NORTHERN UNIVERSITY

B A N G L A D E S H

Knowledge for Innovation and Change

Department of CSE

Software Development - II

Course Code: CSE 2291

Final Project Report

E-Sports Tournament Management System

Submitted to

Amir Labib Khan

Lecturer

Department of CSE

Northern University Bangladesh

Submission Date: September 10, 2025

Project Details

Project Title	E-Sports Tournament Management System
Project Application	Database (MySQL, PHP, HTML, CSS)

Group Name: Tech Hunters

Group Members:

Student ID	Name	Signature
41230301367	Sanzid Zaman	
41230301816	Md. Al Amin	
41230301851	MD. Imran Hosen	

Abstract

The E-Sports Tournament Management System is a web-based application designed to streamline the organization and administration of e-sports events. Developed using PHP for backend logic, MySQL for database management, and basic HTML/CSS for the frontend, this system enables users to manage key entities such as games, tournaments, teams, players, sponsors, matches, and results. It provides functionalities for adding, viewing, and tracking e-sports data, with a focus on simplicity and efficiency.

This report details the system's architecture, implementation, database design, features, and potential improvements. The project addresses the growing need for digital tools in the e-sports industry, which has seen exponential growth in recent years. By automating manual processes like team registration, match scheduling, and result tracking, the system reduces errors and enhances user experience for tournament organizers, teams, and participants.

Key highlights include a user-friendly dashboard, CRUD (Create, Read, Update, Delete) operations for core entities, and a relational database schema that ensures data integrity. The system is deployed on a local server environment (e.g., XAMPP) and can be extended for production use with security enhancements.

With the exponential growth of the esports industry, which is projected to exceed $1.8 billion in revenue by 2025, there is an increasing need for robust management systems. This platform addresses the organizational challenges faced by tournament organizers, teams, and players by providing structured management of competitions, participants, and results.

This project report outlines the development and functionality of the E-Sports Tournament Management System, a web-based application designed to streamline the organization and administration of competitive gaming events. The system provides a centralized platform for managing core esports entities, including players, teams, games, tournaments, and sponsorships. Developed using a standard PHP and MySQL technology stack, the application features a dashboard for quick oversight, comprehensive data management forms, and structured data views. The primary objective was to create a robust and user-friendly tool to facilitate tournament operations, from initial setup to final results tracking.

Table of Contents

1. Introduction

1.1 Project Background

E-sports, also known as electronic sports, represents the competitive side of video gaming where professional players and teams compete in organized tournaments. Over the past two decades, e-sports has grown into a multi-billion-dollar global industry, recognized as one of the fastest-growing segments in the digital entertainment sector. According to recent industry reports, e-sports tournaments attract millions of online and offline viewers worldwide, generating substantial revenue through ticket sales, sponsorships, advertising, and media rights. The scale and popularity of events such as the League of Legends World Championship, The International (Dota 2), and Fortnite World Cup demonstrate the significant cultural and economic impact of the e-sports ecosystem.

Despite this rapid growth, managing e-sports tournaments remains a complex challenge. Organizers are required to handle extensive amounts of data, including player registrations, team rosters, tournament brackets, match schedules, statistical records, and sponsorship details. Traditional methods such as manual documentation or spreadsheet-based systems are not only time-consuming but also prone to human error, data redundancy, and lack of scalability. As tournaments expand to national and international levels, the inefficiencies of such approaches become increasingly evident, often resulting in mismanagement, delayed operations, and diminished credibility of the event.

To address these challenges, the present project has been developed as part of the Software Development Lab 2 course. The primary objective is to design and implement a centralized web-based application that can efficiently manage e-sports tournaments in an organized and systematic manner. The system aims to provide automation, consistency, and accessibility, thereby reducing administrative overhead while improving the user experience for both tournament organizers and participants.

The design of this project is influenced by real-world e-sports management platforms such as ESL Play and Battlefy, which are widely used for hosting online competitions. However, unlike these comprehensive commercial solutions, the proposed system adopts a focused academic scope by concentrating solely on core management functionalities such as player registration, team management, tournament scheduling, and result tracking. Advanced features like live streaming integration, payment gateways, or automated ranking algorithms have been deliberately excluded to maintain a clear project scope, ensure feasibility within the academic timeframe, and emphasize foundational software development principles.

1.2 Motivation

The motivation for developing an E-sports Tournament Management System arises from both the growing importance of e-sports worldwide and the practical challenges faced by tournament organizers. With the industry experiencing unprecedented growth, efficient management tools have become essential to ensure smooth operations, fair competition, and professional event execution.

Traditional methods such as spreadsheets, manual record-keeping, or informal communication channels are no longer sufficient for handling the complex requirements of modern tournaments, especially those involving large numbers of participants, multiple rounds, and diverse stakeholders. These outdated approaches often result in data inconsistency, scheduling conflicts, miscommunication, and administrative delays, which can negatively impact the tournament's credibility and participant experience.

