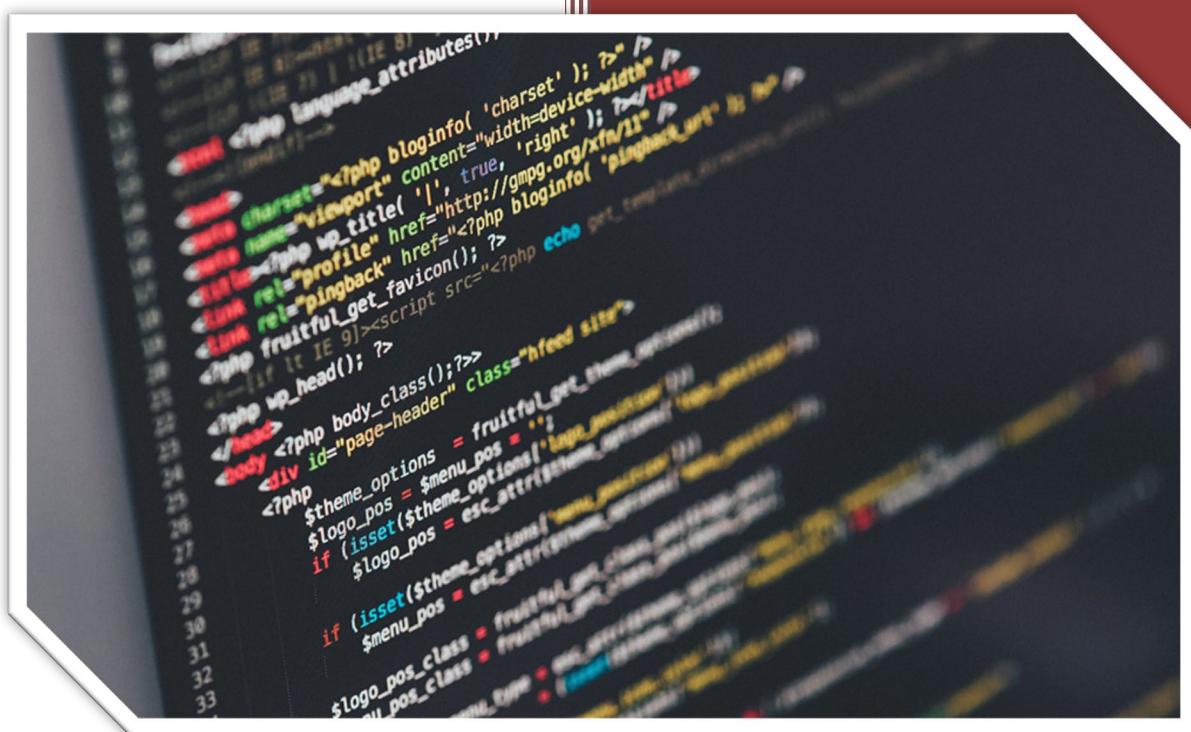


2022

Ejercicios Objetos Ud. 3



ÍNDICE

1. Ejercicio 1	1
EJERCICIOS DE FORMS, OOP	1
QUIZ DE MYSQL	1
EJERCICIOS MYSQL	2
2. Ejercicio 2	3
3. Ejercicio 3	3
4. Ejercicio 4	4
PRIMERA KATA	4
SEGUNDA KATA	5
5. PRÁCTICA BD:	5
6. PRÁCTICA OBJETOS-BD:	9
LEER ANTES	10
Global.php	10
Utils/Utils.php	10
Product.php	14
Controller/index_product.php	18
Add_or_edit_product.php	20
Details_product.php	24
View/index_product.php	26
Index.php	2431
En ejecución	26

	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
	Pág. /	



Puntuación (Los ejercicios tipo test y preguntas cortas valen la mitad que los ejercicios de programación, estos se valorarán utilizando la rúbrica de classroom). El alumno pondrá pantallazos de los ejercicios funcionando en este mismo fichero.

1. Ir a la web <https://www.w3schools.com/php/default.asp> y realizar todos los apartados de los ejercicios de FORMS, OOP , el quiz de mysql https://www.w3schools.com/mysql/mysql_quiz.asp y los ejercicios <https://www.w3schools.com/mysql/exercise.asp> , (pantallazos mostrando todas las tareas realizadas).

EJERCICIOS DE FORMS, OOP

Completed 43 of 47 Exercises:

- PHP Syntax ✓
- PHP Variables ✓
- PHP Strings ✓
- PHP Operators ✓
- PHP If...Else ✓
- PHP Switch ✓
- PHP Loops ✓
- PHP Functions ✓
- PHP Arrays ✓
- PHP Forms ✓
- PHP Dates ✓
- PHP Advanced

codecademy | Develop yours

Reloj 10:14 Hora local 02/12/2022

QUIZ DE MYSQL

Result:

21 of 25
84%
You can be proud of yourself!

10:25 Hora local
02/12/2022

EJERCICIOS MYSQL

	I.E.S. MAR DE CÁDIZ	MODELO	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo "El FSE invierte en tu futuro"
	ENSEÑANZA-APRENDIZAJE	MOD-7501209	
	PROGRAMACIÓN	Pág. /	

Completed 50 of 50 Exercises:

MySQL Select	✓
MySQL Where	✓
MySQL Order By	✓
MySQL Insert	✓
MySQL Null	✓
MySQL Update	✓
MySQL Delete	✓
MySQL Functions	✓
MySQL Like	✓
MySQL Wildcards	✓
MySQL In	✓
MySQL Between	✓
MySQL Alias	✓
MySQL Join	✓

Learn math interactively

Reloj

11:01 Hora local
02/12/2022

MySQL Between	✓
MySQL Alias	✓
MySQL Join	✓
MySQL Group By	✓
MySQL Database	✓

11:02 Hora local
02/12/2022

- Realizar en <https://www.sololearn.com/learning/1059> los ejercicios de los bloques de predefined variables y de Object Oriented Php (pantallazos).

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

Predefined Variables 7/7

Working with Files 4/4

Object-Oriented PHP 5/5

Certification
We are happy to present your certificate to you for completing this course.

Claim Certificate

11:48 Hora local 02/12/2022

3. Realizar en <https://www.codecademy.com/courses/learn-php> todos los ejercicios posibles desde php form validations hasta classes and objects (Pantallazos).

60% **PHP Form Validation**

Learn how to use PHP to validate HTML forms.

[Practice concepts](#) | [View cheatsheet](#)

Lesson	Introduction to Form Validation
Lesson	Introduction to Regular Expressions
Lesson	Introduction to PHP Form Validation
Quiz	PHP Form Validation
Project	Bobby Tables

Reset Progress

14:39 Hora local 04/12/2022

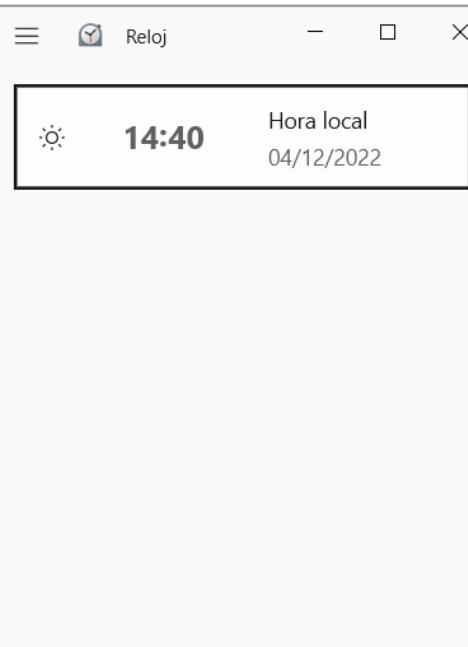
	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		



Classes and Objects in PHP

Learn how to use object-oriented programming in PHP.

[Practice concepts](#) | [View cheatsheet](#)



- Video Classes and Objects
- Lesson Classes and Objects
- Quiz Classes and Objects
- Project Going to Bed

[Reset Progress](#)

4. Realizar en <https://www.codewars.com/dashboard> 2 katas de nivel 7 o superior en php (pantallazos de vuestro perfil donde se vean dichos katas realizados).

PRIMERA KATA

<p>Completed (4)</p> <p>Unfinished</p> <p>Obsolete</p>	<p> Highest and Lowest</p> <p>PHP:</p> <pre>function highAndLow(\$numbers) { \$numbers = explode(" ", \$numbers); return max(\$numbers) . " " . min(\$numbers); } echo highAndLow("8 3 -5 42 -1 0 0 -9 4 7 4 -4");</pre> <p>1 minute ago • Refactor • Discuss</p> <p> Jaden Casing Strings</p> <p>PHP:</p>	
--	---	--

SEGUNDA KATA

	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
	Pág. /	

AyraNixx/completed_solutions

Completed (5) Unfinished Obsolete	 Detect Pangram PHP: <pre>function detect_pangram(\$str) { #your code here //Convertimos la frase, pasada a minúscula, a array //para sacar cada letra \$str = str_split(strtolower(\$str)); //Creamos un array con las letras del alfabeto \$alphabet = range("a", "z"); //Con array_intersect obtenemos un array que contiene todos los valores //que estén presentes en todos los argumentos que se le pasa. //Por lo tanto, si pasamos nuestro array \$alphabet y la cadena pasada a //array, nos devolverá un array que solo tendrá aquellos valores que //coincidan en ambos. \$array_obtain = array_intersect(\$alphabet, \$str); //Ahora para comprobar que tiene todas las letras del alfabeto, //vemos si es igual a 26, que es el numero de letras que hay en //el alfabeto. Para eso, usamos count. if(count(\$array_obtain) === 26) { return true; }else{ return false; } }</pre>	 Reloj  15:33 Hora local 04/12/2022
--	---	---

<pre>if(count(\$array_obtain) === 26) { return true; }else{ return false; } if(detect_pangram("1L%r+f4G!e7w V z q6M h4d F3b+t 02n e K^g+c#S^i4i X7c-u P5d7j Y6a(a B")) { echo "It's a pangram."; }else{ echo "It is not a pangram."; }</pre>	 15:35 Hora local 04/12/2022
---	--

3 minutes ago · Refactor · Discuss

5. PRÁCTICA BD:

Realizar un diseño de una Base de datos de las siguientes características:

	I.E.S. MAR DE CÁDIZ	MODELO	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo  Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

- Al menos tendrá 6 entidades, las cuales tendrán una clave primaria auto-incremental y al menos 4 campos excepto casos excepcionales.
- Deberá haber al menos dos relaciones de tipo n:m y otras dos 1:n.
- La temática será única para cada alumno. Tendrá que existir una entidad especial de la cual exista relaciones con otras entidades. La entidad principal tendrá un campo imagen(texto) que será el nombre de la imagen. (**SUBIR NOTA**: una entidad puede tener varias imágenes).

La base de datos realizada cuenta con las siguientes entidades:

- **Usuarios.** Posiblemente sea modificada durante el transcurso de la práctica de la Ud. 5. Consta de los siguientes atributos:
 - **Id_producto.** INT, AI.
 - **Nombre.** VARCHAR(100).
 - **Apellidos.** VARCHAR(200).
 - **Direccion.** VARCHAR(250).
 - **Telefono.** VARCHAR(11).
 - **Correo.** VARCHAR(100).
 - **Passwd.** VARCHAR(400).
 - **Salt.** VARCHAR(16).
 - **Foto.** VARCHAR(4000).
 - **Activo.** TINYINT.
 - **Cod_activacion.** INT(5).
 - **Id_rol.** INT.
- **Roles.** Formada por 3 atributos al no ser necesario añadir un cuarto: **id_rol** – INT, AI –, **rol** – VARCHAR(100) –, **descripción** – VARCHAR(300) –.
- **Categoría.** Formada por 3 atributos al no ser necesario añadir un cuarto: **id_categoria** – INT, AI –, **nombre** – VARCHAR(100) –, **descripción** – VARCHAR(300) –.
- **Pedidos.** Compuesta por 4 atributos: **id_pedido** – INT, AI –, **fecha_compra** – DATE –, **precio_final** – DECIMAL –, **id_usuario** – INT –.
- **Productos.** Atributos:

	I.E.S. MAR DE CÁDIZ	MODELO	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

- **Id_producto.** INT, AI.
- **Nombre.** VARCHAR(100).
- **Precio.** DECIMAL(10,2).
- **Stock.** INT.
- **Foto.** VARCHAR(4000).
- **Categoría.** INT.

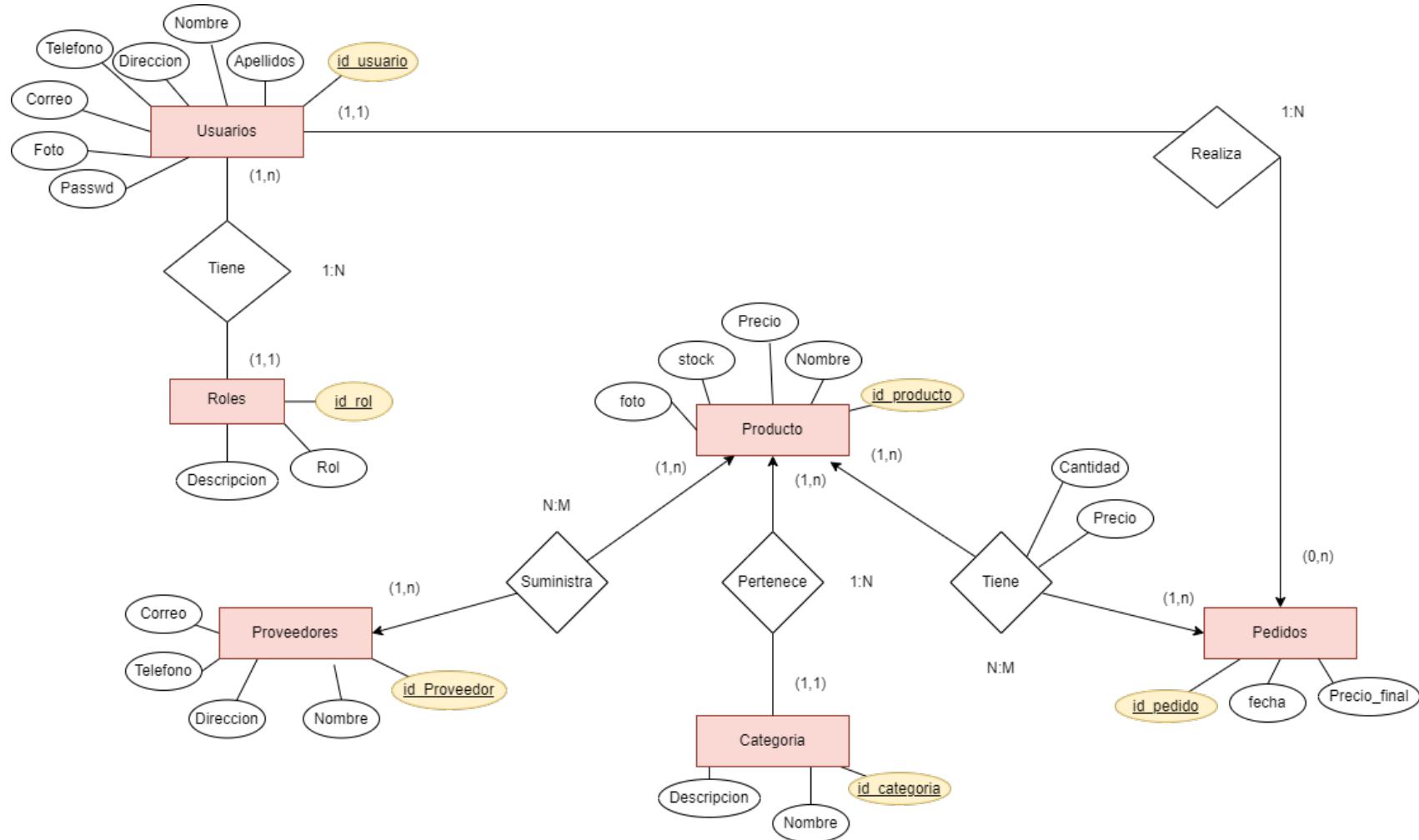
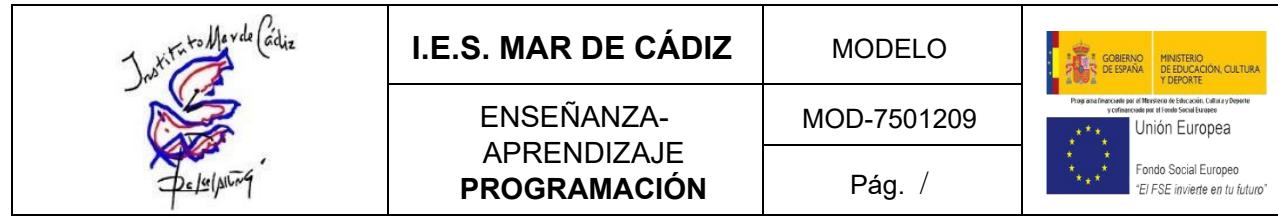
— **Proveedores.** Atributos:

- **Id_proveedor.** INT, AI.
- **Nombre.** VARCHAR(100).
- **Direccion.** VARCHAR(250).
- **Telefono.** VARCHAR(11).
- **Correo.** VARCHAR(100).

