



I E S A L B A R R E G A S

TÉCNICO SUPERIOR EN DESARROLLO DE  
APLICACIONES WEB

Departamento de Informática

PROYECTO



[Subastas Sostenibles]

**Manual Técnico**

## Índice

### Contenido

[Subastas Sostenibles] .....	1
<b>Manual Técnico.....</b>	<b>1</b>
<b>1. Introducción .....</b>	<b>3</b>
<b>2. Arquitectura de la aplicación .....</b>	<b>4</b>
<b>2.1 Frontend .....</b>	<b>8</b>
<b>2.1.1 Tecnologías usadas .....</b>	<b>8</b>
<b>2.1.2 Entorno de desarrollo .....</b>	<b>8</b>
<b>2.2 Backend.....</b>	<b>8</b>
<b>2.2.1 Tecnologías usadas .....</b>	<b>8</b>
<b>2.2.2 Entorno de desarrollo .....</b>	<b>8</b>
<b>3. Documentación técnica.....</b>	<b>9</b>
<b>3.1 Análisis .....</b>	<b>9</b>
<b>3.2 Desarrollo .....</b>	<b>12</b>
<b>3.3 Pruebas realizadas .....</b>	<b>21</b>
<b>4. Proceso de despliegue.....</b>	<b>24</b>
<b>4.1 Prueba de acceso a internet .....</b>	<b>29</b>
<b>5. Propuestas de mejoras .....</b>	<b>30</b>
<b>6. Bibliografía .....</b>	<b>31</b>

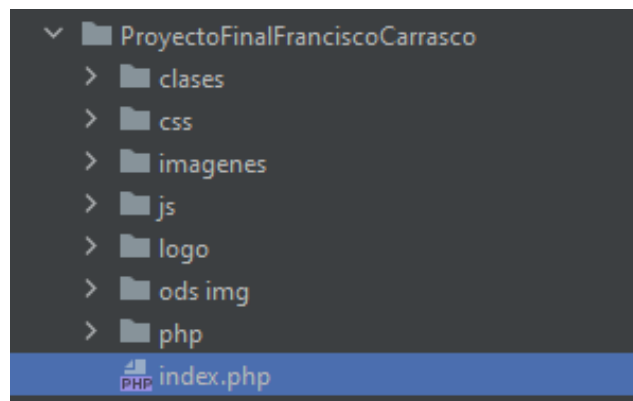
## 1. Introducción

El proyecto que he decidido realizar para el trabajo de fin de curso ha sido una página web de subastas online. Donde cualquier usuario registrado puede tanto pujar sobre diferentes objetos (arte, tecnología, propiedades, etc.) como crear subastas nuevas sobre productos que quieran vender. Tiene un fin comercial con un tinte lúdico (ya que son subastas). Esta aplicación web permitirá la inversión de un porcentaje de las ganancias en el desarrollo social a través de los Objetivos de Desarrollo Sostenible (ODS).

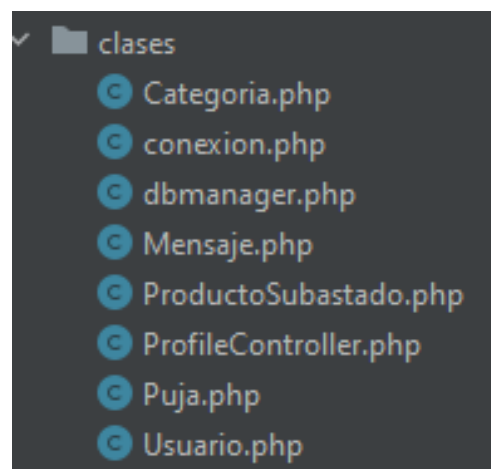
Gran parte del mercado está abarcado por aplicaciones de tiendas online, las cuales ofrecen un servicio concreto que es la venta de un producto. En los últimos años se han visto nuevas webs y aplicaciones de móvil para la venta de segunda mano. Sin embargo, hay un nicho de mercado en el cual no hay gran diversidad de webs para escoger que se adapte a los gustos del consumidor. Además, no hay ninguna aplicación web que invierta una parte de cada venta a la mejora de los Objetivos de Desarrollo Sostenible establecidos para 2030. De esta forma además de cubrir un nicho de mercado específico, se está ayudando a la mejora de la sociedad.

## 2. Arquitectura de la aplicación

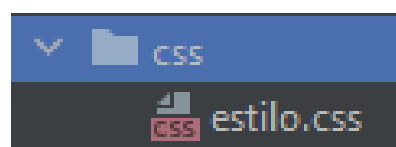
En la siguiente imagen se puede apreciar la estructura del proyecto dividida en carpetas con nombres que identifican correctamente lo que se encuentra en cada una:



La carpeta clases contiene todas las clases además de la conexión a la base de datos:

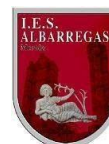


La carpeta CS contiene la hoja de estilo:



# Proyecto “Desarrollo de Aplicaciones Web”

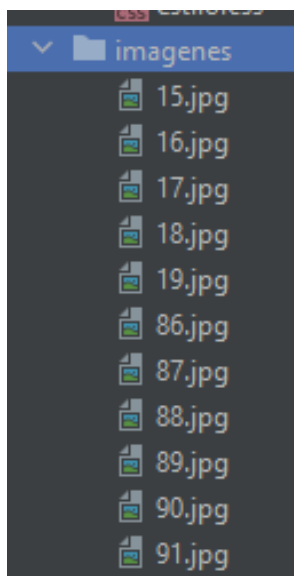
Título del Proyecto: Subastas Sostenibles



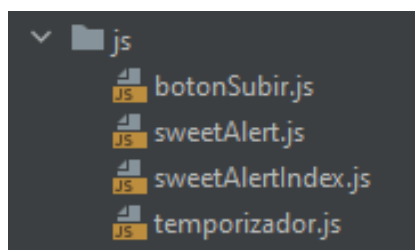
JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

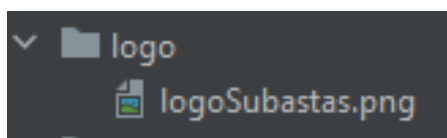
La carpeta imágenes contiene todas las imágenes que suben los usuarios de los productos que están subastando para poder mostrarlas en la aplicación web correctamente. El nombre de la imagen es el código del producto para poder ser llamado más tarde en el servidor y mostrado:



La carpeta JS contiene los archivos de JavaScript usados en la aplicación:



La carpeta logo contiene el logo:



# Proyecto “Desarrollo de Aplicaciones Web”

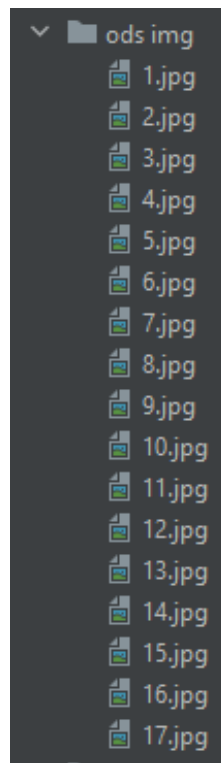
Título del Proyecto: Subastas Sostenibles



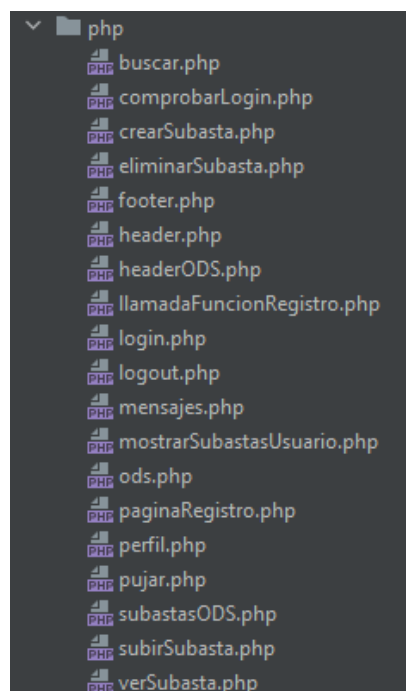
JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

La carpeta ods img contiene las imágenes de los ODS:

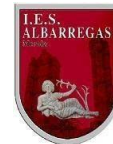


La carpeta php contiene todo el código de las páginas creadas:



# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

El logo de la aplicación ha sido creado con Photoshop. Además, contiene los 17 colores de los Objetivos de Desarrollo Sostenible. Tiene la forma de una S ya que representa Subastas y Sostenibles. También, contiene un icono de puja debido a que es la base de la aplicación y un icono de la tierra. Subastas Sostenibles pretende ayudar al desarrollo social y la ecología gracias a un porcentaje obtenido de cada producto que se invierte en los ODS:



## 2.1 Frontend

### 2.1.1 Tecnologías usadas

- HTML 5
- CSS 3
- JavaScript
- Librería Bootstrap 4.0
- Librería Font Awesome 6.2.1
- Librería SweetAlert 2

### 2.1.2 Entorno de desarrollo

He decidido usar el entorno PHPStorm ya que me gusta mucho las funcionalidades que aporta. Además, es el que he usado siempre para aplicaciones PHP por lo que me resulta más fácil su uso.

