**UNIR - Universidad Internacional de La Rioja**

**Maestría en Dirección e Ingeniería de Sitios Web**

Curso:

Computación Cliente y Servidor

Actividad individual:

Realización de una Página Web Utilizando Javascript

Alumno: Luis de la Garza González

Matrícula: 5589438-499539

22 de julio de 2024

**ÍNDICE**

[**Introducción** 3](#_Toc170727548)

[**Objetivos** 4](#_Toc170727549)

[**Descripción** 4](#_Toc170727550)

[**Utilización de etiquetas HTML5.** 5](#_Toc170727551)

[**Campos de por lo menos 5 tipos.** 5](#_Toc170727552)

[**Funcionalidad, comprobaciones en PHP** 5](#_Toc170727553)

[**Manejo de errores** 6](#_Toc170727554)

[**Explicación del código** 8](#_Toc170727555)

[**Conclusiones** 13](#_Toc170727556)

Actividad 1.- Realización de una página web utilizando PHP

**Introducción**

El sitio Web implementado para la presente práctica se puede acceder en:

<https://yellow-mink-580191.hostingersite.com/index.html>

La codificación se anexó a la tarea como el archivo zip:

formularioPHP-main.zip

El cual incluye la siguiente estructura y archivos:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Objetivos**

Practicar y adquirir soltura de los conocimientos de Javascript explicados en la asignatura.

**Descripción**

Se Realiza una página web una página web utilizando JavaScript con un formulario que se debe validar. Los requisitos son los siguientes:

* El formulario debe estar generado con HTML5, no con JavaScript.
* El formulario contendrá campos de por lo menos 5 tipos diferentes (por ejemplo, texto, fecha, contraseña, etc.).
* Es obligatorio el uso de JavaScript para realizar comprobaciones que no sean posibles en HTML5. Por ejemplo, se pueden validar correos electrónicos, direcciones de sitios web, formato del DNI español con su letra, etc.

Cuando se entregue el formulario se deberá:

* Generar un informe con las validaciones realizadas y mostrarlo por pantalla.
* Al final del código se incluirá un comentario de no más de 200 palabras que explique el funcionamiento del código implementado.
* Se podrá utilizar captura de eventos con JQuery o nativa JavaScript (p.ej. onmouseover) a elección del alumno, pero no se podrán mezclar ambas tecnologías.

**Utilización de etiquetas HTML5.**

Ver archivo formato.php (incluido en el archivo formularioPHP-main.zip)

**Campos de por lo menos 5 tipos.**

Se utilizaron campos de: texto, email, teléfono, fecha y url.

**Funcionalidad, comprobaciones en javaScript**

Todos los campos fueron validados (texto, email, teléfono, fecha y url)

Ver archivo validación.js (incluido en el archivo formularioJS-main.zip)

**Manejo de errores**

• Si hubiera algún campo erróneo, informar de los errores detectados y redirigir al usuario al formulario para que lo pueda cumplimentar debidamente.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteLa siguiente imagen muestra los errores detectados.

• Si todos los campos son correctos, generar un informe con las validaciones realizadas y mostrarlo por pantalla.

La siguiente imagen muestra el informe generado al final del formulario.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Explicación del código**

A continuación se muestra el archivo “validacion.js”, explicando el funcionamiento del código implementado para la validación en los comentarios en verde.

// validacion.js

const btnEnviar = document.getElementById("btnEnviar");

// generamos una función flecha para validar el formulario

const validarFormulario = (e) => {

    e.preventDefault();

// Limpiamos el reporte de la validación, en el caso de que haya un reporte anterior.

    const miElemento = document.getElementById("reporte");

    miElemento.innerHTML = "";

// leemos la información cargada en el formulario

    var autor = document.forms["formulario"]["autor"].value;

    var email = document.forms["formulario"]["email"].value;

    var telefono = document.forms["formulario"]["telefono"].value;

    var libro = document.forms["formulario"]["libro"].value;

    var fecha = document.forms["formulario"]["fecha"].value;

    var resumen = document.forms["formulario"]["resumen"].value;

    var url = document.forms["formulario"]["url"].value;

// inicializamos la variable error a false

    var error = false;

