A PROJECT REPORT ON

SportEase

A Sport Management Platform

SUBMITTED IN PARTIAL

FULFILLMENT OF

DIPLOMA IN ADVANCED COMPUTING (PG-DAC)

UNDER THE GUIDENCE OF

Mr. Jaydeep Apte

PRESENTED BY

220940120177 Satyajeet Jadhav

220940120144 Pratik Patil

220940120145 Pravin Jadhav

220940120213 Swati Singh

220940120169 Samarth Patil

220940120136 Prajeet Chougule

AT

CENTER FOR DEVELOPMENT OF ADVANCED COMPUTING

C-DAC, PUNE

ACKNOWLEDGEMENT

The project “SportEase” was a great learning experience for us and we are submitting this work to Advanced Computing Training School (C-DAC ACTS, Pune).

We are very glad to mention the name of Mr. Jaydeep Apte for his valuable guidance to work on this project.

We are highly grateful to Mr. Gaur Sunder, Head of Advanced Computing Training School (ACTS) and Programme for Advancing Computer Education (PACE) of C-DAC for his guidance and support whenever necessary during the course of our journey to acquire PG-Diploma in Advanced Computing (PG-DAC) through CDAC ACTS, Pune.

Our heartfelt thanks go to Mrs. Namrata Ailwar (Course Coordinator, PG-DAC) who gave us all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day of the course.

We would like to express our sincere gratitude towards Mr. Praful Kolte our faculty for Core Java and Mrs. Madhura Anturkar our faculty for Advanced Java, who were always there for us. Their guidance and support helped us overcome various obstacles and intricacies during the course of our project work. Without their tremendous support, guidance, and efforts, this project would not have been possible.

TABLE OF CONTENTS

ABSTRACT

This report discusses the purpose and scope of ‘ SportEase’ it is a Sport Management Platform which are web applications designed to facilitate the booking of sports facilities such as turf, football fields, and cricket fields. The report highlights the benefits of using SportEase booking app, including convenience, transparency, and real-time availability updates. The scope of SportEase booking is discussed in detail, including the ability to search, compare and book sports facilities, as well as additional features such as reviews, real-time availability updates. The report also discusses the scope of ground booking for Academy owners, including the ability to promote their facilities, streamline the booking process, and attract new customers. Overall, SportEase booking offer significant benefits to both users and Academy owners, providing a convenient and efficient platform for booking sports facilities.

INTRODUCTION

Sports Booking Management is a comprehensive software system designed to manage sports facilities and activities. It is used to manage and automate the entire process of sports booking and management. This includes booking slots, managing sports equipment, scheduling activities and events, and managing customer and payment details. The system is designed to streamline the booking process and ensure the efficient and effective management of sports facilities.

Sports Booking Management software can be used by a variety of organizations, such as sports clubs, fitness centers, community centers, schools, and universities. It is an essential tool for managing sports activities and facilities, as it provides a centralized system for managing all aspects of sports activities and facilities.

The system is designed to be user-friendly, with intuitive interfaces and easy-to-use features. It can be customized to meet the specific needs of different organizations and can be accessed from any device with an internet connection. This allows customers to book sports facilities and activities online, making it more convenient and accessible for them.

In summary, Sports Booking Management is a powerful and flexible software system that helps sports organizations manage and automate their sports activities and facilities. With its comprehensive features and user-friendly interface, it is an essential tool for any organization looking to streamline their sports booking and management process.

So, SportEase is a comprehensive Turf Booking Management System developed using advanced Java technologies such as Spring Rest and Spring Boot, with MySQL as the backend database management system and React.js to Demonstrate and learn the concepts of inheritance and use of visual components to Create innovative , creative & interactive web designs to Boost Customer Engagement. The primary objective of the project was to create a platform that simplifies the process of turf booking and management for sports enthusiasts and facility owners.

2. PROJECT VIEW AND SUMMARY

2.1 **PURPOSE**

Our project, “SportEase App”, is a web-based online sport management portal allows users to easily book sports facilities such as turf, football fields, cricket fields, or any other sports facility for a specific date and time. These apps can help users find and book sports facilities in their local area or in other locations, without the need for manual searching or phone calls.

2.2 **SCOPE**

The primary scope of “SportEase App” is to provide users with a platform to search compare and book sports facilities in their local area or in other locations. This app allow users to view available facilities, check pricing and availability, and make bookings based on their preferred date and time. The scope of this app also includes providing a seamless user experience through easy-to-use interfaces, secure payment methods, and real-time updates on booking status. SportEase App offer significant scope for Academy owners by providing them with a platform to promote their facilities, attract new customers, and streamline the booking process.it also includes the ability to create a profile for their Academy, list available time slots, and set pricing based on their preferred rates. By creating a profile, Academy owners can showcase the features and amenities of their facility, such as lighting, parking, and other amenities that may be attractive to potential customers. Turf booking apps can also provide Academy owners with valuable insights into user behaviour, allowing them to optimize their pricing, availability, and other features based on user demand.

2.3 OVERVIEW

A. TECHNOLOGIES USED

i. Front End

- HTML
- CSS
- Java Script
- BootStrap 5.0
- React.js
- Axios
- Ajax
- JQuery

ii. Back End

- Spring MVC
- Spring Boot
- Spring Data JPA
- Spring Security
- Java Mail API

iii. Database Management System

- MySQL

B. Features Provided

I. For Players -

a. Register – Players can register themselves if they want to register.

b. Login – Successfully registered player receive confirmation email From admin and are now eligible to login.

c. View and Update Profile – After successful login, Players can view and Update their profile and password.

d. Booking- Players can book sports facilities such as turf, football fields, cricket fields, and other sports venues according to best suitable price and slot.

e. Searching- Players can search academy by locations, by sports, by academy name

g. Give Reviews – Players can give reviews for academies based on provided facilities.

h. Logout – After utilizing the app, Players can Logout from the app.

II. For Academy Owner –

a. Register – Academy Owner can register them if they want to register their own academy.

b. Login – Successfully registered Academy Owner receives confirmation email from admin and is now eligible to login.

c. View and Update Profile – After successful login, Academy Owner can view and Update their profile and password.

d. Booking- Open bookings and available time slots for customers, and set pricing based on their preferred rates.

2.4 FEASIBILITY STUDY

Feasibility is the determination of whether a project is worth undertaking or not. Before actually recommending the new system, it is important to investigate if it is feasible to develop it.

Before developing and implementing a system, we have to make sure that the system is feasible in the following ways:

A. TECHINICAL FEASIBILITY

In this type of feasibility study, the system analyst has to check whether it is possible or not to develop the requested system with the available manpower, software, hardware, etc.

This project makes use of cross-platform software and solutions like Java, and hence can run on any operating system. JavaScript, used in front-end, is swift and versatile framework when it comes to delivering the requested page. Also, as JavaScript is popular, it is easy to learn it and utilizing it as front end technology. The combination of Spring Boot, Spring Data JPA and Spring Security for backend makes for a fast, easy to set-up and reliable system to interact with the database, as they are secure and transactional in nature. Since the sensitive data of customers and admins need to be stored in a robust and secure database, MySQL database management system was chosen as it is an industry standard.

