

# Proyecto fin de ciclo CFGS Desarrollo de Aplicaciones Web



Autor: Alejandro Lapeña

Desarrollo de Aplicaciones Web (DAW)

Tutor del proyecto: Daniel Godoy

Curso Académico: 2022/2023

Fecha de exposición: 12/06/2023

Centro de estudios: Serra Perenxisa

## Índice del proyecto

1. Introducción
  1. Resumen
    - i. Summary
    - ii. Resum
  2. Relación del proyecto con el ciclo
  3. Objetivo
  4. Estado del arte
    - i. Lado cliente
    - ii. Lado servidor
  5. Análisis FADO
  6. Recursos de la aplicación
    - i. Recursos hardware
    - ii. Recursos software
    - iii. Recurso de tiempo
2. Sistema de información: Ligas de fútbol
  1. Diseño de la base de datos
  2. Relaciones de las tablas
  3. Diagrama de clases
  4. Roles con Spatie
3. Análisis de requisitos
  1. Descripción de requisitos
  2. Diagrama de casos de uso
    - i. Administrador
    - ii. Visitante

- iii. Sin Rol
- 3. Documentación de actores
- 4. Aplicación
  - 1. Página principal
    - i. Que ve cada usuario en la vista principal
      - 1. Usuario sin permisos
      - 2. Usuario con permisos
  - 2. Vista “TFG”
    - i. Header
      - 1. Vista responsiva de la vista tfg
    - ii. Menú de información de la administración del proyecto
    - iii. Tablas
  - 3. Página “/Admin”
    - i. ¿Qué ve el usuario con rol Visitante?
    - ii. ¿Qué ve el usuario con rol Administrador?
  - 4. Roles
  - 5. CRUD
    - i. Create
    - ii. Read
    - iii. Update
    - iv. Delete
  - 6. Usuarios
  - 7. Tabla de Peticiones
- 5. Implantación
- 6. Conclusiones

7. Bibliografía

8. Anexos

1. Anexo A

2. Anexo B

3. Anexo C

## 1. Introducción

El presente documento describe el trabajo realizado en el proyecto final del ciclo de desarrollo de aplicaciones web y realiza un estudio de un sistema gestor de base de datos en MySQL, el diseño de una base de datos de ligas de fútbol y la implementación de una aplicación en Laravel para la administración y consulta de dichos datos. Además utiliza Bootstrap consiguiendo el diseño responsivo y JavaScript para interactividad con los usuarios

### 1.1 Resumen

El proyecto consiste en la investigación y desarrollo del sistema de gestión de base de datos en MySQL, creación de una base de datos de ejemplo implementado sobre dicho sistema gestor de base de datos y un desarrollo en Laravel que permite administrar y dotar de contenidos a la base de datos.

El objetivo del proyecto es obtener una Información completa sobre este sistema gestor y los servicios de los que dispone, así como proporcionar una serie de comparativas y un estudio detallado del proceso de instalación y de las diferentes opciones de configuración con las que cuenta. En segunda instancia la creación de la base de datos nos ofrece un ejemplo práctico del funcionamiento de este sistema gestor y las diferentes plataformas compatibles las cuales se utilizarán como un ejemplo de aplicación de la base de datos creada. En última instancia un desarrollo en Laravel a medida para la gestión de la base de datos y el estudio y la adaptación de una plantilla Bootstrap que me permite visualizar el contenido adaptado a las dimensiones de los diferentes dispositivos del mercado. Existirá una serie de ficheros con JQuery para la interactividad con el usuario

## **1.1 Summary**

The present project consists of the research and development of a database management system in MySQL, creation of an example database implemented on this database management system and a development in Laravel that allows to manage and provide content to the database.

The objective of the project is to obtain complete information about this management system and its services, as well as to provide a series of comparisons and a detailed study of the installation process and the different configuration options available. Secondly, the creation of the database offers us a practical example of the functioning of this management system and the different compatible platforms which will be used as an example of application of the created database. Ultimately a development in Laravel to measure for the management of the database and the study and adaptation of a Bootstrap template that allows me to display the content adapted to the dimensions of the different devices on the market. There will be some files with JQuery for user interactivity.

### **1.1.2 Resum**

El projecte consisteix en la investigació i desenvolupament del sistema de gestió de base de dades a MySQL, creació d'una base de dades d'exemple implementat sobre aquest sistema gestor de base de dades i un desenvolupament a Laravel que permet administrar i dotar de continguts a la base de dades.

L'objectiu del projecte és obtenir una informació completa sobre aquest sistema gestor i els serveis de què disposa, així com proporcionar una sèrie de comparatives i un estudi detallat del procés d'instal·lació i de les diferents opcions de configuració de què disposa. En segona instància, la creació de la base de dades ens ofereix un exemple pràctic del funcionament d'aquest sistema gestor i les diferents plataformes compatibles que s'utilitzaran com a exemple d'aplicació de la base de dades creada. En última instància un desenvolupament a Laravel a mida per a la gestió de la base de dades i l'estudi i l'adaptació d'una plantilla Bootstrap que em permet visualitzar el contingut adaptat a les dimensions dels diferents dispositius del mercat. Hi haurà una sèrie de fitxers amb JQuery per a la interactivitat amb l'usuari

## **1.2 Relación del proyecto con el ciclo**

El presente proyecto requerirá de conocimientos adquiridos durante los 2 años del ciclo superior cursado en los siguientes ámbitos:

- El proyecto está directamente relacionado con el primer curso del ciclo formativo en el módulo de bases de datos. La implementación se ha realizado con el sistema gestor de bases de datos MySQL.
- El proyecto también está relacionado con el módulo “Desarrollo Web en Entorno Servidor”. Este proyecto se basa en una base de datos de prueba pero administrada en Laravel.
- En tercer lugar tiene conexión con el módulo “Diseño de Interfaces Web” ya que utiliza Bootstrap para conseguir el diseño responsivo apto para todo tipo de dispositivos.
- Y por último lugar tiene asonancia con el módulo “Desarrollo Web en Entorno Cliente”, se trata del módulo donde hice el estudio de JQuery y JavaScript

## **1.3 Objetivo**

**Ligas más influyentes:** Es un proyecto creado en Laravel con una base de datos de ejemplo en un sistema gestor de base de datos MySQL, con el objetivo de entender mejor como funciona el framework que estudiamos en este curso. En este podremos encontrar análisis de datos sobre algunos elementos en el mundo del futbol como: equipos, ligas, temporadas,etc.. En el que experimentaremos un CRUD con cada una de las tablas que conforman la base de datos y un sistema de roles y permisos que limitan que ve cada usuario del programa. El programa tiene un sistema de login que permite registrarse o iniciar sesión en la aplicación y según el rol que se tenga en ese momento podremos visualizar nada, una vista, una tabla o todo el proyecto. Además el usuario visitante siempre tendrá un sistema de correo interno para pedir roles o permisos a este superusuario

## 1.4 Estado del arte

### 1.4.1 Lado Cliente

#### 1.4.1.1 Librería JQuery

jQuery se utiliza principalmente para mejorar la experiencia del usuario en la interfaz de un sitio web al permitir la manipulación dinámica de elementos HTML y la interacción con el usuario sin necesidad de recargar la página completa. Al ser una biblioteca de JavaScript, jQuery se ejecuta en el navegador del cliente y utiliza el poder del lenguaje JavaScript para realizar las operaciones deseadas.

El uso de esta librería es imprescindible para la realización del proyecto, ya que consigue que las vistas, controladores y eventos sean totalmente interactivos.





#### 1.4.1.2 Bootstrap

Para conseguir un diseño adaptable a cada dispositivo existente en el mercado.



#### 1.4.2 Lado Servidor

##### 1.4.2.1 Laravel

Como software en entorno servidor se ha utilizado Laravel, que se trata de un framework de PHP. Permite entre otras cosas la creación sencilla de un sistema de autenticación, comunicarse con la base de datos mediante el ORM Eloquent o el uso de Query Builder, creación de validaciones, etc... Laravel es Open Source y está en continuo crecimiento, además de tener una amplia comunidad.



#### 1.4.2.2 PhpMyAdmin

Es el sistema gestor de base de datos cuando utilizamos bases de datos junto con PHP y Xampp.



