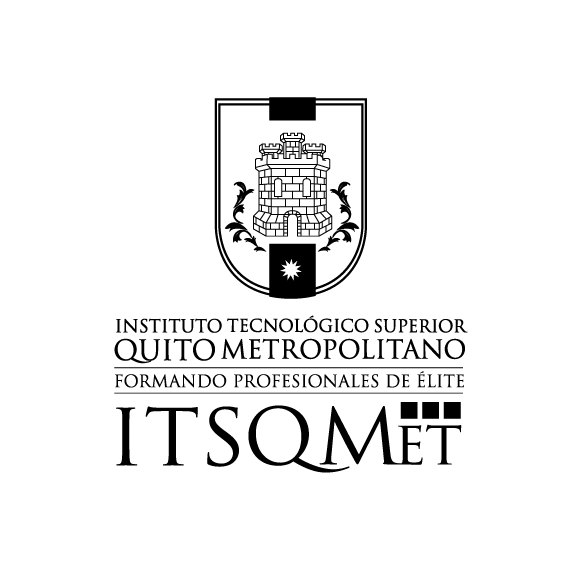


­

**INSTITUTO SUPERIOR universitario**

**QUITO METROPOLITANO**



**CUADERNO DIGITAL DE LA ASIGNATURA**

**DESARROLLO DE APLICACIONES DE INTERNET**

**ING. JULIO ROSERO**

**CRISTIAN ESCANTA**

**ABRIL – SEPTIEMBRE 2023**

**CONTENIDO**

INDICACIONES GENERALES 3

CONTENIDOS ESPECIFICOS 5

1. UNIDAD I 5

TEMAS 5

SUBTEMAS 5

REFLEXIONES: 5

CONCLUSIONES DE APRENDIZAJE: 5

PREGUNTAS DE EVALUACIÓN: 6

PROYECTO DE APLICACIÓN DE A ASIGNATURA 7

OBJETIVO DE PROYECTO DE APLICACIÓN 7

CARACTERISTICAS DEL PROYECTO 7

**BIBLIOGRAFÍA** 8

**Índice de Figuras**

Ilustración 1 Auditoría Informática Fuente: Autor Propio 5

**Índice de Tablas**

[Tabla 1 Tabla ejemplo Fuente: Autor Propio 5](#_Toc73347446)

INDICACIONES GENERALES

* Respetar el formato establecido por el docente.
* Las imágenes tienen que ir con descripciones y fuentes, con normas APA 7ma edición.
* El presente documento tiene por defecto configurado los diferentes recursos de normas APA, en la sección de estilos, usarlos para los diferentes, temas y subtemas desarrollados en clase, además de los márgenes y encabezados y pies de página.
* La fuente de letra es 12 y estilo Times News Román.
* Como recurso estratégico puede usar organizadores gráficos, resúmenes, análisis y síntesis, puede incluir además capturas de sus anotaciones en físico.
* Debe existe un orden coherente dentro de la estructura del cuaderno.
* Generar reflexiones por parte del estudiante en relación a la asignatura y sus contenidos.
* Puede anexar preguntas y respuestas dirigidas de relevancia, consideradas dentro de la clase como parte de su aprendizaje.
* Incluye información por iniciativa propia del estudiante
* Genera correcciones de trabajos o pruebas dentro del cuaderno generando un aprendizaje significativo.
* Conclusiones de aprendizaje por unidad.
* No existe un número de páginas especifico, todo dependerá del aprendizaje del estudiante.

CONTENIDOS ESPECIFICOS

El profesor debe entregar al estudiante durante la primera clase los contenidos específicos por unidad que serán considerados para el desarrollo del cuaderno digital que el estudiante desarrollara por modulo.

1. UNIDAD I

HISTORIA Y EVOLUCIÓN DE LA WEB

La WorldWide Web, comúnmente conocida como la Web, es un  
sistema de información global que se comunica a través de  
Internet.