B. OPERATIONAL FEASIBILITY

In this type of feasibility study, the operation of the system is considered. An analysis is performed on whether it is feasible for the user department to use the application. Thus, the proposed system is said to be operationally feasible only if clients are able to understand the system clearly and correctly, and can use it with ease.

In the design of this project, we always kept user experience in mind. We made an effort to have a good user interface with consistent theme and alluring design to keep the users interested and engaged. In our project, the use of universally known icons and instructions that are easy to understand makes sure that the user will not need any special technical know-how to use the application. We made sure that the information available throughout the application is arranged in a logically coherent and consistent manner, guaranteeing that the users will have a smooth and effortless experience and even enjoy using the application.

C. **ECONOMIC FEASIBILITY**

In this type of feasibility study, the benefits of the system to the organization are considered by taking into consideration the cost-benefit analysis. All the software and technologies used in our project free, open-source, and widely available, with each of the technologies having an extensive community support. This makes “SPORTEASE APP” an economically feasible solution to the organizations that wish to implement it.

3. REQUIREMENTS FULFILLED

3.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements fulfilled by our project:

- Player will be able to login to the system from any device.
- Player will be able to register their profile.
- Player will be able to update their profiles from time to time.
- Player will be able to find sports Academy on the basis of city.
- Player can see academy photos , and extra information provided by the academy.
- Player Can Choose the Academy which are registered on our platform, then he/she can book the ground/turf slots according to availability of booking slots.
- Academy will be able provide booking slots.
- Academy will be able to generate booking availability.
- Academy will be able to advertise their events throughout the system.
- Academy provide the booking invoice to each player for their booking

3.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements fulfilled by our project:

- Since the application uses lightweight and established software components that are also cross-platform, it is remarkably performant and has good support for every operating system.

- The use of JavaScript and JQuery for front end and Spring Boot, Spring Data JPA and Spring Security for back end delivers quick response times to admins and students.

- Card-style UI and well-known icons and symbols used throughout the application provides a consistent theme and user-friendly interface that anyone can grasp easily, even without a technical background.

4. PROJECT DESIGN

4.1. DATA MODEL

The following tables depict the database design used for “SPORTEASE APP” application:

1.User Table

```
mysql> desc users;
+----------+-------------+------+-----+---------+----------------+
| Field    | Type        | Null | Key | Default | Extra          |
+----------+-------------+------+-----+---------+----------------+
| id       | bigint      | NO   | PRI | NULL    | auto_increment |
| email    | varchar(50) | YES  | UNI | NULL    |                |
| password | varchar(20) | NO   |     | NULL    |                |
| type     | varchar(12) | YES  |     | NULL    |                |
+----------+-------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
```

2.Academy Owner Table

```
mysql> desc academy_owner;
+---------------+-------------+------+-----+---------+----------------+
| Field         | Type        | Null | Key | Default | Extra          |
+---------------+-------------+------+-----+---------+----------------+
| id            | bigint      | NO   | PRI | NULL    | auto_increment |
| aadhar_no     | char(14)    | NO   | UNI | NULL    |                |
| email         | varchar(50) | YES  | UNI | NULL    |                |
| gender        | varchar(6)  | YES  |     | NULL    |                |
| mobile_no     | char(10)    | NO   | UNI | NULL    |                |
| owner_address | varchar(100)| YES  |     | NULL    |                |
| owner_name    | varchar(50) | NO   |     | NULL    |                |
| owner_photo   | varchar(50) | YES  |     | NULL    |                |
| reg_date      | date        | YES  |     | NULL    |                |
| type          | varchar(12) | YES  |     | NULL    |                |
+---------------+-------------+------+-----+---------+----------------+
10 rows in set (0.00 sec)
```

3.Academy Table

```
mysql> desc academy;
+-----------------+-------------+------+-----+---------+----------------+
| Field           | Type        | Null | Key | Default | Extra          |
+-----------------+-------------+------+-----+---------+----------------+
| id              | bigint      | NO   | PRI | NULL    | auto_increment |
| academy_name    | varchar(50) | NO   |     | NULL    |                |
| address         | varchar(100)| NO   |     | NULL    |                |
| city            | varchar(100)| NO   |     | NULL    |                |
| contact_details | char(10)    | NO   | UNI | NULL    |                |
| description     | varchar(500)| NO   |     | NULL    |                |
| location        | varchar(30) | NO   |     | NULL    |                |
| no_of_sports    | int         | NO   |     | NULL    |                |
| owner_id        | bigint      | YES  | MUL | NULL    |                |
+-----------------+-------------+------+-----+---------+----------------+
9 rows in set (0.01 sec)
```

4.Academy Photos Table

```
mysql> desc academy_photos;
+------------+---------------+------+-----+---------+----------------+
| Field      | Type          | Null | Key | Default | Extra          |
+------------+---------------+------+-----+---------+----------------+
| id         | bigint        | NO   | PRI | NULL    | auto_increment |
| photo      | varchar(1000) | NO   |     | NULL    |                |
| academy_id | bigint        | YES  | MUL | NULL    |                |
+------------+---------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)
```

5. Player Table

```
mysql> desc player;
+-----------------+------------+------+-----+---------+----------------+
| Field           | Type       | Null | Key | Default | Extra          |
+-----------------+------------+------+-----+---------+----------------+
| id              | bigint     | NO   | PRI | NULL    | auto_increment |
| player_aadharno | char(12)   | NO   | UNI | NULL    |                |
| player_email    | varchar(50)| NO   |     | NULL    |                |
| player_gender   | varchar(6) | NO   |     | NULL    |                |
| player_mobno    | char(10)   | NO   | UNI | NULL    |                |
| player_name     | varchar(50)| NO   |     | NULL    |                |
| type            | varchar(12)| YES  |     | NULL    |                |
+-----------------+------------+------+-----+---------+----------------+
7 rows in set (0.01 sec)
```

6. Booking Table

```
mysql> desc booking;
+--------------+--------------+------+-----+---------+----------------+
| Field        | Type         | Null | Key | Default | Extra          |
+--------------+--------------+------+-----+---------+----------------+
| id           | bigint       | NO   | PRI | NULL    | auto_increment |
| booking_code | varchar(255) | YES  |     | NULL    |                |
| academy_id   | bigint       | YES  | MUL | NULL    |                |
| player_id    | bigint       | YES  | MUL | NULL    |                |
| slot_id      | bigint       | YES  | MUL | NULL    |                |
+--------------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

7. Ground Table

```
mysql> desc ground;
+--------------------+------------+------+-----+---------+----------------+
| Field              | Type       | Null | Key | Default | Extra          |
+--------------------+------------+------+-----+---------+----------------+
| id                 | bigint     | NO   | PRI | NULL    | auto_increment |
| ground_desc        | varchar(10)| NO   |     | NULL    |                |
| equipment_available| bit(1)     | NO   |     | NULL    |                |
| academy_id         | bigint     | YES  | MUL | NULL    |                |
| sport_id           | bigint     | YES  | MUL | NULL    |                |
+--------------------+------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