Las relaciones que han surgido son las siguientes:

- **Productos - Proveedores.** N:M. Se crea una nueva tabla, **Detalles_proveedores**.
- **Productos - Categoría.** 1:N.
- **Productos - Pedidos.** N:M. Se crea una nueva tabla, **Detalles_pedido**.
- **Pedidos - Usuarios.** 1:N.
- **Usuarios - Roles.** 1:N.

A continuación, fotos del modelo entidad-relación:



	I.E.S. MAR DE CÁDIZ	MODELO	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo  Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

6. PRÁCTICA OBJETOS-BD:

Realizar una web que separe el código utilizando el modelo vista controlador. Si el código no está separado correctamente se tendrá que repetir. Desde la vista jamás se accederán a las funciones del modelo directamente, sino al controlador. Se deberá seguir la estructura de acceso a BD vista o una similar.

La web deberá cumplir los siguientes requisitos:

- Realizar una web que separe el código utilizando el modelo vista controlador. Si el código no está separado correctamente se tendrá que repetir. Desde la vista jamás se accederán a las funciones del modelo directamente, sino al controlador. Se deberá seguir la estructura de acceso a BD vista o una similar.
- La web deberá cumplir los siguientes requisitos:
- Jamás se utilizarán variables directamente en las query, siempre se utilizará preparedStatements.
- Los datos de conexión estarán separados del código php
- Se crearán librerías para la conexión y manipulación de al menos 4 entidades de la BD. (Operaciones de borrado, inserción, modificación y consulta Paginada)
- La estética de la web utilizará elementos de Bootstrap
- Los datos del formulario se validarán que son del tipo y rango correspondiente y se les eliminarán todos los caracteres que puedan permitir inyección de código(mirar https://www.w3schools.com/php/php_form_validation.asp)
- Componentes:
 - ▽ **Página Principal:** Mostrará un listado en tabla con los elementos de la entidad principal con paginación de forma adecuada. Cada elemento tendrá en su fila un botón para borrarlo y otro para modificarlo(deben de funcionar), El nombre será un enlace que abrirá un detalle del elemento al pulsarlo, en la misma página con ajax (subir nota) o en otra página distinta.. En la parte superior derecha mostrará un botón de "añadir ...".
 - ▽ **Página Detalle:** Mostrará campos que no se muestran en la página principal, su foto y un listado con los datos de una de las entidades relacionadas. (por ejemplo, un coche al darle a detalle, un listado de concesionarios donde está disponible)
 - ▽ **Página Modificación:** Al pulsar el botón modificación se mostrará un formulario dónde se cargarán los datos de la entidad correspondiente y se podrán modificar.
 - ▽ **Página Inserción:** Igual que la anterior, sólo que aparecerán los campos vacíos y en lugar de modificar un registro en BD, se insertará uno nuevo.

	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
	Pág. /	

LEER ANTES

Las entidades seleccionadas para la realización de esta práctica son: Roles, Categoría, Productos y Proveedores.

He intentado realizar un único controlador para cada entidad para intentar reducir el número de archivos, por lo que sé que el código será un caos.

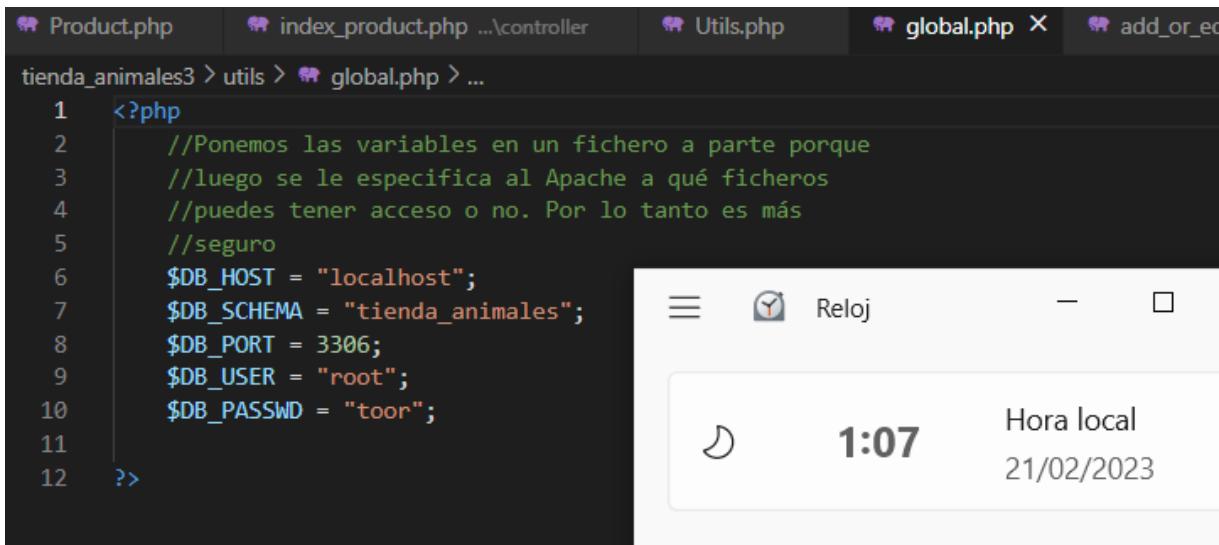
Mi página de Modificación y de Inserción es la misma, dependiendo de la opción elegida se hará una cosa u otra.

Al ser muchos archivos, explicaré la entidad que cuente con más detalles, como, por ejemplo, Productos en la que se sube la url de una imagen a la base de datos.

Empezaremos con global.php.

Global.php

En este archivo guardaremos el nombre del host, el esquema, puerto, usuario y contraseña necesarios para el acceso a la base de datos.



```

Product.php index_product.php ...\\controller Utils.php global.php X add_or_ed
tienda_animales3 > utils > global.php > ...
1 <?php
2 //Ponemos las variables en un fichero a parte porque
3 //luego se le especifica al Apache a qué ficheros
4 //puedes tener acceso o no. Por lo tanto es más
5 //seguro
6 $DB_HOST = "localhost";
7 $DB_SCHEMA = "tienda_animales";
8 $DB_PORT = 3306;
9 $DB_USER = "root";
10 $DB_PASSWD = "toor";
11
12 ?>

```

The screenshot shows a code editor window with several tabs at the top: Product.php, index_product.php, controller, Utils.php, global.php (which is the active tab), and add_or_ed. Below the tabs, the file structure is shown as tienda_animales3 > utils > global.php > The main area displays the contents of the global.php file. The code defines constants for database connection parameters: \$DB_HOST, \$DB_SCHEMA, \$DB_PORT, \$DB_USER, and \$DB_PASSWD, all set to their respective values. The code editor interface includes a toolbar with icons for file operations, a status bar showing the time (1:07) and date (21/02/2023), and a small calendar icon.

Utils/Utils.php

En este fichero, crearemos una clase denominada Utils en la que añadiremos funciones que nos puedan ser útiles en el futuro, por ejemplo, generar un SALT o guardar una imagen en un fichero.

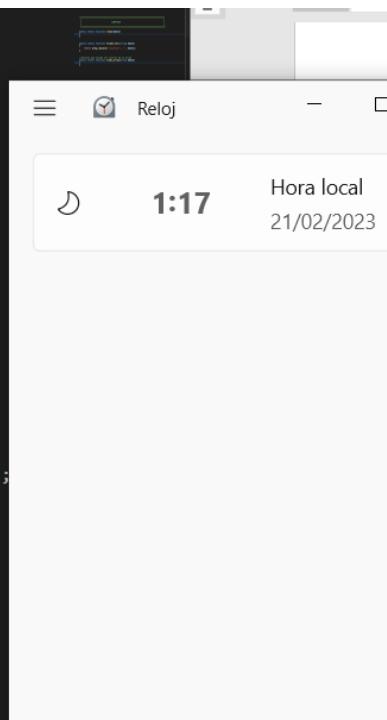
El primer método que crearemos es **conectar()** que será público y estático para que se pueda acceder a él sin tener que declarar un objeto de la clase Utils.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

```

20  /*
21   * public static function conectar()
22  {
23     //Utilizamos require_once para importar las constantes de global.php
24     //Utilizamos require porque estas constantes son necesarias para que vaya
25     //la aplicación, ya que sin ellas no podríamos establecer conexión con la
26     //base de datos
27     require_once("global.php");
28
29     //Definimos conBD
30     $conBD = null;
31
32     //Englobamos el código en un try/catch para que en caso de error de conexión
33     //se lance un objeto PDOException
34     try {
35       //Para establecer una conexión, creamos una instancia de la clase PDO.
36       //Se le pasarán como argumentos el host, el nombre de la base de datos,
37       //el usuario y su contraseña en caso de que tuviese una.
38       //En mi caso, no he especificado contraseña porque no tengo.
39       return $conBD = new PDO("mysql:host=$DB_HOST;dbname=$DB_SCHEMA", $DB_USER);
40     } catch (PDOException $e) {
41       print("¡Error! : " . $e->getMessage() . "<br/>");
42       //Devolvemos $conBD, que será null si no se pudo realizar la conexión
43       //correctamente.
44       return $conBD;
45       //Alias de exit(), finaliza el script.
46       //DUDA?????????
47       die();
48     }
49   }

```

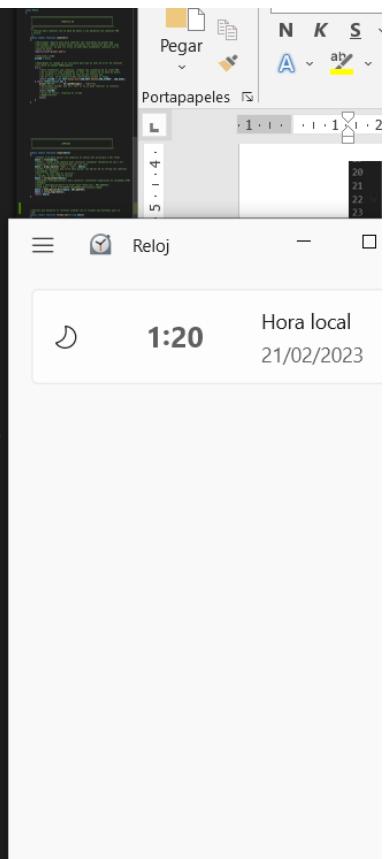


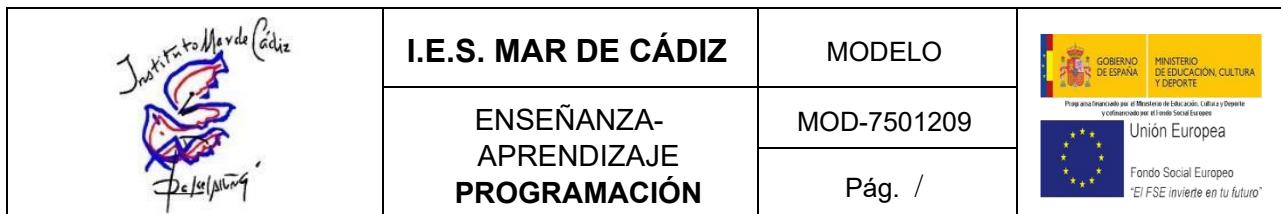
Seguidamente, crearemos tres métodos para la limpieza y validación de datos:

```

65  ****
66  *
67  *          LIMPIEZA
68  *
69  ****
70
71  public static function clean($data)
72  {
73    //Usamos trim para quitar los espacios en blanco del principio y del final
74    $data = trim($data);
75    //Además, usamos preg_replace para sustituir cualquier secuencia de uno o más
76    //espacios en blanco consecutivos por uno solo
77    $data = preg_replace('/\s+/', '/\s/', $data);
78    //Usamos stripslashes que sirve para quitar las barras de un string con comillas
79    //escapadas. Ejemplo:
80    //$str = "Is your name O'reilly?";
81    // Salida: Is your name O'reilly?
82    $data = stripslashes($data);
83    //Empleamos htmlspecialchars para convertir caracteres especiales en entidades HTML
84    //Ejemplo:
85    //$new = htmlspecialchars("<a href='test'>Test</a>", ENT_QUOTES);
86    //echo $new; &lt;a href=&#039;test&#039;&gt;Test&lt;/a&gt;
87    $data = htmlspecialchars($data, ENT_QUOTES);
88    $data = strip_tags($data);
89    return $data;
90  }
91
92
93
94
95
96  //Función que devuelve el teléfono añadido con el formato que queremos para la
97  //BD
98  public static function format_tel(string $data)
99  {
100    return preg_replace('/([\s-])/', '', $data);
101  }

```





(La función clean_array se creó para limpiar y validar los datos pasados por un array y así evitar tener que hacerlos uno a uno)

```
//Funcion que valida los valores de un array (y así no tener que hacerlo uno a uno)
public static function clean_array(array $data)
{
    $alert = "";
    $success = true;
    //Recorremos el array
    foreach ($data as $key => $value) {
        //Si la llave contiene id, sea categoria o stock
        if (strpos($key, "id") != false || $key == "categoria" || $key == "stock" || $key == "option") {
            // Validamos que sea un entero positivo
            if (is_numeric($value) && $value >= 0) {
                $data[$key] = filter_var(self::clean($value), FILTER_VALIDATE_INT);
            } else {
                $success = false;
                $alert = "El valor de la llave \"\$key\" debe ser un número entero positivo.";
            }
        } //En caso de que la llave coincida con 'precio'
        elseif ($key == "precio") {

            // Validamos que sea un float positivo
            if (is_numeric($value) && $value >= 0) {
                $data[$key] = filter_var(self::clean($value), FILTER_VALIDATE_FLOAT);

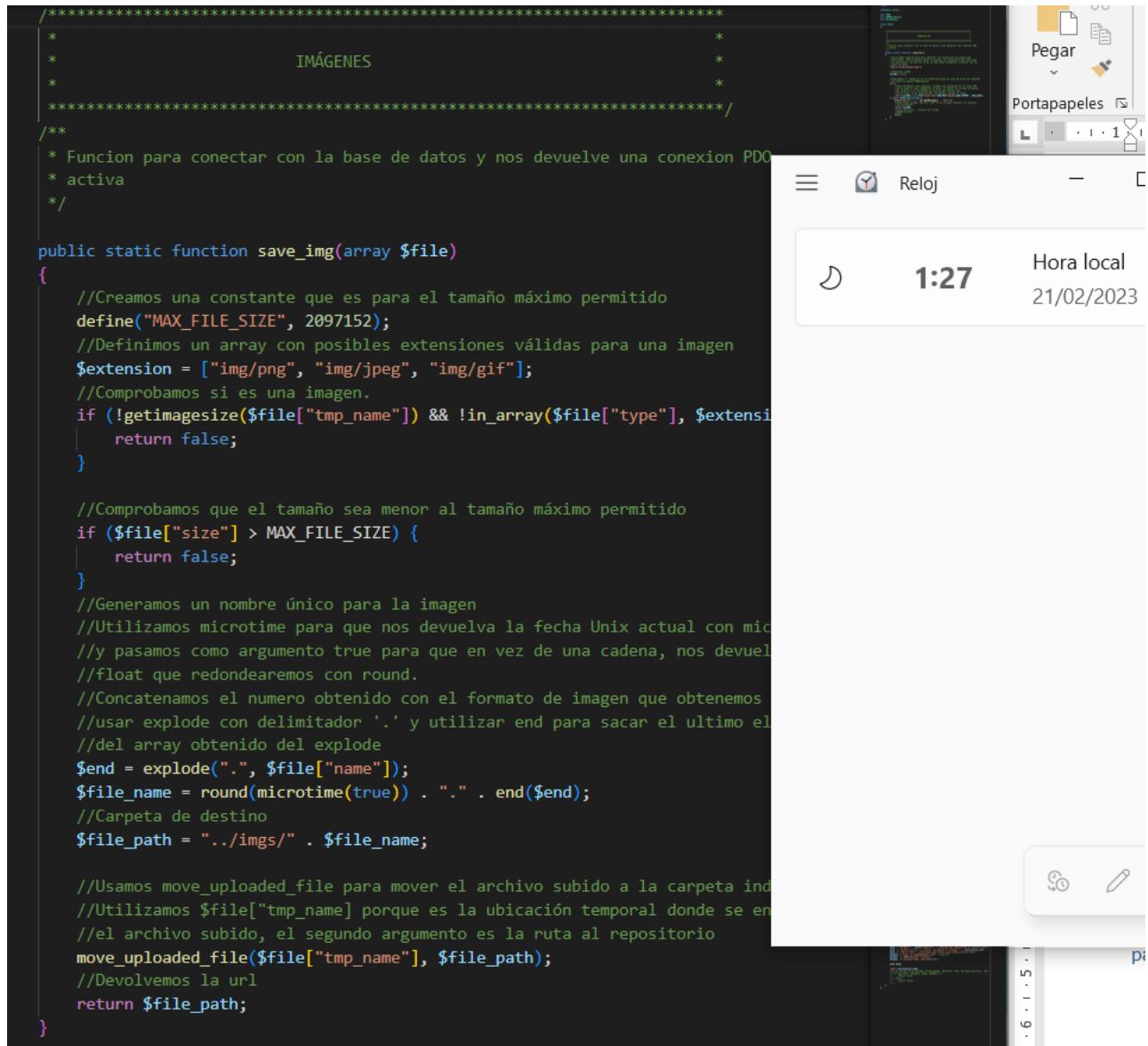
            } else {
                $success = false;
                $alert = "El valor de la llave \"\$key\" debe ser un número decimal positivo.";
            }
        }
        //Si la llave corresponde con correo
        if ($key == "correo") {
            //Validamos el correo
            $data[$key] = filter_var(self::clean($value), FILTER_VALIDATE_EMAIL);

            if ($data[$key] == false) {
                $success = false;
                $alert = "El valor de la llave \"\$key\" debe ser un correo válido.";
            }
        } else {
            //Si no es ninguna de las claves anteriores, limpiamos la cadena
            $data[$key] = self::clean($value);
        }
    }
    //Si la llave corresponde con telefono
}
```

```
149
150         //Si la llave corresponde con telefono
151         if($key == "telefono")
152         {
153             $data[$key] = self::clean(self::format_tel($value));
154         }
155     }
156
157     try {
158         //Si success es false
159         if (!$success) {
160             //Lanzamos una nueva excepcion
161             throw new Exception($alert);
162         }
163     } catch (Exception $e) {
164         //Y mandamos el error a nuestro log
165         self::save_log($e->getMessage());
166         //Devolvemos false
167         return false;
168     }
169     //Si todo esta correcto, devolvemos data validada
170     return $data;
171 }
172 }
```

