



POLITÉCNICA



ETSI SISTEMAS
INFORMÁTICOS

MiW – Práctica 1 PHP 2016

Objetivo:

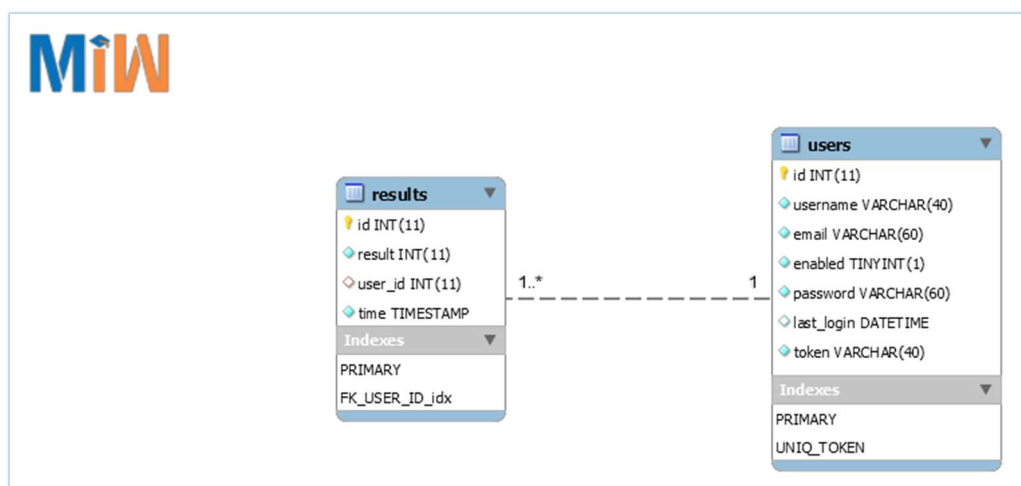
- Progresar en el conocimiento del lenguaje de scripting PHP y familiarizarse con el desarrollo de aplicaciones web completas integrando tecnologías del lado del cliente con el procesamiento en el lado del servidor.

Enunciado:

Esta práctica pretende servir de ejemplo de integración de diferentes técnicas aprendidas en la asignatura “*Back-end con tecnologías de libre distribución (PHP)*”. En la misma se profundizará en el conocimiento del lenguaje de scripting PHP, se emplearán datos almacenados en un gestor de bases de datos y adicionalmente, se utilizarán componentes desarrollados por terceros que permitirán simplificar las labores a desarrollar.

Modelo de Datos:

El modelo de datos que se debe implementar (aunque se podrá adaptar libremente) será similar al siguiente:



Se pide:

Empleando el ORM Doctrine¹, obtener las entidades correspondientes e implementar un conjunto de scripts (en el directorio src/scripts) que desde línea de comandos –y opcionalmente también vía web– permitan añadir, mostrar, actualizar y eliminar los diferentes usuarios y resultados. Los diferentes scripts mostrarán el resultado por pantalla y, en caso de recibir el parámetro --json, generarán la respuesta en formato JSON².

Las operaciones añadir, actualizar y eliminar se aplicarán sobre un único elemento. Por su parte también se implementarán operaciones que permitan mostrar un único elemento y todos los elementos.

Opcional:

- Desarrollar scripts que permitan eliminar y/o modificar múltiples datos para las diferentes entidades
- Añadir la gestión de usuarios y resultados vía web
- Empleando el framework phpunit³, desarrollar pruebas unitarias y de integración del conjunto de entidades generadas

Notas:

- Se podrán emplear cuantos componentes adicionales se estimen oportunos. Para el desarrollo web se sugiere emplear, por ejemplo, el componente symfony/routing⁴ que simplifica el desarrollo del controlador frontal.

Notas sobre la instalación:

La versión mínima necesaria del intérprete de PHP es la 5.5.0, si bien es recomendable actualizar a la última versión disponible (PHP 7.0.13 actualmente). Para conocer la versión disponible en el sistema:

```
1D:\xampp\htdocs> php --version
2PHP 7.0.13 (cli) (built: Nov  8 2016 13:33:54) ( ZTS )
3Copyright (c) 1997-2016 The PHP Group
4Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies
5    with Xdebug v2.4.1, Copyright (c) 2002-2016, by Derick Rethans
6
```

Una vez comprobada la versión del intérprete de PHP se procederá a la generación de la base de datos (miw16_results) y de un usuario y sus

¹ <http://docs.doctrine-project.org/projects/doctrine-orm/en/latest/>

² <http://www.json.org/>

³ <https://phpunit.de/>

⁴ <http://symfony.com/doc/current/book/routing.html>

correspondientes permisos de acceso en el gestor de bases de datos. Desde cualquier cliente MySQL (por ejemplo phpMyAdmin o MySQL Workbench) es posible realizar este paso (desde la opción de gestión de usuarios). En relación a este aspecto, los parámetros deberán ser establecidos en el fichero de configuración (src/config/config.php). Por ejemplo:

```
1 <?php // src/config/config.php
2
3 /*
4  * configuración SGBD
5  */
6 define('DATABASE_DBNAME', 'miw16_results');
7 define('DATABASE_USER', 'miw16_results');
8 define('DATABASE_PASSWD', '*miw16_results*');
9 define('DATABASE_DRIVER', 'pdo_mysql');
10 define('DATABASE_CHARSET', 'UTF8');
11
12 /*
13  * configuración Doctrine
14  */
15 define('PROXY_DIR', '/xampp/tmp');
16 define('ENTITY_DIR', __DIR__ . '/../Entity');
17 define('DEBUG', false); // muestra consulta SQL por la salida estándar
18
```

A continuación se empleará el gestor de dependencias composer, que procederá a instalar y configurar los componentes requeridos por la aplicación. Este comando se ejecutará desde el directorio raíz del proyecto:

```
1 D:\xampp\htdocs\MiW16\ResultsDoctrine> composer install
2 Loading composer repositories with package information
3 Installing dependencies...
4 [...]
```

Con la ejecución del comando anterior se generará en el proyecto la carpeta vendor, que es donde residen los componentes desarrollados por terceros. Una vez hecho esto... ¡ya se puede empezar a desarrollar nuevas opciones!