

Desarrollo Web con HTML

Módulo 4



PHP y servidor

¿Qué es PHP?

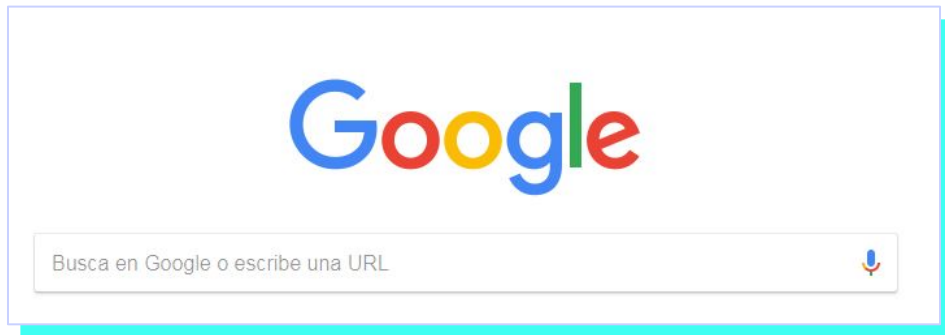
PHP es el **lenguaje de programación** más utilizado mundialmente, es de **código abierto** -es decir, cualquier programador lo puede usar a su gusto y crear sus propias estructuras- y **puede añadirse a nuestro código HTML** siguiendo algunas reglas.

El lenguaje PHP requiere conocimientos más profundos y específicos, ya que cuenta con una **sintaxis muy amplia** y es usado en proyectos de **alta complejidad**.



PHP es un lenguaje de servidor que nos permite **enviar y recibir información del servidor**.

Por ejemplo, está presente cuando enviamos información a través de un formulario de contacto, o recibimos información del formulario de Google al buscar algo:



PHP y formularios

Para poder probar lo anterior, **necesitamos subir nuestro proyecto a un servidor**. En este punto encontramos varias opciones válidas. Entre ellas instalar un servidor local de prueba si es que aún no contamos con un *Hosting*.

Si bien existen varias alternativas, les sugerimos trabajar con **XAMPP**. Un servidor local multiplataforma que nos permite en este caso probar nuestro PHP y el formulario vinculado incluso antes de poder subirlo a la web.

El **lenguaje PHP** como cualquier otro lenguaje de servidor requiere el servidor presente para trabajar.

En algunos casos como **Node.js** el trabajo se realiza a través de una consola.

Si bien es válido y hoy en día el mercado laboral pide en muchos casos tanto uno como el otro (**PHP y Node.js**) , en este caso dado lo introductorio del curso trabajaremos con PHP.

Instalación de XAMPP

Para poder instalar XAMPP vamos a seguir los siguiente pasos:

1. Ir a la [página oficial](#) del sitio.
2. Luego elegir alguna de las siguientes opciones en base al sistema operativo instalado.

Descargar

Pulsa aquí para otras versiones



XAMPP para
Windows

8.2.4 (PHP 8.2.4)



XAMPP para **Linux**

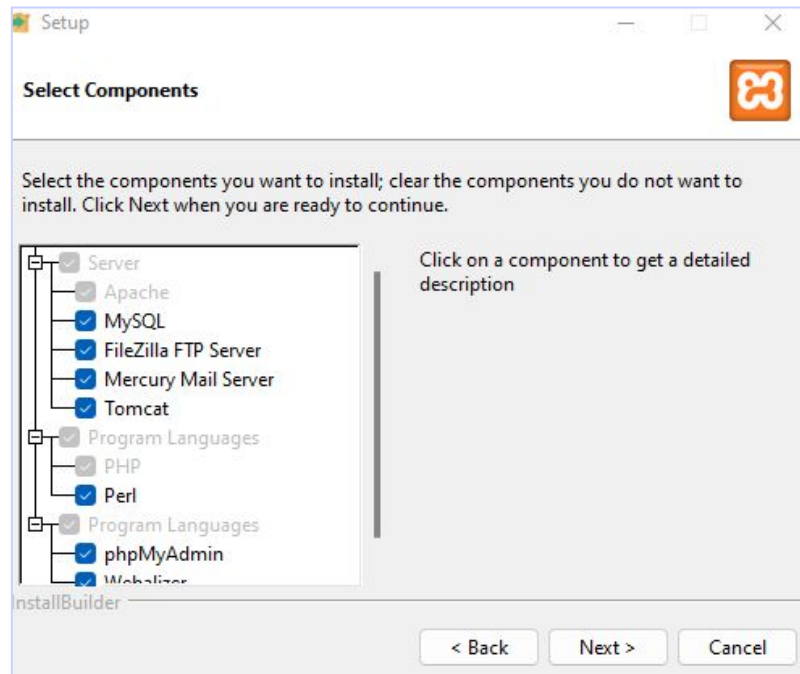
8.2.4 (PHP 8.2.4)



