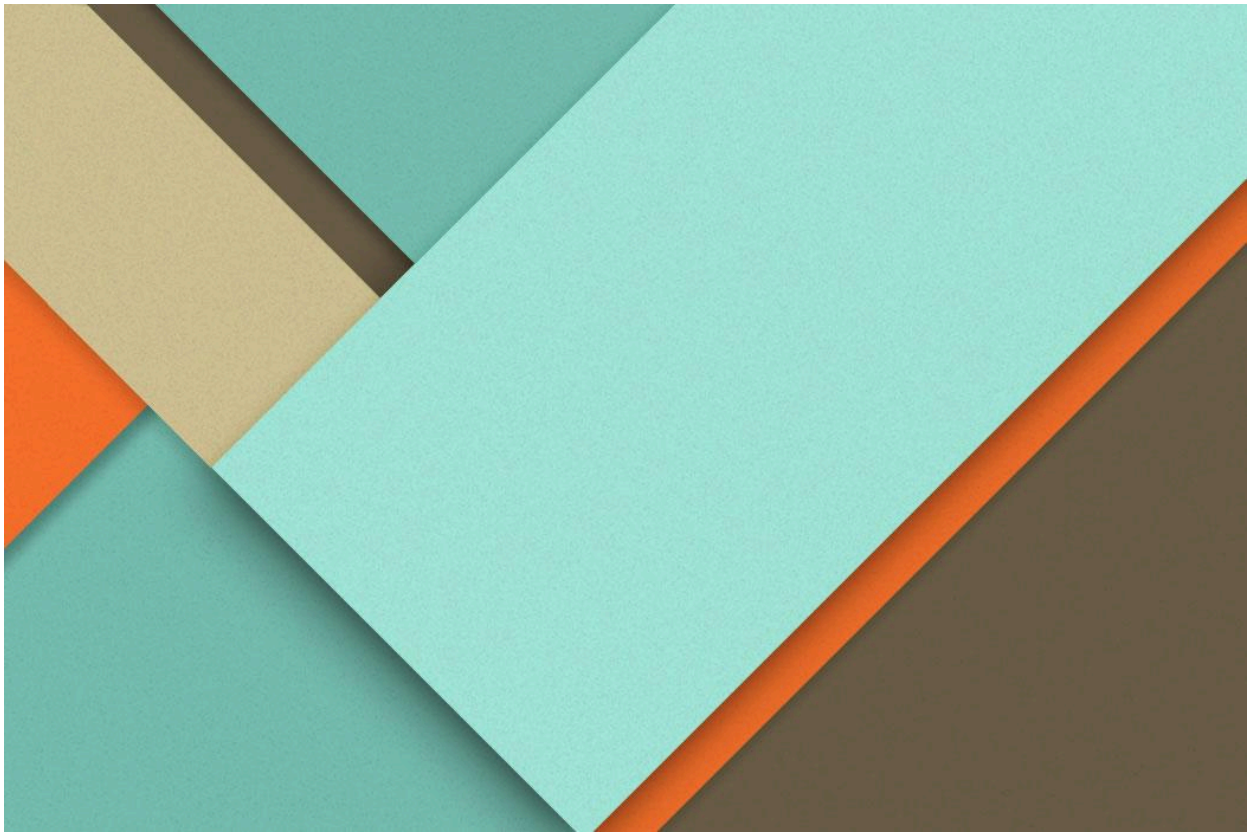


Tennis Timer

Project Explaination

Harikrishnan G
Full Stack Developer || C# || React|| Typescript
harikrishnan14122000@gmail.com
9361513111

Table of Content

Overview

TennisTimer is a Webapp project created using React(Typescript Template) and ASP.NET Core 6.0 API. TennisTimer project showcases my knowledge in concepts like Entity framework, CORS, C#, SQL, Dependency Injection, Role Based authentication, API, handling images in database,React, React, HTML, CSS, Typescript, bootstrap. This project help the user to create a account to search the courts based on location and prebook with dates and slots. User information is protected using role based authentication system.

Technology used in :-

1. ASP.NET CORE 6.0 (API)
2. HTML 5
3. Bootstrap 5.3.2
4. React 18.2.0

Concepts Explaination:-

Entity Framework:-

Entity Framework is a modern object-relation mapper that lets you build a clean, portable, and high-level data access layer with .NET (C#) across a variety of databases, including SQL Database (on-premises and Azure), SQLite, MySQL, PostgreSQL, and Azure Cosmos DB. It supports LINQ queries, change tracking, updates, and schema migrations.

Dependency Injection:-

Dependency injection is used to make a class independent of its dependencies or to create a loosely coupled program. Dependency injection is useful for improving the reusability of code. Likewise, by decoupling the usage of an object, more dependencies can be replaced without needing to change class.

Database(SQL):-

Database for this project is create using SQL language which include seven tables.Three tables stores the data of users like User details, Roles, UserRoles. One table store the Refresh Token. Another three table stores the Court Information, Court Booking Information, Court User Revirw Information.

