

Curso: SEPE10-22: Desarrollo web BackEnd

Docente: David Alcolea

PLA 5_ DETALLE Y DESCRIPCIÓN

Nombre del PLA: Utilizar un formulario para enviar correos y subir archivos al servidor

Objetivos del PLA:

- Aprender a utilizar la función nativa `mail()` de php para envío de correos o, alternativamente, la librería `phpmailer`
- Subir archivos de cualquier tipo al servidor validando su tipo y tamaño
- Escritura y lectura de ficheros secuenciales
- Manejo de fechas, cadenas y números
- Reforzar el uso de funciones para organizar el código
- Reforzar el uso de variables globales y constantes
- Reforzar las técnicas de tratamiento de errores con `try...catch`
- Reforzar el uso de las estructuras de programación `if` y `for`

Competencias asociadas al PLA:

Competencias técnicas	Soft Skills
<ul style="list-style-type: none"> • Envío de correos de forma nativa o con uso de librerías externas • Subir y validar archivos al servidor • Escritura y lectura estructurada de ficheros • Uso de fechas, cadenas y números 	<ul style="list-style-type: none"> • Resolución de problemas • Interpretar requerimientos • Gestionar un proyecto • Búsqueda, gestión y uso de la información

Calendario y horas de dedicación:

Del 20/10/22 al 25/10/22

Síncronas	Webinars	0
	Tutorías presenciales	0
Asíncronas		15
Presenciales		
Totales		15

Modalidad PLA: Ejercicio independiente

Instrucciones que recibirá el alumno para resolver el reto:

En este reto utilizaremos un formulario ya confeccionado para añadirle la funcionalidad de envío de correos electrónicos, selección de ficheros para subirlos al servidor y escritura estructurada en ficheros.

Instrucciones metodológicas para la entrega de los retos:

Este PLA consta de una actividad consistente en la entrega de un formulario al que añadiremos las siguientes operativas:

- Recoger y validar los datos para confeccionar un correo electrónico
- Seleccionar y validar ficheros para su subida al servidor
- Escritura de un fichero de tipo log con información de los correos enviados desde el formulario
- Manipulación de fechas, cadenas de texto y números

CONTACTO

Los campos marcados con * son obligatorios.

Nombre: * Michael Corleone

Email: * peter@mail.com

Teléfono: 77777774

Mensaje: *

Te haré una propuesta que no podrás rechazar

Seleccionar archivo Ningún archivo seleccionado

Enviar

Correo enviado correctamente

Fecha: 28-09-2021
Remitente: Michael Corleone
Teléfono: 77777774
Mensaje: Te haré una propuesta que no podrás rechazar
Código consulta: F53062
Nombre fichero: 9.png

El formato de entrega de la actividad es el siguiente:

Pla5_tu_nombre_tu_apellido.zip

La entrega del fichero será a través de la aplicación Moodle del curso.

Las actividades son individuales y intransferibles.

La resolución de dudas o comentarios se podrán realizar a través de las herramientas de comunicación de la aplicación Moodle del curso

La actividad se entregará antes de la fecha límite de entrega.

Requerimientos de usuario (PLA 05: Correo y ficheros php)

Dotar de funcionalidad al formulario **contacto.php** que encontraréis en recursos del PLA para añadir las operativas que se describen en forma de requerimientos de usuario.

CONTACTO

Los campos marcados con * son obligatorios.

Nombre: * Michael Corleone

Email: * peter@mail.com

Teléfono: 777777744

Mensaje: *

Te haré una propuesta que no podrás rechazar

Seleccionar archivo Ningún archivo seleccionado

Enviar

Correo enviado correctamente

Fecha:	28-09-2021
Remitente:	Michael Corleone
Teléfono:	777777744
Mensaje:	Te haré una propuesta que no podrás rechazar
Código consulta:	F53062
Nombre fichero:	9.png

Especificaciones del usuario:

Al cumplimentar el formulario y pulsar sobre el botón de enviar se solicita realizar las siguientes tareas:

1.- Envío de un correo electrónico con los datos recuperados del formulario

- Al pulsar enviar se confeccionará y enviará un correo electrónico a la dirección ficticia destinatario@mail.com con los datos y el formato que se especifican a continuación:

Asunto del correo: "Correo desde formulario de contacto"

Texto del correo (debe aparecer tal como se muestra a continuación):

Fecha: fecha del día en curso

Remitente: Nombre recuperado del formulario (obligatorio)

Teléfono: Teléfono recuperado del formulario (opcional)

