements***

## Table 2.2 Access Home Page

| **Use Case Name:** | Access Home Page |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | User is on the home page. |
| **Basic Path** | 1. The user connects to the system using a web browser. 2. The user selects the Home link on the website home page. 3. The system passes the user to the website Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | The User is on the Home Page |
| **Exception Path** | If there is a connection failure the website returns to the wait state |
| **Other** |  |

### Table 2.3User Login or Signup

| **Use Case Name:** | User Login or Signup |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Selects |
| **Precondition** | The User is on the Home Page |
| **Basic Path** | 1. The User selects the “Login” link. 2. If the user is already registered, he successfully gets logged in. 3. Else user has to sign up. 4. The online sports academy management system returns the signup form. 5. The User fills in the form. 6. The User clicks submit. 7. The System retains information in the database. 8. The System returns the user to the website Home Page. |
| **Alternate Path** | If after three attempts to match the name and password the website will return a message and block the user from the section. |
| **Postcondition** | The user is on the home page. |
| **Exception Path** | If the connection is terminated before the form is submitted, the fields are all cleared and the website is returned to the wait state. |
| **Other** |  |

### Table 2.4 Create new court

| **Use Case Name:** | Create new court |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | The Admin must be logged in and on Court page. |
| **Basic Path** | 1. The Admin selects the “create court” 2. The website returns the corresponding page. 3. The Admin fills in the entries. 4. The Admin can choose which fields to be entered. 5. The Admin clicks submit. 6. The system adds the data to the Database. 7. The system returns the Admin to the Admin Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | A record is created in the related Table of the Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the website is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before the Admin is returned to the Admin Home Page, the record is created in the Table of the Database. |
| **Other** |  |

### Table 2.5 User creates booking

| **Use Case Name:** | User chooses available slot for court and create booking for that slot. |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | The student must be logged in and on “Courts” page. |
| **Basic Path** | 1. The user selects the “court” link. 2. If the user is not logged in, he’s directed to login page. If he’s not a member, he has to sign up. Once user gets logged in, he may give the test. 3. The user chooses the slot and books it. 4. The user can view all its bookings. 5. The user can cancel bookings later. 6. Once the user books the slot slots are updated according to their availability. |
| **Alternate Path** |  |
| **Postcondition** | Th slot will successfully be booked for user and data will be stored in database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the website is returned to the wait state. |
| **Other** |  |

## 

## 2.4. Hardware Specification

**Client Side:**

* Internet Explorer: 6.0
* Processor: Pentium IV 2.0 and above.
* RAM : 512 MB
* Hard Disk : 80GB

**Server Side:**

* Processor: Pentium IV 2.0 and above.
* RAM : 1 GB
* Disk space : 4GB

**2.5. Software Specification**

**Client Side:**

* .NET Framework
* Web Browser
* Windows XP/Vista/Windows 7

**Web Server:**

* .NET Framework
* Windows XP/Vista/windows 7

**Data Base Server:**

* PostgreSQL Server

**2.6. Hardware and Software Requirements in detail**

**Hardware Requirements:**

* Processor: Pentium IV 2.0 and above.
* Internet Explorer: 6.0
* RAM : 512 MB
* Hard Disk : 80GB
* Disk space : 4GB

**Software Requirements:**

* Microsoft Visual Studio 2010
* Microsoft Visual Studio is an [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) from [Microsoft](http://en.wikipedia.org/wiki/Microsoft). It can be used to develop [console](http://en.wikipedia.org/wiki/Console_application) and [graphical user interface](http://en.wikipedia.org/wiki/Graphical_user_interface)[applications](http://en.wikipedia.org/wiki/Application_software) along with [Windows Forms](http://en.wikipedia.org/wiki/Windows_Forms) applications, [web sites](http://en.wikipedia.org/wiki/Web_site), [web applications](http://en.wikipedia.org/wiki/Web_application), and [web services](http://en.wikipedia.org/wiki/Web_service) in both [native code](http://en.wikipedia.org/wiki/Native_code) together with [managed code](http://en.wikipedia.org/wiki/Managed_code) for all platforms supported by [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows), [Windows Mobile](http://en.wikipedia.org/wiki/Windows_Mobile), [Windows CE](http://en.wikipedia.org/wiki/Windows_CE), [.NET Framework](http://en.wikipedia.org/wiki/.NET_Framework), [.NET Compact Framework](http://en.wikipedia.org/wiki/.NET_Compact_Framework) and [Microsoft Silverlight](http://en.wikipedia.org/wiki/Microsoft_Silverlight).
* .NET Framework 3.5
* Visual C#
* Web Browser Internet Explorer 6.0 and above
* Windows XP Service pack 3/Vista/Windows 7

**Deployment Configuration [Devops]**

Implementing DevOps in Your Sports Academy Project Using

1. Docker
2. EC2
3. CloudFront
4. GitHub Actions
5. Terraform
6. Load Balancer
7. Route 53

# Overall description

In today's fast-paced development environment, implementing DevOps practices can greatly enhance the efficiency, scalability, and security of your software projects. This text will guide you through the process of utilizing DevOps principles in your sports academy project. Specifically, we will explore how to use Docker for separating the backend and frontend, leverage EC2 instances with a load balancer for the backend, employ CloudFront for the frontend, implement a CI/CD pipeline using GitHub Actions, and provision infrastructure using Terraform for seamless deployment and management.

# 1. Dockerizing Backend and Frontend:

Docker provides an excellent solution for isolating and packaging your application components. By containerizing your backend and frontend separately, you gain flexibility, scalability, and ease of deployment. You can define Dockerfiles for each component, specifying the dependencies and configurations necessary for their proper execution. This allows you to ensure consistent environments across development, testing, and production stages.

# 2. EC2 Instances and Load Balancer for the Backend:

To handle the backend services, you can utilize Amazon EC2 instances provisioned through Terraform. Terraform allows you to define and manage your infrastructure as code, providing a declarative approach to provisioning resources. By leveraging Terraform, you can define the desired EC2 instances, networking configurations, and an Elastic Load Balancer (ELB) to ensure scalability, availability, and fault tolerance for your backend services.

# 3. CloudFront for Frontend Content Delivery:

For optimal delivery of your frontend content to end-users, Amazon CloudFront serves as a global content delivery network (CDN) that caches and distributes your static assets. By configuring CloudFront with your frontend application, you benefit from reduced latency, improved performance, and a more reliable user experience. Terraform can be used to define and provision the necessary CloudFront distribution, ensuring that your frontend content is delivered efficiently to your users.

# 4. CI/CD Pipeline with GitHub Actions:

GitHub Actions is a powerful and flexible tool for automating your CI/CD workflows. By defining workflows using YAML files in your project's repository, you can trigger builds, run tests, and deploy your application based on specific events or schedules. With GitHub Actions, you can integrate various testing frameworks, security scanners, and deployment scripts to ensure that your code is thoroughly validated before being deployed to production. You can also leverage secrets management to securely store sensitive information, such as API keys or database credentials.

# 5. Infrastructure Provisioning with Terraform:

Terraform enables you to provision your infrastructure using infrastructure-as-code principles. With Terraform, you can define your EC2 instances, load balancer, networking configurations, and CloudFront distribution in a declarative manner. This allows you to version control and manage your infrastructure alongside your application code. Terraform provides a consistent and reproducible way to create and manage your infrastructure, making it easier to collaborate, track changes, and ensure consistency across environments.

# 6. Security Considerations:

Security is a crucial aspect of any software project. When provisioning your infrastructure with Terraform, make sure to follow security best practices. Utilize Terraform's security features, such as AWS Secrets Manager for securely storing sensitive information, and implement appropriate security group rules, network ACLs, and encryption measures to protect your resources.