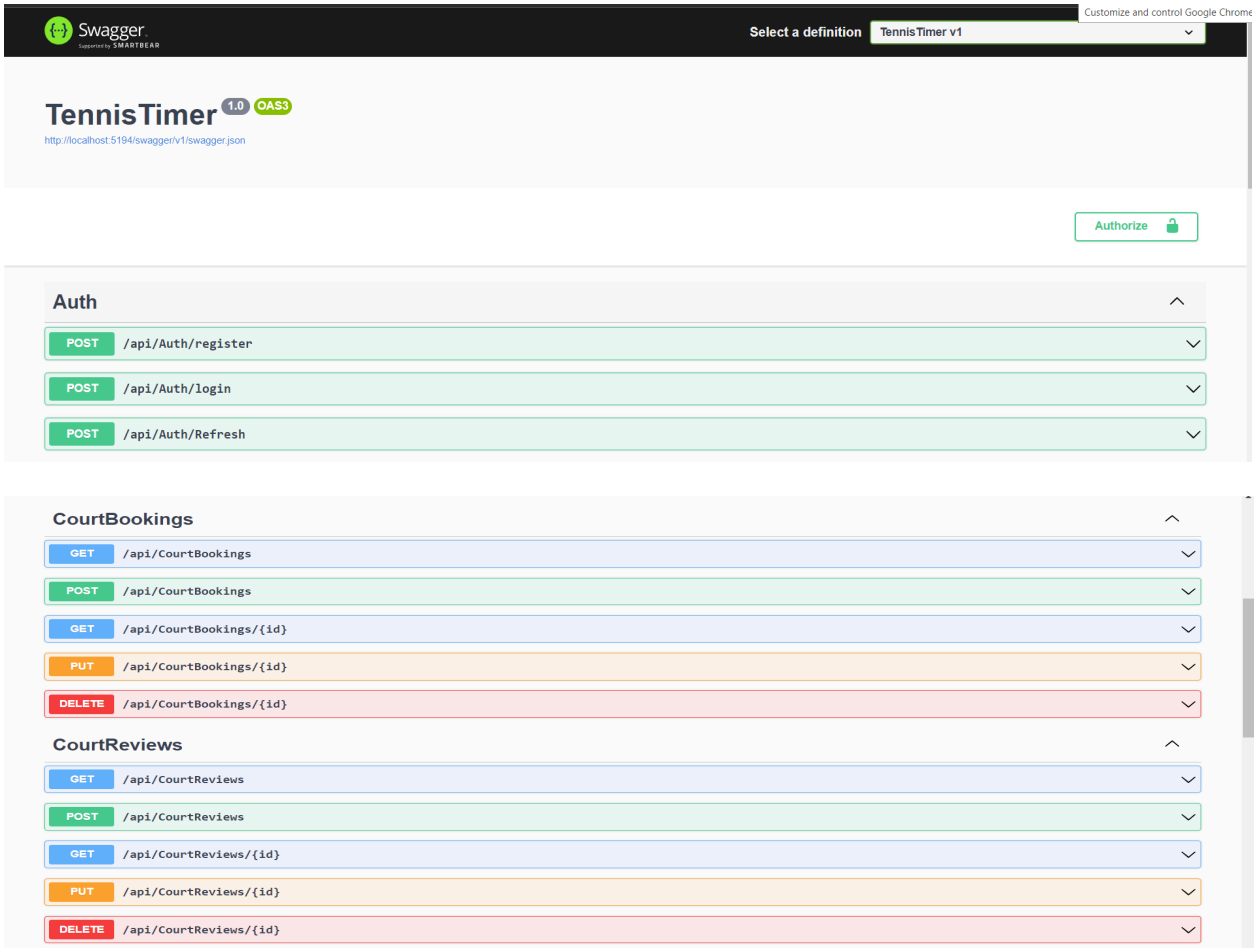
User table is connect with UserRole, Refresh Token, Court Booking and Court Review with ON DELETE CASCADE relation ship.Court Table is connect using Court Booking, Court Review. Below is the image of SQL Code

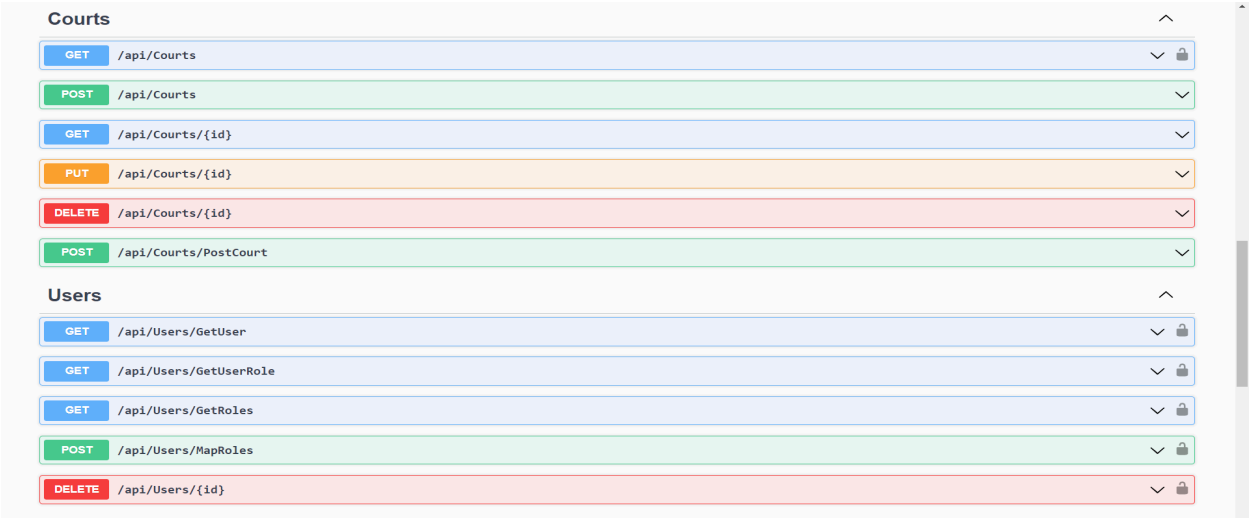
```
Create DATABASE TennisTimer;

Create TABLE Users(
        UserName VARCHAR(20) PRIMARY KEY ,
        PasswordHash VARBINARY(MAX) NOT NULL,
        PasswordSalt VARBINARY(MAX) NOT NULL
);

Create Table Roles(
        RoleId int PRIMARY KEY,
        RoleName Varchar(20) NOT NULL
);

Create Table UserRole(
        UserName VARCHAR(20) PRIMARY KEY,
        RoleId int NOT NULL,
        FOREIGN KEY (UserName) REFERENCES Users(UserName) ON DELETE CASCADE,
        FOREIGN KEY (RoleId) REFERENCES Roles(RoleId) ON DELETE CASCADE
);

Create Table Court (
        CourtId INT IDENTITY(1,1) PRIMARY KEY,
        CourtName VARCHAR(20) NOT NULL,
        CourtLocation VARCHAR(50) NOT NULL,
        CourtImg1 VARBINARY(MAX),
        CourtImg2 VARBINARY(MAX),
        CourtImg3 VARBINARY(MAX)
);

Create Table CourtBooking(
        BookingID INT IDENTITY(1,1) PRIMARY KEY,
        CourtId INT NOT NULL,
        BookingDate DateTime NOT NULL,
        Slot1 INT,
        Slot2 INT,
        Slot3 INT,
        UserName VARCHAR(20) NOT NUll
        FOREIGN KEY (CourtId) REFERENCES Court(CourtId) ON DELETE CASCADE,
        FOREIGN KEY (UserName) REFERENCES Users(UserName) ON DELETE CASCADE
);
```

```
Create Table CourtReview(
        ReviewId INT IDENTITY(1,1) PRIMARY KEY,
        CourtId INT NOT NULL,
        UserName VARCHAR(20) NOT NUll,
        BookingID INT NOT NULL,
        Rating INT
        FOREIGN KEY (CourtId) REFERENCES Court(CourtId) ON DELETE CASCADE,
        FOREIGN KEY (UserName) REFERENCES Users(UserName) ON DELETE CASCADE,
        FOREIGN KEY (BookingID) REFERENCES CourtBooking(BookingID)
);
```

