



ELABORACIÓN DE PÁGINAS WEB CON VALIDACIONES DE FORMULARIOS DESDE CLIENTE

T 7.3

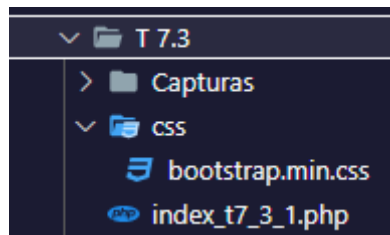


FEBRERO DE 2022

RODRÍGUEZ JÁCOME, DAVID
Desenvolvimento Web en Contorno Servidor

1. Reproduce el ejemplo que se describe en el siguiente link.

Para este ejemplo crearemos un nuevo proyecto con la siguiente estructura de carpetas:



En este ejemplo copiaremos el contenido del código de muestra, que será el que incrustemos en el archivo “index_t7_3_1.php”. Además, necesitaremos la librería Bootstrap 4.5, de la que descargaremos el código de internet y lo incluiremos en el archivo “bootstrap.min.css” dentro del subdirectorio CSS.

En el archivo “index_t7_3_1.php”, la cabecera presenta el título y el enlace de la librería Bootstrap, además de parte de los estilos para el aviso de introducir datos en los campos requeridos.

```
1  <html>
2      <head>
3          <meta charset="utf-8">
4          <title>Validacion de formulario con PHP</title>
5          <link rel="stylesheet" href="css/bootstrap.min.css">
6          <style>
7              .error {
8                  color: #FF0000;
9              }
10             .invalid-feedback {
11                 display: block !important;
12             }
13         </style>
14     </head>
15     <body>
```

En el siguiente bloque de código se establece el control de validación de los campos del formulario. Los datos introducidos en los campos se ajustan (quitar espacios, etc) con el método `validar_input()` y se valida que el correo cumple con los requisitos con la función `FILTER_VALIDATE_EMAIL`.

```

16      <?php
17      // Definir variables.
18      $nombreErr = $emailErr = "";
19      $nombre = $email = "";
20      if ($_SERVER["REQUEST_METHOD"] == "POST") {
21          if (empty($_POST["nombre"])) {
22              $nombreErr = "Nombre es requerido";
23          } else {
24              $nombre = validar_input($_POST["nombre"]);
25          }
26
27          if (empty($_POST["email"])) {
28              $emailErr = "Email es requerido";
29          } else {
30              $email = validar_input($_POST["email"]);
31              // Verifica el correcto formato de email.
32              if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
33                  $emailErr = "Formato de email invalido";
34              }
35          }
36      }
37      function validar_input($data)
38      {
39          $data = trim($data);
40          $data = stripslashes($data);
41          $data = htmlspecialchars($data);
42          return $data;
43      }
44      ?>

```

En la siguiente imagen se muestra el bloque de código que corresponde al aspecto visual del formulario.

```

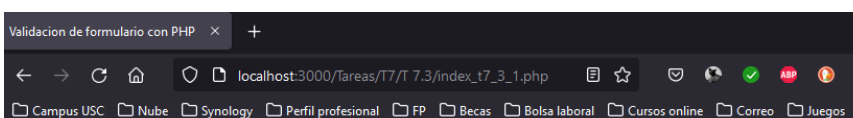
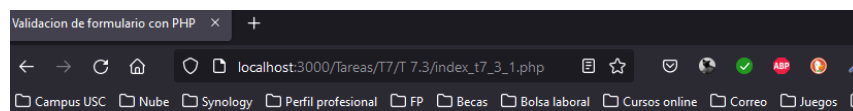
45      <div class="container">
46          <br>
47          <div class="card bg-light">
48              <div class="card-header">
49                  Formulario de registro con validación
50              </div>
51              <div class="card-body">
52                  <span class="error">* Campos requeridos.</span>
53                  <div class="row">
54                      <div class="col-md-6">
55                          <form method="post" action="<?php
56                              echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
57                              <div class="form-group">
58                                  <label class="col-form-label"><span class="error">* </span>Nombre</label>
59                                  <input name="nombre" type="text" class="form-control" value="<?php if (!empty($_POST["nombre"])) {
60                                      echo $nombre;
61                                  } else {
62                                      echo "";
63                                  } ?>">
64                                  <span class="error invalid-feedback"> <?php echo $nombreErr; ?></span>
65                              </div>
66                              <div class="form-group">
67                                  <label class="col-form-label"><span class="error">* </span>Email</label>
68                                  <input name="email" type="text" class="form-control" value="<?php if (!empty($_POST["email"])) {
69                                      echo $email;
70                                  } else {
71                                      echo "";
72                                  } ?>">
73                                  <span class="error invalid-feedback"> <?php echo $emailErr; ?></span>
74                              </div>
75                              <div class="form-group">
76                                  <button class="btn btn-primary btn-block" name="submit">Enviar</button>
77                              </div>
78                          </form>
79                      </div>
80                  </div>
81      </div>