# Conclusion:

By implementing DevOps practices in your sports academy project, using Docker, EC2 instances provisioned with Terraform, CloudFront, GitHub Actions, and incorporating robust security measures, you can streamline development, deployment, and maintenance processes while ensuring the security and scalability of your application. This approach allows you to focus on delivering

***CHAPTER – III***

***SYSTEM ANALYSIS AND DESIGN***

**System Analysis and Design**

**3.1. Study & Weaknesses of Current System**

**Current System**

* The Current system of sports academy does not include approval from admin and payment methods for court booking.
* Whenever a new booking is created it is directly set to successful state, no approval from admin is asked.
* Being a web-based application, the sports academy management system relies on internet connectivity. Any disruptions or poor internet access can hinder user access and functionality.
* Off-the-shelf software solutions may have limitations in terms of customization and flexibility to meet specific requirements of the sports academy.
* The system's performance may be impacted during periods of high load, resulting in downtime, bugs, or performance issues that can negatively affect user experience and productivity.

**Weaknesses in Current System**

* Complexity and Cost: The current system is complex and expensive to implement, making it challenging for sports academies with limited resources.
* Time-Consuming Processes: The current system may involve manual and time-consuming processes for tasks such as court booking, registration, scheduling, and generating reports.
* Limited Accessibility: The current system may not be easily accessible to users from different locations or devices, restricting convenience and flexibility.
* Inefficient Communication: Communication between coaches, staff, athletes, and parents may be fragmented, relying on traditional methods like phone calls or emails, leading to potential delays and miscommunication.
* Lack of Real-Time Updates: The current system may not provide real-time updates on court availability, schedule changes, or cancellations, causing confusion and inconvenience for users.
* Ineffective Data Management: Data management in the current system may be disorganized, leading to difficulties in retrieving and analyzing information, hindering decision-making and reporting.
* Limited Integration: The current system may not integrate well with other software or platforms, creating challenges in sharing data or streamlining processes with external systems.
* Inadequate Security Measures: The current system may have vulnerabilities in terms of data security, potentially exposing sensitive user information to unauthorized access or breaches.
* Lack of Scalability: The current system may struggle to accommodate the growing needs and increasing user base of the sports academy, potentially leading to performance issues or limitations in functionality.
* Insufficient User Training and Support: Users may not receive comprehensive training or ongoing support for effectively utilizing the system, resulting in underutilization of features and reduced productivity.

**3.2. Requirements of New System**

**3.2.1. User Requirements**

The User requirements for the new system are to make the system fast, flexible, less prone to errors and reduce expenses and save time.

* User Registration: Easy account creation and registration process for coaches, staff, athletes, and parents.
* Court Booking: Seamless court availability checking and booking functionality.
* Schedule Management: Centralized schedule for training sessions, matches, and events.
* Communication and Collaboration: Messaging, notifications, and announcements for effective communication.
* Skill Development Resources: Access to training materials, exercise plans, and videos.
* Attendance and Absence Management: Tracking and managing athlete attendance and addressing absences.
* Payment and Fee Management: Online payment processing for registration and training fees.
* Mobile Accessibility: Mobile-friendly interface for easy access on smartphones and tablets.

**3.3. Feasibility Study**

A key part of the preliminary investigation that reviews anticipated costs and benefits and recommends a course of action based on operational, technical, economic, and time factors. The purpose of the study is to determine if the systems request should proceed further.

**3.3.1. Does the New System Contribute to the Overall Objectives of the Organization?**

The new system would contribute to the overall objectives to of the organization. It would provide a quick, error free and cost effective solution to the current process. It would provide a solution to many issues in the current system. As the new system is flexible and scalable it can also be upgraded and extended to meet other complex requirements which may be raised in the future. However it is up to the organization to upgrade or extend it.

**3.3.2. Can the New System be Implemented Using Current Technology?**

The organization has a computer laboratory which has about 50 machines connected by Internet LAN and managed by a server. It would be very easy to set up the system in the current environment as the application is web based it does not require to be installed on every machine. The database and IIS are set up on the server and the .Net Framework installed on every machine, the system can be started as quick as required by the management.

**3.4. Features of the New System.**

Features of the new web-based sports academy management system:

For Admin:

1. Court Management:

* Create new courts with details such as court type, capacity, and features.
* Update court details, such as availability and maintenance status.
* Delete or deactivate courts if necessary.

1. User Management:

* View and manage user accounts, including coaches, staff, athletes, and parents.
* Edit user details, such as contact information and roles.
* Deactivate or delete user accounts as needed.

1. Booking Management:

* View bookings for courts and specific dates, filtered by court or date.
* Modify or cancel bookings upon user requests or changes in court availability.
* Generate reports on court bookings and utilization.

For Users:

1. Court Booking:

* Browse available court slots and book courts based on their preferred date and time.
* Receive confirmation and booking details after successfully reserving a court.

1. Booking Management:

* View their own bookings, including court details, date, and time.
* Cancel bookings if necessary, subject to cancellation policies.

1. Court Details:

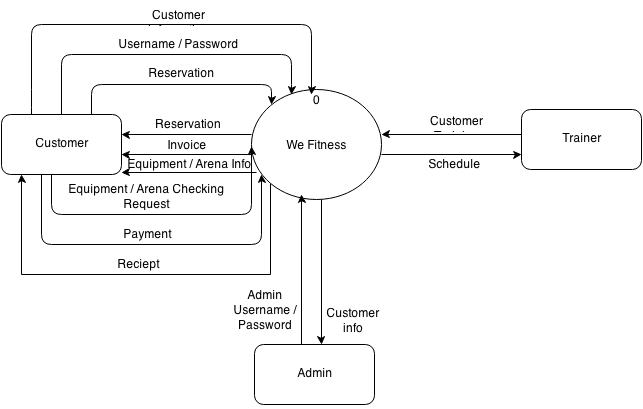
* Access information about each court, including court type, facilities, and availability.
* View court availability in real-time, ensuring accurate booking selections.

These features provide the core functionality for the new web-based sports academy management system, allowing admins to create and manage courts, view user and booking information, while users can book courts, manage their bookings, and access court details.