Back-End (ASP.NET.CORE API):-

Backend is created using ASP.NET Core 6.0 API where I implemented Dependency Injection, Entity Framework, Swagger, Role based authentication, Image to byte array function to store images in SQL server. Below is the image of API swagger.



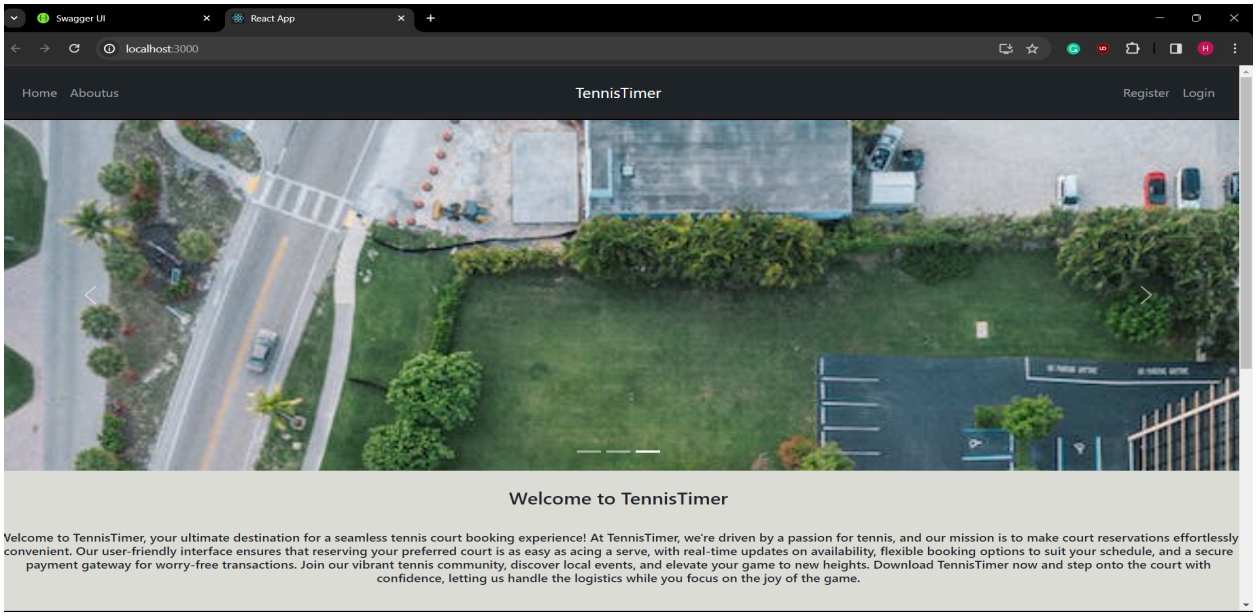


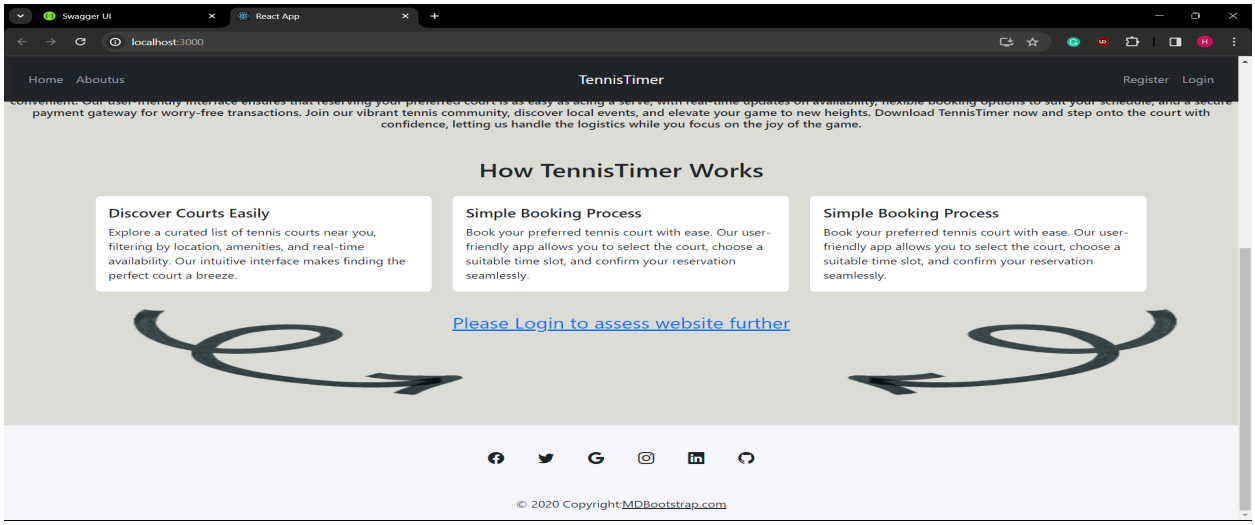
Front-End(React Typescript):-

Front-end of TennisTimer is created using React 18.2.0 with typescript template, it was connected using CORS with ASP.NET CORE API. The flow of the website is given below.

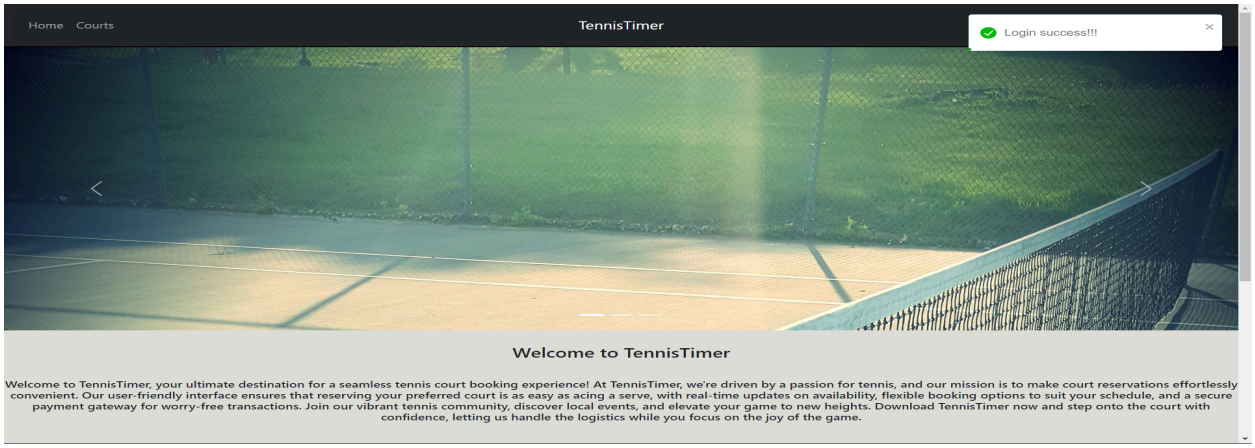
Home Page:-

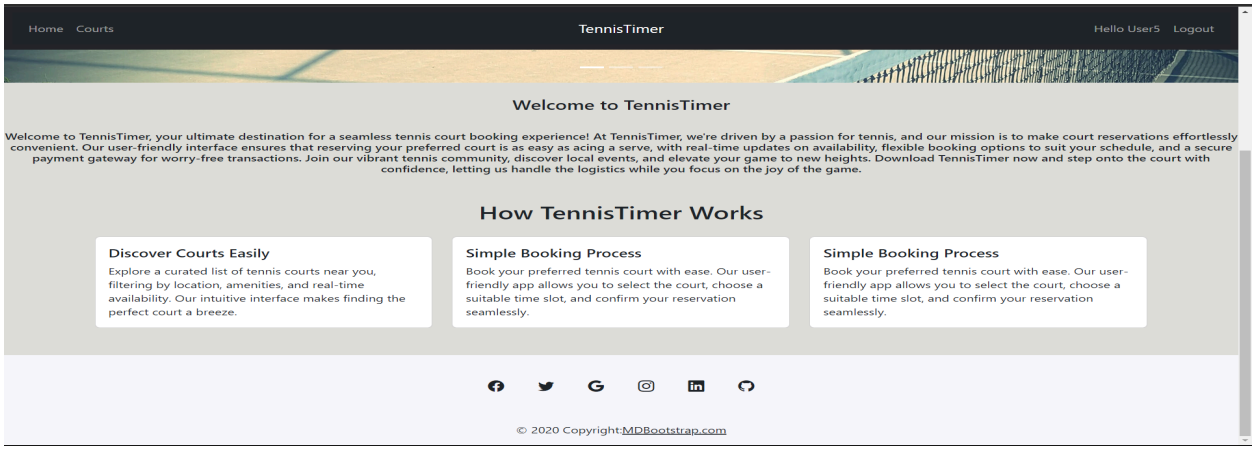
Home pageof the website before login, it has navbar with links for Home page, About Page, Registration Page, Login Page. At bottom of the page If users didn't login it has a message to login to see further information





Once User loggedin message disappear and new links appear in navbar for Courts SearchPage, User Profile page and Logout link

