Objetivos

Poner en práctica los conceptos básicos relacionados con las tecnologías JavaScript y AJAX.

Jose Manuel Pinillos Rubio

pinillosrubio@gmail.com

DESARROLLO DE APLICACIONES EN RED

Tecnologías JavaScript y AJAX

ÍNDICE

[ÍNDICE 2](#_Toc149894561)

[INTRODUCCIÓN 3](#_Toc149894562)

[PROCESO DE DESARROLLO 4](#_Toc149894563)

[Concepción de la idea 4](#_Toc149894564)

[Implementación de la estructura 4](#_Toc149894565)

[Diseño 4](#_Toc149894566)

[CONCLUSIONES 5](#_Toc149894567)

**INTRODUCCIÓN**

En el ámbito de la Ingeniería Informática, el desarrollo de aplicaciones en red desempeña un papel fundamental, exigiendo la integración de tecnologías modernas para garantizar un rendimiento eficiente y una experiencia del usuario óptima. En este contexto, el presente trabajo se centra en la aplicación práctica de los conocimientos adquiridos en las tecnologías JavaScript y AJAX, dos pilares esenciales en el desarrollo web contemporáneo.

En la primera parte, me centro en la programación en JavaScript, abordando una serie de desafíos diseñados para poner a prueba nuestra destreza en la manipulación y gestión de datos. Desde la detección de palíndromos hasta la comparación de números, exploramos las diversas facetas de este lenguaje de programación, destacando su versatilidad y capacidad para resolver problemas variados.

Posteriormente, trabajamos con AJAX (*Asynchronous* JavaScript and XML), una tecnología que revoluciona la interacción entre el usuario y el servidor al permitir solicitudes asíncronas sin necesidad de recargar la página. Mediante una aplicación práctica, hemos abordado escenarios reales, implementando funcionalidades que demuestran la utilidad de AJAX en la obtención de datos de manera dinámica y sin interrupciones perceptibles para el usuario.

La implementación de estos conceptos se ha llevado a cabo en una página web diseñada con HTML y CSS, proporcionando no solo una funcionalidad robusta sino también una interfaz atractiva y amigable para el usuario. A lo largo de esta actividad, detallaré cada ejercicio resuelto, exponiendo el código y los resultados obtenidos, con el objetivo de ofrecer una visión completa de nuestro proceso de desarrollo y aprendizaje.

**JavaScript**

En este apartado describiré el proceso de desarrollo que he seguido para la resolución de los ejercicios planteados de JavaScript.

## **Detectar si la cadena de entrada en un palíndromo**

El código implementa una función llamada ejer1() que se encarga de verificar si una cadena de texto introducida es un palíndromo o no.

1. **Obtención de la cadena de texto:**

let cadena = document.getElementById("ejercicio1").value;

La función comienza obteniendo el valor de un elemento de la página HTML con el id "ejercicio1". Presumiblemente, este elemento es un cuadro de texto (input) en el que el usuario ingresa una cadena de texto.

1. **Validación de entrada vacía:**

if(cadena == "") {

document.getElementById("resultado1").innerHTML = `<b style="color:red">Por favor, introduce algo de texto.</b>`;

}

Se verifica si la cadena de texto está vacía. Si es así, se actualiza el contenido de otro elemento HTML con el id "resultado1" para mostrar un mensaje en rojo indicando que se debe introducir algo de texto.

1. **Manipulación de la cadena:**

cadena = cadena.replace(/\s/g, "").toLowerCase();

Si la cadena no está vacía, se realiza un preprocesamiento en la cadena eliminando los espacios en blanco y convirtiendo todos los caracteres a minúsculas. Esto es importante para asegurar que la comparación de palíndromos no sea sensible a mayúsculas/minúsculas y no se vea afectada por los espacios en blanco.

1. **Verificación de palíndromo:**

if(cadena === cadena.split("").reverse().join("")) {

document.getElementById("resultado1").textContent = "Es palindromo.";

}else{

document.getElementById("resultado1").textContent = "No es palindromo.";

}

Se compara la cadena manipulada con su versión invertida. Si son idénticas, se concluye que la cadena es un palíndromo y se actualiza el contenido del elemento con el id "resultado1" para indicar que es un palíndromo. En caso contrario, se informa que no es un palíndromo. La comparación es estricta (===), lo que significa que la igualdad debe ser tanto en valor como en tipo.

