Prueba Servinform

Sergi Sanz Carreres

Instalaciones Requeridas

**1.- Instalación de PHP 7.4.13:** <https://windows.php.net/download#php-7.4>

**2.- Instalación de Composer**: <https://getcomposer.org/download/>

**3.- Instalación de Symfony**: <https://symfony.com/download>

**4.- MySQL:** <https://dev.mysql.com/downloads/>

Pasos necesarios para su utilización

**1.- Actualizar librerías utilizando composer:** composer u

2.- En el archivo ‘**php.ini**’ que se encuentra en la carpeta donde esta instalado el php, asegurarnos de tener descomentado la extensión ‘**extension=pdo\_mysql’**

3.- En el documento ‘**. env**’ del proyecto modificar los datos correspondientes a la configuración de MySQL (lo encontramos en una línea tal que esta: **# DATABASE\_URL="mysql://db\_user:db\_password@127.0.0.1:3306/db\_name?serverVersion=5.7”)** por los datos correspondientes a los definidos en la instalación de MySQL:

* **mysql**: Motor de base de datos a utilizar
* **db\_user**: Usuario
* **db\_password**: Contraseña a utilizar
* **db\_name**: Nombre de la base de datos

4.- Como la Base de datos ya se encuentra creada en nuestro proyecto, simplemente habrá que utilizar el comando ‘**php bin/console doctrine:migrations:migrate**’ para que cargue la estructura de la base de datos que se ha desarrollado. Posteriormente, nos conectaremos a MySQL para mediante el comando ‘**source ./202011281347.sql**’ poblar la base de datos con los datos para poder ejemplificar el programa desarrollado. Si hacemos un select de las distintas tablas obtendremos los siguientes resultados cerciorando que se han incluido datos en las tablas creadas.

Texto

Descripción generada automáticamente

5.- Ahora será necesario inicializar el servidor en symfony utilizando el comando **‘symfony server:start’** , obteniendo un resultado como el siguiente:

Imagen que contiene Texto

Descripción generada automáticamente

6.- Finalmente, en otra ventana de terminal, utilizamos el comando ‘**php Test.php**’ para ejecutar el script que hace una petición al WebService haciendo uso de CURL. De forma que le indicamos el ID del curso, devuelve un JSON con los nombres y apellidos de todos los alumnos de ese curso. En caso de que no exista ese ID o que le introduzcamos algún carácter no numérico devolverá “Se ha producido un error”.

Texto

Descripción generada automáticamente

Explicación SCRIPT

# Curso Controller:

El archivo **‘Curso Controller.php’** correspondo al WebService en Php que tiene como objetivo recibir un id de curso por POST y devolver un JSON con los nombres y apellidos de todos los alumnos vinculados al curso.

Texto

Descripción generada automáticamente

Donde en la clase ‘**CursoController’** se ha desarrollado una función ‘**cursoAlumno’** que recibe como parámetro el identificador del Curso. Posteriormente, se utiliza la librería Doctrine de Symfony para interactuar con la base de datos, donde en ella accedemos a la tabla curso y buscamos el identificador de ese curso.

Luego utilizo ‘**isset’** para comprobar si el identificador está en null y en caso de estarlo lanzar la excepción.

A continuación, creo la variable alumnos, donde se almacena los alumnos consultados de curso que corresponden con el identificador requerido que cumple con las condiciones.

Finalmente, recorremos los datos de cada alumno y los almacenamos en un array de nombre ‘**data’** que devolvemos finalmente como un json.

# Test:

El archivo **‘Test.php’** correspondo al Desarrollo de un script en PHP con una petición al WebService haciendo uso de CURL.

Texto

Descripción generada automáticamente

Donde mediante el uso de echo solicitamos al usuario el id del curso y posteriormente lo almacenamos en la variable ‘**idCurso’**. A continuación, en la variable ‘**parámetros’** almacenamos un array con **‘CURLOPT\_URL,CURLOPT\_POST,CURLOPT\_RETURNTRANSFER’** y iniciamos el **‘curl\_init()**’ que se almacena en la variable conexión, donde al ejecutarlo usando **‘curl\_exec ‘** estamos haciendo una llamada HTTP asociada a la función **‘CursoAlumno’**. En caso de que esa llamada sea satisfactoria devolveremos la respuesta que en este caso será un json de todos los alumnos que formen parte de ese curso, mientras que en caso contrario indicará un error. Finalmente, se cerrará la conexión del curl.