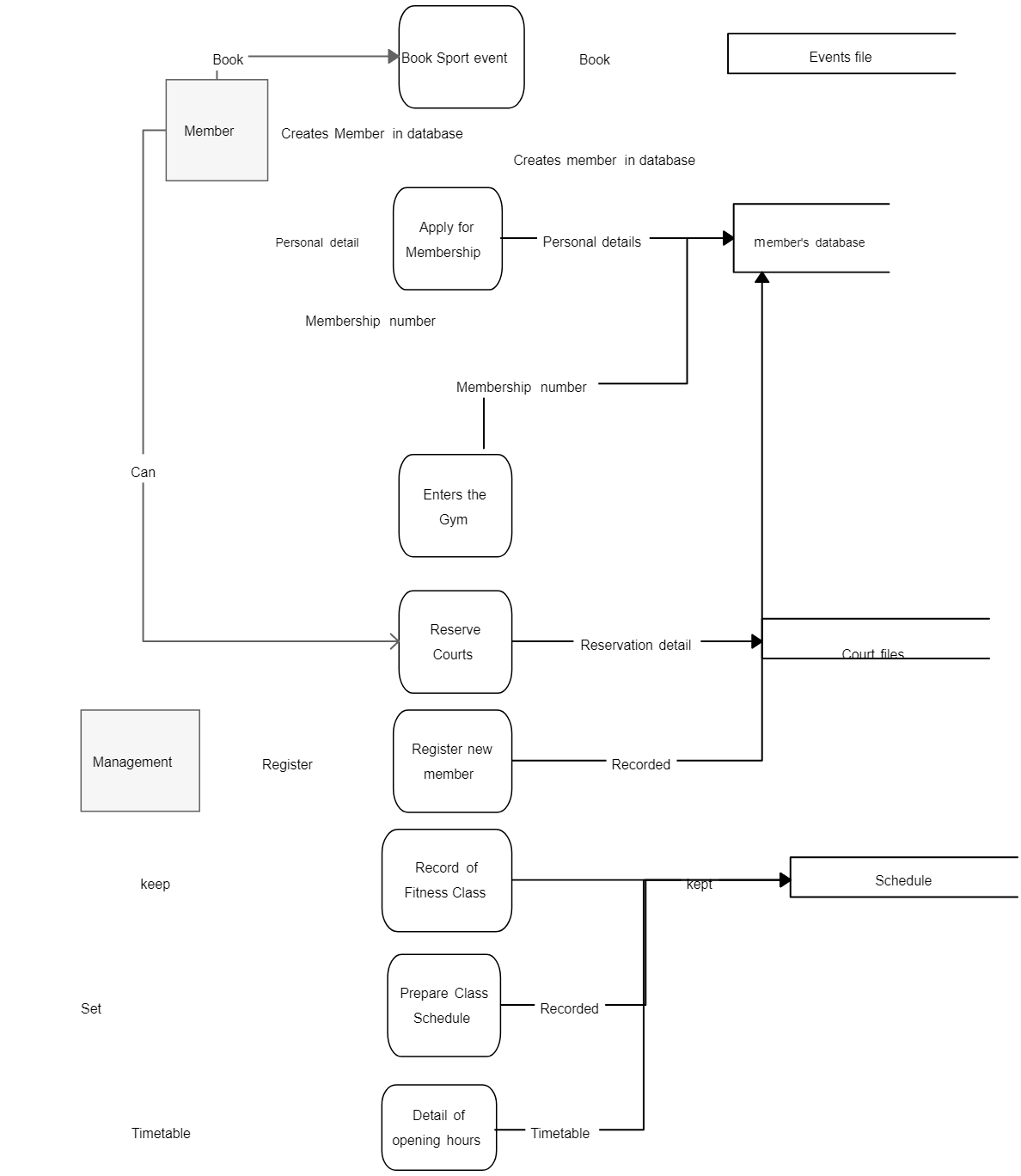
**3.5. Data Flow Diagram (DFD)**

The DFD (also known as *bubble chart*) is a simple graphical formalism that can be used to represent a system in terms of the input data into the system, various processes carried on these data, and the output data generated by the system.

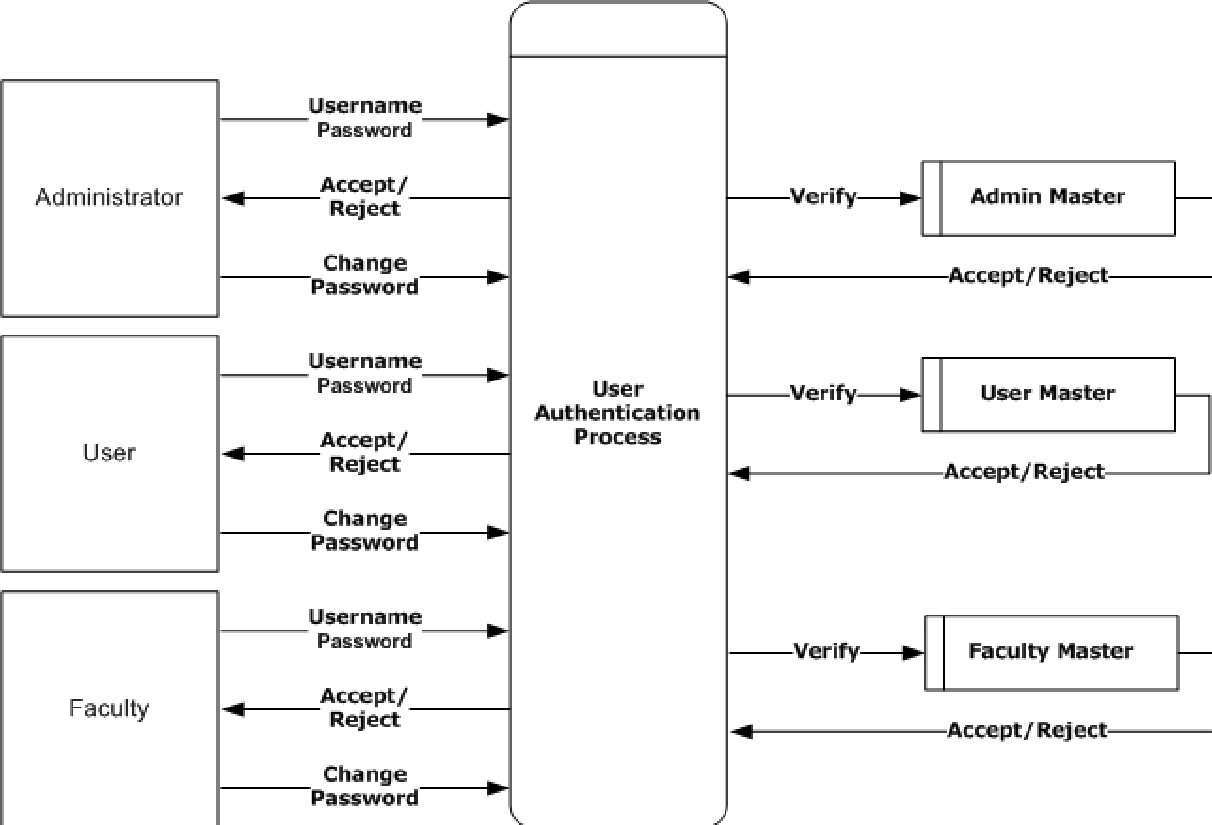
The main reason why the DFD technique is so popular is because the fact that the DFD is a very simple formalism – it is simple to understand and use. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among the functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchy represents various sub-functions.