Mensaje: Comentario recuperado del formulario (obligatorio)

Código consulta: código de la consulta confeccionado de forma automática;

Nombre fichero: nombre del fichero seleccionado (opcional)

- Los datos del correo se mostrarán en una caja situada debajo del formulario (`class='correo'`) en el documento [html](#)

Fecha: 28-09-2021
Remitente: Michael Corleone
Teléfono: 777777744
Mensaje: Te haré una propuesta que no podrás rechazar
Código consulta: F53062
Nombre fichero: 9.png

- Se validarán que los datos obligatorios que se especifican en el texto del correo se encuentren informados y tengan el formato correcto. Se mostrará un mensaje de error en caso contrario. Ejemplo:

Seleccionar archivo	Ningún archivo seleccionado
Enviar	
Email no informado o mal formateado	

- Si el usuario selecciona un fichero para subir al servidor y éste no tiene la extensión correcta o supera el tamaño límite NO se enviará el correo ni se mostrarán los datos en la caja situada debajo del formulario (ver apartado siguiente)
- Cada consulta que realiza el usuario al rellenar el formulario generará un código alfanumérico de 6 caracteres obtenidos carácter a carácter de forma aleatoria a partir de la siguiente secuencia:

A1B2C3D4E5F6G7H8I9J0K1L2M3N4O5P6Q7R8S9T0U1V2W3X4Z5

Código consulta: F53062

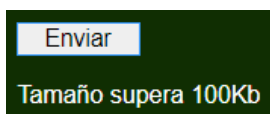
- Si el correo se envía correctamente se mostrará un mensaje del tipo “Correo enviado correctamente”

2.- Subir al servidor el fichero seleccionado en el formulario (el fichero es opcional)

- Al cumplimentar el formulario el usuario puede (opcionalmente) seleccionar un archivo de su dispositivo para subirlo al servidor. En caso que se seleccione un archivo éste debe cumplir las siguientes especificaciones:
 - Solo se permitirán archivos de tipo **jpg, jpeg, png o svg** y con extensión que corresponda a este tipo de archivos. En caso contrario se mostrará un error

Enviar
Extensión fichero no válida

- El tamaño máximo no debe superar los 100Kb



- Si el archivo es correcto éste se moverá a la carpeta *archivos*
- Si se selecciona un archivo tendremos que mostrar su nombre en el resumen del correo enviado (ver apartado anterior) pero, si el archivo no es correcto NO se enviará el email ni se escribirá en el log (este punto se trata en el apartado siguiente)
- Si se ha subido el archivo correctamente y, con posterioridad, se produce algún error en alguna parte del código que impida el envío de correo, SE DEBERA borrar el fichero de la carpeta *archivos* con la finalidad de dejar el sistema tal como se encontraba antes de subir el archivo

3.- Escritura en un fichero csv los datos de los correos enviados

Si el correo electrónico se ha enviado correctamente y, en caso de seleccionarse un fichero, éste se ha movido a la carpeta *archivos* de forma correcta, se grabará una fila en el fichero ***archivos/log.txt*** según las siguientes especificaciones:

- Se grabará una fila en formato **csv** (datos separados por punto y coma) con la siguiente información:
 - Si no se selecciona fichero:
fecha dia;email remitente;nombre remitente;mensaje;código consulta
24-09-2021;connie.corleone@mail.com;Connie Corleone;missatge 2;2636WX
 - Si se selecciona fichero:
fecha dia;email remitente;nombre remitente;mensaje;código consulta;nombre fichero
24-09-2021;peter.clemenza@mail.com;Peter Clemenza;missatge1 ;0JIGM5;imatge1.jpg
- Se mostrará una fila por correo enviado
- Las filas se mostrarán en orden inverso, es decir, los correos más recientes aparecerán los primeros. Ejemplo:

```
24-09-2021;peter.clemenza@mail.com;Peter Clemenza;missatge1 ;0JIGM5;imatge1.jpg
24-09-2021;connie.corleone@mail.com;Connie Corleone;missatge 2;2636WX
21-09-2021;peter@mail.com;Michael Corleone;Te haré una propuesta que no podrás
rechazar;F53062;9.png
21-09-2021;david-alcolea@hotmail.com;Michael Corleone;Te haré una propuesta que no
podrás rechazar;7461I3;alcyon_slider_5.jpg
18-09-2021;david-alcolea@hotmail.com;Michael Corleone;Te haré una propuesta que no
podrás rechazar;E42W18;alcyon_logo.png
```

NUEVO: 4.- Lectura de un fichero csv para mostrar su contenido

Debajo de la caja donde mostramos los datos del correo electrónico enviado, añadiremos una tabla donde mostraremos el contenido del archivo csv que usamos como log de correos.