8. Sports Table

```
mysql> desc sports;
+--------------+-------------+------+-----+---------+----------------+
| Field        | Type        | Null | Key | Default | Extra          |
+--------------+-------------+------+-----+---------+----------------+
| id           | bigint      | NO   | PRI | NULL    | auto_increment |
| no_of_ground | bigint      | NO   |     | NULL    |                |
| sports_name  | varchar(20) | YES  |     | NULL    |                |
| academy_id   | bigint      | YES  | MUL | NULL    |                |
+--------------+-------------+------+-----+---------+----------------+
4 rows in set (0.01 sec)
```

9. TimeSlots Table

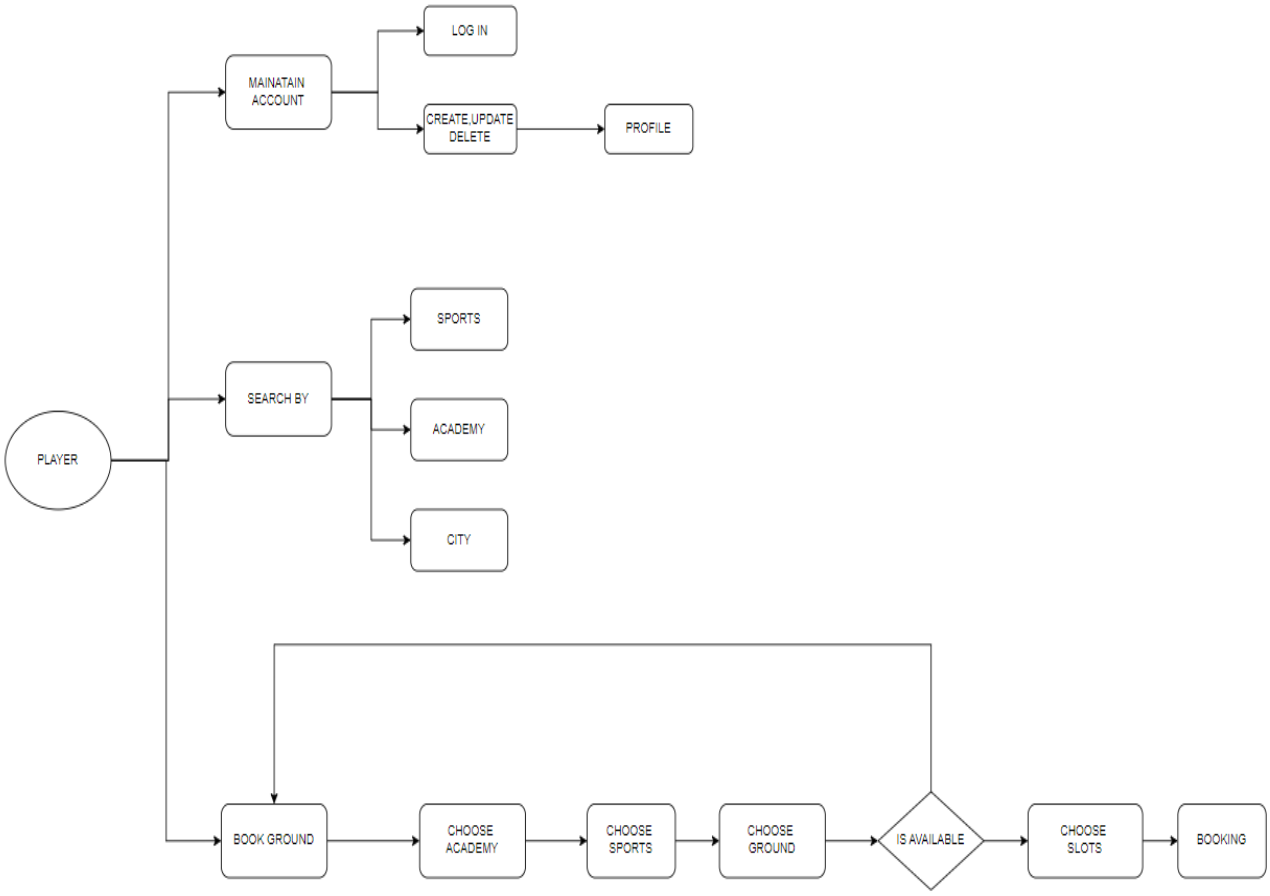
```
mysql> desc time_slot;
+-------------------+--------+------+-----+---------+----------------+
| Field             | Type   | Null | Key | Default | Extra          |
+-------------------+--------+------+-----+---------+----------------+
| id                | bigint | NO   | PRI | NULL    | auto_increment |
| availability_date | date   | YES  |     | NULL    |                |
| available         | bit(1) | NO   |     | NULL    |                |
| rent              | int    | NO   |     | NULL    |                |
| slot_end_time     | time   | NO   |     | NULL    |                |
| slot_start_time   | time   | NO   |     | NULL    |                |
| ground_id         | bigint | YES  | MUL | NULL    |                |
+-------------------+--------+------+-----+---------+----------------+
7 rows in set (0.01 sec)
```

10.Reviews Table

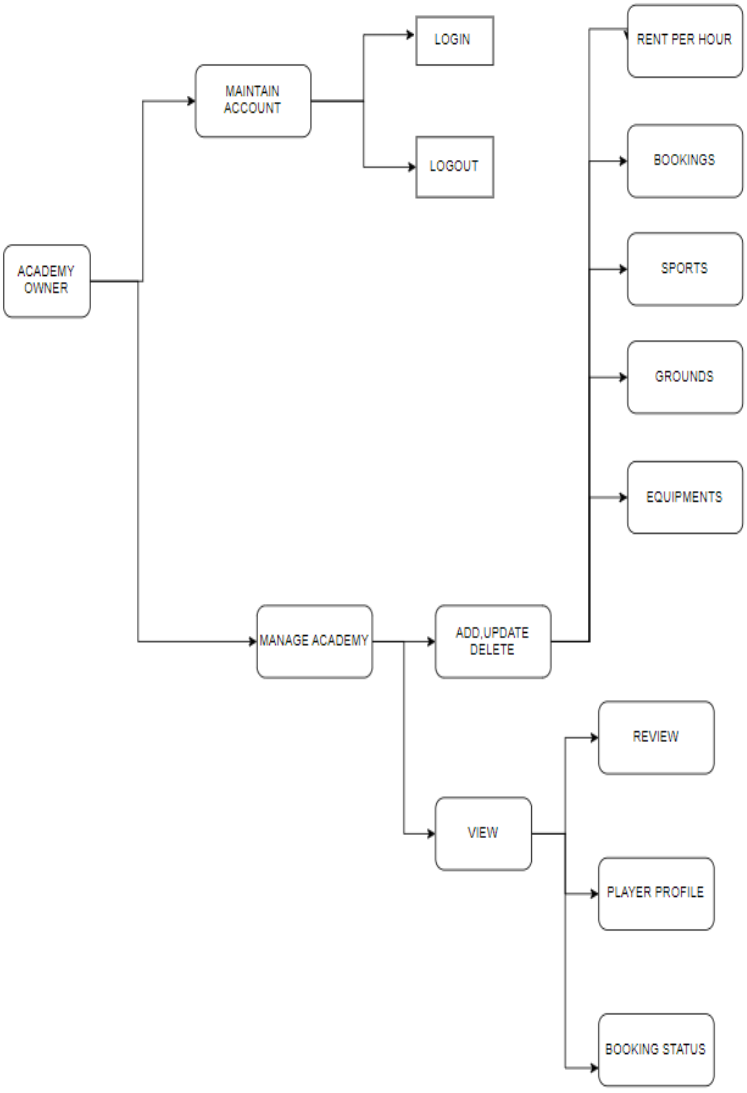
```
mysql> desc reviews;
+-------------+--------------+------+-----+---------+----------------+
| Field       | Type         | Null | Key | Default | Extra          |
+-------------+--------------+------+-----+---------+----------------+
| id          | bigint       | NO   | PRI | NULL    | auto_increment |
| review_date | date         | YES  |     | NULL    |                |
| review_desc | varchar(500) | YES  |     | NULL    |                |
| stars       | int          | YES  |     | NULL    |                |
| academy_id  | bigint       | YES  | MUL | NULL    |                |
| player_id   | bigint       | YES  | MUL | NULL    |                |
+-------------+--------------+------+-----+---------+----------------+
6 rows in set (0.01 sec)
```

