

# Proyecto Zona de Arbitraje

PÁGINA WEB DESARROLLADA USANDO MYSQL,  
JAVA CON SPRING BOOT Y ANGULAR

JORGE GONZÁLEZ SÁENZ

Back: <https://github.com/jorgegs23/intranet-back>

Front: <https://github.com/jorgegs23/intranet-front>

25/05/2023

Grado Superior de Desarrollo de Aplicaciones Web  
IES Comercio  
Curso 2022/23

## ÍNDICE

1. Agradecimientos .....	2
2. Introducción .....	3
3. Objetivos .....	4
4. Fases del proyecto.....	5
4.1. Diseño .....	6
4.1.1. Diseño de base de datos .....	6
4.1.2. Diseño de la página web .....	7
4.2. Implementación y configuración .....	9
4.2.1. Configuración de Back.....	9
4.2.2. Configuración de Front.....	10
4.2.3. Investigación librería PrimeNG .....	11
4.3. Desarrollo.....	15
4.3.1. Login de usuarios.....	15
4.3.2. Listado y detalle de usuarios .....	17
4.3.3. Listado de temporadas.....	20
4.3.4. Listado y detalle de equipos.....	21
4.3.5. Listado y detalle de partidos .....	23
4.3.6. Listado y detalle de designaciones.....	25
4.3.7. Acceso con perfiles.....	28
4.3.8. Generar informe JasperReports .....	29
4.4. Documentación y pruebas generales .....	30
5. Ampliación y posibles mejoras.....	31
6. Conclusión .....	32
7. Bibliografía .....	33
ANEXO - Puesta en marcha del proyecto .....	34
1. Configuración del Back .....	34
2. Configuración del Front.....	35
3. Configuración adicional .....	35

## 1. Agradecimientos

Hay mucha gente a la que podría agradecer incluyendo familia, amigos y compañeros de trabajo, y me gustaría destacar la figura algunos de estos miembros de mi vida.

En primer lugar a mi pareja Lucía por apoyarme en casa y ayudarme a poder sacar más tiempo en estos últimos dos meses para poder trabajar y desarrollar este proyecto pudiendo dejar de lado parte de mis obligaciones en el hogar.

En segundo lugar a mis compañeros de Hiberus, Marcelo Carro y Belén Pérez-Aradros, por enseñarme y apoyarme desde mí inicio con ellos en la empresa y seguir juntos aunque cada uno esté en un proyecto distinto. También agradecer a Alberto Escalona por sus consejos de Angular para algunas funcionalidades de la aplicación.

Hay mucha más gente a la que podría agradecer pero en lo que se refiere a la realización de este proyecto son de los primero me vienen a la cabeza.

## 2. Introducción

Mi nombre Jorge González Sáenz y además de ser desarrollador web en *Hiberus* desde hace año y medio, soy oficial de mesa de baloncesto desde hace 11 años. En mi experiencia en el mundo del arbitraje he vivido el avance de las tecnologías aplicado a mi labor como oficial.

En un principio las designaciones de los encuentros las recibíamos en papel, y los recibos los teníamos que generar a mano usando una plantilla impresa y fotocopiada. Unos años más tarde, con la popularización de tener un correo electrónico personal, se nos enviaba un documento pdf con todos los encuentros donde se incluían todas las designaciones de todos los miembros de la federación.

En 2018, llego un portal web donde se podían descargar los encuentros, y una plantilla a rellenar en formato digital para rellenar y entregarla firmada digitalmente con certificado electrónica, la cual sustituía por fin al recibo manual.

Visto el portal web de la federación y con mis conocimientos adquiridos en mi vida laboral me gustaría plantear una aplicación que sirva para gestionar usuarios, equipos, partidos y designaciones para distintas temporadas.

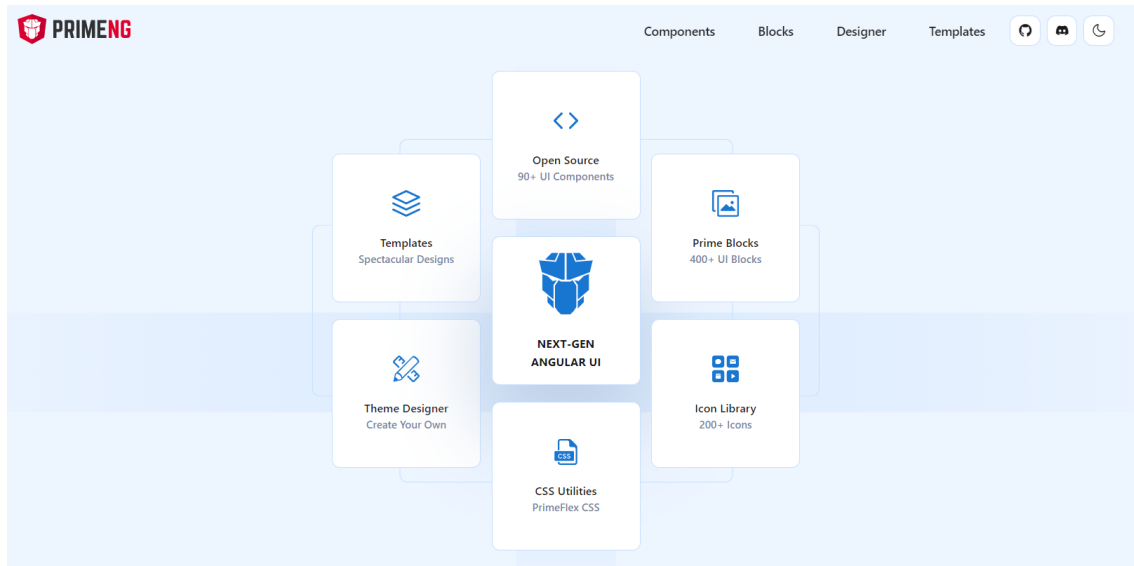
Esta aplicación está basada en el deporte baloncesto, pero técnicamente podría ser usada para la gestión arbitral de distintos deportes.

### 3. Objetivos

El objetivo de la aplicación es poder gestionar encuentros deportivos entre distintos equipos, así como las designaciones de estos encuentros.

La aplicación tendrá una funcionalidad distinta dependiendo del perfil del usuario que acceda a la misma. Si accede un usuario administrador podrá gestionar usuarios, temporadas, equipos, partidos y designaciones, mientras que si entran otros perfiles como árbitro u oficial, solo podrán editar sus datos y consultar sus designaciones, así como obtener un informe de las mismas.

Otro aspecto es el la investigación y aprendizaje de la librería **PrimeNG**, una librería de componentes con estilos para el desarrollo de la parte frontal de páginas web utilizando **Angular** como framework. En mi experiencia he usado previamente algún componente y me parece atractiva la idea de aprender más sobre ella.



Por último añadir que el objetivo principal no es tener una página completa y que tenga cada aspecto comprobado al milímetro, es tener una aplicación funcional, con una buena estructura, la cual pueda ser mejorada en los aspectos que se encuentren pertinentes en un futuro.

## 4. Fases del proyecto

El proyecto ha sido realizando usando un modelo hibrido, con una planificación en cascada y uso de metodologías ágiles a la hora de crear las distintas pantallas de la aplicación, donde cada una será estudiada, desarrollada y probada.

#	Tarea	Inicio	Fin	Abril				Mayo			
				Semana 1 3 abr - 9 abr	Semana 2 10 abr - 16 abr	Semana 3 17 abr - 23 abr	Semana 4 24 abr - 30 abr	Semana 5 1 may - 7 may	Semana 6 8 may - 14 may	Semana 7 15 may - 21 may	Semana 8 22 may - 28 may
1	Definición de la aplicación	03-abr	06-abr								
2	Diseño de la base de datos	07-abr	08-abr								
3	Diseño bajo nivel de las pantallas	08-abr	12-abr								
4	Configuración entorno back	12-abr	13-abr								
5	Configuración entorno front	14-abr	15-abr								
6	Conexión back y front	16-abr	17-abr								
7	Investigación componentes PrimeNG	17-abr	04-may								
8	Componentes comunes frontal	19-abr	22-abr								
9	Login de usuarios	23-abr	28-abr								
10	Listado y detalle de usuarios	29-abr	03-may								
11	Listado de temporadas	04-may	05-may								
12	Listado y detalle de equipos	06-may	08-may								
13	Listado y detalle de partidos	09-may	10-may								
14	Listado y detalle de designaciones	11-may	13-may								
15	Acceso con perfiles	11-may	15-may								
16	Generar informe Jasper	13-may	14-may								
17	Documentación y pruebas generales	15-may	20-may								

En esta planificación se diferencian las tareas en cuatro categorías, como se puede ver en el diagrama por la distinción de colores:

1. Diseño.
2. Configuración e implementación.
3. Desarrollo.
4. Pruebas y documentación.

Además de conocer la planificación y gestión de la misma debemos conocer un aspecto de gran importancia, las tecnologías en las que desarrollaremos la aplicación., la cual consta de tres apartados indispensables para el funcionamiento completo de la misma:

1. Base de datos en **MySQL**
2. Back desarrollado en **Java** con **Spring Boot**, desde el entorno de desarrollo Eclipse.
3. Front desarrollado en **Angular**, utilizando las librerías de **PrimeNG**, desde el entorno de desarrollo Visual Studio Code.

He elegido **MySQL** como gestor de base de datos ya que la he usado previamente para proyectos similares y ha sido fácil de usar y adaptar.