PHPStorm 2021.3.2

## 2.2 Backend

### 2.2.1 Tecnologías usadas

- PHP 7.4.26
- MariaDB 15.1

### 2.2.2 Entorno de desarrollo

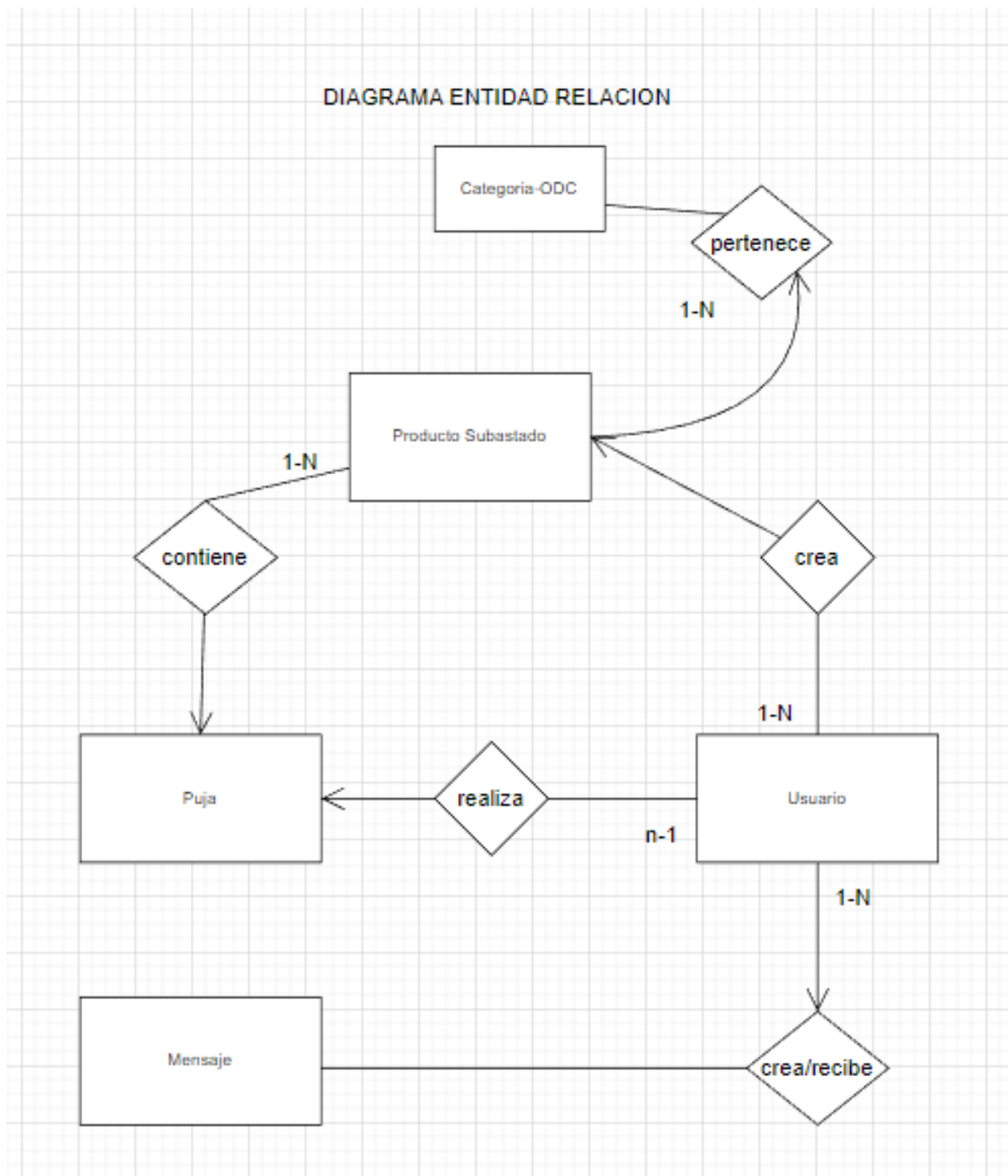
PHPStorm 2021.3.2



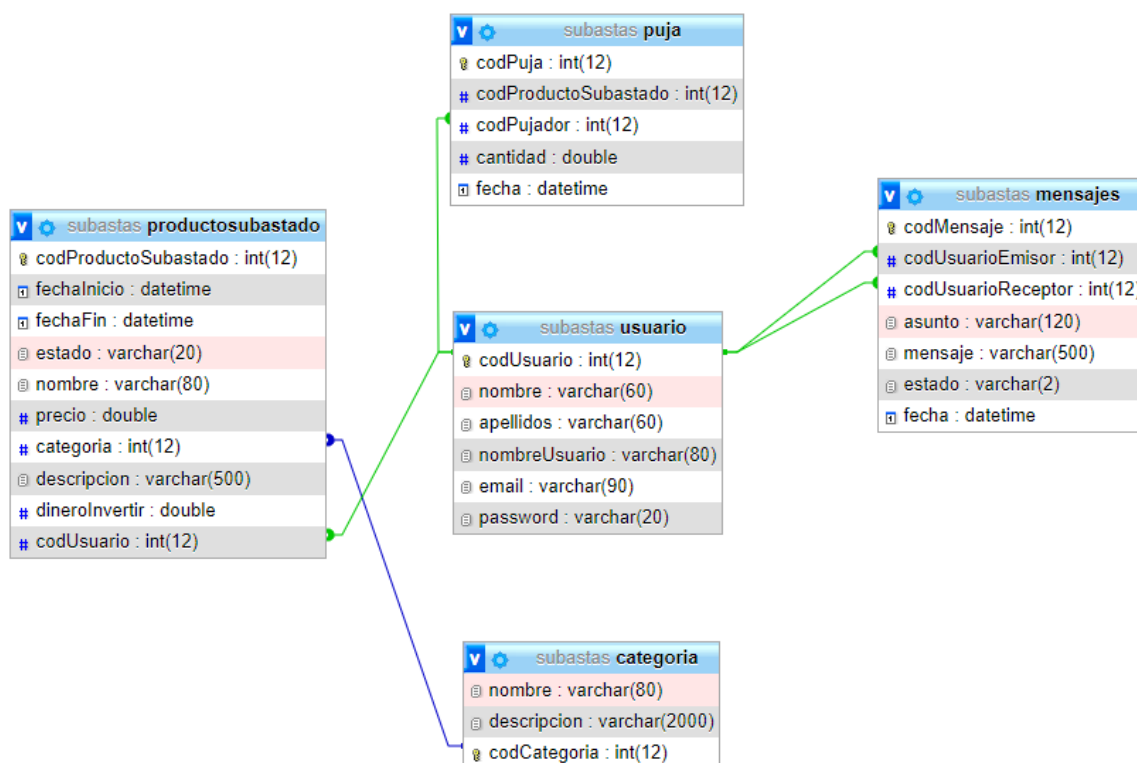
## 3. Documentación técnica

### 3.1 Análisis

Diagrama entidad relación de la base de datos. He decidido incluir :



Modelo relacional:



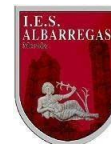
Las funcionalidades se dividen en usuarios registrados y usuario sin registrados:

- usuarios registrados pueden:
  - o loguearse
  - o consultar subastas
  - o crear subasta
  - o pujar en subasta
  - o editar datos de usuario
  - o consultar ODS
  - o enviar y recibir mensajes
- usuarios sin registrar pueden:
  - o consultar ODS
  - o registrarse
  - o Consultar Subastas sin poder hacer nada

Diagrama de casos de uso:

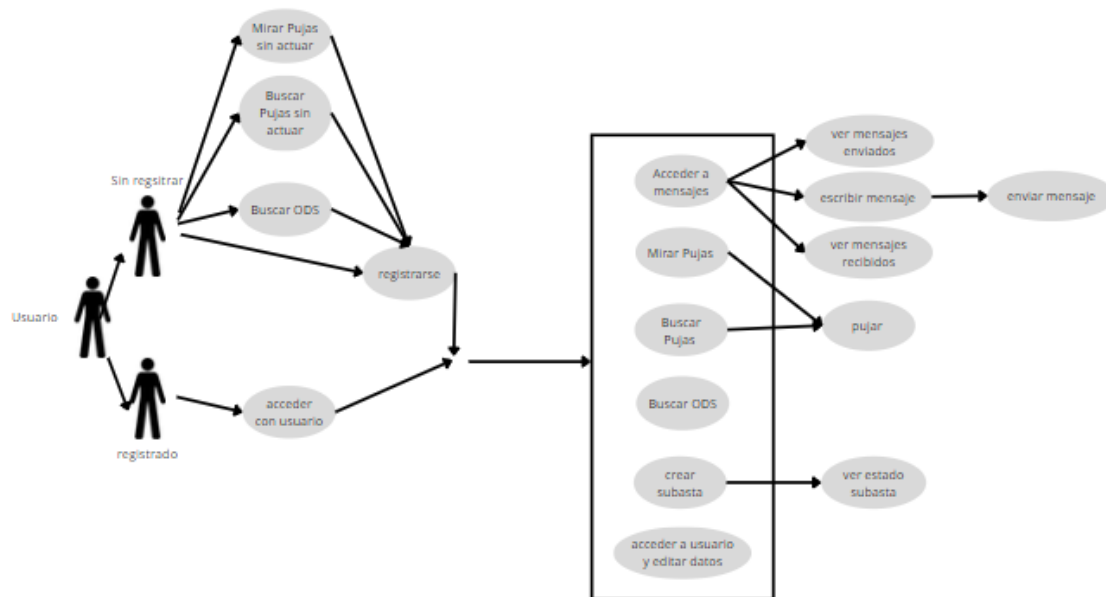
# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles

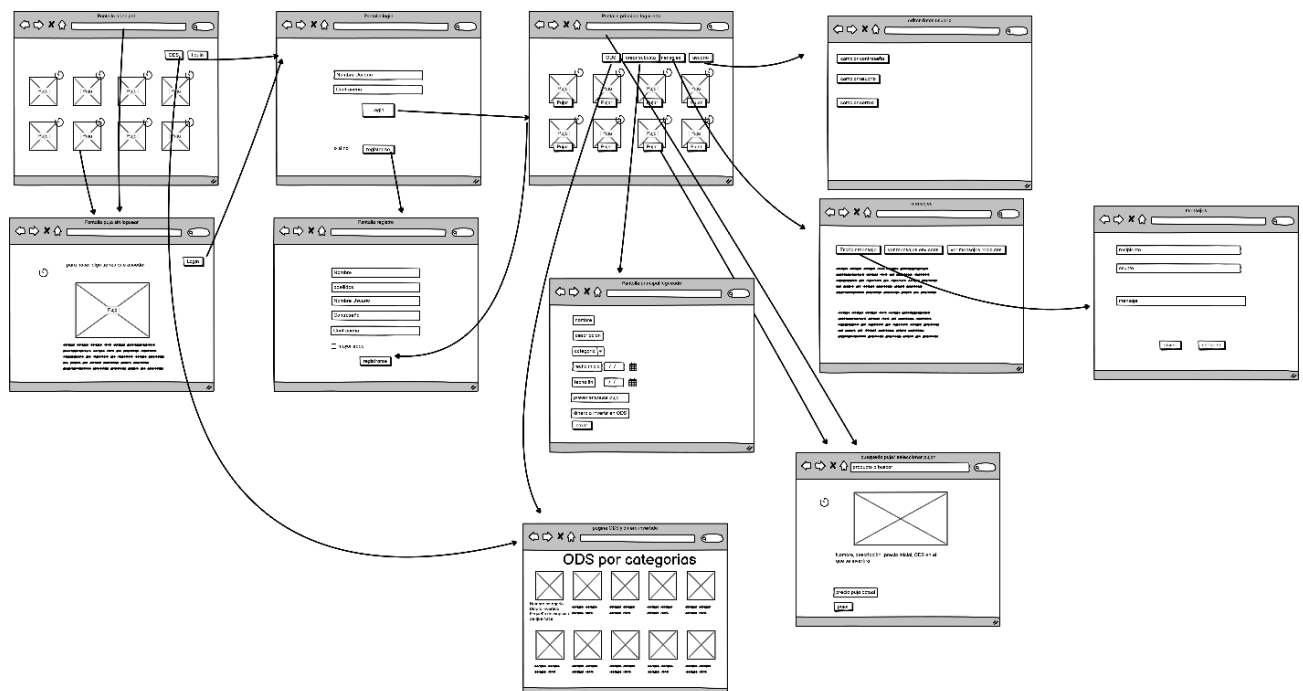


JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

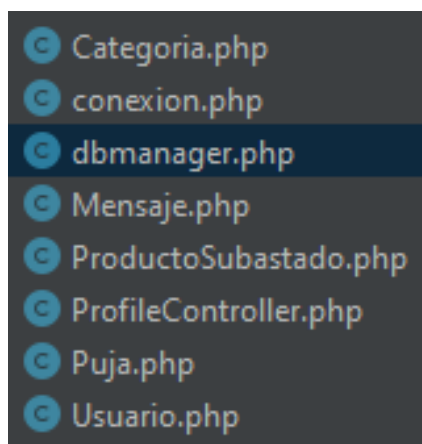


Pantallas para ver correcto funcionamiento de la aplicación:



## 3.2 Desarrollo

Se han creado objetos con los nombres de las tablas de la base de datos, además de dos archivos (conexion.php y dbmanager.php) que son los que se encargan de la conexión y obtención de los datos en sí:



La mayoría del código utilizado es PHP ya que en todo momento se obtienen datos de la Base de Datos. Sin embargo, también se ha utilizado JavaScript para poder hacer cambios en entorno cliente sin tener que refrescar las páginas constantemente y no afectar a la experiencia del usuario.

La aplicación se ha creado para que sea responsive. Se adapta al tamaño de pantalla de ordenador, de Tablet Surface Pro 7 y al iPhone XR/12. Este se ha realizado gracias a Bootstrap y CSS. En la hoja de estilos se encuentra todo el código necesario para distinguir entre pantallas de móvil, de Tablet y de ordenador.

Para comprobar que la aplicación se ajusta correctamente, se ha utilizado el modo inspeccionar que ofrece Chrome y la opción de activar emulación de dispositivo.

# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles

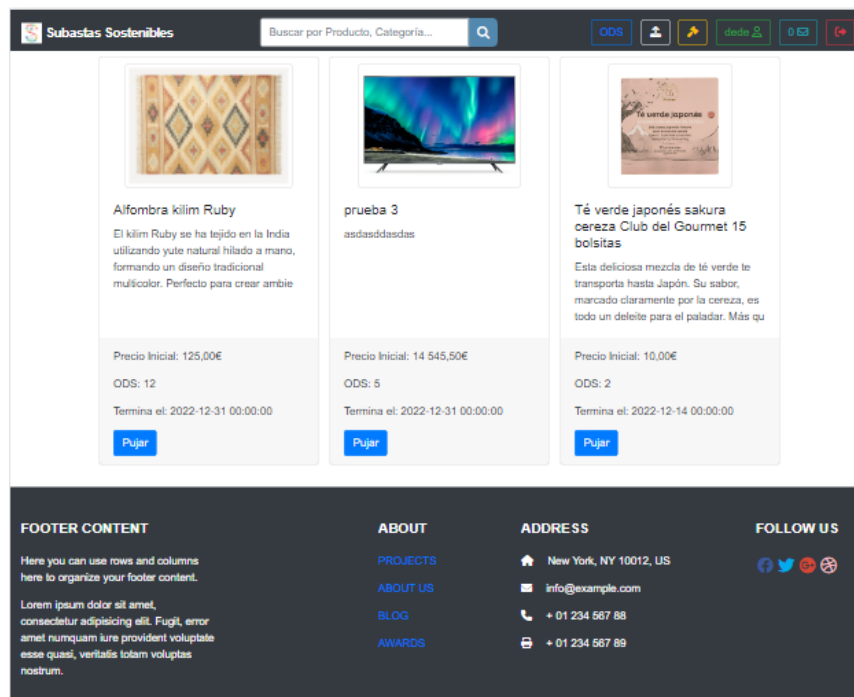


JUNTA DE EXTREMADURA

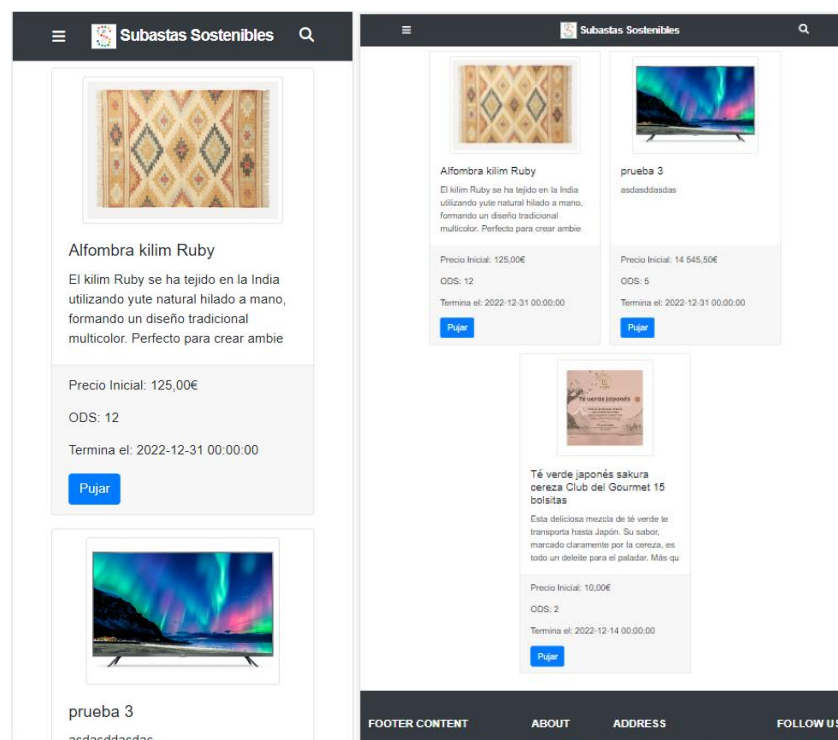
Consejería de Educación y Empleo

Se muestra la pantalla principal en los 3 tipos:

- Ordenador:

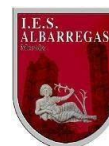


- Móvil iPhone XR y Tablet Surface Pro 7:



# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles

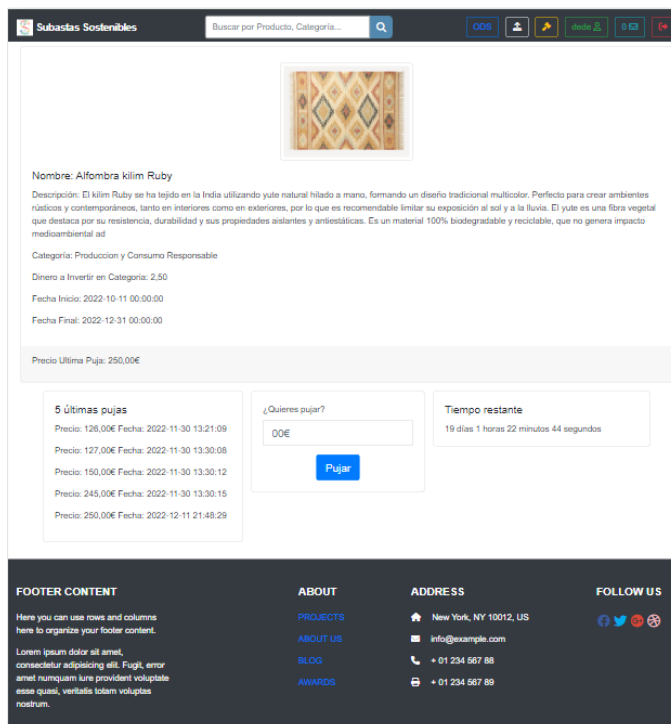


JUNTA DE EXTREMADURA

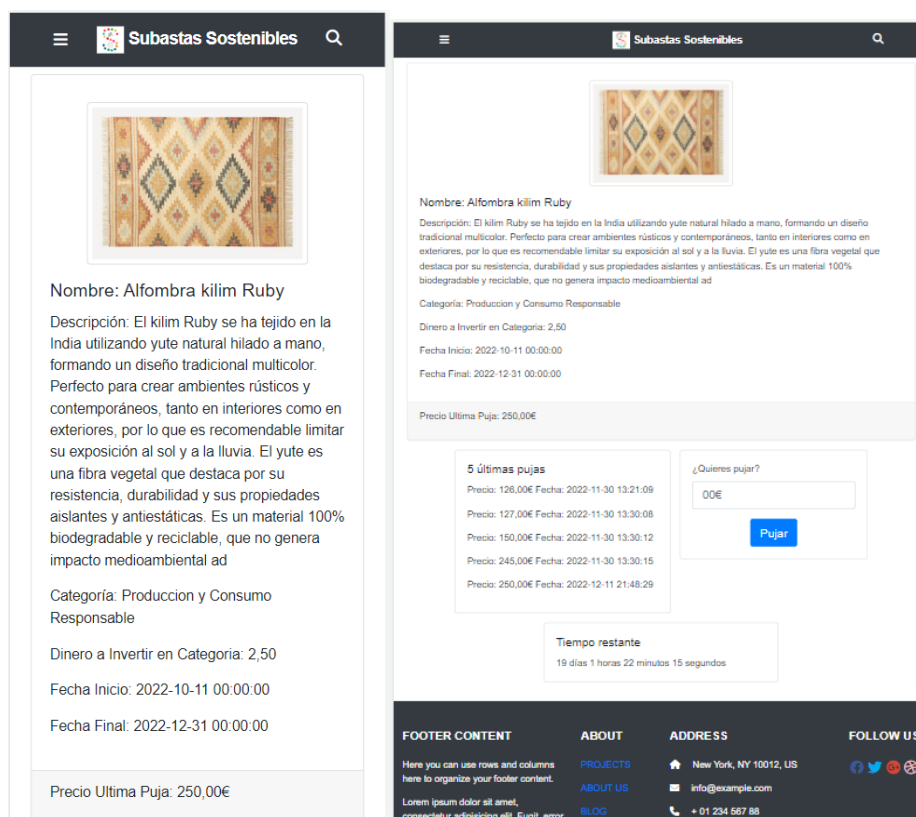
Consejería de Educación y Empleo

Se muestra la pantalla de puja en los 3 tipos:

- Ordenador:



- Móvil y Tablet:



# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Se muestra la pantalla de ODS en los 3 tipos:

- Ordenador



- Móvil y Tablet





Algunas de las pruebas que se han realizado son las de comprobar la correcta subida de datos a la base de datos a través de los formularios. Se ha realizado con PHP. Aunque también se han usado los validadores que provee HTML.

- **Al crear usuario** se comprueba por PHP que se hayan enviado todos los datos por post, se comprueba que las contraseñas coincidan y que no haya otro usuario con el mismo nombre. El email se comprueba con validación HTML además del resto de campos:

```
if (isset($_POST['nombre'], $_POST['apell'], $_POST["mail"], $_POST['usr'], $_POST['pass'])) {  
    $nombre = $_POST['nombre'];  
    $apell = $_POST['apell'];  
    $mail = $_POST["mail"];  
    $usr = $_POST['usr'];  
    $pass = $_POST['pass'];  
    $pass2 = $_POST['pass2'];  
  
    //comprobamos que la contraseña sea igual  
    if ($pass != $pass2){  
        header( header: "Location: ./paginaRegistro.php?e=3");  
    }else{  
        //compruebo que no exista el nombre de usuario  
        $usuarioComprobar= DbManager::getCodUsr($usr = $_POST['usr']);  
  
        if ($usuarioComprobar){  
            header( header: "Location: ./paginaRegistro.php?e=4");  
        }else{  
            /* LLAMAMOS A BASE DE DATOS PARA INSERTAR USUARIO */  
            $res = DbManager::insertUser($nombre, $apell, $usr, $mail, $pass);  
  
            if ($res) {  
                header( header: "Location: ./comprobarLogin.php?usr=" . $usr . "&pass=" . $pass);  
            } else {  
                //Lanza un error si no se guardan los datos en la base de datos  
                header( header: "Location: ./paginaRegistro.php?e=1");  
            }  
        }  
    }  
}  
} else {  
    /*Lanza un error */  
    header( header: "Location: ./paginaRegistro.php?e=1");  
}
```

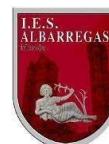
Se muestra el ejemplo de correo electrónico:

```
<div class="form-group">  
    <label for="email">Email</label>  
    <div class="input-group">  
        <div class="input-group-prepend">  
            <div class="input-group-text">@</div>  
        </div>  
        <input type="email" class="form-control form-control-lg" name="mail" id="email" placeholder="algo@ejemplo.com" required>  
    </div>  
</div>
```



# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

- Al **subir producto subastado** se tiene en cuenta que la fecha de fin sea 3 días después de la de inicio, se muestran en un desplegable las categorías disponibles, esto se hace a través de PHP y JavaScript:

Formulario para crear un producto subastado. Incluye campos para: Nombre de Producto, Descripción del Producto, Categoría (desplegable), Fecha Inicio, Fecha fin, Precio Inicial, Porcentaje que quiere invertir en el ODS (desplegable), y Subir imagen del producto (botón de selección de archivo). Botones: Subir Producto, Cancelar.

```
<div class="form-group">
  <p class="mt-2">Categoría</p>
  <select name="categorias" id="categorias" class="form-control">
    <?php
    foreach ($categorias as $cat) {
      >
      <option value="<?php echo $cat->getCodCategoria(); ?>" <?php echo $cat->getCodCategoria() . " - " . $cat->getNombre(); ?>">
      <?php
    }
    </select>
</div>
```

```
<script type='text/javascript'>
  //generamos la fecha de hoy para pasarla a fechas

  var fecha = fechaI.max = new Date().toISOString().split("T")[0];
  var fechaMaxima = new Date();
  fechaMaxima.setMonth(fechaMaxima.getMonth() + 1);

  document.getElementById('fechaI').value = fecha;
  document.getElementById('fechaI').min = fecha;
  //ya que si no da error y no deja coger otra fecha
  document.getElementById('fechaI').max = fechaMaxima;
  //fecha final se le adjunta la misma fecha
  document.getElementById('fechaF').value = document.getElementById('fechaI').value;

  //añadimos un evento a fechaI
  document.getElementById('fechaI').addEventListener("foco", myFunction);

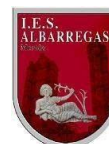
  //con esta función hacemos que se cambie la fecha mínima a 3 días después de la fecha de creación
  function myFunction() {

    var fechaTresDias = new Date(document.getElementById('fechaI').value);
    fechaTresDias.setDate(fechaTresDias.getDate() + 3);

    //hay que extraer la fecha para entrarlo en el input
    document.getElementById('fechaF').value = fechaTresDias.toLocaleDateString('en-CA');
    document.getElementById('fechaF').min = fechaTresDias.toLocaleDateString('en-CA');
  }
</script>
```

# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Se comprueba el precio, que el nombre no sea nulo y se inserta el producto y se hacen cambios para guardar la imagen en la carpeta imágenes (a través de PHP):

```
if ($precioI <= 0) {
    header( 'header: "Location: ../index.php?e=1"');
} else {
    $codUsu = $_SESSION['sesion']['codUsuario'];

    //hacemos calculos para ver cuanto dinero pasa a invertir
    $porcentaje = $_POST['cantidad'];

    if ($nombreImg["name"] != "") {

        $resultado = DbManager::insertarProductoSubastado($fechaI, $fechaF, $estado, $nombreP, $precioI, $categoria, $descripcion, $porcentaje, $codUsu);

        //recuperamos el objeto para usar el codigo para cambiarle el nombre al archivo que vamos a guardar
        $productoSubastado = DbManager::getCodigoSubastaTodo($fechaI, $nombreP, $precioI, $categoria, $codUsu);

        //guardamos la imagen
        move_uploaded_file(
            // Temp image location
            $nombreImg["tmp_name"],

            // nueva localizacion y nombre de la imagen con el codigo de producto para poder ser traído de vuelta despues
            "../imagenes/" . $productoSubastado->getCodProductoSubastado().".jpg"
        );

        if ($resultado) {
            header( 'header: "Location: ../index.php?b=1"');
        } else {
            header( 'header: "Location: ../index.php?e=1"');
        }
    } else {
        header( 'header: "Location: ../index.php?e=2"');
    }
}
```

- Al **pujar** se comprueba que la puja no sea mejor a la anterior. Si no hay puja se comprueba que no sea menor al precio mínimo establecido por el usuario que ha creado la subasta:

```
//comprobamos si la puja es inferior o no
if (isset($pujaAlta)){
    //comprobamos que la cantidad sea mayor para insertar nueva puja
    if ($cantidad > $pujaAlta->getCantidad()) {
        $subirPuja = DbManager::insertarPuja($codigoSubasta, $codPujador, $cantidad, $fecha);

        //si se sube la puja cambiamos el precio a subasta
        if ($subirPuja) {
            $precioNuevo = $pujaAlta->getCantidad();
        }
    } else {
        ?>
        <div class="alert alert-warning text-center mb-5" role="alert">
            <h3>El precio de la puja tiene que ser superior</h3>
        </div>
        <?php
    }
} elseif (is_null($pujaAlta)) {
    if ($cantidad <= constant( name: "cantidadInicial")) {
        ?>
        <div class="alert alert-warning text-center mb-5" role="alert">
            <h3>El precio de la puja tiene que ser superior</h3>
        </div>
        <?php
    } else {
        $subirPuja = DbManager::insertarPuja($codigoSubasta, $codPujador, $cantidad, $fecha);
    }
}
```

# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

- También se ha ajustado el código para que no de errores de conexión a base de datos. Esto se ha hecho tanto en el archivo donde están todas las funciones de base de datos **DBmanager.php** como en los archivos que se recupera la información.

Se muestra un ejemplo con la recuperación de los productos subastados pujados para mostrarlos en `mostrarSubastasUsuario.php`.

Función de obtener las subastas pujadas para mostrarla posteriormente :

```
public static function getSubastasPujadas($codPujador)
{
    $connection = self::getConnection();
    $sql = "SELECT * FROM productosSubastado WHERE codProductoSubastado IN (SELECT codProductoSubastado FROM puja WHERE codPujador = " . $codPujador . ")";
    $result = $connection->query($sql);
    $subasta = [];
    if ($result) {
        while ($row = $result->fetch_array()) {
            $subasta[] = new ProductoSubastado($row);
        }
        return $subasta;
    }
    return "FALLO Obteniendo ProductoSubastado";
}
```

Código para comprobar que se ha obtenido algo y si no mostrar mensaje de que no hay nada:

```
//traemos los productos en los que se ha pujado
$productosPujados = DbManager::getSubastasPujadas($codigoUsuario);

//si no se han hecho pujas se muestra
if (empty($productosPujados)) {
    ?>
    <div class="alert alert-primary d-flex align-items-center" role="alert">
        <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor"
            class="bi bi-exclamation-triangle-fill flex-shrink-0 me-2" viewBox="0 0 16 16" role="img"
            aria-label="Warning:">
            <path d="M8.982 1.566a1.13 1.13 0 0 0-1.96 0L1.65 13.233c-.457 7.78.091 1.767.98 1.767h13.713c.889 0 1.438-.99.98-1.767L8.982 1.566z" data-bbox="8.982 1.566 13.713 13.233"/>
        </svg>
        <div>
            No has hecho pujas
        </div>
    </div>
</div>
<?php
}
```

- Se han hecho pruebas para que, si un usuario borra su cuenta, se borren las subastas, pujas y mensajes también y no de errores en la base de datos:

```
public static function deleteUser($codUsr)
{
    $connection = self::getConnection();
    $subastas = self::getSubastasVendedor($codUsr);

    if (count($subastas)>0) {
        foreach ($subastas as $sub) {
            $borraPujas = self::deletePujas($sub->getCodProductoSubastado());
            $delete = "DELETE FROM productosubastado WHERE codProductoSubastado = " . $sub->getCodProductoSubastado();
            $resDelete = $connection->query($delete);
        }
        if ($resDelete) {
            //borramos los mensajes enviados
            $deleteMsg = "DELETE FROM mensajes WHERE codUsuarioEmisor = " . $codUsr;
            $resDeleteMsg = $connection->query($deleteMsg);

            //borramos los mensajes recibidos
            if ($resDeleteMsg) {
                $deleteMsgReceptor = "DELETE FROM mensajes WHERE codUsuarioReceptor = " . $codUsr;
                $resDeleteMsgReceptor = $connection->query($deleteMsgReceptor);
                //borramos usuario
                if ($resDeleteMsgReceptor) {
                    $sql = "DELETE FROM usuario WHERE codUsuario = " . $codUsr;
                    $resultado = $connection->query($sql);
                    if ($resultado) {
                        return true;
                    }
                }
                return false;
            }
        }
    }
}
```

```
} else {
    $delete = "DELETE FROM puja WHERE codPujador = " . $codUsr;
    $resDelete = $connection->query($delete);
    if ($resDelete) {
        //borramos los mensajes
        $deleteMsg = "DELETE FROM mensajes WHERE codUsuarioEmisor = " . $codUsr;
        $resDeleteMsg = $connection->query($deleteMsg);
        if ($resDeleteMsg) {
            $deleteMsgReceptor = "DELETE FROM mensajes WHERE codUsuarioReceptor = " . $codUsr;
            $resDeleteMsgReceptor = $connection->query($deleteMsgReceptor);
            if ($resDeleteMsgReceptor) {
                $sql = "DELETE FROM usuario WHERE codUsuario = " . $codUsr;
                $resultado = $connection->query($sql);
                if ($resultado) {
                    return true;
                }
            }
            return false;
        }
    }
}
return false;
}
```

Respecto al HTML, se ha utilizado para dar la estructura a las páginas webs y poder mostrar los contenidos.