#### 1.4.3 Otras tecnologías

**Visual Studio Code:** El editor de código con el que hice el programa.

**Xampp:** Entorno de desarrollo en local.

**Draw.io:** Para el diseño de mockups y diagramas ER y UML.

### 1.5 Análisis DAFO

Fortalezas:

1. Framework robusto: Laravel es uno de los frameworks más utilizados y con más apoyo en la comunidad, lo que significa que hay muchas herramientas y recursos disponibles para los desarrolladores.
2. Modularidad: Laravel está diseñado para ser modular, lo que significa que se pueden crear componentes y módulos específicos para la aplicación de fútbol, como tablas de posiciones, calendarios, resultados y estadísticas de jugadores.
3. Facilidad de uso: Laravel es fácil de usar y aprender para los desarrolladores, lo que puede acelerar el tiempo de desarrollo y reducir los costos.
4. Base de datos y migraciones: Laravel incluye herramientas para la creación de bases de datos y migraciones, lo que permite una fácil gestión de los datos de la aplicación.
5. Autenticación: Laravel tiene un sistema de autenticación integrado que se puede personalizar para las necesidades específicas de la aplicación.

### Debilidades:

1. Curva de aprendizaje: Si bien Laravel es fácil de aprender, la curva de aprendizaje para un desarrollador nuevo en el framework puede ser alta, especialmente si no tienen experiencia previa en Laravel.
2. Flexibilidad limitada: Laravel es un framework opinionado, lo que significa que puede ser menos flexible en comparación con otros frameworks. Esto puede limitar la capacidad de personalizar la aplicación según las necesidades específicas del cliente.
3. Problemas de escalabilidad: Si la aplicación se vuelve muy popular, puede haber problemas de escalabilidad debido a las limitaciones de rendimiento de Laravel.

### Oportunidades:

1. Mercado en crecimiento: El fútbol es uno de los deportes más populares en todo el mundo, lo que significa que hay una gran oportunidad de mercado para una aplicación de ligas de fútbol.
2. Monetización: Hay varias formas de monetizar una aplicación de ligas de fútbol, como la publicidad, la venta de entradas, la venta de merchandising y las suscripciones.
3. Personalización: Los equipos y las ligas pueden tener necesidades específicas, lo que significa que hay oportunidades para personalizar la aplicación según sus necesidades.

### Amenazas:

1. Competencia: Hay muchas aplicaciones y sitios web que ofrecen información sobre ligas de fútbol, por lo que la aplicación deberá diferenciarse para destacar en un mercado competitivo.
2. Cambios en el mercado: Los gustos y necesidades de los usuarios pueden cambiar con el tiempo, lo que significa que la aplicación debe estar en constante evolución para mantenerse relevante.
3. Cambios en la tecnología: Las tendencias tecnológicas pueden cambiar rápidamente, lo que significa que la aplicación debe estar al día con las últimas tendencias para mantenerse relevante y competitiva.

## 1.6 Recursos de la aplicación

### Recursos de hardware

Para la aplicación no hace falta unos requisitos muy elevados, con cualquier ordenador se podría ejecutar el programa. Lo único que necesitas es 2GB de Ram, un teclado, un ratón un procesador no muy potente y un espacio en disco de 500MB

Para la máquina virtual, se ha utilizado una suscripción de Azure con los requisitos de hardware más básicos es suficiente: 1 núcleo, 3,6GB de ram y 30GB por 0.015 €/hora

### Recursos de software

Todo el software empleado para el desarrollo es gratuito y open source:

Laravel, JQuery, Bootstrap, Visual Studio Code, Xampp, etc...

### Recurso de tiempo

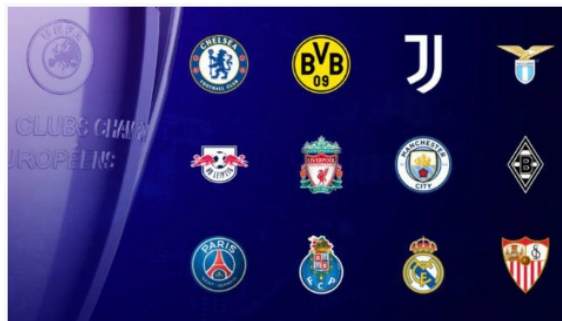
El recurso del tiempo es óptimo, son 3 meses para desarrollar la aplicación. Sin embargo hay peculiaridades que acortan el tiempo, como el tiempo de trabajo en las prácticas, la organización del tiempo en las diferentes partes del proyecto o el tiempo de estudio de asignaturas pendientes.

Por ello el proyecto se realiza poco a poco durante el tiempo establecido

## 2. Sistema de Información: Liga de fútbol

Descripción del funcionamiento de la base de datos de ligas creada en MySQL se compone de 8 tablas y estas son:

- 1 Equipos:** La tabla principal de la base de datos que otorga Información de los mejores equipos del mundo. Cada equipo tendrá muchos jugadores en una temporada.



Equipos

Los principales equipos de las ligas europeas

- 2 Ligas:** Tabla basada en las mejores ligas del mundo. Cada liga tendrá un equipo campeón en cada temporada.



Ligas

Las ligas mas importantes y relevantes de europa

- 3 Temporadas:** Esta tabla fundamenta un histórico de las temporadas existentes.



Temporadas

Una tabla que muestra el origen de las ligas

- 4 **Traspasos:** Histórico de cada jugador en cada temporada.



Traspasos

Una tabla que muestra los clubes a los que han pertenecido los Jugadores

- 5 **Capitanes:** Esta tabla explica el capitán de un equipo en una temporada.



Capitanes

Los capitanes de los equipos

- 6 **Jugadores:** Tabla que sustenta los jugadores que forman una plantilla. Un jugador puede ser traspasado, ser capitán o simplemente ser un jugador.



Jugadores

Una tabla dedicada a los capitanes de los equipos antes mencionados

**7 Campeones:** Histórico de equipos campeones en una temporada.



Campeones

Se trata de una tabla dedicada a el equipo campeon de una liga en una temporada

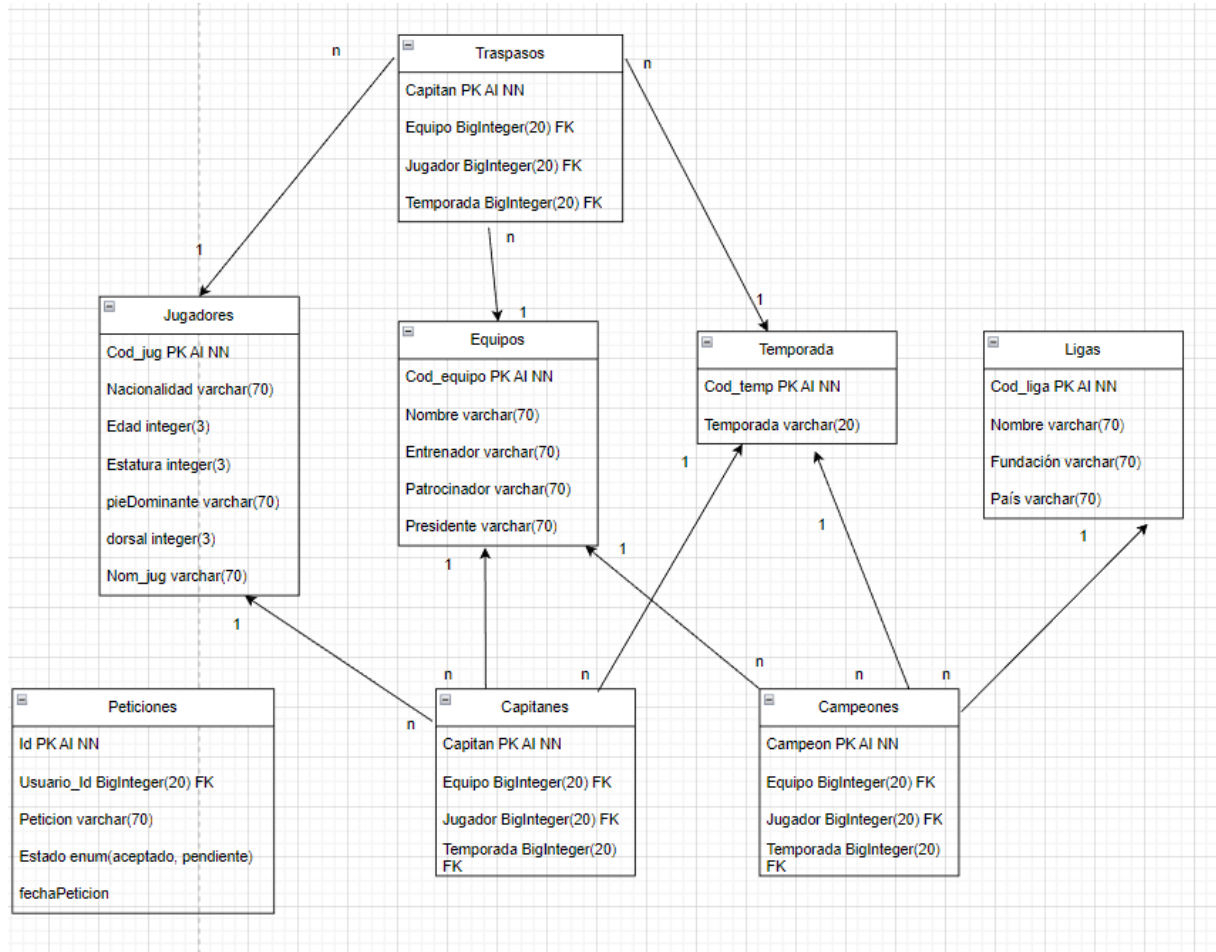
**8 Peticiones:** Dedicada a que los usuarios visitantes puedan pedir roles



Usuarios

Los usuarios que pueden acceder a la aplicación

## 2.1 Diseño de la base de datos



## 2.2 Relaciones de las tablas

- **Capitanes: Será un Jugador de un Equipo en una Temporada**

Capitan Integer Not Null Autoincrement PK +  
 Equipo FK (Relación uno a muchos con cod\_equipo(Equipos)) +  
 Jugador FK (Relación uno a muchos con cod\_jug(Jugadores)) +  
 Temporada FK (Relación uno a muchos con temporada(Temporadas)).