L web ha evolucionado desde su creación de 1989, en 1991, Tim Berners-Lee creó el primer sitio web, aun podemos encontrar el sitio web en el siguiente link.

<http://info.cern.ch/hypertext/WWW/TheProject.html>

¿COMO FUNCIONA EL INTERNET?

Internet es una red mundial conectada que permite el intercambio de información y la comunicación en todo el mundo.

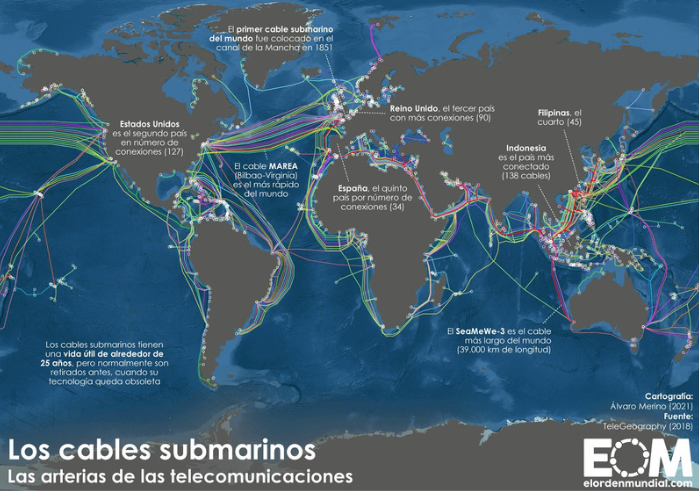
Esto no ayuda para que las computadoras estén comunicadas entre sí en internet, los protocolos que se utilizan son TCP/IP o HTTP.

TCP/IP es el principal que se utiliza en Internet para la transmisión de datos.

HTTP se utiliza para la transmisión de páginas web.

MAPA DE CABLES SUBMARINOS

Conectan diferentes partes del mundo y permiten la transmisión de datos a través de internet.



Los cables submarinos son la columna vertebral de la infraestructura del internet, han sido utilizados desde la década de 1850 para la comunicación telegráfica a través de los océanos.

Con el advenimiento del internet en la década de 1990, la necesidad de cables submarino se volvió aún mas importante para la transmisión de datos.

A medida que los usuarios aumentan, también lo hace la demanda de ancho de banda y la necesidad de más cables submarinos.

En resumen, los cables submarinos son esenciales para mantener la conectividad y el flujo de información en el mundo moderno.

ARQUITECTURA DE LA WEB

Se refiere a la estructura u organización de los componentes que conforman la World Wide Web, la misma que esta compuesta por una serie de protocolos, estándares y tecnologías que permite la transmisión de información en línea.

Componentes clave de la arquitectura de la web incluyen:

* Protocolos de internet
* Lenguajes de marcado
* Servidores Web
* Navegadores Web

La arquitectura cliente-servidor es la base de la Web, este modelo solicita información o recursos de los servidores web,

La web está impulsada por una amplia gama de tecnologías, incluyendo lenguajes de programación del lado del servidor.

USO DE INTERNET

Según la Unión Internacional de Telecomunicaciones, más de la mitad de la población mundial utiliza internet.

En América Latina, el número de usuarios de internet aumentó un 50% en los últimos 5 años.

Es importante conocer los riesgos y desafíos que con lleva el uso de internet:

* Riesgo de privacidad y seguridad en línea
* Ciberacoso y otros problemas de seguridad en línea
* Disminución de las habilidades sociales

HISTORIA DE LOS NAVEGADORES

INTRODUCCIÓN A HTML5

HTML5 es la última versión del lenguaje de marcado para la web, es un lenguaje que se utiliza para crear y diseñar páginas web.

HTML5 introduce nuevas etiquetas semánticas que describe mejor la estructura del contenido de una página web.

Formularios mejorados HTML5 ofrece nuevas opciones para la creación de formularios, lo que facilita la entrada de datos y la validación de los mismos.