Fecha: 29-09-2021
Remitente: Peter Clemenza
Teléfono:
Mensaje: missatge 4
Código consulta: 35SW41

0JIGM5	24-09-2021	peter.clemenza@mail.com
2636WX	24-09-2021	connie.corleone@mail.com
F53062	21-09-2021	peter@mail.com
746113	21-09-2021	david-alcolea@hotmail.com
E42W18	18-09-2021	david-alcolea@hotmail.com
35SW41	29-09-2021	peter@mail.com

Solo mostraremos, en cada una de las filas, los siguientes datos

- Código de consulta
- Fecha del correo
- Email del remitente

Especificaciones técnicas solicitadas:

- Utilizar funciones para organizar el código siempre que sea necesario
- Utilizar variables globales dentro de las funciones si lo consideráis necesario
- Recordad que deberíais utilizar la estructura `try...catch` para el tratamiento de errores (no hay que definir una estructura para cada error)
- Si hay algún error de validación y ya se ha movido el fichero a la carpeta archivos (en caso que se haya seleccionado uno) se deberá borrar de esta carpeta con `unlink()`

NOTA IMPORTANTE:

Con toda seguridad la función `mail()` o la librería `phpmailer` os fallará en localhost al no tener configurada una cuenta de destinatario del correo. En este caso os aparecerá el siguiente error al intentar enviar el correo

```
Warning: mail(): Failed to connect to mailserver at "localhost" port 25, verify your "SMTP" and "smtp_port" setting in php.ini or use ini_set()
```

Para probar el ejercicio mi consejo es que comentéis la función `mail` después de comprobar que os aparece este error y así podréis seguir probando el ejercicio

O bien:

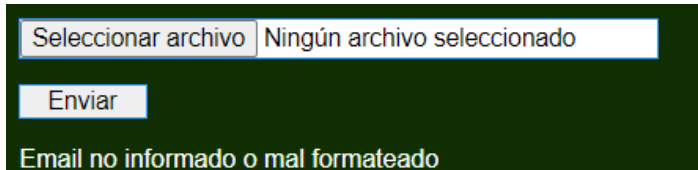
Configurar el archivo `php.ini` para añadir una cuenta de correo. Ver instrucciones en:

<http://albertotain.blogspot.com/2018/02/como-configurar-xampp-para-enviar.html>

Explicación técnica del Reto (PLA 05: Correo y ficheros php)

EJERCICIO 1: SUBIR FICHEROS AL SERVIDOR

A partir del fichero **contacto.php** proporcionado por el profesor, se añadirá la operativa para poder subir un fichero al servidor:



1.- Adaptaciones en el formulario de contacto

Para poder subir archivos al servidor hay que incorporar las siguientes modificaciones en el formulario (estas modificaciones ya están incorporadas al no ser el objetivo del curso de backend):

- En la etiqueta `form` tenemos que incluir la directiva siguiente:
`<form name="form" method="post" action="#" enctype="multipart/form-data">`
- Incorporaremos el `input` específico para selección de ficheros
`<input type="file" name="fichero">`

2.- Recuperar el archivo seleccionado en el servidor

La selección de un fichero es opcional, por lo tanto tendremos que poder identificar si se ha seleccionado o no un fichero.

Todos los datos de los ficheros que llegan al servidor desde un formulario los encontraremos en la super variable `$_FILES['fichero']` (donde 'fichero' es el atributo `name` de la etiqueta `input` del formulario) la cual nos proporciona la siguiente información:

<code>\$_FILES['fichero']['name']</code>	nombre del fichero seleccionado
<code>\$_FILES['fichero']['type']</code>	tipo de fichero obtenido de los metadatos del mismo (ojo que no es la extensión)
<code>\$_FILES['fichero']['tmp_name']</code>	ubicación del archivo en el servidor en espera de que indiquemos que hacer con él una vez subido (veréis que se encontrará en la carpeta <code>tmp</code>)
<code>\$_FILES['fichero']['size']</code>	tamaño del fichero en bytes
<code>\$_FILES['fichero']['error']</code>	código de error (0 = OK; 4 = no se ha seleccionado ningún fichero)

Con la información anterior podemos comprobar si se ha seleccionado un archivo (preguntando por el índice `error`) así como validar el tipo del archivo (índice `type`) y el tamaño que no supere los 100Kb (índice `size`)