XAMPP para **OS X**

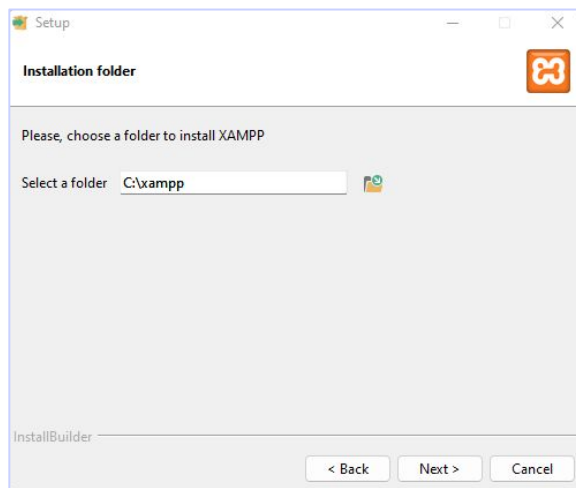
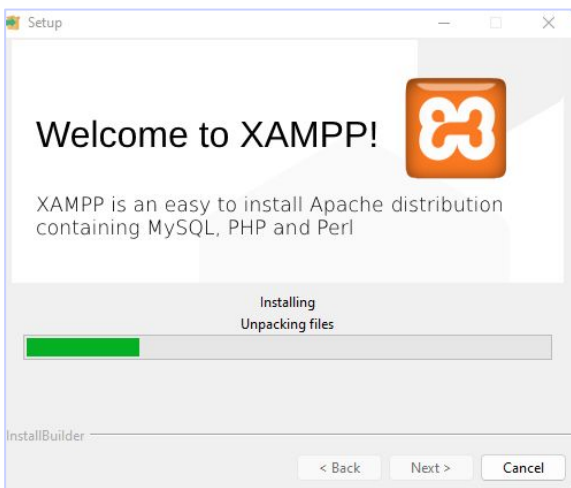
8.2.4 (PHP 8.2.4)

3. Luego, nos preguntará qué elementos o componentes deseamos instalar. En este punto dejaremos todas las opciones marcadas:



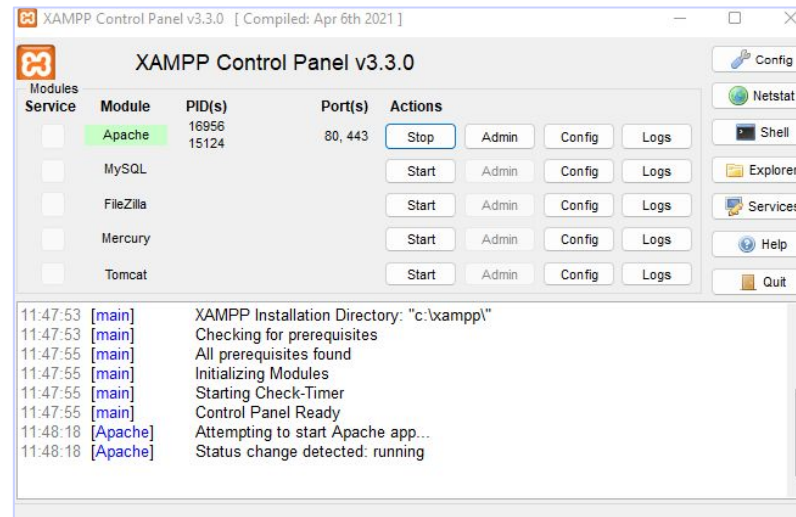
4. Como último paso, nos solicitará saber en qué disco guardaremos el programa y en qué idioma.