This project is motivated by the need to create a centralized, reliable, and user-friendly system that simplifies the core processes of tournament organization. By automating essential tasks such as player registration, team formation, scheduling, result recording, and bracket management, the system minimizes human error, saves time, and ensures accuracy. Moreover, it provides an accessible platform that can be used by both organizers and participants, fostering transparency, efficiency, and fairness.

From an academic perspective, this project also serves as a valuable learning opportunity. It allows students to apply software engineering concepts such as database design, system architecture, and web development to a real-world problem. The project bridges theoretical knowledge with practical application, preparing students for future challenges in both academic and professional domains.

Ultimately, the motivation behind this project is to combine technological innovation with practical necessity, addressing real-world challenges in the e-sports industry while simultaneously enhancing the academic growth of the developers involved.

1.3 Objective

- To develop a robust system for adding, viewing, and managing e-sports entities (games, tournaments, teams, players, sponsors, matches, and results).
- To ensure data integrity through a relational database model.
- To provide an intuitive user interface for administrators and users.
- To demonstrate practical application of PHP, MySQL, HTML, and CSS in web development.
- To facilitate easy tracking of tournament progress, from upcoming events to completed results.

1.4 Scope

The system covers:

- CRUD operations for tournaments, teams, players, and sponsors.
- Dashboard for quick statistics and recent activities.
- Basic match scheduling and result recording (though full implementation for matches is referenced but not fully detailed in provided files).
- Integration with a MySQL database for persistent storage.

1.4 Limitations

- Lacks security features like input validation against SQL injection (though real_escape_string is used partially).
- No update or delete functionalities in the provided code snippets.
- Assumes a single-user environment; no multi-user support.
- Basic styling without advanced CSS frameworks.
- Sample data is hardcoded in SQL, limiting scalability without further development.

2. System Requirements

2.1 Hardware Requirements

- Processor: Intel Core i3 or equivalent.
- RAM: Minimum 4 GB.
- Storage: 500 MB free space.
- Internet: Not required for local deployment, but needed for any future cloud hosting.

2.2 Software Requirements

- Operating System: Windows, Linux, or macOS.
- Web Server: Apache (via XAMPP or similar).
- Database: MySQL 5.7 or later.
- Programming Languages: PHP 7.4+.
- Browser: Modern web browsers like Chrome, Firefox, or Edge.

2.3 Functional Requirements

- Users can add new tournaments, specifying details like name, game, dates, prize pool, location, and status.
- Users can manage teams, including assigning them to games and coaches.
- Players can be added to teams with personal details like name, in-game name, role, email, and date of birth.
- Sponsors can be registered with name, email, and sponsorship amount.
- Dashboard displays counts of entities and lists upcoming tournaments and recent match results.
- Basic querying for lists of all entities with view links to details pages.

2.4 Non-Functional Requirements

- Performance: Response time under 2 seconds for database queries.
- Usability: Simple forms and tables for data entry and display.
- Reliability: Data persistence via MySQL with error handling for queries.
- Maintainability: Modular code with separate files for each entity (e.g., teams.php, players.php).

3. Design and Architecture

3.1 System Architecture

The system follows a three-tier architecture:

- Presentation Layer: HTML forms and tables for user interaction, styled with CSS (folder referenced in README).
- Application Layer: PHP scripts handle form submissions, database queries, and business logic.
- Data Layer: MySQL database stores all persistent data.

Key files include:

- db_connect.php: Establishes database connection.
- index.php: Dashboard page.
- teams.php, players.php, sponsors.php, tournaments.php: Entity management pages.

3.2 Use Case Diagram

User Flow:

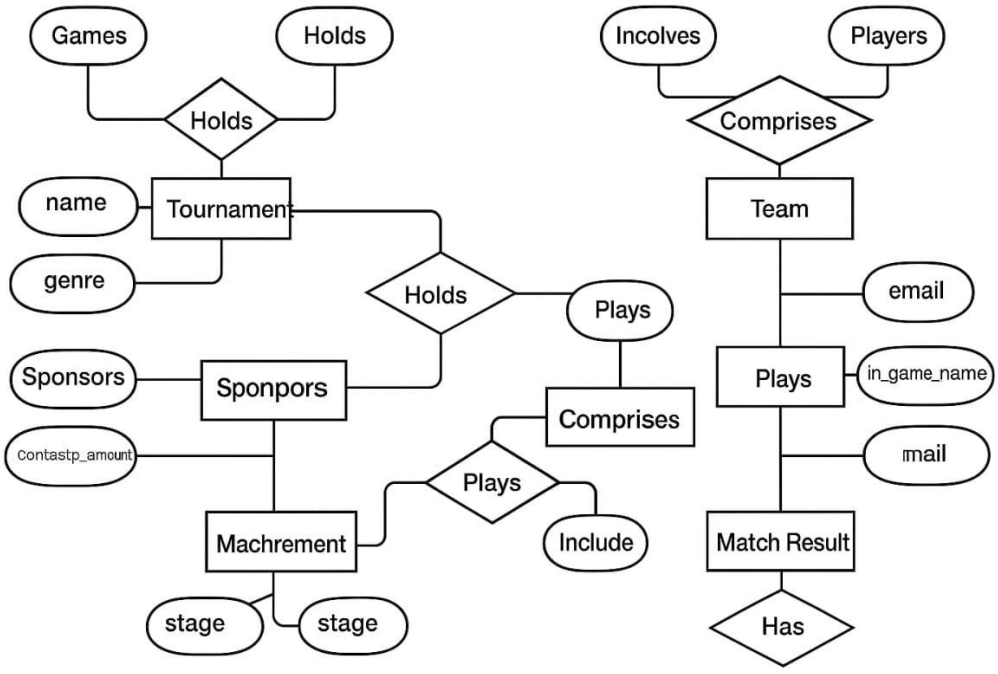

Fig 3.2.1 User Flow

ER Diagram:

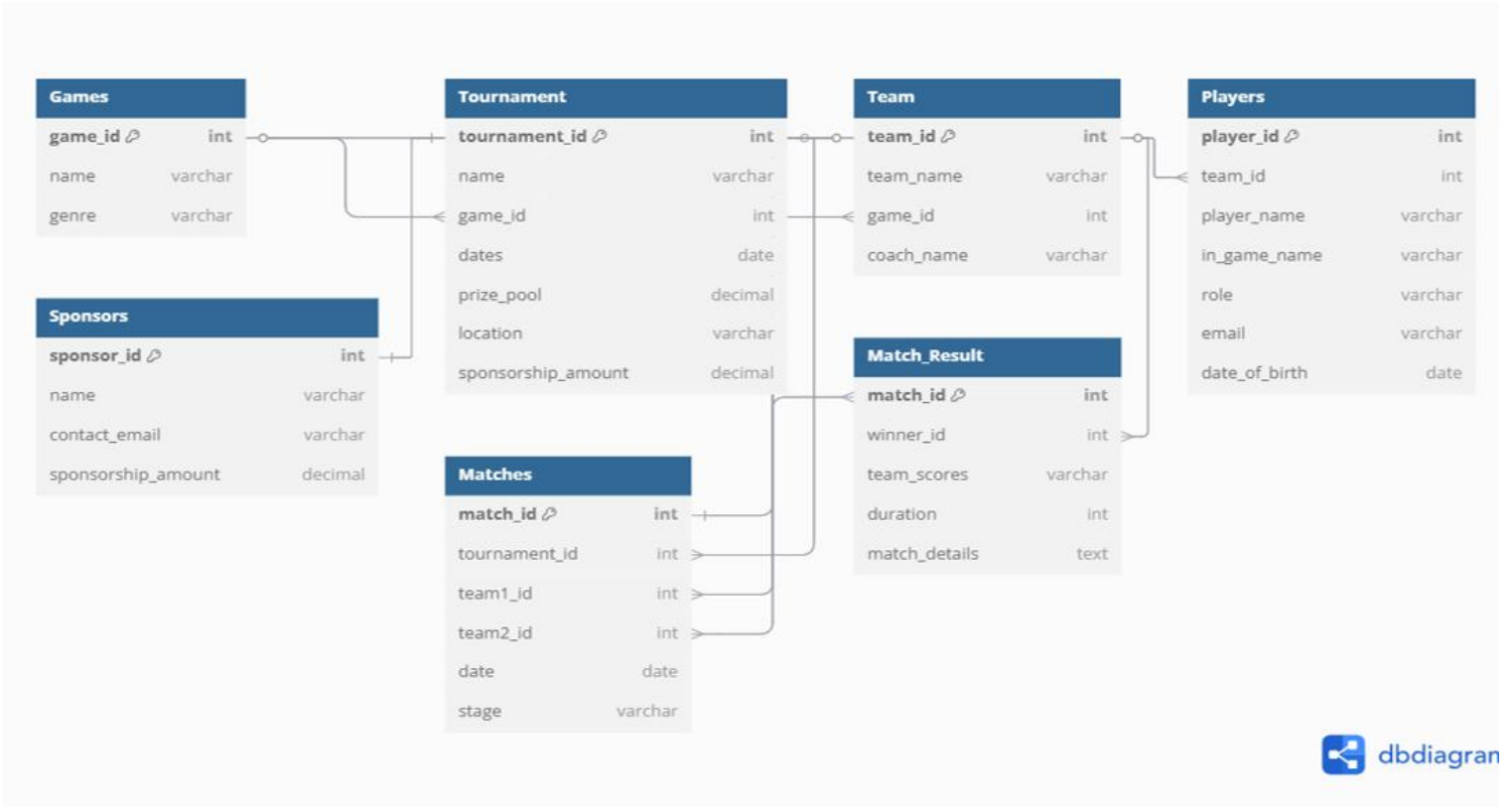

Fig 3.2.2 ER Diagram

3.3 Relationship between tables

- Games to Tournaments: One-to-many (a game can have multiple tournaments).
- Games to Teams: One-to-many (a game can have multiple teams).
- Teams to Players: One-to-many (a team can have multiple players).
- Tournaments to Matches: One-to-many (a tournament can have multiple matches).
- Teams to Matches: Many-to-many (via team1_id and team2_id).
- Matches to Match Results: One-to-one (each match has one result).
- Teams to Match Results: One-to-many (a team can win multiple matches).