****

**Fig. 3.1 ]DFD Level 0**

****

**Fig. 3.User booking (DFD Level 1)**

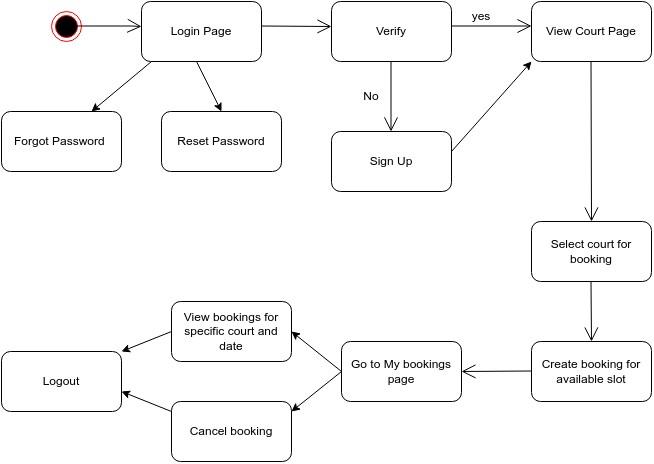
****

**Fig. 3.8 User Authentication (DFD Level 3)**

**3.6. UML Modelling**

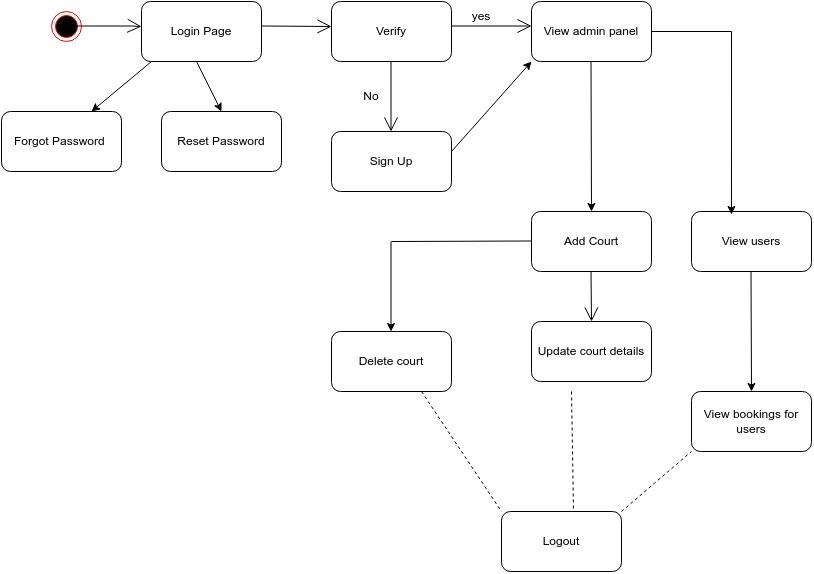
**3.6.1 Activity Diagram**

**3.6.1a. Activity Diagram for User**



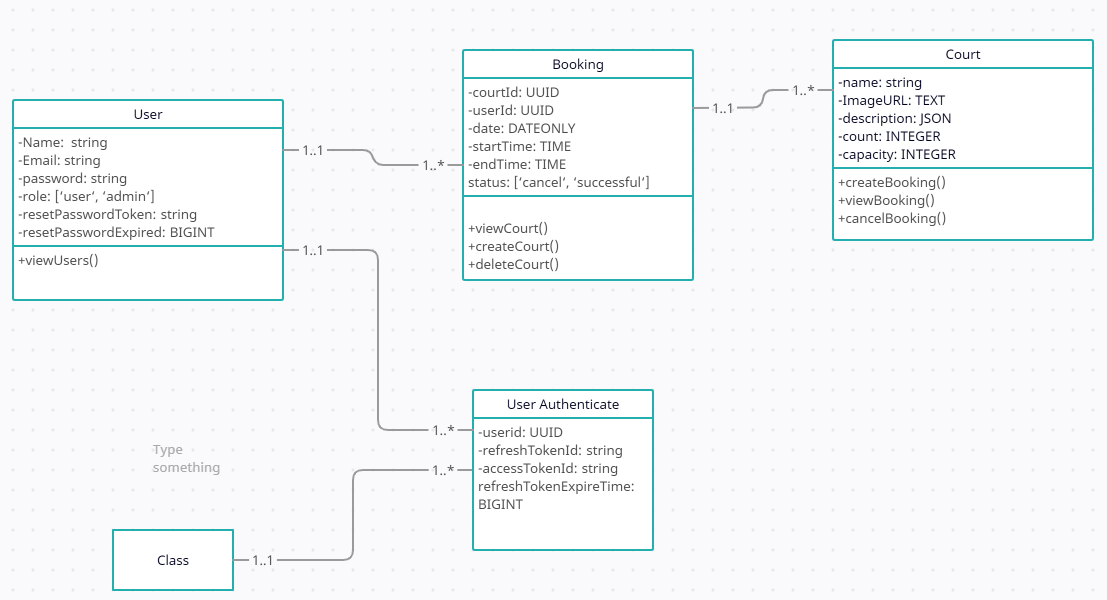
**Fig. 3.10 Activity Diagram for Student**

**3.6.1b. Activity Diagram for Admin**

****

**Fig. 3.11 Activity Diagram for Admin**

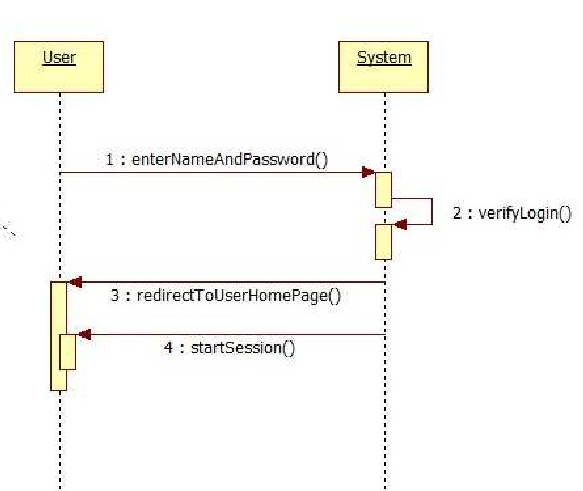
**3.6.2. Class Diagram**

****

**Fig. 3.12 Class Diagram**

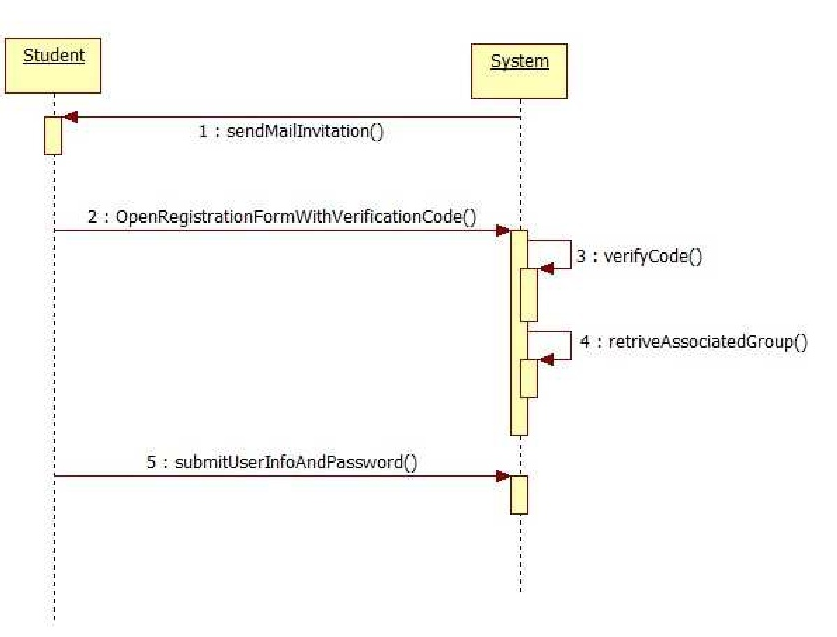
**3.6.4. Sequence Diagrams**

**3.6.4a. Login Sequence Diagram**

****

**Fig. 3.14 Login Sequence Diagram**

**3.6.4b. User Registration Sequence Diagram**

****

***CHAPTER – IV***

***DATA DICTIONARY***

**Data Dictionary**

A data dictionary is a catalog-a-repository of the elements in a system. As the name suggests, their elements center on data and the way they are structured to meet user requirements and organization needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores and processes. The data dictionary stores details and descriptions of these elements.

If analysis want to know characters are in a data item by what other names it is referenced in the system, or where it is referenced in the system, or where it is issued in the system, they should be able to find the answers in issued in the system, they should be able to find the answer in properly developed data dictionary.

The Dictionary contains two types of description for the data following through the system.

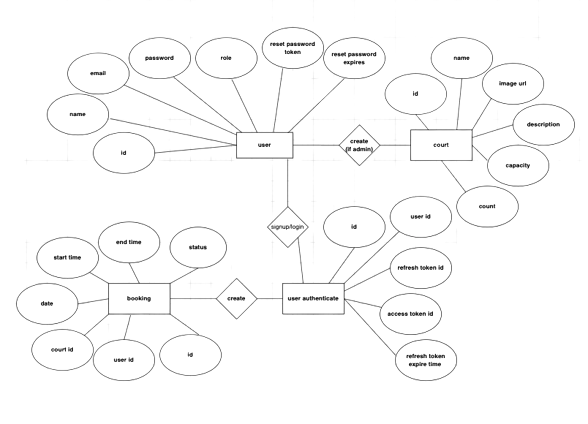
**1. Data Elements**

The most fundamental data is the elements. They are building blocks for all other data in the system. Data elements are also alternatively known as fields, data item or elementary item.

**2. Data Structure**

A data structure is a set if items that are related to one another and described a components in the system.