Incluye la validación de campos en tiempo real, la selección de fecha y hora, la carga de archivos, entre otras.

HTML5 es la última versión del lenguaje de marcado estándar utilizado para crear y estructurar contenido en la web. Introducido por primera vez en 2014, HTML5 ofrece una serie de características nuevas y mejoradas en comparación con sus versiones anteriores.

Una de las principales mejoras de HTML5 es su capacidad para admitir contenido multimedia sin necesidad de complementos adicionales, como Flash. Ahora es posible incrustar audio y video directamente en una página web utilizando las etiquetas <audio> y <video>, lo que facilita la reproducción de contenido multimedia sin problemas en diferentes dispositivos y navegadores.

Otra característica clave de HTML5 es su soporte para gráficos y animaciones vectoriales. Con la introducción de la etiqueta <canvas>, se puede dibujar gráficos, gráficos interactivos y animaciones utilizando JavaScript. Esto ha abierto nuevas posibilidades en el desarrollo de juegos en línea y aplicaciones ricas en gráficos.

ELEMENTOS BÁSICOS DE HTML

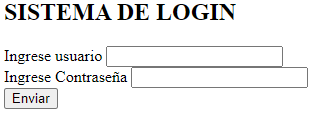
PRÁCTICA

FORMULARIOS

Para crear formularios se debe usar la etiqueta <form>, esta etiqueta define la estructura del formulario y su formulario.

Usa la etiqueta <input> para crear campos de entrada de texto. Puedes especificar el tipo de entrada de texto (por ejemplo, texto, número o correo electrónico), y también puedes establecer

una etiqueta de identificación para el campo de entrada.



**CODIGO**

<h2>SISTEMA DE LOGIN</h2>

        <form action="formLogin">

            <label for="user">Ingrese usuario</label>

            <input type="text" id="user"><br>

            <label for="pass">Ingrese Contraseña</label>

            <input type="password" id="pass"><br>

            <input type="button" onclick="login()" value="Enviar">

INTRODUCCIÓN A CSS

CSS, siglas de "Cascading Style Sheets" (Hojas de estilo en cascada), es un lenguaje de estilo utilizado para definir la presentación y el aspecto visual de documentos HTML, XHTML y XML, incluyendo colores, diseños, fuentes, bordes, márgenes y otros aspectos visuales.

En resumen, CSS es una herramienta esencial para diseñadores  
web y desarrolladores que desean crear páginas web visualmente

atractivas y funcionales.

SINTAXIS CSS

La sintaxis se compone de tres elementos básicos:

* Los selectores
* Las propiedades
* Los valores

**SELECTORES**

Pueden ser el nombre de una etiqueta HTML, un identificador o una clase, o pueden combinar varios selectores para seleccionar elementos específicos. Los selectores se escriben seguidos de un bloque de llaves ({}) que contendrá las propiedades y valores de

estilo.

**PROPIEDADES**

Las propiedades indican qué aspecto del elemento HTML se está estilizando. Cada propiedad se escribe seguida de dos puntos (:), y se asigna un valor específico a la propiedad.  
**VALORES**

Los valores son los ajustes específicos que se aplican a una propiedad. Los valores pueden ser números, colores, palabras clave o funciones, dependiendo de la propiedad que se esté utilizando.

MANEJO DE BORDES Y ESTILOS

Los bordes son una forma de mejorar la apariencia visual de un elemento, y pueden ayudar a enfatizar su contenido:

* Border-width establece el ancho de borde
* Border-style establece el estilo del borde
* Border-color establece el color del borde
* Border-radius establece el radio de las esquinas del borde

GESTIÓN DE FUENTES E ÍCONOS

En CSS, la gestión de fuentes y estilos se realiza mediante el uso  
de propiedades CSS específicas. Aquí hay algunas propiedades  
clave para la gestión de fuentes y estilos en CSS:

* Font-family fuente que se utiliza para el texto
* Font-size especificar el tamaño de la fuente
* Font-weight especifica el grosor de la fuente
* Font-style utiliza para especificar el estilo de la fuente
* Text-decoration decoración del texto como subrayado o tachado
* Color especificar el color del texto

FRAMEWORKS DE ESTILOS

Los frameworks de estilo son herramientas que se utilizan para acelerar y simplificar el proceso de diseño y desarrollo de sitios web y aplicaciones.