// la validación la vamos a realizar con expresiones regulares

    var expRegular = "";

    expRegular = /^[a-zA-ZàáâäãåąčćęèéêëėįìíîïłńòóôöõøùúûüųūÿýżźñçčšžÀÁÂÄÃÅĄĆČĖĘÈÉÊËÌÍÎÏĮŁŃÒÓÔÖÕØÙÚÛÜŲŪŸÝŻŹÑßÇŒÆČŠŽ∂ð ,.'-]+$/u;

// si encontramos un error, lo indicamos junto al campo detectado.

    if (autor === null || autor.length === 0 || ! expRegular.test(autor)) {

        document.getElementById('autorErr').textContent = 'Por favor ingresa el nombre del autor.';

        error = true;

    } else {

        document.getElementById('autorErr').textContent = "";

    }

    expRegular = /^[a-zA-Z0-9.\_-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}$/

    if (expRegular.test(email))

        document.getElementById('emailErr').textContent = "";

    else {

        document.getElementById('emailErr').textContent = 'Por favor ingresa un email válido.';

        error = true;

    }

    // Validamos el número de teléfono

    expRegular = /^[0-9]\*$/;

    if (telefono === null || telefono.length != 10 || !expRegular.test(telefono)) {

        document.getElementById('telefonoErr').textContent = 'Por favor ingresa un número telfónico válido de 10 dígitos.';

        error = true;

    } else {

        document.getElementById('telefonoErr').textContent = "";

           }

    // Validando el nombre del libro

    expRegular = /^[a-zA-ZÀ-ž0-9\_\sñáéíóúÁÉÍÓÚ]\*$/;

    if (libro === null || libro.length == 0 || !expRegular.test(libro)) {

        document.getElementById('libroErr').textContent = 'Por favor ingresa el nombre del libro.';

        error = true;

    } else {

        document.getElementById('libroErr').textContent = "";

    }

    // Validando la fecha de publicación

    // alert("Fecha: " + fecha);

    expRegular = /^\d{4}\-\d{2}\-\d{2}$/;

    // if (fecha === null || fecha.length == 0 || ! expRegular.test(fecha)) {

        if (fecha === null || fecha.length == 0 || ! expRegular.test(fecha)) {

        document.getElementById('fechaErr').textContent = 'Por favor ingresa una fecha válida de publicación del libro.';

        error = true;

    }

    else {

        document.getElementById('fechaErr').textContent = "";

    }

    // Validando el resumen

    if (resumen === null || resumen.length == 0) {

        document.getElementById('resumenErr').textContent = "El resumen del libro es requerido";

        error = true;

    }

    else {

        document.getElementById('resumenErr').textContent = "";

    }

    // Validando la URL

    expRegular = /\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i;

    if (url === null || url.length == 0 || ! expRegular.test(url)) {

        document.getElementById('urlErr').textContent = "Un URL válido es requerido, debe iniciar con https o ftp";

        error = true;

    } else {

        document.getElementById('urlErr').textContent = "";

    }

    // Si todas las comprobaciones son correctas, se presenta el reporte de campos validados manipulando el DOM

    if (!error) {

            param = document.createElement("h2");

            node = document.createTextNode("Los datos se han validado exitosamente 👍");

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Autor: " + autor);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Email: " + email);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Teléfono: " + telefono);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Nombre del libro: " + libro);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Fecha de publicación : " + fecha);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Resumen : " + resumen);

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Website del libro : " + url );

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

// si hay errores lo informamos manipulando el DOM

        } else {

            param = document.createElement("h2");

            node = document.createTextNode("Hay información inválida ❌");

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

            param = document.createElement("p");

            node = document.createTextNode("Por favor corrija los datos!");

            param.appendChild(node);

            document.getElementById('reporte').appendChild(param);

        }

    }

btnEnviar.addEventListener("click", validarFormulario);

**El siguiente texto se encuentra comentado al final del archivo validación.js (incluido en el archivo formularioJS-main.zip).**

Los comentarios se presentan en color verde y el código en azul.