4.2. USE CASE DIAGRAM

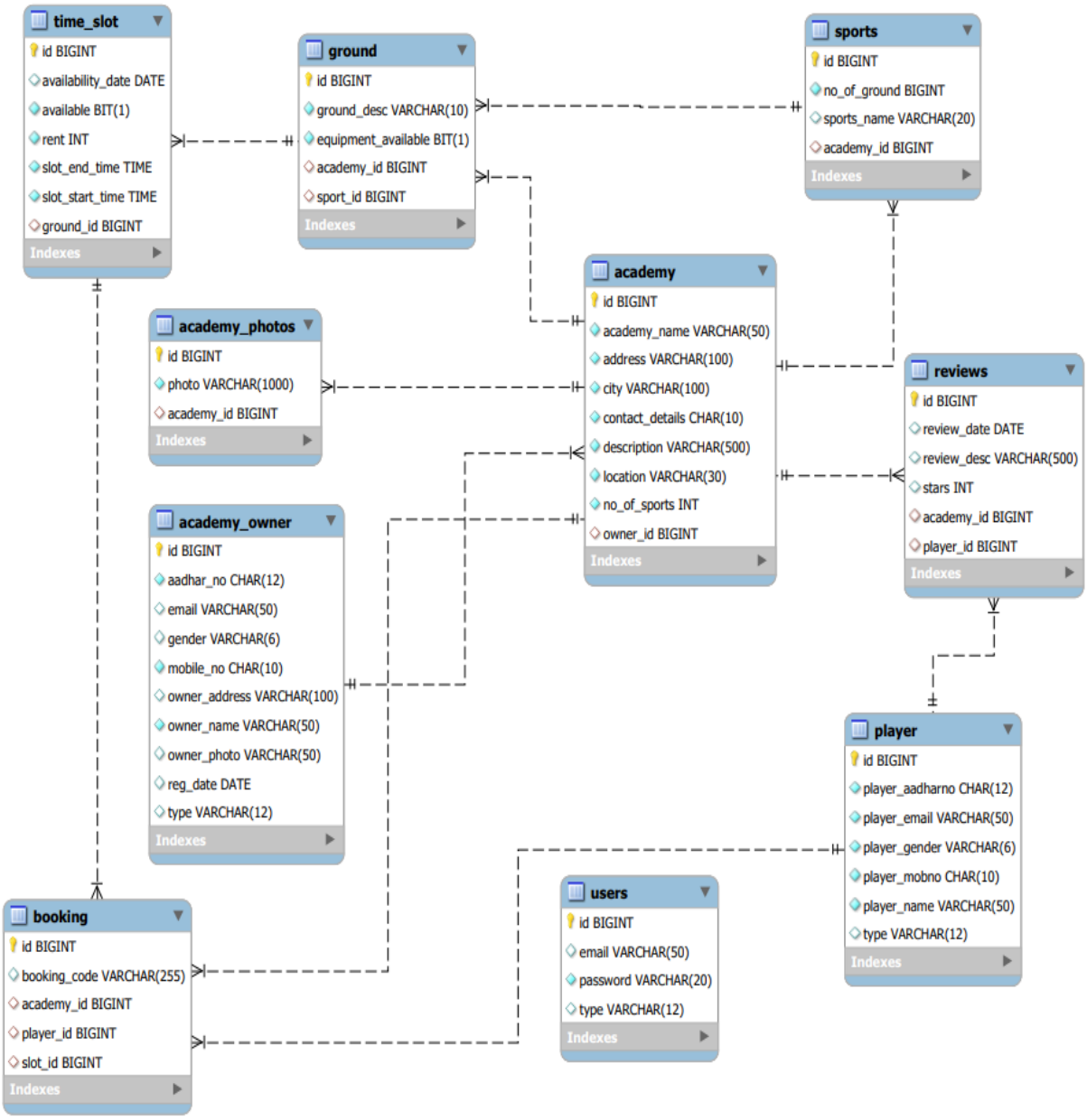
A. Player



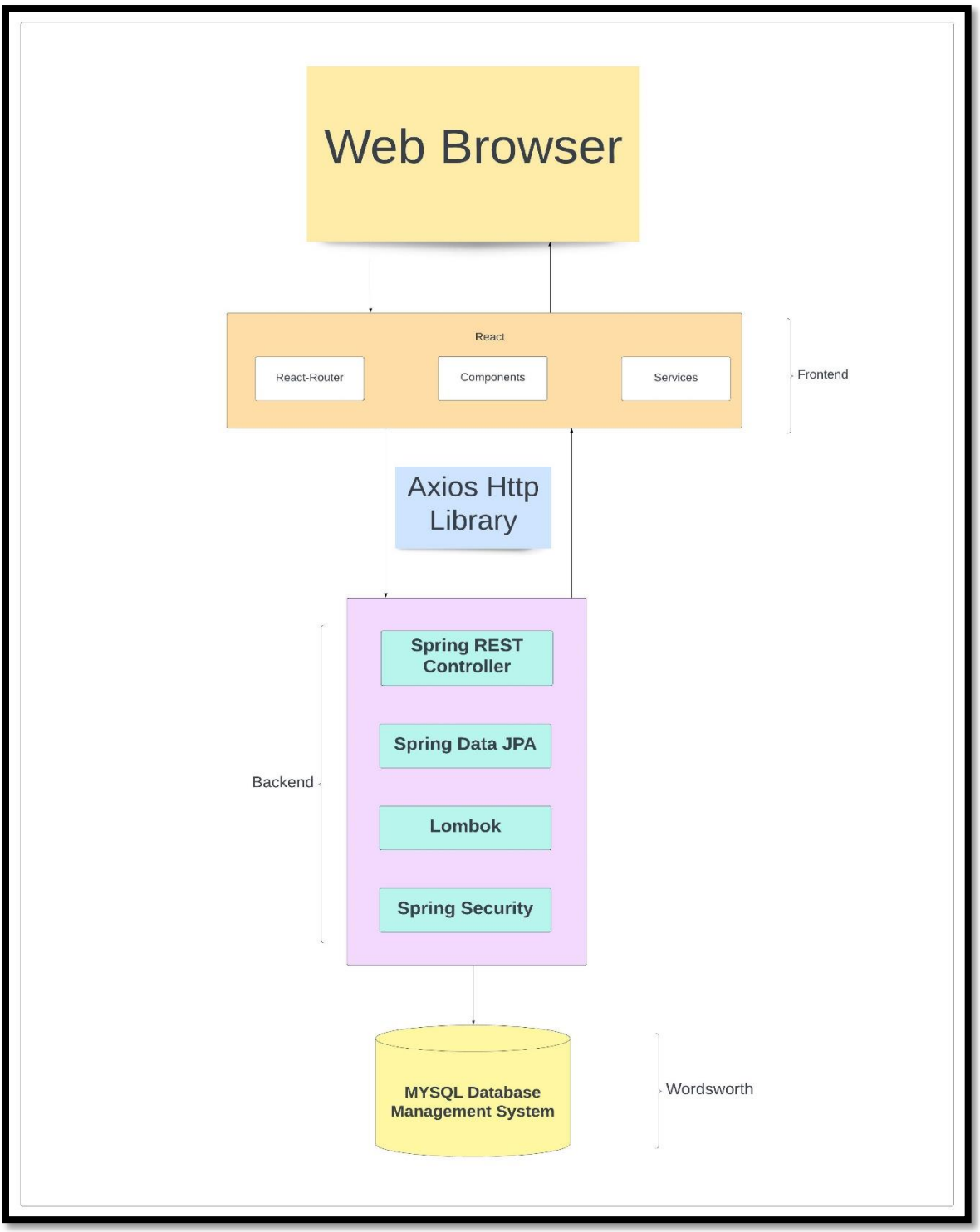
B. Academy Owner



4.3 ER- Diagram-

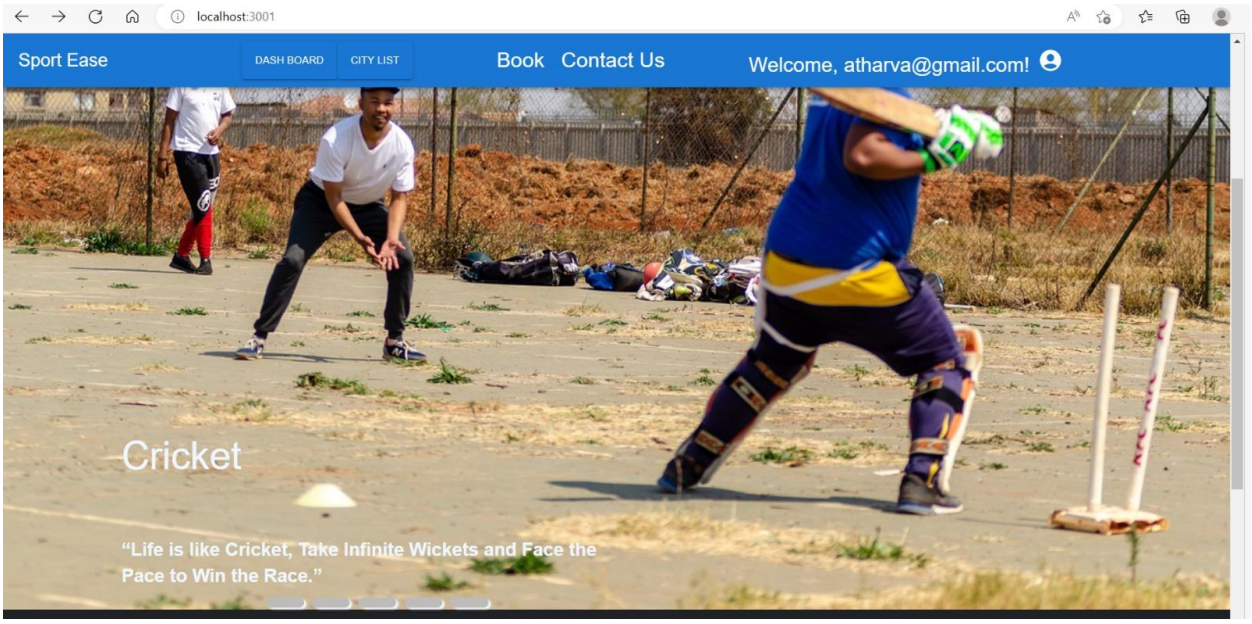


4.4 Project Architecture Diagram

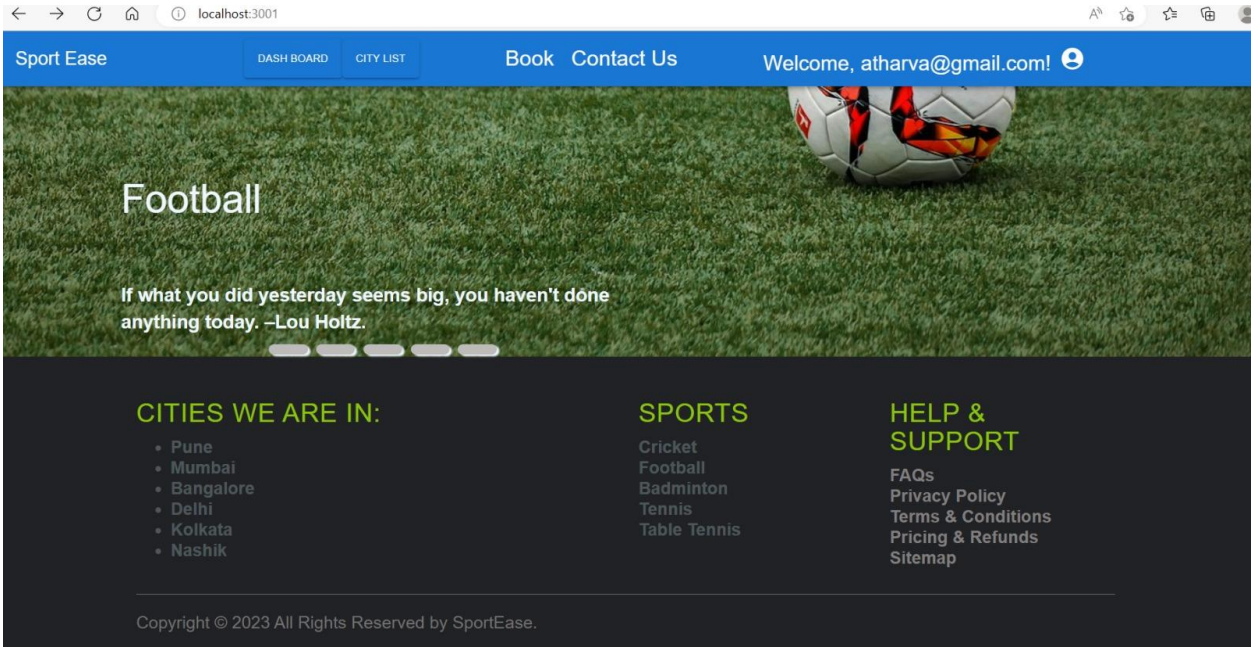


5. PROJECT SCREENSHOTS

1.Home Page



2.Home Page 2



2. After Integration and run Table Created At BackEnd

```
Hibernate: create table academy (id bigint not null auto_increment, academy_name varchar(50) not null, address varchar(100) not null, city varchar(100) not null, contact_details CHAR(1
Hibernate: create table academy_owner (id bigint not null auto_increment, aadhar_no CHAR(14) not null, email varchar(50), gender varchar(6), mobile_no CHAR(10) not null, owner_address v
Hibernate: create table academy_photos (id bigint not null auto_increment, photo varchar(1000) not null, academy_id bigint, primary key (id)) engine=InnoDB
Hibernate: create table booking (id bigint not null auto_increment, booking_code varchar(255), academy_id bigint, player_id bigint, slot_id bigint, primary key (id)) engine=InnoDB
Hibernate: create table ground (id bigint not null auto_increment, ground_desc varchar(10) not null, equipment_available bit not null, academy_id bigint, sport_id bigint, primary key (i
Hibernate: create table player (id bigint not null auto_increment, player_aadharno CHAR(12) not null, player_email varchar(50) not null, player_gender varchar(6) not null, player_mobno
Hibernate: create table reviews (id bigint not null auto_increment, review_date date, review_desc varchar(500), stars integer, academy_id bigint, player_id bigint, primary key (id)) eng
