



Desarrollo Full Stack

Fundación Telefónica Movistar
CURSO DE DESARROLLO FULL STACK.

Desarrollo de una aplicación web.

Vamos a empezar con el desarrollo de una aplicación web real, aplicando los conocimientos vistos en clase.

El objetivo es hacer una aplicación que contenga todos los elementos que necesitan para el desarrollo del trabajo práctico y que les brinde los conocimientos básicos para encarar cualquier tipo de desarrollo web basado en PHP, Bootstrap y MySQL.

Parte 1. Plantilla HTML con Bootstrap.

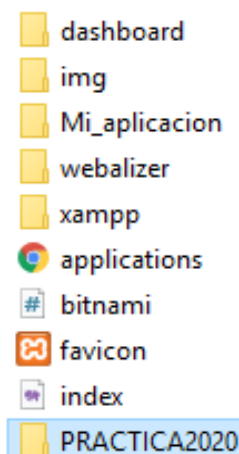
En esta primera parte, vamos a armar una plantilla base en HTML, con Bootstrap descargado en nuestro equipo y un menú de navegación.

Esta plantilla nos servirá para ir agregando funcionalidades a nuestra aplicación a medida que las veamos en forma teórica.

PASO 1. Crear la carpeta de trabajo.

Como ya sabemos, todas nuestras aplicaciones PHP deberán estar dentro de la carpeta HTDOCS.

Vamos a crear la carpeta de trabajo, en mi caso se llamará PRACTICA2020, pero pueden ponerle el nombre que quieran.



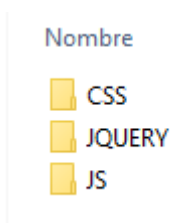
- dashboard
- img
- Mi_aplicacion
- webalizer
- xampp
- applications
- bitnami
- favicon
- index
- PRACTICA2020**



PASO 2. Creamos las carpetas principales de la aplicación.

Si bien no es obligatorio crear las siguientes carpetas, es una buena práctica mantener agrupados y ordenados todos los archivos que nuestra aplicación necesita, para facilitar el trabajo en equipo y el mantenimiento de esta.

Vamos a empezar con las carpetas básicas, las destinadas a los archivos CSS, JS y JQUERY.



PASO 3. Descargamos Bootstrap.

Desde la página de Bootstrap, nos descargamos todos los archivos de la última versión.

(<https://getbootstrap.com/>)

El anterior LINK nos lleva a la página de descarga de la versión actual, y dentro de dicha página, tendremos varias opciones de descarga.

Vamos a optar por la primera opción que veamos, la de descargar los archivos compilados y listos para utilizar:



Compiled CSS and JS

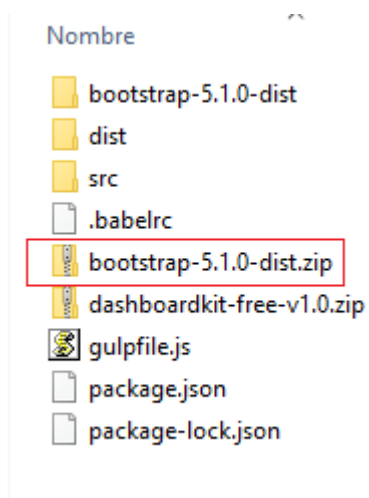
Download ready-to-use compiled code for **Bootstrap v4.5.2** to easily drop into your project, which includes:

- Compiled and minified CSS bundles (see [CSS files comparison](#))
- Compiled and minified JavaScript plugins

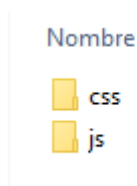
This doesn't include documentation, source files, or any optional JavaScript dependencies (jQuery and Popper.js).

Download

La descarga consta de un solo archivo comprimido, que podemos guardar en cualquier parte de nuestro equipo, porque solamente vamos a utilizar algunos de los archivos que están comprimidos, no todos.



Una vez descargado, lo descomprimos y vemos que nos quedan dos carpetas:








Ya tenemos Bootstrap descargado, el siguiente paso es encontrar el archivo CSS y el archivo JS que necesitamos y los llevamos a su correspondiente carpeta dentro de nuestra aplicación.


PASO 4. Archivo BOOTSTRAP.MIN.CSS

Ingresamos a la carpeta CSS que acabamos de descomprimir con la descarga de Bootstrap.

Dentro de ella, ubicamos el archivo llamado “bootstrap.min”.