- Equipos: **Cada Equipo tendrá muchos Jugadores en una Temporada.**  
 Cod\_equipo Integer Not Null Autoincrement PK +  
 Nombre Varchar(40)  
 Entrenador Varchar(50) +  
 Patrocinador Varchar(20) +  
 Presidente Varchar(40).
- Ligas: **Cada Liga tendrá un Equipo Campeón en cada temporada**  
 Cod\_Lig Integer Not Null Autoincrement PK +  
 Nombre Varchar(30) +  
 Fundación Integer +  
 País Varchar2(40).
- Jugadores: **Un Jugador puede ser Capitán o ser un Traspaso**  
 Cod\_Jug Integer Not Null Auto\_increment PK +  
 Nacionalidad Varchar2(40) +  
 Edad Integer +  
 Estatura Integer +  
 PieDominante Varchar(40) +  
 Dorsal Integer +  
 Nom\_jug Varchar(40).
- Temporadas: **Histórico de Temporadas**  
 Cod\_temp Integer Not Null Autoincrement PK +  
 Temporada Varchar(20).
- Traspasos: **Histórico de un Jugador en una Temporada**  
 Traspaso Integer Not Null Autoincrement PK +  
 Equipo Integer PK Not Null + FK (Relación uno a muchos con  
 cod\_equipo(Equipos)) +  
 Jugador Integer PK Not Null + FK (Relación uno a muchos con  
 cod\_jug(Jugadores)) +  
 Temporada PK Not Null + FK (Relación uno a muchos con  
 cod\_temp(Temporadas)).
- Campeones: **Histórico de un Equipo Campeón en una Temporada**  
 Campeon Not Null Autoincrement PK +  
 Cod\_temp FK (Relación uno a muchos con Temporada (Temporadas)) +

Liga FK (Relación uno a muchos con Cod\_Lig (Ligas)) +  
Equipo FK (Relación uno a muchos con cod\_equipo (Equipos)).

● Peticiones: **Peticiones de usuarios visitantes**

Id Integer Not Null Autoincrement PK +

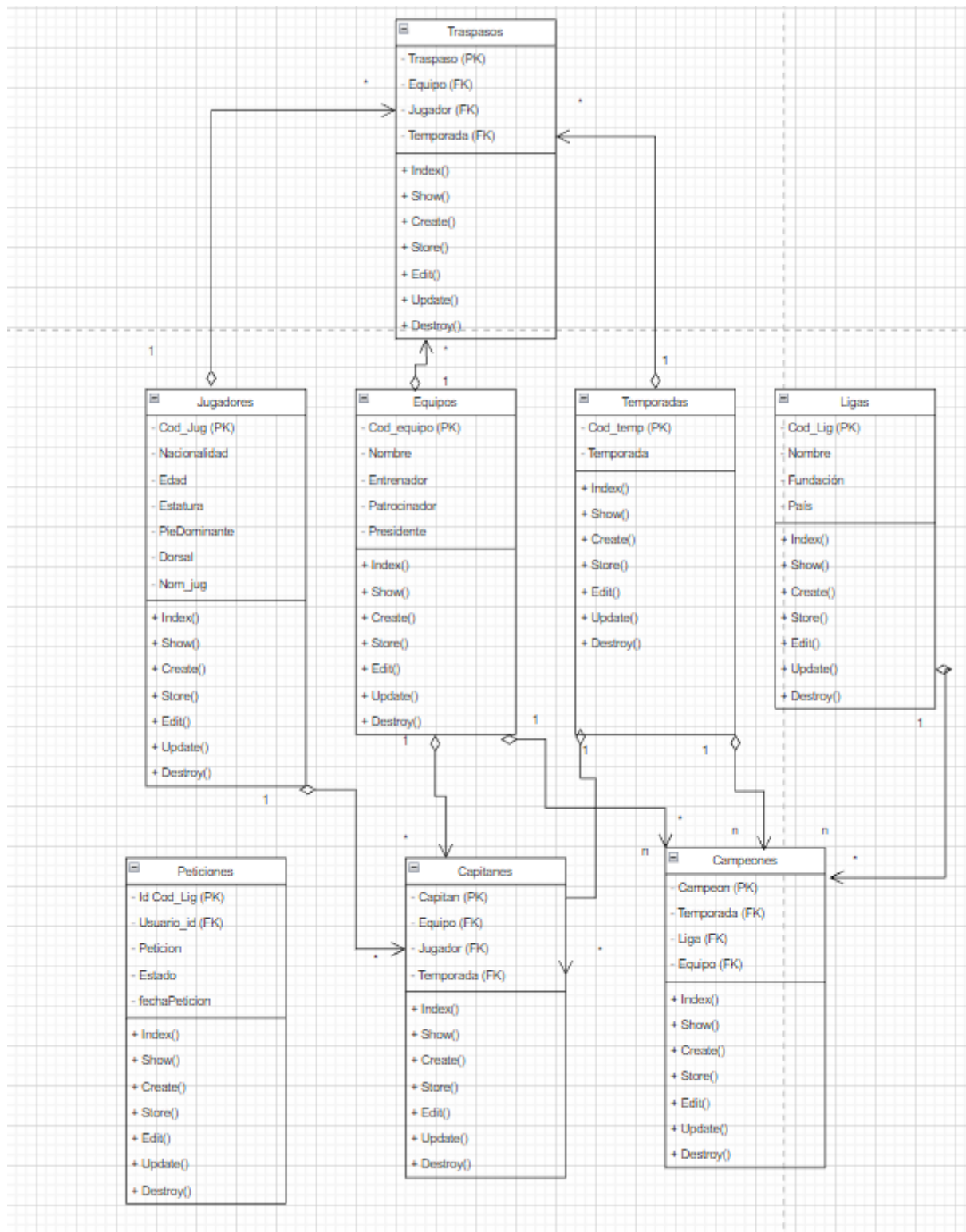
Usuario\_id Integer Not Null + FK (Relación uno a muchos con Id(Usuarios)) +

Peticion Varchar2(70) +

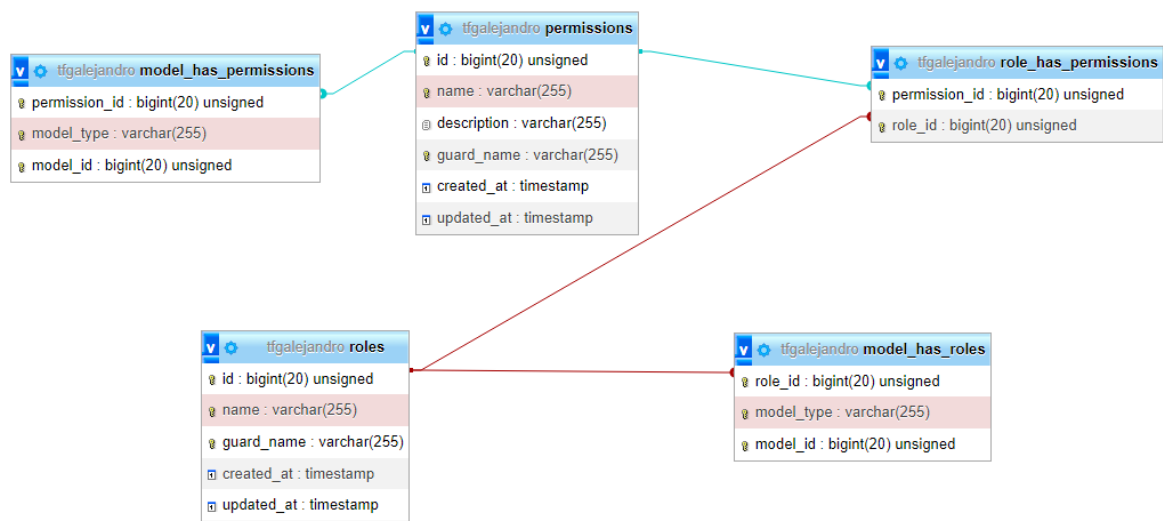
Estado enum(aceptado, pendiente)->default(pendiente)

