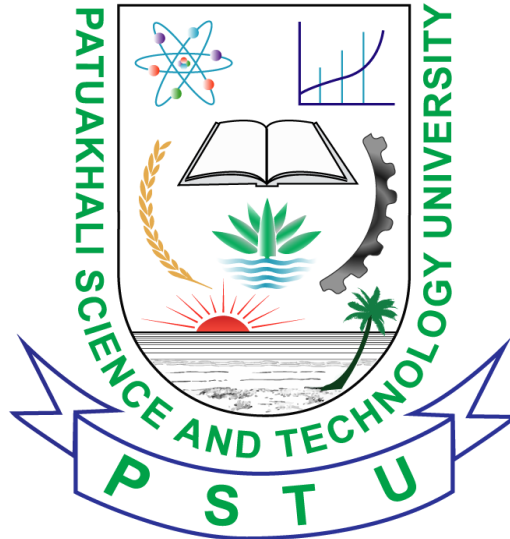


Patuakhali Science & Technology University (PSTU)

Faculty of Computer Science & Engineering (CSE)



Department of Computer science & Information Technology (CSIT)

Course code: CIT-320

Course Title: Software Development Project-II

Project Report on **CPL Live Scoring App**

Supervised by:

Prof. Dr. Md Abdul Masud.

&

Prof. Md Atikur Rahman.

Department of Computer Science &
Information Technology

Submitted by:

Name: Md. Rad Shahmat

Student ID: 1802070

Reg no: 08480

Session: 2018-19

Faculty of Computer Science &
Engineering.

Table of Contents

1. Executive Summary	01
2. Introduction	01
3. Project Objectives	01
4. System Design and Architecture	02
5. Technology Stack	02
6. Detailed Component Analysis	03
○ Admin Panel (React)	
○ User Application (Flutter)	
○ Server-Side Backend (PHP with REST APIs)	
7. User Experience and Interface Design	04
8. Core Functionalities and Features	07
9. Testing and Quality Assurance	07
10. Challenges and Solutions	07
11. Future Improvements	08
12. Conclusion	08
13. Appendices	08

1. Executive Summary

1

This document presents a comprehensive overview of the CPL (CSE Faculty Cricket League) Score Management and Live Score Sharing Application, a project aimed at digitizing the management and dissemination of cricket scores within the CSE Faculty at Patuakhali Science and Technology University (PSTU). The application streamlines scorekeeping for admins while providing real-time score updates to users via a Flutter-based mobile app, and efficiently handles data requests through PHP-based REST APIs. This innovative solution aims to enhance the fan experience for CPL audiences and serve as a useful tool for administrators.

2. Introduction

The CPL Score Management and Live Score Sharing Application is designed as a three-tier system to deliver a seamless scoring experience. It integrates three major components:

1. **Admin Panel (React)** – A feature-rich web interface for administrators to control and manage scores and match details.
2. **User Mobile Application (Flutter)** – A cross-platform app for end-users to access live scores and match information.
3. **Backend Server (PHP with REST APIs)** – The server side providing data management and secure, reliable communication between the frontend applications and the database.

3. Project Objectives

The application's primary goal is to:

- Digitally manage cricket scores and match details in real time.
- Facilitate real-time information sharing through a responsive, user-friendly mobile app.
- Ensure data security, consistency, and integrity across all user interfaces.

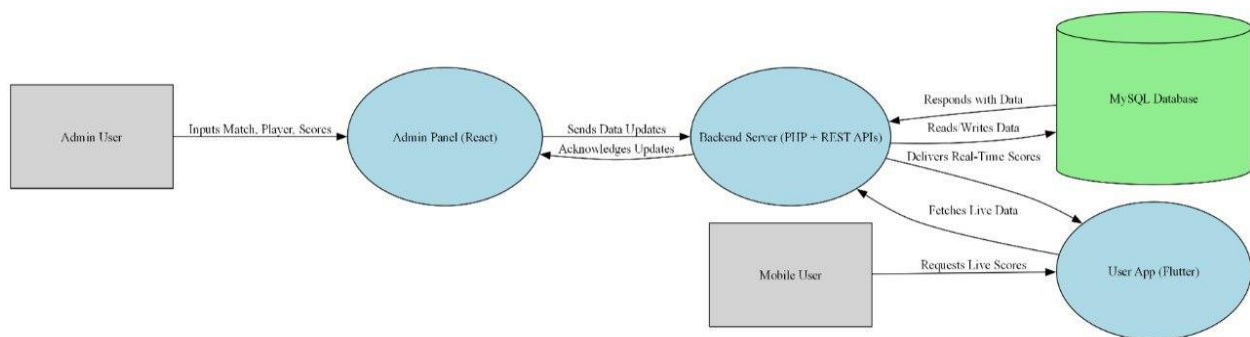
Secondary Goals:

- Reduce administrative workload with efficient data entry.
- Enhance user experience by providing easy access to schedules, scores, and team information.

4. System Design and Architecture

The CPL Score Management and Live Score Sharing Application adopts a modular architecture to maximize scalability and maintainability. Each component interacts seamlessly through RESTful APIs, allowing for effective division of responsibilities across the system. The architecture is organized as follows:

- **Frontend Layer:** The presentation layer where users and admins interact, built using React and Flutter.
- **Middleware/Backend Layer:** API endpoints written in PHP handle the business logic and data processing.
- **Data Layer:** A MySQL database maintains consistent and reliable storage of match scores, team information, and player statistics.



5. Technology Stack

- **Frontend for Admin Panel:** React.js
- **Frontend for Mobile App:** Flutter for cross-platform functionality
- **Backend:** PHP with RESTful APIs
- **Database:** MySQL for relational data management

- **Deployment:** N/A

6. Detailed Component Analysis

6.1 Admin Panel (React)

The React-based Admin Panel is a robust web interface designed for CPL administrators to handle various match-related operations, including real-time score updates. Key features include:

- **Dashboard View:** Displays a summary of ongoing matches, recent scores, and team stats.
- **Match Management:** Ability to add, edit, or delete matches and control match states.
- **Real-Time Score Update:** Admins can update scores on-the-go, which will reflect instantly on the user application.
- **Player and Team Management:** Maintain a roster of players and teams with relevant statistics.

6.2 User Mobile Application (Flutter)

The Flutter mobile application is designed to deliver real-time updates to users in an engaging and interactive format. Core features include:

- **Live Score Display:** Users receive instant updates for ongoing matches.
- **Match Schedules and Upcoming Matches:** A dedicated section for viewing match schedules and team lineups.
- **Team and Player Profiles:** Detailed statistics and player bios for fans.