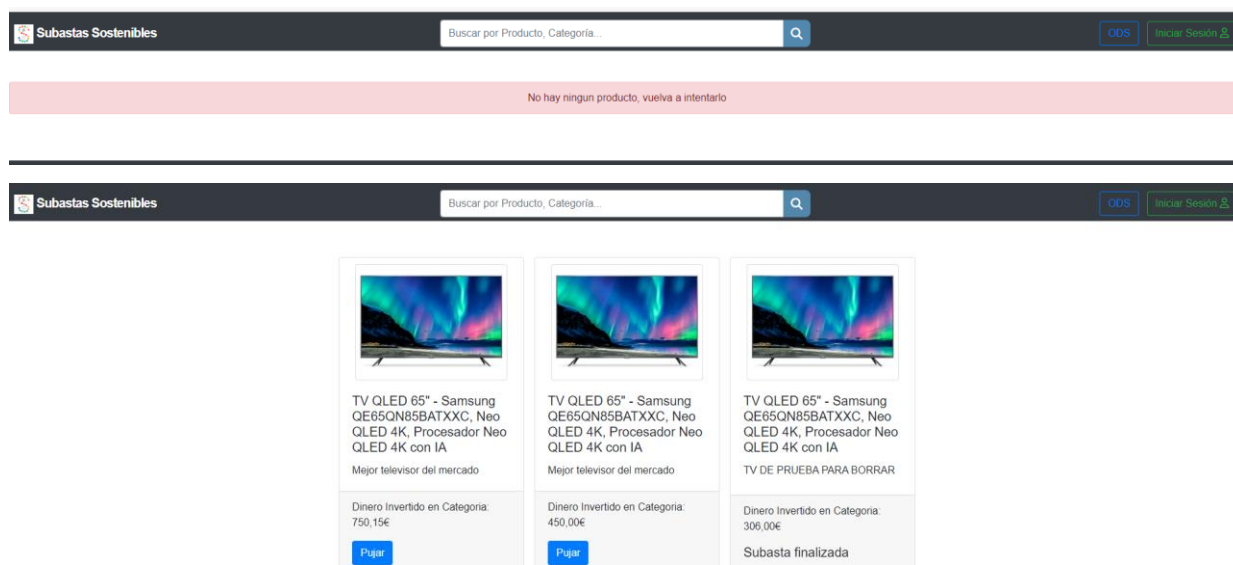
## 3.3 Pruebas realizadas

Se han hecho comprobaciones constantemente para que la aplicación tenga los mínimos fallos. Las más importantes son las siguientes:

Casos de Uso	Resultado	Resultado
me registro y da error	me reenvia al registro de nuevo y pone que ha habido un error	OK
me registro y funciona	me redirige a pagina principal de subastas	OK
intento logearme y falla	me vuelve a salir la misma pagina con error	OK
me logueo	me redirige a pagina principal de subastas	OK
accedo a página que no existe	redireccion a pagina de error 404	OK
click en boton ODS	muestra la pagina ODS	OK
click en boton editar usuario	muestra pagina de edicion de parametros de usuario	OK
click en boton mensajes	muestra pagina de mensajes	OK
click en boton enviar mensaje	muestra pagina de redaccion de mensajes	OK
click en boton mostrar mensajes recibidos	muestra mensajes recibidos	OK
click en boton mostrar mensajes enviados	muestra mensajes enviados	OK
click en boton crear subasta y error	sale mensaje de error arriba en la pagina en rojo	OK
click en boton crear subasta sin error	se genera la subasta y redirige a pagina de la subasta	OK
click en boton pujar	dirige a pagina de la subasta para pujar	OK
meter cantidad de puja a tiempo	se efectua	OK
meter cantidad de puja a destiempo	sale error de puja finalizada	OK
finaliza contador de puja	se envia mensaje a todos los pujadores con resultado, si es ganador se informa de ello	OK
buscar en barra de busqueda por string sin resultado	se muestra una pagina donde pone que no hay resultado	OK
buscar en barra de busqueda por string con resultado	se muestran los resultados de busqueda de puja y muestra por nombre	OK

Se muestran algunos ejemplos de las pruebas realizadas:

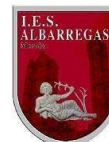
- La búsqueda de productos por nombre se ha controlado para que muestre un mensaje si no hay producto y muestre los productos si los hay:



- Se han hecho pruebas para que los productos que se muestran en pantalla son los que están como activos en ese mismo día. Si tienen un día de más se

## Proyecto “Desarrollo de Aplicaciones Web”



Título del Proyecto: Subastas Sostenibles




JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

pasan a productos finalizados y se envía un mensaje a cada pujador de que ha terminado la puja del producto. Si la subasta deja de estar activa se le quita el botón de pujar:

	
<p><b>Alfombra kilim Ruby</b></p> <p>El kilim Ruby se ha tejido en la India utilizando yute natural hilado a mano, formando un diseño tradicional multicolor. Perfecto para crear ambiente</p> <p>Dinero Invertido en Categoría: 1,25€</p> <p>Subasta finalizada</p>	<p><b>Alfombra kilim Ruby</b></p> <p>El kilim Ruby se ha tejido en la India utilizando yute natural hilado a mano, formando un diseño tradicional multicolor. Perfecto para crear ambiente</p> <p>Dinero Invertido en Categoría: 2,46€</p> <p>Pujar</p>

- Al igual que se ha comprobado que un subastador no pueda pujar en su propia subasta:



eReader - Kobo Clara 2E,  
Para eBook, 6", 16 GB,  
300 ppp, 1448 x 1072, E-  
Ink, Az

Si tienes dispositivos que ya no usas, podemos darles una segunda vida. Ven a tu tienda MediaMarkt con tu antiguo aparato tecnológico o te lo pasa

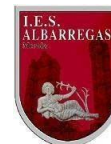
Dinero Invertido en Categoría:  
6,03€

No puedes pujar

- Se han hecho pruebas para que un usuario sin registrar no pueda hacer nada dentro de una subasta que esta activa:

# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Subastas Sostenibles

Buscar por Producto, Categoría...

Q

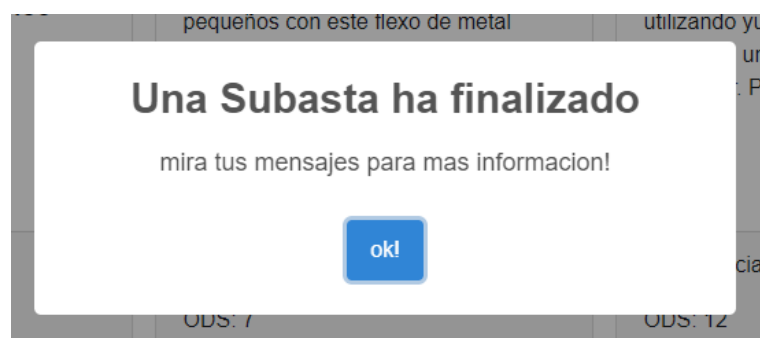
ODS

Iniciar Sesión

No puedes pujar sin registrarte o estar conectado

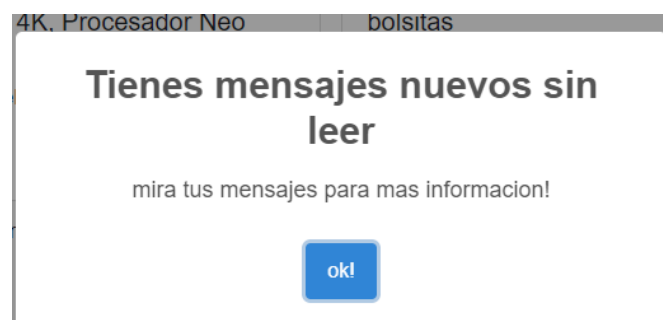
Iniciar Sesión

- Se han hecho pruebas para comprobar que aparece un pop-up y se envía un mensaje al usuario cuando se termina una subasta. Al ganador se le envía un mensaje más avisándole de que ha ganado:



Enviado por: admin	Asunto: Subasta Ganada	Enviado: 2022-12-12 05:36:06	Mensaje: Enhorabuena, has ganado la puja con nombre= Té verde japonés sakura cereza Club del Gourmet 15 bolsitas		
Enviado por: admin	Asunto: Subasta Finalizada	Enviado: 2022-12-12 05:36:06	Mensaje: Ha finalizado la subasta con nombre= Té verde japonés sakura cereza Club del Gourmet 15 bolsitas		

- Se han hecho pruebas para comprobar que si un usuario tiene un mensaje sin leer se le envíe un pop-up cada vez que accede a la página principal

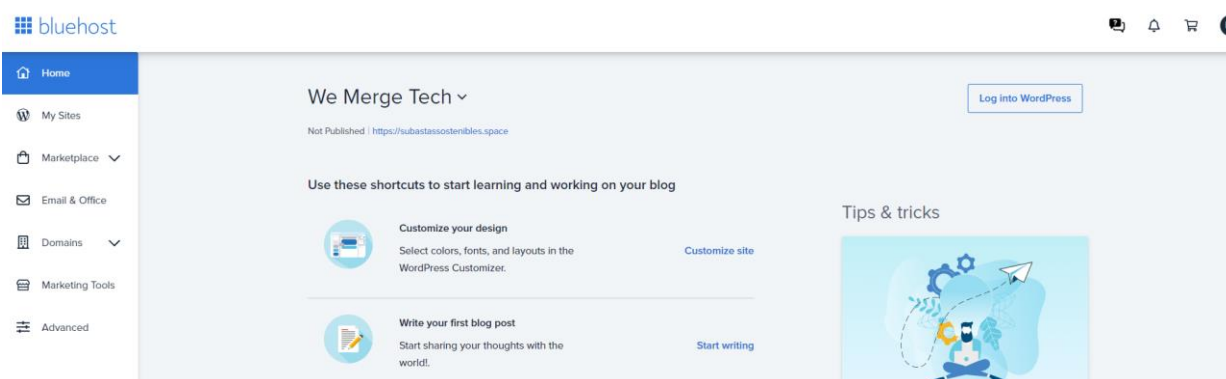




## 4. Proceso de despliegue

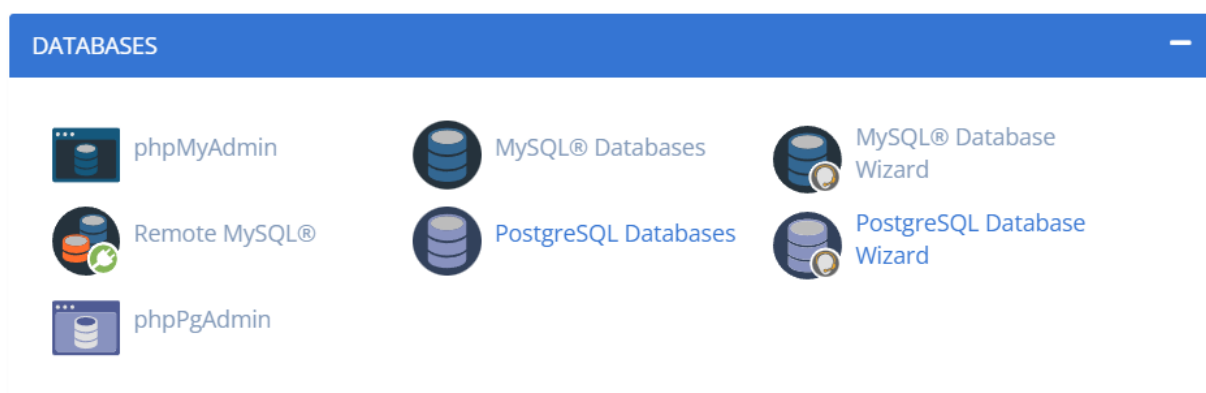
Para poder desplegar la aplicación, se va a hacer una demostración con Blue Host de cómo se haría, puede variar en función del Host.

Hay que acceder con usuario y contraseña al Host. Es necesario que se tenga un dominio para poder acceder a la aplicación cuando esté desplegada.



Una vez dentro hay que hacer clic en el menú izquierdo donde pone avanzado:

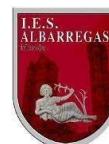
Hay que buscar el apartado de bases de datos para poder crear la base de datos que necesitamos. Se hace clicando en el botón MySQL Databases:





# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Se crea la base de datos y el usuario y después se añade el usuario a la base de datos como se muestra en la imagen c:

## Create New Database

New Database:

wemerget\_

Create Database

a

## MySQL Users

Add New User

Username

wemerget\_

Password

Password (Again)

Strength ⓘ

Very Weak (0/100)

Password Generator

Create User

b

## Add User To Database

User

wemerget\_WPBG9

Database

wemerget\_WPBG9

Add

c

Una vez hecho todo esto se puede acceder a PHPmyAdmin:

Advanced

File Manager

Databases

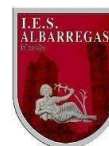
phpMyAdmin

FTP

SSH

# Proyecto “Desarrollo de Aplicaciones Web”

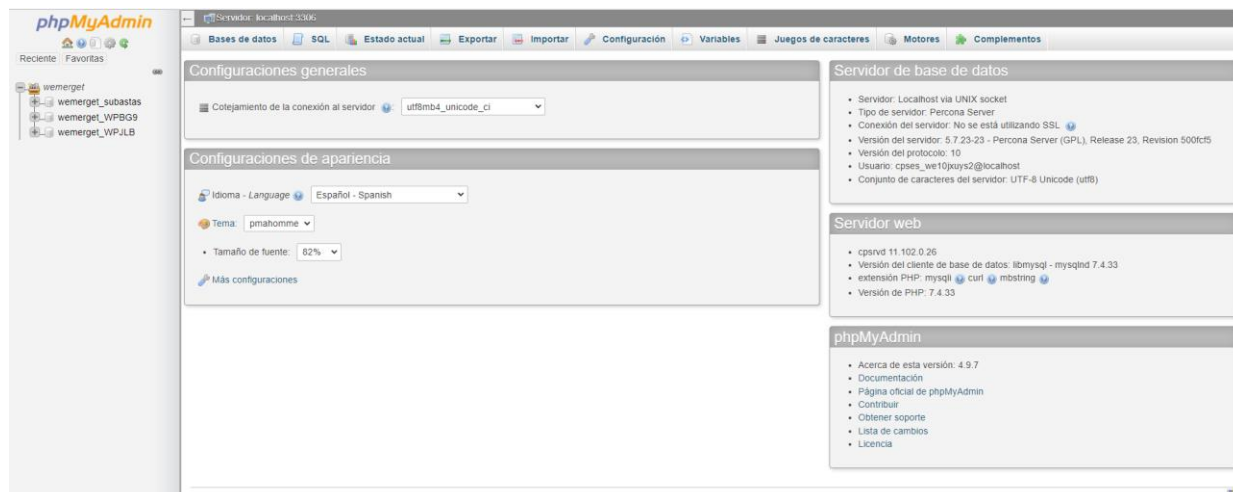
Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Dentro tenemos que elegir la base de datos que hemos creado y dar a importar:



Se tendrá que importar el archivo **provisto** que termina en .sql que contiene todas las tablas con los datos y se crearán automáticamente:

script creacion final.sql

## Importando al servidor actual

### Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.

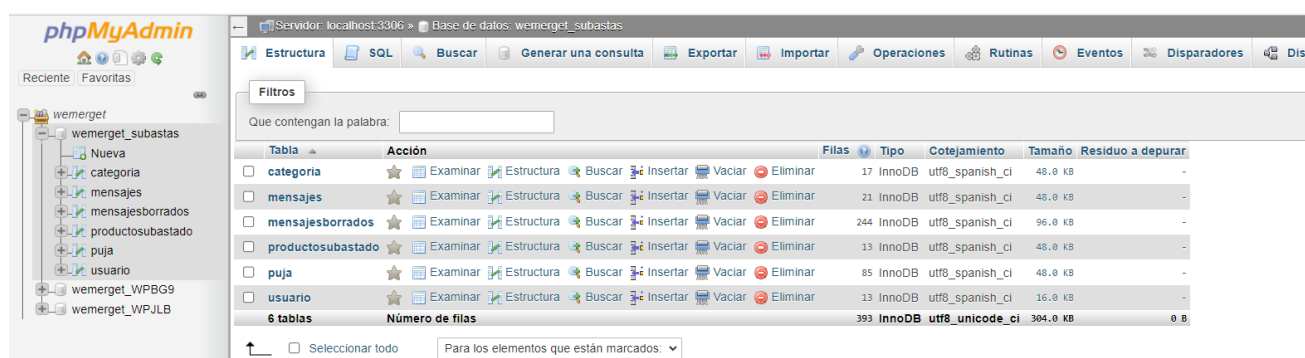
Un archivo comprimido tiene que terminar en **[formato].[compresión]**. Por ejemplo: **.sql.zip**

Buscar en su ordenador:  No se ha seleccionado ningún archivo (Máximo: 512MB)

También puede arrastrar un archivo en cualquier página.