```

Y esta última parte se ejecuta si la introducción de los datos del formulario ha sido correcta.

```
82     <?php
83     if ($_SERVER["REQUEST_METHOD"] == "POST") {
84         if (empty($nombreErr) and empty($emailErr)) {
85             echo "<br><h6>Tus valores son:</h6>";
86             echo "<strong>Nombre</strong>: " . $nombre;
87             echo "<br>";
88             echo "<strong>Email</strong>: " . $email;
89             echo "<br>";
90         }
91     }
92     ?>
```

El resultado de la ejecución por pantalla es el siguiente.



Formulario de registro con validación

* Campos requeridos.

* Nombre

* Email

Enviar

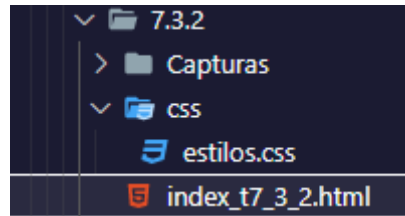
Tus valores son:

Nombre: David

Email: ejemplo@ejemplo.es

2. El siguiente enlace explica qué es la validación de formularios del lado cliente. Intenta reproducir los ejemplos que se muestran en él.

La estructura de carpeta es la siguiente:



En el archivo index nos encontramos una cabecera en la que se enlaza un css que se le da estética a la vista de formulario.

```
1  <!DOCTYPE html>
2  <html lang="en">
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      <link href="css/estilos.css" type="text/css" rel="stylesheet" />
9      <title>Index T7.3.2</title>
10 </head>
```

