



## Caso práctico

Se requiere desarrollar una API REST la cual permita la gestión de usuarios y de sus respectivos fichajes de trabajo.

Para ello, vamos a necesitar 2 modelos con los siguientes atributos:

User:

- id → número identificativo del usuario
- createdAt → fecha de creación del usuario
- updatedAt → fecha de edición del usuario
- deletedAt → fecha de borrado del usuario
- name → nombre del usuario
- email → correo electrónico del usuario

WorkEntry

- id → número identificativo del fichaje
- userId → número identificativo del usuario que ha realizado el fichaje
- createdAt → fecha de creación del fichaje
- updatedAt → fecha de edición del fichaje
- deletedAt → fecha de borrado del fichaje
- startDate → fecha de entrada (Y-m-d H:i:s)
- endDate → fecha de salida (Y-m-d H:i:s)

Un usuario puede tener muchos fichajes pero un fichaje solo tiene un único usuario.

Necesitaremos los siguientes endpoints/rutas:

User:

- POST create user
- PUT update user
- DELETE user by id
- GET user by id
- GET all users

WorkEntry:

- POST create workEntry
- PUT update workEntry
- DELETE workEntry by id
- GET workEntry by id
- GET workEntry by userId

### A tener en cuenta:

- Al eliminar un usuario o fichaje, se rellenará el campo deletedAt y nunca se devolverá al listar usuarios o fichajes, en ningún caso se borrará la fila de bbdd.
- Todos los campos tanto de usuario como de fichaje deben de ser rellenados obligatoriamente a excepción de los deletedAt que serán null por defecto y endDate que puede ser null.
- El atributo endDate de fichaje nunca puede ser inferior a startDate.

### Requisitos:

- Crear una aplicación funcional sobre Symfony u otro framework PHP.
- Definir un sistema que permita la detección de problemas en nuestra aplicación.
- Definir un entorno sobre el que ejecutarlo (preferiblemente Docker).

### Se valora:

- Uso de Symfony.
- Uso de PHP 7.4 o superior (fuertemente tipado).
- Uso de Arquitectura Hexagonal, DDD y principios SOLID.

### Método de entrega:

- Subir el proyecto a un repositorio y enviar tanto el enlace al repositorio como el explicativo al correo proporcionado.