6.3 Server-Side Backend (PHP with REST APIs)

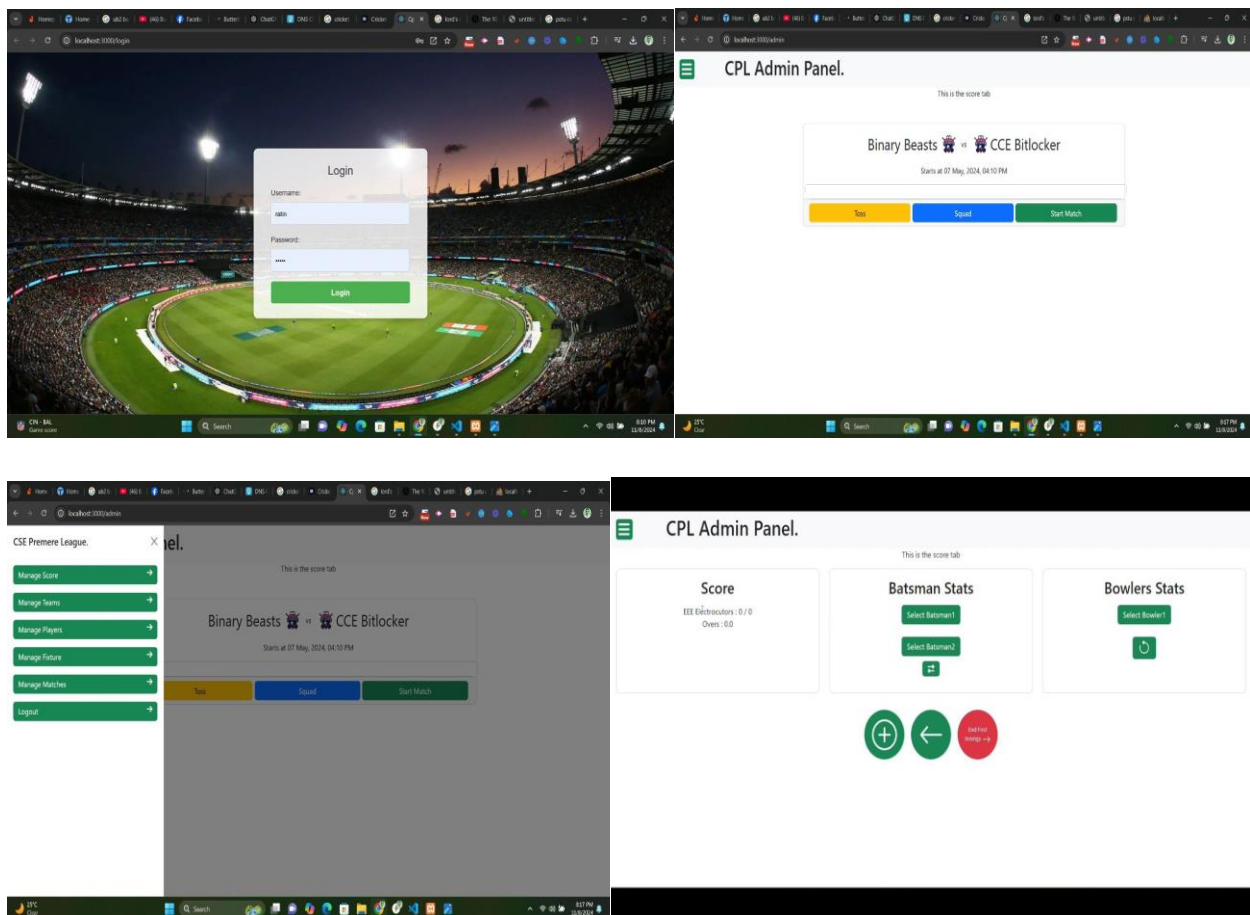
The PHP backend serves as the central data processing hub, facilitating seamless data flow and ensuring secure communication between frontend applications and the database. Key aspects of the backend include:

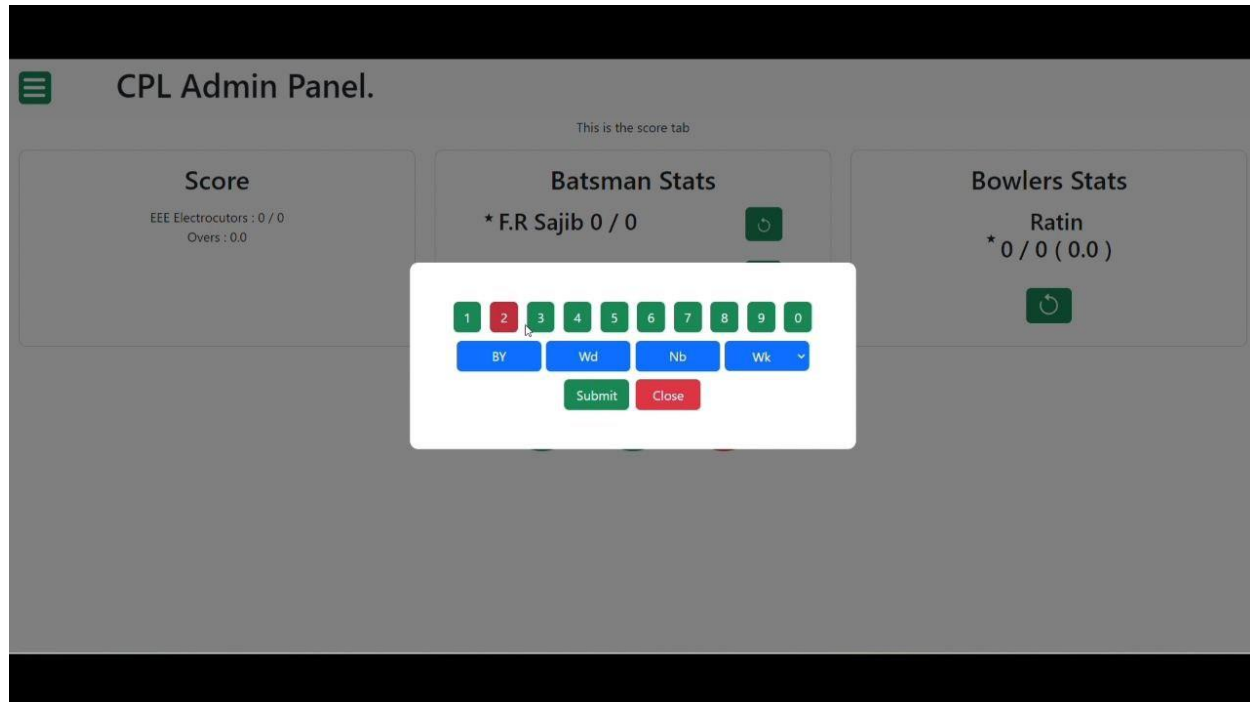
- **RESTful APIs:** Exposes APIs for CRUD operations on matches, players, and scores. 4
- **Authentication and Security:** Ensures secure access, protecting sensitive data.
- **Database Management:** MySQL tables are optimized for quick access and updates.
- **Error Handling and Logging:** Comprehensive error logging for easy troubleshooting and tracking.