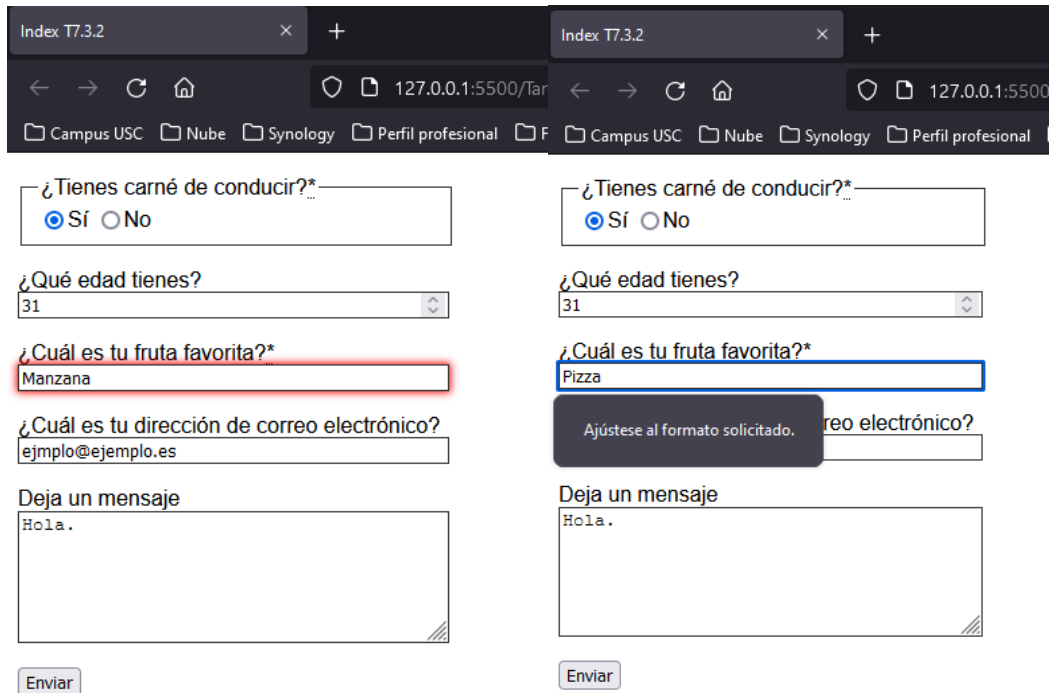
El formulario está hecho en HTML. En esta primera parte se detalla cómo funciona la restricción para la selección de si se es o no conductor: sólo se puede elegir una opción. Esto se hace con un input de tipo radio.

```
12 <body>
13 <form>
14 <p>
15 <fieldset>
16 <legend>¿Tienes carné de conducir?<abbr title="Este campo es obligatorio" aria-label="required">*</abbr>
17 </legend>
18 <!-- Solo se puede seleccionar un botón de opción en un grupo con el mismo nombre,
19 y, por lo tanto, solo un botón de opción en un grupo con el mismo nombre que tiene marcado el atributo «requerido»
20 basta para hacer de una selección un requisito -->
21 <input type="radio" required name="driver" id="r1" value="yes"><label for="r1">Sí</label>
22 <input type="radio" required name="driver" id="r2" value="no"><label for="r2">No</label>
23 </fieldset>
24 </p>
25 <p>
26 <label for="n1">¿Qué edad tienes?</label>
27 <!-- El atributo pattern puede actuar como una alternativa para los navegadores que
28 no implementan el tipo de entrada de número, pero admiten el atributo pattern.
29 Ten en cuenta que los navegadores que admiten el atributo pattern lo harán
30 fallar silenciosamente cuando se use con un campo numérico.
31 Su uso aquí solo actúa como una alternativa -->
32 <input type="number" min="12" max="120" step="1" id="n1" name="age" pattern="\d+>
33 </p>
```

En la segunda parte del formulario se hace un control de entrada de datos según el campo requerido. En la pregunta de fruta favorita se hace una restricción a las opciones disponibles prefijadas; en el correo electrónico se usa un input de tipo email, y en el cuadro de mensaje se usa una etiqueta textarea con una restricción de máximo 140 caracteres y formateado en cinco líneas.

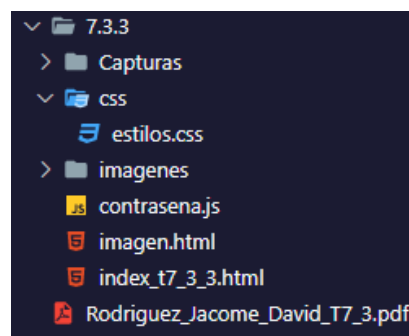
```
34     <p>
35         <label for="t1">¿Cuál es tu fruta favorita?<abbr title="Este campo es obligatorio"
36             aria-label="required">*</abbr></label>
37         <input type="text" id="t1" name="fruit" list="l1" required
38             pattern="[Bb]anana|[Cc]herry|[Aa]pple|[Ss]trawberry|[Ll]emon|[Oo]range ">
39         <datalist id="l1">
40             <option>Plátano</option>
41             <option>Cereza</option>
42             <option>Manzana</option>
43             <option>Fresa</option>
44             <option>Limón</option>
45             <option>Naranja</option>
46         </datalist>
47     </p>
48     <p>
49         <label for="t2">¿Cuál es tu dirección de correo electrónico? </label>
50         <input type="email" id="t2" name="email">
51     </p>
52     <p>
53         <label for="t3">Deja un mensaje</label>
54         <textarea id="t3" name="msg" maxlength="140" rows="5"></textarea>
55     </p>
56     <p>
57         <button>Enviar</button>
58     </p>
59 </form>
```