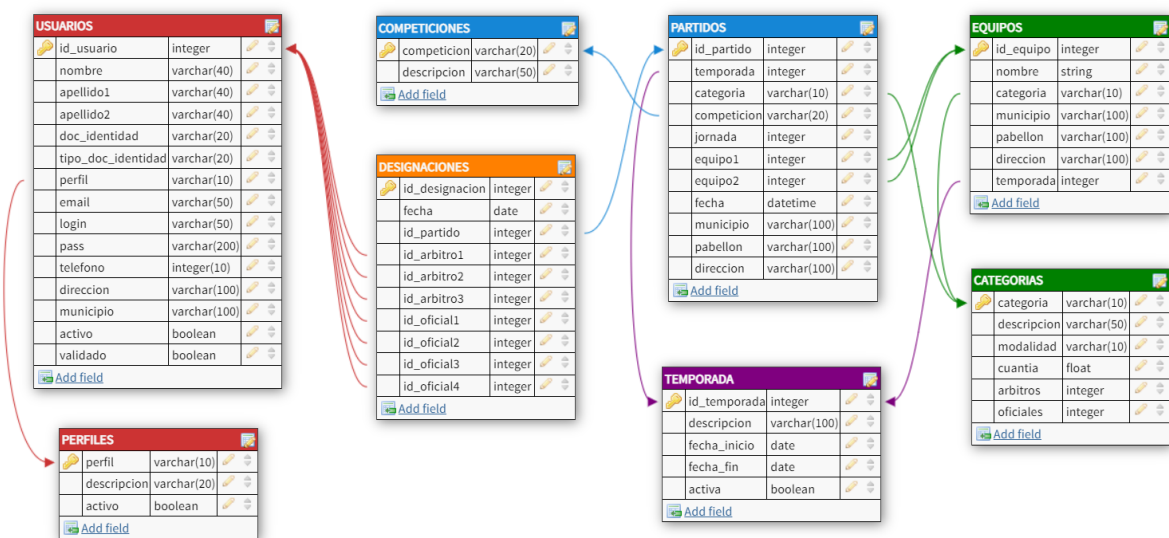
La parte back con **Java** y front con **Angular** han sido elegidos porque llevo desarrollando aplicaciones con esta misma arquitectura desde hace más de un año me siento muy cómodo, aunque el añadido de la librería **PrimeNG** es para aumentar mi conocimiento sobre esta librería que me resulta tan interesante.

## 4.1. Diseño

En esta primera fase comenzaremos por definir la aplicación, su funcionalidad y los requisitos iniciales de la misma. Conocemos las tecnologías con las que vamos a trabajar, y para empezar crearemos un diseño de la base de datos y un esquema funcional.

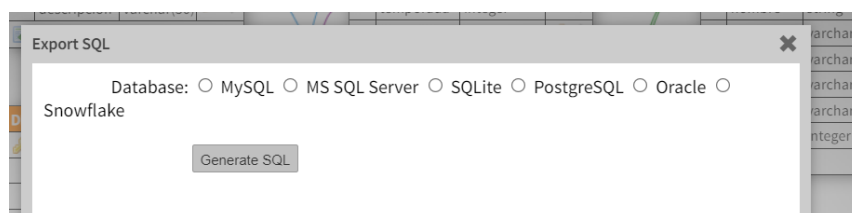
### 4.1.1. Diseño de base de datos

Para iniciar el desarrollo de la aplicación, primero he creado la base de datos utilizando la web [dbdesigner](#), la cual te permite diseñar el modelo de datos de la aplicación de una manera muy sencilla gracias a su interfaz gráfica.

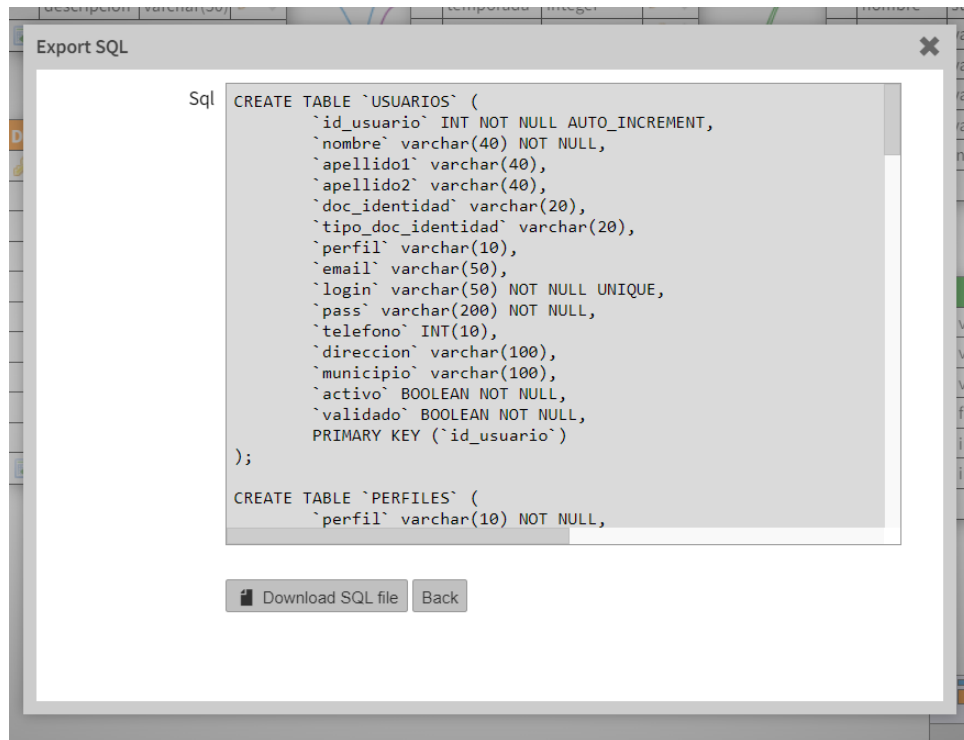


Se cómo crear una base de datos mediante comandos, pero me parece buena idea recurrir a una aplicación que te ayuda a presentar los datos de esta manera, ya que una simple imagen presenta mucha información: tipo de datos, claves primarias y relaciones entre tablas por ejemplo.

En caso de registrarnos gratuitamente en la página, podremos guardar nuestros diseños. Otra funcionalidad interesante que nos ofrece esta página es la exportación de los datos a la base de datos que queramos dentro de las siguientes opciones.



En mi caso elegiré **MySQL** y pulsaré el botón de generar, automáticamente creará un fichero sql para crear las tablas y las relaciones entre ellas en mi base de datos, con lo que crearemos la base de datos de una manera muy sencilla.



#### 4.1.2. Diseño de la página web

Para comenzar la página web, empezaré realizando un diseño a bajo nivel de la misma, para ello hay muchas opciones como por ejemplo hacerlo a mano, pero en mi caso realizare un diseño de las pantallas esquemático mediante la aplicación de código abierto [Pencil](#).





Mediante esta aplicación realizaremos un primer boceto de nuestra aplicación, por ejemplo a continuación se muestran la pantalla de acceso.

**Acceso a la Intranet**

login

password

Acceder

[No tengo cuenta, registrarse](#)

Parece un simple esquema de nuestras pantallas finales pero cuanto más contenido tienen estas, más información nos aportan como en el caso del listado usuario y detalle de usuarios, donde podemos apreciar un menú, unos campos que representan un filtros, una tabla con información, acciones...

**Intranet**

Usuarios Temporadas Equipos Partidos Designaciones Area Personal

nombre y apellidos perfil validado activo

Crear Usuario Editar Eliminar

<input type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado
<input checked="" type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado
<input type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado

**Intranet**

Usuarios Temporadas Equipos Partidos Designaciones Area Personal

**Detalle de Usuario**

Guardar Volver

login password

nombre apellido1 apellido2

tipoDoc docIdentidad email perfil

telefono municipio direccion

Validado Activo

☐ Si ☐ No ☐ Si ☐ No

Estos diseños iniciales serán fundamentales a la hora de crear la parte frontal de la aplicación, ya que aportan un diseño a cada pantalla funcional que puede ser cambiado para mejorar el funcionamiento de esta.

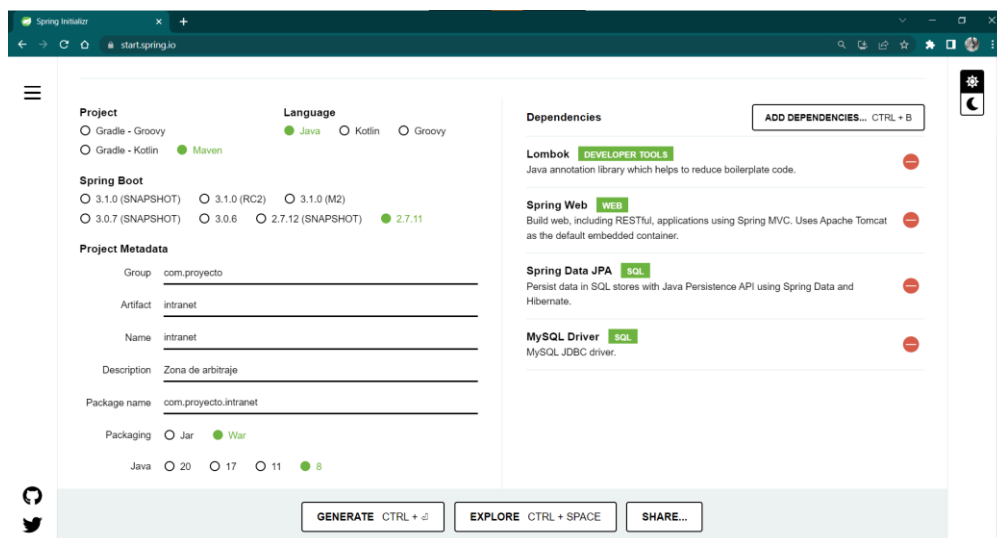
## 4.2. Implementación y configuración

En esta fase del proyecto nos centramos en configurar e inicializar los entornos, creando ambas ramas del proyecto, conectándolas entre sí e investigar la librería **PrimeNG**, para el desarrollo del Front de la aplicación.

### 4.2.1. Configuración de Back

Empezaremos con la creación de parte back de nuestro proyecto desde el inicializador de proyectos de **Spring Boot** (<https://start.spring.io/>).

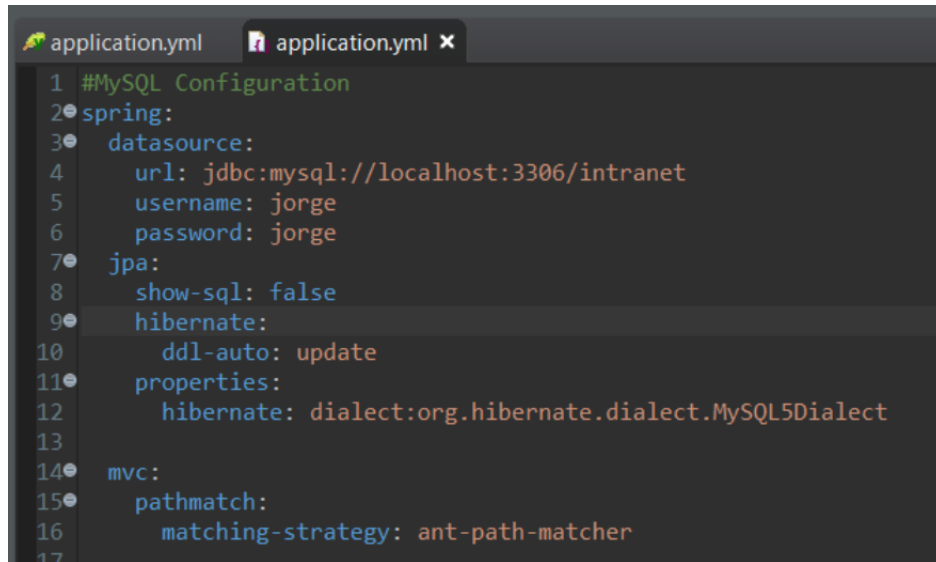
Crearemos un proyecto **Maven**, que utilice **Java** como lenguaje en su versión **8**, nombre de los paquetes y proyecto, versión de **Spring Boot**, y algunas librerías de gran utilidad, como el driver para conectar a la base de datos o JPA para las consultas del back a esta.



