

## Software y estándares para la Web

---

### **P9. COMPUTACIÓN EN EL SERVIDOR**

## Contenido

|  |   |
|--|---|
| Temática del proyecto: Escritorio virtual (Virtual Desktop) .....      | 2 |
| Ejercicio 1: Creación del usuario de base de datos.....                | 3 |
| Guía para resolver el ejercicio 1 .....                                | 4 |
| Ejercicio 2: Introducción a PHP y MariaDB .....                        | 5 |
| Tarea 1. Modificación del fichero crucigrama.html.....                 | 5 |
| Tarea 2. Creación de la clase Record .....                             | 5 |
| Guía para resolver la tarea 2 .....                                    | 5 |
| Tarea 3. Creación del formulario de almacenaje de récords .....        | 5 |
| Guía para resolver la tarea 3 .....                                    | 6 |
| Tarea 4. Tratamiento de la información enviada por el formulario ..... | 6 |
| Guía para resolver la tarea 4 .....                                    | 6 |
| Tarea 5. Informe de los récords.....                                   | 6 |
| Guía para resolver la tarea 5 .....                                    | 7 |
| Resultado del ejercicio 2.....   | 7 |
| Recuerda.....  | 8 |

## Objetivos

En esta práctica se va a realizar:

- El arranque y uso del servidor de bases de datos MariaDB existente en la aplicación XAMPP en la máquina del usuario.
- La creación de una clase en PHP para el manejo y uso de una conexión con una base de datos.
- La creación de objetos con PHP para la realización de diferentes tareas en combinación con HTML y ECMAScript.
- La validación del código HTML estático y el código HTML generado

**IMPORTANTE: Recuerda las pautas de trabajo establecidas en la primera sesión de prácticas (P0. Pautas de trabajo): valida todos los documentos HTML, valida todas las hojas de estilo CSS, comprueba la adaptabilidad y la accesibilidad con las herramientas proporcionadas.**

## Temática del proyecto: Escritorio virtual (Virtual Desktop)

El proyecto Escritorio Virtual (Virtual Desktop) es evolutivo y será creado, completado y modificado en las diferentes prácticas de la asignatura.



## Ejercicio 1: Creación del usuario de base de datos

Se debe crear el usuario y la password específico en MySQL para poder utilizarlo en el resto de los ejercicios. Es obligatorio usar los siguientes datos de usuario, host y password.

