Software Requirements Specification

for

Optimized Football League Database

Version 1.0

Prepared by: Sachith V P, Arshabrata Bhaumik, Karthik Pandurang Pai

Batch 1

17th March 2025

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This SRS document defines the requirements for the **Optimized Football League Database**. *The purpose is to provide a complete description of the database system designed for managing comprehensive football league data—including match details, team and player information, statistical performance, and event tracking. This document is intended to guide the development, testing, and maintenance of the database while ensuring scalability, reliability, and high performance for advanced analytics and reporting.*

1.2 Document Conventions

- *Keywords & Notations:*
  - ***Primary Keys** are denoted in **bold**.*
  - *Foreign Keys are italicized.*
  - *SQL keywords appear in uppercase.*
  - *Data types and constraints (e.g., NUMBER, VARCHAR2, DATE, DEFAULT) are specified as per Oracle SQL conventions.*
- *Formatting:*
  - *Section headings and subheadings follow the IEEE style.*
  - *Requirement identifiers use the format REQ-[Module]-[Number] (e.g., REQ-M1 for a match management requirement).*
- *Terminology:*
  - *Terms like xG (Expected Goals) and club_id will be defined in the glossary.*

1.3 Intended Audience and Reading Suggestions

- ***Developers:** To understand the detailed structure of tables, relationships, and database constraints for system implementation.*
- ***Database Administrators (DBAs):** For insights into data integrity, performance optimization, and indexing strategies.*
- ***Project Managers:** To obtain an overview of functional and nonfunctional requirements, module breakdowns, and system integration points.*
- ***Data Analysts and Scientists:** To leverage the stored data for generating analytics reports and insights on team/player performance.*
- ***Quality Assurance Testers:** To plan test cases and validate both functional and nonfunctional requirements.*

1.4 Product Scope

The Optimized Football League Database is designed to:

- *Store detailed match data (scores, possession, expected goals, shots, etc.).*
- *Maintain comprehensive records on teams, players, and match lineups.*
- *Track match events (goals, assists, and related events).*
- *Provide a foundation for advanced performance analytics (e.g., calculating shot accuracy).*
- *Ensure data integrity via foreign key relationships, triggers, and sequences.*

- *Support future expansion including web-based administration, API integration, and machine learning analytics.*

1.5 References

Key references for this document include the IEEE SRS Template, Oracle SQL and PL/SQL documentation, and prior documentation from related football league database projects. These resources have informed the design and specifications of the current system.

2. Overall Description

2.1 Product Perspective

The database is a stand-alone system designed to integrate seamlessly with existing league management software and future analytical platforms. It uses a relational model to enforce data integrity, leverages normalization (3NF) to reduce redundancy, and implements triggers and sequences to automate key generation and data validation.

2.2 Product Functions

- ***Match Management:*** *Capture and store match details including match date, participating teams, scores, xG values, possession percentages, and shot metrics.*
- ***Player Management:*** *Record player demographics, physical attributes, club affiliations, positions, and performance statistics over multiple matches.*
- ***Team Management:*** *Manage teams with details such as names, home stadiums, coaching information, historical titles, fanbase metrics, and online presence.*
- ***Event Tracking:*** *Log all in-game events such as goals, assists, including the minute of occurrence, and link related events (e.g., assists to goals).*
- ***Performance Analytics:*** *Generate statistical summaries and provide insights on distance covered and shot accuracy.*
- ***Match Lineups:*** *Track starting players and substitutes for every match, including minutes played and positional data.*

2.3 User Classes and Characteristics

- ***League Officials:*** *Require read and write access to update match records and view summary reports.*
- ***Coaches and Analysts:*** *Use detailed match and performance data to strategize and improve team performance.*
- ***Developers:*** *Need comprehensive technical documentation for database design, maintenance, and feature enhancements.*
- ***Database Administrators:*** *Focus on system performance, backup strategies, data recovery, and ensuring referential integrity.*
- ***Data Scientists:*** *Utilize structured query access for in-depth analysis and predictive modeling.*

2.4 Operating Environment

- ***Database Platform:*** *Oracle SQLPlus (or similar Oracle Database environment).*
- ***Development Tools:*** *SQL Management Studio, DBeaver, and other SQL client tools.*
- ***Hosting Environment:*** *Deployed on AWS RDS or local server setups supporting Linux or Windows.*
- ***Network Requirements:*** *Stable high-speed connectivity for remote database management and query execution.*

2.5 Design and Implementation Constraints

The design of the Optimized Football League Database is governed by several constraints. Data integrity is maintained through strict primary and foreign key enforcement, while the system's structure adheres to 3NF to eliminate redundancy. Automated triggers and sequences are employed to manage unique key generation and data validation, and performance optimization is achieved through indexing and query tuning. The design must also be compatible with the Oracle SQL environment and allow for future expansion, such as API integration and web-based administration, without compromising the existing data model.