Hibernate: create table sports (id bigint not null auto_increment, no_of_ground bigint not null, sports_name varchar(20), academy_id bigint, primary key (id)) engine=InnoDB
Hibernate: create table time_slot (id bigint not null auto_increment, availability_date date, available bit not null, rent integer not null, slot_end_time time not null, slot_start_time
Hibernate: create table users (id bigint not null auto_increment, email varchar(50), password varchar(20) not null, type varchar(12), primary key (id)) engine=InnoDB
Hibernate: alter table academy drop index UK_dejy38e6ulqg6i1ml17efwjgg
Hibernate: alter table academy add constraint UK_dejy38e6ulqg6i1ml17efwjgg unique (contact_details)
Hibernate: alter table academy_owner drop index UK_ofw5igpat3c4iybhqwkupuf1k
Hibernate: alter table academy_owner add constraint UK_ofw5igpat3c4iybhqwkupuf1k unique (aadhar_no)
Hibernate: alter table academy_owner drop index UK_h9iy1riv2b36c5ik7ecbtq1mn
Hibernate: alter table academy_owner add constraint UK_h9iy1riv2b36c5ik7ecbtq1mn unique (email)
Hibernate: alter table academy_owner drop index UK_6gpy7ijdyme0jqd5p4meq83ub
Hibernate: alter table academy_owner add constraint UK_6gpy7ijdyme0jqd5p4meq83ub unique (mobile_no)
Hibernate: alter table player drop index UK_li4f268171il8v41ffx198w9m
Hibernate: alter table player add constraint UK_li4f268171il8v41ffx198w9m unique (player_aadharno)
Hibernate: alter table player drop index UK_g74nkwrsf7uefcsoi79grsfgd
Hibernate: alter table player add constraint UK_g74nkwrsf7uefcsoi79grsfgd unique (player_mobno)
Hibernate: alter table users drop index UK_6dotkott2kjsp8vw4d0m25fb7
Hibernate: alter table users add constraint UK_6dotkott2kjsp8vw4d0m25fb7 unique (email)
Hibernate: alter table academy add constraint FKb71773c6546pkxkayht7bb09x foreign key (owner_id) references academy_owner (id)