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

Ahora, vienen dos funciones una que devolverá la ruta de la imagen que se quiere subir y otra para eliminar la imagen indicada.



```

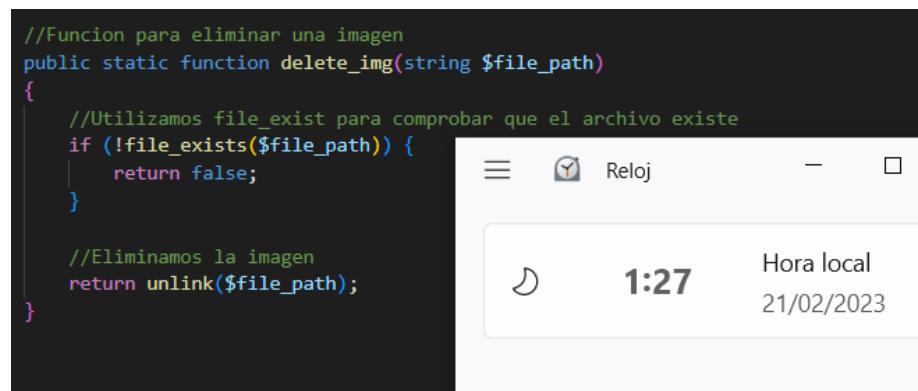
/*
 *          IMÁGENES
 */
/**
 * Funcion para conectar con la base de datos y nos devuelve una conexion PDO
 * activa
 */

public static function save_img(array $file)
{
    //Creamos una constante que es para el tamaño máximo permitido
    define("MAX_FILE_SIZE", 2097152);
    //Definimos un array con posibles extensiones válidas para una imagen
    $extension = ["img/png", "img/jpeg", "img/gif"];
    //Comprobamos si es una imagen.
    if (!getimagesize($file["tmp_name"]) && !in_array($file["type"], $extension))
        return false;
}

//Comprobamos que el tamaño sea menor al tamaño máximo permitido
if ($file["size"] > MAX_FILE_SIZE) {
    return false;
}
//Generamos un nombre único para la imagen
//Utilizamos microtime para que nos devuelva la fecha Unix actual con microsegundos
//y pasamos como argumento true para que en vez de una cadena, nos devuelva un float
//float que redondearemos con round.
//Concatenamos el numero obtenido con el formato de imagen que obtenemos
//usar explode con delimitador '.' y utilizar end para sacar el ultimo elemento
//del array obtenido del explode
$end = explode(".", $file["name"]);
$file_name = round(microtime(true)) . "." . end($end);
//Carpeta de destino
$file_path = "../imgs/" . $file_name;

//Usamos move_uploaded_file para mover el archivo subido a la carpeta indicada
//Utilizamos $file["tmp_name"] porque es la ubicación temporal donde se encuentra el archivo subido, el segundo argumento es la ruta al repositorio
move_uploaded_file($file["tmp_name"], $file_path);
//Devolvemos la url
return $file_path;
}

```



```

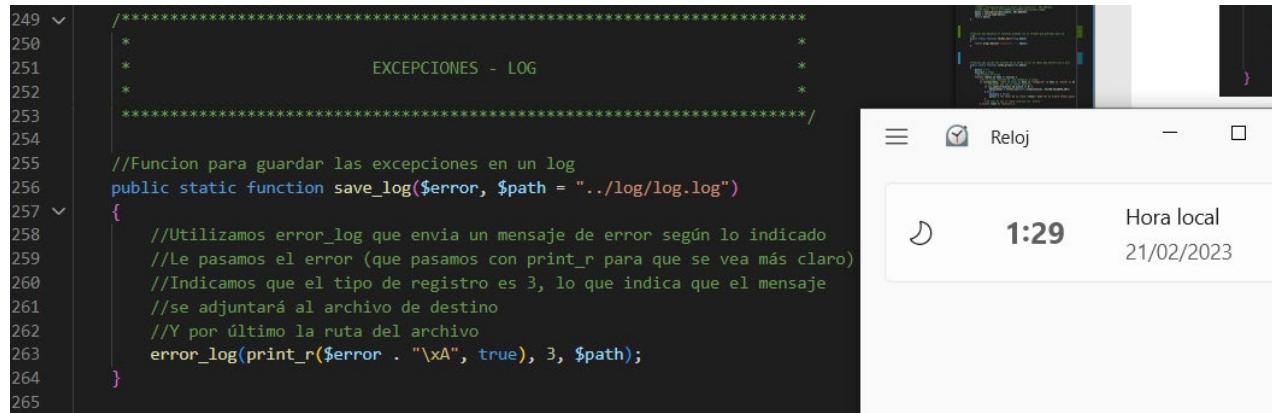
//Funcion para eliminar una imagen
public static function delete_img(string $file_path)
{
    //Utilizamos file_exists para comprobar que el archivo existe
    if (!file_exists($file_path)) {
        return false;
    }

    //Eliminamos la imagen
    return unlink($file_path);
}

```

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
---	---	--	--

Por último, tenemos una función que guardará los errores generados en un log.



```

249 /**
250 *      ****
251 *          *
252 *          EXCEPCIONES - LOG
253 *          *
254 *      ****
255
256 //Funcion para guardar las excepciones en un log
257 public static function save_log($error, $path = "../log/log.log")
258 {
259     //Utilizamos error_log que envia un mensaje de error según lo indicado
260     //Le pasamos el error (que pasamos con print_r para que se vea más claro)
261     //Indicamos que el tipo de registro es 3, lo que indica que el mensaje
262     //se adjuntará al archivo de destino
263     //Y por último la ruta del archivo
264     error_log(print_r($error . "\xA", true), 3, $path);
265 }

```

Más adelante, con la práctica de la Ud. 5 se añadirán más métodos.

Product.php

En este archivo crearemos una clase llamada Product (se hará lo mismo para las otras 3 entidades), que tendrá un atributo privado llamado \$conBD para la conexión con la base de datos, que se inicializará en el constructor.

En esta clase crearemos varias funciones:

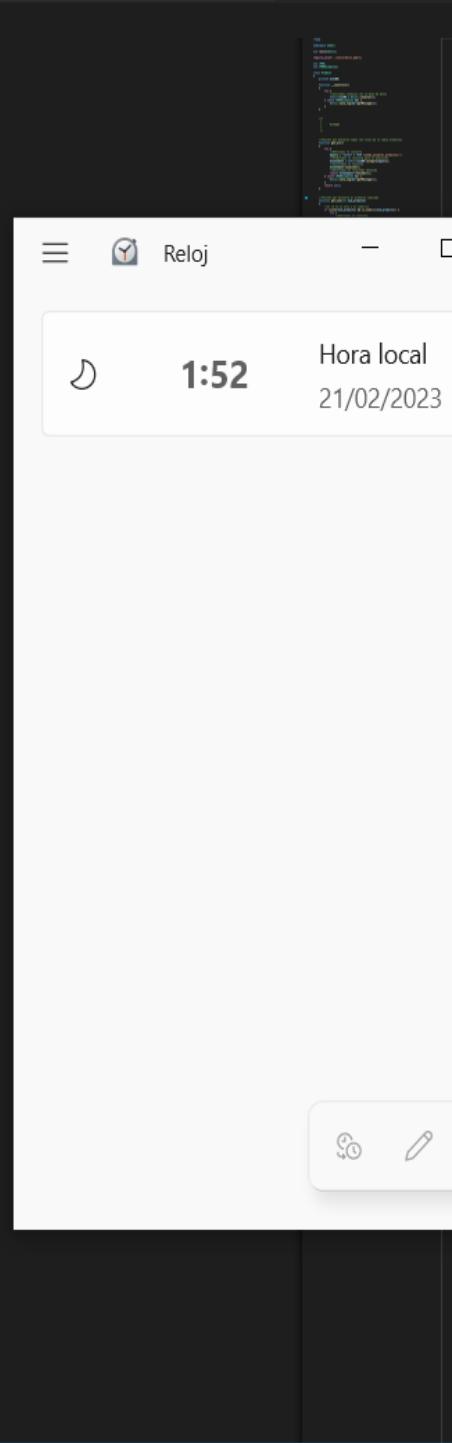
- **Get_all.** Devuelve todas las filas de la tabla productos.
- **Get_one.** Devuelve el product indicado.
- **Add.** Añade un nuevo producto.
- **Upadete.** Modifica un producto.
- **Delete.** Elimina un producto.
- **Paginación.** Devuelve los datos paginados.
- **Get_total_pages.** Devuelve el número de páginas que hay según la cantidad de elementos que se muestran en ella.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

```

tienda_animales3 > model > 🏷 Product.php > 📁 Product > ⚙ pagination
1  <?php
2
3  namespace model;
4
5  use \utils\Utils;
6
7  require_once("../utils/Utils.php");
8
9  use \PDO;
10 use \PDOException;
11
12 class Product
13 {
14     private $conBD;
15
16     function __construct()
17     {
18         try {
19             //Iniciamos conexión con la base de datos
20             $this->conBD = Utils::conectar();
21         } catch (PDOException $e) {
22             Utils::save_log($e->getMessage());
23         }
24     }
25
26
27     /**
28      *
29      *      MÉTODOS
30      *
31     */
32
33
34     //Función que devuelve todas las filas de la tabla productos
35     function get_all()
36     {
37         try {
38             //Definimos la consulta
39             $query = "SELECT * FROM tienda_animales.productos;";
40             //Preparamos la consulta para su ejecución
41             $statement = $this->conBD->prepare($query);
42             //Ejecutamos la consulta
43             $statement->execute();
44             //Devolvemos el resultado obtenido
45             return $statement->fetchAll();
46         } catch (PDOException $e) {
47             Utils::save_log($e->getMessage());
        }
    }
}

```





I.E.S. MAR DE CÁDIZ

MODELO

ENSEÑANZA- APRENDIZAJE **PROGRAMACIÓN**

MOD-7501209

Pág. /



```
Product.php M x index_product.php ... \controller add_or_edit_product.php details_product.php D
tienda_animales3 > model > Product.php > Product > pagination
44     //Devolvemos el resultado obtenido
45     return $statement->fetchAll();
46 } catch (PDOException $e) {
47     Utils::save_log($e->getMessage());
48 }
49     return null;
50 }
51
52
53 //Función que devuelve el producto indicado
54 function get_one(int $id_producto)
55 {
56     //Si id no es nulo y es numerico
57     if (isset($id_producto) && is_numeric($id_producto)) {
58         try {
59             //Definimos la consulta
60             $query = "SELECT * FROM tienda_animales.productos WHERE id_producto = :id_p
61             //Preparamos la sentencia
62             $statement = $this->conBD->prepare($query);
63             //Vinculamos los parametros al nombre de la variable especificada
64             $statement->bindParam(":id_producto", $id_producto, PDO::PARAM_INT);
65             //Ejecutamos la consulta
66             $statement->execute();
67             //Devolvemos el resultado obtenido
68             return $statement->fetch();
69         } catch (PDOException $e) {
70             Utils::save_log($e->getMessage());
71         }
72         return null;
73     }
74 }
75
76 //Funcion que añade un nuevo producto
77 function add(array $product)
78 {
79     //Si $product no es nulo, así como ninguno de sus valores
80     if (isset($product) && isset($product["nombre"]) && isset($product["precio"]) && is_
81     try {
82         //Definimos la consulta
83         $query = "INSERT INTO tienda_animales.productos (nombre, precio, stock, cate
84         //Preparamos la sentencia
85         $statement = $this->conBD->prepare($query);
86         //Vinculamos los parametros al nombre de variable especificado
87         $statement->bindParam(":nombre", $product["nombre"], PDO::PARAM_STR);
88         $statement->bindParam(":precio", $product["precio"], PDO::PARAM_INT);
89         $statement->bindParam(":stock", $product["stock"], PDO::PARAM_INT);
90         $statement->bindParam(":categoria", $product["categoria"], PDO::PARAM_TINY
```

Página 17 de 18 1253 palabra

```
if (isset($product) && isset($product["nombre"]) && isset($product["precio"]) && is
81    try {
82        //Definimos la consulta
83        $query = "INSERT INTO tienda_animales.productos (nombre, precio, stock, cat
84        //Preparamos la sentencia
85        $statement = $this->conBD->prepare($query);
86        //Vinculamos los parametros al nombre de variable especificado
87        $statement->bindParam(":nombre", $product["nombre"], PDO::PARAM_STR);
88        $statement->bindParam(":precio", $product["precio"], PDO::PARAM_INT);
89        $statement->bindParam(".stock", $product["stock"], PDO::PARAM_INT);
90        $statement->bindParam(":categoria", $product["categoria"], PDO::PARAM_INT);
91        $statement->bindParam(":img", $product["img"], PDO::PARAM_STR);
92        //Devolvemos el resultado de la ejecucion (será un boolean true si todo ha
93        return $statement->execute();
94    } catch (PDOException $e) {
95        Utils::save_log($e->getMessage());
96    }
97
98    return null;
99}
```

1:55 Hora local
21/02/2023

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
---	---	--	---

```

102 //Funcion que actualiza un nombre
103 function update(array $product)
104 {
105     //Comprobamos que no sea nulo
106     if (isset($product["id_producto"]) && isset($product["new_id"]) && isset($product["nombre"]))
107     {
108         try {
109             //Definimos la consulta
110             $query = "UPDATE tienda_animales.productos SET id_producto=:new_id, nombre=:nombre WHERE id_producto=:id_producto";
111             //Preparamos la sentencia
112             $statement = $this->conBD->prepare($query);
113             //Vinculamos los parametros al nombre de variable especificado
114             $statement->bindParam(":id_producto", $product["id_producto"], PDO::PARAM_INT);
115             $statement->bindParam(":new_id", $product["new_id"], PDO::PARAM_INT);
116             $statement->bindParam(":nombre", $product["nombre"], PDO::PARAM_STR);
117             $statement->bindParam(":precio", $product["precio"], PDO::PARAM_STR); //Utilizamos PDO::PARAM_STR para precios
118             $statement->bindParam(":stock", $product["stock"], PDO::PARAM_INT);
119             $statement->bindParam(":categoria", $product["categoria"], PDO::PARAM_INT);
120             $statement->bindParam(":img", $product["img"], PDO::PARAM_STR);
121             //Devolvemos el resultado de la ejecucion (será un boolean true si todo ha ido bien)
122             $result = $statement->execute();
123             return $result;
124         } catch (PDOException $e) {
125             Utils::save_log($e->getMessage());
126             return null;
127         }
128     }
129 }
130
131
132
133 //Funcion que elimina el nombre indicado
134 function delete(int $id_producto)
135 {
136     //Si id no es nulo y es numerico
137     if (isset($id_producto) && is_numeric($id_producto)) {
138         try {
139             //Definimos la consulta
140             $query = "DELETE FROM tienda_animales.productos WHERE id_producto=:id_producto";
141             //Preparamos la sentencia
142             $statement = $this->conBD->prepare($query);
143             //Vinculamos los parametros al nombre de variable especificado
144             $statement->bindParam(":id_producto", $id_producto, PDO::PARAM_INT);
145             //Devolvemos el resultado de la ejecucion (será un boolean true si todo ha ido bien)
146             $result = $statement->execute();
147             Utils::save_log($e->getMessage());
148             return $result;
149         } catch (PDOException $e) {
150             Utils::save_log($e->getMessage());
151             return null;
152         }
153     }
154
155     //Funcion que devuelve la tabla paginada
156     function pagination(string $ordAsc, string $field, int $numPag, int $amount)
157     {
158         try {
159             //Calculamos desde que linea se empieza
160             $offset = ($numPag - 1) * $amount;
161
162             $query = "SELECT * FROM tienda_animales.productos ORDER BY $field $ordAsc LIMIT $offset, $amount";
163
164             //Preparamos la consulta
165             $statement = $this->conBD->prepare($query);
166             //Por alguna razón que desconozco con mis escasos conocimientos, el bindparam no funciona bien
167             // Vinculamos los parámetros al nombre de la variable especificada
168             // Utilizamos PDO::PARAM_INT para indicar que se trata de un número entero
169             // ya que en ocasiones puede contarlo como una cadena.