NOTA: No está de más validar, además del tipo de archivo obtenido de los metadatos, la extensión del mismo para que se encuentre dentro de las extensiones válidas *jpg*, *jpeg*, *png* o *gif*:

Para ello:

- Tendríamos que recuperar la extensión utilizando las expresiones `strrpos()` para localizar la posición del último punto (que indicará la extensión del archivo) y, una vez obtenida la posición, recuperar la extensión con `substr()`
- Obtenida la extensión podríamos comprobar si ésta se corresponde con una de las autorizadas. Mi consejo es utilizar un array con las extensiones válidas y realizar la validación contra este array utilizando `in_array()`

3.- Mover el archivo a la carpeta definitiva

Si se han superado correctamente las validaciones tendremos que mover el archivo (que había quedado en la carpeta temporal **tmp**) a la carpeta definitiva **archivos**.

Si no hacemos nada con el archivo, éste se borrará automáticamente de la carpeta temporal cuando el servidor devuelva la página al navegador.

Para ello utilizaremos la función:

`move_uploaded_file(carpeta temporal, carpeta y nombre definitivo)`

Donde carpeta temporal la podemos obtener del índice `tmp_name` y nombre definitivo lo obtendremos del índice `name` (habrá que concatenar la ruta completa)

Esta función devolverá `true` si toda ha ido bien y `false` si ha habido algún error (en cuyo caso lanzaremos una excepción)

Puesto que el correo electrónico no hay que enviarlo si se produce algún error de validación en el fichero seleccionado os aconsejo que primero hagáis el tratamiento del archivo (en caso de seleccionar uno) y, posteriormente, el tratamiento del envío de correo (ver ejercicio 2)

4.- Borrar el archivo en caso de error

Si se han superado correctamente las validaciones y hemos movido el archivo a la carpeta definitiva y, con posterioridad, se produce algún otro error (por ejemplo al enviar el correo), tendremos que borrar el archivo de la carpeta archivos.

Para ello podemos utilizar la función `unlink(ruta/nombreArchivo)`

EJERCICIO 2: ENVIO DE CORREO ELECTRÓNICO

Si se ha seleccionado un fichero y éste se ha subido correctamente o bien si no hemos seleccionado ninguno realizaremos la operativa necesaria para enviar un correo electrónico con los datos descritos en el apartado de especificaciones de usuario:

- Correo electrónico a la dirección ficticia `destinatario@mail.com` con los datos y el formato que se especifican a continuación:

Asunto del correo: *"Correo desde formulario de contacto"*

Texto del correo (debe aparecer tal como se muestra a continuación):

Fecha: *fecha del día en curso*

Remitente: Nombre recuperado del formulario (*obligatorio*)

Teléfono: *Teléfono recuperado del formulario (opcional)*

Mensaje: *Comentario recuperado del formulario (obligatorio)*

Código consulta: *código de la consulta confeccionado de forma automática (ver más abajo);*

Nombre fichero: *nombre del fichero seleccionado (opcional)*

NOTA: Para obtener la fecha del día podemos utilizar `date('d-m-Y')`

- Validaremos que nombre, email y mensaje estén informados y mostraremos en el `span` con id `mensajes` el mensaje de error correspondiente. Teléfono es opcional

``

- Necesitamos obtener un código generado automáticamente para, si fuera una situación real, enviarlo al usuario para que pueda realizar el seguimiento de su consulta.

Este código lo obtendremos de la siguiente forma:

- Veréis que en `contacto.php` hay una constante predefinida con el valor:

`A1B2C3D4E5F6G7H8I9J0K1L2M3N4O5P6Q7R8S9T0U1V2W3X4Z5`

- Confeccionaremos un código de 6 posiciones con 6 caracteres obtenidos aleatoriamente a partir de la secuencia de letras y números de la constante.

- Determinar la longitud del texto de la constante con `strlen()`
- Mediante un bucle que se ejecutará 6 veces, obtener, en cada iteración, un número aleatorio entre 0 y la longitud de la constante utilizando `rand()`
- Recuperar el carácter que ocupa la posición que nos indique el número aleatorio anterior con `substr()`

- Para enviar el correo podemos utilizar indistintamente la función nativa `mail()` de php o bien la librería `phpmailer` descrita en el pdf que encontraréis en recursos

La función `mail()` devolverá un `true` si el envío ha sido correcto o un `false` en caso de error. Puesto que siempre os devolverá el siguiente error al no tener configurado en localhost el servidor de correo:

Warning: mail(): Failed to connect to mailserver at "localhost" port 25, verify your "SMTP" and "smtp_port" setting in php.ini or use ini_set()

