Sistema de Torneos e Competencias Deportivas

**Nome Alumno/a:**

***Jorge Sobrino Mojón***

**Curso: *2º DAM* Materia:** ***Proxecto Final Módulo Acceso a Datos***

**Docente:** ***Javier Feijóo López***

***🗐 Proxecto: https://github.com/Jsobrino98/ProyectoFinalAD***

Contido

[1. Introdución 2](#_Toc189548783)

[2. Deseño da Base de Datos – Modelo EER 2](#_Toc189548784)

[3. Tecnoloxías Empregadas e Configuración da Contorna 2](#_Toc189548785)

[4. Deseño do Backend – API REST con Spring Boot 2](#_Toc189548786)

[5. Implementación de Autenticación e Seguridade 3](#_Toc189548787)

[6. Desenvolvemento do Backend con Thymeleaf 3](#_Toc189548788)

[7. Documentación da API e Probas en Postman 3](#_Toc189548789)

[8. Conclusións e Valoración Persoal 3](#_Toc189548790)

[9. Anexos (Opcional) 3](#_Toc189548791)

# 1. Introdución

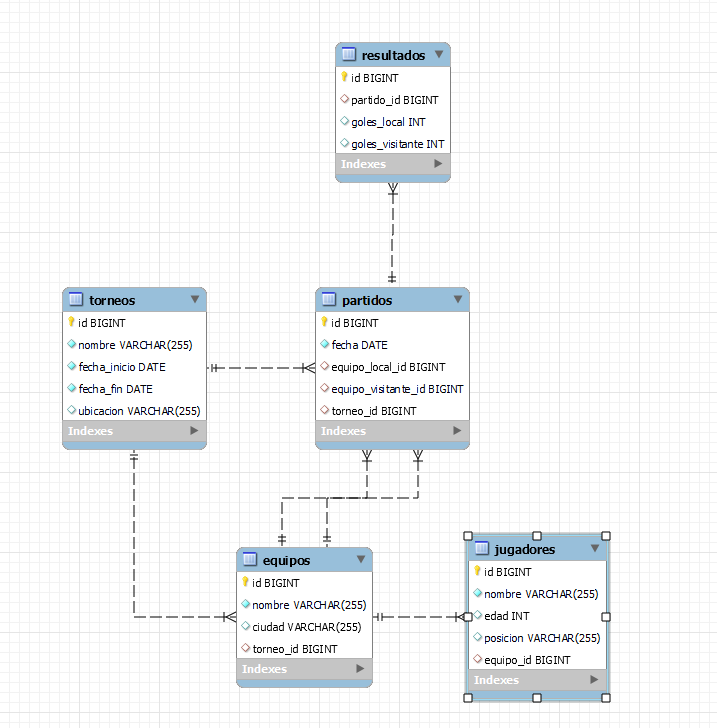
Breve presentación do proxecto, respondendo a preguntas clave:

* Cal é a finalidade do sistema?
* Que problema resolve?
* Quenes son os usuarios principais?
* Resumen das tecnoloxías empregadas.

# 2. Deseño da Base de Datos – Modelo EER

Explicación do deseño da base de datos, incluíndo:

* Modelo Entidade-Relación Estendido (EER) cunha imaxe.



* Explicación de cada entidade e das súas relacións.
* Descrición das claves primarias e foráneas.
* Restricións e regras de integridade.

**Equipos**

**id**: Identificador único del equipo.

**nombre**: Nombre del equipo.

**ciudad**: Ciudad del equipo.

**entrenador**: Nombre del entrenador.

Relaciones:

Relacionado con jugadores a través de equipo\_id.

Relacionado con partidos como equipo\_local\_id y equipo\_visitante\_id.

**Jugadores**

**id**: Identificador único del jugador.

**nombre**: Nombre del jugador.

**edad**: Edad del jugador.

**equipo\_id**: Identificador del equipo al que pertenece (clave foránea a equipos).

**Torneos**

**id**: Identificador único del torneo.

**nombre**: Nombre del torneo.

**fecha\_inicio**: Fecha de inicio del torneo.

**fecha\_fin**: Fecha de finalización del torneo.

Relaciones:

Relacionado con partidos a través de torneo\_id.

**Partidos**

**id**: Identificador único del partido.

**torneo\_id**: Identificador del torneo (clave foránea a torneos).

**equipo\_local\_id**: Identificador del equipo local (clave foránea a equipos).

**equipo\_visitante\_id**: Identificador del equipo visitante (clave foránea a equipos).

**fecha**: Fecha del partido.

**resultado**: Resultado del partido.

Relaciones:

Relacionado con resultados a través de partido\_id.

**Resultados**

**id**: Identificador único del resultado.

**partido\_id:** Identificador del partido (clave foránea a partidos).

**goles\_equipo\_local**: Goles del equipo local.

**goles\_equipo\_visitante:** Goles del equipo visitante.

# 3. Tecnoloxías Empregadas e Configuración da Contorna

Explicación das ferramentas empregadas:

* Linguaxe de programación: Java 17 / 21.
* Framework: Spring Boot 3.
* Base de datos: H2 (embebida) / MySQL / PostgreSQL.
* ORM: Spring Data JPA.
* Autenticación: Spring Security + JWT.
* Frontend para administradores: Thymeleaf + Bootstrap.
* Ferramentas de documentación: Swagger, Postman.
* Xestión de dependencias: Maven.

Incluír pasos para configurar o contorno:

* IDE recomendado (IntelliJ, Eclipse, VS Code).
* Como executar a aplicación (mvn clean package, java -jar, ou capturas das accións).

# 4. Deseño do Backend – API REST con Spring Boot

Explicación da arquitectura do backend:

* Estrutura do proxecto en Spring Boot (paquetes e clases principais).
* Explicación dos controladores e servizos.
* Implementación de CRUD con exemplos de código.

# 5. Implementación de Autenticación e Seguridade

Explicación da seguridade no sistema:

* Rexistro e inicio de sesión con JWT.
* Protección de endpoints con roles (USER, ADMIN).
* Encriptación de contrasinais con BCrypt.

# 6. Desenvolvemento do Backend con Thymeleaf

Explicación do panel de administración con Thymeleaf:

* Vistas e plantillas HTML.
* Formularios de inicio de sesión e xestión de datos.
* Controladores MVC e servizos para a web.

# 7. Documentación da API e Probas en Postman

Incluir exemplos de endpoints documentados con Swagger e Postman:

* Capturas de Swagger UI.
* Explicación da colección de Postman e exemplos de respostas.

# 8. Conclusións e Valoración Persoal

Reflexión sobre o desenvolvemento do proxecto:

* Dificultades atopadas e como se resolveron.
* Que se podería mellorar en futuras versións.
* Opinión sobre o uso de Spring Boot e das súas tecnoloxías.

# 9. Anexos (Opcional)

Se o proxecto inclúe documentación adicional, achegar:

* Estrutura JSON de respostas da API.
* Código SQL para crear a base de datos.
* Instrucións adicionais de despregue nun servidor.de despliegue en un servidor.