```

Reloj 2:02 Hora local 21/02/2023

Página 19 de 19 1253 palabras

2:03 Hora local 21/02/2023

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 GOBIERNO DE ESPAÑA  MINISTERIO DE EDUCACIÓN, CULTURA Y DEPORTE <small>Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo</small>  Unión Europea  Fondo Social Europeo <small>'El FSE invierte en tu futuro'</small>
---	---	--	---

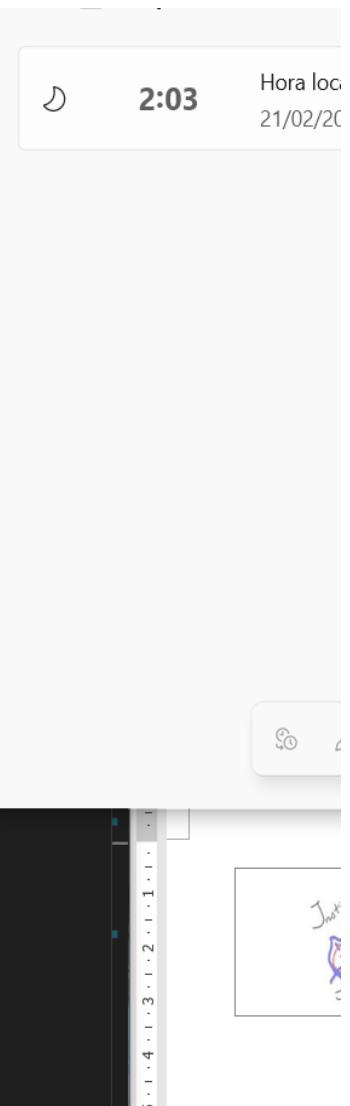
```

168     // Utilizamos PDO::PARAM_INT para indicar que se trata de un número entero
169     // ya que en ocasiones puede contarlo como una cadena.
170     $statement->bindValue(":amount", $amount, PDO::PARAM_INT);
171     $statement->bindValue(":offset", $offset, PDO::PARAM_INT);

172
173     //No quito lo de debugDumpParams porque es algo interesante
174     // echo "<pre>";
175     // var_dump($statement->debugDumpParams());
176     // echo "</pre>";

177
178     $statement->execute();
179     //Devolvemos las filas resultantes
180     return $statement->fetchAll();
181 } catch (PDOException $e) {
182     Utils::save_log($e->getMessage());
183 }
184 return null;
185
186
187
188 //Funcion que devuelve el total de páginas que tendremos en función de la cantidad
189 //de elementos que queramos mostrar en cada una de llas
190 function get_total_pages(int $amount)
191 {
192     try {
193         //Definimos la consulta (usamos count para que cuente el total de filas de la tabla
194         //lo dividimos entre la cantidad de elementos que queremos por cada página y usamos
195         //round para que nos redondee el resultado obtenido)
196         $query = "SELECT CEILING(COUNT(*)/:amount) AS pages FROM tienda_animales.productos";
197         //Preparamos la sentencia
198         $statement = $this->conBD->prepare($query);
199         //Vinculamos los parámetros al nombre de la variable especificada
200         $statement->bindParam(":amount", $amount, PDO::PARAM_INT);
201         //Ejecutamos la consulta
202         $statement->execute();
203         //Devolvemos el total de páginas (como la función fetch nos devuelve un array
204         //asociativo, pongo pages para que me devuelva el valor de la clave pages)
205         return $statement->fetch()["pages"];
206     } catch (PDOException $e) {
207         Utils::save_log($e->getMessage());
208     }
209     return null;
210 }
211 }

```



Controller/index_product.php

Este archivo puede parecer lioso a simple vista, pero en realidad no lo es tanto.

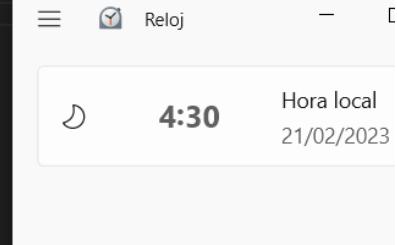
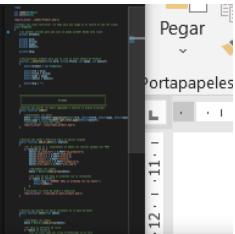
Creamos una clase llamada **ProductC** – *Controlador para Producto* –. En esta clase, definimos una variable llamada **\$element** que inicializaremos en el constructor para crear un nuevo objeto de la clase **Product**, otros cuatro que corresponden al campo por el que queremos ordenar los datos, el tipo de ordenación, la página y la cantidad de elementos que queremos ver por cada página y que al crear un nuevo objeto de la clase **ProductC** se pasarán como argumentos y uno para almacenar el mensaje resultante.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

```

1 <?php
2
3 use \model\Product;
4 use utils\Utils;
5
6 require_once("../model/Product.php");
7
8 //Creamos una clase controller (lo hago para que luego en el switch no sea tan lioso)
9 class ProductC
10 {
11     //Lo ponemos privado para que solo se pueda acceder desde esta clase
12     private $element;
13
14     private $ord;
15     private $field;
16     private $page;
17     private $amount;
18
19     private $msg;
20

```



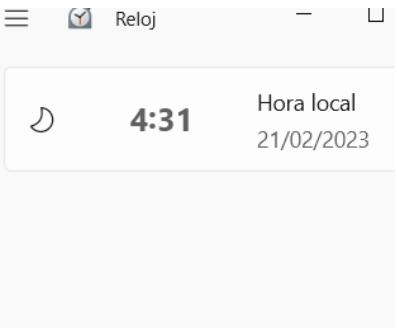
```

//Inicializamos element para que se cree un nuevo objeto de Product
function __construct(string $ord, string $field, int $page, int $amount)
{
    $this->element = new Product();

    $this->ord = $ord;
    $this->field = $field;
    $this->page = $page;
    $this->amount = $amount;

    $this->msg = "";
}

```



Tras esto, crearemos varios métodos.

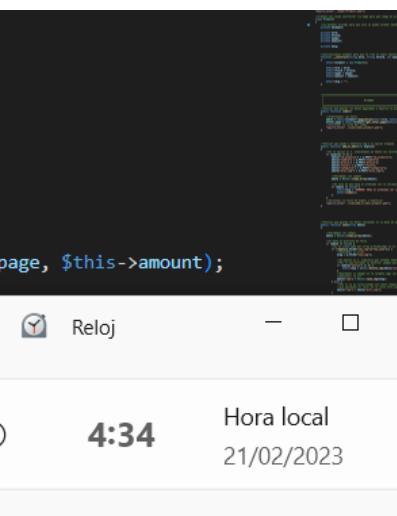
La primera función es **index()** que nos llevará a la página principal de Producto, en la que se verán los datos paginados.

```

***** MÉTODOS *****
*                                                               *
*                                                               *
*****                                                               */
//Funcion que guarda los datos paginados y muestra la página principal
public function index()
{
    //Almacenamos los datos
    $data = $this->element->pagination($this->ord, $this->field, $this->page, $this->amount);
    $total_page = $this->element->get_total_pages($this->amount);
    //Incluimos la vista del index.
    require_once("../view/index_product.php");
}

//Funcion que añade o modifica según la opcion elegida

```



	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo  Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
---	---	--	--

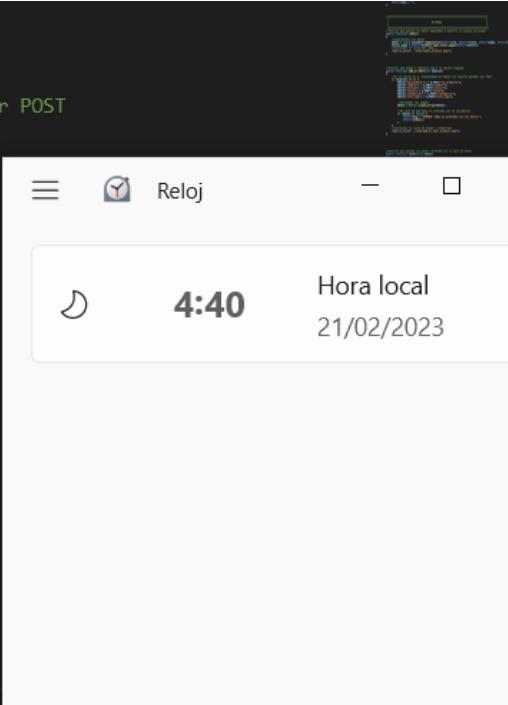
Luego, tenemos la función **add_or_edit()** que, realizará una acción u otra dependiendo de la opción elegida. Si queremos modificar un elemento preexistente, almacenamos los datos pasados por **POST** y validamos y limpiamos los datos con la función **clean_array()** de **Utils**.

Si el resultado es false, nos saltará un mensaje de error y volveremos al índice. Si todo va bien, vamos al archivo **add_or_edit_product.php** que se encuentra en **view**.

```
//Funcion que añade o modifica según la opcion elegida
public function add_or_edit(int $option)
{
    //Si la opcion es 2, almacenamos en $data los valores pasados por POST
    if ($option == 2) {
        $data["id_producto"] = $_POST["id_producto"];
        $data["nombre"] = $_POST["nombre"];
        $data["precio"] = $_POST["precio"];
        $data["stock"] = $_POST["stock"];
        $data["categoria"] = $_POST["categoria"];
        $data["prev_img"] = $_POST["prev_img"];

        //Validamos los campos
        $data = Utils::clean_array($data);

        //En caso de que haya un problema con la validación
        if ($data == false) {
            $this->msg = "¡ERROR! ¡Hay un problema con los datos!";
            $this->index();
        }
    }
    //Incluimos la vista de añadir o modificar
    require_once("../view/add_or_edit_product.php");
}
```



Como creo que es necesario ver el archivo **add_or_edit_product.php** para continuar con la explicación de la siguiente función, lo mostraremos ahora mismo.

Este fichero, corresponde a la vista de Modificación e Inserción.

En caso de que la opción elegida sea modificar un elemento preexistente, **\$option** valdrá 2 y se mostrará un input con el **id_producto** del objeto seleccionado por si deseamos modificarlo. En caso de actualizar su **id**, deberemos de pasar el valor antiguo – **hidden** – y el nuevo por **POST**.

En el resto de inputs, realizaremos un condicional ternario para establecer el valor del input en función de si la opción seleccionada es 2 – **modificar** – o 1 – **añadir** –.

Por último, en caso de que la opción sea dos y tendremos que guardar la url previa y pasarlpor **POST** por si no cambiamos la imagen guardada por otra. Por lo tanto, en caso de seleccionar modificar, pasaremos la ruta anterior con un **input hidden** y la nueva ruta con un **input file**.

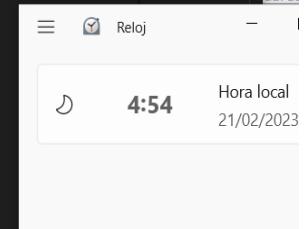
Si nos fijamos, el botón submit tiene un valor 5, lo que nos indica que se debe de llamar a la función **save()** que veremos a continuación.

Y es muy importante que en el formulario pongas **enctype="multipart/form-data"** para indicar que vas a pasar un archivo.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 <small>Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo</small>
---	---	--	--

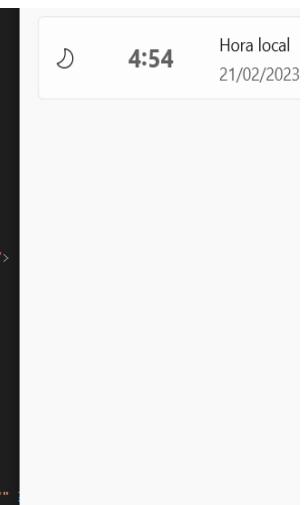
```

tienda_animales3 > view > add_or_edit_product.php > html > head > title
1  <html lang="en">
2
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <!-- Si la opción es 1, el título de la página será New Product, si no, será Edit Product -->
8      <title><?php ($option == 1) ? "New Product" : "Edit Product" ?></title>
9      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Zenh8qX5JnK2J10vWa8C" crossorigin="anonymous">
10     <script src="https://kit.fontawesome.com/8d125d2b91.js" crossorigin="anonymous"></script>
11
12 </head>
13 <body class="bg-secondary">
14     <div class="container mt-5">
15         <div class="d-flex align-items-center justify-content-center">
16             <form action="index_product.php" method="POST" enctype="multipart/form-data" class="bg-dark text-white p-4" style="width: 370px;" id="form">
17                 <!-- Si la opción es 1, ponemos Add Product, si no lo es, edit Product -->
18                 <h2 class="text-center my-4"><?php ($option == 1) ? "Add Product" : "Edit Product" ?></h2>
19                 <!-- Si la opción es 1, no aparecerá el recuadro de id_product-->
20                 <?php
21                     //Si la opción es 2, nos aparece un input para modificar el identificador
22                     if ($option == 2) {
23                         echo "<div class='form-group col-md-12'>";
24                         echo "<label for='id_producto' class='mb-3'>Id</label>";
25                         echo "<input type='text' name='new_id' class='form-control' value=' . $data['id_producto'] . '>";
26                         echo "<input type='hidden' name='id_producto' value=' . $data['id_producto'] . '>";
27                         echo "</div>";
28                     }
29                 >
30
31             //Si la opción es 2, nos aparece un input para modificar el identificador
32             if ($option == 2) {
33                 echo "<div class='form-group col-md-12'>";
34                 echo "<label for='id_producto' class='mb-3'>Id</label>";
35                 echo "<input type='text' name='new_id' class='form-control' value=' . $data['id_producto'] . '>";
36                 echo "<input type='hidden' name='id_producto' value=' . $data['id_producto'] . '>";
37                 echo "</div>";
38
39             <div class="form-group col-md-12">
40                 <label for="nombre" class="mt-2 mb-1">Nombre</label>
41                 <!-- Si la opción vale 2, se muestran datos del nombre a cambiar -->
42                 <input type="text" name="nombre" class="form-control" value=<?php ($option == 2) ? $data["nombre"] : "" ?>>;
43             </div>
44
45             <div class="form-group col-md-12">
46                 <label for="precio" class="mt-2 mb-1">Precio</label>
47                 <input type="text" name="precio" class="form-control" value=<?php ($option == 2) ? $data["precio"] : "" ?>>;
48             </div>
49
50             <div class="form-group col-md-12">
51                 <label for="stock" class="mt-2 mb-1">Stock</label>
52                 <input type="text" name="stock" class="form-control" value=<?php ($option == 2) ? $data["stock"] : "" ?>>;
53             </div>
54
55             <div class="form-group col-md-12">
56                 <label for="categoria" class="mt-2 mb-1">Categoria</label>
57                 <input type="text" name="categoria" class="form-control" value=<?php ($option == 2) ? $data["categoria"] : "" ?>>;
58             </div>
59
60             <div class="form-group col-md-12">
61                 <label for="img" class="mt-2 mb-1">Foto/s</label>
62                 <?php
63                     //Si la opción es 2, significa que queremos modificar el elemento
64                     //Sin embargo, puede ser que no queramos cambiar la foto ya subida
65                     //así que guardamos su url
66                     if ($option == 2) {
67                         echo "<input type='hidden' name='prev_img' value=' . $data['prev_img'] . ''>";
68                     }
69                 >
70
71                 <input type="file" name="new_img" class="form-control">
72             </div>
73             <input type="hidden" name="option" value=<?php ($option == 2) ? "2" : "1" ?>>
74             <div class="form-group col-md-12 text-center">
75                 <button type="submit" value="5" name="submit" class="mt-3 btn btn-secondary">Guardar</button>
76             </div>
77         </form>
78     </div>
79
80     <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcWSD95VBN60yTFYIQUQvHRRt22sHQJmOoymWExBbph+OwqK2dZcaQaF9" crossorigin="anonymous">
81     <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/javascript"></script>
82     <script src="..view/js/index.js"></script>
83
84 >
```