3.4 User Interface Dashboard

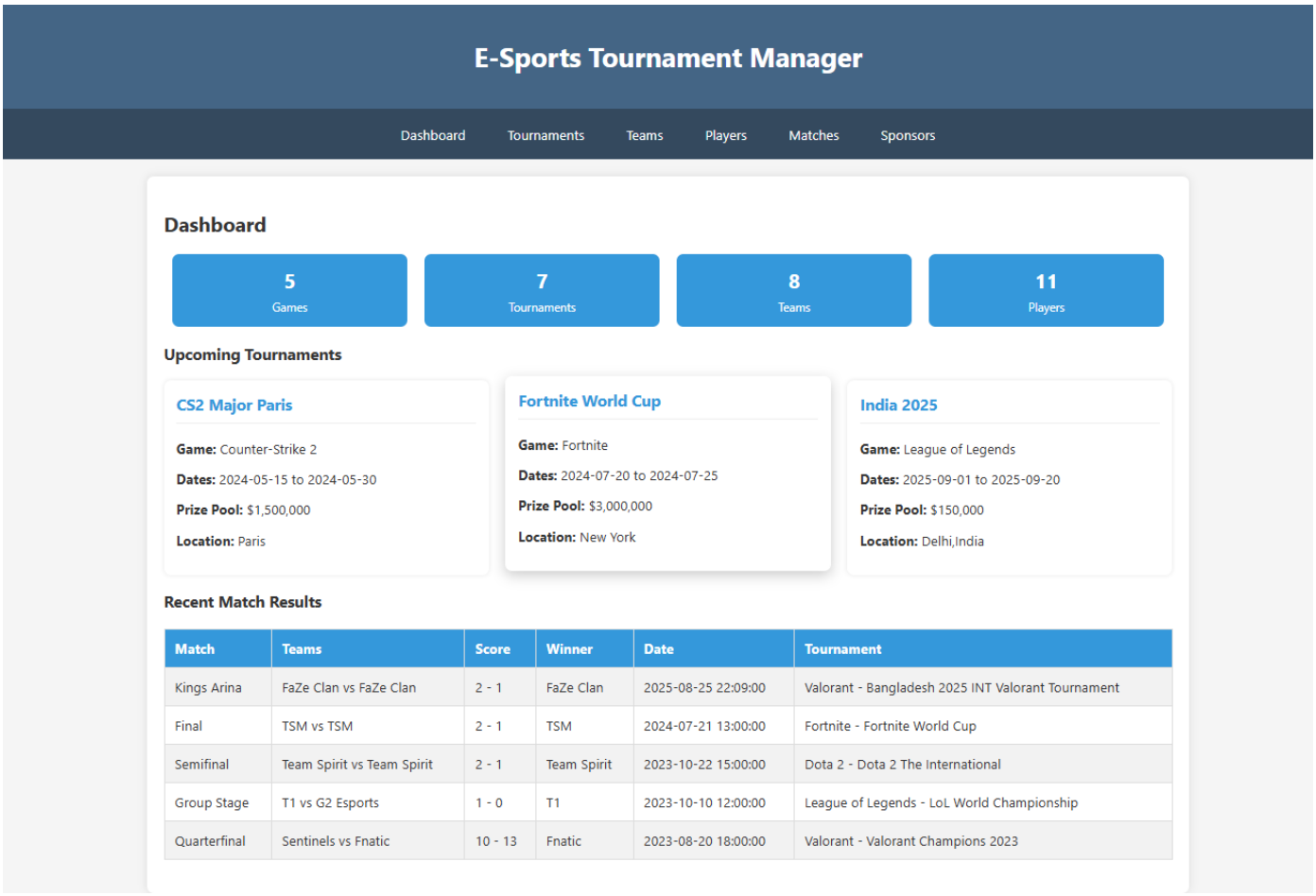

Fig 3.4.1 User Dashboard

4. Management

4.1 Tournaments Management

Add: Form with validation (required fields).

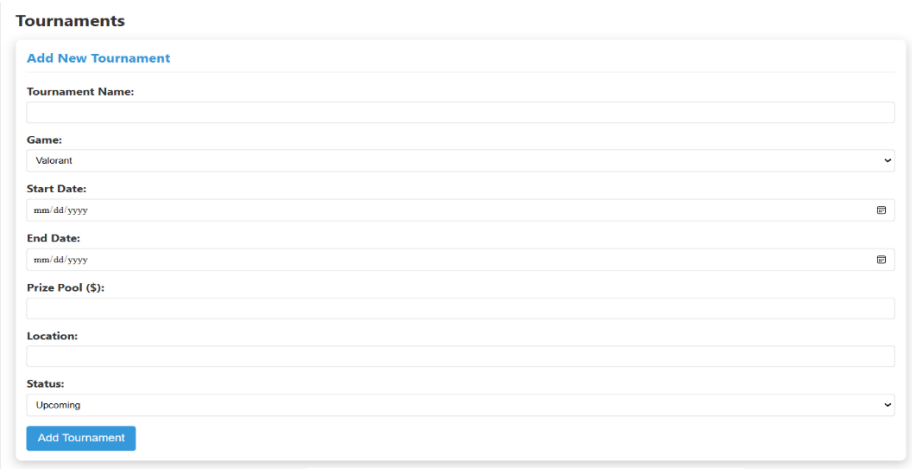

Fig 4.1.1 Add tournament form

List: Table with name, game, dates, prize, location, status, view link.