2.6 User Documentation

- ***User Manuals:*** *Detailed guides on database operations, query execution, and administrative procedures.*
- ***API Documentation:*** *(For future API integrations) to describe endpoints and data interchange formats.*
- ***Tutorials and Quick-Start Guides:*** *Step-by-step instructions for new users and developers.*
- ***Maintenance and Backup Guides:*** *Documentation for DBAs covering routine maintenance, backup procedures, and performance tuning.*

2.7 Assumptions and Dependencies

- ***Assumptions:***
  - *Users have a basic understanding of SQL and relational databases.*
  - *The underlying infrastructure (AWS RDS or local servers) is set up and maintained.*
  - *Adequate network security measures are in place to protect data integrity.*
- ***Dependencies:***
  - *Dependency on Oracle Database and its SQL engine.*
  - *Reliance on third-party tools for reporting and analytics integration.*
  - *Future enhancements may depend on additional hardware resources or integration with web-based interfaces.*

3. External Interface Requirements

3.1 User Interfaces

- ***Command-Line Interface (CLI):*** *Primary access through SQL queries using tools like SQLPlus.*

- ***Graphical User Interface (GUI):*** *Future web-based administrative panels for viewing and updating records.*
- ***Reporting Dashboards:*** *Integration with BI tools for data visualization and analytics reporting.*

3.2 Hardware Interfaces

The database is designed to operate on modern enterprise-grade server hardware, whether hosted on AWS RDS or local servers. It is compatible with the hardware typically found in data centers, including support for additional peripheral devices like backup drives and network-attached storage systems, which are critical for data redundancy and disaster recovery. The system's design ensures that it can scale to accommodate increases in data volume and transaction loads without hardware limitations becoming a bottleneck.

3.3 Software Interfaces

The Optimized Football League Database is built to integrate seamlessly with a variety of software components. It interfaces directly with Oracle-supported operating systems and middleware components, such as application servers and web services, ensuring smooth data exchange and interoperability. Additionally, the system is designed to work with external reporting and analytics tools, facilitating data extraction and integration with platforms like Tableau or PowerBI. Future API development will further enhance interoperability by allowing third-party applications to access and manipulate the database data.

3.4 Communications Interfaces

Communications with the database occur over standard TCP/IP protocols, ensuring compatibility with existing network infrastructure. Security measures include the use of SSL/TLS encryption for all remote communications, ensuring that data transmitted between clients and the database is secure. The system supports both real-time and batch data transfers, allowing for flexible synchronization between distributed systems while maintaining data integrity and security throughout the process.

4. System Features

4.1 Match Management

4.1.1 Description and Priority

The Match Management module is responsible for recording all match-related data with a high level of detail. Each match is assigned a unique identifier generated by an automated sequence, and all relevant details—including match date, home and away team identifiers, scores, possession percentages, shots, and advanced metrics such as expected goals (xG)—are stored in the system. Given its central role in both operational and analytical processes, this module is considered a high-priority feature.

4.1.2 Stimulus/Response Sequences

- ***Stimulus:*** *When a new match is scheduled, an administrator inputs match details.*
- ***Response:*** *The system validates team IDs (via foreign key constraints), generates a unique match_id, and records match statistics.*
- ***Stimulus:*** *Post-match, updated statistics (e.g., scores, possession) are submitted.*
- ***Response:*** *The system updates the record and triggers any associated analytical processes.*

4.1.3 Functional Requirements

- ***REQ-M1:*** *The system shall generate a unique match_id for each match using an appropriate sequence.*

- ***REQ-M2:*** *The system shall enforce foreign key constraints to ensure that home_team_id and away_team_id reference existing teams.*

- ***REQ-M3:*** *The system shall record match statistics including scores, xG values, possession percentages, and shot details.*

- ***REQ-M4:*** *The system shall support update operations post-match to modify statistics as needed.*

4.2 Player Management

4.2.1 Description and Priority

- ***Description:*** *Stores detailed player information such as personal details (first name, last name, age), club affiliation, playing position, nationality, and performance statistics.*

- ***Priority:*** *High. Accurate player data is essential for team management and performance analytics.*