Hibernate: alter table academy_photos add constraint FKai7692aojuyrfx3t8xt6wxehq foreign key (academy_id) references academy (id)
Hibernate: alter table booking add constraint FKl5axm3rpb2vyagldrjv42lsq foreign key (academy_id) references academy (id)
Hibernate: alter table booking add constraint FK9ipxppfolovye3luie3h30ln5 foreign key (player_id) references player (id)
Hibernate: alter table booking add constraint FKss0h2aq4ggglgwylpr7blbec6 foreign key (slot_id) references time_slot (id)
Hibernate: alter table ground add constraint FKl6njijkuac9pvat22tq2kllwd foreign key (academy_id) references academy (id)
Hibernate: alter table ground add constraint FKd5jxrrt9ob0s9h66gr85c60w6 foreign key (sport_id) references sports (id)
Hibernate: alter table reviews add constraint FKc865b81wv2ki6pklbtp417u3v foreign key (academy_id) references academy (id)
Hibernate: alter table reviews add constraint FKrvuqdlqkmc0bm3nhvqxm9751i foreign key (player_id) references player (id)
Hibernate: alter table sports add constraint FKeac4r1tf99ysgowumvdmmx2pc foreign key (academy_id) references academy (id)
Hibernate: alter table time_slot add constraint FKi6tuxr5qcg0h979pcb7653133 foreign key (ground_id) references ground (id)
```

3. Table At MySql DataBase

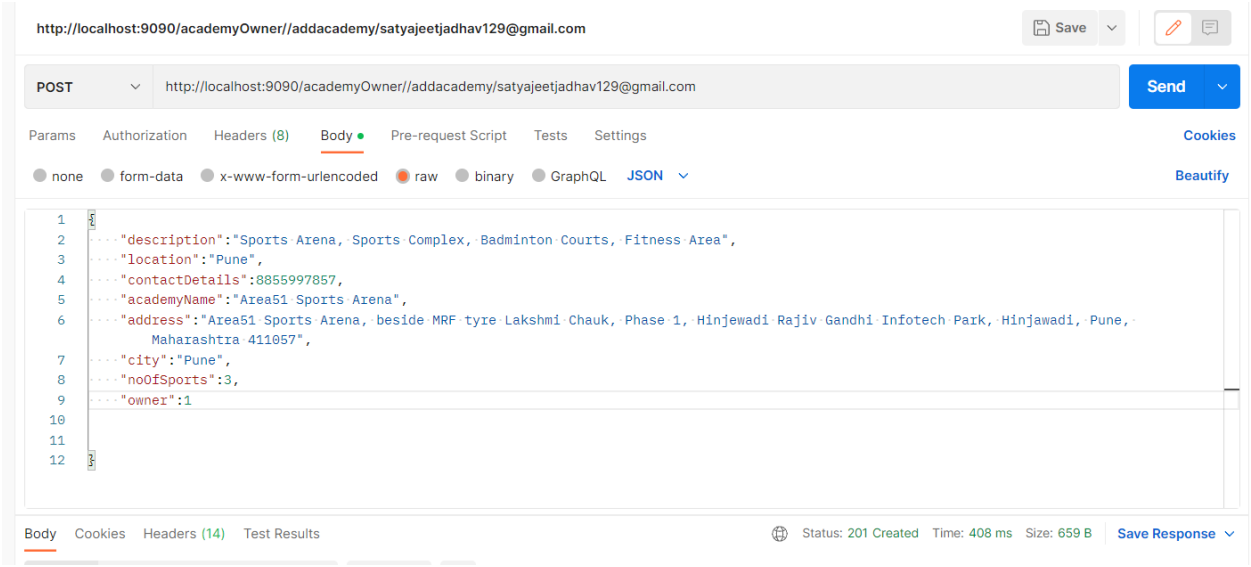
```
mysql> use sports;
Database changed
mysql> show tables;
+------------------+
| Tables_in_sports |
+------------------+
| academy          |
| academy_owner    |
| academy_photos   |
| booking          |
| ground           |
| player           |
| reviews          |
| sports           |
| time_slot        |
| users            |
+------------------+
10 rows in set (0.01 sec)

mysql>
```