fechaPeticion Timestamps

## 2.3 Diagrama de clases



2.4 Roles con Spatie



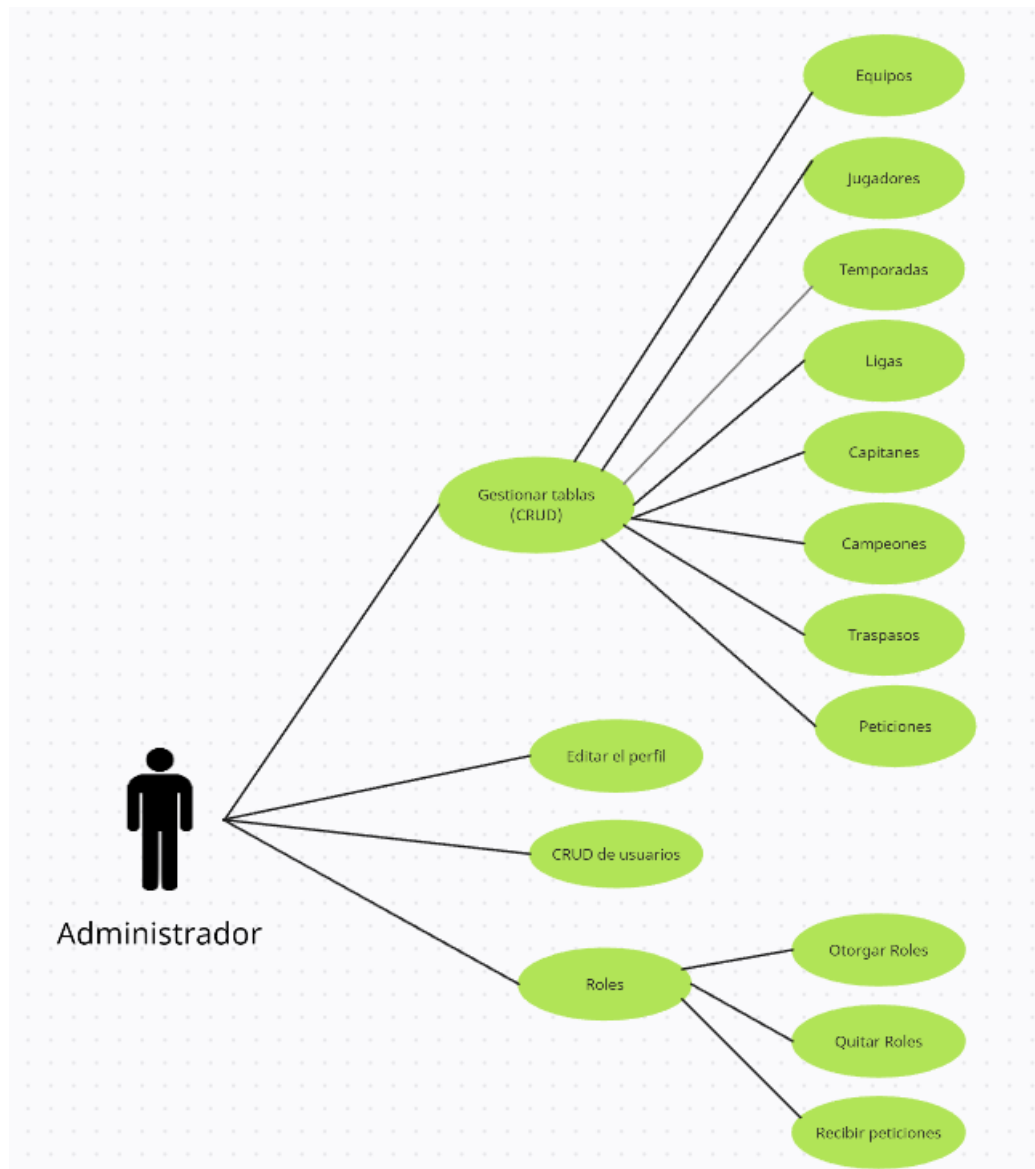
### 3. Análisis de requisitos

#### 3.1.Descripción de requisitos

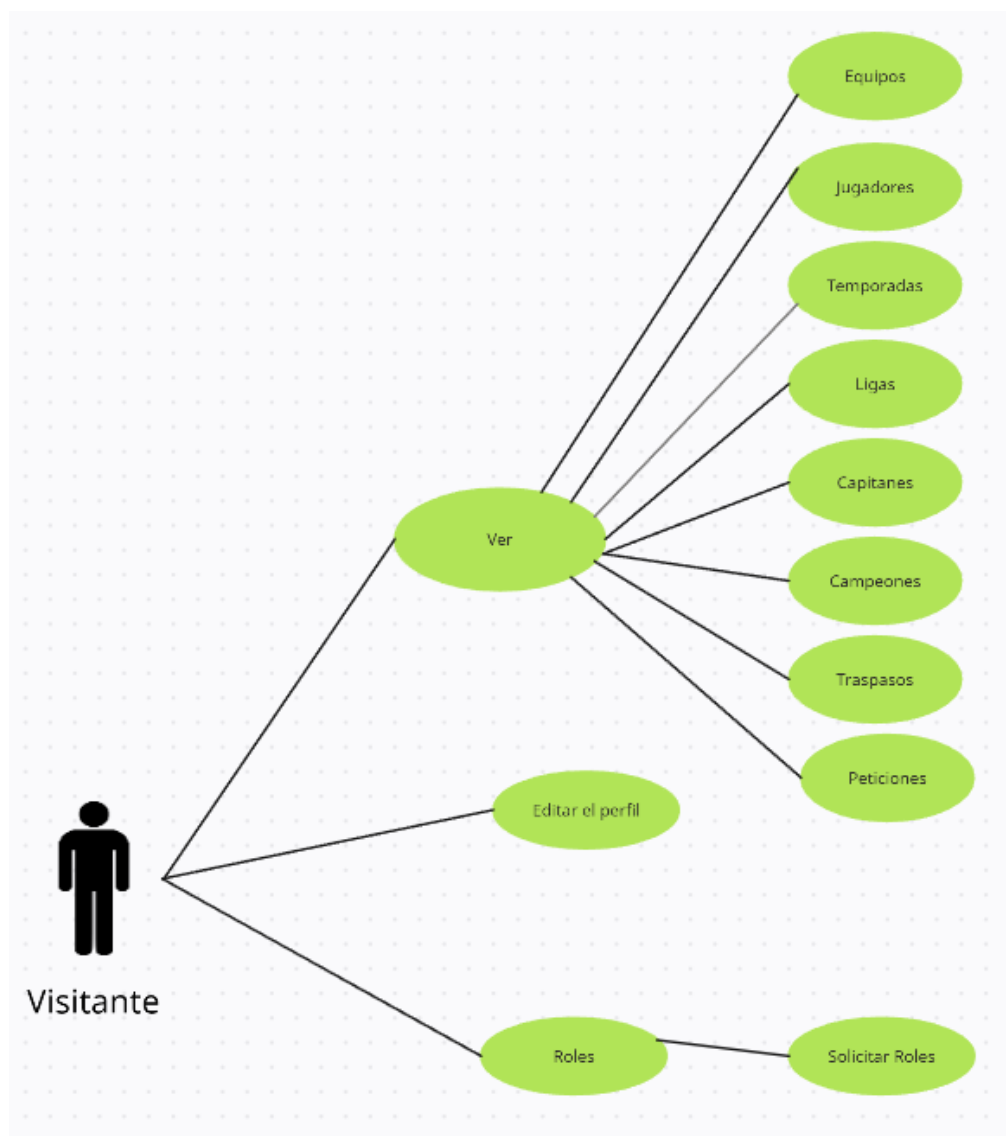
1. Creación de varios tipos de usuario. Cada uno con sus permisos para crear diferentes funciones. Los tipos de usuarios existentes, son: Sin Rol, Visitante y Administrador
2. Mediante un entorno amigable y visual el usuario administrador podrá gestionar las diferentes tablas que conforman la aplicación (CRUD)
3. El usuario sin roles, podrá pedir rol de Visitante para poder acceder a el entorno y ver la aplicación
4. El usuario accede a la vista principal si tiene el rol de Administrador o visitante
5. El usuario visitante tiene permisos unicamente de ver el entorno y las tablas
6. Gestionar las equipos (CRUD), como el jugador capitán y a que liga pertenece
7. Gestión CRUD de los equipos que conforman las ligas.
8. Gestión CRUD de las temporadas existentes.
9. Gestión CRUD de los capitanes de los equipos.
10. Gestión CRUD de los equipos campeones de las ligas.
11. Gestión CRUD de los jugadores de las ligas.
12. Gestión CRUD de los jugadores traspasados.
13. Gestión CRUD de usuarios.
14. Gestión CRUD de roles.
15. Las gestiones CRUD anterior solamente la podrá realizar el usuario administrador.  
Por ello se crea una tabla extra, llamada peticiones que su funcionalidad es igual a la de el correo donde un usuario Visitante puede pedir roles de Administrador
16. Un usuario puede realizar peticiones, estas tendrán 2 estados: Pendiente y aprobado. Si el estado de la petición es aprobado, podrá volver a pedir de lo contrario no podrá hacerlo
17. Si el usuario administrador aprueba la petición deberá crear un nuevo Rol, que será el nombre de el usuario visitante con los permisos otorgados
18. Todos los usuarios pueden administrar su perfil