Al darle al botón generar tendremos un proyecto preparado para ser abierto en el entorno de desarrollo Eclipse donde tendremos que tener previamente instalado **Lombok** y **Spring Boot**.



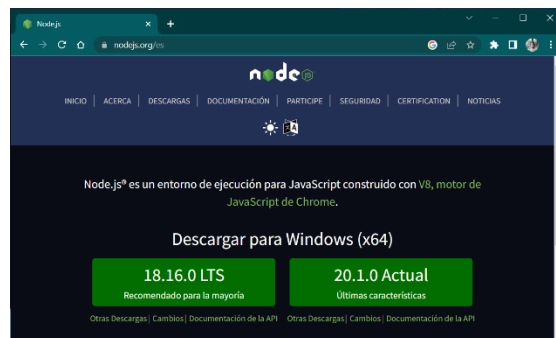
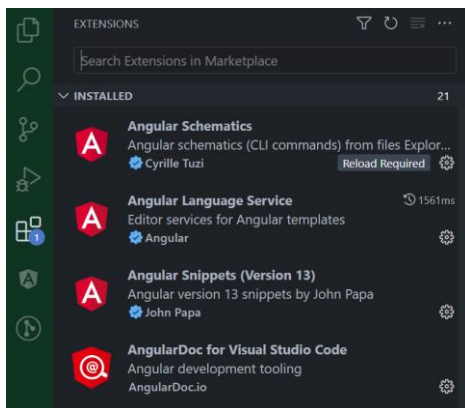
Finalmente editaremos el fichero de configuración *application.yml* (puede llamarse también *application.properties*) para conectarlo con nuestra base de datos **MySQL**, creada a partir de los scripts generados en el diseño.



```
1 #MySQL Configuration
2 spring:
3   datasource:
4     url: jdbc:mysql://localhost:3306/intranet
5     username: jorge
6     password: jorge
7   jpa:
8     show-sql: false
9   hibernate:
10    ddl-auto: update
11    properties:
12      hibernate: dialect:org.hibernate.dialect.MySQL5Dialect
13
14 mvc:
15   pathmatch:
16     matching-strategy: ant-path-matcher
```

#### 4.2.2. Configuración de Front

Una vez configurado el Back, empezaremos a configurar el Front de nuestra aplicación. Lo primero instalaremos en el entorno de desarrollo **Visual Studio Code** las siguientes extensiones de **Angular** y la versión 16 de **NodeJS** o superior.



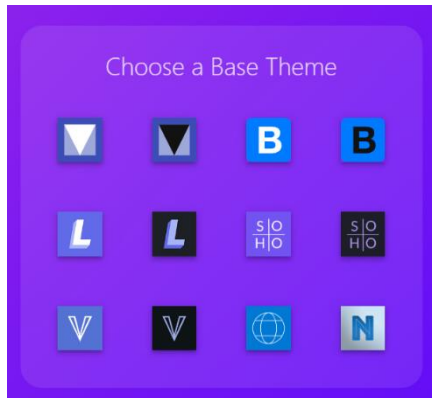
Con el entorno ya preparado crearemos el proyecto mediante el uso comandos como se puede ver en la documentación de su página oficial ([tutorial](#)).

A continuación instalaremos las librerías de **PrimeNG**, lo cual descargará e instalará en nuestro proyecto esta librería de manera automática, los únicos pasos que quedarían sería importar los estilos y los módulos a usar en el fichero *app.module.ts*. La información de estos pasos se puede encontrar en el apartado instalación de su página web donde tendremos unas instrucciones y un vídeo explicativo ([indicaciones](#)).

### 4.2.3. Investigación librería PrimeNG

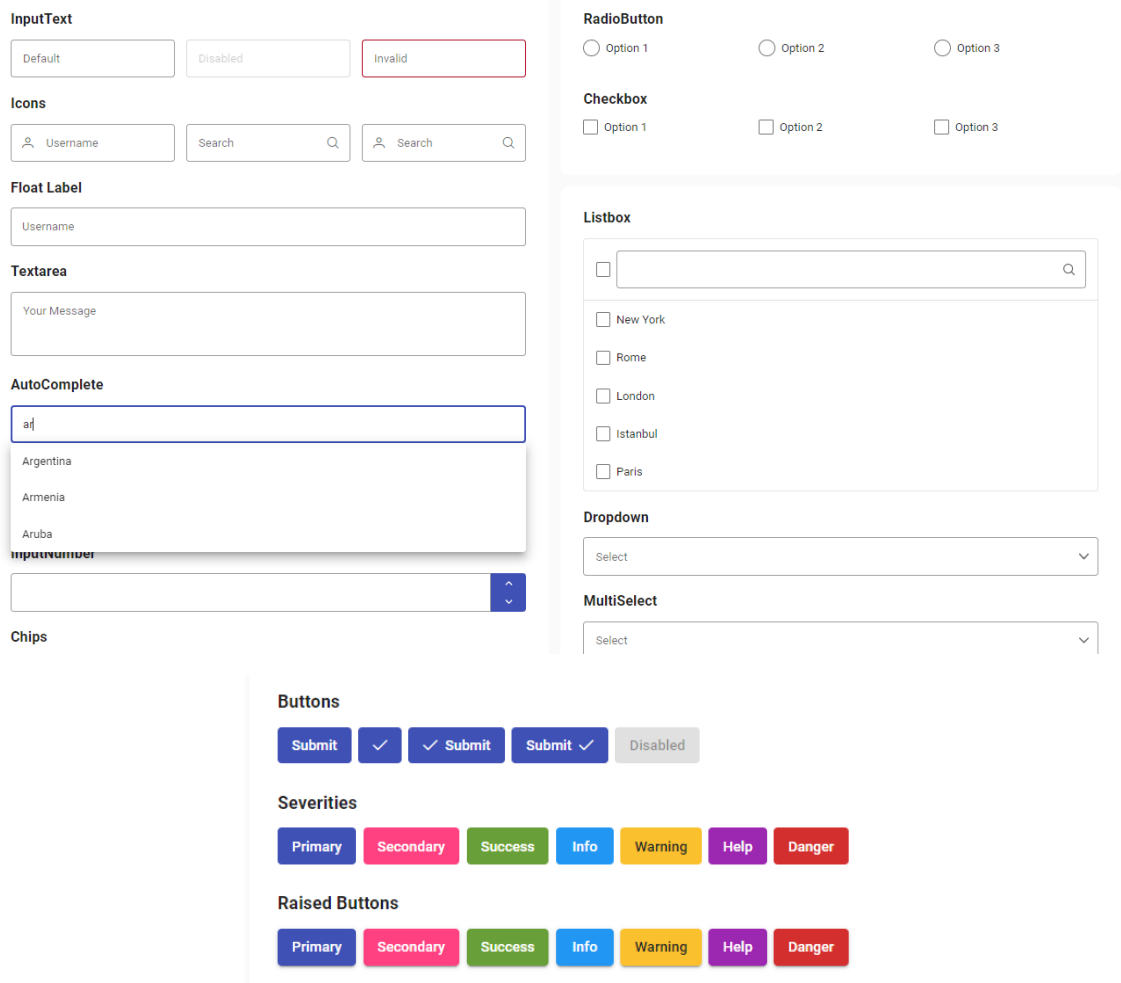
Un apartado muy importante de este proyecto dentro de los objetivos es aprender y aprovecharse de una gran librería como es PrimeNG.

El proceso de instalación ya ha sido realizado y es muy sencillo, lo verdaderamente complejo es comprender como funcionan estos componentes ya desarrollados para poder sacarles el máximo partido dentro de nuestra aplicación y tenerlo presente para el desarrollo de futuras aplicaciones como una librería realmente útil.



Una buena primera manera de adentrarse es consultando su [diseñador de temas](#), donde podemos encontrarnos con una interesante variedad de estilos ya creados. Accediendo a estos, podemos ver un catálogo de componentes y probarlos.

Encontraremos desde componente básicos como campos de texto, botones, radios, checkboxes... a componentes de gran utilidad como autocompletables.

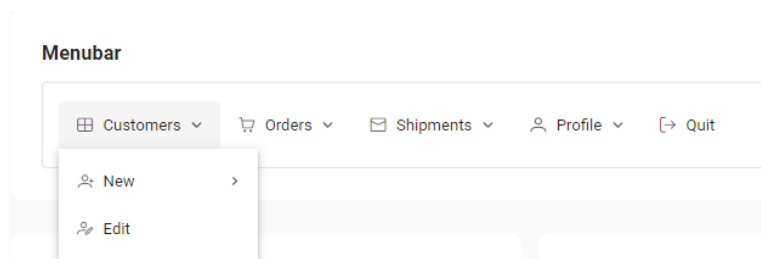


Un elemento muy interesante y de gran utilidad que usaremos en la aplicación es el componente tabla, donde podremos tener un campo de búsqueda, acciones, selección y paginación entre otras utilidades.

