
Práctica T04 - Formularios y Fuentes

Javier Beteta

2022-01

1. Introducción

Vamos a realizar un par de ejercicios. Primero mejoraremos un formulario realizado con HTML4 con las mejoras que podemos encontrar en HTML5. En segundo lugar aplicaremos las mejoras a la web que estamos creando. Creando una nueva página con un formulario de contacto.

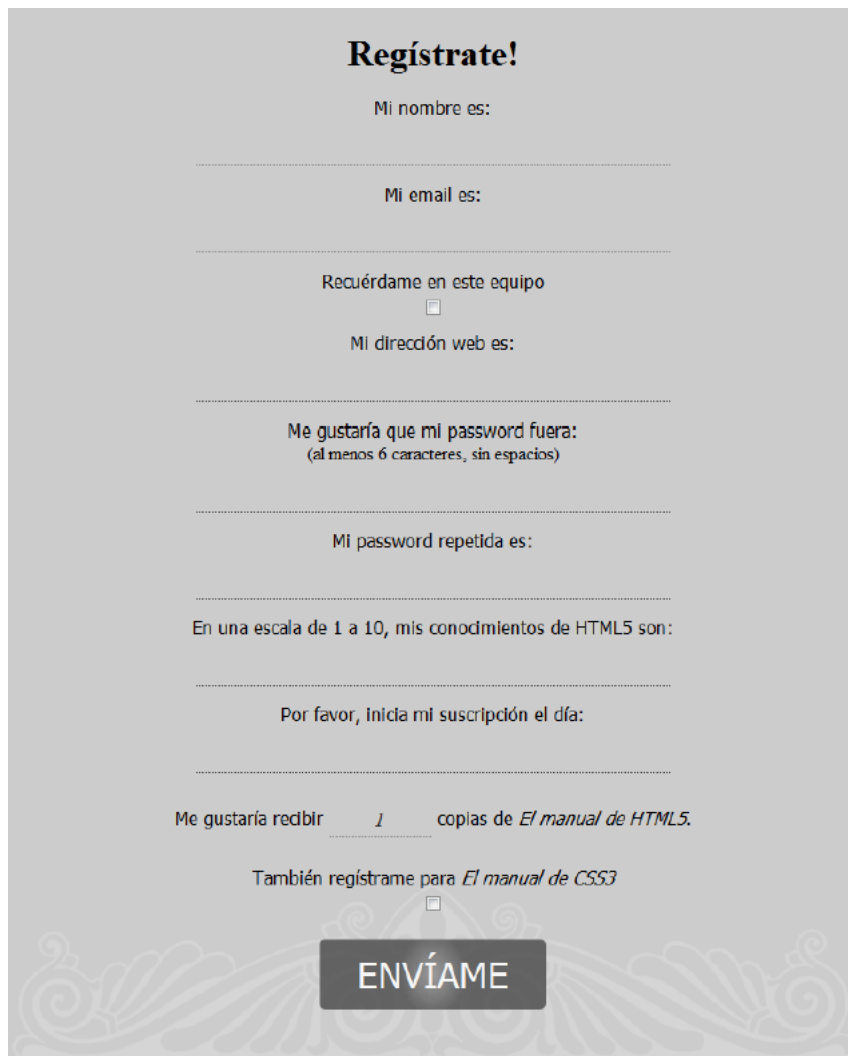
1.1. Mejorar formulario.

