**FINAL PROJECT PROPOSAL**

Sports Team Management System

Name: ANAM TABASSUM

Name of Project: Sports Team Management System

**SUMMARY :**

The Sports Team Management System is a comprehensive database application designed to streamline the management of sports teams, their players, coaches, matches, and training activities. It maintains detailed records of teams, including their levels and affiliations, as well as personal and professional information of players and coaches. The system tracks player positions, dates of joining, and associates each player and coach with their respective teams. Additionally, it manages match schedules, venues, and results, providing a clear picture of team performance across different locations.

Player performance is carefully recorded for each match, capturing key statistics such as goals, assists, and overall ratings. The system enforces data integrity through triggers that prevent invalid performance data, ensuring the accuracy of records. Training sessions and player attendance are also managed, along with leave records that note the reasons and dates of player absences.

To maintain consistency, the system uses transactions when adding new players and registering them for training sessions. It offers automated updates on player information through triggers that track the last modification time. The database supports robust reporting capabilities, including views that summarize player statistics and stored procedures that retrieve players by team, enabling easy access to meaningful insights.

**Technologies Used :**

MySQL: Relational database management system used to store and manage all data.

SQL: Used for creating tables, inserting data, querying information, and building views, procedures, and triggers.

Constraints: Primary keys, foreign keys, and checks are used to maintain data integrity.

Stored Procedures: Encapsulate repetitive logic such as retrieving players by team.

Triggers: Automatically enforce rules, such as preventing negative values or updating timestamps.

Transactions: Ensure data consistency by grouping multiple operations into a single unit.

Views: Provide summarized and filtered data views, such as player statistics.

ER Diagram Tools: Tools like MySQL Workbench, dbdiagram.io are used to visualize database structure and relationships.

**Features Used in the Project :**

Table Creation with Relationships:  
Designed normalized tables with primary and foreign key constraints to maintain data integrity and establish relationships (e.g., Player–Team, Match–Venue).

Data Insertion:  
Populated tables with realistic data for teams, players, coaches, matches, venues, performance stats, and training sessions.

Joins and SQL Queries:  
Used various JOIN operations to fetch meaningful reports like player performance, match history, and training participation.

Views:  
Created custom views (e.g., Player\_Stats) to present summarized data on player goals, assists, and average ratings.

Stored Procedures:  
Implemented procedures like GetPlayersByTeam(team\_id) to modularize and reuse query logic.

Triggers:  
Developed triggers to automatically enforce business rules:

Preventing insertion of negative performance values.

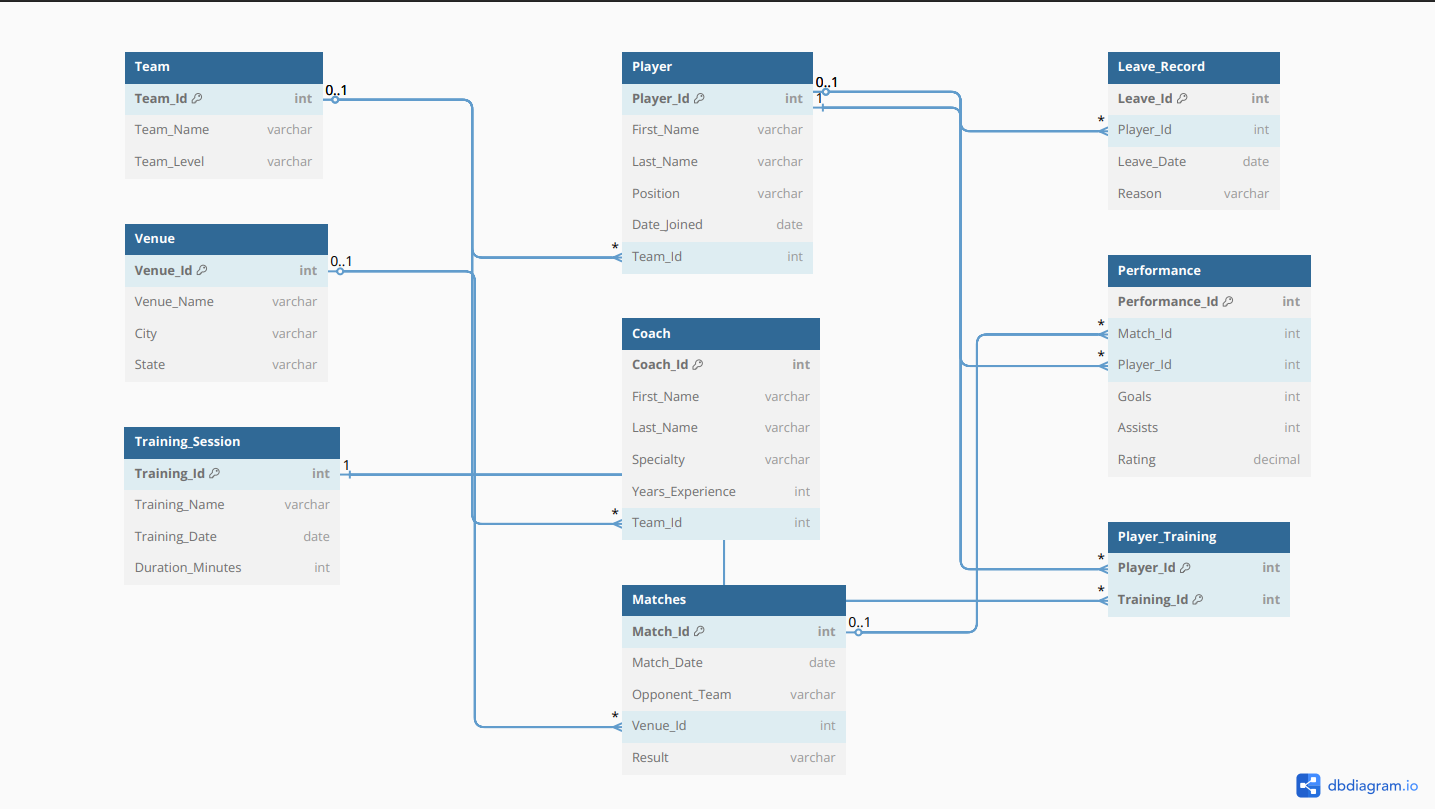
Updating a player’s last modified timestamp on data change.

Transactions:  
Used START TRANSACTION, COMMIT, and ROLLBACK to ensure atomic operations while inserting new players and related training data.

Aggregate Functions & Grouping:  
Utilized COUNT(), SUM(), AVG(), GROUP BY, and HAVING for analytics (e.g., players with high average ratings).

ER Diagram Design:  
Modeled the complete database structure with Entity-Relationship diagrams to visualize data flow and table dependencies.

**ER-DIAGRAM:**

****