# Requerimientos funcionales:

## Gestión de actividades:

* Los usuarios deben poder gestionar actividades, incluyendo asignación y seguimiento del estado actual.
* Debe existir un registro histórico que muestre la evolución de cada actividad.

## Roles de usuario:

* Debe haber dos roles de usuario: "Administrador" y "Ejecutor".
* Los administradores pueden crear, modificar y visualizar actividades.
* Los ejecutores pueden ver y actualizar las tareas asignadas a ellos.

## Funcionalidades de administrador:

* Crear actividades con descripción, prioridad y usuario responsable.
* Modificar detalles de actividades y su estado (Pendiente [por defecto], Finalizado, Eliminado).
* Visualizar información actual de actividades y registros de auditoría.
* Exportar datos a formato CSV.

## Funcionalidades de ejecutor:

* Ver y actualizar tareas asignadas pendientes.

## Navegación de datos:

* Permitir a los usuarios navegar los datos con criterios de ordenamiento, filtros y ajuste de cantidad de registros visibles.

# Requerimientos no funcionales:

## Seguridad:

* Implementar autenticación JWT para garantizar la seguridad de las operaciones.

## Rendimiento y escalabilidad:

* La aplicación debe ser eficiente y escalable para gestionar grandes volúmenes de datos y usuarios concurrentes.

## Usabilidad:

* La interfaz debe ser intuitiva y fácil de usar para los diferentes roles de usuario.

# Restricciones:

## Tecnologías requeridas:

* Utilizar el stack tecnológico pedido por la organización (NestJS, TypeORM, Angular, JWT, MySQL, Apache HTTP Server, PM2).
* La aplicación debe ser desarrollada como una aplicación web.

## Roles y permisos:

* Los permisos y funcionalidades deben estar restringidos según el rol de cada usuario (Administrador o Ejecutor).

## Exportación de datos:

* La exportación de datos debe ser compatible con el formato CSV para cumplir con los requerimientos de los administradores.