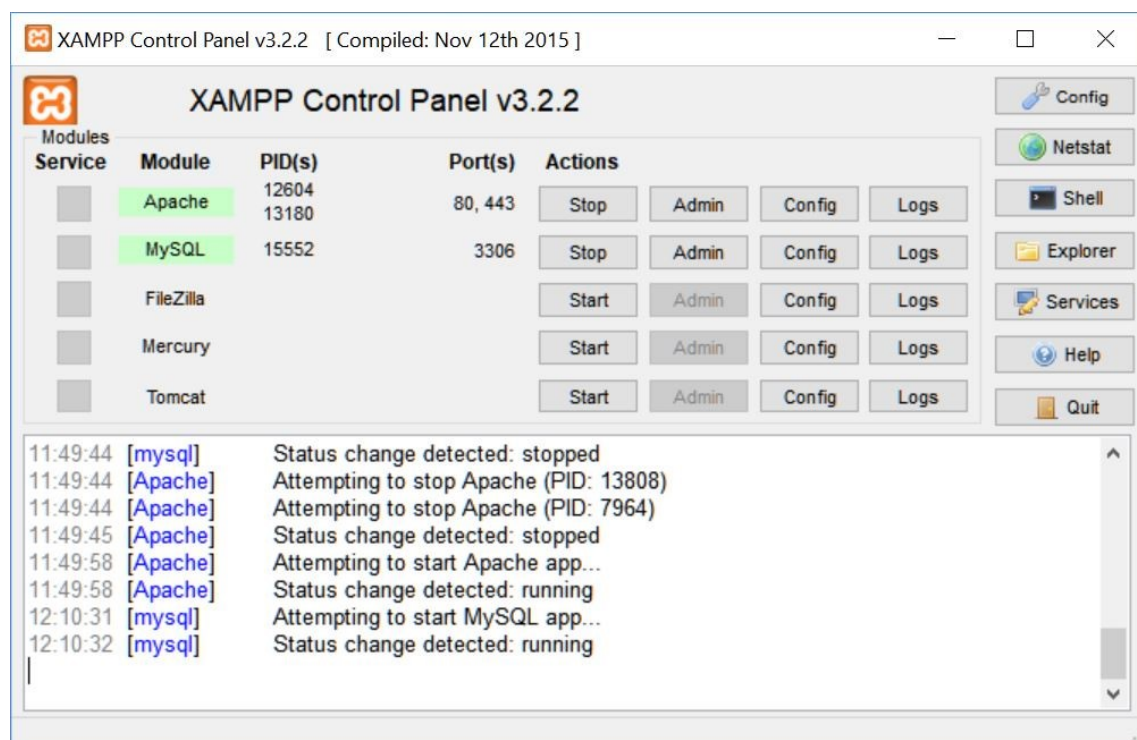
**DATOS DEL USUARIO, HOST Y PASSWORD – OBLIGATORIAMENTE DEBEN SER:**

USUARIO: **DBUSER2023**

Nombre del HOST: **Local** (localhost)

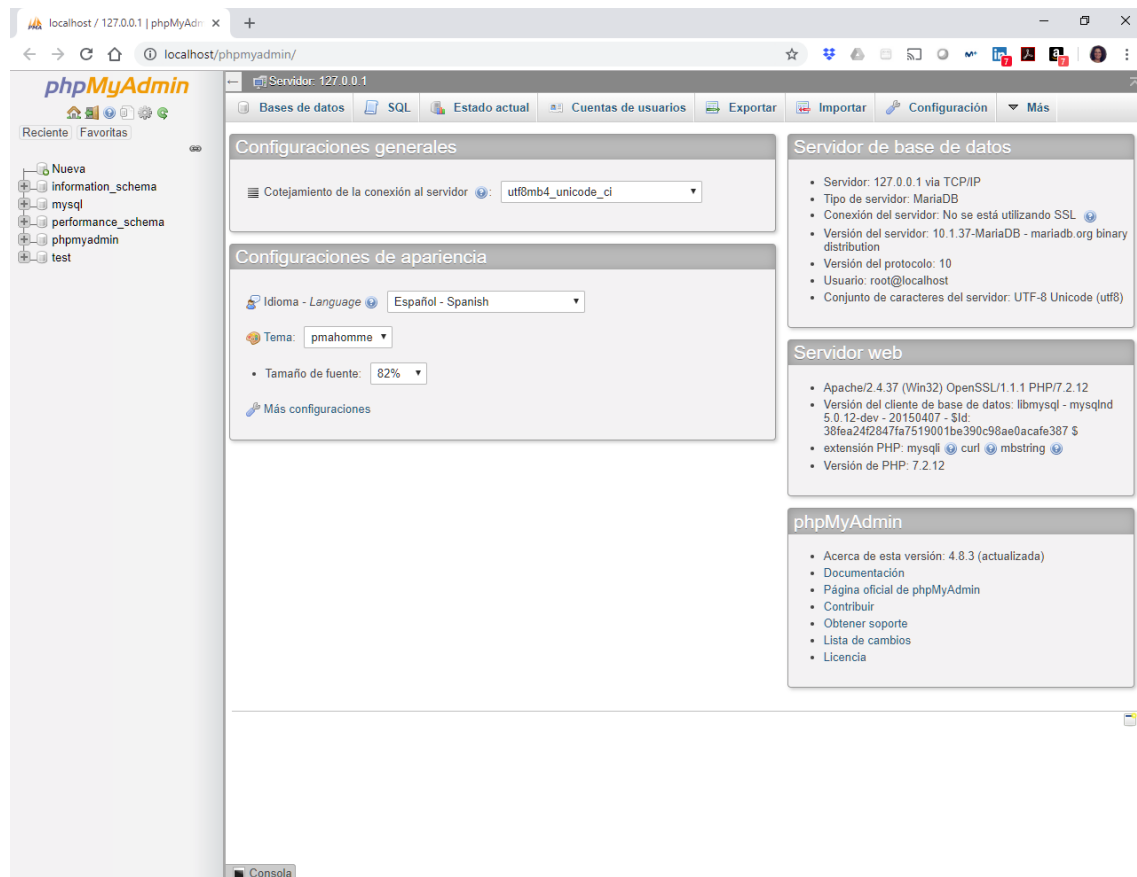
PASSWORD: **DBPSWD2023**

Una vez arrancado MySQL en XAMPP, pulsamos en Admin para entrar en la administración del MySQL

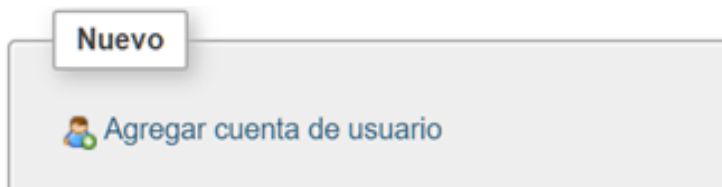


En el navegador web se abrirá la web <http://localhost/phpmyadmin/> desde donde se pueden administrar las bases de datos y los usuarios.