Nombre	Fecha de modifica...	Tipo	Tamaño
 bootstrap	6/8/2020 13:08	Archivo de origen CSS	194 KB
 bootstrap.css.map	6/8/2020 13:08	Archivo MAP	496 KB
 bootstrap.min	6/8/2020 13:08	Archivo de origen CSS	157 KB
 bootstrap.min.css.map	6/8/2020 13:08	Archivo MAP	632 KB
 bootstrap-grid	6/8/2020 13:08	Archivo de origen CSS	66 KB

Lo copiamos y lo pegamos dentro de la carpeta CSS de nuestra aplicación.









Unidad de USB (D:) > Programas > xampp > htdocs > PRACTICA2020 > CSS			
Nombre	Fecha de modifica...	Tipo	Tamaño
 bootstrap.min	6/8/2020 13:08	Archivo de origen ...	157 KB




PASO 5. Archivo BOOTSTRAP.MIN.JS

Ingresamos a la carpeta JS que acabamos de descomprimir con la descarga de Bootstrap.

Dentro de ella, ubicamos el archivo llamado “bootstrap.min”.

Nombre	Fecha de modifica...	Tipo	Tamaño
 bootstrap.bundle	6/8/2020 13:08	JavaScript File	224 KB
 bootstrap.bundle.js.map	6/8/2020 13:08	Archivo MAP	401 KB
 bootstrap.bundle.min	6/8/2020 13:08	JavaScript File	80 KB
 bootstrap.bundle.min.js.map	6/8/2020 13:08	Archivo MAP	309 KB
 bootstrap	6/8/2020 13:08	JavaScript File	133 KB
 bootstrap.js.map	6/8/2020 13:08	Archivo MAP	250 KB
 bootstrap.min	6/8/2020 13:08	JavaScript File	59 KB
 bootstrap.min.js.map	6/8/2020 13:08	Archivo MAP	188 KB

Lo copiamos y lo pegamos dentro de la carpeta JS de nuestra aplicación.

Unidad de USB (D:) > Programas > xampp > htdocs > PRACTICA2020 > JS			
Nombre	Fecha de modifica...	Tipo	
 bootstrap.min	6/8/2020 13:08	JavaScript File	

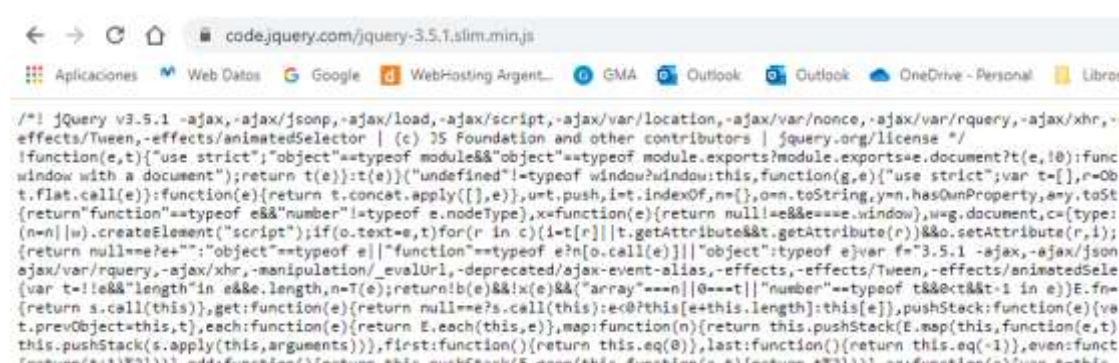
PASO 6. Archivo JQUERY (Solo en caso de ser necesario)

Ingresamos a la página de descarga de JQUERY, para ahorrar el trabajo de navegar entre las diferentes opciones de descarga que nos ofrece la página



<https://jquery.com/> les dejo a continuación el link de la última versión del plugin que vamos a usar: <https://code.jquery.com/jquery-3.5.1.slim.min.js>

Esto es lo que veremos en nuestro navegador:

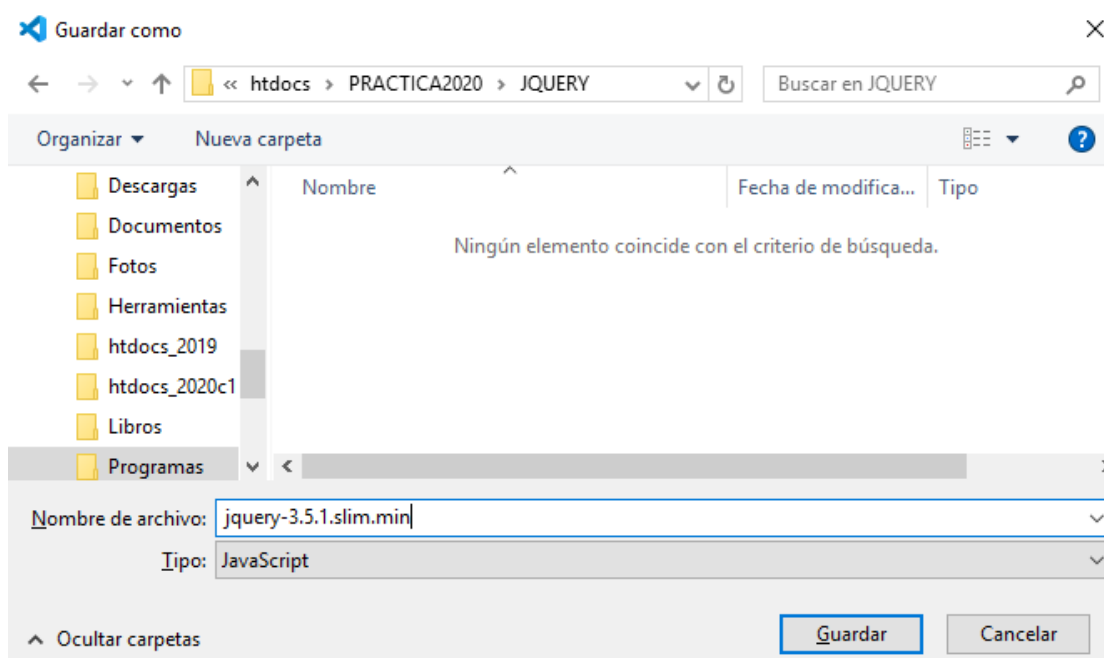


Seleccionamos todo el contenido (CTRL + A) y lo copiamos (CTRL + C).


Abrimos nuestro editor de texto, creamos un nuevo archivo y pegamos todo el contenido que acabamos de copiar.

Guardamos el archivo dentro de la carpeta JQUERY de nuestra aplicación web con el nombre "jquery-3.5.1.slim.min" y con extensión JS.





Así nos queda el archivo guardado como JAVASCRIPT dentro de la carpeta JQUERY de nuestra aplicación:

Unidad de USB (D:) > Programas > xampp > htdocs > PRACTICA2020 > JQUERY				
Nombre	Fecha de modifica...	Tipo	Tamaño	
 jquery-3.5.1.slim.min	14/9/2020 20:29	JavaScript File	71 KB	

PASO 7. Creamos un archivo modelo para la aplicación.

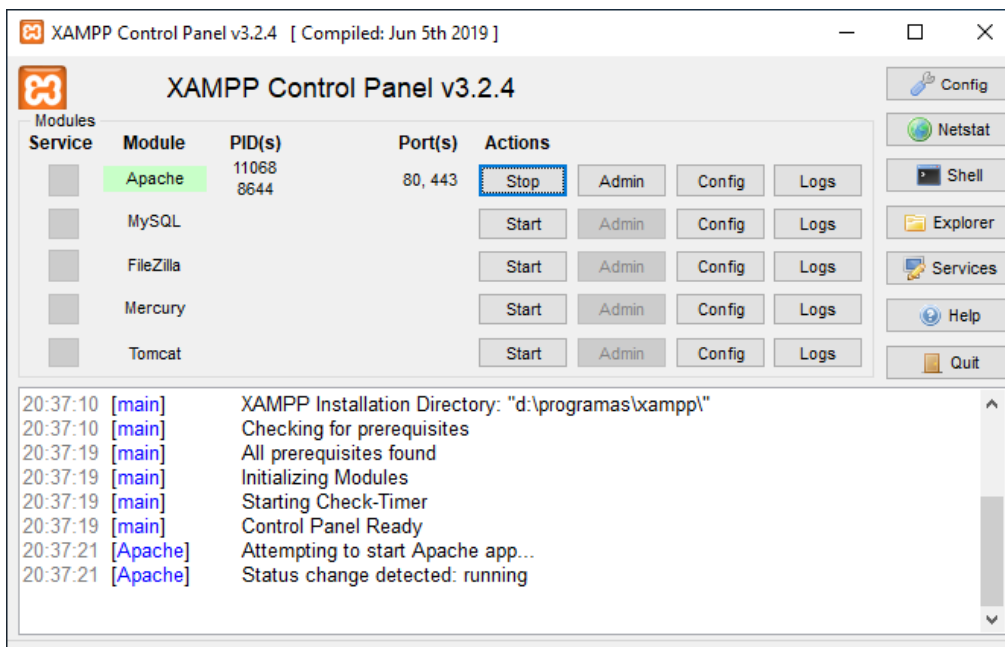
Ya tenemos todos los archivos necesarios y la estructura básica para armar cualquier tipo de aplicación web basada en Bootstrap.