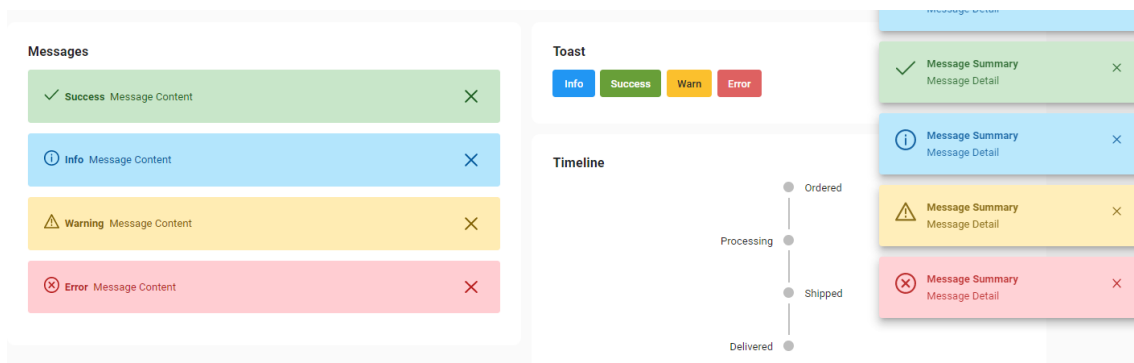
<input type="checkbox"/>	Name ↑↓	Country ↑↓	Representative ↑↓	Date ↑↓	Status ↑↓	Activity ↑↓	
<input type="checkbox"/>	James Butt	Algeria	Ioni Bowcher	2015-09-13	UNQUALIFIED	<div><div></div></div>	
<input type="checkbox"/>	Josephine Darakjy	Egypt	Amy Elsner	2019-02-09	PROPOSAL	<div><div></div></div>	
<input type="checkbox"/>	Art Venere	Panama	Asiya Javayant	2017-05-13	QUALIFIED	<div><div></div></div>	
<input type="checkbox"/>	Lenna Paprocki	Slovenia	Xuxue Feng	2020-09-15	NEW	<div><div></div></div>	
<input type="checkbox"/>	Donette Foller	South Africa	Asiya Javayant	2016-05-20	PROPOSAL	<div><div></div></div>	
<input type="checkbox"/>	Simona Morasca	Egypt	Ivan Magalhaes	2018-02-16	QUALIFIED	<div><div></div></div>	
<input type="checkbox"/>	Mitsue Tollner	Paraguay	Ivan Magalhaes	2018-02-19	RENEWAL	<div><div></div></div>	
<input type="checkbox"/>	Leota Dilliard	Serbia	Onyama Limba	2019-08-13	RENEWAL	<div><div></div></div>	
<input type="checkbox"/>	Sage Wieser	Egypt	Ivan Magalhaes	2018-11-21	UNQUALIFIED	<div><div></div></div>	
<input type="checkbox"/>	Kris Marrier	Mexico	Onyama Limba	2015-07-07	PROPOSAL	<div><div></div></div>	

Showing 1 to 10 of 200 entries

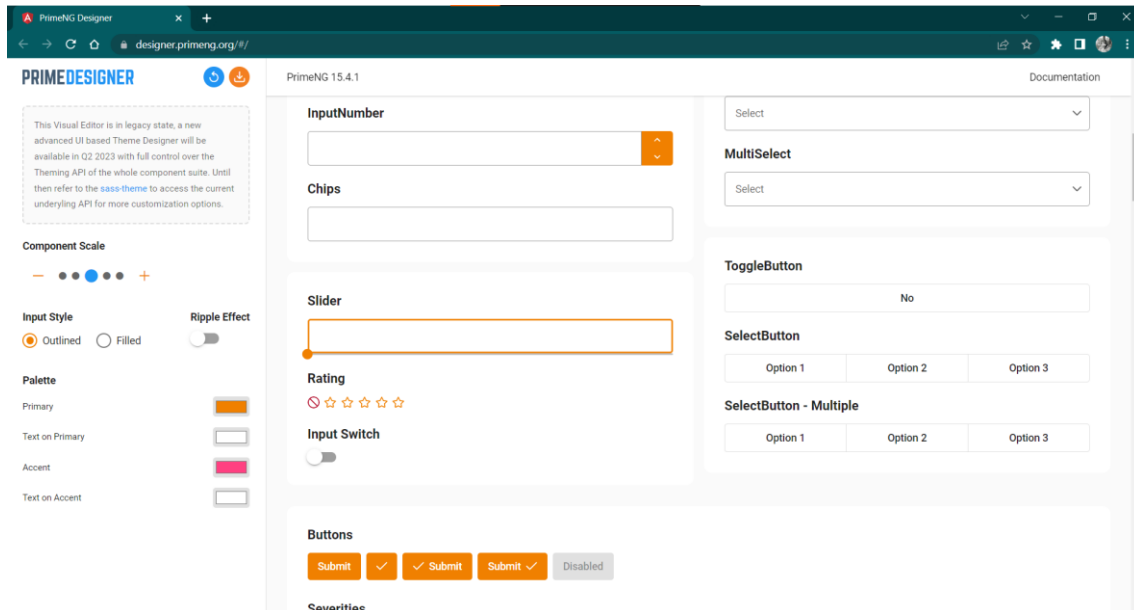
Otro componente a destacar será el menú superior de navegación de nuestra aplicación, para el cual disponemos de una gran opción.



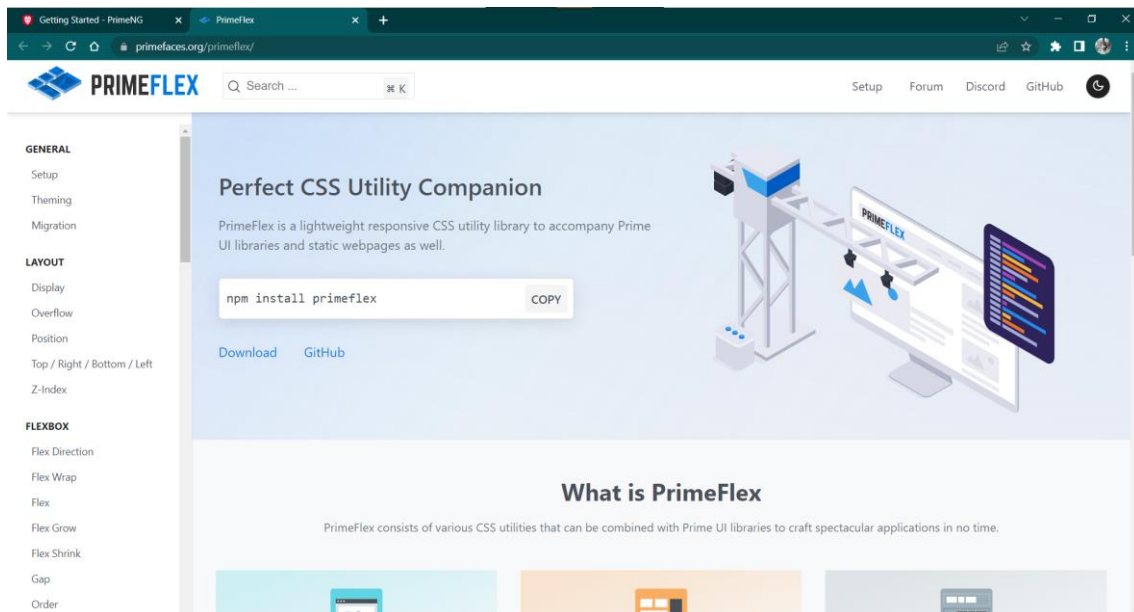
Aunque no parezca gran cosas los componentes de aviso me parecen un componente muy necesario en estas aplicaciones, ya que es la manera que tiene la aplicación de comunicarse con el usuario. En este caso contaremos con mensajes que podremos eliminar si queremos, aunque los avisos emergentes son otra gran opción.



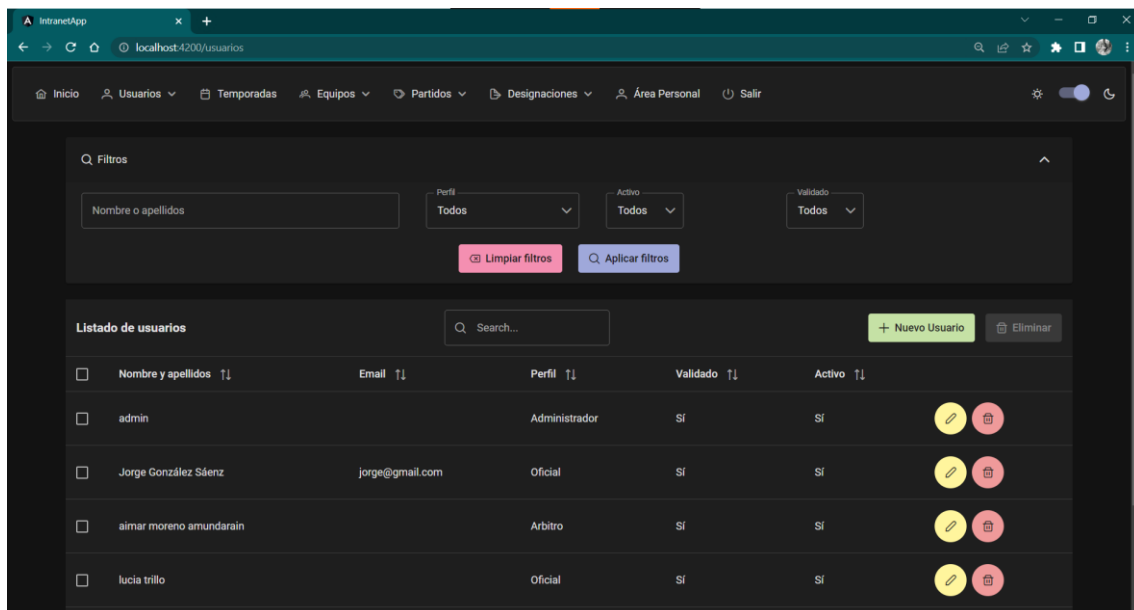
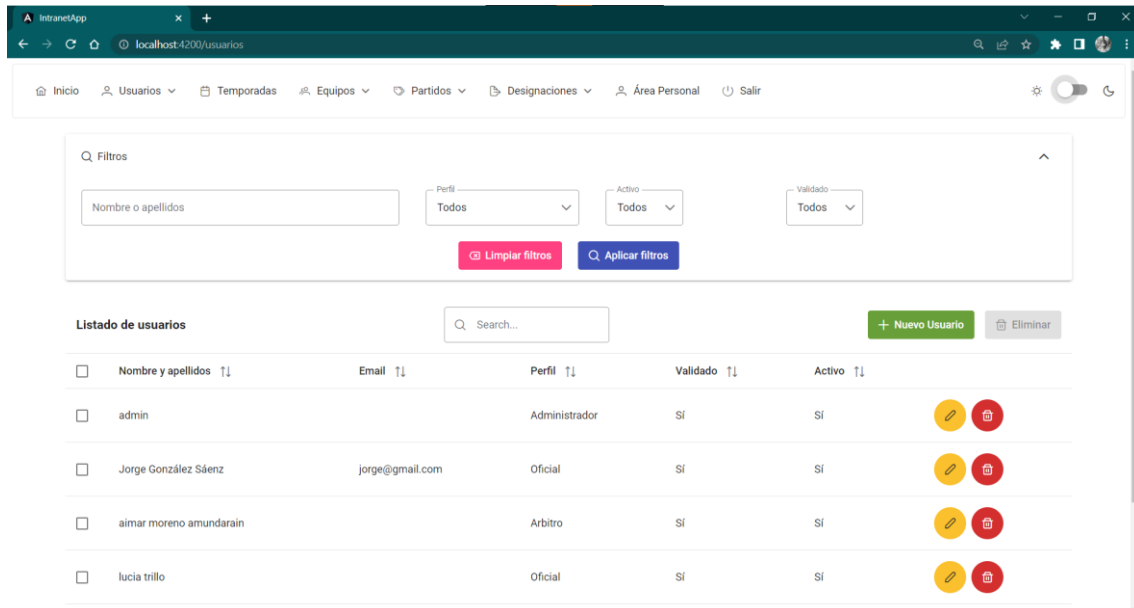
Un gran aspecto de esta librería es que ofrece unos temas por defecto que me agradan personalmente, los cuales incluso podemos modificar desde el propio editor de temas y descargarlos la hoja de estilos.