En la siguiente línea de código definimos el elemento btnEnviar el cual nos servirá para escuchar el evento clic (ver última línea del código)

const btnEnviar = document.getElementById("btnEnviar");

A continuación, creamos una función flecha que nos servirá para validar el formulario:

const validarFormulario = (e) => {

e.preventDefault();

const miElemento = document.getElementById("reporte");

Limpiamos el reporte de la validación.

miElemento.innerHTML = "";

y leemos los datos cargados en el formulario:

var autor = document.forms["formulario"]["autor"].value;

var email = document.forms["formulario"]["email"].value;

var telefono = document.forms["formulario"]["telefono"].value;

var libro = document.forms["formulario"]["libro"].value;

var fecha = document.forms["formulario"]["fecha"].value;

var resumen = document.forms["formulario"]["resumen"].value;

var url = document.forms["formulario"]["url"].value;

Vamos a validar con expresiones regulares, antes inicializamos la variable “error” a false, si encontramos algún error en la validación le asignamos true a dicha variable

var error = false;

var expRegular = "";

Ya que todas las validaciones son similares, solamente explicaremos la relativa al campo email:

expRegular = /^[a-zA-Z0-9.\_-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}$/

if (expRegular.test(email))

document.getElementById('emailErr').textContent = "";

else {

document.getElementById('emailErr').textContent = 'Por favor ingresa un email válido.';

error = true;

}

Si todas las comprobaciones son correctas, se presenta el reporte de campos validados

if (!error) {

param = document.createElement("h2");

node = document.createTextNode("Los datos se han validado exitosamente 👍");

param.appendChild(node);

document.getElementById('reporte').appendChild(param);

param = document.createElement("p");

node = document.createTextNode("Autor: " + autor);

param.appendChild(node);

document.getElementById('reporte').appendChild(param);

… y seguimos desplegando todos los campos del formulario con sus valores.

En el caso de que exista algún error, éste se despliega junto al campo en cuestión y adicionalmente se genera una nota al final del formulario:

} else {

param = document.createElement("h2");

node = document.createTextNode("Hay información inválida ❌");

param.appendChild(node);

document.getElementById('reporte').appendChild(param);

param = document.createElement("p");

node = document.createTextNode("Por favor corrija los datos!");

param.appendChild(node);

document.getElementById('reporte').appendChild(param);

}

}

Aquí escuchamos el clic soble el botón “btnEnviar:

btnEnviar.addEventListener("click", validarFormulario);

**Conclusiones**

JavaScript es un lenguaje de programación Web del lado del cliente o scripting que permite implementar funciones complejas en páginas web, incluyendo contenido dinámico. Trabaja en conjunto con HTML y CSS.

Algunas ventajas de javaScript son:

* Velocidad: JavaScript se ejecuta directamente en el navegador del cliente, lo que lo hace rápido. No depende de las llamadas al servidor backend a menos que se necesiten recursos externos.
* Simplicidad: Inspirado en la sintaxis de Java, JavaScript es relativamente fácil de aprender en comparación con otros lenguajes como C++.
* Popularidad: JavaScript es omnipresente en la Web y se usa cada vez más en el backend a través de tecnologías como Node.js. Hay abundantes recursos para aprender JavaScript, y su tracción sigue creciendo.
* Interoperabilidad: A diferencia de algunos lenguajes de scripting, JavaScript se puede insertar en cualquier página web. También es compatible con otros lenguajes como Perl y PHP.
* Interfaces enriquecidas: JavaScript habilita funciones como la funcionalidad de arrastrar y soltar y controles deslizantes, lo que mejora las interfaces de usuario.
* Funcionalidad ampliada: Los desarrolladores pueden ampliar la funcionalidad de la página web utilizando complementos de terceros.
* Versatilidad: Con Node.js, puede crear aplicaciones completas, desde el frontend hasta el backend, utilizando únicamente JavaScript y nodeJS que es una librería de JavaScript para programación del lado del servidor.

Sin embargo, también es esencial tener en cuenta las desventajas:

* Seguridad del lado del cliente: Dado que JavaScript se ejecuta en el lado del cliente, los errores y los descuidos pueden explotarse maliciosamente. Algunos usuarios incluso optan por deshabilitar JavaScript por completo.
* Compatibilidad con navegadores: Los diferentes navegadores pueden interpretar el código JavaScript de manera diferente, aunque esta diferencia es mínima hoy en día.

Por todo lo anterior, considero que javaScript es una muy buena opción para la programación del lado del cliente y también del lado del servidor utilizando nodeJS.