## 3.2 Diagramas de casos de uso

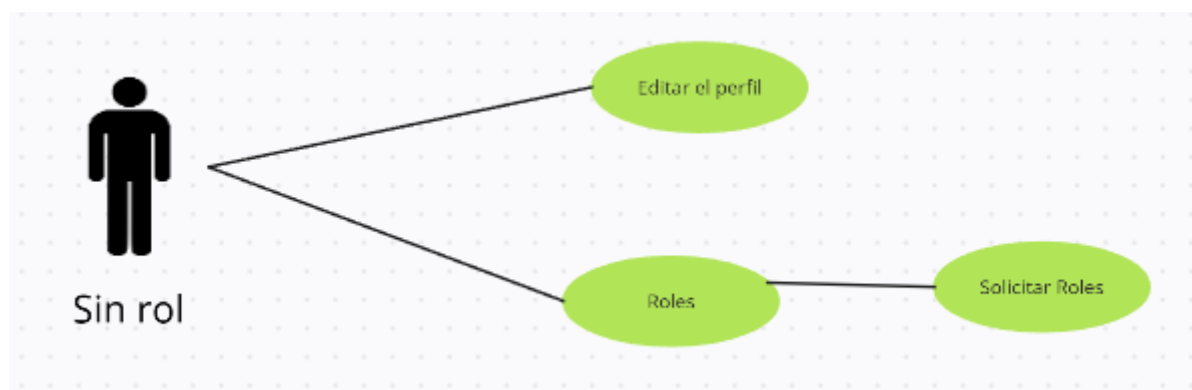
### 3.2.1 Administrador



### 3.2.2 Visitante



### 3.2.3 Sin Rol



### 3.3 Documentación de actores

Administrador	Gestiona las tablas de la aplicación – El CRUD de usuarios y puede otorgar y quitar roles según las peticiones que los usuarios le hagan
Visitante	Puede ver el entorno de trabajo así como tablas y sistema de petición de roles
Sin Rol	Unicamente podrá pedir roles

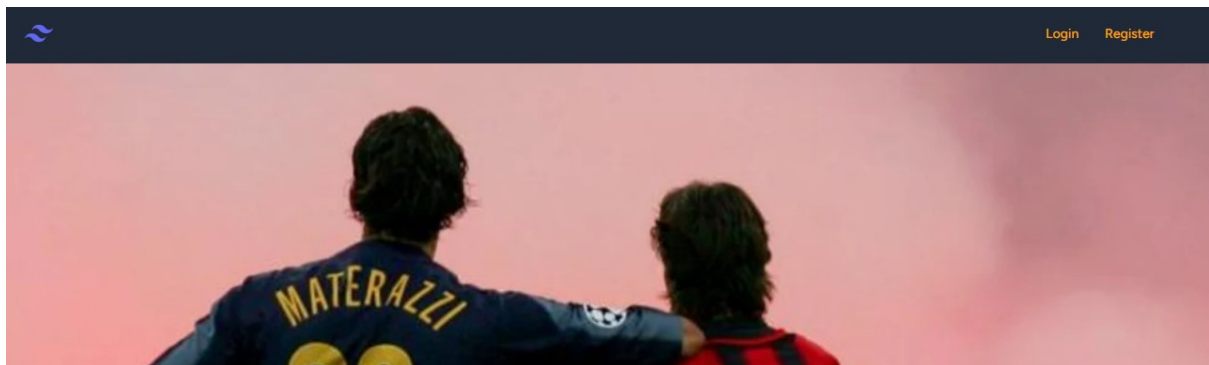


## 4. Aplicación

La funcionalidad de la aplicación es bastante sencilla. Es una aplicación interactiva y de rápido entendimiento

### 4.1 Página principal

Cuando la aplicación es lanzada aparecerá esta pestaña

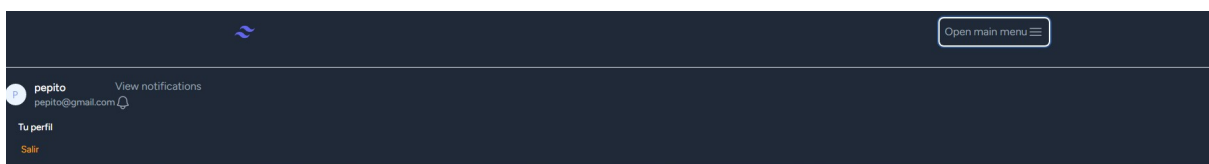


Administrada mediante breeze, estará esta ventana con 2 enlaces. Uno para registrarse y uno para iniciar sesión. La aplicación ofrece 3 opciones, usuario sin rol, usuario visitante o usuario administrador.

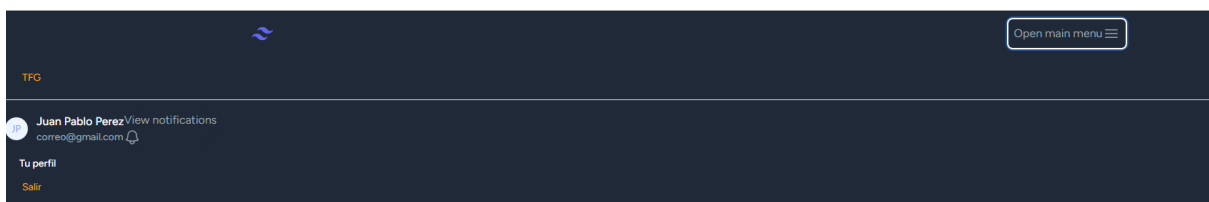
El usuario sin permisos dispondrá de una ruta para pedirle permisos a el administrador del sistema.

#### 4.1.1 Que ve cada usuario en la vista principal

##### 4.1.1.1 Usuario sin permisos



##### 4.1.1.2 Usuario con permisos



El usuario con permisos podrán ser de dos tipos:

- **Administrador:** Podrá ver, editar y borrar cada una de las tablas. También administrar roles y permisos de usuarios. Recibirá peticiones de usuarios con o sin permisos para otorgarle permisos y roles. Podrá acceder a la página de administración. Por último podrá editar su perfil
- **Visitante:** Podrá ver las tablas. Podrá acceder a la página de administración, solicitar roles y personalizar su perfil