Añadir que cuenta con su propio sistema de **CSS**, con clases predefinidas para usar cómodamente a la hora de aplicar estilos a los elementos de nuestras páginas **HTML** de los componentes **Angular**.



Por último comentar que en esta fase de investigación, desarrollé un selector de modo de cambiar el tema, mediante un interruptor en el menú de navegación de modo claro a modo oscuro.



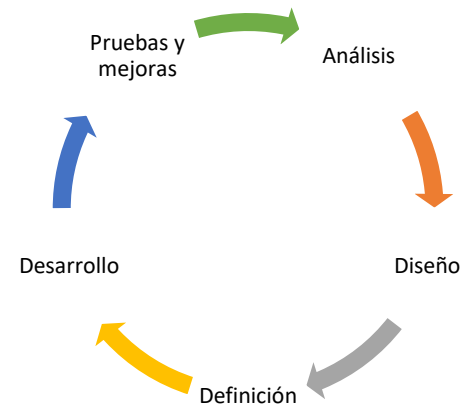
Puede parecer una funcionalidad no necesaria a la hora de desarrollar una página web, pero veo una gran utilidad en cambiar de modo si se desea, como usuario que en su vida personal usa el modo oscuro siempre que es posible.

Todo ello es gracias al uso de esta librería, ya que usar los componentes y los temas de su misma librería estos cambian su estilo como se puede apreciar en la paleta de colores de los distintos de estos.

### 4.3. Desarrollo

Comenzaremos con el apartado de desarrollo dividiendo el desarrollo total de la aplicación por pantallas o desarrollos diferenciados, donde para cada uno de ellos aplicaremos un desarrollo similar a las metodologías ágiles:

- Análisis inicial.
- Revisión del diseño realizado previamente por si fuera necesario algún cambio.
- Definición de los métodos necesarios a desarrollar.
- Creación del Back/Front.
- Pruebas evaluación de posibles cambios.



Estos pasos no terminan al llegar al último, sino que es un proceso cíclico hasta que se considere en el paso de pruebas y mejoras que está finalizado.

#### 4.3.1. Login de usuarios

Aparentemente parece un desarrollo sencillo dado que se trata de un formulario de usuario y contraseña que dará acceso a la aplicación, como se puede observar en el diseño inicial. Esto implica crear el back de los usuarios y sus servicios en Front, donde no solo accederemos a la aplicación, sino tener que dispondremos de la información del usuario está logueado en todo momento, como por ejemplo su perfil por si fuera necesario algún momento.

Acceso a la Intranet

login

password

Acceder

[No tengo cuenta, registrarse](#)


El formulario muestra un campo de texto con el placeholder 'login', un campo de texto con el placeholder 'password' y un botón 'Acceder'. Debajo del botón hay un enlace hipervinculado que dice 'No tengo cuenta, registrarse'.

Para este apartado estuve investigando el modulo **Spring Security**, el cual permite la autenticación por clave de usuario o token. En el momento que un usuario accede a la aplicación en vez de almacenar toda la información de este en la sesión del navegador almacena un código, por el cual mediante una llamada al back le devuelve toda la información del usuario.

Estoy acostumbrado a usarlo, pero no lo he implementado en otros proyectos, por ello decidí investigar si era posible añadirlo a mi aplicación sin suponer un gran esfuerzo.



Tras dedicarle unos días tuve que abandonarlo ya que no podía dedicarle más tiempo a esta tarea dado el alcance del proyecto, y decidí guardar la información del usuario en sesión para poder continuar con el resto de desarrollos, ya que mi objetivo final es tener una aplicación completa y funcional, pudiendo añadir esta mejora en un futuro sin problemas.



The screenshot shows a web browser window with the address bar displaying 'localhost:4200/login'. The page has a header with 'IntranetApp' and navigation links for 'Acceso' and 'Registro'. The main content area is titled 'Zona de Arbitraje' and 'Acceso a la aplicación'. It contains a login form with fields for 'Usuario' and 'Contraseña' (password), a 'Login' button, and a link for 'Acceso a registro'.

En caso de introducir el loguearnos correctamente nos redirigirá a la pantalla de inicio con un mensaje satisfactorio, mientras que si no es correcta nos aparecerá un mensaje de error bajo el formulario.



The screenshot shows the application's main interface. At the top, there is a navigation bar with links for 'Inicio', 'Usuarios', 'Temporadas', 'Equipos', 'Partidos', 'Designaciones', 'Area Personal', and 'Salir'. Below the navigation bar, there is a green message box indicating a successful login: 'Usuario "admin" logueado perfil: Administrador'. Below this, there is a 'Login' button and a link for 'Acceso a registro'. At the bottom, there is a red error message box: 'Error Usuario o contraseña incorrectos'.

Tras realizar varias pruebas y comprobar su funcionamiento, decidí mejorar la pantalla con el uso del componente para contraseñas de la librería **PrimeNG**, el cual nos permite ver el contenido de esta mediante un icono, o incluso en los formularios futuros el nivel de seguridad de esta, dando por concluido el desarrollo.

### 4.3.2. Listado y detalle de usuarios

En un proyecto con más horas de desarrollo o mayor alcance tanto esta tarea con las siguientes que consten de un listado y detalle, debieran ser dos tareas separadas, pero en este caso, como no es un proyecto muy grande se encuentran en la misma fase.

Tanto el listado como el detalle, al ser las primeras pantallas de este tipo conllevarán un desarrollo más largo puesto que serán la base a usar en el resto de páginas de la aplicación. El diseño inicial del listado constaba de una tabla, con unos campos para filtrar los registros de esta y unos botones para crear, editar y eliminar.

<input type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado
<input checked="" type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado
<input type="checkbox"/>	Nombre y apellidos	email	perfil	activo	validado

En la creación de este desarrollo no solo crearemos las clases para obtener los usuarios, sino que crearemos los servicios para los perfiles.

Gracias al proceso de investigación previo, utilicé el componente de tabla para crear el listado, al cual le modifiqué la paginación para que la hiciera en back y acorde al filtro superior. Permite ordenar los registros visibles por columnas, buscar entre estos o la multiselección para una acción de eliminado masivo, incluyendo la acción de edición y eliminación para cada registro.

<input type="checkbox"/>	Nombre y apellidos	Email	Perfil	Validado	Activo	
<input type="checkbox"/>	admin		Administrador	Si	Si	
<input type="checkbox"/>	Jorge González Sáenz	jorge@gmail.com	Oficial	Si	Si	
<input type="checkbox"/>	aimar moreno amundarain		Arbitro	Si	Si	
<input type="checkbox"/>	lucia trillo		Oficial	Si	Si	
<input type="checkbox"/>	sandra		Arbitro	Si	No	

La acción de eliminado será física en caso de no tener ninguna referencia del usuario en la aplicación, mientras que si hay algún elemento asociado a este, realizaremos un borrado lógico donde se desactivará el usuario.

El filtro de la parte superior es un componente acordeón el cual contiene campos de interés para filtrar en está tabla.

The screenshot shows a filter section at the top with a search bar containing 'jorge', a 'Perfil' dropdown set to 'Oficial', an 'Activo' dropdown set to 'Todos', and a 'Validado' dropdown set to 'Todos'. Below these are buttons for 'Limpiar filtros' and 'Aplicar filtros'. The main section is titled 'Listado de usuarios' and includes a search bar with 'ofi'. It features a table with columns: 'Nombre y apellidos', 'Email', 'Perfil', 'Validado', 'Activo', and action icons. One user is listed: Jorge González Sáenz with email jorge@gmail.com, profile Oficial, validated Si, and active Si. The pagination shows '1 1 de 1 usuarios'.

Para poder aprovechar la búsqueda en el componente de la tabla, se ofrece aumentar los registros por página, ya que sólo podremos aplicar esta búsqueda sobre los registros visibles.

This screenshot is similar to the previous one but shows two users in the table: Jorge González Sáenz and lucia trillo. The pagination dropdown is open, showing options for 5, 10, and 20 records per page. The current selection is 5.

Para finalizar con el listado, comentar que la acción de eliminar tiene un dialogo de confirmación, ya que toda acción que elimine registros de la base de datos debería ser obligatoria una confirmación.

The dialog box is titled 'Eliminación de usuario'. It contains a warning icon and the text: 'Va a ser eliminado el usuario "Jorge", esta acción no se puede deshacer ¿Estás seguro?'. At the bottom, there are two buttons: 'Cancelar' and 'Confirmar eliminación'.

Esta pantalla ha sido un gran ejemplo de cambio y mejoras, hasta obtener el diseño final, por ejemplo el cambio del paginado del Front al Back, la paginación y búsqueda en la tabla, el control de la eliminación (si no puede ser físico por restricciones se hace lógico), el dialogo de confirmación entre otros.

La pantalla de detalle de usuario es la misma para el proceso de creación de un nuevo registro como para la edición del seleccionado, y dicho componente es usado también en el **Área Personal y Registro**, ya que al ser un componente de Angular nos da esta versatilidad de reutilización de componentes.