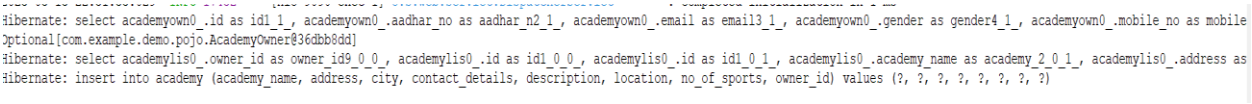
4. Successful Date Insertion from FrontEnd

```
mysql> select * from academy_owner;
+----+--------------+-----------------------------+--------+------------+-----------------------------+------------+-------------+------------+----------
----+
| id | aadhar_no    | email                       | gender | mobile_no  | owner_address               | owner_name | owner_photo | reg_date   | type
   |
+----+--------------+-----------------------------+--------+------------+-----------------------------+------------+-------------+------------+----------
----+
|  1 | 222244448888 | satyajeetjadhav129@gmail.com | MALE   | 8855997857 | Row House No.7,Katraj Kondhwa | Satyajeet  | url         | 2022-12-12 | ROLE_ACAD
EMY |
+----+--------------+-----------------------------+--------+------------+-----------------------------+------------+-------------+------------+----------
----+
```

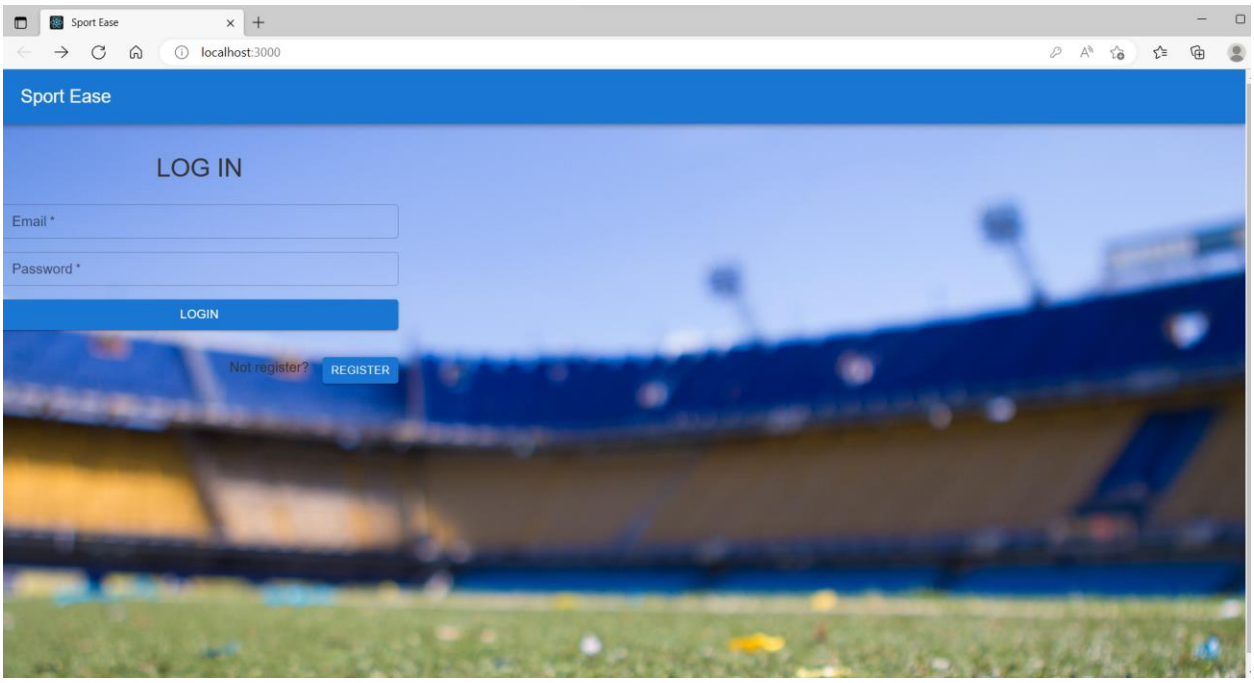
5.Test By Postman



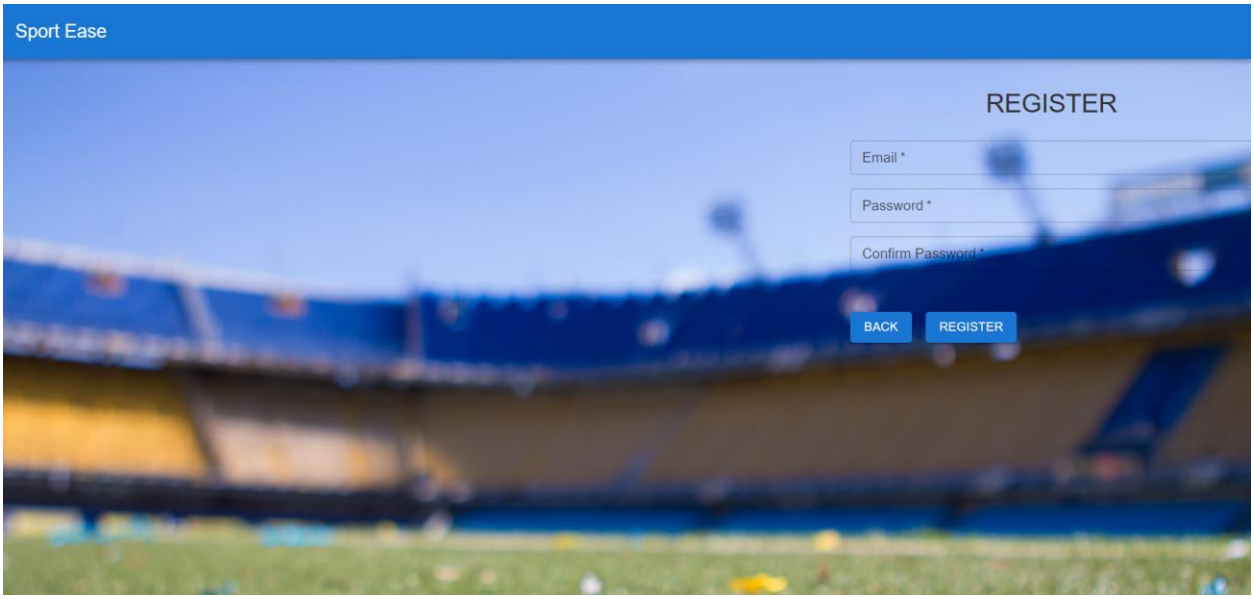
5. Backend Console After Data Insertion From Frontend