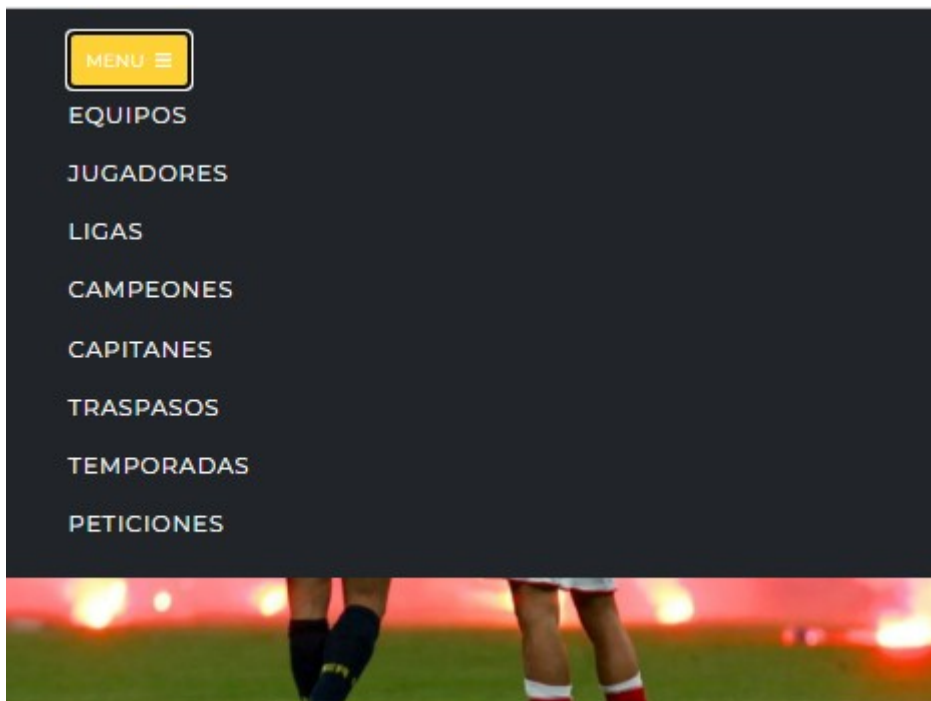
Los usuarios con permisos tendrán el enlace de la vista “TFG” que está compuesta por 3 partes

## 4.2 Vista “TFG”

### 4.2.1 Header



#### 4.2.1.1 Vista responsiva de la vista tfg



Es la primera parte de la vista, se compone de un menú de navegación por el que se podrá acceder a cualquiera de las tablas alojadas en “/admin”. Justo debajo habrá una imagen que cubrirá todo el div, junto con el título del proyecto. Cabe destacar que todo el proyecto será aplicable a cualquier pantalla

#### 4.2.2 Menú de información de la administración del proyecto



Es la segunda partes de la vista. Se basa en 2 columnas de 3 elementos que explican cada una de las funcionalidades de las tablas competentes

## 4.2.3 Tablas

## TABLAS

Se indica al usuario las tablas de la base de datos y además se especifica un enlace a cada una de ellas



**Equipos**  
Los principales equipos de las ligas europeas



**Jugadores**  
Una tabla dedicada a los capitanes de los equipos antes mencionados



**Ligas**  
Las ligas mas importantes y relevantes de europa



**Campeones**  
Se trata de una tabla dedicada a el equipo campeon de una liga en una temporada



**Capitanes**  
Los capitanes de los equipos



**Traspasos**  
Una tabla que muestra los clubes a los que han pertenecido los Jugadores



**Temporadas**  
Una tabla que muestra el histórico de todas las temporadas



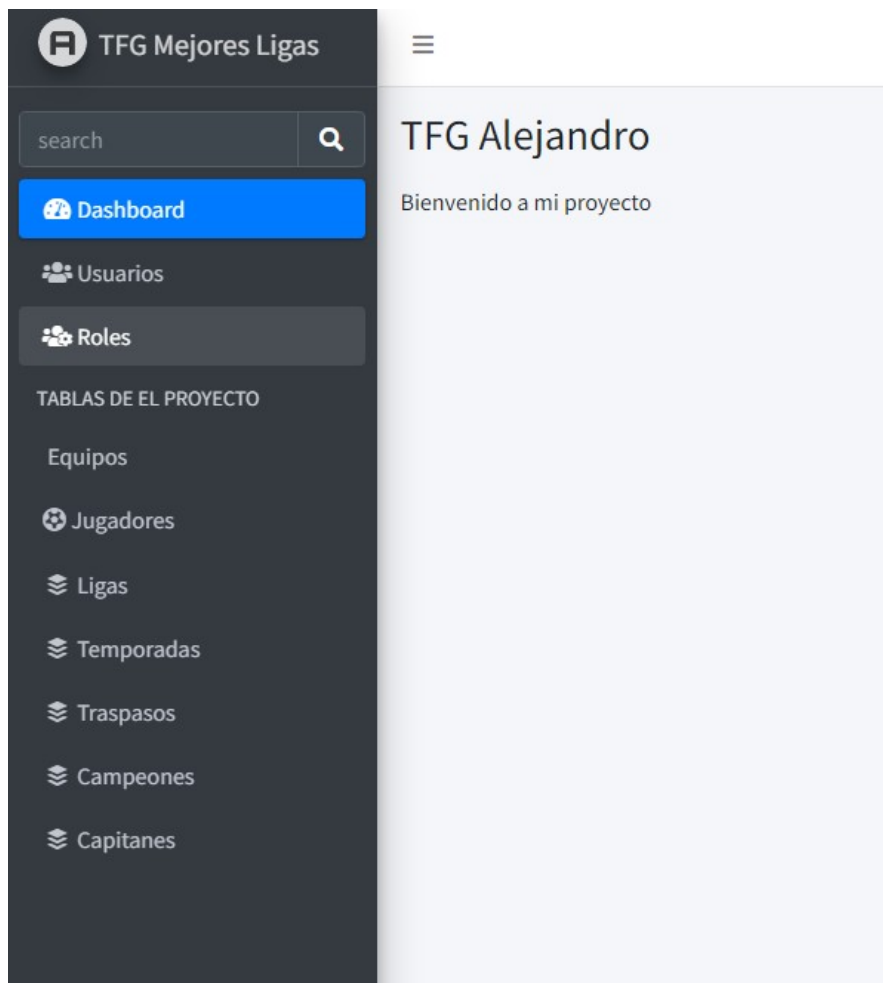
**Peticiones**  
Una tabla para que el usuario visitante pueda pedir roles



**Salir**  
Sale de la aplicación para logearse como otro usuario o por otro motivo

La última parte de la vista, se trata de una tabla de tablas, formadas por una imagen junto con una descripción de cada una de las, tienen la misma utilidad que los enlaces del header.

### 4.3 Página “/Admin”



Esta es la página principal dentro de la sección de los usuarios con permisos. Esta es una parte importante. Spatie permite a los usuarios acceder a “/admin” siempre que el usuario tenga permisos para acceder a esta página.

## 4.3.1 ¿Qué ve el usuario con rol Visitante?

















<div>TFG Mejores Ligas</div> <div>search</div> <div>Dashboard</div> <div>Roles</div> <div>TABLAS DE EL PROYECTO</div> <div><b>Equipos</b></div> <div>Jugadores</div> <div>Ligas</div> <div>Temporadas</div> <div>Traspasos</div> <div>Campeones</div> <div>Capitanes</div>	nombre	entrenador	patrocinador	presidente
	F.C Barcelona	Xavi Hernandez	Nike	Joan Laporta Estruch
	Real Madrid C.F	Carlo Ancelotti	Adidas	Florentino Perez
	Atletico de Madrid	Diego Pablo Simeone	Plus500	Enrique Cerezo
	Valencia C.F	Rubén Baraja Vegas	Puma	Peter Lim
	Sevilla F.C	Jorge Sampaoli	Nike	Jose Castro Carmona
	Juventus	Massimiliano Allegri	Adidas	Andrea Agnelli
	Napoli	Luciano Spalletti	Kappa	Aurelio de Laurentis
	A.S Roma	Jose Mourinho	Adidas	James Pallotta
	Atalanta B.C	Gian Piero Gasperini	Joma	Antonio Percassi
	F.C Inter de Milan	Simone Inzaghi	Adidas	Steven Zhang
	Liverpool F.C	Jurgen Klopp	New Balance	Tom Werner
	Manchester City	Pep Guardiola	Puma	Khaldoon Mubarak
	Manchester United	Erik ten Hag	Nike	Joe Glazer
	Tottenham F.C	Antonio Conte	Nike	Daniel Levy
	Chelsea F.C	Graham Stephen Potter	Nike	Todd Boehly

Este usuario solamente podrá ver el contenido de las tablas y crear un rol con limitaciones de administración. También podrá pedir un rol con permisos a un usuario administrador.