All Tournaments

Name	Game	Dates	Prize Pool	Location	Status	Actions
Bangladesh 2025 INT Valorant Tournament	Valorant	2025-11-10 to 2025-11-20	$800,000	Dhaka, Bangladesh	Upcoming	View
India 2025	League of Legends	2025-09-01 to 2025-09-20	$150,000	Delhi,India	Upcoming	View
Fortnite World Cup	Fortnite	2024-07-20 to 2024-07-25	$3,000,000	New York	Upcoming	View
CS2 Major Paris	Counter-Strike 2	2024-05-15 to 2024-05-30	$1,500,000	Paris	Upcoming	View
Dota 2 The International	Dota 2	2023-10-20 to 2023-10-30	$4,000,000	Singapore	Completed	View
LoL World Championship	League of Legends	2023-10-05 to 2023-11-05	$2,500,000	Seoul	Completed	View
Valorant Champions 2023	Valorant	2023-08-15 to 2023-08-30	$1,000,000	Los Angeles	Completed	View

Fig 4.1.2 Tournament table

4.2 Team Table

Add: Name, game (dropdown), coach.

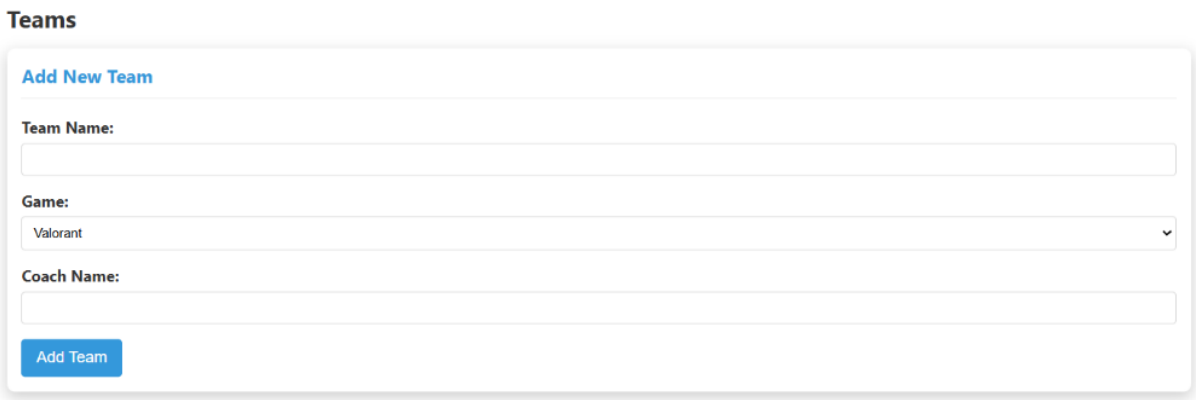

Fig 4.2.1 Add team form

List: Name, game, coach, player count, view link.

All Teams

Name	Game	Coach	Players	Actions
FaZe Clan	Counter-Strike 2	Filip "NEO" Kubski	1 players	View
Fnatic	Valorant	Jacob "mini" Harris	2 players	View
G2 Esports	League of Legends	Dylan Falco	1 players	View
Natus Vincere	Counter-Strike 2	Andrey "B1ad3" Gorodenskiy	1 players	View
Sentinels	Valorant	Shane "Rawkus" Flaherty	2 players	View
T1	League of Legends	Bae "Bengi" Seong-woong	2 players	View
Team Spirit	Dota 2	Dmitriy "Larl" Karpov	1 players	View
TSM	Fortnite	Connor "Co1azo" Colazo	1 players	View

Fig 4.2.2 Teams table

4.3 Players Table

Add: Team (dropdown), name, in-game name, role, email, DOB.

Fig 4.3.1 Add players form

List: Name, in-game, team, game, role, email, view link.