## **Comparar dos números y detectar cual es el mayor**

Este código en JavaScript implementa una función llamada ejer2 que compara dos números ingresados por el usuario y muestra un mensaje indicando cuál de los dos números es mayor o si son iguales.

1. **Obtención de los números del formulario:**

let num1 = document.getElementById("num1").value;

let num2 = document.getElementById("num2").value;

La función comienza obteniendo los valores de dos campos de texto en un formulario. Presumiblemente, estos campos de texto tienen los IDs "num1" y "num2", respectivamente, y se espera que el usuario ingrese números en estos campos.

1. **Validación de entrada vacía:**

if(num1 == "" || num2 == "") {

document.getElementById("resultado2").innerHTML = `<b style="color:red">Por favor, introduzca dos números.</b>`;

}

Se verifica si al menos uno de los campos de texto está vacío. Si es así, se actualiza el contenido de otro elemento HTML con el id "resultado2" para mostrar un mensaje en rojo indicando que se deben introducir dos números.

1. **Comparación de números:**

if(num1 == num2) {

document.getElementById("resultado2").textContent = "Los números son iguales.";

}else{

if(num1 < num2) {

document.getElementById("resultado2").innerHTML = `<b>${num2}</b> es el número mayor.`;

}else{

document.getElementById("resultado2").innerHTML = `<b>${num1}</b> es el número mayor.`;

}

}

Si ambos campos de texto tienen valores, se procede a comparar los números. Primero, se verifica si los números son iguales. Si son iguales, se muestra un mensaje indicando que los números son iguales. Si no son iguales, se compara cuál de los dos números es mayor y se actualiza el contenido del elemento con el id "resultado2" para indicar cuál es el número mayor. El formato HTML utilizado para resaltar el número mayor puede ser útil para hacer que el resultado sea más visible.

## **Detectar las vocales que aparecen en un texto**

El diseño de nuestro blog se basó en el uso de etiquetas HTML para estructurar el contenido de manera coherente y semántica. Se aplicaron etiquetas de encabezado (<h1>, <h2>, etc.) para organizar los textos, <img> para la integración de imágenes y <a> para enlazar a diferentes secciones y páginas. El archivo CSS (style.css) desempeñó un papel fundamental en la determinación de la tipografía, el tamaño y la disposición del contenido, lo que asegura que el blog se visualice adecuadamente en diferentes tamaños de pantalla.

## **Contabilizar las vocales que aparecen en un texto**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS, lo que brinda a los visitantes una introducción a estos conceptos esenciales. La idea se alinea con el propósito de esta actividad académica, que es proporcionar a los estudiantes una comprensión profunda de la programación web.

**AJAX**

En este apartado describiré el proceso de desarrollo que he seguido para la resolución de los ejercicios planteados de AJAX.

## **Detectar la URL de la página cargada en un cuadro de texto**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS, lo que brinda a los visitantes una introducción a estos conceptos esenciales. La idea se alinea con el propósito de esta actividad académica, que es proporcionar a los estudiantes una comprensión profunda de la programación web.

## **Descargar mediante peticiones AJAX el contenido correspondiente de la URL**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS.

### **Contenidos del archivo**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS.

### **Estados de la petición**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS.

### **Contenido de las cabeceras**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS.

### **Código y texto de respuesta del servidor**

La concepción de la idea detrás de nuestro blog se centró en la divulgación de contenidos, con un enfoque particular en la explicación de los elementos HTML y CSS.

**CONCLUSIONES**

El proceso de desarrollo de nuestro blog ha resultado en una experiencia enriquecedora que nos ha permitido aplicar los conceptos fundamentales de programación web de manera práctica, logrando crear un sitio web funcional y atractivo que cumple con los objetivos de la actividad.

Los conocimientos y habilidades adquiridos en esta actividad servirán como una base sólida para futuros desarrollos y proyectos web. Esta actividad representa un primer paso en la creación de sitios web desde cero y la divulgación de contenidos en línea.

Algo que no hemos llegado a conseguir por falta de tiempo, es que el diseño del blog sea responsivo, con lo que solo se puede visualizar adecuadamente a través de un ordenador.