Sports Booking Website Project Report

Introduction

The sports booking website was developed for a sports technology company to allow its operations team to manage facility bookings across two centers: Indiranagar and Koramangala. The system enables users to reserve courts for various sports, such as badminton, tennis, squash, ping pong, and basketball, across multiple time slots.

Design Decisions

The project design was centered around simplicity, usability, and functionality, with a clean user interface and an efficient backend system to manage bookings.

Database Design: The database was structured into separate tables to manage users, centers, sports, resources (courts/tables), and bookings. This separation allows for scalability and efficient queries. Relationships were established between centers, sports, and resources to ensure that each resource belongs to a specific sport and center.

`centers` table to hold details about the different centers (Indiranagar, Koramangala).
`sports` table to manage the sports available at each center.
`resources` table to manage specific courts or tables within a sport.
`bookings` table to log each booking with respect to the resource, time, date, and user.

User Interface Design: The front end was designed to be simple and intuitive, with colorful elements to make navigation easier. Users are prompted to input their name, date, select a center, and sport, and then choose either to view existing bookings or create a new booking.

- The design makes use of forms with dropdowns for user-friendly navigation.
- Separate pages were created for each step of the booking process (e.g., selecting a sport, viewing bookings, and creating new bookings).

Implementation Details

The website was built using the following technologies:

- **PHP**: The server-side logic was implemented in PHP to interact with the database and handle booking operations. PHP was chosen due to its compatibility with XAMPP and ease of integration with MySQL.
- **MySQL**: A MySQL database was used to store all booking, sport, and resource information. MySQL provides an efficient and scalable solution for handling multiple booking records and user queries.
- **HTML/CSS**: The front end was designed using HTML for structure and CSS for styling. The pages were designed to be clean and colorful with simple navigation to ensure a user-friendly experience.
- **JavaScript**: JavaScript was used for client-side validation and dynamic interactions (such as form submissions and handling dropdown updates based on center selection).
- **XAMPP**: The project was run locally using XAMPP, which provides an Apache server and MySQL database, ideal for development and testing.

Rationale for Technologies: The combination of PHP and MySQL was selected because it is widely supported, integrates well, and runs smoothly on XAMPP, a local environment familiar to many developers. HTML, CSS, and JavaScript were chosen to create an interactive and responsive front end.

Challenges and Solutions

1. Linking Resources to Centers and Sports:

- **Challenge**: Ensuring that resources (courts/tables) are tied to specific sports and centers, which required clear relational database design.
- **Solution**: We added a *sport_id* in the resources table to link each resource to a specific sport. This allowed the system to easily fetch the appropriate resources for a selected sport at a particular center.

2. **Preventing Double Bookings**:

- **Challenge**: Ensuring that users cannot book the same court/resource during the same time slot.
- **Solution**: A query was implemented to check for existing bookings at the specified time and resource before allowing a new booking to be created.

3. User-Friendly Navigation:

- **Challenge**: Creating a multi-step booking process that remains simple and user-friendly.
- **Solution**: The booking process was broken down into smaller steps (select center, select sport, view/create bookings). Separate pages were created for each action, and the design was kept consistent with easy-to-use buttons and dropdowns.

Future Improvements

1. Database Locks for Consistency: In a real-time environment, where multiple users may attempt to book the same slot simultaneously, implementing database locks or transaction management is crucial to maintain data consistency and prevent double bookings. This could be done by using MySQL's transaction features (`START TRANSACTION`, `LOCK TABLES`, etc.) to ensure that booking slots are handled in an atomic operation, preventing race conditions and maintaining integrity across concurrent users.

2. Admin Dashboard: A feature for the operations team to monitor and manage all bookings, view reports on resource usage, and handle cancellations could be added.

3. User Authentication and Profiles: In future iterations, a user login system can be added to allow customers to manage their bookings, view history, and receive notifications about upcoming bookings or availability.

4. Integration with Payment Gateways: The website could be expanded to include payment integration, where users could pay for their bookings online.