Football Championship Database

University:	Constructor University, Bremen
Course:	Data Management and Databases
Instructor:	Dr. Shahid Yousaf
Project Group:	Group 1
Team Members:	Ahmed Wezdar (awezdar@constructor.university)
	Amine Habbou (ahabbou@constructor.university)
	Mariem Nour Ghozzi (mghozzi@constructor.university)
	Shuyu Gui (sgui@constructor.university)
Date of Submission:	29.09.2025

Submission Date: September 27, 2025

1 Functionality of the Database (User Perspective)

This database supports the end-to-end management of a football league by organizing and updating information on **leagues, seasons, teams, persons (players, coaches, referees, staff), contracts, stadiums, matches, goals, assists, tickets, match statistics, and seasonal rankings**. It provides both operational workflows (scheduling, ticketing, contracting) and analytical views (statistics, rankings, histories), while enforcing key consistency rules derived from the conceptual ER model.

1.1 League & Season Management

- Maintain basic **League** information (name, country) and associated **Season**s (start_year, end_year).
- Each season belongs to exactly one league; users can browse historical seasons within a league.

1.2 Teams & Home Stadiums

- Maintain **Team** profiles (name, city, founded_year).
- Assign a default **home Stadium** (name, city, location, capacity); by default, home fixtures are scheduled there, while neutral venues remain possible when explicitly chosen.

1.3 Persons, Roles (ISA) & Contracts

- Consolidate all people in a unified **Person** registry (ID, name), specialized via an ISA hierarchy into **Player**, **Coach**, **Referee**, and **Staff**, each with role-specific attributes (e.g., player position/number, coach license_type, referee level, staff role).
- Manage **Contract**s linking Person and Team with validity windows (start/end dates), supporting creation, update, termination, and historical queries.

1.4 Match Scheduling & Statistics

- Schedule **Match** fixtures (date, home/away teams, venue) within a season; record final scores (home_score, away_score).

- Record a single **MatchStats** entry per match (e.g., possession home/away, shots home/away).

1.5 Goals & Assists

- Register **Goal**s (minute_scored, is_penalty) and, when applicable, an **Assist** (minute_assisted) attached to the goal.
- Aggregated goals determine the official match score for the home/away sides.

1.6 Ticketing & Venue Entry

- Sell **Ticket**s for a match hosted at a specific stadium; manage seat/section, price, purchase status.
- Support venue entry workflows (ticket validation), ensuring seat uniqueness per match and capacity adherence.

1.7 Seasonal Rankings & Reports

- Maintain **Ranking** per (Team, Season): position, points, matches_played, goals_for, goals_against.
- Provide standings, match reports, and player/club statistics across seasons for analytical and historical views.

1.8 Consistency & Business Rules (Selected)

- **Contracts:** For a given time instant, a player/coach must not hold overlapping active contracts for multiple teams in the same role; team assignment at match time is derived from contract validity.
- **Scheduling:** A team (and stadium) cannot be double-booked at overlapping times; home and away must be different teams.
- **Scoring:** The sum of goals recorded for each side equals the match's final score; assists (if present) belong to a specific goal and occur no later than the goal minute.
- **Statistics:** Match statistics use non-negative values and reasonable bounds (e.g., possession near 100% total with tolerance).

- **Ticketing:** Seats are unique per match; tickets sold must not exceed the stadium's capacity; invalid/duplicate tickets are rejected at entry control.

