



Africa Cup of Nations **Mobile App & API** Project

2025/2026 Tournament Management System



React Native



REST API




PostgreSQL

Project Overview

Mobile App Prototype for 2025/2026 Africa Cup of Nations

 **24** participating national teams

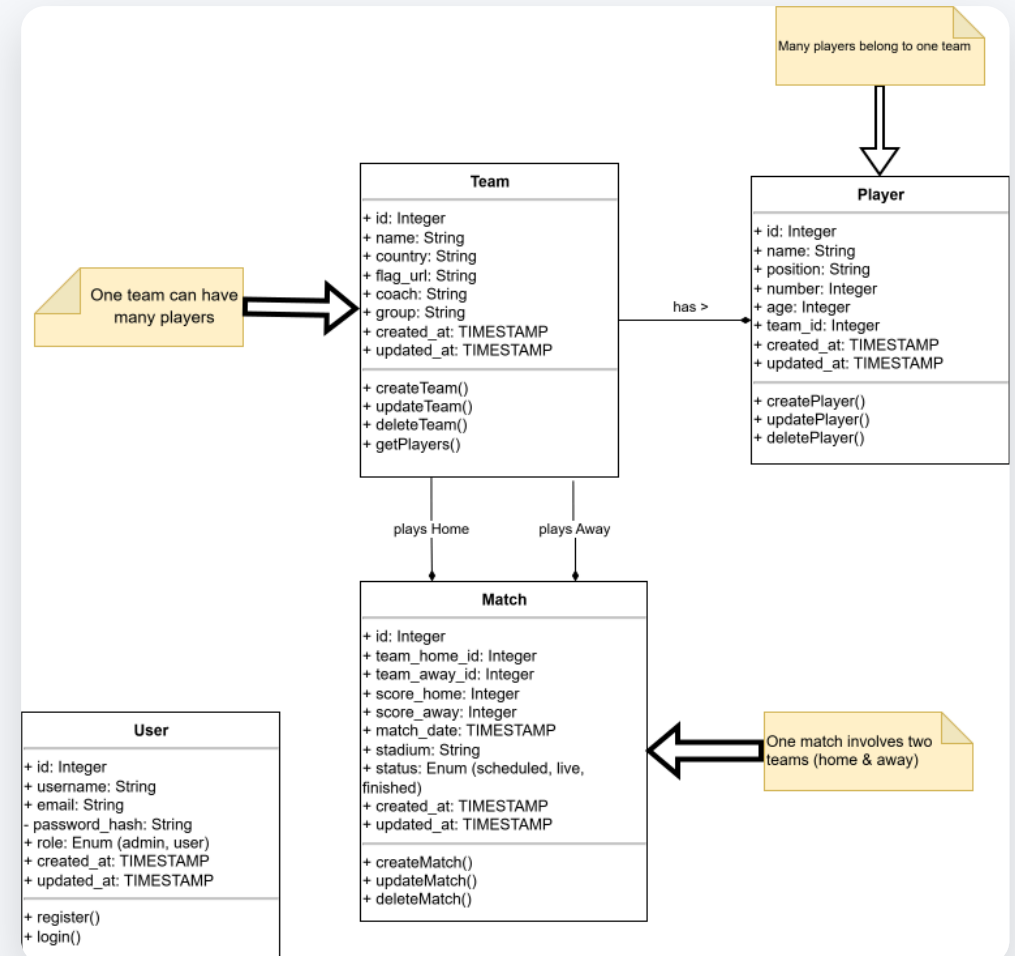
 Matches across **multiple host cities**

 **Real-time** results for millions of fans

 Complete **team & player** management

 Integrated **ticketing** system

UML Class Diagram - System Architecture



Current Situation

 **No backend API** - All data is stored locally

 **Scattered data** across multiple Excel files

 **Urgent need** before tournament starts

 **Multiple administrators** need data access

From Chaos to Organization

Current System



- ❗ Manual data entry
- ❗ No real-time updates
- ❗ Inconsistent formatting
- ❗ Limited accessibility

Proposed Solution



- ✅ Centralized database
- ✅ Real-time synchronization
- ✅ Structured data model
- ✅ Secure multi-user access

Database Model

Entity Relationship Diagram (ERD)

👤 users		
id	SERIAL	PRIMARY KEY
username	VARCHAR	UNIQUE
email	VARCHAR	UNIQUE
password	VARCHAR	HASHED
role	ENUM	admin/user
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

👥 teams		
id	SERIAL	PRIMARY KEY
name	VARCHAR	UNIQUE
country	VARCHAR	
flag_url	VARCHAR	
coach	VARCHAR	
group	VARCHAR	A,B,C...
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

⚽ players		
id	SERIAL	PRIMARY KEY
name	VARCHAR	
position	VARCHAR	
number	INTEGER	
age	INTEGER	
team_id	INTEGER	FOREIGN KEY
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

📅 matches		
id	SERIAL	PRIMARY KEY
team_home_id	INTEGER	FOREIGN KEY
team_away_id	INTEGER	FOREIGN KEY
score_home	INTEGER	
score_away	INTEGER	
match_date	TIMESTAMP	
stadium	VARCHAR	
status	ENUM	scheduled/live/finished
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