Mi consejo es que, una vez que comprobéis que efectivamente os aparece este error, comentéis la función para seguir probando el resto del ejercicio:

- Si el envío es correcto (o habéis comentado la función `mail()`) se mostrará en la caja con `class correo` los datos del correo enviado

`<div class='correo'></div>`

Fecha: 28-09-2021

Remitente: Michael Corleone

Teléfono: 77777744

Mensaje: Te haré una propuesta que no podrás rechazar

Código consulta: F53062

Nombre fichero: 9.png

NOTA: El nombre del fichero solo se mostrará si se ha seleccionado uno en el formulario

EJERCICIO 3: GUARDAR UN FICHERO DE LOG CON LOS CORREOS ENVIADOS

Si se ha enviado el correo correctamente, se han mostrado los datos del mismo en la caja del formulario y, en caso de haberlo seleccionado, se ha subido el archivo correctamente, se escribirá una fila con los datos del correo en el fichero **archivos/log.txt** (una fila por correo) y en orden inverso, es decir, los correos más recientes aparecerán los primeros en el ficheros

```
24-09-2021;peter.clemenza@mail.com;Peter Clemenza;missatge1 ;0JIGM5;imatge1.jpg
24-09-2021;connie.corleone@mail.com;Connie Corleone;missatge 2;2636WX
21-09-2021;peter@mail.com;Michael Corleone;Te haré una propuesta que no podrás
rechazar;F53062;9.png
21-09-2021;david-alcolea@hotmail.com;Michael Corleone;Te haré una propuesta que no
podrás rechazar;7461I3;alcyon_slider_5.jpg
18-09-2021;david-alcolea@hotmail.com;Michael Corleone;Te haré una propuesta que no
podrás rechazar;E42W18;alcyon_logo.png
```

Para ello:

- Abrir el fichero en modo lectura/escritura con **fopen()** de forma que las operaciones de escritura se realicen al final sin borrar el contenido previo
- Recuperar todo su contenido con **fread()**
- Confeccionar la fila a grabar con los datos solicitados separados por punto y coma (ver imagen anterior) teniendo en cuenta que el nombre del fichero solo debe mostrarse si se ha seleccionado uno
- Escribir la fila que hemos confeccionado en el fichero con **fwrite()**
- Escribir el contenido previo que hemos recuperado en la lectura anterior
- Cerrar el fichero con **fclose()**

EJERCICIO 4: MOSTRAR FICHERO CON EL LOG DE CORREOS ENVIADOS

Debajo de la caja donde mostramos los datos del correo enviado construiremos una tabla para mostrar el contenido del fichero archivos/logs.txt. Se mostrará todo el contenido de forma que aparezca un correo por línea:

```
<div class='log'><table></table></div>
```

0JIGM5	24-09-2021	peter.clemenza@mail.com
2636WX	24-09-2021	connie.corleone@mail.com
F53062	21-09-2021	peter@mail.com
7461I3	21-09-2021	david-alcolea@hotmail.com
E42W18	18-09-2021	david-alcolea@hotmail.com
35SW41	29-09-2021	peter@mail.com

Para ello:

- Solo entrar en la página se debe mostrar el contenido del fichero con el contenido que tenga en ese momento
- Al enviar un nuevo correo éste debe aparecer inmediatamente en la tabla
- Para cada correo se mostrarán los siguientes datos
 - Código consulta
 - Fecha del correo
 - Email del remitente
- Necesitamos abrir el fichero de logs en modo lectura
- Leeremos el fichero línea a línea con `fgets()` hasta llegar al final del fichero.


```
while (!feof($fileHandler)) { ... }
```
- Para cada fila del fichero confeccionaremos una fila en la tabla con la siguiente estructura:


```
<tr><td>código consulta</td><td>fecha correo</td><td>email</td></tr>
```
- Para facilitar la extracción de los datos que necesitamos sería conveniente convertir cada fila del fichero (que se encuentra en formato csv - datos separados por punto y coma -) a array escalar. Para ello podemos utilizar la función:


```
explode(delimitador, csv);
```

Recursos de aprendizaje vinculados:

Toda la documentación (manuales, vídeo tutoriales, esquemas, etc) se encontrarán en el aula virtual así como los recursos necesarios para resolver el reto (imágenes, etc)

- Manual de cadenas, fechas, números, correo y subir ficheros en PHP
- Recursos complementarios para realizar los ejercicios *slot machine* y ficheros csv propuestos en el manual anterior