4.2.2 Stimulus/Response Sequences

- ***Stimulus:*** *A new player is signed or an existing player's details are updated.*

- ***Response:*** *The system validates the input (e.g., ensuring club_id exists in the Teams table) and stores or updates the player record.*

- ***Stimulus:*** *A match is played and player performance statistics are generated.*

- ***Response:*** *The system updates the Player_Match_Stats table, and triggers recalcualtion of cumulative player statistics.*

4.2.3 Functional Requirements

- ***REQ-P1:*** *The system shall store a unique player_id along with jersey number, personal details (first_name, last_name, age), and visual identifiers (face_icon URL).*

- ***REQ-P2:*** *The system shall enforce that club_id in the Players table references an existing team in the Teams table.*

- ***REQ-P3:*** *The system shall support detailed recording of player attributes including position, nationality, birth_date.*

- ***REQ-P4:*** *The system shall record and update match-specific statistics (goals, assists, shots, passes, tackles, etc.) in the Player_Match_Stats table.*

- ***REQ-P5:*** *The system shall allow for historical tracking of matches_played and cumulative performance data.*

4.3 Team Management

4.3.1 Description and Priority

- ***Description:*** *Manages team data including team names, home stadiums, coach information, historical performance (e.g., previous premiership titles), fan base statistics, and digital assets like badges and team websites.*
- ***Priority:*** *High. Team data is central to establishing match contexts and linking with players and matches.*

4.3.2 Stimulus/Response Sequences

- **Stimulus:** A new team is added to the league.

- **Response:** The system verifies the uniqueness of the team name, assigns a unique team_id, and links the team to a home stadium and coach if available.

- **Stimulus:** Updates to team performance or fan metrics occur after matches.

- **Response:** The system updates the corresponding team record and ensures that these updates cascade to any linked analytical modules.

4.3.3 Functional Requirements

- ***REQ-T1:*** *The system shall generate a unique team_id for each team and enforce uniqueness of the team name.*

- ***REQ-T2:*** *The system shall store detailed team information including home_stadium_id, coach_id, number of previous premiership titles, fan count, badge (image URL), points, founded_year, and team_website.*

- ***REQ-T3:*** *The system shall enforce foreign key constraints for home_stadium_id and coach_id, ensuring these references are valid.*

- ***REQ-T4:*** *The system shall allow updates to team statistics post-match (e.g., points, fan count) and maintain historical records.*

4.4 Event Tracking

4.4.1 Description and Priority

- ***Description:*** *Captures all match-related events such as goals, assists, and any event that affects match outcome. The events are linked to the specific match, player, and team, with the ability to track relationships (e.g., linking an assist to a corresponding goal).*

- ***Priority:*** *Medium to High. Accurate event logging is essential for performance analysis and disciplinary reviews.*

4.4.2 Stimulus/Response Sequences

- ***Stimulus:*** *During a match, when an event occurs (e.g., goal scored, yellow card issued), the event is recorded in real time.*
- ***Response:*** *The system logs the event with a unique event_id, validates player and team associations, and updates cumulative statistics if applicable.*
- ***Stimulus:*** *A related event (e.g., an assist linked to a goal) is recorded.*
- ***Response:*** *The system creates a linkage using related_event_id for traceability.*

4.4.3 Functional Requirements

- ***REQ-E1:*** *The system shall generate a unique event_id for every event using a sequence mechanism.*
- ***REQ-E2:*** *The system shall record event details including match_id, player_id, team_id, event_type (e.g., GOAL, ASSIST, YELLOW_CARD, RED_CARD), minute of occurrence, and flags for penalty and own goals.*
- ***REQ-E3:*** *The system shall enforce that event_type values are restricted to a pre-defined set.*
- ***REQ-E4:*** *The system shall support linking related events (e.g., assist to goal) via a related_event_id field.*

4.5 Performance Analytics

4.5.1 Description and Priority

- ***Description:*** *Provides analytical functions that calculate performance metrics such as shot accuracy, distance covered, and other advanced statistics based on stored match and player data.*
- ***Priority:*** *Medium. While not essential for data storage, analytics add significant value for performance review and strategic planning.*

4.5.2 Stimulus/Response Sequences

- ***Stimulus:*** *A data analyst executes a query to generate a performance report.*
- ***Response:*** *The system aggregates data from match, player, and event tables, computes necessary analytics, and returns a detailed report within defined performance thresholds.*
- ***Stimulus:*** *A scheduled process recalculates season-long metrics for each team and player.*
- ***Response:*** *The system performs batch processing with indexing and caching to optimize query response time.*