Esto es una elección del usuario , en nuestro caso lo hemos dejado en inglés y se guardará en el disco C:



5. Luego de instalado buscaremos **XAMPP** en nuestro menú de programas y lo ejecutaremos.

En este punto vamos a encontrarnos con la siguiente pantalla, donde debemos asegurarnos que **Apache** esté iniciado.

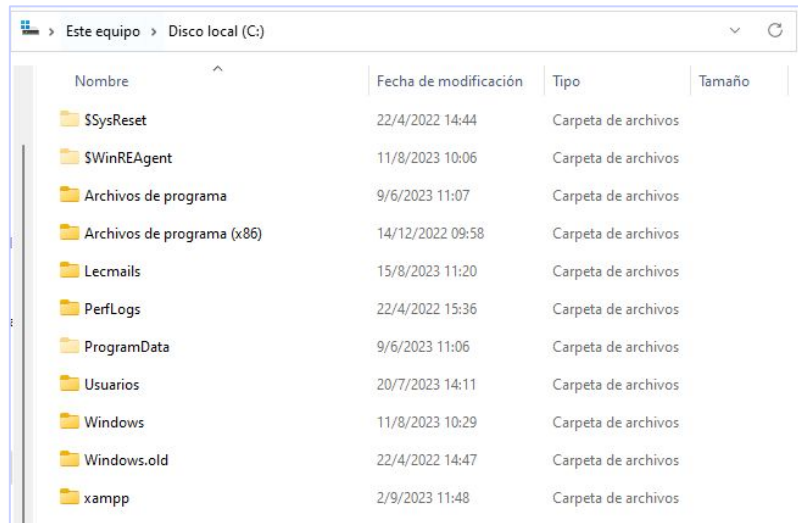


Información importante

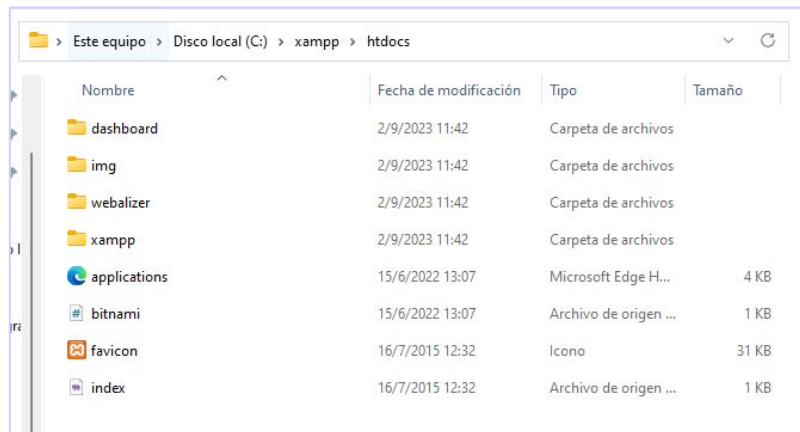
Para probar correctamente nuestro formulario anterior, debemos **subir nuestros archivos a la carpeta que designa XAMPP para trabajar.**

Recordemos que en pasos anteriores, determinamos en nuestro caso qué XAMPP estaría instalado en nuestro disco C.

