



APLICACIÓN WEB EMPLEANDO UN FRAMEWORK MVC

T 5.4



ENERO DE 2022

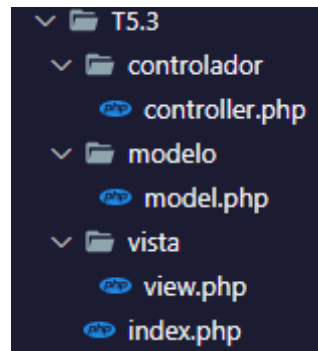
DESENVOLVEMENTO WEB EN CONTORNO SERVIDOR

Rodríguez Jácome, David

La siguiente actividad consistirá en comprobar cómo funciona el patrón MVC mediante una aplicación sencilla accediendo a una base de datos, orientada a objetos y una aplicación propia.

- 5.4.1: aplicación sencilla con MVC.

En este punto seguiremos los pasos de ejemplo para crear una aplicación sencilla mediante MVC. Organizaremos la estructura de archivos de la siguiente manera:



Los archivos tendrán esta codificación, que consiste en la conexión a una base de datos previamente creada, la consulta de su información y la vista del resultado por pantalla. El funcionamiento es el siguiente: se ejecuta el archivo “index”, que requiere del controlador; éste a su vez requiere del modelo para operar la conexión y la consulta, que devolverá al controlador este enviará los datos a la vista, que mostrará por pantalla la consulta en un formato legible para el usuario.

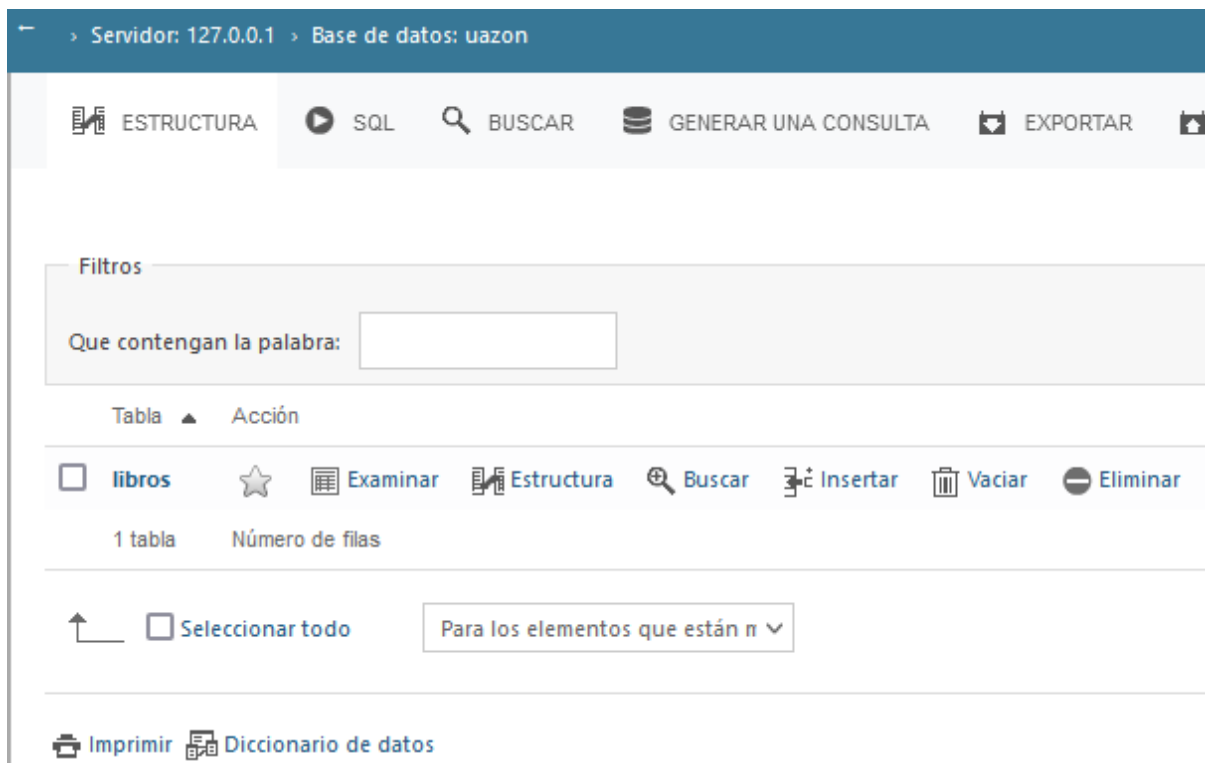
```
index.php
1 <?php
2 //Incluir variables de configuración y el único controlador