**4.2. E-R Diagram**



**Fig. 4.1 E-R Diagram**

***CHAPTER – V***

***TESTING***

**Testing**

**Testing Methodology**

Testing for the sports academy management system would involve various aspects to ensure its functionality, usability, and reliability. Here are some types of testing that can be conducted:

* Unit Testing: Verify the individual components or modules of the system, such as user registration, court creation, and booking management, to ensure they function correctly.
* Integration Testing: Test the integration and interaction between different modules and components of the system to ensure seamless communication and data flow.
* User Interface Testing: Evaluate the user interface for usability, consistency, and responsiveness across different devices and browsers. Ensure that users can easily navigate, book courts, and view information.
* Functional Testing: Validate the system against functional requirements, such as verifying that users can create bookings, update court details, view user information, and manage bookings.
* Performance Testing: Assess the system's performance under various load conditions to ensure it can handle multiple simultaneous users and court bookings without slowdowns or crashes.
* Security Testing: Test the system's security measures, such as user authentication, data encryption, and protection against common vulnerabilities to safeguard user information.
* Compatibility Testing: Ensure the system works seamlessly across different web browsers, operating systems, and devices to accommodate users with various preferences and setups.
* Usability Testing: Gather feedback from actual users to evaluate the system's ease of use, intuitiveness, and overall user experience. Identify any pain points or areas for improvement.
* Regression Testing: Conduct testing after implementing updates or changes to ensure that existing functionality is not affected and that new features work as expected.
* User Acceptance Testing: Involve end-users in the testing process to ensure that the system meets their specific needs and requirements. Validate that the system aligns with their expectations and effectively supports their tasks.
* It is crucial to plan and execute a comprehensive testing strategy, including both manual and automated testing techniques, to identify and resolve any issues or bugs before deploying the sports academy management system to production.

**Table 6.1 Test Report with test data**

Test Report: Sports Academy Management System

Date: [Insert Date]

Test Scenario: Court Booking Functionality

Test Data:

User credentials for admin:

* email: admin@mailinator.com
* Password: Admin@123

Court details:

Court 1:

* Type: Indoor
* Capacity: 4 players
* Availability: 9:00 AM - 6:00 PM

Court 2:

* Type: Outdoor
* Capacity: 6 players
* Availability: 8:00 AM - 10:00 PM

Test Results:

1. Unit Testing:

* User Registration: PASSED
* Court Creation: PASSED
* Booking Management: PASSED

1. User Interface Testing:

* User interface usability: PASSED
* Responsiveness on different devices and browsers: PASSED

1. Functional Testing:

* Admin Functions:
* Create Court: PASSED
* Update Court Details: PASSED
* View User Information: PASSED
* View Court Bookings by Court and Date: PASSED
* User Functions:
* Create Booking: PASSED
* Cancel Booking: PASSED
* View Court Details: PASSED
* View Own Bookings: PASSED

1. Security Testing:

* User Authentication: PASSED
* Data Encryption: PASSED
* Protection against common vulnerabilities: PASSED

1. Compatibility Testing:

* Tested on Chrome, Firefox, and Safari: PASSED
* Tested on Windows and macOS: PASSED
* Tested on desktop and mobile devices: PASSED

1. Usability Testing:

* User feedback and observations: Overall positive feedback, easy to navigate and use.

1. Regression Testing:

* Existing functionality: PASSED
* New features: PASSED

1. User Acceptance Testing:

* End-user feedback: Gathered positive feedback and user satisfaction.

Conclusion:

The Sports Academy Management System has undergone comprehensive testing, covering various aspects of functionality, usability, performance, security, compatibility, and user acceptance. The system has successfully passed all tests, demonstrating its reliability and suitability for managing court bookings, court details, user information, and booking management. User feedback has been positive, indicating that the system meets the requirements and expectations of the stakeholders.

***CHAPTER – VI***

***CONCLUSION AND FUTURE ENHANCEMENTS***

**7.1. Limitations**

Limitations of the Current Designed Sports Academy Management System:

* Limited Accessibility: The current system may only be accessible from specific computers or devices within the sports academy premises, limiting user convenience and accessibility. Users may not be able to access the system remotely or make bookings outside of the academy.
* Lack of Mobile Support: If the current system is not optimized for mobile devices, users may face challenges in accessing and using the system on smartphones or tablets. This could restrict the user's ability to make bookings or access information while on the go.
* Manual Court Availability Updates: If the current system does not provide real-time updates for court availability, users may face difficulties in booking courts. They may need to rely on manual communication or physically check court availability, leading to inefficiencies and potential booking conflicts.
* Limited User Self-Service: The current system may not offer self-service features that empower users to manage their bookings, modify details, or cancel reservations independently. This can create additional administrative overhead and dependency on the sports academy staff.
* Lack of Automated Notifications: Without automated notifications, users may not receive timely updates regarding their bookings or any changes in court availability. This can lead to confusion, missed opportunities, or wasted time for both users and staff.
* Insufficient Reporting and Analytics: The current system may not provide comprehensive reporting and analytics capabilities, making it challenging for administrators to gain insights into booking patterns, user preferences, or overall system performance. This limits the ability to make data-driven decisions and improve operational efficiency.
* Inefficient Booking Management: If the current system lacks advanced features for managing bookings, such as drag-and-drop scheduling, waitlist management, or flexible time slot options, administrators may face challenges in efficiently organizing and optimizing court bookings.
* Lack of Integration with Payment Gateways: If the current system does not integrate with popular payment gateways, users may face inconvenience in making online payments for court bookings. This could result in additional manual payment collection processes or delays in confirming bookings.
* Limited User Support: If the current system lacks adequate user support channels, such as online help resources, FAQs, or a dedicated support team, users may struggle to find answers to their queries or resolve any issues they encounter while using the system.
* Scalability Constraints: The current system may have limitations in scaling up to accommodate the growing needs of the sports academy, such as increasing the number of courts, users, or concurrent bookings. This could hinder the academy's expansion or lead to performance issues as the system reaches its capacity.
* Addressing these limitations and incorporating relevant enhancements into the current designed sports academy management system can significantly improve its functionality, user experience, and operational efficiency.
  1. **Future Enhancements**

Future Enhancements for the Sports Academy Management System:

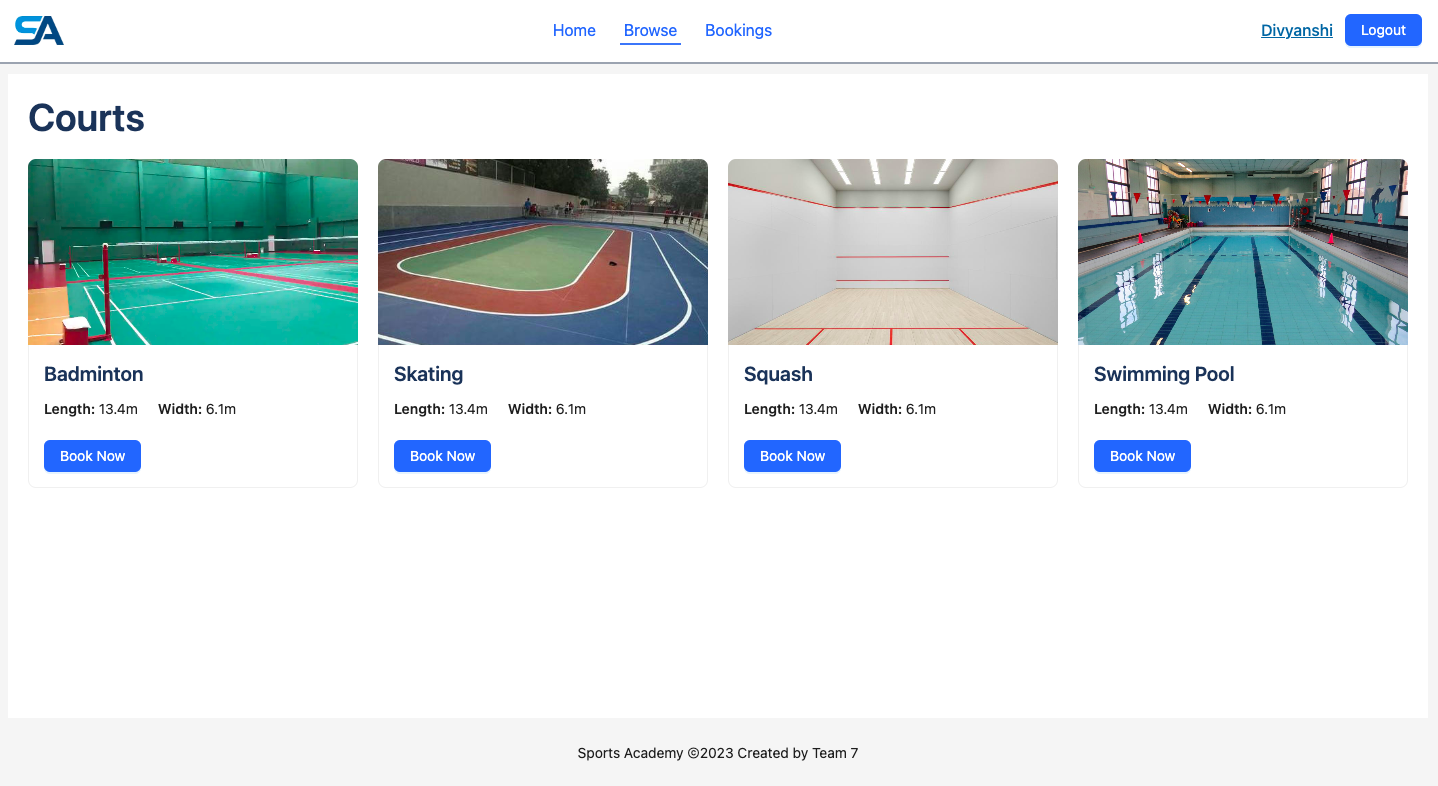
* Mobile Application: Develop a mobile application version of the system to provide users with greater flexibility and convenience in accessing and managing their bookings, court details, and user information from their smartphones or tablets.
* Online Payment Integration: Integrate popular payment gateways to enable users to make secure online payments for court bookings. This streamlines the payment process, eliminates manual collection, and improves the overall user experience.
* Real-Time Court Availability Updates: Implement real-time updates for court availability, allowing users to see the current status of each court, including booked slots and available time slots. This reduces conflicts and improves the accuracy of court bookings.
* Self-Service Portal: Create a user self-service portal where users can manage their bookings, modify details, cancel reservations, and view their booking history. This empowers users and reduces administrative overhead for the sports academy staff.
* Automated Notifications: Implement automated notifications via email or SMS to keep users informed about their bookings, changes in court availability, or any updates related to their reservations. This improves communication and reduces manual follow-ups.
* Advanced Reporting and Analytics: Enhance the reporting and analytics capabilities of the system to provide administrators with valuable insights into booking patterns, user preferences, revenue generation, and overall system performance. This data-driven approach enables informed decision-making and process optimization.
* Resource Management: Introduce features to manage and allocate resources such as sports equipment, coaching staff, or training materials. This helps in effective resource utilization and planning, ensuring that the academy runs smoothly and efficiently.
* Integration with Calendar Systems: Enable integration with popular calendar applications such as Google Calendar or Outlook to allow users to sync their court bookings with their personal calendars. This ensures better organization and scheduling for users.
* User Feedback and Ratings: Incorporate a feedback and rating system where users can provide feedback on their experiences, rate the quality of courts or coaching sessions, and provide suggestions for improvement. This helps in understanding user needs and enhancing service quality.
* Social Media Integration: Integrate the system with social media platforms to enable users to share their bookings or invite friends to join their sessions. This helps in promoting the sports academy and increasing user engagement.
* By implementing these future enhancements, the sports academy management system can stay competitive, improve user satisfaction, streamline operations, and adapt to evolving user needs and technological advancements in the sports industry.
  1. **Conclusion**