Seleccionar “**Cuentas de Usuarios**” en el menú, aparece la lista de los usuarios definidos, el tipo y sus privilegios.



Pulsar en “Agregar cuenta de usuario”.



Introducir el nuevo usuario con los datos especificado y con Privilegios globales. El resto de opciones se dejan por defecto. Pulsar en “**Continuar**” y el usuario ya estará creado.

## Guía para resolver el ejercicio 1

Seguir el enunciado y consultar el tema de teoría “3 - PHP: Archivos y MySQL”

## Ejercicio 2: Introducción a PHP y MariaDB

PHP es un lenguaje de programación interpretado que se utiliza en el paradigma de computación en el servidor, donde la realización de las diferentes partes de computación se delega en el servidor y el código que se recibe en el cliente es el resultado de dicha computación.

Para la realización de este ejercicio es indispensable haber completado el desarrollo de la aplicación del crucigrama matemático explicado en la sesión 2 de prácticas de ECMAScript, ya que este ejercicio utilizará el tiempo empleado por el usuario en completar el crucigrama para guardar los récords de solución del crucigrama en una base de datos.

### Tarea 1. Modificación del fichero crucigrama.html

Modifica el fichero crucigrama.html, cambiando su nombre a crucigrama.php.

Realiza los cambios necesarios en el resto de ficheros del proyecto de Escritorio Virtual para actualizar los enlaces al crucigrama.

### Tarea 2. Creación de la clase Record

Modifica el fichero crucigrama.php añadiendo la declaración de la clase Record en la parte superior del fichero.

Añade a esta clase un método constructor sin parámetros. Dentro de ese constructor añade los siguientes atributos para la clase:

- server, con el valor "localhost"
- user, con el valor "DBUSER2023"
- pass, con el valor "DBPSWD2023"
- dbname, con el valor "records"

### Guía para resolver la tarea 2

Consulta el ejemplo incluido a continuación para entender de qué forma se construye una clase utilizando el lenguaje PHP.

<http://di002.edv.uniovi.es/~cueva/php/015-Clases.php>

### Tarea 3. Creación del formulario de almacenaje de récords

Modifica el fichero crucigrama.js para añadir un nuevo método *createRecordForm*. Este método será el encargado de añadir, mediante el uso de la biblioteca jQuery, un formulario debajo del crucigrama cuando se haya completado el mismo.

Modifica el método *introduceElement* de la clase Crucigrama para añadir en dicho método la llamada a *createRecordForm* cuando se completa el crucigrama, de tal manera que se pinte el formulario una vez que se haya completado el mismo.

### Guía para resolver la tarea 3

El formulario debe tener los siguientes campos: nombre de la persona que completó el crucigrama, apellidos de la persona que completó el crucigrama, nivel (fácil, intermedio o difícil) del crucigrama realizado y tiempo empleado en completar el crucigrama (en segundos).

Los campos del nivel y el tiempo empleado vendrán ya rellenos cuando se pinte el formulario y no podrán ser modificados por el usuario.

Consulta los ejemplos incluidos a continuación para comprobar cómo se debe configurar un formulario en PHP que realice una llamada al propio fichero en el que está incluido.

<http://di002.edv.uniovi.es/~cueva/php/022-Calculadora.php>

<http://di002.edv.uniovi.es/~cueva/php/025-CalculadoraDiabetica.php>

### Tarea 4. Tratamiento de la información enviada por el formulario

Añade a la clase Record el código necesario para tratar la información recibida desde el formulario y almacenar en una tabla en la base de datos “records” la información que recibida.

### Guía para resolver la tarea 4

Utilizando la herramienta phpMyAdmin incluida en el XAMPP crea en la base de datos “records” una tabla de nombre “registro” que tenga cuatro columnas: nombre, apellidos, nivel y tiempo. Cada columna debe ser del tipo que mejor se ajuste a los datos que va a contener.

Consulta los ejemplos incluidos a continuación para comprender como reconocer el envío de información desde un formulario en PHP.

<http://di002.edv.uniovi.es/~cueva/php/022-Calculadora.php>

<http://di002.edv.uniovi.es/~cueva/php/025-CalculadoraDiabetica.php>

<http://di002.edv.uniovi.es/~cueva/php/026-ClaseBotones.php>

<http://di002.edv.uniovi.es/~cueva/php/027-Botonera.php>

Consulta los enlaces incluidos a continuación para comprender como se realiza la conexión a una base de datos utilizando MySQLi desde una clase de PHP.