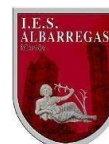
Conjunto de caracteres del archivo:

Se muestra el resultado:



# Proyecto “Desarrollo de Aplicaciones Web”

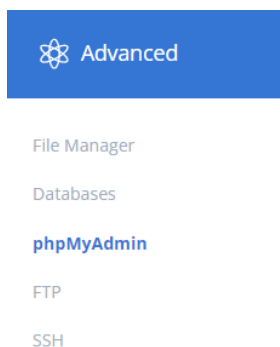
Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

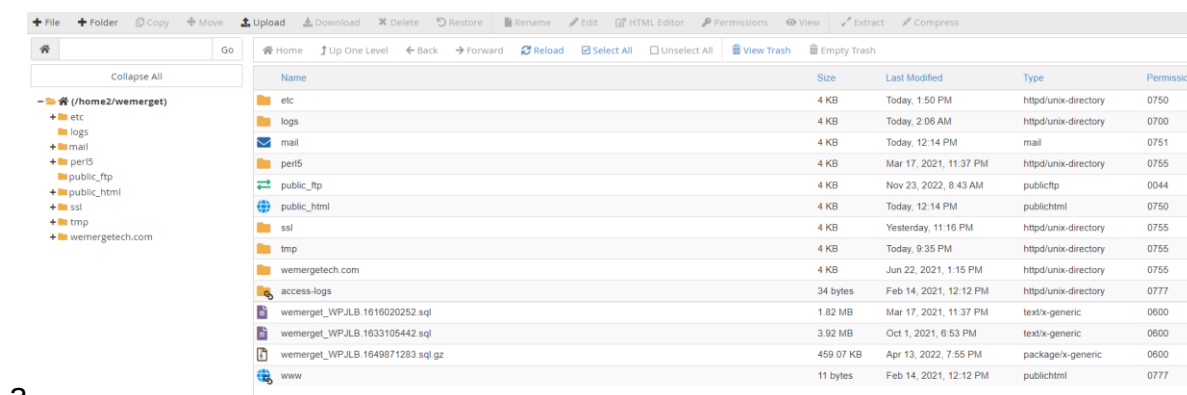
Consejería de Educación y Empleo

Una vez este todo lo relacionado con la base de datos **establecido**, se accede a File Manager



Ahí se accede a la carpeta `public_html` y es donde hay que subir el zip con todo el proyecto y descomprimirlo, en la imagen b lo muestro descomprimido:

Se muestra donde hay que acceder



a

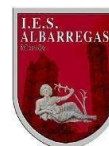
Se muestra todo descomprimido dentro de `public_html`



b

# Proyecto “Desarrollo de Aplicaciones Web”

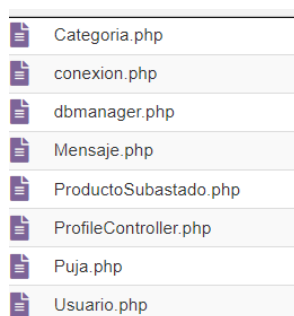
Título del Proyecto: Subastas Sostenibles



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Finalmente tenemos que acceder al archivo donde se cambian las variables de conexión a la base de datos y poner el nombre de base de dato, nombre de usuario y contraseña, en este caso se encuentra en clases->**conexion.php**:



Para abrir el archivo hay que clicar con el botón derecho y darle a editar y cambiamos los datos dichos anteriormente:



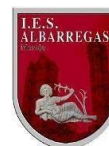
Una vez hecho todo esto, si se tiene activo el dominio, se podrá acceder directamente a la aplicación.

La aplicación funciona sobre un servidor BlueHost como se ha dicho anteriormente. En este caso el dominio es:

<https://www.subastassostenibles.space/subastasSostenibles>

# Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: Subastas Sostenibles

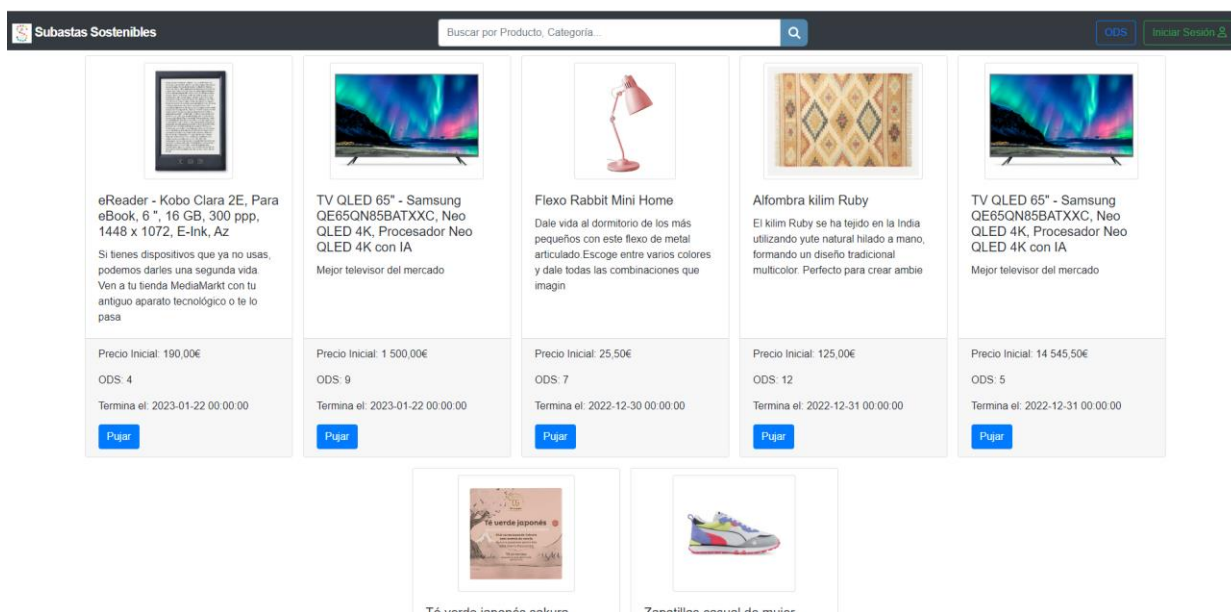


JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

## 4.1 Prueba de acceso a internet

Accediendo al link anterior, el usuario aparecerá en la pantalla principal sin las funcionalidades de usuario registrado:



Si se requiere más información del uso de la aplicación, acceda al manual de usuario donde está todo detallado correctamente.

## 5. Propuestas de mejoras

La aplicación funciona y cumple con lo que en un principio se quería conseguir, pero aun así tiene varias cosas que pueden mejorar claramente:

- Puede simular una cartera para hacer los pagos.
- Puede añadirse una pasarela de pagos real.
- Añadir lista de contactos y mostrarlos para poder enviarles mensajes.
- Poder subir más de una imagen por producto subastado.
- Validar si la puja es más alta o baja con JavaScript en vez de php para no tener que recargar la página y que la experiencia de usuario sea más fluida.
- Se puede hacer bajo algún framework como angular, el cual haría la aplicación más robusta.
- Se podría implementar seguridad a nivel del servidor.
- Se podría implementar un servicio post venta a través de una página de contacto.

## 6. Bibliografía

- Foro de dudas:

[Stack Overflow - Where Developers Learn, Share, & Build Careers](#)

- Librería para customizar JavaScript>

[SweetAlert2 - a beautiful, responsive, customizable and accessible \(WAI-ARIA\) replacement for JavaScript's popup boxes](#)

- Manual Bootstrap:

[Introduction · Bootstrap \(getbootstrap.com\)](#)

- Manual PHP:

[PHP: Manual de PHP - Manual](#)

- Manual JavaScript:

[Guía de JavaScript - JavaScript | MDN \(mozilla.org\)](#)

- Manuales de ayuda para HTML, CSS, JS y SQL:

<https://www.w3schools.com/>

- Fuentes Awesome:

[Font Awesome](#)

- Repositorio con servicio GIT:

[GitHub](#)

- Programas para hacer diagramas:

[diagrams.net](#)

[Balsamiq. Rapid, Effective and Fun Wireframing Software | Balsamiq](#)

- Servidor para Desarrollo local:

[XAMPP Installers and Downloads for Apache Friends](#)

- Entorno de desarrollo:

[PhpStorm: IDE de PHP y editor de código de JetBrains](#)