Estos frameworks proporcionan un conjunto predefinido de estilos y componentes de diseño que pueden ser utilizados para construir interfaces de usuario consistentes y atractivas.

Tenemos algunos ejemplos tales como:

* Bootstrap es uno de lo mas populares y ampliamente utilizados
* Tailwind se centra en la utilidad y flexibilidad
* Materialize basado en el diseño de Material Desig de Google
* Bulma código abierto que utiliza clases CSS

INTRODUCCIÓN A JAVASCRIPT

Javascript es un lenguaje de programación diseñado para crear aplicaciones web interactivas.

Es usado como interactividad en páginas web, validación de formularios, animaciones y efectos visuales, y desarrollo de aplicaciones web completa

Algunos frameworks de desarrollo utilizan JavaScript para su

desarrollo, por ejemplo: Angular, React, Vue, etc.

Java es un lenguaje de programación orientado a objetos que se utiliza ampliamente en el desarrollo de aplicaciones de software. Fue creado por Sun Microsystems (ahora propiedad de Oracle) en 1995 y se ha convertido en uno de los lenguajes más populares y utilizados en la industria.

Una de las principales características de Java es su portabilidad. Esto significa que una vez que se escribe un programa en Java, puede ejecutarse en diferentes plataformas y sistemas operativos sin la necesidad de realizar modificaciones significativas. Esto es posible gracias a la máquina virtual de Java (JVM), que interpreta el código Java y lo ejecuta en el entorno de destino.

INSTALACIÓN DE NODE

Para proceder a instalar Nodees necesario dirigirse a su sitio oficial. [https://nodejs.org/es](https://www.canva.com/link?target=https%3A%2F%2Fnodejs.org%2Fes&design=DAFgZbPUiYk&accessRole=viewer&linkSource=document)

Se recomienda elegir siempre las versiones LTS



SINTAXIS BÁSICA DE JAVASCRIPT

TIPOS DE DATOS CON JAVASCRIPT

INICIALIZACIÓN DE VARIABLES

Declaración de datos:

* Varvariable=valor;
* Letvariable=valor;
* Constvariable=valor;

Datos reales:

* varnombre="Juan"
* letedad=30;
* constPI = 3.14;

BUENAS PRÁCTICAS

Es importante utilizar nombres de variables y funciones que reflejen su propósito y funcionalidad para hacer que el código sea más legible y fácil de entender para otros desarrolladores.

Los comentarios son una herramienta muy útil para explicar el

propósito y funcionamiento del código. Es recomendable utilizar comentarios para explicar cómo funciona cada sección de código y qué hace cada función o variable.

Las variables locales pueden causar conflicto y errores en el código, especialmente en aplicaciones más grandes.

En lugar de utilizar la plabra clave “var” para declarar variables, es mejor utilizar “const” y “let”, ya que tienen un alcance más específico.

Ejemplo de declaración de variables:

const usuario=document.getElementById("user")

    const pass=document.getElementById("pass")

OPERADORES EN JAVASCRIPT

OPERADORES ARITMÉTICOS

Se utilizan para realizar operaciones matemáticas básicas, como sumar, restar, multiplicar dividir.

* Suma (+)
* Resta (-)
* Multiplicación (\*)
* División (/)
* Módulo (%)

OPERADORES DE INCREMENTO Y DECREMENTO

Es un tipo de operador que se utiliza para aumentar o disminuir el valor de una variable.