Vamos a ver a continuación un ejemplo de formulario de registro codificado con los elementos antiguos existentes en HTML4/XHTML. Después, a medida que vayamos avanzando, mejoraremos este formulario aplicando los nuevos elementos que vayamos estudiando.

```
<form id="registro" method="post">
  <h1>Regístrate!</h1>
  <ul>
    <li>
      <label for="nombre">Mi nombre es:</label>
      <input type="text" id="nombre" name="nombre">
    </li>
    <li>
      <label for="email">Mi email es:</label>
      <input type="text" id="email" name="email">
    </li>
    <li>
      <label for="recuerdame">Recuérdame en este equipo</label>
      <input type="checkbox" value="true" id="recuerdame">
    </li>
    <li>
      <label for="url">Mi dirección web es:</label>
      <input type="text" id="url" name="url">
    </li>
    <li>
      <label for="password">Me gustaría que mi password
        ↪ fuera:</label>
      <span>(al menos 6 caracteres, sin espacios)</span>
      <input type="password" id="password" name="password">
    </li>
    <li>
      <label for="conocimientos">En una escala de 1 a 10, mis
        ↪ conocimientos de HTML5 son:</label>
      <input type="text" id="conocimientos" name="conocimientos">
    </li>
  </ul>
</form>
```

```
</li>
<li>
  <label for="fechaInicio">Por favor, inicia mi suscripción el
    ↪ día:</label>
  <input type="text" id="fechaInicio" name="fechaInicio">
</li>
<li>
  <label for="cantidad">Me gustaría recibir <input type="text"
    ↪ name="cantidad" id="cantidad" value="1"> copias de
    ↪ <cite>El manual de HTML5</cite>.</label>
</li>
<li>
  <label for="tambienCSS3">También reístrame para <cite>El
    ↪ manual de CSS3</cite></label>
  <input id="tambienCSS3" name="tambienCSS3" type="checkbox">
</li>
<li>
  <input type="submit" id="registro-submit" value="Envíame"/>
</li>
</ul>
</form>
```

Este formulario *registro.html* lo podéis encontrar en el material de la práctica. Como así las imágenes utilizadas y el css aplicado al formulario.



Regístrate!

Mi nombre es:

Mi email es:

Recuérdame en este equipo ☐

Mi dirección web es:

Me gustaría que mi password fuera:
(al menos 6 caracteres, sin espacios)

Mi password repetida es:

En una escala de 1 a 10, mis conocimientos de HTML5 son:

Por favor, inicia mi suscripción el día:

Me gustaría recibir copias de *El manual de HTML5*.

También regístrate para *El manual de CSS3* ☐

ENVÍAME

Figura 1: Formulario

Para ello utilizaremos los nuevos tipos de **<input>** ofrecidos por HTML5. - search - email - url - tel - number (min, max, step) - range (min, max, step) - color - date, month, week, time, datetime, datetime-local (min, max)

Nuevos atributos para los campos del formulario: - required - placeholder - pattern - multiple - form - autofocus - readonly

Otros tipos de campos de HTML5: - progress: Para describir el estado actual de un proceso. Ejemplo una descarga - meter: Elemento cuyo rango se conoce. Ejemplo espacio libre en un disco - datalist: se utiliza junto a otros elementos como un *select* - output: Visualizar el resultado de un cálculo.

1.1.1. API Forms

Existen diferentes formas de aprovechar el proceso de validación en HTML5: podemos usar los tipos de campo para activar el proceso de validación por defecto (por ejemplo, email) o convertir un campo como text (o cualquier otro) en un campo obligatorio usando el atributo required. También podemos crear campos con validaciones especiales usando el atributo pattern para personalizar requisitos de validación. Sin embargo, cuando se trata de aplicar mecanismos complejos de validación (por ejemplo, combinando campos o comprobando los resultados de un cálculo) deberemos recurrir a nuevos recursos provistos por esta API.

1.1.1.1. Personalizar los mensajes de error. Podemos crear nuestras propias validaciones con mensajes de validación personalizados usando el método `setCustomValidity(mensaje)`. El funcionamiento es muy sencillo, si en el momento del envío alguno de los campos tiene un mensaje personalizado no vacío, el navegador entenderá que ha habido un error de validación y mostrará el mensaje en el campo indicado. El error se eliminará cuando llamemos a la función `setCustomValidity` pasando un mensaje vacío: `setCustomValidity("")`.

```
function iniciar() {
    registro=document.getElementById("registro");
    registro.addEventListener("invalid", accionInvalid, true);
}

function accionInvalid(evento) {
    /* obtenemos el elemento que no ha pasado la validación */
    var elemento = evento.target;
    elemento.style.background="yellow";
}
window.addEventListener("load", iniciar, false);
```