All Players

Name	In-Game Name	Team	Game	Role	Email	Actions
Bae Seong-woong	Bengi	T1	League of Legends	Jungle	bengi@t1.gg	View
Håvard Nygaard	rain	FaZe Clan	Counter-Strike 2	Rifler	rain@faze.gg	View
Illya Mulyarchuk	Yatoro	Team Spirit	Dota 2	Carry	yatoro@teamspirit.gg	View
Jake Howlett	Boaster	Fnatic	Valorant	Initiator	boaster@fnatic.gg	View
Kyle Giersdorf	Bugha	TSM	Fortnite	Solo	bugha@tsm.gg	View
Lee Sang-hyeok	Faker	T1	League of Legends	Mid	faker@t1.gg	View
Nikita Sirmitev	Derke	Fnatic	Valorant	Duelist	derke@fnatic.gg	View
Oleksandr Kostyliev	s1mple	Natus Vincere	Counter-Strike 2	AWPer	s1mple@navi.gg	View
Rasmus Winther	Caps	G2 Esports	League of Legends	Mid	caps@g2.gg	View
Shahzeb Khan	ShahZaM	Sentinels	Valorant	Initiator	shahzam@sentinels.gg	View
Tyson Ngo	TenZ	Sentinels	Valorant	Duelist	tenz@sentinels.gg	View

Fig 4.3.2 Players table

4.4 Matches & Result(partial) Table

All Matches

Tournament	Game	Match	Teams	Score	Winner	Date	Actions
Bangladesh 2025 INT Valorant Tournament	Valorant	Kings Arina	FaZe Clan vs FaZe Clan	2 - 1	FaZe Clan	August 25, 2025 22:09	View Results | Add Result
Fortnite World Cup	Fortnite	Final	TSM vs TSM	2 - 1	TSM	July 21, 2024 13:00	View Results | Add Result
CS2 Major Paris	Counter-Strike 2	Group Stage	Natus Vincere vs FaZe Clan	0 - 0	TBD	May 16, 2024 16:00	View Results | Add Result
Dota 2 The International	Dota 2	Semifinal	Team Spirit vs Team Spirit	2 - 1	Team Spirit	October 22, 2023 15:00	View Results | Add Result
LoL World Championship	League of Legends	Group Stage	T1 vs G2 Esports	1 - 0	T1	October 10, 2023 12:00	View Results | Add Result
Valorant Champions 2023	Valorant	Quarterfinal	Sentinels vs Fnatic	10 - 13	Fnatic	August 20, 2023 18:00	View Results | Add Result
Valorant Champions 2023	Valorant	Group Stage	Sentinels vs Fnatic	13 - 9	Sentinels	August 16, 2023 14:00	View Results | Add Result

Fig 4.4.1 Match table

Result Details

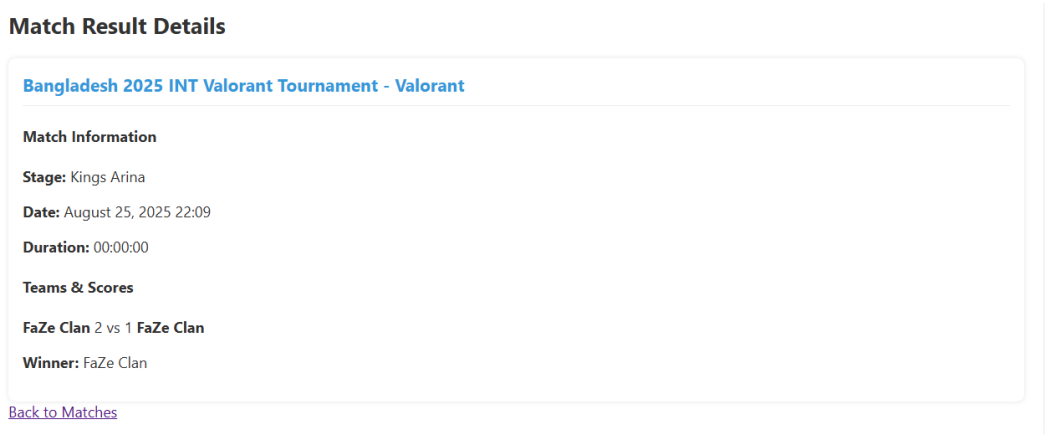

Fig 4.4.2 Result details

4.5 Sponsors Table

Add: Name, email, amount.

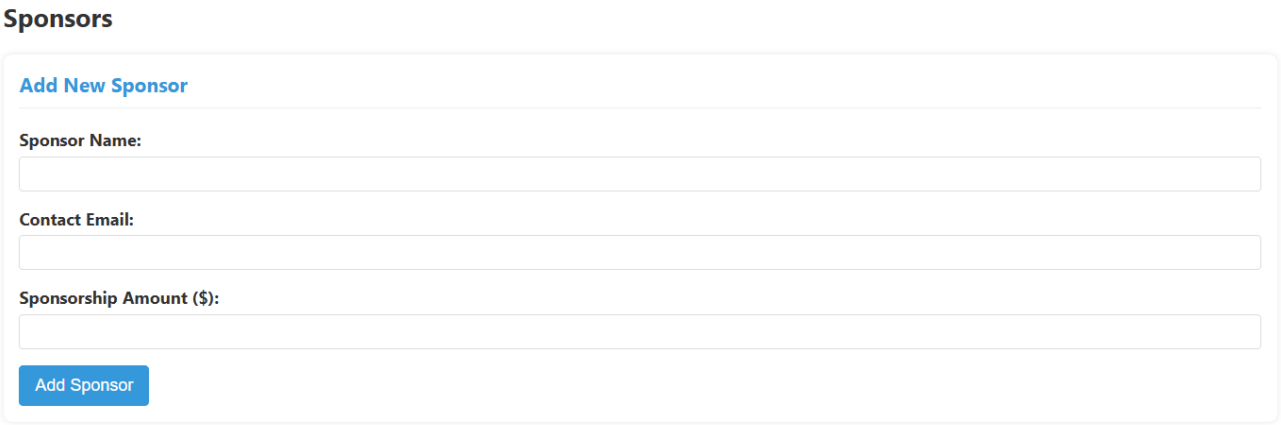

Fig 4.5.1 Add sponsors form

List: Name, email, formatted amount, view link.