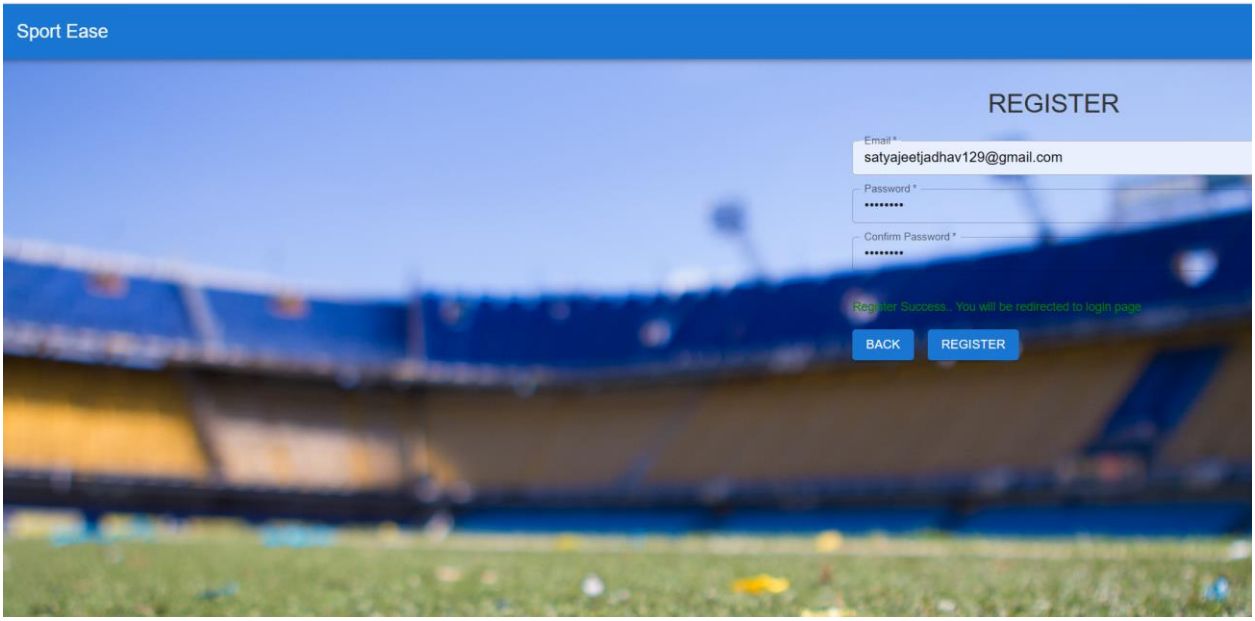
6. Login Page



7. Registration page



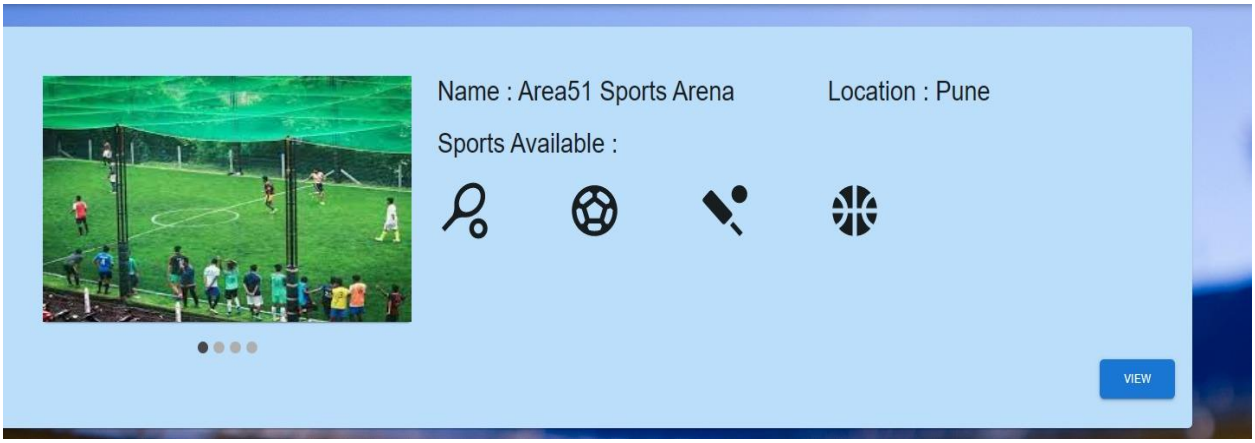
8. Successful Login



9. Token Generation After Login

```
BEFORE: UsernamePasswordAuthenticationToken [Principal=satyajeetjadhav129@gmail.com, Credentials=[PROTECTED], Authenticated=false, Details=null, Grant
Hibernate: select users0_.id as id1_9_, users0_.email as email2_9_, users0_.password as password3_9_, users0_.type as type4_9_ from users users0_ wher
AFTER: UsernamePasswordAuthenticationToken [Principal=org.springframework.security.core.userdetails.User [Username=satyajeetjadhav129@gmail.com, Passw
```

10. Academy List



6. TESTING

One of the main purposes of testing is to validate and verify that the system works as intended. No program or system design is perfect. However, if we implement the system without proper testing, then it may cause problems and lead to a bad user experience.

Testing and checking outcomes of each test gives us the best chance to detect and correct errors before the system is implemented in a production environment.

In the course of our project, we made an effort to manually test each component. In all cases, we obtained the desired results as demonstrated below.

7. CONCLUSION

"SportEase" is a robust online sports booking management system that has been developed by our project team using the latest technologies that are cross-platform and open-source. Our team has put's effort into designing a user-friendly interface to ensure that navigating the application is an easy and seamless experience.

In conclusion, "SportEase" is an ideal choice for sports academies and individuals who wish to streamline their sports booking processes. With its numerous features and visually appealing interface, we will be working on future functionality.

8. FUTURE SCOPE

Sports Booking Management system has a lot of potential for further development and improvement. Here are some possible future scopes for this project:

Integration with more payment gateways: Currently, the system supports only one payment gateway. Integration with more payment gateways like PayPal, Stripe, etc. can be done to provide more payment options for the users.

Real-time updates: The system can be improved to provide real-time updates on the availability of sports slots. This can be achieved by integrating the system with IoT devices that can detect the usage of sports facilities.

Analytics: Incorporating analytics tools to track user behavior and sports facility usage can help in identifying trends and making data-driven decisions. This can help in optimizing the usage of sports facilities and improving the overall user experience.

Social media integration: Integration with social media platforms can help in promoting the Sports Booking Management system and increasing user engagement. Users can share their bookings and reviews on social media, which can help in attracting new users.

9. REFERENCES

Following is the list of websites we referred during the course of our project:

1. https://getbootstrap.com/docs/5.1/getting-started/introduction/

2. https://www.baeldung.com/

3. https://www.w3schools.com/

4. https://docs.spring.io/springdata/jpa/docs/current/reference

5. https://javaee.github.io/javaee-spec/javadocs/

6. https://javadoc.io/doc/org.springframework.data/spring-datajpa/latest/index.html

7. https://mui.com/material-ui/getting-started/overview/