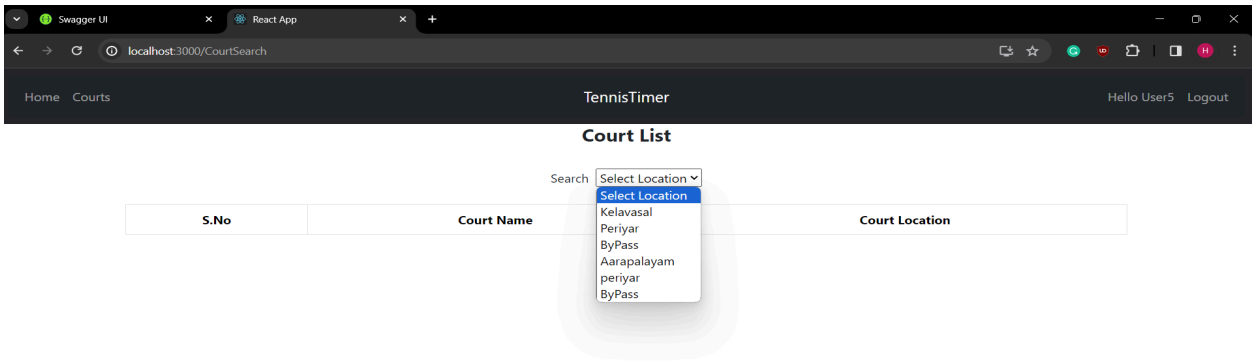


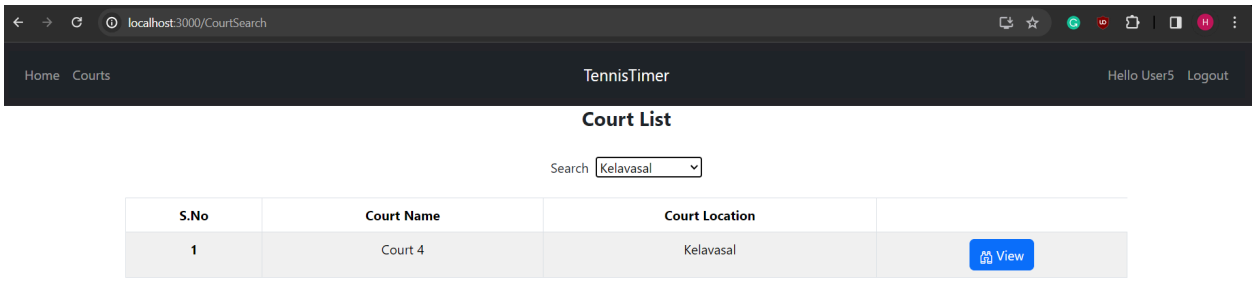


TennisTimer has a fixed position navbar at the top.

Courts Search:-

Court Search page as a table with search window on the top to selet the location, where thay want to book the court. Court Location Data was fetched form the database to display in the dropdown

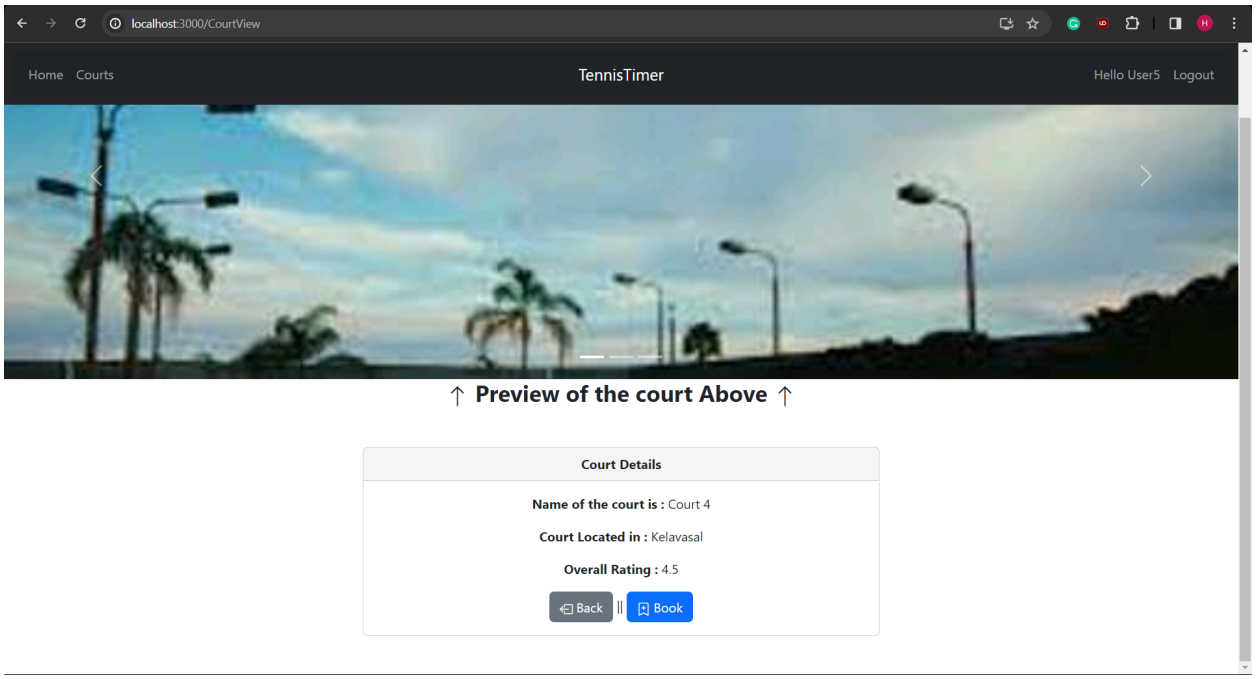




Once location is selected, the table display the courts present in the location.