7. User Experience and Interface Design

7.1 Admin Panel UI Design

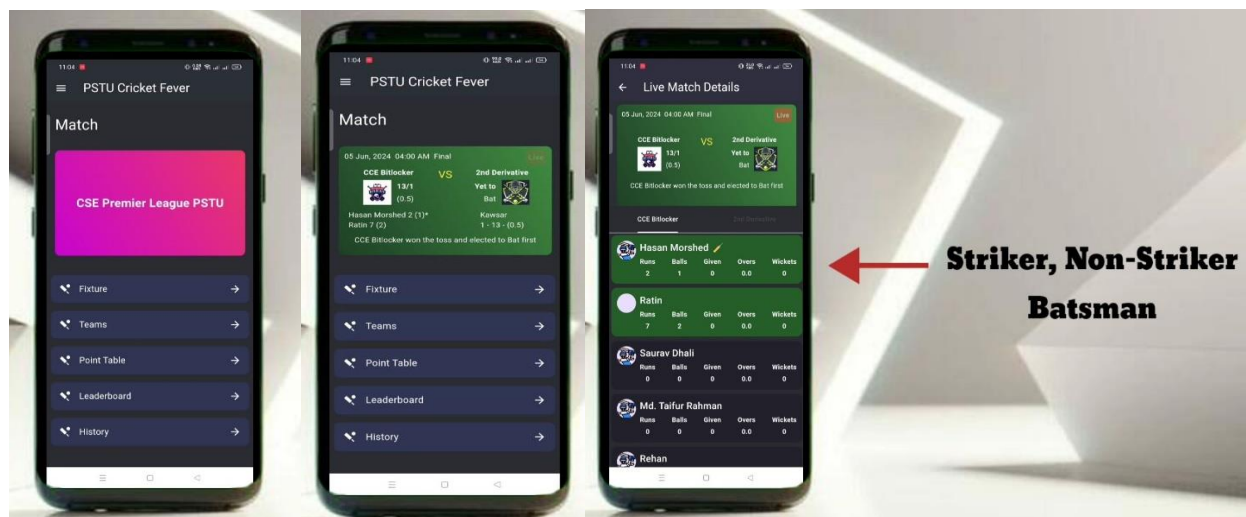
The Admin Panel has a clean, organized layout focused on ease of use and data clarity. Each section provides clear pathways to perform actions, enabling admins to quickly navigate and manage data.