3 require 'controlador/controller.php';
4 ?>

controller.php
1 <?php
2 //Se incluye el modelo
3 require 'modelo/model.php';
4 //En $libros tenemos un array con todos los libros gracias al modelo
5 //La vista recibe un array para mostrarlo por pantalla
6 include 'vista/view.php';
7 ?>

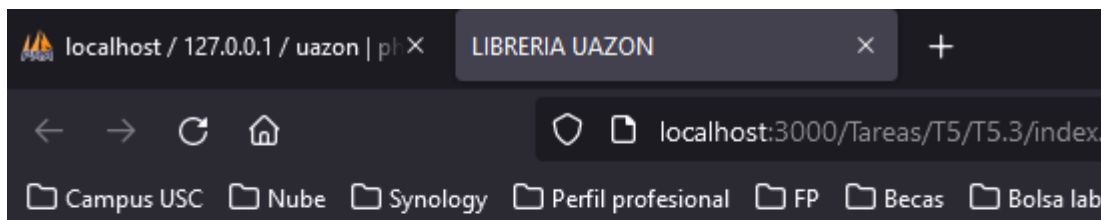
model.php
1 <?php
2 $user = 'root';
3 $pwd = '';
4 $db = new PDO('mysql:host=localhost;dbname=uazon', $user, $pwd);
5 $result = $db->query('SELECT titulo, precio FROM libros');
6 $libros = array();
7 while ($libro = $result->fetch())
8     $libros[] = $libro;
9 ?>

view.php
1 <html>
2
3 <head>
4     <title>LIBRERIA UAZON</title>
5 </head>
6
7 <body>
8     <h1>Libros dados de alta en nuestra libreria</h1>
9     <table border="1">
10         <tr>
11             <th>TITULO</th>
12             <th>PRECIO</th>
13         </tr>
14         <?php foreach ($libros as $libro) : ?>
15             <tr>
16                 <td><?php echo $libro['titulo'] ?></td>
17                 <td><?php echo number_format($libro['precio'], 2) ?></td>
18             </tr>
19         <?php endforeach; ?>
20     </table>
21 </body>
22
23 </html>
```