* El operador de incremento “++” aumenta en 1 unidad
* El operador de decremento “--” disminuye en 1 unidad

Estos operadores se pueden utilizar de dos maneras:

* Prefijo: se coloca antes de la variable lo que significa que se incrementa o decrementa el valor de la variable
* Sufijo: se coloca después de la variable lo que significa que se incrementa o decrementa el valor de la variable después de realizar cualquier operación.

OPERADOR DE COMPARACIÓN Y LÓGICOS

En JavaScript, los operadores de comparación se utilizan para comparar dos valores y devolver un resultado de verdad (true o false) que indica si la comparación es verdadera o falsa.

Estos operadores se pueden utilizar para comparar valores de diferentes tipos de datos, incluyendo números, cadenas de texto, objetos y otros tipos de datos.

|  |  |  |
| --- | --- | --- |
| OPERADORES LÓGICOS Y RELACIONALES | DESCRIPCIÓN | EJEMPLO |
| = = | Es igual | A == b |
| = = = | Es estrictamente igual | A === b |
| ¡ = | Es distinto | A ¡= b |
| ¡ = = | Es estrictamente distinto | A ¡== b |
| <,<=. >, >= | Menor, menor o igual, mayor, mayor o igual | A <= b |
| && | Operador and (y) | A && b |
| | | | Operador or (o) | A | | b |
| ¡ | Operador not (not) | ¡a |

SENTENCIAS DE CONTROL

Las sentencias de control son herramientas importantes en JavaScript que permiten controlar el flujo de ejecución del programa.

SENTENCIAS ITERATIVAS

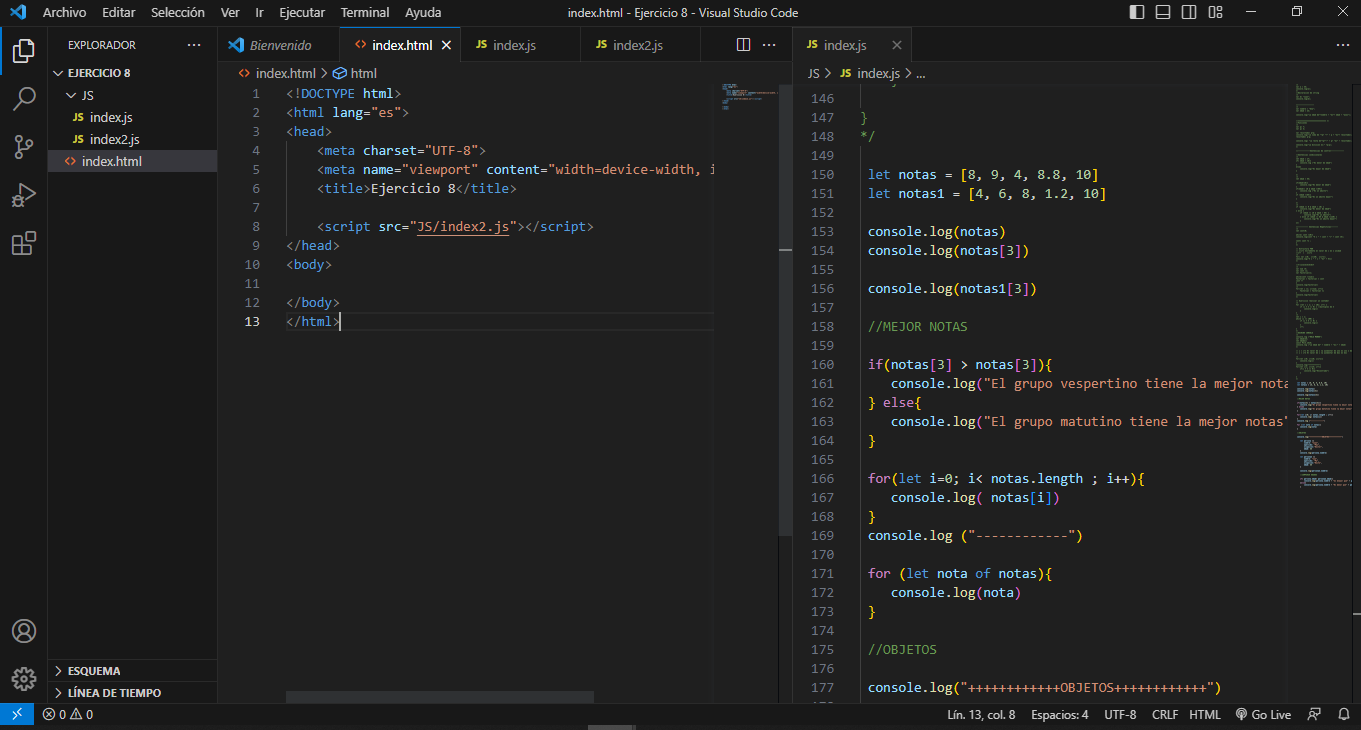
Los bucles permiten repetir un bloque de código varias veces. Los bucles más comunes en JavaScript son:

* While permite repetir un bloque de código mientras se cumpla una condición determinada.
* Do…while ejecuta el bloque de código al menos una vez antes de comprobar.
* Condición for permite repetir un bloque de código determinadas veces.

PARADIGMA POO EN JAVASCRIPT

Es importante entender algunos conceptos de POO:

* Clase es una plantilla que define las propiedades y métodos de un objeto
* Objeto es una instancia de una clase
* Propiedad son las características o atributos de un objeto
* Método son las funciones o comportamientos que tiene un objeto
* Herencia es un mecanismo mediante el cual una clase puede heredar las propiedades y métodos de otra clase



INTRODUCCIÓN A REACT

React es una biblioteca de JavaScript de código abierto diseñada para crear interfaces de usuario (UI) interactivas en aplicaciones de una sola página (SPA).

React ha pasado por varias versiones importantes, y ha crecido para convertirse en una de las bibliotecas de JavaScript más populares y utilizadas. Hoy en día, React es utilizado por

empresas y desarrolladores de todo el mundo.

React ha revolucionado la forma en que se construyen aplicaciones web, y ha hecho que la creación de aplicaciones de una sola página sea más rápida, eficiente y fácil de mantener.

DIFERENCIAS ENTRES REACTJS VS REACT NATIVE

ReactJS y React Native son dos tecnologías diferentes, pero relacionadas, desarrolladas por Facebook.

Ambas comparten una sintaxis similar y se basan en los mismos principios fundamentales, pero tienen diferencias clave en su funcionamiento y objetivo.

ReactJS es una biblioteca de JavaScript de código abierto diseñada para crear interfaces de usuario (UI) interactivas en aplicaciones web.

Algunas de las principales características de ReactJS son su capacidad para manejar datos de manera eficiente, su enfoque en la reutilización de componentes y su capacidad para crear

interfaces de usuario dinámicas y fáciles de mantener.

React Native es un marco de desarrollo móvil de código abierto que utiliza la sintaxis de ReactJS para crear aplicaciones móviles nativas para iOS y Android.

FORMULARIOS

Ejercicios de formularios:

**INDEX.HTML**

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <script src="/JS/index.js"></script>

</head>

<body>

    <header>

        <H1>EJERCICIOS DE FORMULARIOS</H1>

    </header>

    <main>

        <!----Login-->

        <h2>SISTEMA DE LOGIN</h2>

        <form action="formLogin">

            <label for="user">Ingrese usuario</label>

            <input type="text" id="user"><br>

            <label for="pass">Ingrese Contraseña</label>

            <input type="password" id="pass"><br>

            <input type="button" onclick="login()" value="Enviar">

        </form>

        <div id="mensaje1"></div>

        <!----------------CALCULADORA-->

        <hr>

        <h2>CALCULADORA</h2>

        <form>

            <input type="number" id="num1" placeholder="Número 1">

            <input type="number" id="num2" placeholder="Número 2"><br>

            <button type="button" onclick="sumar()">SUMA</button>

            <button type="button" onclick="restar()">RESTAR</button>

            <button type="button" onclick="dividir()">DIVIDIR</button>

            <button type="button" onclick="multiplicar()">MULTIPLICAR</button>

        </form>

        <!----------LISTA DE TAREAS------>

        <h2>LISTA DE TAREA</h2>

        <form>

            <input type="text"  placeholder="Ingrese Tareas" id="tarea">

            <button type="button" onclick="agregarTarea()">Agregar</button>

        </form>

**INDEX.JS**

///////LOGIN////////

function login(){

    const usuario=document.getElementById("user")