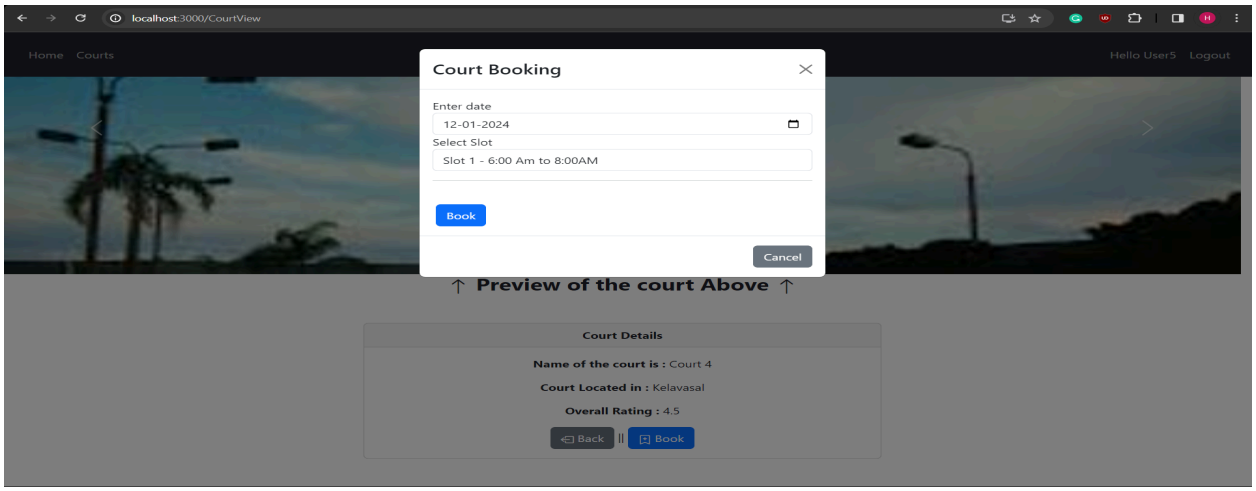
Court View:-

Court View page has the information about the court like name, images of the courts given while register the court, overall rating of the court and Link to book the court.



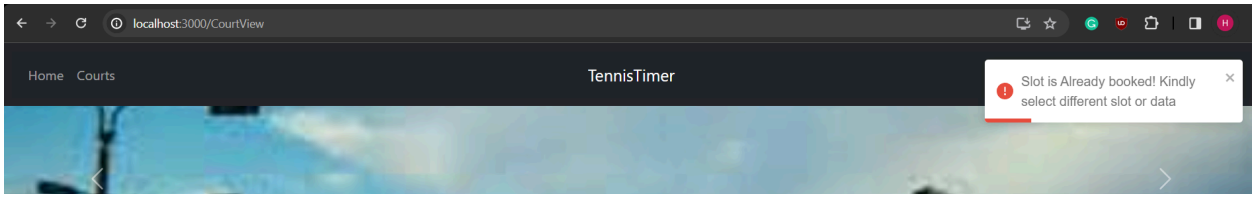
Court Booking:-

If user want to book the court they have to provide the date and slot for the booking purpose.



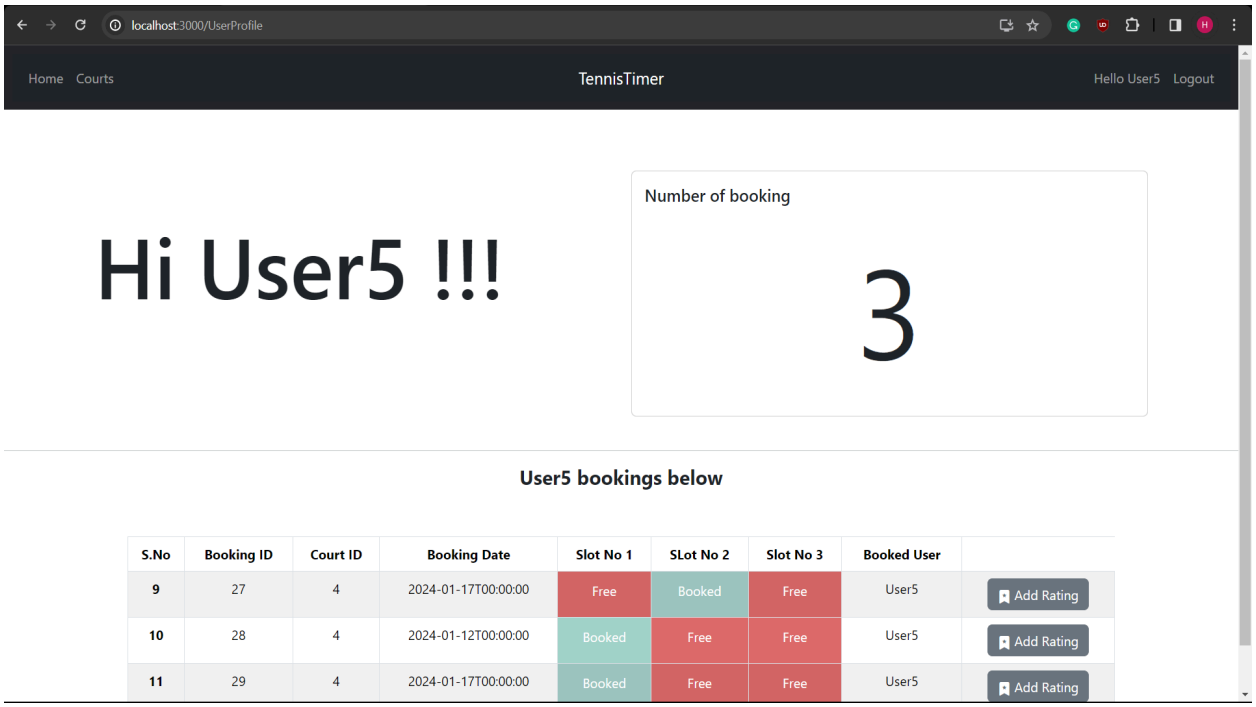
Once user booked the court, they will be navigated to the user profile page which has there booking history