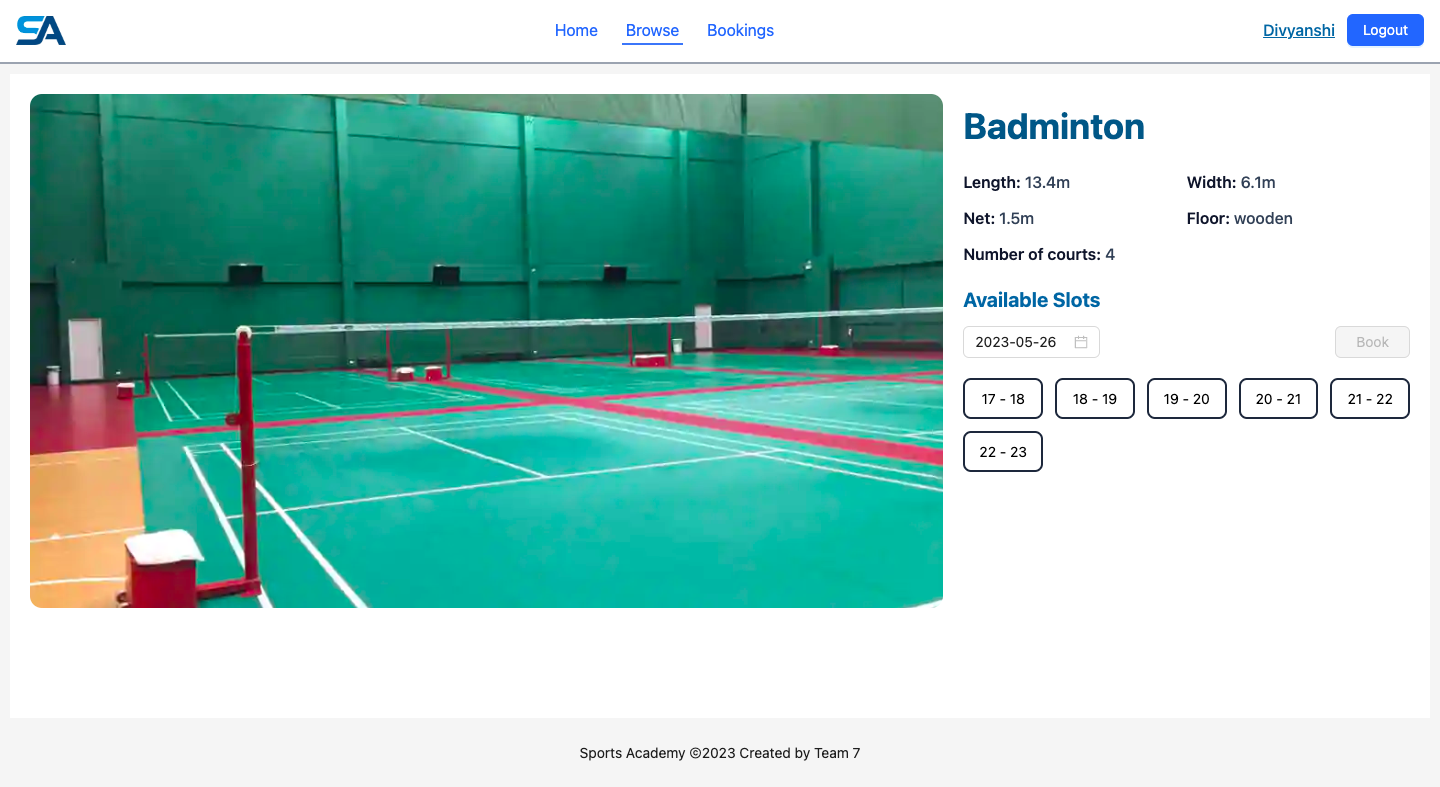
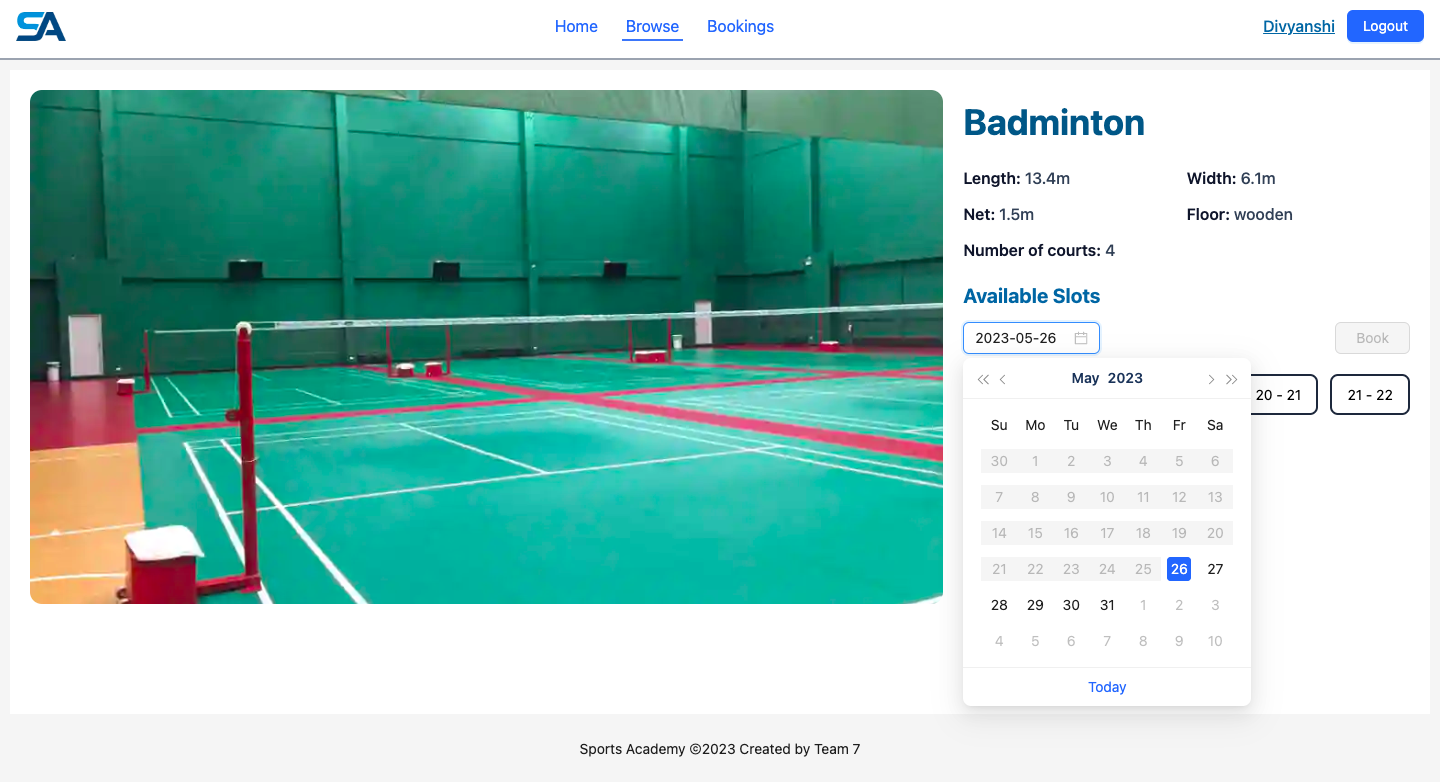
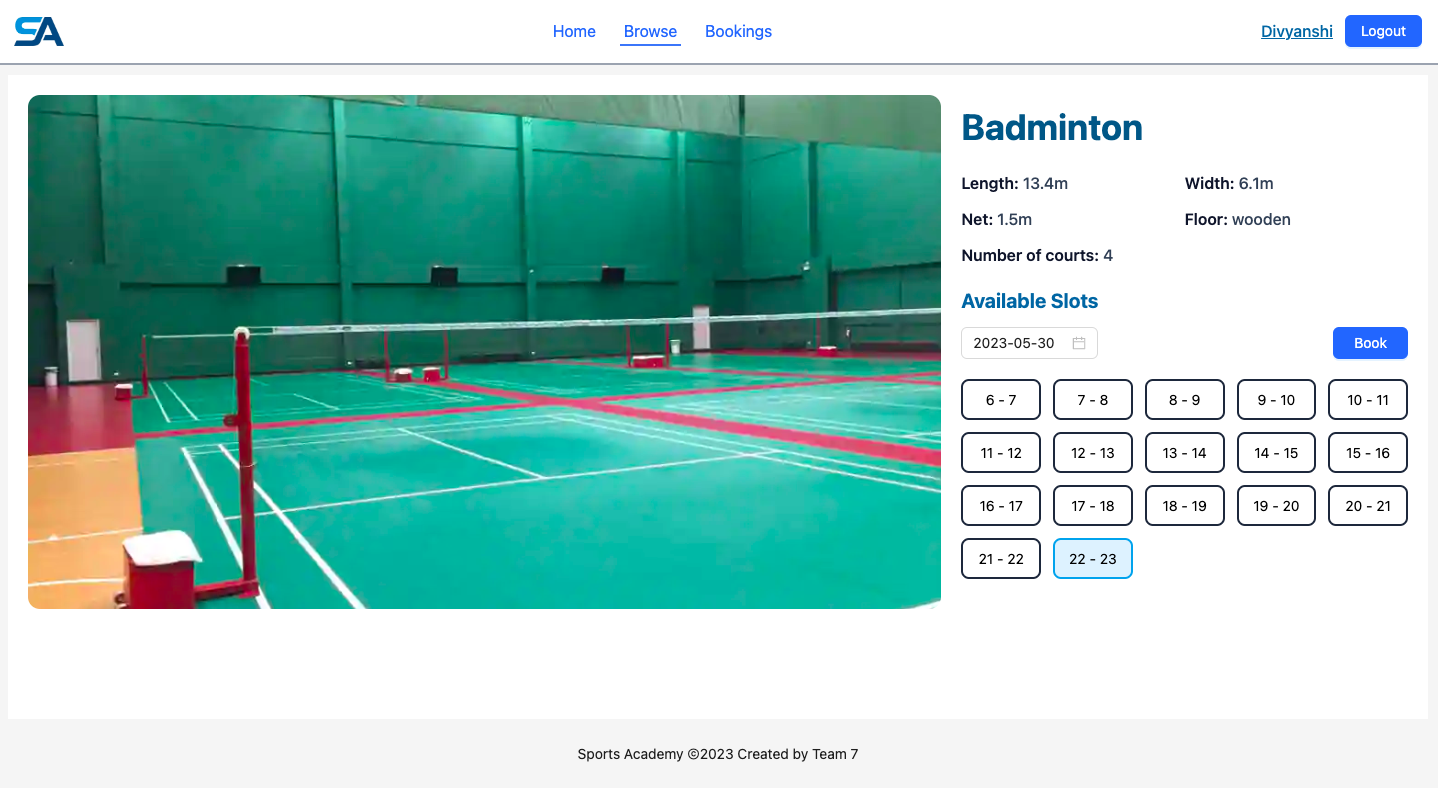
In conclusion, the Sports Academy Management System is a web-based application that offers convenient features for administrators and users to manage court bookings, court details, and user information. It provides flexibility in accessing the system from anywhere at any time. The system automates various tasks, reducing human involvement and improving efficiency. While the system has limitations, such as potential downtime and the need for ongoing maintenance, it helps sports academies streamline operations, enhance user experience, and make court bookings more convenient. With future enhancements, the system can further evolve to meet the changing needs of sports academies and provide a robust management solution.