El diseño inicial muestra un formulario sencillo, pero con la opción de tipo de documento de identidad que se dejó como posible mejora, la cual incluiría su validación pertinente.

El resultado varía un poco del diseño, pero cumple perfectamente con su función.

Los desplegables son otro componente de la librería **PrimeNG**, y en este caso no tienen una opción inicial de *“Seleccione un valor”* ya que al ser campos obligatorios siempre van a estar rellenos con un valor.

Comentar por último el comportamiento ya mencionado previamente del componente de contraseñas, el cual nos permite ver el texto que estamos relleno y ver el nivel de seguridad a medida que escribimos en el campo.

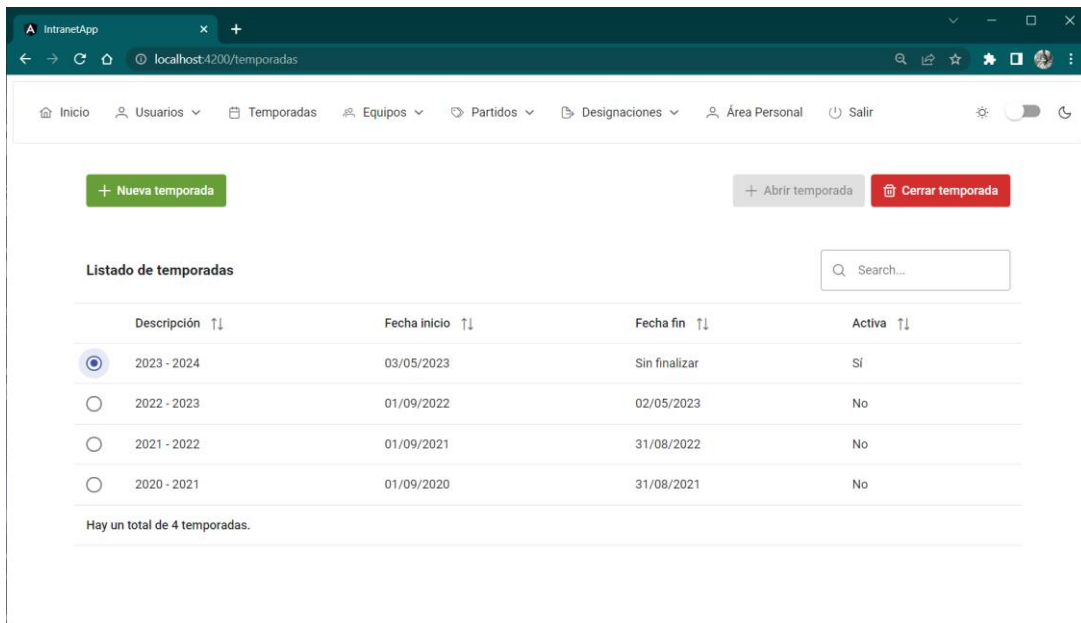
### 4.3.3. Listado de temporadas

Este desarrollo es un mantenimiento para gestionar las temporadas, ya que solo podremos crear equipos, partidos y designaciones para la temporada actual, con lo que veremos un listado con todas las temporadas, pero solo podremos realizar acciones sobre la última.



<input type="checkbox"/>	Descripcion	Fecha Inicio	Fecha Fin	Activa
<input checked="" type="checkbox"/>	Descripcion	Fecha Inicio	Fecha Fin	Activa
<input type="checkbox"/>	Descripcion	Fecha Inicio	Fecha Fin	Activa

La acción de **Crear temporada**, cerrará la última si esta está activa, y creará un nuevo registro, la acción de **Abrir temporada** solo se habilitará si la última esta inactiva, activando la temporada, y la acción de **Cerrar temporada** funciona de manera inversa a la anterior.



Descripción	Fecha inicio	Fecha fin	Activa
2023 - 2024	03/05/2023	Sin finalizar	Si
2022 - 2023	01/09/2022	02/05/2023	No
2021 - 2022	01/09/2021	31/08/2022	No
2020 - 2021	01/09/2020	31/08/2021	No

Hay un total de 4 temporadas.

Al cerrar una temporada se registra la fecha de la acción en el campo fecha fin.

#### 4.3.4. Listado y detalle de equipos

Este desarrollo constará de un listado de equipos y su creación/edición, para el cual no solo crearemos los servicios necesarios de estos, sino que tendremos que crear las pertinentes a categorías, ya que cada equipo pertenece a una categoría.

**Intranet**

Usuarios Temporadas **Equipos** Partidos Designaciones Area Personal

nombre categoria municipio

Crear Equipo Editar Eliminar

<input type="checkbox"/>	Nombre	Categoria	Municipio
<input checked="" type="checkbox"/>	Nombre	Categoria	Municipio
<input type="checkbox"/>	Nombre	Categoria	Municipio

El listado de equipos tiene las mismas funcionalidades que el de usuarios ya que su estructura está basado en él, el único cambio notable son los campos del filtro, ya que estos dependen del contenido de la tabla.

IntranetApp x +

localhost:4200/equipos

Inicio Usuarios Temporadas **Equipos** Partidos Designaciones Area Personal Salir

Q Filtros

Temporada 2023 - 2024 Nombre Categoria Junior Femenino Municipio

Limpiar filtros Aplicar filtros

Listado de equipos

Search...

+ Nuevo Equipo Eliminar

<input type="checkbox"/>	Nombre ↑↓	Categoria ↑↓	Municipio ↑↓	Temporada ↑↓	
<input type="checkbox"/>	Najerilla	Junior Femenino	Najería	2023 - 2024	
<input type="checkbox"/>	Najería B	Junior Femenino	Najería	2023 - 2024	

1 2 de 2 equipos.

<< < 1 > >> 5

Una vez desarrollado este listado, en la fase de pruebas y mejora del desarrollo, pensé que una mejor idea es hacer al equipo dependiente de la temporada, ya que de un año para otro pocos son los equipos que se mantienen igual, y además resultara más sencillo a la hora de crear partidos ya que estos sí que dependían de la temporada actual.

La pantalla de creación y edición es un formulario sencillo, el cual cambia de su diseño original al añadir/crear el partido.

**Intranet**

Usuarios Temporadas **Equipos** Partidos Designaciones Area Personal

**Detalle del Equipo** Guardar Volver

nombre categoria municipio pabellon direccion

Inicio Usuarios Temporadas Equipos Partidos Designaciones Área Personal Salir

**Detalle de equipo** Volver Editar equipo

Nombre \* Najerilla Categoría \* Junior Femenino Temporada \* 2023 - 2024

Municipio \* Najera Pabellón \* Multiusos Dirección \* Calle pradillo 2

En este caso todos los campos son obligatorios y en caso de no tener alguno rellenado al darle a guardar muestra un mensaje de error.

Como se puede observar en la imagen anterior el campo temporada está deshabilitado ya que sólo podremos crear/editar equipos para la temporada activa, si intentamos crear un equipo sin temporada activa, nos mostrará un error, mientras que si entramos a editar un equipo de otra temporada todos los campos se mostraran inhabilitados y no tendremos opción de guardar.

**Detalle de equipo** Volver

Nombre \* Clavijo Categoría \* Junior Femenino Temporada \* 2022 - 2023

Municipio \* Logroño Pabellón \* Pol espartero Dirección \* Calle breton

**Detalle de equipo** Volver Añadir equipo

**Error** No se ha encontrado una temporada activa

Nombre \* Categoría \* Seleccione un valor Temporada \*

Municipio \* Pabellón \* Dirección \*

En este último caso se podría deshabilitar el botón de añadir, pero al no tener temporada rellenada y ser obligatoria no nos dejaría guardar el registro.

### 4.3.5. Listado y detalle de partidos

Este desarrollo constará de un listado de partidos y su creación/edición, para el cual no sólo crearemos los servicios necesarios de estos, sino que tendremos que crear las pertinentes a competiciones, y alguno para clases ya creadas como equipos.

**Intranet**

Usuarios Temporadas Equipos **Partidos** Designaciones Area Personal

temporada competicion jornada categoria

equipo local equipo visitante fecha

Crear Partido Editar Eliminar

<input type="checkbox"/>	Temporad	Competici	Jornad	Categoría	Local	Visitante	fecha
<input checked="" type="checkbox"/>	Temporad	Competicio	Jornada	Categoría	Local	Visitante	fecha
<input type="checkbox"/>	Temporad	Competicio	Jornada	Categoría	Local	Visitante	fecha

El listado de partidos tiene las mismas funcionalidades que los anteriores, y como en el caso anterior, el único cambio notable son los campos del filtro, ya que estos dependen del contenido de la tabla. La relación con temporada es la misma que con equipos, sólo podremos crear/editar para la temporada actual.

Q Filtros

Temporada 2023 - 2024 Categoría Todas Competición Todas Jornada Fecha

Equipo Local Equipo Visitante

Limpiar filtros Aplicar filtros

**Listado de partidos** Search...

+ Nuevo Partido Eliminar

<input type="checkbox"/>	Temporada ↑↓	Competición ↑↓	Jornada ↑↓	Categoría ↑↓	Local ↑↓	Visitante ↑↓	Fecha ↑↓	
<input type="checkbox"/>	2023 - 2024	Liga	1	Junior Femenino	Najerilla	Najera B	30/05/2023 12:11	
<input type="checkbox"/>	2023 - 2024	Copa	2	Senior Masculino	Arnedo SM	Alfaro Twin SM	13/05/2023 13:35	

1 - 2 de 2 partidos. << < 1 > >> 5

Una característica de este filtro, es que los campos de equipos no se habilitarán hasta que se haya cumplimentado la categoría, llamando a un servicio que recogerá solo los pertinentes. Estos serán listados en un componente autocompletable que también funciona como delegable si se pulsa en el icono lateral.

Q Filtros

Temporada 2023 - 2024 Categoría Junior Femenino

Equipo Local Equipo Visitante

Najerilla

Najera B



La pantalla de creación y edición es un formulario sencillo, el cual no cambia de su diseño original al añadir/crear el partido.

#### Detalle de partido

En esta pantalla todos los campos son obligatorios. En caso de no tener alguno rellenado, al darle a guardar muestra un mensaje de error. Se comportará igual que la anterior pantalla con respecto a solo poder actuar sobre la temporada activa.

Una característica a destacar de estos formularios es que cuando rellenas y borras el contenido de campos obligatorios, se marca como error en color rojo, ya que son obligatorios en los formularios, ocurre lo mismo al estar editando y eliminar un campo.