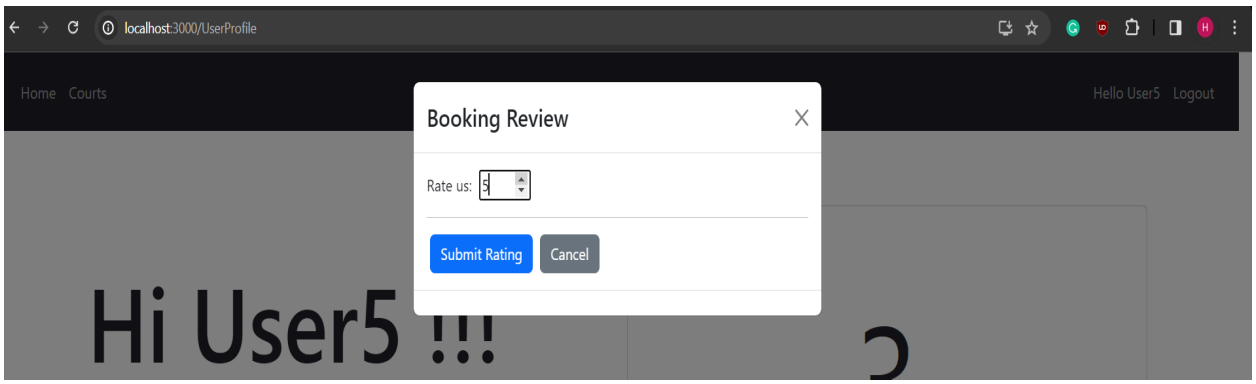
If User booked the slot which was already booked by other user, they will get the notification to book another slot.



User Profile:-

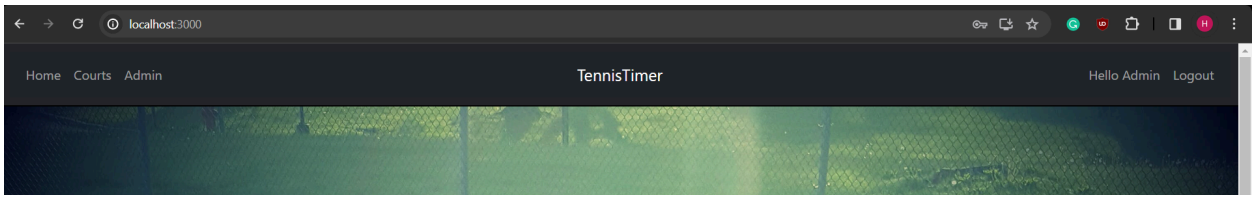
User Profile page has the user booking history and have a link to place reviews on the court they already booked