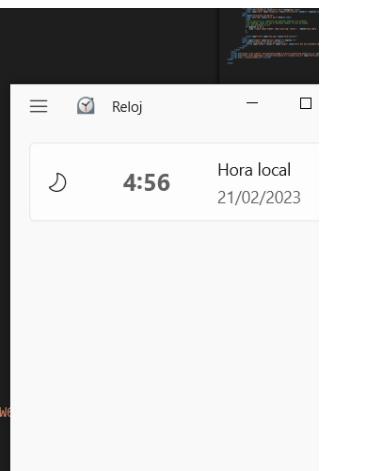
```

85
86             <div class="form-group col-md-12">
87                 <label for="id_producto" class="mb-3" style="color: #ccc;">Id</label>
88                 <input type="text" name="new_id" class="form-control" value=" . $data['id_producto'] . ">
89             </div>
90
91             <div class="form-group col-md-12">
92                 <label for="nombre" class="mb-3" style="color: #ccc;">Nombre</label>
93                 <input type="text" name="nombre" class="form-control" value=" . $data['nombre'] . ">
94             </div>
95
96             <div class="form-group col-md-12">
97                 <label for="precio" class="mb-3" style="color: #ccc;">Precio</label>
98                 <input type="text" name="precio" class="form-control" value=" . $data['precio'] . ">
99             </div>
100
101            <div class="form-group col-md-12">
102                <label for="stock" class="mb-3" style="color: #ccc;">Stock</label>
103                <input type="text" name="stock" class="form-control" value=" . $data['stock'] . ">
104            </div>
105
106            <div class="form-group col-md-12">
107                <label for="categoria" class="mb-3" style="color: #ccc;">Categoria</label>
108                <input type="text" name="categoria" class="form-control" value=" . $data['categoria'] . ">
109            </div>
110
111            <div class="form-group col-md-12">
112                <label for="img" class="mb-3" style="color: #ccc;">Foto/s</label>
113                <?php
114                    //Si la opción es 2, significa que queremos modificar el elemento
115                    //Sin embargo, puede ser que no queramos cambiar la foto ya subida
116                    //así que guardamos su url
117                    if ($option == 2) {
118                        echo "<input type='hidden' name='prev_img' value=' . $data['prev_img'] . ''>";
119                    }
120
121                    <input type="file" name="new_img" class="form-control">
122                </div>
123                <input type="hidden" name="option" value="2">
124                <div class="form-group col-md-12 text-center">
125                    <button type="submit" value="5" name="submit" class="mt-3 btn btn-secondary">Guardar</button>
126                </div>
127            </form>
128        </div>
129
130        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcWSD95VBN60yTFYIQUQvHRRt22sHQJmOoymWExBbph+OwqK2dZcaQaF9" crossorigin="anonymous">
131        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/javascript"></script>
132        <script src="..view/js/index.js"></script>
133
134 >
```



```

135
136             <div class="form-group col-md-12">
137                 <label for="id_producto" class="mb-3" style="color: #ccc;">Id</label>
138                 <input type="text" name="new_id" class="form-control" value=" . $data['id_producto'] . ">
139             </div>
140
141             <div class="form-group col-md-12">
142                 <label for="nombre" class="mb-3" style="color: #ccc;">Nombre</label>
143                 <input type="text" name="nombre" class="form-control" value=" . $data['nombre'] . ">
144             </div>
145
146             <div class="form-group col-md-12">
147                 <label for="precio" class="mb-3" style="color: #ccc;">Precio</label>
148                 <input type="text" name="precio" class="form-control" value=" . $data['precio'] . ">
149             </div>
150
151            <div class="form-group col-md-12">
152                <label for="stock" class="mb-3" style="color: #ccc;">Stock</label>
153                <input type="text" name="stock" class="form-control" value=" . $data['stock'] . ">
154            </div>
155
156            <div class="form-group col-md-12">
157                <label for="categoria" class="mb-3" style="color: #ccc;">Categoria</label>
158                <input type="text" name="categoria" class="form-control" value=" . $data['categoria'] . ">
159            </div>
160
161            <div class="form-group col-md-12">
162                <label for="img" class="mb-3" style="color: #ccc;">Foto/s</label>
163                <?php
164                    //Si la opción es 2, significa que queremos modificar el elemento
165                    //Sin embargo, puede ser que no queramos cambiar la foto ya subida
166                    //así que guardamos su url
167                    if ($option == 2) {
168                        echo "<input type='hidden' name='prev_img' value=' . $data['prev_img'] . ''>";
169                    }
170
171                    <input type="file" name="new_img" class="form-control">
172                </div>
173                <input type="hidden" name="option" value="2">
174                <div class="form-group col-md-12 text-center">
175                    <button type="submit" value="5" name="submit" class="mt-3 btn btn-secondary">Guardar</button>
176                </div>
177            </form>
178        </div>
179
180        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcWSD95VBN60yTFYIQUQvHRRt22sHQJmOoymWExBbph+OwqK2dZcaQaF9" crossorigin="anonymous">
181        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/javascript"></script>
182        <script src="..view/js/index.js"></script>
183
184 >
```



 <p>I.E.S. MAR DE CÁDIZ</p>	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 <p>Este proyecto ha sido financiado por el Ministerio de Educación, Cultura y Deporte y co-financiado por el Fondo Social Europeo. "El FSE invierte en tu futuro"</p>
--	---	--	---

Volvemos de nuevo a [controller/index_product.php](#) para continuar explicando la función **save()** que se llamará al enviar el formulario anterior.

A este método, se le pasará como argumento un array que validaremos y limpiaremos con la función **clean_array()** de la clase **Utils**.

Si el resultado es **false**, nos dirigirá a la **página principal** de **Productos** y nos saltará una advertencia.

En caso de que no lo sea, se comprobará que **\$_FILES["new_img"]["tmp_name"]** no esté vacío. Si está vacío, significa que la imagen no ha sido modificada (si fuese la opción de añadir, no debería de estar vacío, pero si lo está ponemos la ruta “../default.jpeg” para una imagen por defecto) y por lo tanto, seguimos almacenando la ruta anterior.

Si no lo está, almacenamos el array **\$_FILES["new_img"]**.

El hecho de que no esté vacío puede significar que o bien queremos añadir un nuevo producto y por lo tanto subir la imagen de este o bien cambiar la imagen de un elemento ya existente.

En caso de que se tratase de una modificación, eliminaríamos la imagen anterior.

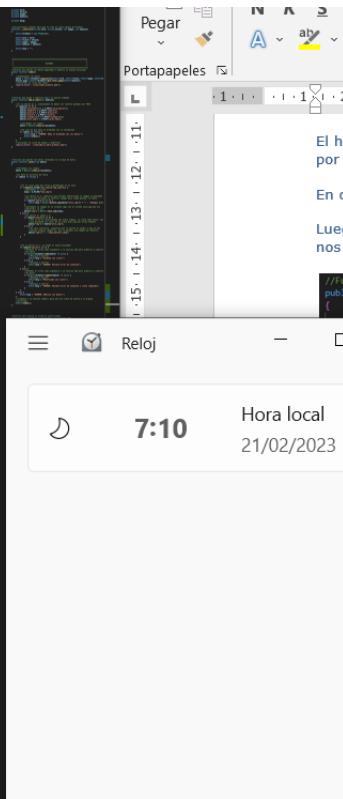
Luego, tanto si queremos añadir o modificar, guardamos la imagen con **Utils::save_img()** que nos devolverá la ruta de la imagen.

```
//Funcion que guarda los datos recibidos en la base de datos
public function save(array $data)
{
    //Validamos los campos
    $data = Utils::clean_array($data);

    //Si data es distinta de false
    if ($data != false) {

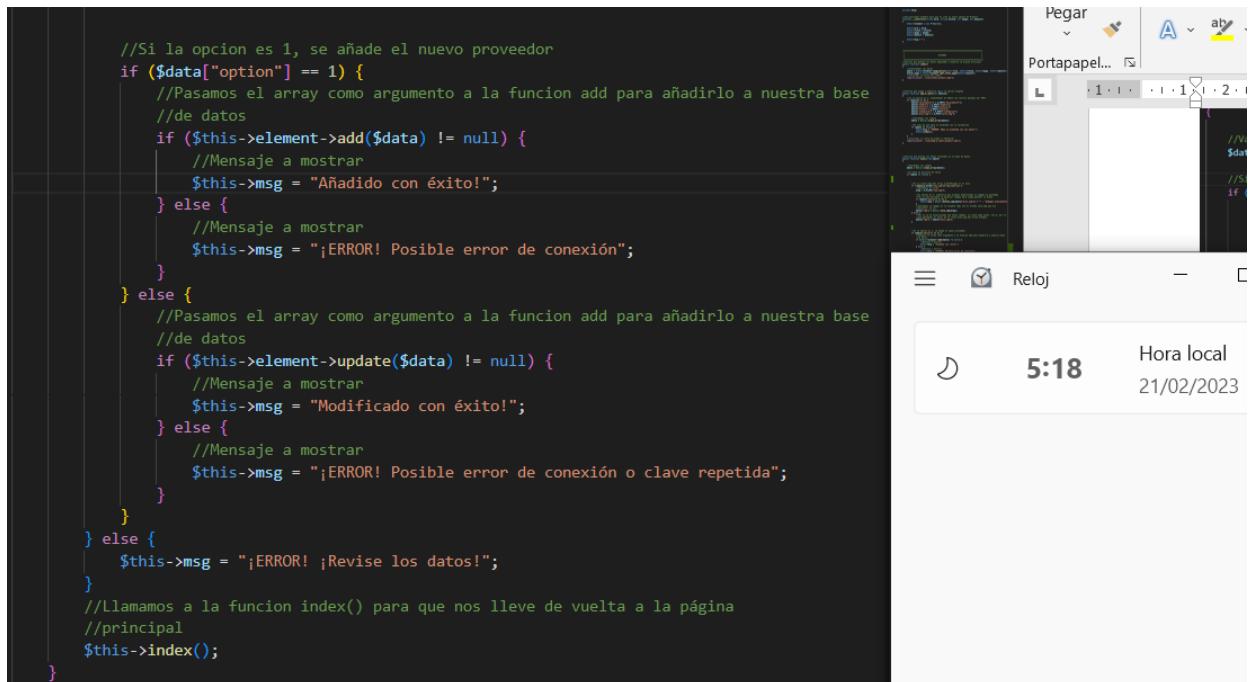
        //Si la clave name del array $_FILES[img] no es nula
        if (!empty($_FILES["new_img"]["tmp_name"])) {
            //Guardamos el array
            $img = $_FILES["new_img"];

            //Si option es 2, significa que estamos modificando la imagen ya guardada
            //Por lo que borramos la anterior imagen para luego guardar la nueva
            if ($data["option"] == 2) {
                $this->msg = Utils::delete_img($data["prev_img"]) ? "" : "¡Imagen previa!";
            }
            //Guardamos la imagen en la carpeta img con el método save_img que nos
            //devolverá la url
            $data["img"] = Utils::save_img($img);
        } else {
            //Si option es igual a 2;
            if ($data["option"] == 2) {
                //PSi no se ha seleccionado una nueva imagen, la clave name estará vacía
                //que guardamos el valor de la clave prev_img del array element
                $data["img"] = $data["prev_img"];
            } else {
                //En caso contrario, significa que la opcion es añadir y que no han
                //metido ninguna imagen así que le ponemos una imagen por defecto
                $data["img"] = "../img/default.jpeg";
            }
        }
    }
}
```



Luego de hacer esto, se llamará a la función **add** o **update** de la clase **Product** en función de si hemos seleccionado una opción u otra.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

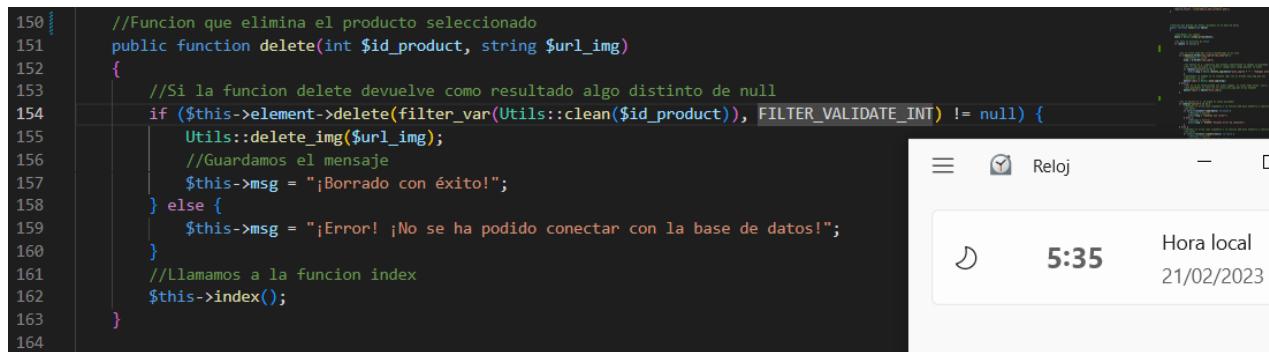


```

//Si la opcion es 1, se añade el nuevo proveedor
if ($data["option"] == 1) {
    //Pasamos el array como argumento a la funcion add para añadirlo a nuestra base
    //de datos
    if ($this->element->add($data) != null) {
        //Mensaje a mostrar
        $this->msg = "Añadido con éxito!";
    } else {
        //Mensaje a mostrar
        $this->msg = "¡ERROR! Posible error de conexión";
    }
} else {
    //Pasamos el array como argumento a la funcion add para añadirlo a nuestra base
    //de datos
    if ($this->element->update($data) != null) {
        //Mensaje a mostrar
        $this->msg = "Modificado con éxito!";
    } else {
        //Mensaje a mostrar
        $this->msg = "¡ERROR! Posible error de conexión o clave repetida";
    }
}
else {
    $this->msg = "¡ERROR! ¡Revise los datos!";
}
//Llamamos a la funcion index() para que nos lleve de vuelta a la página
//principal
$this->index();
}

```

Tras esta función tenemos **delete()**, que eliminará el producto seleccionado.



```

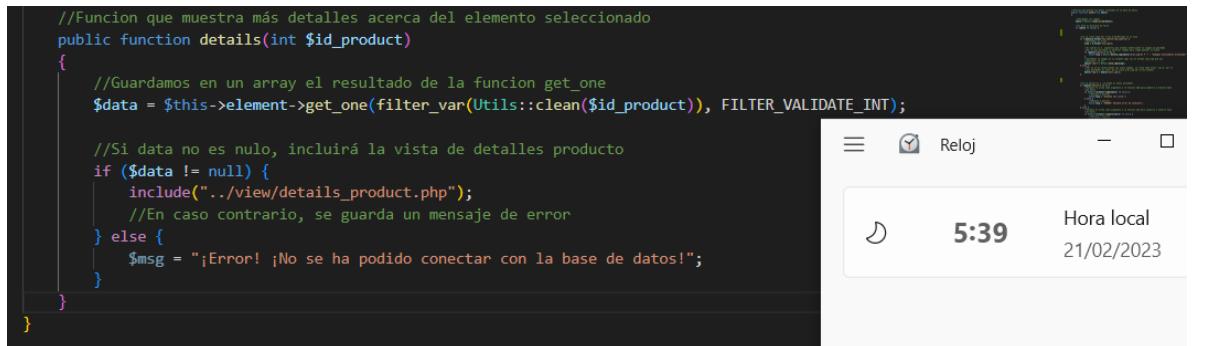
150 //Funcion que elimina el producto seleccionado
151 public function delete(int $id_product, string $url_img)
152 {
153     //Si la funcion delete devuelve como resultado algo distinto de null
154     if ($this->element->delete(filter_var(Utils::clean($id_product), FILTER_VALIDATE_INT) != null) {
155         Utils::delete_img($url_img);
156         //Guardamos el mensaje
157         $this->msg = "¡Borrado con éxito!";
158     } else {
159         $this->msg = "¡Error! ¡No se ha podido conectar con la base de datos!";
160     }
161     //Llamamos a la funcion index
162     $this->index();
163 }
164

```

Y la función **details()**, que nos mostrará más detalles acerca del elemento seleccionado.