2 ER Diagram of the Miniworld (with ISA)

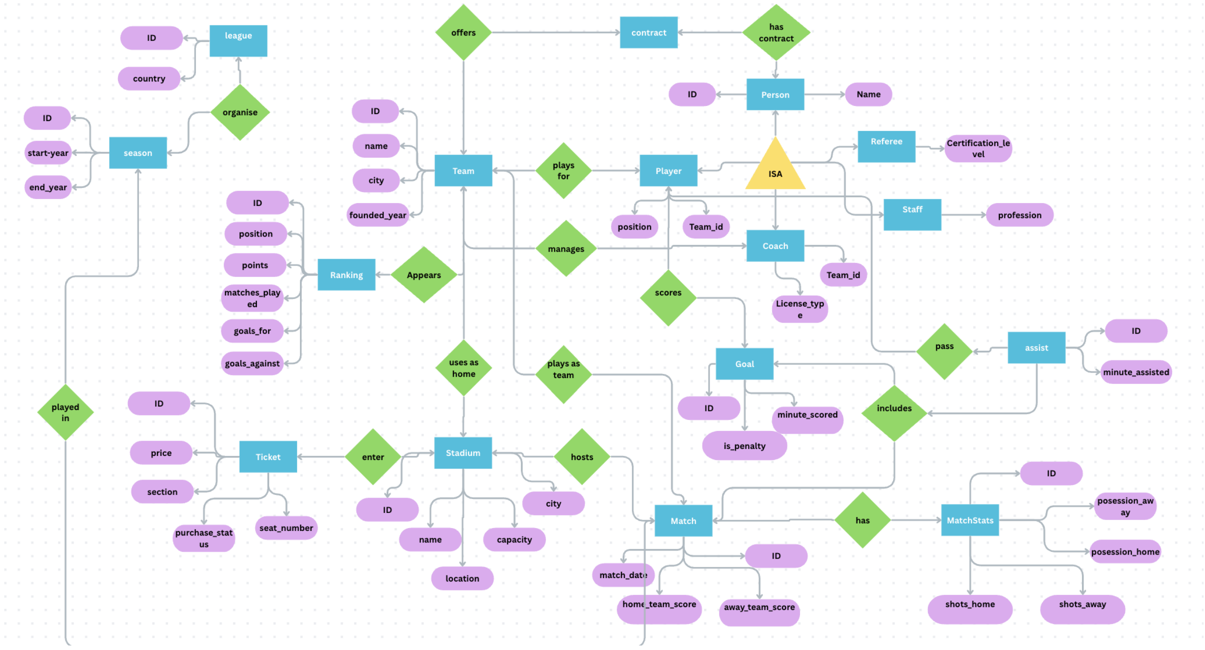

Figure 1: Entity-Relationship Diagram of the Football Championship Database (with ISA hierarchy).

3 User Interactions with the Database System

User Action	**Inputs / UI**	**System Output / State Change**	**Illegal Input / Error Handling**
Create / Update League & Season	League name, country; Season start_year, end_year	League created/updated; Season linked to League; history queries enabled	Invalid year range; duplicate Season within same League → reject
Team CRUD & set Home Stadium	Team (name, city, founded_year); choose default Stadium	Team stored/updated; relation "uses as home" established	Unknown Stadium; negative founded_year; duplicate Team name in League → reject
Register Person (ISA)	Person basics (ID, name); subtype attrs: Player (position, number), Coach (license_type), Referee (level), Staff (role)	Person stored; subtype row created; visible in rosters/search	Duplicate person ID; missing subtype attributes → reject
Create / Terminate Contract	Person, Team, start_date, end_date, role	Contract created or terminated; history kept	Overlapping contracts for same person/role; start ≥ end → reject
Appoint / Change Head Coach	Team, Coach, validity window	Coach "manages" Team for window; previous coach closed	Two head coaches overlap in time; invalid license_type → reject
Player Transfer (Team assignment by contract)	Player, new Team, contract dates	Player's active team switches per contract validity	Contract gap at match time; overlapping team contracts → reject

(continued on next page)

User Action	**Inputs / UI**	**System Output / State Change**	**Illegal Input / Error Handling**
Plan Match (Schedule)	Season, date/time, home Team, away Team, Stadium	Match record created in Season; defaults to home Stadium unless overridden	Home = Away; team or stadium double-booked at overlapping times → reject
Enter Match Statistics	poss_home/away, shots_home/away, foulsH/A	One MatchStats row created/updated and linked to Match	Negative values; possession not near 100% total (beyond tolerance); match not found → reject
Record Goal	Match, scoring Player, minute, is_penalty	Goal stored; aggregates into Match home/away totals	Player not belonging to participating side at match time; minute exceeds duration → reject
Record Assist	Goal id, assisting Player, minute_assisted	Assist linked to Goal; analytics updated	Assist minute > goal minute; assisting Player not on scoring side → reject
Sell Ticket (Seat / Section)	Match, section, seat_number, price	Ticket created; seat reserved; capacity accounted	Seat already sold for this match; total sold > stadium capacity → reject
Validate Entry (Gate)	Ticket scan (match + seat)	purchase_status updated (entered); audit log	Ticket not for this match/stadium; invalid/duplicate scan → deny entry
Update Standings	Submit final score / confirm goals	Ranking row for (Team, Season) updated: points, matches_played, GF/GA	Sum of recorded goals ≠ submitted score; season/team missing → reject

(continued on next page)

User Action	**Inputs / UI**	**System Output / State Change**	**Illegal Input / Error Handling**
Reports & Queries	Filters by league/season/team/player/referee/stadium	Standings, fixtures, match reports, top scorers/assists	No data found → friendly “no results” notice