En este caso el formulario vuelve a funcionar como en el filtro donde tendremos inicialmente los equipos inactivos y en cuanto cambiemos a una categoría podremos buscarlos escribiendo parte del nombre, o seleccionarlos desde el desplegable.

Por último añadir que se ha usado otro componente de la librería **PrimeNG**, en este caso el de calendarios, donde podemos habilitarle una botonera inferior y añadirle horas, minutos y segundos si se desea. Este mismo componente también ha sido usado en el filtro del listado pero sin horas y minutos.

#### 4.3.6. Listado y detalle de designaciones

Este desarrollo constará de un listado de designaciones y su creación/edición, para el cual no sólo crearemos los servicios necesarios de estos, también algunos para clases ya creadas como partidos o competiciones.

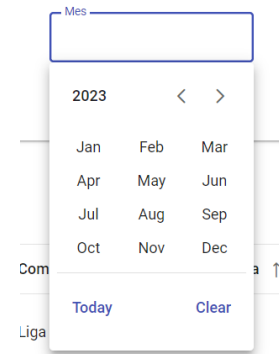
El listado de equipos tiene las mismas funcionalidades que los anteriores, y como en el caso anterior el único cambio notable son los campos del filtro, ya que estos dependen del contenido de la tabla. La relación con temporada es la misma que con equipos, solo podremos crear/editar para la temporada actual.

Esta pantalla es la primera donde nos aparecerán distintas opciones dependiendo del perfil con el usuario que accede a la aplicación, a diferencia del resto de listados que sólo tiene acceso un usuario **Administrador**, en ésta los usuarios con perfiles **Arbitro** y **Oficial**, solo pueden ver las designaciones propias y generar un informe.

En este caso para el filtro hemos aprovechado el componente calendario para mostrar sólo los meses del año para poder filtrar a nivel mensual, ya que suele ser el periodo al que se facturan los partidos habitualmente.

Añadir que el filtro de usuarios únicamente está habilitado para los administradores ya que el resto de usuarios tendrán las búsquedas filtradas por su propio usuario.

El resto de usuarios sólo podrán filtrar por sus designaciones y generar un informe de las mismas que abordaremos un otro apartado (4.3.8).



	Temporada ↑↓	Competición ↑↓	Jornada ↑↓	Categoría ↑↓	Local ↑↓	Visitante ↑↓	Fecha designación ↑↓
<input type="checkbox"/>	2023 - 2024	Liga	1	Junior Femenino	Najerilla	Najera B	19/05/2023
<input type="checkbox"/>	2023 - 2024	Copa	2	Senior Masculino	Arnedo SM	Alfaro Twin SM	26/05/2023

1 - 2 de 2 designaciones.

La pantalla de creación y edición es un formulario sencillo, el cual cambia de su diseño original al añadir/crear la designación.

En esta pantalla no todos los campos son obligatorios, pero la mayoría necesarios y en caso de no tener alguno de estos rellenado, al darle a guardar muestra un mensaje de error. Se comportará igual que la anterior pantalla con respecto a solo poder actuar sobre a temporada activa.

El formulario se comportará de manera dinámica, como los filtros previos, pero de una manera más compleja:

- Seleccionando una categoría cargaremos cuanto personal se puede asignar a dicho encuentros (nº de árbitros y oficiales), cargando la información de estos para poder usarla en los componentes autocompletables; sino que también habilitará y cargará las competiciones con partidos creados para dicha categoría.
- Seleccionado las competición habilitaremos y cargaremos el campo jornadas, en función de esta y la categoría.
- Seleccionando la jornada, se habilitarán los partidos disponibles en función a este y la competición y jornada seleccionadas previamente, si sólo se encuentra una se seleccionará y buscara los partidos de manera automática.

Con esto ya podremos seleccionar un partido de los disponibles, ya que este campo es obligatorio para crear/editar la designación, el resto actuaría como un buscador o simplemente a nivel informativo. En caso de cambiar algún campo en el proceso volver a buscar por los registros que queden seccionados.

**Detalle de designación**

[← Volver](#) [Añadir designación](#)

Temporada 2023 - 2024	Categoría Junior Femenino	Competición Seleccione un valor	Jornada ▼
Partido *			
Arbitro principal *	Arbitro auxiliar 1	Liga	
Oficial 1 *	Oficial 2		

Los usuarios cargados en este proceso deberán estar validados y activados, y como mínimo se debe seleccionar un usuario de cada tipo. El resto son opcionales y dependiendo de los que tenga la categoría configurados veremos más o menos campos.

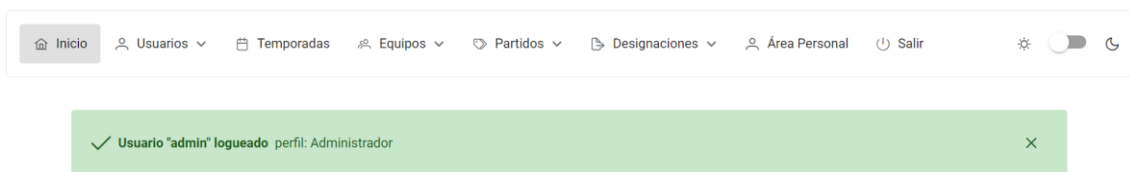
#### 4.3.7. Acceso con perfiles

En este apartado del desarrollo trataremos el acceso con los distintos perfiles a la aplicación, mencionado en el anterior apartado, pero realizado en este punto.

Para empezar cuando estamos fuera de la aplicación veremos que en la barra superior de navegación solo vemos las opciones de **Acceso y Registro**.



Si accedemos a la aplicación con perfil **Administrador**, veremos cómo se nos muestran todos los apartados que acabamos de desarrollar, pudiendo acceder a todos los listados y a su área personal.



Por otro lado si accedemos con un perfil de **Árbitro** u **Oficial**, solo podremos acceder a sus designaciones y área personal.



Al pulsar el botón salir nos aparecerá un dialogo de confirmación el cual si aceptamos cambiara los elementos del menú a los iniciales, y eliminara la información del usuario de la sesión del navegador.

Por último el cambio más notable sería el listado de designaciones a nivel funcional, donde un administrador puede crear, editar y eliminar registros mientras que el resto de usuarios únicamente pueden ver sus designaciones y generar un informe.

Listado de partidos								+ Nueva Designación		Eliminar
<input type="checkbox"/>	Temporada ↑↓	Competición ↑↓	Jornada ↑↓	Categoría ↑↓	Local ↑↓	Visitante ↑↓	Fecha designación ↑↓			
<input type="checkbox"/>	2023 - 2024	Liga	1	Junior Femenino	Najerilla	Najera B	19/05/2023			

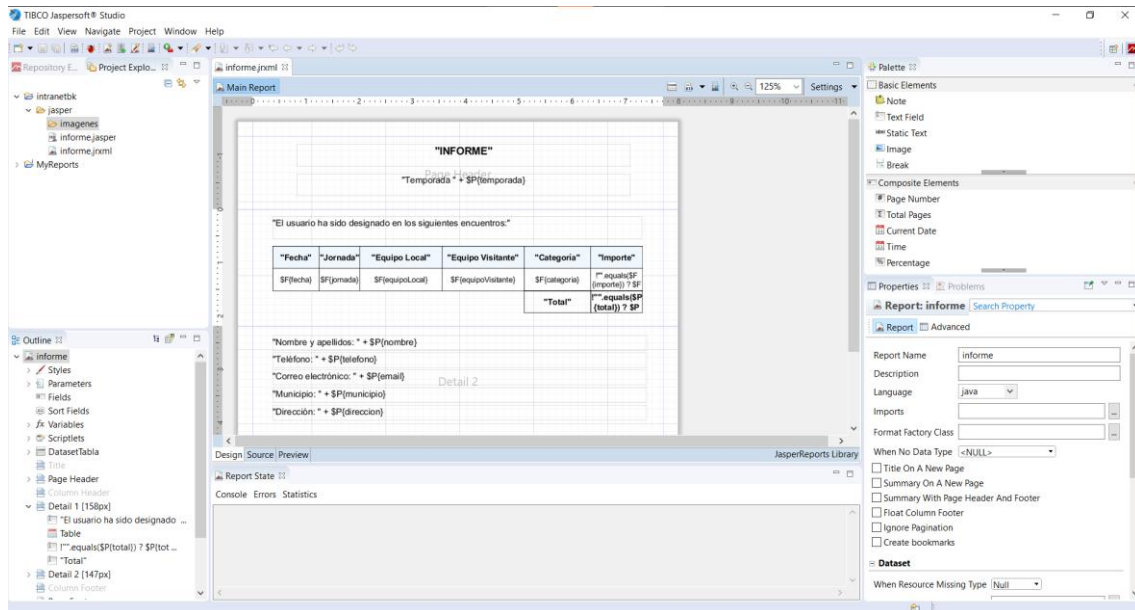
  

Listado de partidos								+ Descargar informe designaciones		
<input type="checkbox"/>	Temporada ↑↓	Competición ↑↓	Jornada ↑↓	Categoría ↑↓	Local ↑↓	Visitante ↑↓	Fecha designación ↑↓			
<input type="checkbox"/>	2023 - 2024	Liga	1	Junior Femenino	Najerilla	Najera B	19/05/2023			

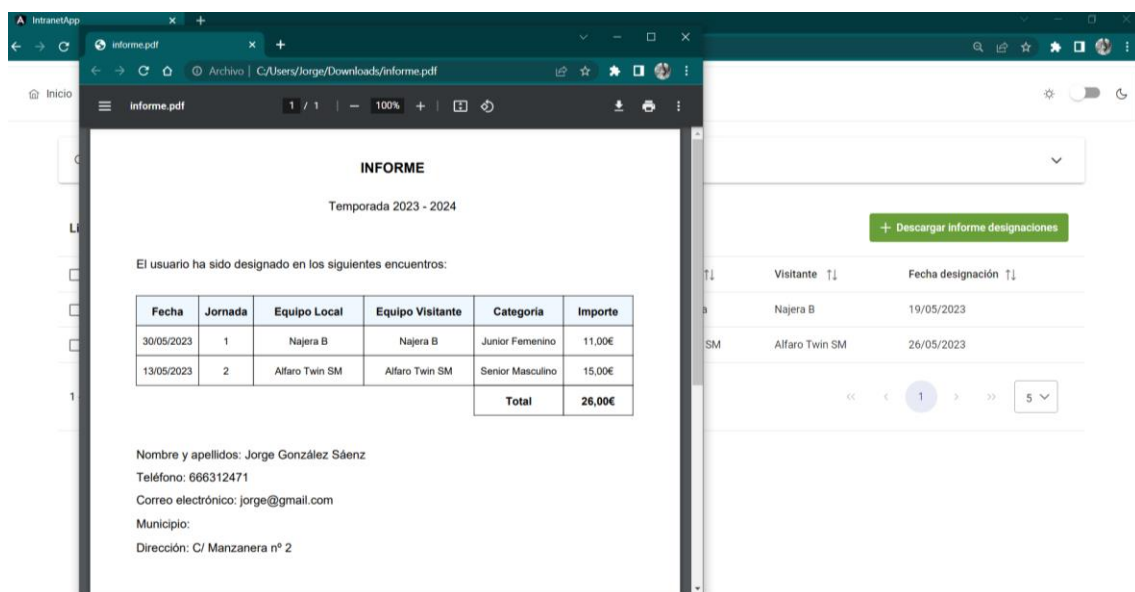
### 4.3.8. Generar informe JasperReports

**JasperReports** es una librería para la creación de informes, donde crearemos un documento de manera dinámica desde la aplicación. Esto nos permitirá crear un informe de las designaciones de los usuarios al pulsar un botón desde la aplicación.