El resultado por pantalla es el siguiente. Si se introduce un valor incorrecto, aparecerá un mensaje:



3. Realiza de forma original una página web en la que tengas que validar formularios desde cliente. Explica en detalle los pasos que tengas que dar. Debe aparecer:
 - ¿Eres de carne o pescado?
 - Los campos a cumplimentar deben ser obligatorios para poder continuar con la visualización de la página, por ejemplo, que se pueda ver una imagen.
 - Introduce tu nombre.
 - Introduce tu contraseña. Cuando se crea una contraseña por primera vez debe tener las siguientes características: entre 5 y 10 caracteres alfanuméricos, en los cuales debe aparecer la “L” y el 3 de forma obligatoria.

Para este punto creamos un proyecto nuevo en el que la estructura de carpeta será la siguiente:



El archivo index contiene la cabecera que aloja el enlace al CSS que da la estética al documento y al JavaScript que comprueba la contraseña.

```
index_t7_3_3.html X
Tareas > T7 > T7.3 > 7.3.3 > index_t7_3_3.html > html
1  <!DOCTYPE html>
2  <html lang="en">
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      <link href="css/estilos.css" type="text/css" rel="stylesheet" />
9      <script src="contrasena.js" type="text/javascript"></script>
10     <title>Index T7.3.3</title>
```

Este archivo también contiene el formulario que validará los datos. Se compone de tres campos: un input radio donde elegimos una opción u otra mediante una restricción de selección; un campo input para el nombre y un campo contraseña con restricción obligatoria de medir entre 5 y 10 caracteres el botón final enviará la contraseña al JS para validarla.

```
13 <body>
14     <form action="index_t7_3_3.html" method="post">
15         <fieldset class="form">
16             <p>Los campos en rojo son obligatorios</p>
17             <legend>Formulario</legend>
18             <fieldset id="campo1">
19                 <legend>¿Prefieres carne o pescado?</legend>
20                 <ul>
21                     <li>
22                         <input type="radio" required name="carne" id="carne" value="yes">
23                         <label for="carne">Carne</label>
24                     </li>
25                     <li><input type="radio" required name="pescado" id="pescado" value="no">
26                         <label for="pescado">Pescado</label>
27                     </li>
28                 </ul>
29             </fieldset>
30             <fieldset id="campo2">
31                 <legend>Introduce tu nombre</legend>
32                 <p><label for="nombre"></label>Nombre:
33                 <input type="text" id="nombre" placeholder="Tu nombre" name="nombre" size="10" required />
34                 </p>
35             </fieldset>
36             <fieldset id="campo3">
37                 <legend>Introduce una contraseña</legend>
38                 <p>La contraseña debe tener entre 5 y 10 caracteres e incluir L y 3:</p>
39                 <p><label for="contrasena"></label>Contraseña:
40                 <input type="password" id="contrasena" placeholder=" Tu contraseña" name="contrasena"
41                     minlength="5" maxlength="10" required />
42                 </p>
43             </fieldset>
44             <button type="button" onclick="comprobarContrasena(contrasena.value)">Enviar</button>
45         </form>
46     </body>
```


El archivo JS contiene dos funciones que comprueban los datos de la contraseña.

```
contrasena.js X
Tareas > T7 > T 7.3 > 7.3.3 > contrasena.js > comprobarContrasena
1  let vector = new Array();
2
3  function comprobarContrasena(contrasena) {
4      if (vector.length < 1) {
5          guardarContrasena(contrasena);
6      } else {
7          if (contrasena == vector[0]) {
8              window.open("imagen.html");
9          } else {
10             alert("La contraseña no es correcta. Inténtalo de nuevo.");
11         }
12     }
13 }
14
15 function guardarContrasena(contrasena) {
16     let contNumero = "0";
17     let contLetra = "0";
18
19     for (let i = 0; i < contrasena.length; i++) {
20         if (contrasena.charAt(i) == "L") {
21             contLetra++;
22         };
23         if (contrasena.charAt(i) == 3) {
24             contNumero++;
25         }
26     }
27     if (contNumero > 0 && contLetra > 0 && contrasena.length > 4 && contrasena.length < 11) {
28         vector.push(contrasena)
29     } else {
30         alert("Introduce una contraseña que cumpla con los requisitos:\n"
31             + "Letra L máyuscula\n"
32             + "número 3\n"
33             + "la contraseña debe tener entre 5 y 10 caracteres.");
34     }
35 }
```