Esta función se llama con **AJAX** por lo que no se carga una nueva página y nos permitirá ver más detalles acerca del dato clicado. Más adelante, en este documento veremos el documento de JavaScript que hemos creado.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---



```

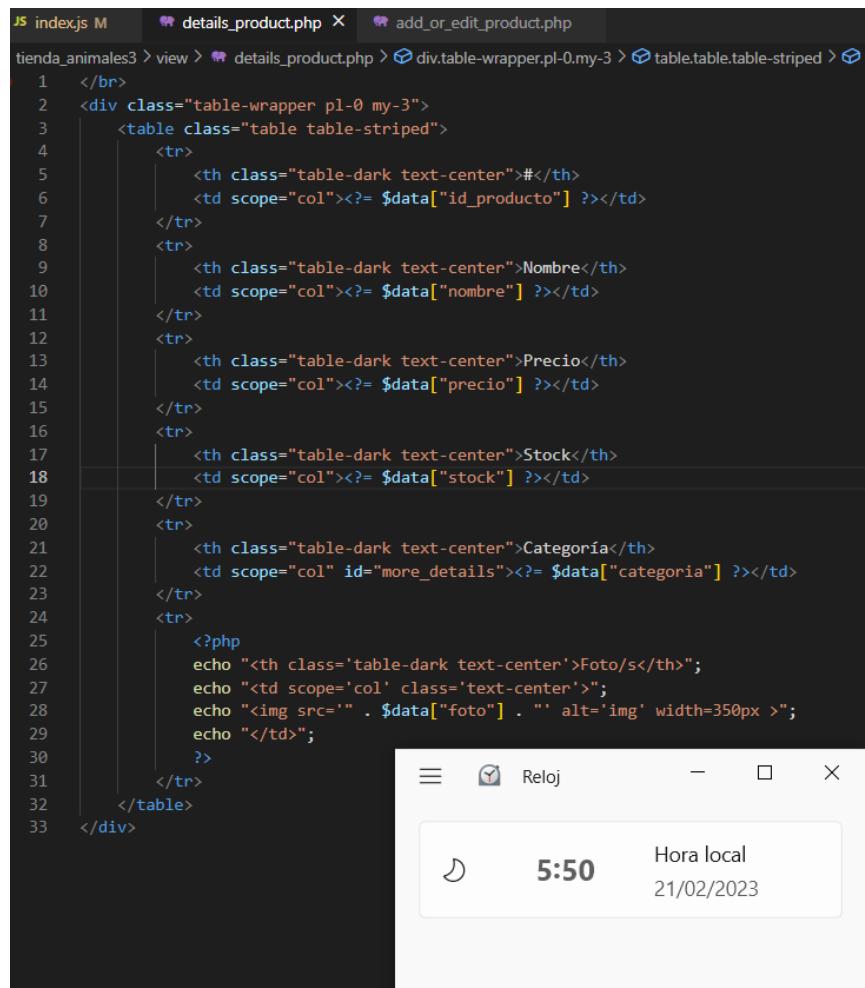
//Funcion que muestra más detalles acerca del elemento seleccionado
public function details(int $id_product)
{
    //Guardamos en un array el resultado de la función get_one
    $data = $this->element->get_one(Utils::clean($id_product), FILTER_VALIDATE_INT);

    //Si data no es nulo, incluirá la vista de detalles producto
    if ($data != null) {
        include("../view/details_product.php");
        //En caso contrario, se guarda un mensaje de error
    } else {
        $msg = "¡Error! ¡No se ha podido conectar con la base de datos!";
    }
}

```

Al igual que con add_or_edit_product.php, veo necesario mostrar el archivo **details_product.php** ya que es lo que veremos cuando se llame a la función.

Es bastante simple, solo mostramos los valores de \$data.



```

indexjs M  details_product.php X  add_or_edit_product.php
tienda_animales3 > view > details_product.php > div.table-wrapper.pl-0.my-3 > table.table.table-striped > t
1   </br>
2   <div class="table-wrapper pl-0 my-3">
3     <table class="table table-striped">
4       <tr>
5         <th class="table-dark text-center">#</th>
6         <td scope="col"><?= $data["id_producto"] ?></td>
7       </tr>
8       <tr>
9         <th class="table-dark text-center">Nombre</th>
10        <td scope="col"><?= $data["nombre"] ?></td>
11      </tr>
12      <tr>
13        <th class="table-dark text-center">Precio</th>
14        <td scope="col"><?= $data["precio"] ?></td>
15      </tr>
16      <tr>
17        <th class="table-dark text-center">Stock</th>
18        <td scope="col"><?= $data["stock"] ?></td>
19      </tr>
20      <tr>
21        <th class="table-dark text-center">Categoría</th>
22        <td scope="col" id="more_details"><?= $data["categoria"] ?></td>
23      </tr>
24      <tr>
25        <?php
26        echo "<th class='table-dark text-center'>Foto/s</th>";
27        echo "<td scope='col' class='text-center'>";
28        echo "<img src='' . $data["foto"] . '' alt='img' width=350px >";
29        echo "</td>";
30      ?>
31      </tr>
32    </table>
33  </div>

```

Con este último método, hemos terminado la clase **ProductC**.

Ahora, fuera de la clase, comprobamos que **\$_REQUEST** no esté vacío con la función **isset()**.

	I.E.S. MAR DE CÁDIZ	MODELO	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo 'El FSE invierte en tu futuro'
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

Se comprueba que sus claves **field**, **ord**, **page** y **submit** no sean nulos y en caso de que así sea, se establece unos valores predeterminados.

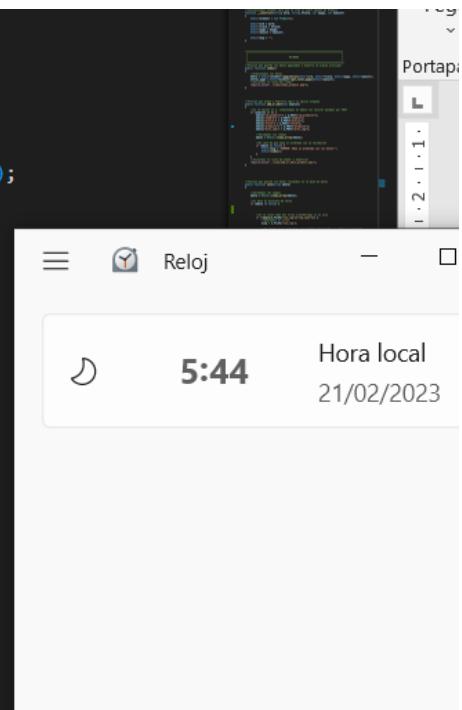
```
//Si el array $_REQUEST no está vacío
if (isset($_REQUEST)) {

    //Comprobamos sus valores
    if (isset($_REQUEST["field"])) {
        $field = filter_var($_REQUEST["field"], FILTER_SANITIZE_SPECIAL_CHARS);
    } else {
        $field = "id_producto";
    }

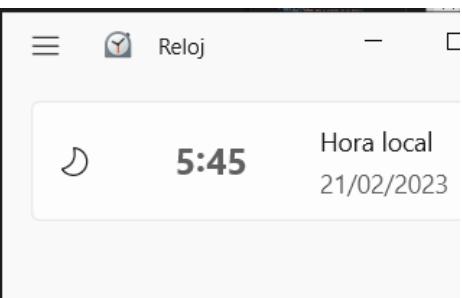
    if (isset($_REQUEST["ord"])) {
        $ord = filter_var($_REQUEST["ord"], FILTER_SANITIZE_SPECIAL_CHARS);
    } else {
        $ord = "ASC";
    }

    if (isset($_REQUEST["page"])) {
        $page = filter_var($_REQUEST["page"], FILTER_VALIDATE_INT);
    } else {
        $page = 1;
    }

    //Cantidad que queremos que salga
    $amount = 5;
}
```

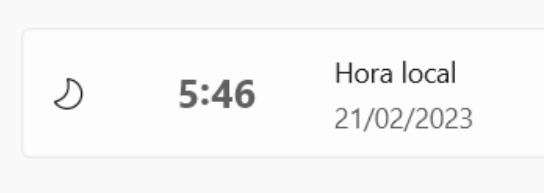


```
//Si la clave submit no es nula
if (isset($_REQUEST["submit"])) {
    //Guardamos la opción elegida y filtramos su valor con filter_var
    $option = filter_var($_REQUEST["submit"], FILTER_VALIDATE_INT);
    //En caso contrario, inicializamos option
} else {
    $option = 0;
}
```



Luego, creamos un nuevo objeto de la clase **ProductC** y le pasamos como argumento las variables **ord**, **field**, **page** y **amount**.

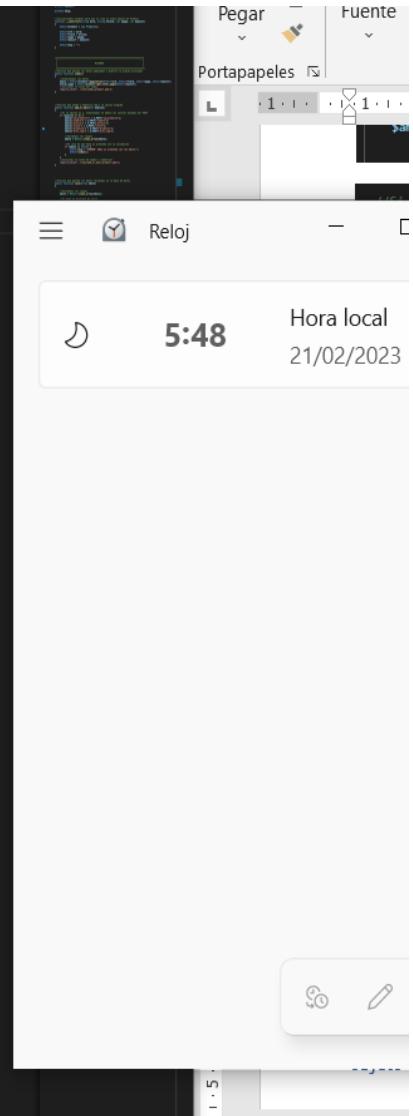
```
//Creamos un nuevo objeto de la clase productC
$element = new productC($ord, $field, $page, $amount);
```



Usamos un switch para que dependiendo del valor de **\$option**, se llame a una función u otra del objeto de la clase **ProductC**.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 Proyecto financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo 'El FSE invierte en tu futuro'
---	---	--	--

```
//Creamos un switch para que dependiendo del valor de option se haga una cosa u otra
switch ($option) {
    //Si es 0, llamamos al index
    case 0:
        $element->index();
        break;
    //Si es 1 (añadir) o 2 (modificar)
    case 1:
    case 2:
        //Llamamos a la funcion add_or_edit y le pasamos $option para que dependiendo
        //de la opcion elegida nos salga una cosa u otra
        $element->add_or_edit($option);
        break;
    //Si es 3, llamamos a la funcion delete
    case 3:
        //Si id_proveedor no es nulo y es numerico
        if (isset($_POST["id_producto"]) && is_numeric($_POST["id_producto"])) {
            //Se llama a la funcion delete
            $element->delete($_POST["id_producto"], $_POST["img"]);
        } else {
            $this->msg = "Clave no numérica!";
            $element->index();
        }
        break;
    //Si es 4, llamamos a la funcion details que nos mostrará más detalles acerca
    //del elemento que hayamos pinchado
    case 4:
        //Si el id_proveedor no está vacío y es un número
        if (isset($_POST["id"]) && is_numeric($_POST["id"])) {
            //Llamamos a la funcion details
            $element->details($_POST["id"]);
        } else {
            $this->msg = "Clave no numérica!";
            $element->index();
        }
        break;
    //La opcion 5, la obtenemos al enviar los datos del formulario de modificar
    //o añadir y lo que hará es llamar a la función save, pasándole el array $_POST
    //para su introducción en la base de datos
    case 5:
        $element->save($_POST);
}
```



view/index_product.php

En la función index() de la clase ProductC, vemos el archivo view/index_product.php que mostrará los datos paginados.

Esta vista se ha creado usando Bootstrap y es bastante simple.

En ella tenemos un modal que se activa cuando tenemos un mensaje que mostrar.

Una tabla con los datos, en la que se puede decidir si queremos que tengan un orden ascendente o descendente y que se ordenen según la casilla clicada y podremos añadir, modificar o eliminar un elemento.

No hay mucho misterio en esto, así que pondré las capturas directamente.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

index.js M index_product.php M

```

1 ienda_animales3 > view > index_product.php > html > body > div.container.mt-5 > div.table-wrapper > table.table-striped > tbody
2   <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <meta http-equiv="X-UA-Compatible" content="IE=edge">
6       <meta name="viewport" content="width=device-width, initial-scale=1.0">
7       <title>Productos</title>
8       <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="..."/>
9       <script src="https://kit.fontawesome.com/8d125d2b91.js" crossorigin="anonymous"></script>
10      </head>
11
12      <body>
13        <div class="container mt-5">
14          <!-- MODAL -->
15          <div id="aviso" class="modal" tabindex="-1" role="dialog" data-backdrop="static" data-keyboard="false">
16            <div class="modal-dialog" role="document">
17              <div class="modal-content">
18                <div class="modal-header">
19                  <h5 class="modal-title">Aviso</h5>
20                  <button type="button" class="btn-close" id="cerrarModalBtn" aria-label="Close" data-dismiss="modal"></button>
21                </div>
22                <div class="modal-body">
23                  <p><?= $this->msg ?></p>
24                </div>
25              </div>
26            </div>
27          </div>
28
29          <div class="table-wrapper">
30            <div class="table-tittle">
31              <div class="row">
32                <div class="col-sm-8">
33                  <h2><b>Productos</b></h2>
34                </div>

```

Reloj Hora local 21/02/2023

5:57

```

<div class="col-sm-6">
  <h2><b>Productos</b></h2>
</div>
<div class="col-sm-4 text-end">
  <form action="index_product.php" method="POST">
    <button name="submit" value="1" class="btn btn-success add-new">
      Añadir
    </button>
  </form>
</div>
</div>
<table class="table table-striped">
  <thead class="table-dark text-center">
    <tr>
      <!-- Si $this->ord es ASC, tendrá la clase desc, si NO lo es, la clase será ASC -->
      <!-- Se puede mejorar pero ya si eso en otra vida porque yo no puedo más -->
      <th scope="col">Nombre<a class="p-1 text-white" href="index_product.php?field=nombre"><i class="fa-sharp fa-solid fa-share-nodes"></i></a></th>
      <th scope="col">Precio<a class="p-1 text-white" href="index_product.php?field=precio"><i class="fa-sharp fa-solid fa-dollar-sign"></i></a></th>
      <th scope="col">Stock<a class="p-1 text-white" href="index_product.php?field=stock"><i class="fa-sharp fa-solid fa-database"></i></a></th>
    </tr>
  </thead>
  <tbody>
    <?php
    //Contador
    $i = 1;
    //Recorremos el array data
    foreach ($data as $element) {

```

Reloj Hora local 21/02/2023

6:00

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
---	---	--	--

```

:body>
<?php
//Recorremos el array data
foreach ($data as $element) {
    //Url destino
    $url = "index_product.php";

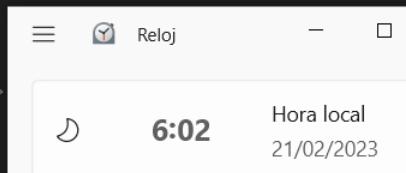
    echo "<tr>";
    //Mostramos los datos de cada elemento
    //Llamamos a la funcion details para mostrar más datos del elemento seleccionado
    echo "<td><a style='cursor:pointer;' onclick=details(" . $element["id_producto"] . ',' . $url . '')>" . $element["nombre"] . "</a>";
    echo "<td class='text-center'>" . $element["precio"] . "</td>";
    echo "<td class='text-center'>" . $element["stock"] . "</td>";
    echo "<td class='p-0'>";

?>
    <!-- Creamos dos botones para modificar o eliminar y pasamos los datos con input hidden -->
    <form action="index_product.php" method="POST" class="d-inline-block" enctype="multipart/form-data">
        <input type="hidden" name="id_producto" value=<?= $element["id_producto"] ?>>
        <input type="hidden" name="nombre" value=<?= $element["nombre"] ?>>
        <input type="hidden" name="precio" value=<?= $element["precio"] ?>>
        <input type="hidden" name="stock" value=<?= $element["stock"] ?>>
        <input type="hidden" name="categoria" value=<?= $element["categoria"] ?>>
        <input type="hidden" name="prev_img" value=<?= $element["foto"] ?>>
        <button value="2" name="submit" class="mt-2 border-0 bg-transparent text-warning">
            <i class="fa-solid fa-marker"></i>
        </button>
    </form>

    <i class="text-muted">|</i>

    <form action="index_product.php" method="POST" class="d-inline-block">
        <input type="hidden" name="id_producto" value=<?= $element["id_producto"] ?>>
        <input type="hidden" name="img" value=<?= $element["foto"] ?>>
        <button value="3" name="submit" class="mt-2 border-0 bg-transparent text-danger">
            <i class="fa-solid fa-trash-can"></i>
        </button>
    </form>
<?php
    echo "</td>";
    echo "</tr>";
}
?>
'tbody'