Esta es la base de datos que hemos creado previamente:



Y el resultado de la ejecución de la aplicación MVC es la siguiente:



Libros dados de alta en nuestra librería

TÍTULO	PRECIO
Los pilares de la Tierra	30.00
Caperucita Roja	20.00
Los tres cerditos	10.00

- 5.4.2: aplicación MVC mediante programación orientada a objetos.

Ahora haremos esa misma aplicación, pero mediante objetos. La estructura de archivos es la misma, excepto que no hay base de datos a la que conectarse. La diferencia radica en que la información de consulta que se reflejará en la vista se encuentra almacenada en un array de objetos Coche en el controlador. En el modelo hay una clase Coche con setters y getters que

toma los datos de la clase Coche del controlador y almacena los atributos de los objetos en variables. Desde el “index” se instancia un objeto de la clase Coche del controlador y además requiere del modelo; una vez hecho esto, se envía la información a la vista, que contiene una tabla donde se rellenan los campos con la información extraída de los objetos de la clase Coche en el controlador por el modelo, mostrando el resultado por pantalla al usuario.

Archivos del patrón MVC orientado a objetos:

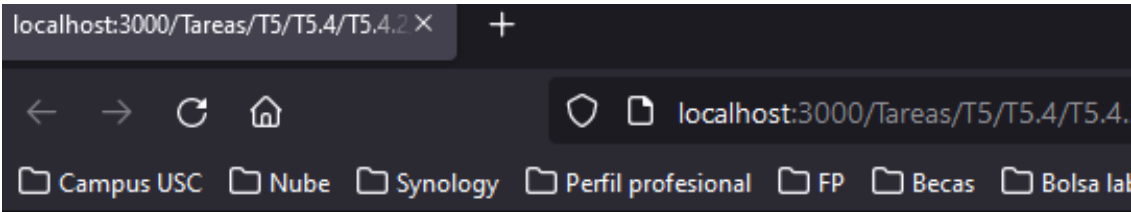
```
index.php x
1 <?php
2 //Incluyo los archivos necesarios
3 require("../modelo/model.php");
4 require("../controlador/controller.php");
5
6 //Instancio el controlador
7 $controller = new CocheController;
8
9 //Ejecuto el método
10 $controller->index();
11 ?>
```

```
controller.php x
1 <?php
2 class CocheController{
3     var $coches;
4
5     function __construct() {
6         $this->coches = [
7             1 => new Coche("Wolkswagen", "Polo", "negro", "Rebeca"),
8             2 => new Coche("Toyota", "Corolla", "verde", "Marcos"),
9             3 => new Coche("Skoda", "Octavia", "gris", "Mario"),
10            4 => new Coche("Kia", "Niro", "azul", "Jairo")
11        ];
12    }
13
14    public function index() {
15        //Asigno los coches a una variable que estard
16        // esperando la vista.
17        $rowset = $this->coches;
18        //Le paso los datos a la vista
19        require("vista/view.php");
20    }
21 }
22
```

```
model.php x
6 var $modelo;
7 var $color;
8 var $propietario;
9
10 function __construct($miMarca, $miModelo,
11     $miColor, $miPropietario) {
12     $this->marca = $miMarca;
13     $this->modelo = $miModelo;
14     $this->color = $miColor;
15     $this->propietario = $miPropietario;
16 }
17
18 //Funciones o métodos
19 function setMarca($miMarca) {
20     $this->marca = $miMarca;
21 }
22
23 function getMarca() {
24     return $this->marca;
25 }
26
27 function setModelo($miModelo) {
28     $this->modelo = $miModelo;
29 }
30
31 function getModelo() {
32     return $this->modelo;
33 }
34
35 function setColor($miColor) {
36     $this->color = $miColor;
37 }
38
39 function getColor() {
40     return $this->color;
41 }
42
43 function setPropietario($miPropietario) {
44     $this->propietario = $miPropietario;
45 }
46
47 function getPropietario() {
48     return $this->propietario;
49 }
50 }
51
52
```

```
view.php x
1 <style>
2     th {
3         width: 8rem;
4         text-align: left;
5         border-bottom: 1px solid black;
6     }
7
8     td {
9         width: 8rem;
10    }
11 </style>
12
13 <h1>Ejemplo 5: Listado de coches</h1>
14 <table>
15     <tr>
16         <th>Marca</th>
17         <th>Modelo</th>
18         <th>Color</th>
19         <th>Propietario</th>
20     </tr>
21     <?php foreach ($rowset as $row) : ?>
22
23         <tr>
24             <td><?php echo $row->marca ?></td>
25             <td><?php echo $row->modelo ?></td>
26             <td><?php echo $row->color ?></td>
27             <td><?php echo $row->propietario ?></td>
28         </tr>
29
30     <?php endforeach; ?>
31 </table>
```

Resultado de la salida por pantalla:



Ejemplo 5: Listado de coches

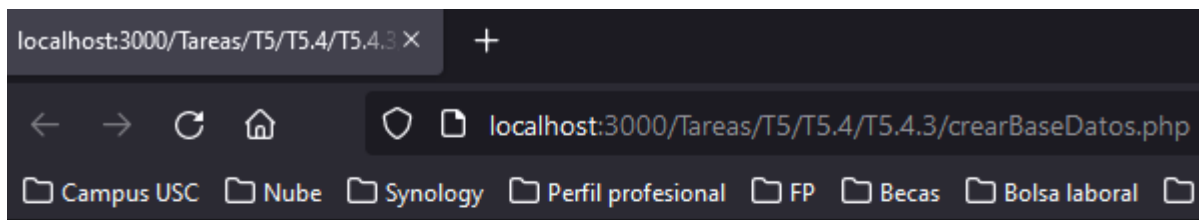
Marca	Modelo	Color	Propietario
Wolkswagen	Polo	negro	Rebeca
Toyota	Corolla	verde	Marcos
Skoda	Octavia	gris	Mario
Kia	Niro	azul	Jairo

- 5.4.3: aplicación propia con patrón MVC.