<https://www.php.net/manual/es/class.mysqli.php>

<https://www.ionos.es/digitalguide/servidores/know-how/guia-para-aprender-a-utilizar-mysql/>

### Tarea 5. Informe de los récords

Añade a la clase Record el código necesario mostrar la tabla con los 10 mejores récords de los usuarios para el crucigrama del nivel concreto que se está jugando.

## Guía para resolver la tarea 5

Una vez almacenada la información del último récord recibida desde el formulario, realiza una llamada al método creado en esta tarea para obtener los 10 mejores resultados del crucigrama para el nivel actual.

Utilizando el lenguaje PHP recorre la información recibida desde la base de datos y genera una lista ordenada de HTML5 donde se puedan ver los datos de los 10 récords obtenidos desde la base de datos: nombre de la persona, apellidos de la persona y tiempo empleado (en formato horas:minutos:segundos).

Añade esta lista al HTML, debajo del crucigrama matemático.

## Resultado del ejercicio 2

Una vez completado el ejercicio debería haberse modificado el fichero crucigrama.html para modificar su nombre y añadir el código necesario para completar el ejercicio.

Además, se habrá modificado el fichero crucigrama.js para añadir un nuevo método que pinta el formulario de inserción de récords en la base de datos.

Se debe incluir en el proyecto del Escritorio Virtual un fichero records.sql que contenga la sentencia de creación de la tabla registro para poder realizar la corrección del ejercicio. La sentencia se puede extraer desde phpMyAdmin.



## Recuerda

Se requiere el uso correcto de los elementos HTML5 establecidos en los ejercicios y se valorará positivamente el uso correcto y adecuado al contexto y funcionalidad, de elementos adicionales HTML5.

Se requiere el uso correcto de las propiedades de los módulos CSS establecidos en los ejercicios y se valorará positivamente el uso correcto y adecuado al contexto, de propiedades y módulos adicionales CSS.

Solamente se permite el paradigma de orientación de objetos y todo debe estar organizado con clases y objetos, tanto en ECMAScript como en PHP. Además, solamente se permite el uso de la biblioteca jQuery para realizar las llamadas a los servicios externos y la creación de contenido en el html, quedando prohibido el uso de cualquier otra biblioteca.

**IMPORTANTE:** Todas las llamadas que se realicen a la biblioteca jQuery deben estar encapsuladas en los métodos pertenecientes a las clases que se utilicen para resolver los ejercicios.

Las siguientes condiciones son de obligado cumplimiento en todo el proyecto:

- **TODOS** los documentos HTML que componen el proyecto deben ser HTML5 válidos y sin advertencias utilizando el validador de lenguajes de marcado del W3C.
  - Recuerda que el contenido de los documentos HTML puede cambiar después de la ejecución del código ECMAScript. En ese caso, se debe comprobar la validez del código HTML en todos los diferentes estados por los que pase el documento.
- No está permitido el uso indiscriminado de bloques anónimos (**div**) en los documentos HTML5, se debe priorizar el uso de contenedores semánticos HTML5.
- Se deben utilizar selectores específicos, el uso de selectores id y class deberá estar justificado a través de comentarios en las hojas de estilo que expliquen la necesidad de su utilización y para que se utilizan.
- **TODAS** las reglas de todas las hojas de estilo deben estar precedidas por un comentario donde se indique la especificidad del (o los) selectores de la regla.
- **TODAS** las hojas de estilo que se utilizan en el sitio web deben ser validas utilizando el validador CSS del W3C
- **TODAS** las hojas de estilo deben tener 0 advertencias.
  - Recuerda seleccionar en “Más opciones” el informe de “Todas las advertencias” dentro de las opciones del validador CSS del W3C.
  - Excepcionalmente se permite las advertencias referidas a la verificación de los colores (color y background-color). **OBLIGATORIAMENTE** se debe indicar mediante un comentario en la regla de la hoja de estilo afectada la herencia de colores garantizando que la advertencia ha sido comprobada, verificada y garantizando que no provoca efectos laterales no deseados.
  - Excepcionalmente se permiten las advertencias referidas a la redefinición de propiedades derivadas del uso de @media-queries. **OBLIGATORIAMENTE** se debe indicar mediante un comentario en las reglas de la hoja de estilo afectada que propiedades se están redefiniendo.
- Se debe garantizar la adaptabilidad y realizar su verificación para todos los documentos que componen el proyecto.

- Se deben utilizar medidas relativas en las hojas de estilo.
- Se debe garantizar la accesibilidad del proyecto mediante los test de las herramientas de accesibilidad para el nivel AAA de las WCAG 2.0 con 0 errores de modo automático en todos los documentos que lo componen.

El no cumplimiento de las características anteriores derivará en la invalidación del proyecto.