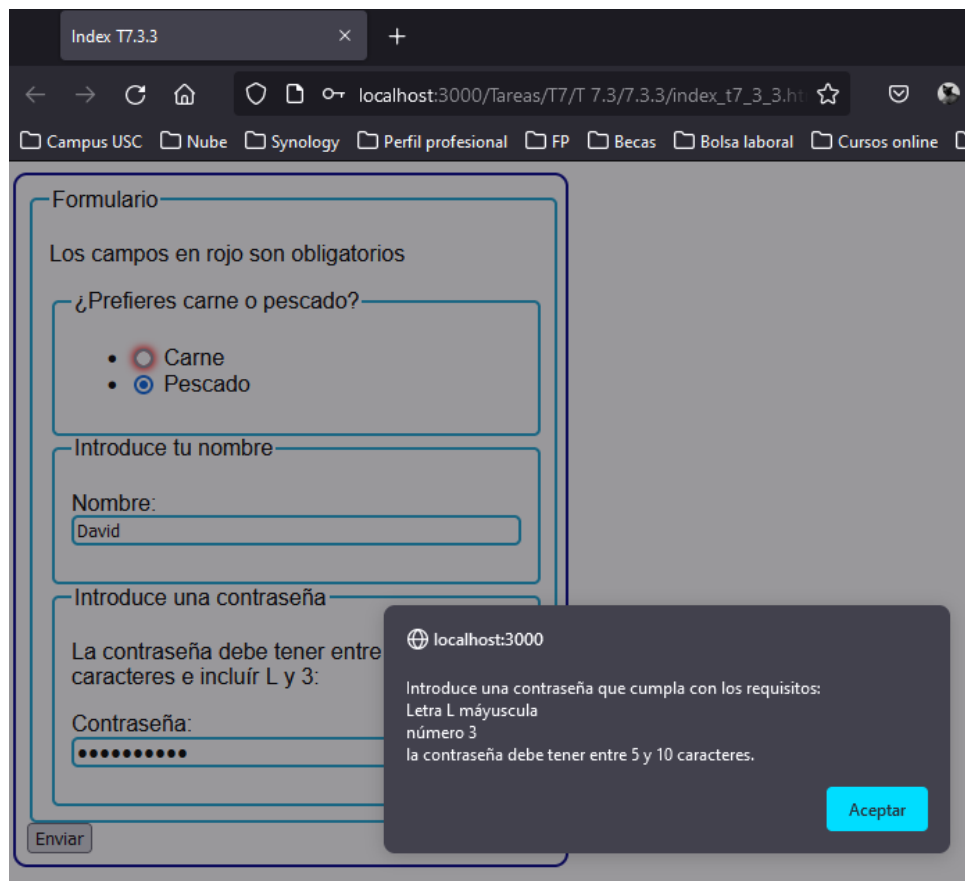
Si los datos son correctos, se abrirá una segunda página imagen.html que indicará al usuario que todos los datos se han introducido correctamente.

```
imagen.html X
Tareas > T7 > T 7.3 > 7.3.3 > imagen.html > ...
1  <!DOCTYPE html>
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>Logueado</title>
8  </head>
9  <body>
10     
11 </body>
12 </html>
```