En esta ocasión crearemos un programa propio de patrón MVC aplicando uno de los métodos anteriores. En primer lugar, codificaremos un programa sencillo cuya ejecución creará una base de datos que almacenaremos en PHPMyAdmin:

```
index.php  crearBaseDatos.php X
Tareas > T5 > T5.4 > T5.4.3 > crearBaseDatos.php > ...
1  <?php
2  $servername = "localhost";
3  $username = "root";
4  $password = "";
5
6  // Crear conexión.
7  echo "<h2>Conexión a base de datos para la tarea 5.4.3</h2>";
8  $conexion = new mysqli($servername, $username, $password);
9
10 // Verificar conexión.
11 if (!$conexion) {
12     die("La conexión ha fallado.<br>" . mysqli_connect_error());
13 } else {
14     echo "Conexión establecida.<br>";
15 }
16
17 // Borrar base de datos.
18 $sql = "drop database if exists baseDatost5_4_3";
19 if (mysqli_query($conexion, $sql)) {
20     echo "<b>NOTA</b>: La base de datos se ha eliminado por defecto<br>
21     para prevenir errores de cara a una nueva creación o inserción de datos.<br>";
22 } else {
23     echo "Error al eliminar la base de datos.<br>" . mysqli_connect_error();
24 }
25
26 // Crear base de datos.
27 $sql = "create database if not exists baseDatost5_4_3";
28 if (mysqli_query($conexion, $sql)) {
29     echo "Base de datos creada correctamente.<br>";
30 } else {
31     echo "Error en la creación de la base de datos.<br>" . mysqli_connect_error();
32 }
33
34 // Selecciona la base de datos a manipular.
35 $sql = "use baseDatost5_4_3";
36 if (mysqli_query($conexion, $sql)) {
37     echo "Base de datos 'baseDatost5_4_3' seleccionada.<br>";
38 } else {
39     echo "Error al seleccionar la base de datos.<br>" . mysqli_error($conexion);
40 }
41
42 // Crear tabla en base de datos.
43 $sql = "create table if not exists usuarios (
44     dni int(8) unsigned primary key,
45     nombre varchar(100),
46     contrasena varchar(100)
47 ) engine InnoDB";
48
49 if (mysqli_query($conexion, $sql)) {
50     echo "Tabla creada correctamente.<br>";
51 } else {
52     echo "Error al crear la tabla.<br>" . mysqli_error($conexion);
53 }
54
55 // Insertar datos en la tabla.
56 $sql = "INSERT INTO `usuarios` (`dni`, `nombre`, `contrasena`)
57     VALUES ('77665588', 'Marta', 'marta1'),
58     ('11223344', 'Pedro', 'pedro1'),
59     ('99887766', 'Laura', 'laura1')";
60
61 if (mysqli_query($conexion, $sql)) {
62     echo "Datos introducidos.<br>";
63 } else {
64     echo "Error en la inserción de datos.<br>" . $sql . "<br>" . mysqli_error($conexion);
65 }
66 echo "<p><b>Operaciones realizadas satisfactoriamente. Conexión finalizada.</b></p>";
67 mysqli_close($conexion);
```

Este será el resultado de su ejecución:



Conexión a base de datos para la tarea 5.4.3

Conexión establecida.

NOTA: La base de datos se ha eliminado por defecto

para prevenir errores de cara a una nueva creación o inserción de datos.

Base de datos creada correctamente.