```



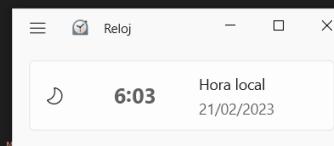
```

</tbody>
</table>
<div id="info_content">
</div>
<div>
<nav>
    <ol class="pagination justify-content-end" style="list-style-type: none;">
        <?php
            //Si la página actual no es la primera, activamos previous. Si es la primera
            //lo desactivamos con la clase disabled
            if ($this->page != 1) {
                echo "<li class='page-item'><a href='index_product.php?field=".($this->field)."&ord=".($this->ord)."&page=".($this->page - 1)."' class='page-link'>Previous</a></li>";
            } else {
                echo "<li class='page-item disabled'><a class='page-link'>Previous</a></li>";
            }

            for ($i = 1; $i <= $total_page; $i++) {
                echo "<li class='page-item'".($i == $this->page) ? ' active' : '' . "><a href='index_product.php?field=".($this->field)."&ord=".($this->ord)."&page=$i' class='page-link'>$i</a></li>";
            }

            //Hacemos lo mismo que con previous
            if ($this->page != $total_page) {
                echo "<li class='page-item'><a href='index_product.php?field=".($this->field)."&ord=".($this->ord)."&page=".($this->page + 1)."' class='page-link'>Next</a></li>";
            } else {
                echo "<li class='page-item disabled'><a class='page-link'>Next</a></li>";
            }
        </ol>
    </nav>
</div>
</div>

```



cXn/tWtIxVXM" crossorigin="anonymous">

```

is->msg != "") { ?>

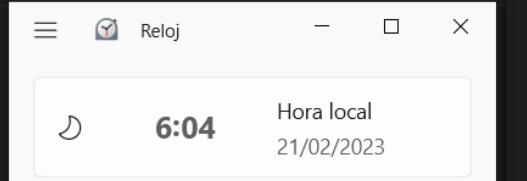
```

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

```

30   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
31   <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/
32   <?php
33   if ($this->msg != "") { ?>
34
35     <script>
36       $(document).on("ready", function() {
37         $("#aviso").modal("show");
38       })
39     </script>
40   </?php
41   }
42 ?>
43   <script language="JavaScript" type="text/javascript" src="../view/js/index.js"></script>
44 </body>
45
46 </html>

```



View/index.js

En este archivo tenemos la función **details()** en la que se utiliza AJAX para mostrar más datos del elemento seleccionado.

```

//Funcion que muestra más detalles de un elemento seleccionado de la tabla
function details(id, url) {

    //Creamos objeto XMLHttpRequest
    let xhr = new XMLHttpRequest();

    //Cada vez que haya un cambio de estado, se llama a la función
    xhr.onreadystatechange = () => {
        //Cuando el estado de nuestra petición sea 4 (completa) y el estado sea 200
        // (éxito)
        if (xhr.readyState == 4 && xhr.status == 200) {
            //Mostramos la respuesta a la petición en el elemento info_content
            $("#info_content").html(xhr.response);
        }
    }

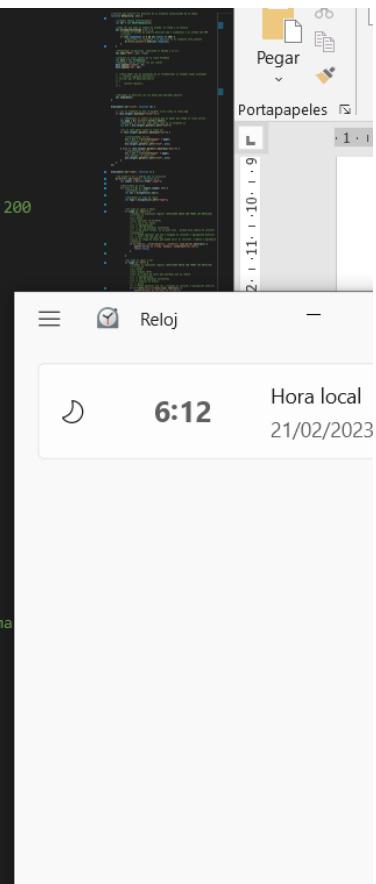
    //Preparamos la petición, indicando el método y la url
    xhr.open("POST", url, true);

    //Creamos un nuevo objeto de la clase FormData
    let data = new FormData();
    //Añadimos nuevos valores con sus claves
    data.append("submit", 4);
    data.append("id", id);

    //Para poder ver el contenido de un FormData (por si tenemos algún problema)
    // console.log(data.entries());
    // for(var par of data.entries())
    // {
    //     console.log(par);
    // }

    //Enviamos la petición con los datos que queremos pasarle
    xhr.send(data);
}

```



	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---

Luego, tenemos otra que cambia las clases asc y desc y, dependiendo de la clase que tenga el elemento al que se le ha realizado click, modificamos su atributo **href**.

```

$(document).on("click", function (e) {
    // Si el elemento al que le hacemos click tiene la clase pag
    if ($(e.target).hasClass("filter")) {

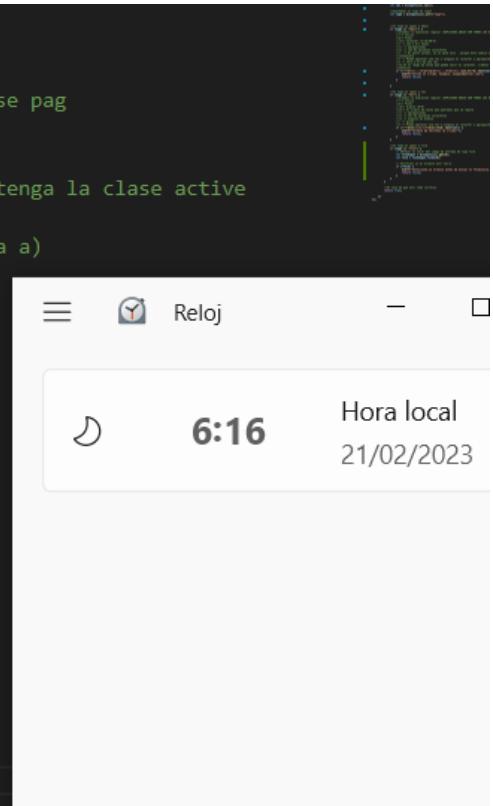
        // Obtenemos el numero de página que es aquel que tenga la clase active
        let page = $(".active").first().text();
        //Obtenemos la url del nodo padre (que es la etiqueta a)
        let url = $(e.target).parent().attr("href");

        //Si el nodo padre tiene la clase asc
        if ($(e.target).parent().hasClass("asc")) {

            //concatenamos la url
            url = url + ("&ord=DESC&page=" + page);
            //Le pasamos la nueva url
            $(e.target).parent().attr("href", url);

        } else if ($(e.target).parent().hasClass("desc")) {
            //concatenamos la url
            url = url + ("&ord=ASC&page=" + page);
            //Le pasamos la nueva url
            $(e.target).parent().attr("href", url);
        }
    });
});

```



Por último, he creado otra para que se validen ciertos campos de un formulario antes de enviarlo como pueden ser un correo electrónico, el teléfono o la subida de una imagen obligatoriamente. Está hecho a lo vago, lo siento.

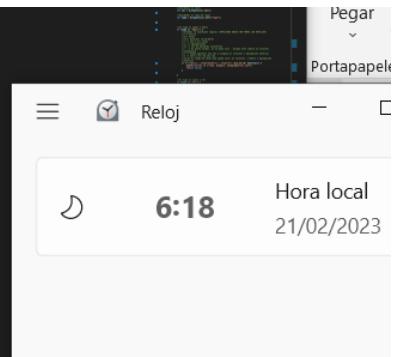
```

$(document).on("ready", function () {
    //Se añadirán más a medida que se necesiten
    $("#form").on("submit", function () {
        let inputs = $(this).find(":input");

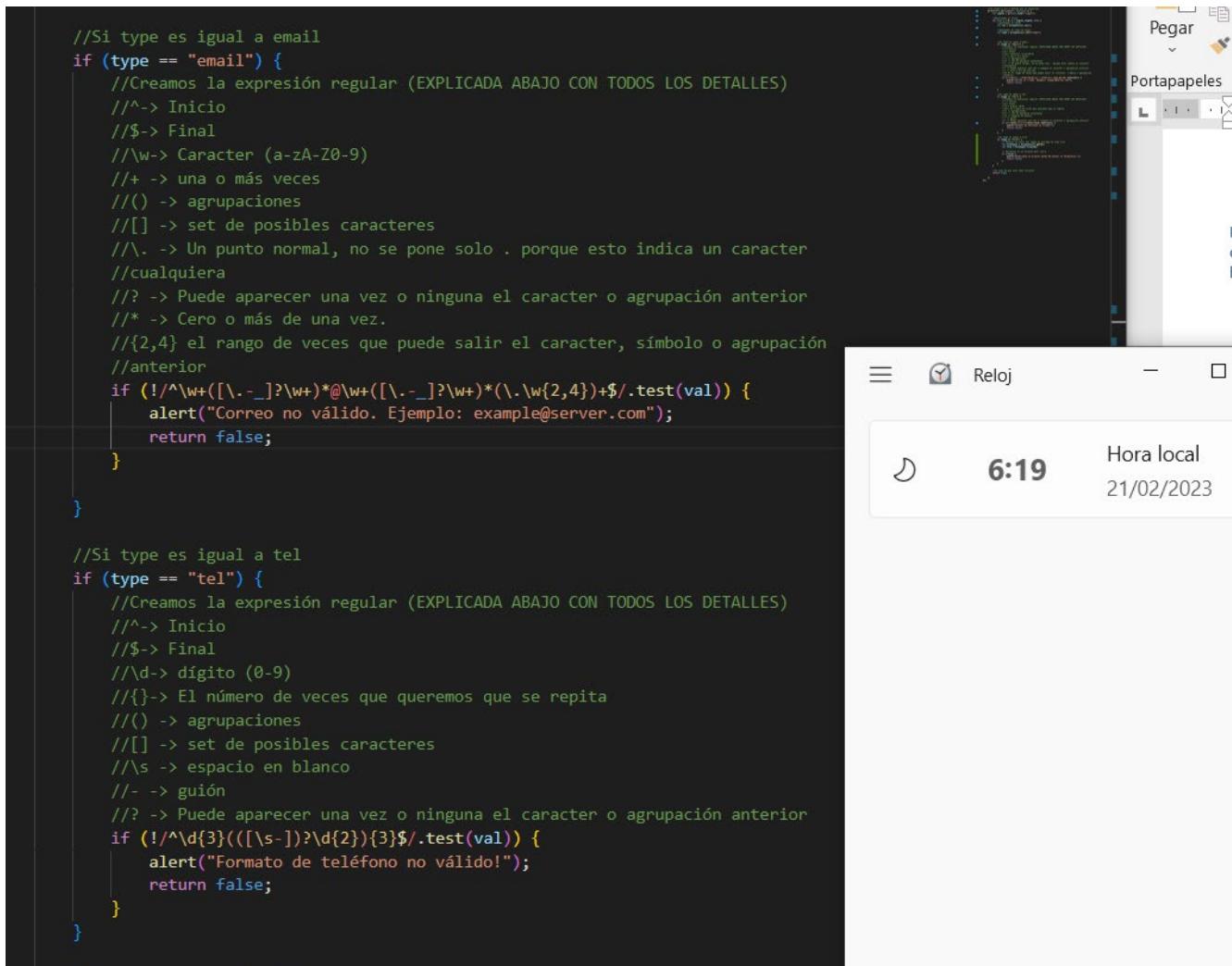
        //Recorremos el array
        for (let i = 0; i < inputs.length; i++) {
            //Guardamos el valor
            let val = $(inputs[i]).val();

            //Guardamos el tipo de input
            let type = $(inputs[i]).attr("type");
        }
    });
});

```



	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	
---	---	--	---



The screenshot shows a Windows desktop environment. On the left, a code editor window displays two sections of JavaScript code for validating email and telephone numbers. The right side of the screen features the Windows taskbar with icons for the Start button, File Explorer, Task View, and a pinned application. The system tray at the bottom right shows the date and time (21/02/2023, 6:19), a battery icon, and a network connection icon.

```

//Si type es igual a email
if (type == "email") {
    //Creamos la expresión regular (EXPLICADA ABAJO CON TODOS LOS DETALLES)
    //^ -> Inicio
    //\$ -> Final
    //\w -> Caracter (a-zA-Z0-9)
    //+ -> una o más veces
    //() -> agrupaciones
    //[] -> set de posibles caracteres
    //\. -> Un punto normal, no se pone solo . porque esto indica un carácter
    //cualquiera
    //? -> Puede aparecer una vez o ninguna el carácter o agrupación anterior
    //* -> Cero o más de una vez.
    //{2,4} el rango de veces que puede salir el carácter, símbolo o agrupación
    //anterior
    if (!/\w+([\._]?[\w+])*@\w+([\._]?[\w+])*(\.\w{2,4})+$/.test(val)) {
        alert("Correo no válido. Ejemplo: example@server.com");
        return false;
    }
}

//Si type es igual a tel
if (type == "tel") {
    //Creamos la expresión regular (EXPLICADA ABAJO CON TODOS LOS DETALLES)
    //^ -> Inicio
    //\$ -> Final
    //\d -> dígito (0-9)
    //{} -> El número de veces que queremos que se repita
    //() -> agrupaciones
    //[] -> set de posibles caracteres
    //\s -> espacio en blanco
    // -> guión
    //? -> Puede aparecer una vez o ninguna el carácter o agrupación anterior
    if (!/\d{3}(([[-]\d{2}]{3})$/.test(val)) {
        alert("Formato de teléfono no válido!");
        return false;
    }
}

```

Index.php

Archivo desde el cuál accederemos a las páginas principales de las 4 entidades seleccionadas para este ejercicio.