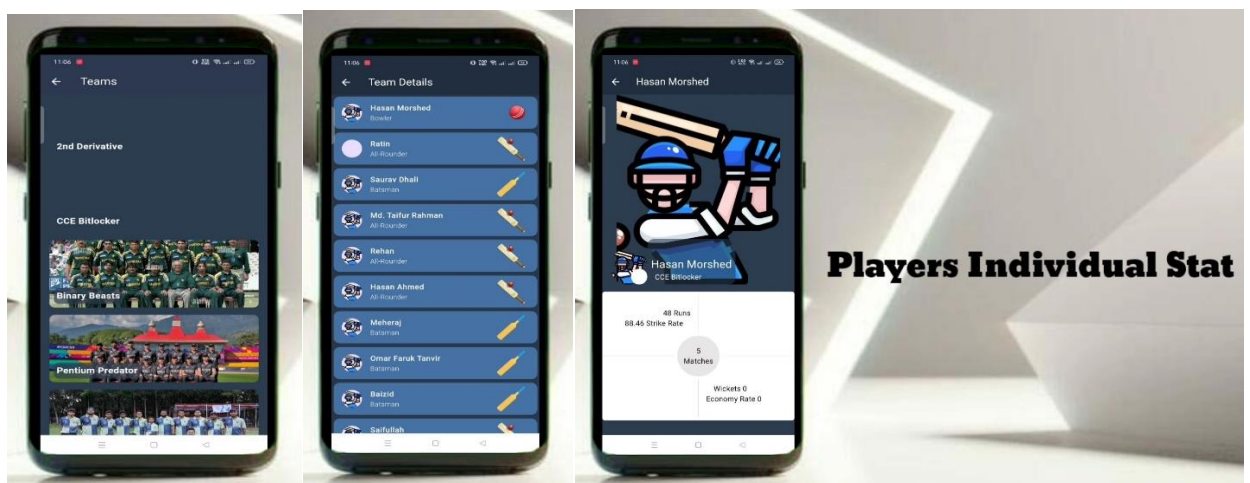




7.2 User Mobile Application UI Design

The mobile app UI emphasizes real-time updates and intuitive navigation. Designed with the end-user in mind, it provides an engaging experience with instant access to match data, live scores, and player stats.





8. Core Functionalities and Features

7

- **Admin Score Management:** Simplifies score entry and management with a user-friendly interface.
- **Real-Time Data Synchronization:** Updates scores in real time for the user app, enhancing the spectator experience.
- **Player and Team Management:** Allows admin to manage and update player and team data.
- **Live Score Display:** Provides users with immediate access to the latest scores.
- **Interactive Match Schedules:** Users can access schedules for upcoming matches and track match progress.

9. Testing and Quality Assurance

9.1 Unit Testing

- **React:** Tests cover key components in the Admin Panel.
- **Flutter:** Validates each widget and ensures proper rendering and functionality.

9.2 Integration Testing

- Verifies API interactions and data consistency across different parts of the application.

9.3 User Acceptance Testing (UAT)

- Conducted with sample users to validate ease of use, responsiveness, and accuracy of live updates.

10. Challenges and Solutions

10.1 Real-Time Data Synchronization

- **Challenge:** Managing real-time updates in a distributed system.

- **Solution:** Implemented optimized API calls to ensure data is updated efficiently across both platforms.

10.2 Cross-Platform Compatibility

- **Challenge:** Ensuring consistent UI/UX across different device types.
- **Solution:** Extensive testing on both Android and iOS to ensure a seamless experience.

11. Future Improvements

- **Enhanced Data Security:** Introduce advanced security measures for data handling.
- **Push Notification System:** Notify users of major match events and score updates.
- **Advanced Analytics:** Provide admins with match and player analytics for better decision-making.
- **Push Notifications:** Users receive timely alerts for match start, updates, and results (optional).

12. Conclusion

The CPL Score Management and Live Score Sharing Application demonstrates a successful integration of cross-platform tools to deliver a high-quality user experience. The system meets all primary objectives and is positioned for future growth with potential feature enhancements and scalability.

13. Appendices

- **Screenshots:** Admin Panel, User App Screenshots
- **Database Schema:** Diagrams