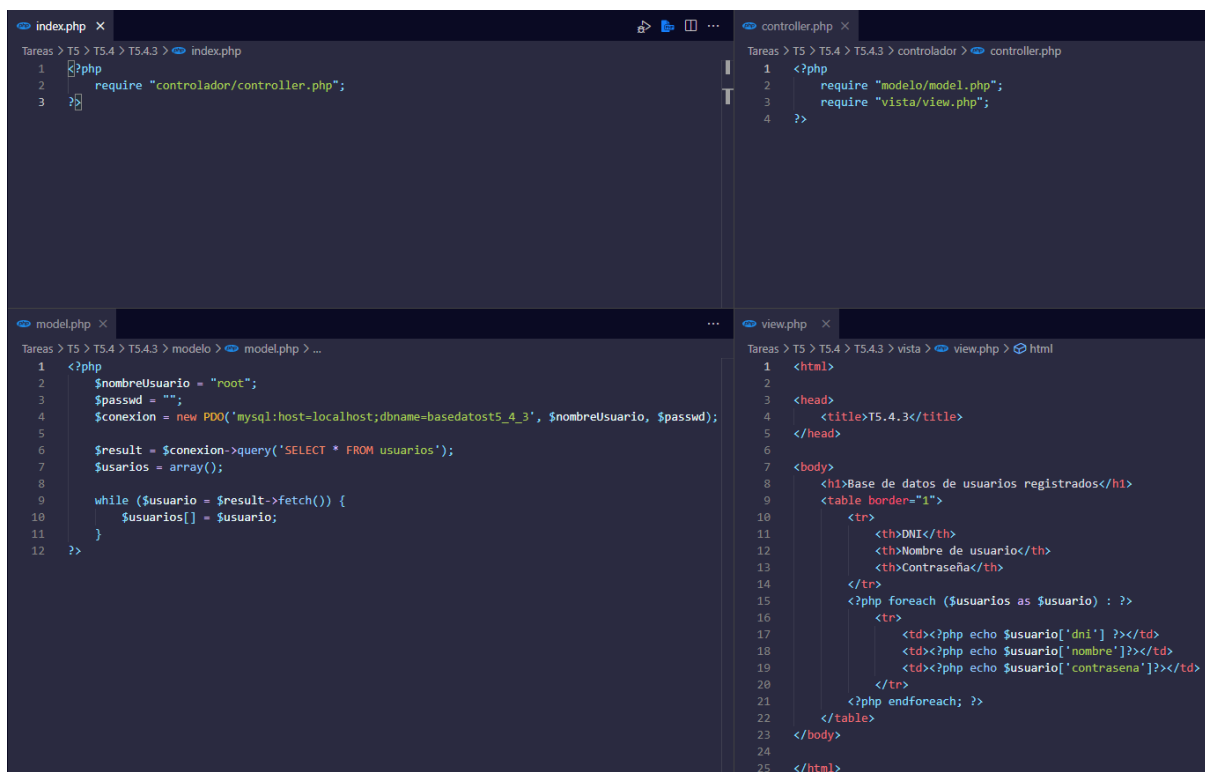
Base de datos 'baseDatosT5_4_3' seleccionada.

Tabla creada correctamente.

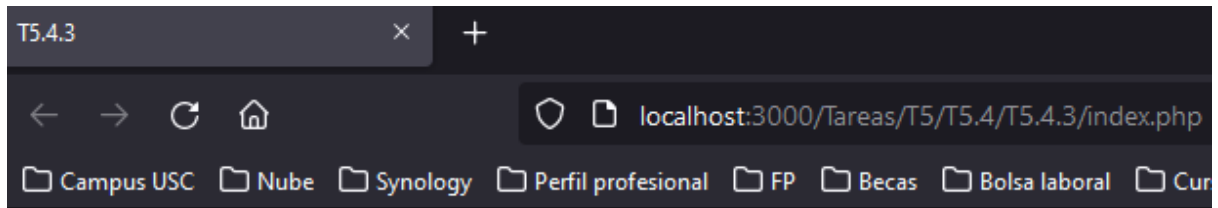
Datos introducidos.

Operaciones realizadas satisfactoriamente. Conexión finalizada.

Luego crearemos los archivos y carpetas de nuestro programa propio siguiendo la estructura MVC, cuyo objetivo será conectarse a la base de datos y realizar una consulta para mostrar los datos por pantalla al usuario:



Finalmente, esto será lo que muestre nuestro programa al usuario:



Base de datos de usuarios registrados

DNI	Nombre de usuario	Contraseña
11223344	Pedro	pedro1
77665588	Marta	marta1
99887766	Laura	laura1

BILIOGRAFÍA/WEBGRAFÍA

- Cazola, M.: *Ejemplo de cómo crear un sistema MVC paso a paso*, en gitbooks.io [recurso electrónico], consulta el 20 de enero de 2022 (https://mcazorla.gitbooks.io/programacion-en-el-servidor/content/patrones_de_diseno_en_php_ii__patron_mvc/ejemplo_de_como_cre_ar_un_sistema_mvc_paso_a_paso_1.html).
- García Rincón, J. (2021): *Patrón MVC*, en jairogarciarincon.com [recurso electrónico], consulta el 20 de enero de 2022 (<https://www.jairogarciarincon.com/clase/programacion-orientada-a-objetos-en-php/patron-mvc>).