Solo resta ponerse a programar, y lo primero que vamos a hacer, es una plantilla HTML para reutilizar en nuestra aplicación.

El objetivo es tener un archivo que nos sirve de molde, en lo que se refiere a estilo, para todos los archivos que necesitemos crear.



Antes de ponernos a programar, es necesario encender el servidor PHP.

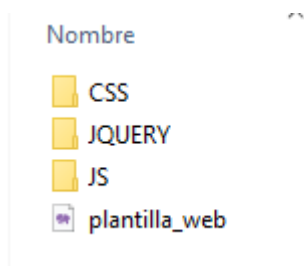


A continuación, verificamos si podemos ingresar al LOCALHOST.



Comenzamos ahora a crear la plantilla para nuestra aplicación.

Desde el editor de texto, creamos un archivo y lo guardamos, dentro de nuestra aplicación con un nombre cualquiera, en mi caso, la voy a guardar como `plantilla_web.php`



Dentro del archivo, escribimos la estructura básica de una página HTML tal y como la aprendimos en clase.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Plantilla Web</title>
  <h1>Plantilla Web.</h1>
</head>
<body>

</body>
</html>
```



Ubicamos desde el navegador nuestro archivo recién creado.

*** IMPORTANTE: ubicar correctamente en la URL del navegador, la ubicación de nuestro archivo:

http://localhost/practica2020/plantilla_web.php

PASO 8. Agregamos los enlaces para los archivos de Bootstrap.

Necesitamos agregar a nuestra plantilla, los enlaces para que levante los archivos necesarios para poder trabajar con Bootstrap.

El primer archivo que necesitamos incorporar es el de estilos, el archivo CSS.

Lo vamos a incorporar mediante la etiqueta LINK que colocaremos en el encabezado de nuestra página, entre las etiquetas HEAD.

```
<!-- CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
```

El siguiente archivo a incorporar, será el de funciones JavaScript que descargamos desde la web de Bootstrap, y que tenemos guardado dentro de la carpeta JS de nuestra aplicación.

Lo usual es colocar todos los enlaces a archivos JS al final de nuestro código HTML, para que sea lo último que cargue y no demore en mostrar la página a los usuarios.

Estos enlaces los indicaremos mediante el uso de la etiqueta SCRIPT.



```
<!-- JavaScript -->  
  
<script src="jquery/jquery-3.5.1.slim.min.js"></script>  
  
<script src="js/bootstrap.min.js"></script>
```

PASO 9. Probamos elementos de Bootstrap.

Ya tenemos lista nuestra plantilla HTML base para nuestra aplicación, que incorpora todos los archivos de Bootstrap necesarios para que podamos utilizar los elementos de dicho framework.

Vamos a probar si funciona correctamente y no cometimos ningún error en los pasos que dimos.

Tenemos que probar el estilo y las funcionalidades de JavaScript.

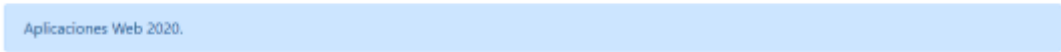
Para comprobar los estilos, vamos a aplicar al BODY la clase CONTAINER y vamos a agregar un DIV del tipo ALERT a manera de título de la página.

```
<body class="container">
```

```
<br>  
  
<div class="alert alert-primary" role="alert">  
    Aplicaciones Web 2020.  
</div>
```




Si todo salió bien, esto es lo que deberíamos ver en nuestro navegador:



Probemos ahora algunos de los elementos que requieren las funciones de JS, como, por ejemplo, los SPINNERS.

```
<br>
<div class="spinner-border" role="status">
  <span class="sr-only">Loading...</span>
</div>
```



Si vemos el elemento en movimiento en nuestro navegador, es porque pudimos colocar los enlaces correctamente dentro de nuestra plantilla y ya la tenemos lista para utilizar como base de todos nuestros archivos PHP.

