**Name: - Anil Acharya**

**Level-6 , Section-B**

**Course: - Advanced Web Engineering**

**ID: - C7356682**

**Project Name – Cricket Hub**

Cricket is one of the most loved sports globally, drawing millions of fans, players, and administrators. Managing teams, tracking player performance, and staying updated with real-time matches can be challenging without a central platform. Cricket-Hub addresses this by offering a powerful web application designed specifically for cricket organizations. It helps administrators manage teams and players easily while providing a live match interface for fans.

The platform ensures data security through role-based access control, granting sensitive permissions only to authorized users. General users enjoy features like live scores and secure profiles. With real-time data APIs, advanced search options, and responsive design, Cricket-Hub boosts efficiency for administrators and keeps users engaged.

Basic Features Covered

* Authentication System (Login, Register, Logout).
* Role-Based Access Control (Admin and Regular User roles).
* CRUD Operations for Teams and Players.
* Pagination for teams and players pages.
* Search functionality to filter players and teams.
* Filtering players by role and team.
* Sorting players and teams by attributes like name and city.
* Profile management with edit, password update, and account deletion options.
* Integration of live cricket match data using an external API.
* Blade templating engine for dynamic views.
* Laravel's built-in validation for secure data inputs.
* Use of Laravel Eloquent ORM for database interactions.

Extra Features Covered

1)Dynamic Navbar:

* Adjusts based on user authentication status.
* Displays "Login" and "Sign Up" buttons for guests and "User Name," "Edit Profile," and "Logout" options for authenticated users.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

2)Search, Sorting and Filter Functionality

* Advanced filtering by team and role.
* Global search bar for players and teams.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

3)Real-Time Cricket Data Integration:

* Utilized an external cricket API to fetch live match data.
* Displays match details such as scores, teams, venue, and status dynamically.

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

4)Role-Based Views and Restrictions:

* Admin users can manage CRUD operations on teams and players.
* Regular users can only view public pages like the homepage and live match updates.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

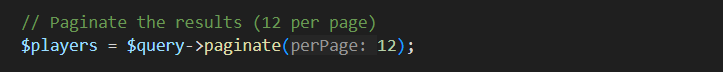
Description automatically generated

5)Pagination

* Added pagination to the **Players** and **Teams** pages for handling large datasets effectively.
* Ensures smooth navigation across multiple pages and improves usability for administrators managing content.

A screenshot of a computer

Description automatically generated



A screen shot of a computer code

Description automatically generated

6)Unit Testing

* Added unit tests for **Player** and **Team** models to validate CRUD operations, relationships, and data constraints.
* Ensures data consistency and robustness of core functionalities.

A black screen with white text

Description automatically generated

A black screen with colorful text

Description automatically generated

Missing Enhancements and Features

1)Role-based Dashboard Customization

* Currently, the admin and customer views share the same dashboard structure. The dashboards could be further customized to display unique and relevant data for each role.
* Reason Not Covered: Time constraints and focus on implementing role-based authorization and CRUD functionalities.

2)Image Upload for Teams and Players

* The ability to upload images for teams and players would enhance the visual appeal of the platform.
* Reason Not Covered: Integration of file storage and validation for image uploads required additional time and resources, which were not feasible during this project.

3)Real-Time Match Updates

* The API integration displays live cricket match data, but real-time updates using WebSocket’s or polling could have provided a more dynamic experience.
* Reason Not Covered: Real-time functionality required additional infrastructure setup, which was outside the project’s scope.

4)Customer Account Management

* Currently, only the admin can manage teams and players. Customers could have had functionality to bookmark teams/players or comment on match data.
* Reason Not Covered: Focus was placed on implementing core CRUD operations for admin users.

5)Email Notifications

* Adding email notifications for events like successful registration, team/player updates, or match alerts would enhance user experience.
* Reason Not Covered: Email setup and configuration required additional time for integrating Laravel's mail functionality.

6)Improved User Analytics

* Adding a feature to track user interactions (e.g., most viewed teams/players) could provide insights for administrators.
* Reason Not Covered: Required setting up an analytics system, which was deemed beyond the scope of this project.

YouTube Demo Video link - https://youtu.be/kcZ\_1m1hSAw

GitHub Repository link - https://github.com/tbc-bsc-l6/component-2-AWDANIL.git