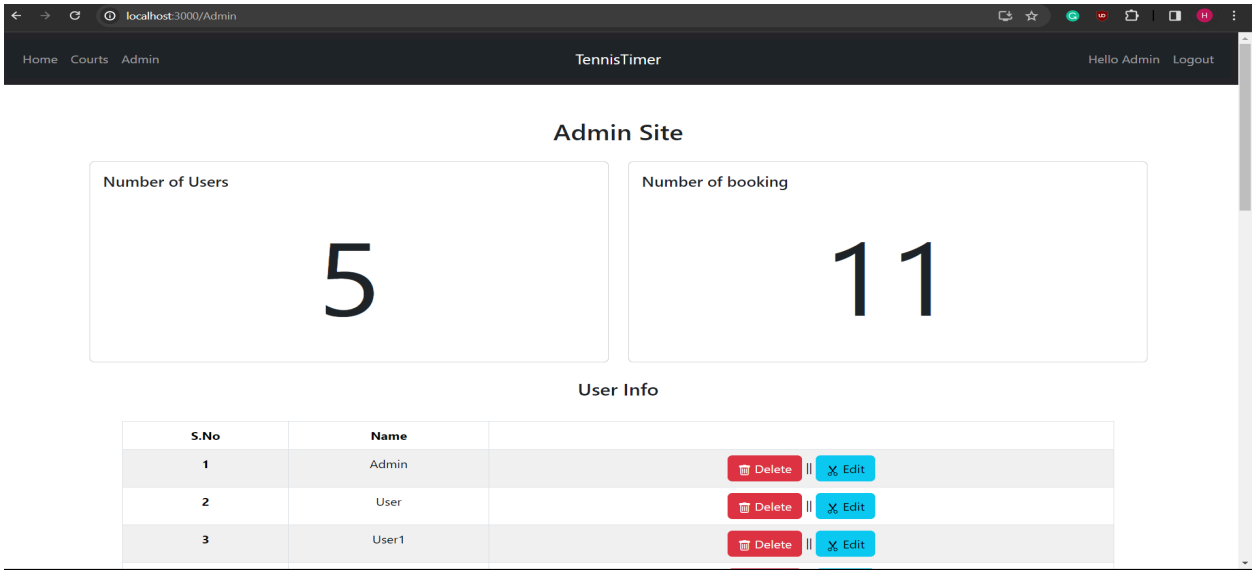


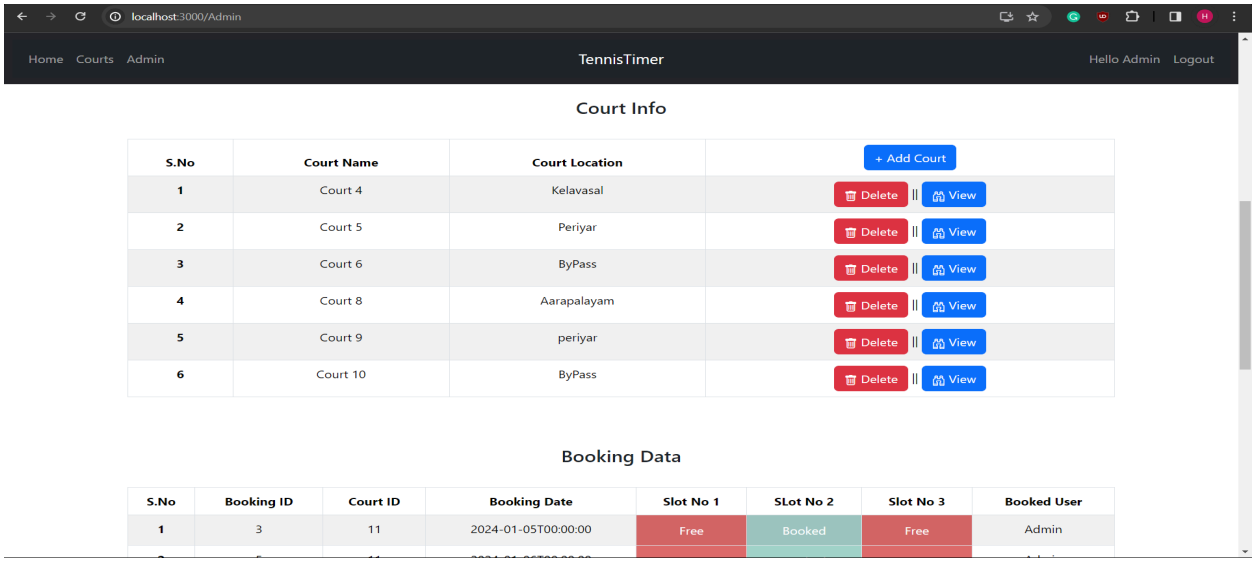
Admin Page:-

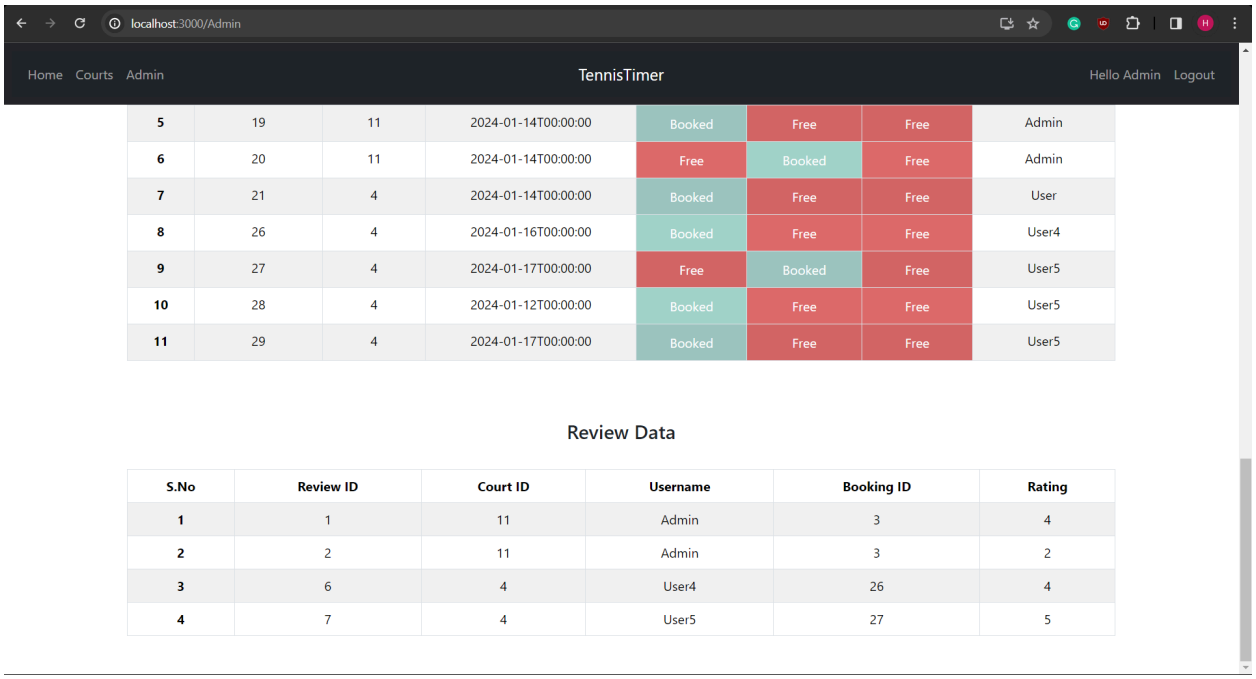
Admin page link was only available to the “Admin” role user account in the navbar.

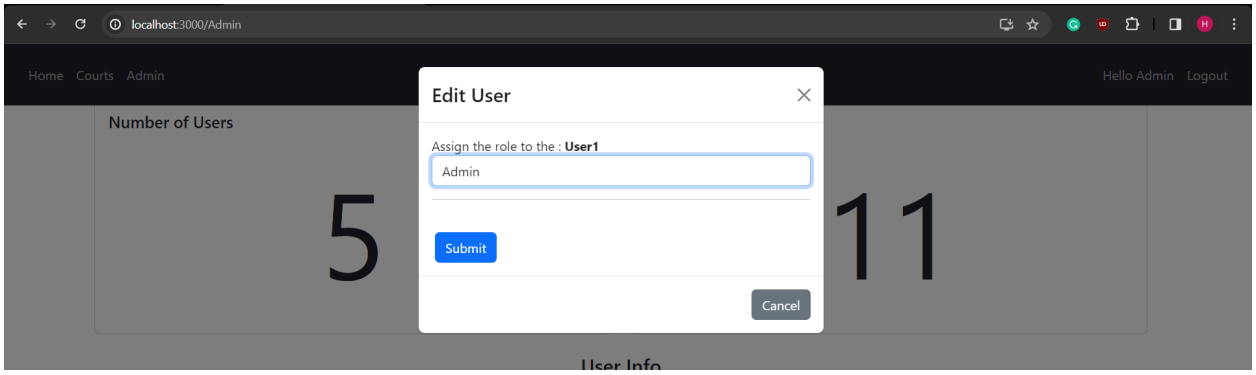


Admin page has the all User Information, User Count Court Information, Court Count, Booking Information, Review Information. Admin can delete the users and Courts, Admin can give another user admin access using UI.









Conclusion:-

This project showcase my skills in above concepts and technology. I am eager to use my knowledge in different project and learn new concepts and teachnology