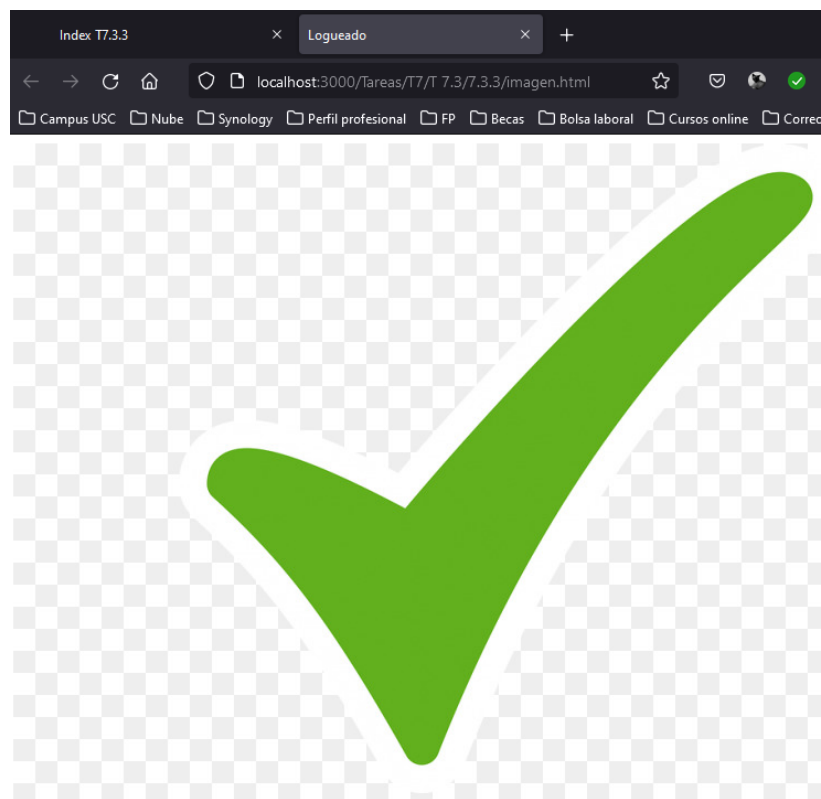
La ejecución es la siguiente:

The screenshot shows a web browser window with the title 'Index T7.3.3'. The address bar displays 'localhost:3000/'. Below the browser window, a form titled 'Formulario' is shown. The form contains the following elements:

- A heading 'Formulario'.
- A note: 'Los campos en rojo son obligatorios'.
- A question: '¿Prefieres carne o pescado?'. Below it are two radio buttons: 'Carne' and 'Pescado'. Both radio buttons are highlighted with a red glow.
- A section titled 'Introduce tu nombre'. It contains a label 'Nombre:' and a text input field with the placeholder 'Tu nombre'. The input field is highlighted with a red glow.
- A section titled 'Introduce una contraseña'. It contains a note: 'La contraseña debe tener entre 5 y 10 caracteres e incluir L y 3:'. Below it is a label 'Contraseña:' and a text input field with the placeholder 'Tu contraseña'. The input field is highlighted with a red glow.
- An 'Enviar' button at the bottom left of the form.



Con los datos introducidos correctamente cambia la pantalla indicándonos que hemos tenido éxito.



BIBLIOGRAFÍA/WEBGRAFÍA

- Cristian Sulbaran (2021): *Validación de datos: cómo verificar la entrada del usuario en formularios HTML con código JavaScript de ejemplo*, en freecodecamp.org [recurso electrónico], consulta el 20 de febrero de 2022 (<https://www.freecodecamp.org/espanol/news/validacion-de-datos-como-verificar-la-entrada-del-usuario-en-formularios-html-con-codigo-javascript-de-ejemplo/>).
- InformáticaPC: *Validar en PHP los datos recibidos de un formulario*, col. Tutorial de PHP, en informaticapc.com [recurso electrónico], consulta el 20 de febrero de 2022 (<https://informaticapc.com/tutorial-php/validar-formulario.php>).
- J&G Proyectos Web (2017): *Validar formulario con PHP*, en YouTube [recurso electrónico], consulta el 20 de febrero de 2022 (<https://www.youtube.com/watch?v=0i2WulWScw0&t=1s>).
- Tapia, N. (2021): *Validación de formularios con PHP*, en baulphp.com [recurso electrónico], consulta el 20 de febrero de 2022 ()).