Para ello diseñaremos una plantilla desde la aplicación [Jaspersoft Studio](#), donde se definirán una serie de parámetros; crearemos un método en back que rellenará estos y generará un documento, en este caso en formato pdf, el cual será descargado por el front de la aplicación.



En caso de no seleccionar ningún registro, aparecerán todas las designaciones acorde con los campos del filtro de búsqueda, mientras que si seleccionamos alguno sólo aparecerán dichas designaciones en la tabla del informe. El resto de datos se recogen de la información del usuario.



#### 4.4. Documentación y pruebas generales

Una vez concluidos todos los desarrollos comenzará la documentación de los mismos, ya que al utilizar metodologías ágiles no todas las funcionalidades están definidas desde un principio, con lo que una vez terminados éstos, comenzaré con la documentación.

Este paso debiera ser realizado conforme avanza el desarrollo, pero no ha sido el caso, aun así se tenían definidas las funcionalidades básicas de cada apartado y se ha tomado nota de las mejoras conforme se ha ido desarrollando.

Junto con la documentación se han realizado pruebas de toda la aplicación, uniando todas sus funcionalidades. Para poder probar la aplicación se pueden realizar por ejemplo las siguientes acciones:

- Creación de usuarios desde registro o listado, edición para validar y activar dichos usuarios, para poder ser designados.
- Creación/edición de varios equipos, a ser posible de la misma categoría, creación/edición de partidos para dicha categoría.
- Creación/edición de designación para partidos y usuarios creados previamente.
- Descarga de informe de designaciones.

Se pueden realizar más pruebas en los distintos apartados, pero con las acciones anteriores, recorreremos la aplicación completa.

## 5. Ampliación y posibles mejoras

La aplicación es completa a nivel funcional, pero como todas puede ser mejorada en muchos aspectos, y gracias a las tecnologías usadas tenemos una aplicación modular la cual puede ser mejorada tanto en general como algún componente en específico.

Las principales mejoras que implementaría serían:

- Implementación del módulo de Spring Security, ya mencionado previamente para aumentar la seguridad d la aplicación proporcionando autenticación, autorización y protección para aplicaciones Java.
- Aumento de las validaciones en formularios, para hacerlos más completos y funcionales, ya que no se utilizar del todo el control de errores en los componentes utilizados.
- Mejorar el aspecto visual de la página web, incluyendo imágenes y logos personalizados, que le doten de un aspecto más amigable para el usuario que la fuera a utilizar.
- Subir la página web a un portal o servicio compatible, para poder utilizarla sin necesidad de tener la base de datos, el back y front en un entorno local.
- Mejorar la calidad del informe generado, dado que este es un prototipo de lo que podría ser un informe de calidad para ser usado como recibo.



## 6. Conclusión

Has sido un proyecto con el que he disfrutado y aprendido, aspectos fundamentales que deben ir relacionados si se quiere obtener un buen resultado.

Se han cumplido los objetivos iniciales de la aplicación, obteniendo una página web funcional que utiliza herramientas actuales y nos permite añadir y mejorar funcionalidades de manera coherente.

El resultado final no solo cumple con los objetivos iniciales, sino que puede ser útil como base para proyectos reales, tanto a nivel general como algún módulo en concreto.

El uso de metodologías ágiles ha permitido organizar mejor el trabajo, separando en distintas tareas las partes a desarrollar. Esto nos ha permitido modificarlas o mejorarlas sin alterar al resto del proyecto, y las pruebas realizadas en cada tarea han minimizado la cantidad de errores en el proyecto a nivel general.

Estoy satisfecho a nivel técnico con las tecnologías que elegí, ya que mi experiencia con ellas me ha permitido llevar a cabo la fase de desarrollo de manera fluida. Aunque inicialmente no estaba familiarizado con la librería **PrimeNG**, me dediqué a investigar y aprender su uso sin mayores dificultades. He logrado implementar con éxito los diferentes componentes, comprendiendo su funcionamiento y adaptándolos según mis necesidades específicas.

En resumen, he creado una aplicación para la gestión de arbitrajes que permite la gestión de usuarios, equipos, partidos y designaciones, con diferentes acciones dependiendo del perfil de usuario.

## 7. Bibliografía

**Pencil:** Prototipado de interfaz de usuario de código abierto y multiplataforma.

Enlace: <https://pencil.evolus.vn/>

**DB Designer:** Herramienta de diseño de bases de datos en línea.

Enlace: <https://app.dbdesigner.net/designer>

**Spring Boot:** Framework de desarrollo de aplicaciones Java.

Enlace: <https://spring.io/projects/spring-boot>

**Visual Studio Code:** Editor de código fuente desarrollado por Microsoft.

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

**Node.js:** Entorno de ejecución de JavaScript del lado del servidor.

Enlace: <https://nodejs.org/es>

**Angular:** Plataforma de desarrollo de aplicaciones web de código abierto basada en TypeScript.

Enlace: <https://angular.io/>

**Angular CLI:** Interfaz de línea de comandos (CLI) para trabajar con Angular.

Enlace: <https://www.angularjswiki.com/angularcli/>

**PrimeNG:** Librería de componentes para el desarrollo con Angular.

Enlace: <https://primeng.org/>

**Stack Overflow:** sitio de preguntas y respuestas para programadores.

Enlace: <https://stackoverflow.com/>

## ANEXO - Puesta en marcha del proyecto

### 1. Configuración del Back

Realizar los siguientes pasos:

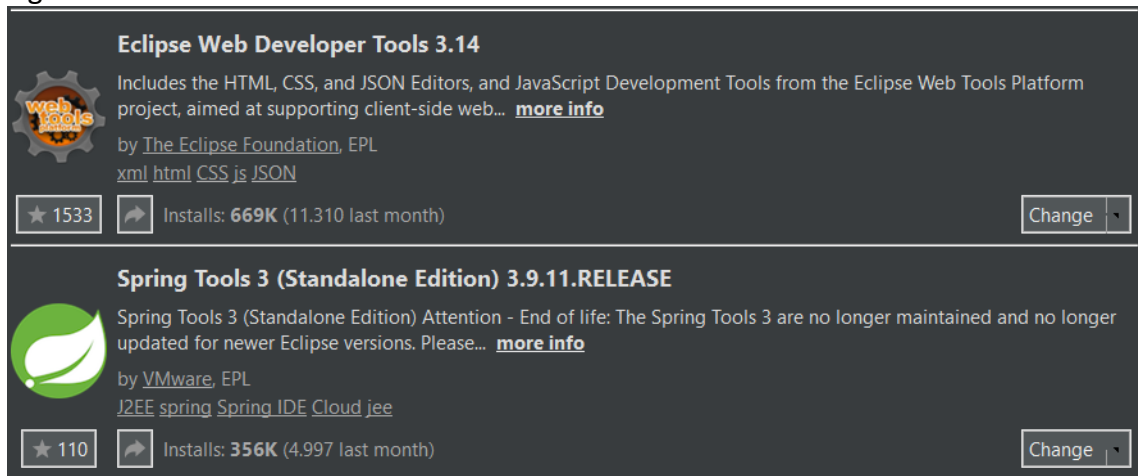
1. Descargar e instalar **Java 1.8**.
2. Descargar y dejar en un directorio a la elección **Apache Maven** en su versión **3.6.3**.
3. Configurar variables de entorno:
  - Crear las siguientes variables.

JAVA_HOME	C:\Program Files\Java\jdk1.8.0_291
MAVEN_HOME	C:\apache-maven-3.6.3

- Las añadimos las variables creadas en *Path*.

%JAVA_HOME%\bin
%MAVEN_HOME%\bin

4. Descargamos **Eclipse** portable o instalable y una vez iniciado instalaremos las siguientes extensiones:



5. Ejecutaremos **lombok.jar**, realizando la instalación de este en **Eclipse** (<https://projectlombok.org/download>).
6. Cambiar la versión de java en el entorno desde: *Window -> Preferences -> Java -> Installed JREs -> Add jdk1.8.X*.
7. Abriremos proyecto desde el entorno de desarrollo.
8. Ejecutaremos el proyecto en el siguiente orden:
  - Maven Update
  - Maven Clean
  - Maven Install
  - Run as Spring Boot App

En al siguientes ejecuciones solo será necesario realizar el último paso.

## 2. Configuración del Front

Realizar los siguientes pasos:

1. Descargar e instalar **Node.js** (versión **16.14** o superior).
2. Descargar e instalar **Visual Studio Code**.
3. Instalar las extensiones de angular desde el menú o con los siguientes comandos:  
code --install-extension Angular.ng-template  
code --install-extension AngularDoc.angular-doc-vsc
4. Abrir la carpeta que contiene el proyecto y ejecutar los siguientes comandos desde un terminal:
  - npm install
  - ng serve

En las siguientes ejecuciones solo será necesario realizar el último paso.

## 3. Configuración adicional

Crear la base de datos con los scripts ubicados en la carpeta *setup* de la rama de back.

Crear el siguiente directorio *C:\compartido\app\intranetbk\jasper* y ubicar el contenido de la carpeta *\src\main\resources\jasper*, para ubicar la plantilla del informe.