## 4.3.2 ¿Qué ve el usuario con rol Administrador?

Lista de Equipos

Agregar Equipo

cod_equipo	nombre	entrenador	patrocinador	presidente	Editar	Borrar
1	F.C Barcelona	Xavi Hernandez	Nike	Joan Laporta Estruch		
2	Real Madrid C.F	Carlo Ancelotti	Adidas	Florentino Perez		
3	Atletico de Madrid	Diego Pablo Simeone	Plus500	Enrique Cerezo		
4	Valencia C.F	Rubén Baraja Vegas	Puma	Peter Lim		
5	Sevilla F.C	Jorge Sampaoli	Nike	Jose Castro Carmona		
6	Juventus	Massimiliano Allegri	Adidas	Andrea Agnelli		
7	Napoli	Luciano Spalletti	Kappa	Aurelio de Laurentis		
8	A.S Roma	Jose Mourinho	Adidas	James Pallotta		

El usuario administrador podrá ver, crear, editar y borrar cada una de las tablas. Otorgar permisos a un usuario, crear roles con permisos y recibir peticiones de los usuarios

## 4.4 Roles

Lista de Roles			Crear rol
ID	Role		
1	Administrador	<a href="#">Editar</a>	<a href="#">Eliminar</a>
2	Visitante	<a href="#">Editar</a>	<a href="#">Eliminar</a>

### Asignar un Rol

Se asigno los roles correctamente.

Nombre

alex

Listado de Roles

☐ Administrador

☒ Visitante

Asignar Rol

### Asignar un Rol

Nombre

Alejandro Lapeña Ruiz

Listado de Roles

☒ Administrador

☐ Visitante

Asignar Rol

Este es el CRUD de todos los permisos existentes que el administrador podrá añadir y quitar a los usuarios que le pidan permisos



Nombre

Lista de Permisos

- ☐ Ver el dashboard
- ☐ Ver listado de usuarios
- ☐ Asignar un Rol
- ☐ Asignar un Rol
- ☐ Ver listado de equipos
- ☐ Crear equipos
- ☐ Editar equipos
- ☐ Eliminar equipos
- ☐ Ver listado de jugadores
- ☐ Crear jugadores
- ☐ Editar jugadores
- ☐ Eliminar jugadores
- ☐ Ver listado de ligas
- ☐ Crear ligas
- ☐ Editar ligas
- ☐ Eliminar ligas
- ☐ Ver listado de temporadas
- ☐ Crear temporadas
- ☐ Editar temporadas
- ☐ Eliminar temporadas
- ☐ Ver listado de traspasos
- ☐ Crear traspasos
- ☐ Editar traspasos
- ☐ Eliminar traspasos
- ☐ Ver listado de campeones
- ☐ Crear campeones
- ☐ Editar campeones
- ☐ Eliminar campeones
- ☐ Ver listado de capitanes
- ☐ Crear capitanes
- ☐ Editar capitanes
- ☐ Eliminar capitanes

[Crear Rol](#)

## 4.5 CRUD

### 4.5.1 Create

#### Crear Jugador

**nacionalidad**

Ingrese la nacionalidad de el jugador

**edad**

Ingrese la edad de el jugador

**estatura**

Ingrese la estatura de el jugador

**pieDominante**

Ingrese el pieDominante de el jugador

**dorsal**

Ingrese el dorsal de el jugador

**nom\_jug**













Ingrese el nombre de el jugador

Crear jugador

## 4.5.2 Read

Lista de Jugadores

[Agregar Jugador](#)

cod_jug	nacionalidad	edad	estatura	pieDominante	dorsal	nom_jug	Editar	Borrar
1	ARG	32	170	Zurdo	10	Lionel Messi		
2	ESP	34	184	Diestro	4	Sergio Ramos		
3	ESP	28	176	Diestro	6	Jorge Resurrecion		
4	ESP	30	182	Diestro	10	Daniel Parejo		
5	ESP	34	170	Diestro	16	Jesus Navas		
6	ITA	35	187	Zurdo	3	Giorgio Chiellini		

## 4.5.3 Update

Editar

nacionalidad	<input type="text" value="ARG"/>
edad	<input type="text" value="32"/>
estatura	<input type="text" value="170"/>
pieDominante	<input type="text" value="Zurdo"/>
dorsal	<input type="text" value="10"/>
nom_jug	<input type="text" value="Lionel Messi"/>
<input type="button" value="Crear jugador"/>	

## 4.5.4 Delete

24	BRA	34	192	Diestro	4	Jardel		
----	-----	----	-----	---------	---	--------	---	---

4.6 Usuarios

Lista de Usuarios

Ingrese el nombre o correo de usuario

ID	Nombre	Email	
1	Alejandro Lapeña Ruiz	alejandrolapena90@gmail.com	Editar
2	Prof. Tess Gottlieb	dooley.bethel@example.com	Editar
3	Miss Tara Hermann	manuela.langosh@example.com	Editar
4	Magnolia Ortiz	hans.bartell@example.com	Editar
5	Ms. Mireya Pagac	alana.kerluke@example.net	Editar
6	Roosevelt Rohan V	harber.gretchen@example.net	Editar
7	Shirley Stark	iokeefe@example.org	Editar
8	Dr. Virginie Blick PhD	bzemlak@example.net	Editar
9	Providenci Kassulke	fframi@example.net	Editar
10	Ambrose Cummings	lebert@example.net	Editar
11	Adalberto Rowe	marcus.kesler@example.org	Editar
12	Jerel Jenkins	freida86@example.com	Editar
13	Jordi Hagenes	mswift@example.net	Editar
14	Alicia Hermiston	volkman.shea@example.com	Editar
15	Andre Terry	bschuppe@example.org	Editar

1

2

3

4

5

6

7

4.7 Tabla de Peticiones

Se trata de una tabla en el que el usuario podrá solicitar permisos y así el administrador le otorgará los roles. Con esos roles, el usuario visitante podrá administrar tablas como si fuera un usuario administrador más

Lista de Peticiones

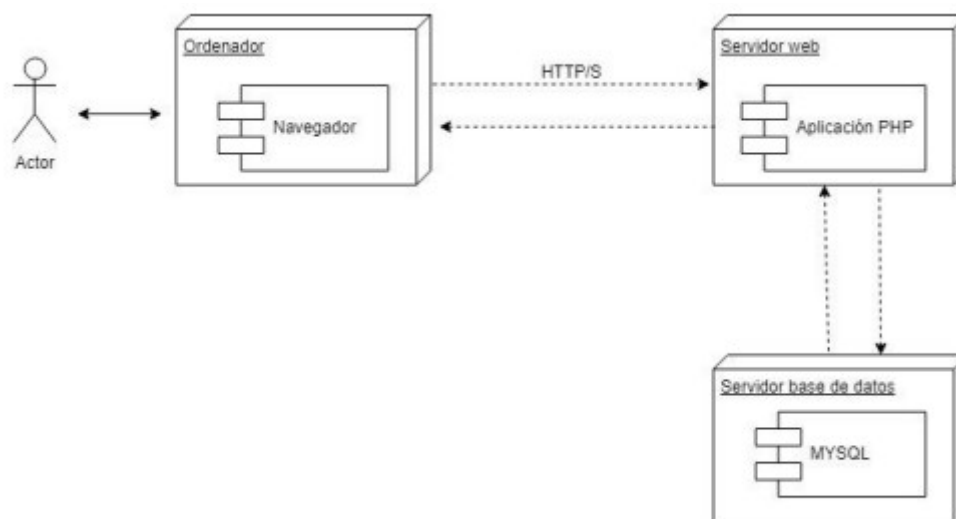
Crear Peticion

ID	Usuario	Peticion	Asunto	Estado	Fecha de Petición	Editar	Borrar
1	Alejandro Lapeña Ruiz	Me gustaría administrar la tabla de equipos y jugadores	Administracion de tablas	pendiente	2023-05-23 18:31:46		
2	Juan Pablo Perez	Me gustaría administrar la tabla de ligas y temporadas	Administracion de tablas	pendiente	2023-05-23 18:31:46		

Se trata de un CRUD normal y corriente, pero está hecho de manera lo más parecida a un buzón de correo. Cuando se envían están en estado pendiente, si el usuario las edita/lee pasan a estado aceptado y el usuario administrador le otorgará los permisos como se puede ver en la imagen inferior. Mientras la solicitud no este siendo aceptada, el usuario no podrá volver a solicitar permisos

## 5. Implantación

### 5.1 Diagrama de despliegue



### 5.2 Proceso de despliegue

El despliegue se ha realizado en un servidor virtual privado de Azure con un sistema operativo Ubuntu server 20.04. A continuación se explica el proceso.

#### 5.2.1 Primeros pasos

Se crea un usuario con privilegios de administrador para poder acceder al servidor y subir archivos. Se le quita permiso a el usuario root de acceder.

Una vez creada la máquina virtual siguiendo este tutorial [https://www.youtube.com/watch?v=83QyqBwqLE4&t=1s&ab\\_channel=SergiGarcia](https://www.youtube.com/watch?v=83QyqBwqLE4&t=1s&ab_channel=SergiGarcia) , accedemos al servidor de esta forma **ssh usuario@ipdeldominio -port N.º de puerto habilitado -host ip pública de la máquina**

#### 5.2.2 Instalación de servicios y programas para el despliegue

Se instala putty para la ejecución de comandos y control de la máquina virtual, además de WinSCP para la transferencia de archivos.