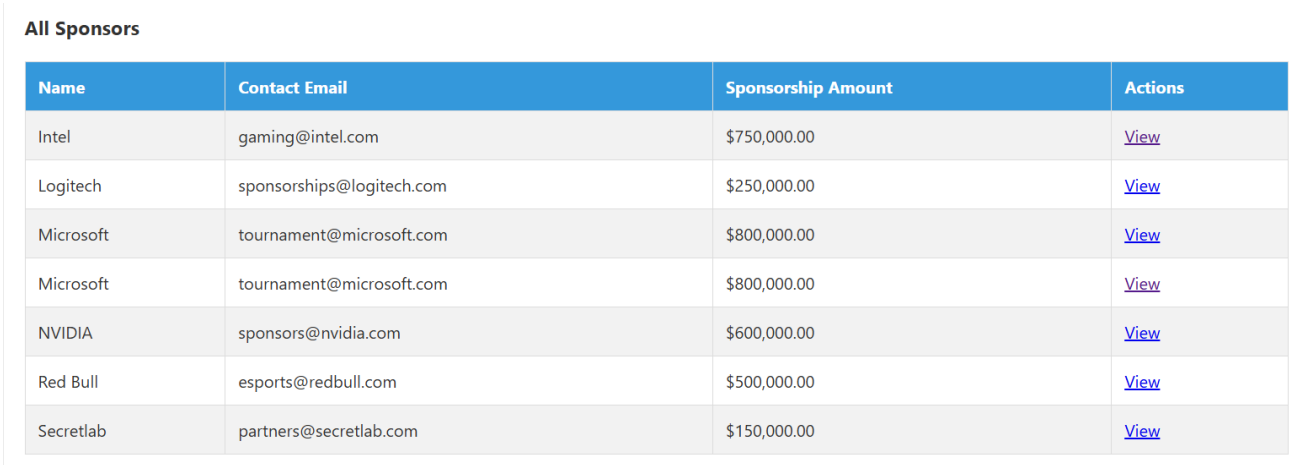

Fig 4.5.2 Sponsors table

5. Advantage

- **Simplicity and Ease of Use:** The system employs a straightforward web-based interface with basic HTML forms and tables, making it accessible for users without advanced technical skills. Administrators can quickly add and view entities like teams, players, and tournaments through intuitive dropdowns and input fields, reducing the learning curve for e-sports organizers.
- **Cost-Effective Development:** Built using open-source technologies such as PHP, MySQL, HTML, and CSS, the project incurs minimal costs. It runs on local servers like XAMPP, eliminating the need for expensive proprietary software or cloud services during initial deployment.
- **Data Integrity and Relational Structure:** The MySQL database schema features well-defined relationships with foreign keys, ensuring consistency (e.g., teams linked to games, players to teams). This prevents data anomalies and supports efficient querying, such as joining tables for comprehensive reports on tournaments and matches.
- **Modular Design:** Code is organized into separate PHP files for each entity (e.g., teams.php, players.php), promoting maintainability. This allows easy expansion, such as adding new features like match scheduling, without overhauling the entire system.
- **Dashboard for Quick Insights:** The index page provides at-a-glance statistics (e.g., counts of teams and players) and lists upcoming tournaments and recent results, enabling efficient monitoring of e-sports activities.
- **Sample Data for Testing:** The SQL script includes realistic sample data (e.g., popular games like Valorant and teams like Sentinels), facilitating immediate testing and demonstration without manual data entry.
- **Automation of Manual Processes:** By digitizing team registration, player management, and result tracking, the system reduces errors associated with traditional methods like spreadsheets, improving accuracy in tournament administration.

6. Drawbacks

- **Limited Security:** Vulnerable to SQL injection due to reliance on real_escape_string and hardcoded credentials, lacking robust input validation.
- **No Authentication:** Single-user setup without login or access control, risking unauthorized access.
- **Incomplete CRUD:** Lacks update and delete functionalities, limiting data management.
- Basic UI: Non-responsive, lacks modern frameworks or dynamic features, reducing user engagement.
- **Scalability Issues:** Designed for local servers, struggles with large datasets or high traffic.
- Partial Features: Matches/results management incomplete, as key pages are referenced but not provided.
- **Weak Validation:** Minimal server-side checks and no logic for date constraints, risking invalid data.
- **Local Dependency:** Tied to specific environments (e.g., Apache/MySQL), complicating cloud deployment.