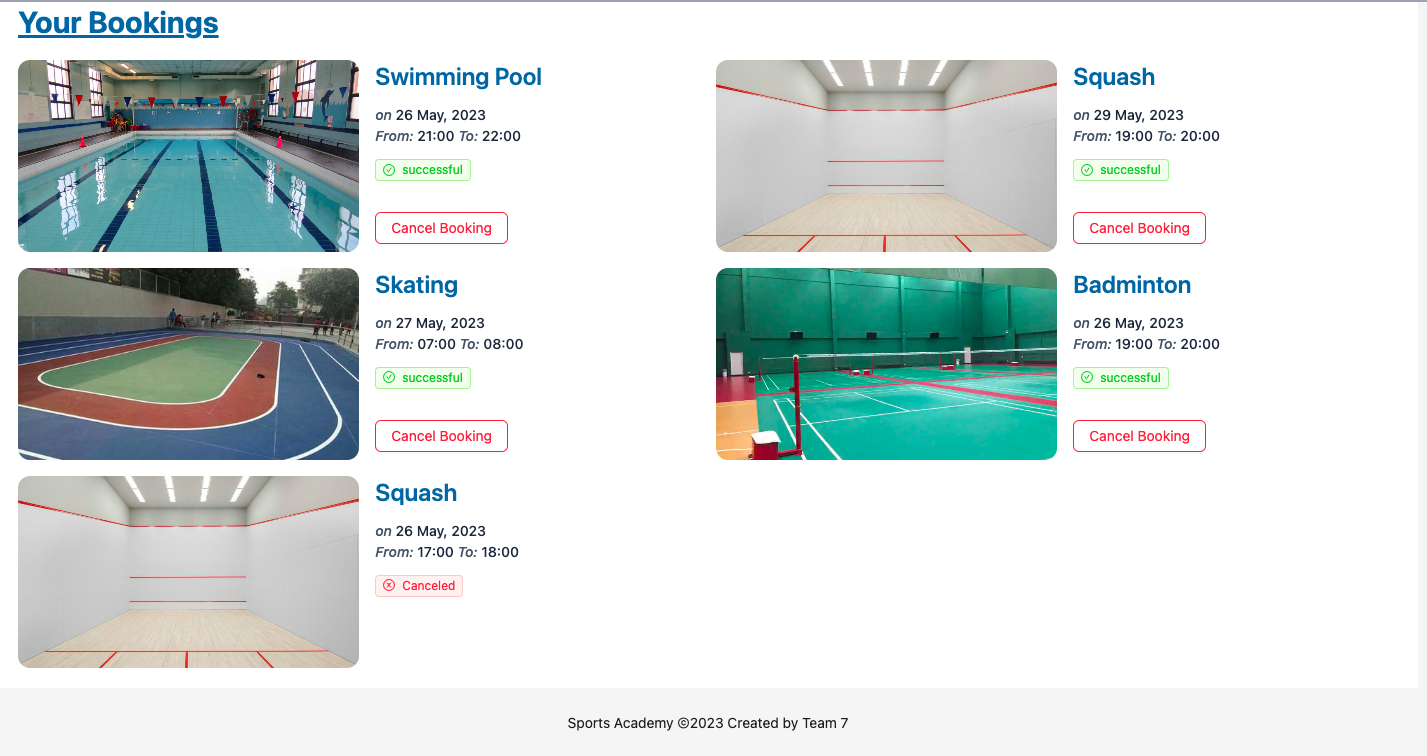
We have gained a lot of practical knowledge from this project, which we think, shall make us stand in a good state in the future.

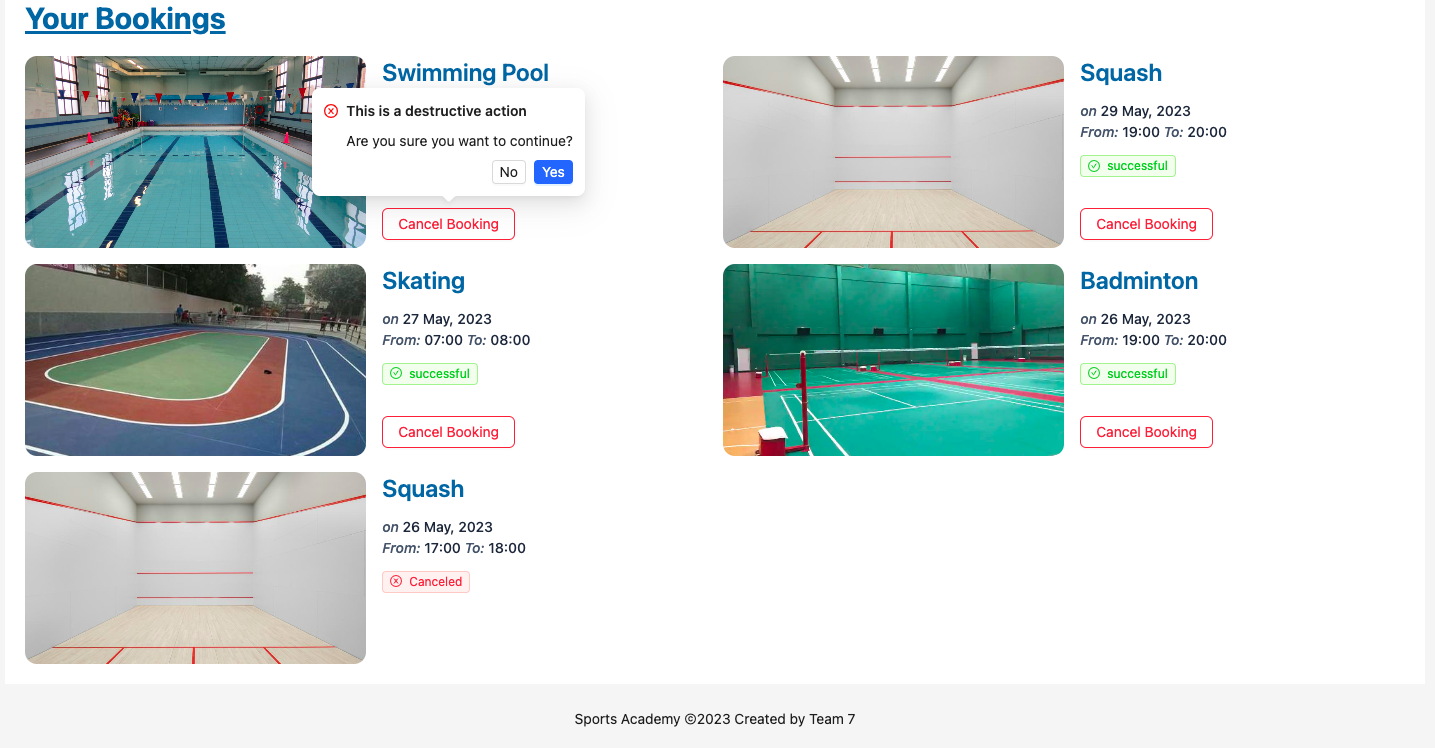
***CHAPTER – VII***

***SCREEN SHOTS***









**A**

***MAJOR PROJECT REPORT***

*on*

**SPORTS ACADEMY**

*Submitted in partial fulfillment of the requirements of*

***CHAPTER – VIII***

***BIBLIOGRAPHY***

Repository link - <https://github.com/dthakurani/sports-academy-backend>

Link of useful websites -

<https://www.freecodecamp.org/learn/back-end-development-and-apis>

[https://medium.com/javarevisited/top-10-courses-to-learn-frontend-and-backend-develop ment-in-2020-710d2d57e008](https://medium.com/javarevisited/top-10-courses-to-learn-frontend-and-backend-develop%20ment-in-2020-710d2d57e008)

[https://www.udemy.com/topic/devops/ https://github.com/getify/You-Dont-Know-JS](https://www.udemy.com/topic/devops/%20https://github.com/getify/You-Dont-Know-JS)