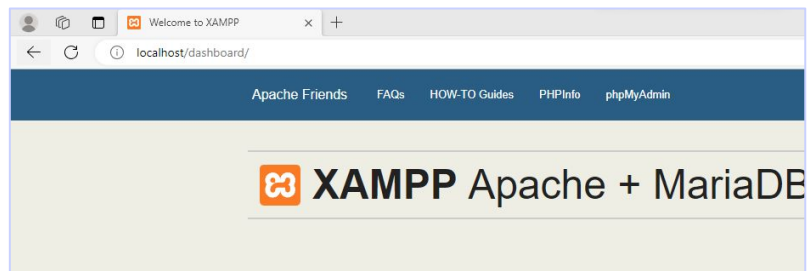
Iremos a tal disco, y buscaremos la carpeta XAMPP.



Subiremos nuestros archivos en cuestión, el HTML con el formulario y el código PHP a la carpeta HDOCS, dentro de la carpeta XAMPP.



Para poder visualizar nuestro archivo y probarlo correctamente, no podemos levantar nuestro html como siempre lo hacemos sino que tenemos que hacerlo desde la ubicación actual en la carpeta de XAMPP, qué es el localhost como se muestra en la imagen a la derecha.



Código de PHP

Para poder probar nuestro XAMPP **ya instalado**, vamos a realizar un simple formulario con un campo de nombre como el siguiente.

```
<form action="codigo.php">
  <label for="campoNombre">Nombre/s</label>
  <input required name="nombre" type="text" id="campoNombre">

  <button>Enviar</button>

</form>
```

PHP y formularios

Un dato importante a notar es que el **action** de nuestro form referencia al código php que se ejecutará al enviar el formulario.

Se va a realizar una primera prueba luego de instalado XAMPP.

```
<form action="codigo.php"
method="indica el modo a través del
cual transmito la
información"></form>
```

Crearemos un archivo llamado **codigo.php**.

En el mismo colocaremos lo que vemos en la imagen de la derecha.

Nota: recordar subir los archivos a la carpeta HTDOCS y probarlo desde el localhost.

```
1 <?php
2
3 echo 'Gracias por enviar tu mail |'
4
5 ?>
```

El código anterior de PHP nos devolverá solo una respuesta, pero para hacer que nuestro formulario funcione realice algo más interesante, por ejemplo, devolvernos el nombre del usuario en una oración, vamos a hacer algunos cambios tanto en el PHP como en el formulario.

```
<form action="php/codigo.php" method="post">
  <label for="campoNombre">Nombre/s</label>
  <input required name="nombre" type="text"
    id="campoNombre">

  <button>Enviar</button>

</form>
```

```
1 <?php
2
3 echo 'Tu nombre es ' . ($_POST['nombre']);
4
5
6 ?>
```

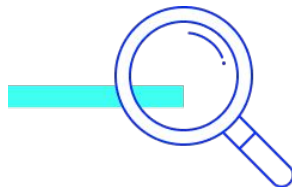
Vamos a hacer algunas aclaraciones. En el caso del name, necesitamos que **el campo de nombre tenga el name “nombre”** para que esto funcione, Lo probamos y el servidor devolverá la respuesta.

Si bien podemos intentar con muchos códigos distintos, una opción más compleja e involucrando más elementos en nuestro formulario sería:

```
5 $estado = $_POST['estado'];
6 $comentarios = $_POST['comentarios'];
7 $opcion = $_POST['pais'];
8 $condiciones = $_POST['condiciones'];
9 $to = 'tumail';
10 $email_subject = "Nuevo mensaje: de la web";
11 $email_body = '|El nombre es<br>'.$nombre 'El mail es<br>'. $email . ;
12 $headers = "From: $email";
13
14 mail($to, $email_subject, $email_body, $headers);
15 echo 'Pronto nos estaremos comunicando con usted!';
16 ?>
17
```

Revisión

- Repasar los **elementos de formulario** vistos en el módulo anterior.
- Si te interesa profundizar en **programación o PHP**, **recomendamos realizar un curso acorde con sus intereses**, ya que en este curso ofrecemos un ejemplo simple dado el conocimiento y tiempo disponible.
- Realizar el **laboratorio** propuesto para que nuestro formulario de contacto funcione.



**¡Sigamos
trabajando!**