4.5.3 Functional Requirements

- ***REQ-PA1:*** *The system shall calculate aggregate performance metrics (e.g., total goals, assists, shots on target) using data from Player_Match_Stats.*

- ***REQ-PA2:*** *The system shall support real-time calculation of advanced metrics such as shot accuracy, and distance covered.*

- ***REQ-PA3:*** *The system shall provide APIs or stored procedures to extract analytics data for integration with external BI tools.*

- ***REQ-PA4:*** *The system shall allow filtering by time period, team, or individual player to generate customizable reports.*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- ***Query Performance:*** *Average query response time should be under 2 seconds for common operations even with large datasets (e.g., 1M+ transactions/day).*
- ***Scalability:*** *The database should efficiently handle growth in data volume, ensuring that indexing and query optimization techniques remain effective.*
- ***Throughput:*** *Support for concurrent access by multiple users (DBAs, analysts) without performance degradation.*

5.2 Safety Requirements

- ***Data Integrity:*** *Use of primary/foreign key constraints, triggers, and transaction controls to maintain consistency.*
- ***Backup and Recovery:*** *Regular automated backups and a recovery plan in case of hardware failure or data corruption.*
- ***Failover:*** *Support for high availability via replication or clustering.*

5.3 Software Quality Attributes

- ***Reliability:*** *Ensure that the system maintains operational stability under peak loads and recovers quickly from failures.*
- ***Maintainability:*** *Clear documentation, modular design, and adherence to coding standards for ease of updates and debugging.*
- ***Portability:*** *The design should allow migration between Oracle environments and potential compatibility with other RDBMS in future.*
- ***Testability:*** *Comprehensive test plans covering unit, integration, and performance tests.*
- ***Usability:*** *While the primary interface is SQL-based, future GUIs should be intuitive and require minimal training.*

5.4 Business Rules

- ***Team Uniqueness:*** *Each team must have a unique name and a distinct team_id.*
- ***Match Composition:*** *Every match must involve two distinct teams (home and away).*
- ***Player Association:*** *A player can be affiliated with only one team at any given time.*
- ***Event Validity:*** *Event types are restricted to predefined categories (e.g., GOAL, ASSIST, YELLOW_CARD, RED_CARD) to maintain consistency in reporting.*
- ***Data Update Policy:*** *Changes to match, team, or player records after finalization must be logged and approved by administrators.*

6. Other Requirements

- ***Internationalization:*** *The system should be adaptable for multi-language support in future user interface modules.*
- ***Integration:*** *Future API endpoints may be developed to interface with external football analytics platforms and mobile applications.*
- ***Reporting Standards:*** *Data export formats should include CSV, JSON, and XML for compatibility with various reporting tools.*
- ***Legal and Compliance:*** *Ensure compliance with data protection regulations and intellectual property rights related to digital assets (e.g., team badges).*

Appendix A: Glossary

- ***xG:*** *Expected Goals – a statistical measure of the quality of scoring chances.*
- ***Primary Key:*** *A unique identifier for a record in a database table (e.g., match_id, player_id).*
- ***Foreign Key:*** *A field in a table that uniquely identifies a row in another table (e.g., home_team_id referencing Teams).*
- ***Trigger:*** *A database object that automatically executes predefined actions in response to certain events.*
- ***Sequence:*** *A database object that generates a sequence of unique numbers for primary keys.*

Appendix B: Analysis Models

- ***Entity-Relationship Diagrams (ERD):***
    - *Diagrams depicting relationships among Matches, Teams, Players, Player_Match_Events, Player_Match_Stats, and Match_Lineups.*
- ***Data Flow Diagrams (DFD):***
    - *Visual representations of how data moves between modules such as Match Management, Event Tracking, and Performance Analytics.*
- ***Architecture Diagram:***
    - *A high-level view of the database within the broader football league management system, showing integration points with web-based interfaces and external analytics tools.*

Appendix C: To Be Determined List

- ***User Interface Details:*** *Finalize design and usability studies for the planned web-based administrative interface.*
- ***API Specifications:*** *Define endpoints and protocols for future integration with third-party analytics platforms.*
- ***Reporting Templates:*** *Determine standard report layouts and data export formats.*
- ***Data Archiving Strategy:*** *Develop policies for long-term data retention and archival.*
- ***Internationalization Requirements:*** *Evaluate the need for multi-language support and regional customization.*