	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
		Pág. /



```

index.php X
tienda_animales3 > index.php > ...
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Index</title>
9      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet">
10     <script src="https://kit.fontawesome.com/8d125d2b91.js" crossorigin="anonymous"></script>
11 </head>
12
13 <body>
14
15     <div class="container">
16         <h1 class="mt-3">OPCIONES</h1>
17         <hr class="p-1 m-1">
18         <nav class="navbar navbar-expand-sm">
19             <ul class="navbar-nav">
20                 <li class="nav-item">
21                     <a class="nav-link link-primary" href=".controller/index_category.php">Categorías</a>
22                 </li>
23                 <li class="nav-item">
24                     <a class="nav-link link-primary" href=".controller/index_product.php">Productos</a>
25                 </li>
26                 <li class="nav-item">
27                     <a class="nav-link link-primary" href=".controller/index_rol.php">Roles</a>
28                 </li>
29                 <li class="nav-item">
30                     <a class="nav-link link-primary" href=".controller/index_supplier.php">Proveedores</a>
31                 </li>
32             </ul>
33         </nav>
34     </div>
35
36     <!-- Enlace a los archivos JavaScript de Bootstrap
37     <script src="https://ajax.googleapis.com/ajax/libs/
38     <script src="https://cdnjs.cloudflare.com/ajax/lib/
39     <script src="https://maxcdn.bootstrapcdn.com/bootstrap/5.2.2/js/bootstrap.min.js">
40
41 </body>
42
43 </html>

```

☰ ⌚ Reloj - □ ×

⌚ **6:45** Hora local
 21/02/2023

En Ejecución

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

localhost/DES/tienda_animales3/index.php

OPCIONES

Categorías Productos Roles Proveedores

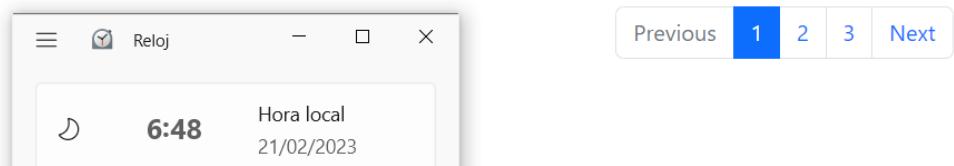


Si hacemos clic en Productos

Productos

Añadir

Nombre ↓	Precio ↓	Stock ↓	
Dudi Pet Cortauñas y Lima para Perro y Gato	22.86	129	
Royal Canin Diabetic	17.63	136	
LYL Etiquetas de identificación de Mascotas de Acero Inoxidable	22.61	232	
Acana Classic Red	23.02	194	
TK-Pet Gallina de Peluche para perros	8.52	227	



Si hacemos clic sobre el nombre

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

	Nombre ↓	Precio ↓	Stock ↓	
	Dudi Pet Cortauñas y Lima para Perro y Gato	22.86	129	
	Royal Canin Diabetic	17.63	136	
	LYL Etiquetas de identificación de Mascotas de Acero Inoxidable	22.61	232	
	Acana Classic Red	23.02	194	
	TK-Pet Gallina de Peluche para perros	8.52	227	

#	3
Nombre	LYL Etiquetas de identificación de Mascotas de Acero Inoxidable
Precio	22.61
Stock	232
Categoría	2
Foto/s	

6:49
Hora local
21/02/2023

Si hacemos clic en la papelera, borramos este elemento

Productos	Aviso	Stock ↓
Dudi Pet Cortauñas y Lima para Perro y Gato	¡Borrado con éxito!	
Royal Canin Diabetic		
Acana Classic Red		
TK-Pet Gallina de Peluche para perros		
Disfras de muñeco diabólico		

Previous 1 2 3 Next

	I.E.S. MAR DE CÁDIZ	MODELO	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

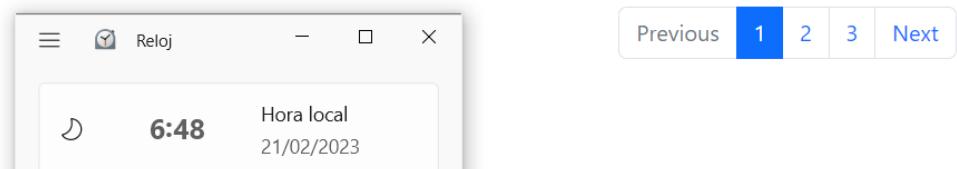
Si hacemos clic en modificar

Productos

Añadir



Nombre ↓	Precio ↓	Stock ↓	
Dudi Pet Cortauñas y Lima para Perro y Gato	22.86	129	
Royal Canin Diabetic	17.63	136	
LYL Etiquetas de identificación de Mascotas de Acero Inoxidable	22.61	232	
Acana Classic Red	23.02	194	
TK-Pet Gallina de Peluche para perros	8.52	227	



#	1
Nombre	Dudi Pet Cortauñas y Lima para Perro y Gato
Precio	2286.00
Stock	129
Categoría	5
Foto/s	

Sale esto

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

Edit Product

Id: 1

Nombre: Dudi Pet Cortauñas y Lima para Perro y Gato

Precio: 22.86

Stock: 129

Categoría: 5

Foto/s: Ninguno...vo selec.

Guardar

Reloj: 6:59 Hora local 21/02/2023

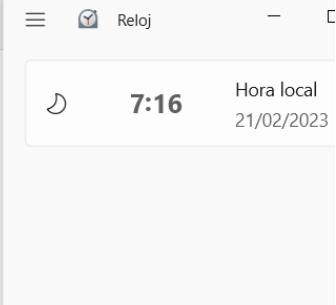
Si cambiamos el precio a 350.67 y dejamos la foto sin modificar, vemos que sigue manteniendo la misma foto.

Productos	Aviso	X
	Modificado con éxito!	
Dudi Pet Cortauñas y Lima para Perro y Gato	Reloj: 7:15 Hora local 21/02/2023	Stock ↓ 129 Editar Eliminar
Royal Canin Diabetic		17.63 Editar Eliminar
LYL Etiquetas de identificación de Mascota		22.61 Editar Eliminar
Acana Classic Red		22.02 Editar Eliminar

Vemos que funciona.

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

#	1
Nombre	Dudi Pet Cortauñas y Lima para Perro y Gato
Precio	350.67
Stock	129
Categoría	5
Foto/s	



Si le doy a añadir y NO selecciono una foto, se pone la imagen de default.

Add Product

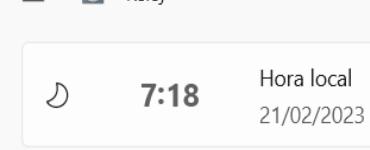
Nombre

Precio

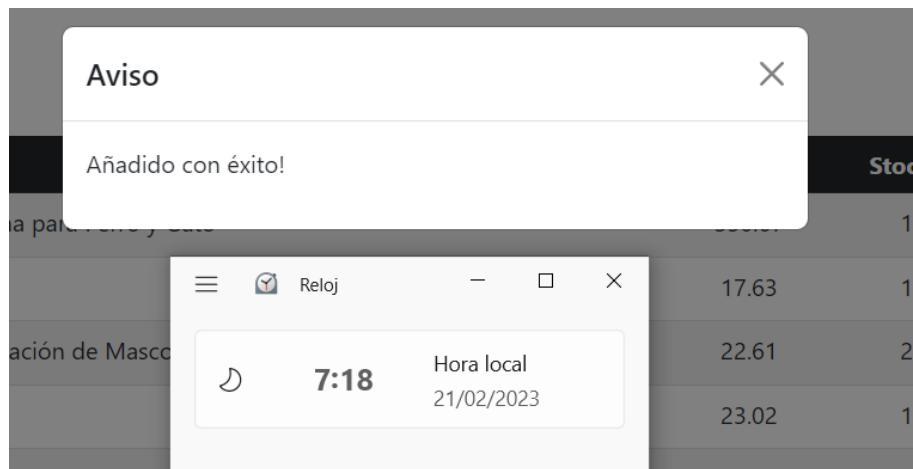
Stock

Categoría

Foto/s



	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		



#	18
Nombre	pedro
Precio	34.00
Stock	3
Categoría	1
Foto/s	

Si le doy a añadir de nuevo, pero esta vez añado una imagen

Add Product

Nombre
Chispa

Precio
10000

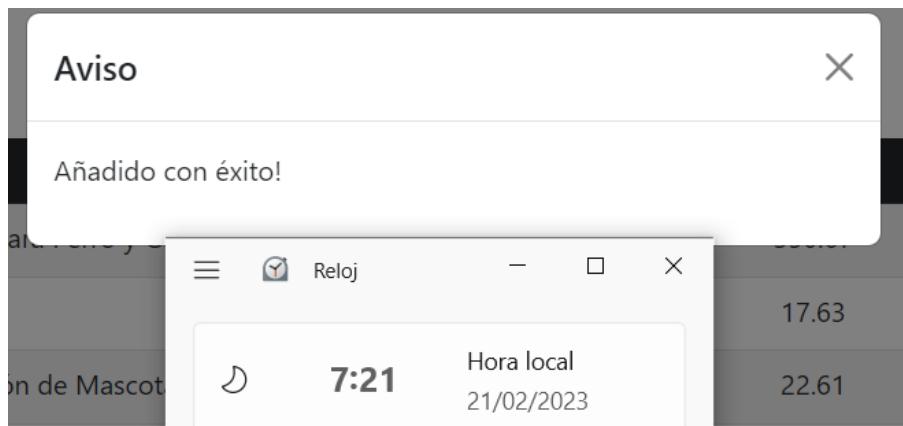
Stock
200

Categoría
3

Foto/s
Seleccionar archivo

Guardar

	I.E.S. MAR DE CÁDIZ	MODELO	
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		



Productos

[Añadir](#)

Nombre ↓	Precio ↓	Stock ↓	
pedro	10.00	4	
pedro	34.00	3	
Chispa	10000.00	200	

#	19	Reloj 7:21 Hora local 21/02/2023
Nombre	Chispa	
Precio	10000.00	
Stock	200	
Categoría	3	
Foto/s		

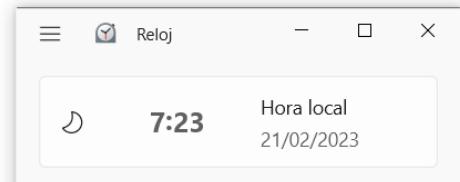
Para las otras entidades, es básicamente lo mismo. Sin embargo, hay algunas cosas más que quiero mostrar, por eso nos vamos a la página principal de proveedores.

	I.E.S. MAR DE CÁDIZ	MODELO	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo "El FSE invierte en tu futuro"
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

Proveedores

Añadir

Nombre ↓	Correo ↓	
Dapibus Quam Quis PC	et.rutrum.non@protonmail.net	
Donec Tincidunt Inc.	erat.eget@protonmail.net	
Sem Eget Foundation	orci.consectetuer@outlook.org	
Rutrum Magna LLP	malesuada@protonmail.edu	
Diam Luctus Lobortis PC	tincidunt@yahoo.net	



Previous **1** 2 Next

En proveedores, tenemos tanto correos electrónicos como números de teléfono.

Cómo hemos visto anteriormente, tengo una función que me valida y limpia los datos y un archivo js que también me valida correos y números de teléfono.

Veamos que pasa si intentamos introducir un correo electrónico erróneo.

Add Supplier

Nombre

Direccion

Telefono

Correo

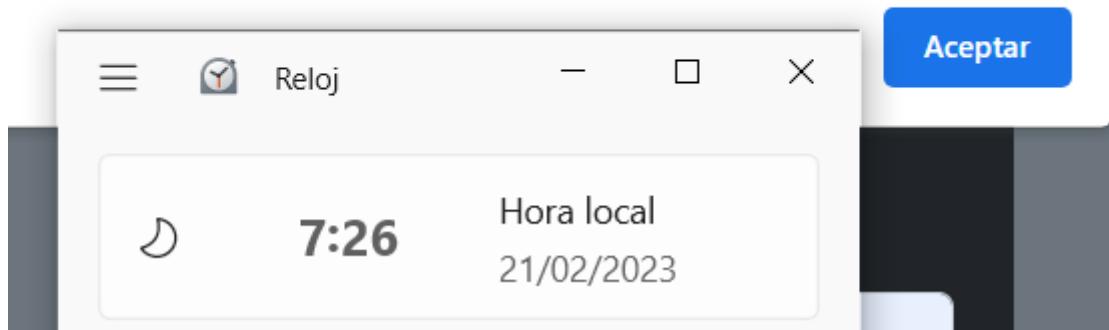
Guardar

	I.E.S. MAR DE CÁDIZ	MODELO	 Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo Unión Europea Fondo Social Europeo <i>'El FSE invierte en tu futuro'</i>
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209	
	Pág. /		

Nos salta una alerta, indicando que el formato del correo no es válido y no se manda el formulario.

localhost dice

Correo no válido. Ejemplo: example@server.com



Lo mismo pasaría con el número de teléfono.

Además, el formato válido para el número de teléfono puede tener espacios, guiones o nada en grupos de 3222, pero a la hora de ser guardados en la base de datos aparecerán así 123456789.

Add Supplier

Nombre
pedro

Direccion
C/Nardos 3

Telefono
988-29-32-98

Correo
cliente@gmail.com

Guardar

Reloj

7:29
Hora local
21/02/2023

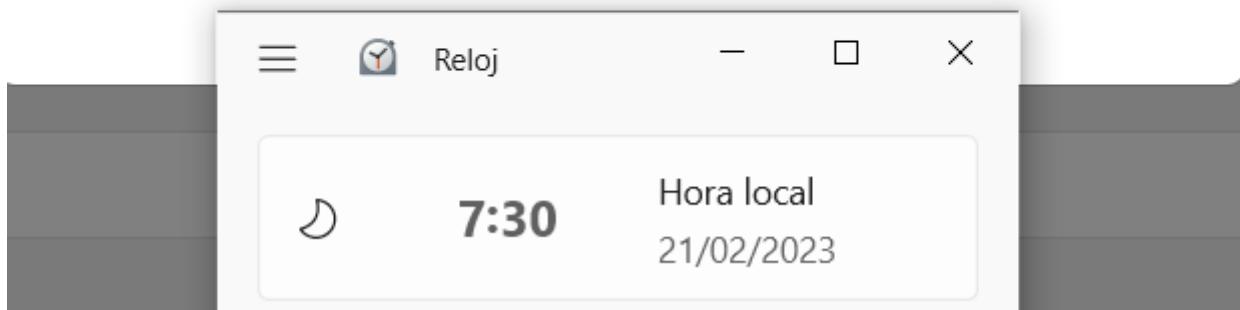
	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
		Pág. /



Aviso

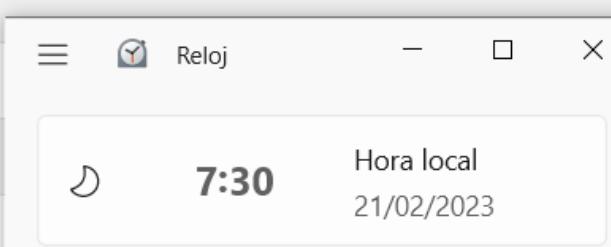


Añadido con éxito!



Proveedores

Nombre ↓		Correo ↓
pedro		cliente@gmail.com

#	11	
Nombre	pedro	
Dirección	C/Nardos 3	
Teléfono	988293298	
Correo	cliente@gmail.com	

Ahora, que pasaría si intentásemos introducir un nuevo id que no fuese válido.

Pues que el error se recogería en el log y nos saltaría un aviso.

	I.E.S. MAR DE CÁDIZ ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MODELO MOD-7501209 Pág. /	 GOBIERNO DE ESPAÑA <small>Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo</small>	 MINISTERIO DE EDUCACIÓN, CULTURA Y DEPORTE <small>Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo</small>
---	---	--	---	---

Edit Supplier

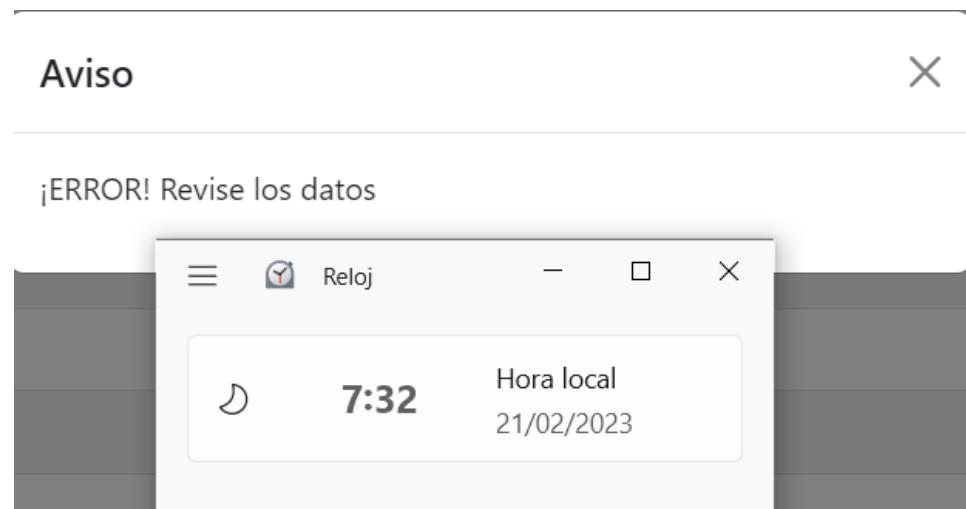
Id: 11q

Nombre:	pedro
Direccion:	C/Nardos
Telefono:	988293298
Correo:	cliente@gmail.com

Guardar

🕒
Reloj

⌚
7:32
Hora local



	I.E.S. MAR DE CÁDIZ	MODELO
	ENSEÑANZA-APRENDIZAJE PROGRAMACIÓN	MOD-7501209
		Pág. /

No cambia.

Proveedores

Nombre ↓		Correo ↓
pedro		cliente@gmail.com
#	11	
Nombre	pedro	
Dirección	C/Nardos 3	
Teléfono	988293298	
Correo	cliente@gmail.com	

☰ 🕒 Reloj - □ X

⌚ **7:33** Hora local
21/02/2023

Y en el log nos aparece

```
≡ log.log M X
tienda_animales3 > log > ≡ log.log
1  El valor de la llave "new_id" debe ser un número entero positivo.
2
```

☰ 🕒 Reloj - □ X

⌚ **7:33** Hora local
21/02/2023

Sé que no está muy bien ordenado ni optimizado el código pero he intentado hacer lo que he podido, lo siento.

 <p>I.E.S. MAR DE CÁDIZ</p>	MODELO	 <p>GOBIERNO DE ESPAÑA</p> <p>Ministerio de Educación, Cultura y Deporte</p> <p>Programa financiado por el Ministerio de Educación, Cultura y Deporte y cofinanciado por el Fondo Social Europeo</p>  <p>Unión Europea</p> <p>Fondo Social Europeo "El FSE invierte en tu futuro"</p>
	MOD-7501209	
	Pág. /	