Objetivo de la prueba:

El objetivo de la prueba es conocer las habilidades del candidato. La prueba nos ayudará a reconocer las debilidades y fortalezas del candidato de cara al desarrollo.

¿Qué necesitas construir?

Dado el esquema de BDD diseña un API REST con las siguientes funciones:

1. Listado de empleados con el departamento, cargo y salario actuales. Ordenado por fecha de contratación y limitado a 50

Los campos a devolver son:

id (employees.emp\_no)

nombre (employees.first\_name)

apellidos (employees.last\_name)

departamento (departments.dept\_name)

cargo (titles.title)

salario (salaries.salary)

fecha\_contratacion (employees.hire\_date)

1. Obtención del perfil de un empleado con el departamento, cargo y salario actuales. Los campos a devolver son:

id (employees.emp\_no)

nombre (employees.first\_name)

apellidos (employees.last\_name)

genero (employees.gender)

departamento (departments.dept\_name)

cargo (titles.title)

salario (salaries.salary)

fecha\_contratacion (employees.hire\_date)

fecha\_nacimiento (employees.birth\_date)

1. Inserción de un empleado con los siguientes parámetros:

first\_name

last\_name

birth\_name

gender

dept\_no

title

salary

1. A continuación, y teniendo como base el API, construir una interfaz sencilla que muestre el listado de empleados obtenido a través de una petición http a la API
2. Una vez obtenido el listado el usuario debe ser capaz de hacer click en el identificador de usuario y ver el perfil del empleado. Para obtener el perfil se utilizará otra petición http a la API en la que se le pasará el id de empleado
3. Como requisito adicional añadir en el listado de empleados un botón de “Crear empleado” con el correspondiente formulario de alta. Una vez creado el empleado, volver al listado.

Para el alta de empleado hacer uso de otra petición http al API

Adjuntamos el dump de la base de datos

Requisitos

Usar PHP 7.2

No usar frameworks de terceras partes, hacerlo todo desde cero.

Está permitido el uso de librerías propias y de terceros como abstracción de base de datos (adodb), gestión de plantillas (smarty, twig, blade, etc…), objetos de caché, jquery, bootstrap, etc..

¿Qué estamos buscando?

* Adecuación a los requisitos - ¿Cumple tu app todos los requisitos solicitados?
* Calidad del código - ¿Tu código es limpio y está bien estructurado?
* Utilización de MVC u otros patrones de diseño (NO FRAMEWORKS)
* Se valorará hacer la parte de interface con Bootstrap

¿Cómo entregarlo?

Adjunta en un zip todos los archivos necesarios para usar tu app

Para el API Rest añade instrucciones y/o ejemplos de uso y envíanoslo (documentación)

Si te atreves, además, cuelga tu aplicación en algún server para que podamos probarla “live”, si no dispones de ningún server también puedes colgarla en github, bitbucket o gitlab y pasarnos el link del repositorio.

Conclusiones

Esperemos que disfrutes de la prueba. Si tienes alguna duda háznosla llegar antes de comenzar la prueba