Una vez instalado putty, este pedirá el nombre de usuario y la contraseña, después instalaremos y configuraremos: mariadb, php, net tools y composer.

Lo siguiente sería instalar los paquetes adicionales de laravel

### **5.3 Dominio**

Por último se ha creado el siguiente subdominio

[https://portal.azure.com/#@bilko69alargmail.onmicrosoft.com/resource/subscriptions/3d1bceba-47c7-4911-9435-0d9079eaa3b4/resourceGroups/TFGALEJANDRO\\_group/providers/Microsoft.Compute/virtualMachines/TFGALEJANDRO/overview](https://portal.azure.com/#@bilko69alargmail.onmicrosoft.com/resource/subscriptions/3d1bceba-47c7-4911-9435-0d9079eaa3b4/resourceGroups/TFGALEJANDRO_group/providers/Microsoft.Compute/virtualMachines/TFGALEJANDRO/overview)

40.68.79.142

## 6. Conclusiones

Como se ha podido comprobar, El proyecto consiste en la investigación y desarrollo del sistema de gestión de base de datos MySQL, creación de una base de datos de ejemplo implementado sobre dicho sistema gestor de base de datos y un desarrollo en Laravel que permite administrar y dotar de contenidos a la base de datos.

- 1 El diseño de la base de datos realizado ha resultado un poco complejo, ya que hace 4 años que no hacía nada de base de datos. He elegido MySQL, que es el gestor más utilizado, simplemente por la compatibilidad que tiene con la herramienta Laravel con "PhpMyAdmin"
- 2 El lenguaje de programación utilizado en el proyecto que es Laravel, su aprendizaje fue un poco tedioso ya que se componía de una lista de YouTube de 79 videos algunos de duración corta y otros de más de media hora. Una vez acabe de ver los videos vi que era bastante cómodo y me gustó más que lo que era PHP vanilla
- 3 La aplicación "Ligas": digamos que es la parte capital del proyecto que como he dicho en la introducción está formado por varios tipos de software. Pero los más destacados son Laravel, Xampp, Bootstrap y cómo no MySQL. He conseguido que tenga un diseño original apto para cualquier dispositivo. La página principal tiene mucho código, que contiene varias imágenes, fotos y menús que te llevarán a la primera sección de cada tabla. El uso responsivo de Bootstrap ha sido el software más sencillo de implementar de la aplicación porque su aprendizaje fue. prácticamente guiado en el módulo de Lenguaje de marcas del primer curso académico.
- 4 El proceso que he llevado desde que era una idea de papel y boli hasta lo que se ha llegado a convertir: Empecé muy pronto, antes que cualquier alumno y a dedicarle tiempo a pensar que podía hacer para que el proyecto con el que acabó el ciclo sea un gran proyecto. Después de escribir mucho código y documentarme en mil páginas conseguí hacer la base de datos y conectarla con Laravel, me empecé a tropezar con mil fallos como errores de desarrollo en las relaciones de las tablas, prueba y error para saber si faltaba alguna extensión de Laravel o cuando hacía el "seed", la aplicación no



explotara y tener que borrar algunas tablas a mano para poder hacer el rollback... Pero al final la conseguí hacer y la estoy presentando en frente de todos ustedes.

- 5 El final: Hemos llegado al final del grado en el que hemos aprendido mucho, algunos han notado el salto del grado medio, otros del bachiller que empezaban de cero o gente que venía de otro superior como es mi caso. Estoy orgulloso de mí mismo por haberme sacado este grado teniendo 0 de base de programación y haber podido poco a poco y con mucho esfuerzo obtener el nivel para lograrlo.

## 7. Bibliografía

1. <https://www.youtube.com/watch?v=3e1lsZJuYAw&list=PLZ2ovOgdl-kWWS9aq8mfUDkJRfYib-SvF>

Se tratan de la lista de reproducción de los videos de Laravel

2. <https://www.MySQL.com/>

Es el sistema gestor de base de datos con el que se hizo el proyecto

3. <https://getbootstrap.com/docs/5.0/getting-started/introduction/>

Es el software para conseguir el diseño responsivo

4. <https://jquery.com/>

Lenguaje para conseguir la interactividad con el usuario

5. <https://laravel.com/>

Framework de PHP con el que esta hecha la aplicación

6. <https://laravel.com/docs/10.x>

Documentación para ayuda en Laravel

7. [https://aules.edu.gva.es/fp/pluginfile.php/5004109/mod\\_resource/content/2/Documento%20informativo%20Proyecto%20Fin%20de%20Ciclo.pdf](https://aules.edu.gva.es/fp/pluginfile.php/5004109/mod_resource/content/2/Documento%20informativo%20Proyecto%20Fin%20de%20Ciclo.pdf)

Pdf con las normas a seguir con la presentación que están leyendo

8. [https://www.youtube.com/watch?v=83QyqBwqLE4&t=1s&ab\\_channel=SergiGarcia](https://www.youtube.com/watch?v=83QyqBwqLE4&t=1s&ab_channel=SergiGarcia)

Guía para crear la máquina de azure

9. [https://www.youtube.com/watch?v=J\\_n8m-tliAs&ab\\_channel=SergiGarcia](https://www.youtube.com/watch?v=J_n8m-tliAs&ab_channel=SergiGarcia)

Guía para el despliegue en Laravel

10. <https://azure.microsoft.com/es-es>

Dominio utilizado en el despliegue

11. <https://www.putty.org/>

Consola de comandos donde nos conectamos con nuestro dominio en Azure

12. <https://winscp.net/eng/download.php>

Programa para subir los archivos a la máquina de azure

13. <https://code.visualstudio.com/>

Editor de texto usado en el proyecto

14. <https://www.apachefriends.org/es/index.html>

Página oficial de Xampp para el despliegue en local

15. <https://app.diagrams.net/>

Página para crear los diseños de el proyecto

## 8. Anexos

### 8.1 Anexo A

Código JQuery en las vistas index.blade.php de la aplicación:

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  $(document).ready(function() {
    // Agregar confirmación antes de enviar el formulario de borrado
    $('form').submit(function() {
      var form = this;
      var confirmMessage =
        '¿Está seguro de que desea eliminar esta temporada? Esta acción no se puede
deshacer.';
      if (confirm(confirmMessage)) {
        // Enviar el formulario si se confirma la eliminación
        return true;
      } else {
        // Cancelar el envío del formulario si se cancela la eliminación
        return false;
      }
    });
  });
</script>
```

Este código se realizó por parte de el autor para agregarle interactividad a la vista y cuando se pulsa el icono de la papelera mostrando el mensaje de que si esta seguro de borrar la fila

### 8.2 Anexo B

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  $(document).ready(function() {
    $('#create-form').submit(function(e) {
      e.preventDefault(); // Evitar que el formulario se envíe de forma predeterminada

      // Obtener los datos del formulario
      var formData = $(this).serialize();

      // Realizar una solicitud AJAX para crear el equipo
      $.ajax({
        url: $(this).attr('action'),
        type: $(this).attr('method'),
        data: formData,
```

```
success: function(response) {
    // Mostrar mensaje de éxito
    alert('Temporada creada correctamente');

    // Redireccionar a la página de lista de equipos
    window.location.href = '{{ route('admin.temporadas.index') }}';
},
error: function(xhr) {
    // Mostrar errores de validación en caso de que ocurran
    if (xhr.status === 422) {
        var errors = xhr.responseJSON.errors;
        var errorMessages = "";
        for (var key in errors) {
            errorMessages += errors[key][0] + '\n';
        }
        alert(errorMessages);
    } else {
        // Mostrar mensaje de error genérico
        alert('Se produjo un error al crear la temporada');
    }
}
});
});
});
</script>
```

El autor creó este código con el fin de agregarle interactividad y cumplimentación junto con el controlador, para ofrecerle un “feedback” al usuario y que sepa si el registro ha sido creado correctamente

### 8.3 Anexo C

```
public function create()
{
    $usuarios = User::all();
    return view('admin.peticiones.create', compact('usuarios'));
}

public function store(Request $request)
{
    $user = auth()->user();

    $fechaPeticiones = Carbon::now();
```

```
// Verificar si el usuario ya tiene una petición pendiente
$existingPeticones = Peticones::where('usuario_id', $user->id)
->where('estado', 'pendiente')
->first();

if ($existingPeticones) {
    // Si existe una petición pendiente, mostrar un mensaje de error
    return redirect()->back()->with('error', 'Ya tienes una petición pendiente');
}
```