7. Conclusion

The E-Sports Tournament Management System, as demonstrated by the provided project files, represents a successful implementation of a core administrative tool for the e-sports industry. The system achieves its primary objective of offering a centralized platform for managing essential tournament data, including player registrations, team rosters, match schedules, and results. By automating these processes, the system reduces human error, improves data accuracy, and streamlines administrative workflows, making tournament management significantly more efficient.

The project's well-structured database design ensures that all information is stored in a consistent, organized manner, allowing for easy retrieval and updates. The logical separation of code into modules enhances maintainability, readability, and scalability, making it easier for future developers to extend or modify the system according to evolving requirements. The use of a web-based interface further contributes to accessibility, enabling both organizers and participants to interact with the system conveniently.

In addition to fulfilling its immediate objectives, the project provides a solid foundation for future enhancements. Potential extensions could include advanced features such as live match updates, ranking algorithms, automated notifications, payment integration for tournament fees, and even streaming support, which would further increase the system's utility and relevance in real-world scenarios.

From an academic perspective, this project demonstrates the practical application of software development principles, including system analysis, database management, user interface design, and modular programming. It serves not only as a learning tool for understanding e-sports management but also as a realistic prototype that reflects industry practices on a smaller scale.

In conclusion, the E-Sports Tournament Management System successfully bridges the gap between theoretical knowledge and practical application, offering a reliable, efficient, and adaptable solution for tournament management while providing a strong basis for future growth and research in this rapidly expanding field.

7.1 Assessment of the Project

The E-Sports Tournament Management System has been evaluated based on several key criteria, including functionality, usability, scalability, maintainability, and overall alignment with project objectives. The assessment highlights the system's strengths, limitations, and potential areas for improvement.

The system successfully addresses the core requirements of e-sports tournament management. Key functionalities include:

- Player and Team Management: Efficient registration and storage of participant information, including profiles, team assignments, and contact details.
- Tournament Scheduling and Brackets: Automated generation and management of match schedules, including tracking progression through multiple rounds.
- Result Recording and Reporting: Accurate storage of match results and real-time updates on tournament standings.
- Centralized Data Management: Consolidation of all critical tournament data in a structured database, reducing redundancy and ensuring data integrity.
- These functionalities demonstrate that the system meets its primary objectives and provides a practical tool for tournament organizers.

Overall, the E-Sports Tournament Management System represents a successful academic and practical implementation of a core tournament management tool. It provides a reliable, efficient, and centralized platform for managing essential tournament operations. While there are areas for future enhancement, the project demonstrates strong technical execution, logical system design, and alignment with real-world industry needs.

7.2 Future Work

While the current version of the E-Sports Tournament Management System successfully implements the core functionalities required for managing tournaments, there are several opportunities to expand and enhance the system to make it more comprehensive, user-friendly, and closer to real-world industry standards. Future work can be categorized into the following areas:

1. Advanced Tournament Features

- Live Match Updates: Integration of real-time score updates and live match streaming to enhance participant and audience engagement.
- Automated Ranking and Statistics: Implementation of ranking algorithms, player performance metrics, and statistical analysis to provide deeper insights into tournament outcomes.
- Dynamic Bracket Generation: Support for different tournament formats such as double-elimination, round-robin, and Swiss-system brackets.

2. User Experience Enhancements

- Modern Web Interface: Development of a more responsive, visually appealing, and interactive user interface to improve usability across devices.
- Notifications and Alerts: Automated email or in-app notifications to inform participants about match schedules, results, and tournament announcements.
- Multilingual Support: Adding support for multiple languages to make the system accessible to a wider range of users globally.

3. Administrative and Security Improvements

- Role-Based Access Control: Introducing different user roles such as organizers, referees, and players to manage permissions and enhance security.
- Data Backup and Recovery: Implementation of automated backup systems and recovery protocols to ensure data integrity in case of system failures.
- Audit Logs: Tracking changes made by users for accountability and transparency in tournament management.

4. Integration with External Services

- Payment Gateway Integration: Allowing participants to pay registration fees or purchase tournament-related merchandise through secure online payment systems.
- Social Media Integration: Sharing tournament updates, results, and highlights directly to social media platforms for wider engagement.
- API Development: Creating APIs to allow third-party applications or websites to access tournament data and results programmatically.

5. Scalability and Performance Optimization

- Cloud-Based Deployment: Migrating the system to cloud infrastructure to improve scalability, accessibility, and reliability for large-scale tournaments.
- Performance Optimization: Enhancing database queries, server response times, and application efficiency to handle large numbers of concurrent users.
- Mobile Application: Developing companion mobile apps for iOS and Android to allow participants and organizers to interact with the system on-the-go.

6. Research and Academic Applications

- AI-Powered Features: Incorporating artificial intelligence to predict match outcomes, suggest optimal team formations, or detect cheating.
- Data Analytics for E-Sports Trends: Using tournament data to study player performance, team strategies, and audience engagement patterns for research purposes.