➔ One-to-Many: Team has many Players

➔ One-to-Many: Team plays many Matches

➔ Many-to-One: Match involves two Teams

API Architecture

🛡️ Authentication Routes

POST	/api/auth/register	Registration	
POST	/api/auth/login	Login	
GET	/api/auth/profile	User profile	Protected 🔒

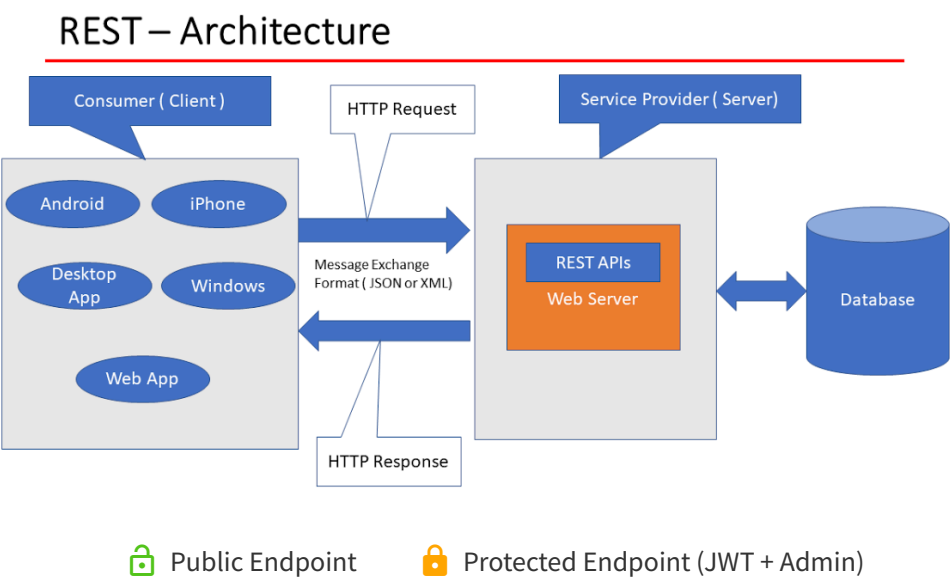
👥 Teams Routes

GET	/api/teams	Lists all teams	
GET	/api/teams/:id	Team details	
POST	/api/teams	Create a team	Admin 🔒
PUT	/api/teams/:id	Edit a team	Admin 🔒
DELETE	/api/teams/:id	Delete a team	Admin 🔒

⚽ Players & Matches Routes

GET	/api/players	List all players	
POST	/api/players	Create a player	Admin 🔒
GET	/api/matches	List all matches	
GET	/api/matches/upcoming	Upcoming matches	
POST	/api/matches	Create a match	Admin 🔒

REST API Architecture



Team Organization

Development Team Structure

Developer 1

Developer 2





Developer 3

Developer 4



Developer 1





Architecture & Database

-  **UML diagrams** (use cases, classes, sequence, ERD)
-  **PostgreSQL** configuration
-  Creating models with **ORM**
-  Migrations and table relationships



Developer 2





Authentication System

-  Register, login, logout functionality
-  **JWT Tokens** implementation
-  **Bcrypt** for password hashing
-  Authentication and role middleware



Developer 3





CRUD API Teams & Players

-  Install **Expo** and create project
-  React Native and Expo tutorials
-  CRUD operations for **Teams**
-  CRUD operations for **Players**



Developer 4

CRUD Matches & Documentation






-  CRUD operations for **Matches**
-  Match officials management
-  **API Documentation** (Postman Collection)
-  Deployment and README creation

Development Timeline






5-Day Project Schedule

Efficiently organized tasks to deliver the Africa Cup of Nations mobile app and API






1 Setup & Discovery

-  Install **Expo** and create project
-  React Native tutorial (10 min)
-  Expo tutorial (10 min)
-  Expo Router tutorial (15 min)
-  Basic app **wireframe**






2 Architecture & Database

-  Create **UML diagrams**
-  Configure PostgreSQL
-  Design database models
-  Set up table relationships
-  Create initial **migrations**






3 Authentication System

-  Implement **register** endpoint
-  Implement login endpoint
-  JWT token generation
-  Bcrypt password hashing
-  Role-based middleware

4 CRUD API Teams & Players

-  **Teams** CRUD endpoints
-  **Players** CRUD endpoints
-  Connect frontend to API
-  **useState** implementation
-  **useEffect** for data loading

5 CRUD Matches & Documentation

-  **Matches** CRUD endpoints
-  Match officials management
-  API documentation
-  Testing and debugging
-  Deployment and submission

Technical Stack & Skills Development

<> Required Technologies



React Native



Expo



Node.js



Express.js



PostgreSQL



Sequelize/Prisma



JWT



Bcrypt

🎓 Skills to Develop



Complete UML design



Professional MVC Backend Architecture



PostgreSQL and relational database management



Use of an ORM (Sequelize or Prisma)



JWT authentication and security



Password hashing with Bcrypt



Full CRUD functionality with Express.js



Professional API documentation with Postman



Deploying APIs in production



Teamwork with Git