    const pass=document.getElementById("pass")

///////////VALIDACION DE CAMPOS QUE NO ENVIEN DATOS VACIOS//////////

if(usuario.value== "" || pass.value == ""){

    document.getElementById("mensaje1").innerHTML ="No se admite campos vacíos"

alert("No se admite vacios")

}else if (usuario.value =="admin" && pass.value=="root"){

    document.getElementById("mensaje1").innerHTML="Acceso correcto"

    alert("Acceso Correcto")

}else{

    document.getElementById("mensaje1").innerHTML="Credemciales Correctas"

    alert("Credenciales incorrectas")}}

////////////CALCULADORA/////////////////

function sumar(){

    const num1 = parseFloat(document.getElementById("num1").value)

    const num2 = parseFloat(document.getElementById("num2").value)

    let suma;

    //VALIDAMOS CAMPOS//////

    if(isNaN(num1) || isNaN(num2)){

        alert("No se admite campos en blanco")

    }else {

        suma=num1+num2

        alert("La suma es:" + suma)}}

///////////////RESTAR//////////

function restar(){

    const num1 =parseFloat(document.getElementById("num1").value)

    const num2 =parseFloat(document.getElementById("num2").value)

let resta;

    ///VALIDAR CAMPOS

    if( isNaN(num1) || isNaN(num2)){

        alert("No se admite campos en blanco")

    }else{

        resta=num1 - num2

        alert("La resta es :" + resta)}}

//////////DIVIDIR///////////

function dividir(){

    const num1 =parseFloat(document.getElementById("num1").value)

    const num2 =parseFloat(document.getElementById("num2").value)

let divi;

if( isNaN(num1) || isNaN(num2)){

    alert("No se admite campos en blanco")

}else{

    divi=num1 / num2

    alert("La divicion es :" + divi)}}

//////////////MULTIPLICAR//////////////

function multiplicar(){

    const num1 =parseFloat(document.getElementById("num1").value)

    const num2 =parseFloat(document.getElementById("num2").value)

let multi;

if( isNaN(num1) || isNaN(num2)){

    alert("No se admite campos en blanco")

}else{

    multi=num1 \* num2

    alert("La multiplicacion es :" + multi)}}

//////////////LISTA DE TAREA///////////////

function agregarTarea(){

    const tarea=document-getElementById("tarea").value

    ////////VALIDAR CAMPOS//////////

    if( tarea == ""){

        alert("No se admiten campos en blanco")

    }else{

        let nuevaTarea = document.createElement("li")

        nuevaTarea.textContent = tarea

        let lista = document.getElementById("lista")

        lista.appendChild(nuevaTarea)}}

**VISUALIZACIÓN**



ARQUITECTURA WEB

En React, los componentes son la base de cualquier aplicación. Un componente es una pieza aislada y reutilizable de la interfaz de usuario que se compone de HTML, CSS y JavaScript.

En la carpeta de componentes, se almacenan todos los componentes de React de la aplicación.

En la carpeta de acciones, se almacenan todas las acciones de Redux de la aplicación.

INTRODUCCIÓN A JSX

JSX hace que el código sea más legible y fácil de entender, ya que combina HTML y JavaScript en una sola sintaxis.

JSX permite utilizar las herramientas de desarrollo HTML en el código de React, lo que se hace más fácil depurar el código.

JSX es más rápido que otras formas de renderizar HTML en React.

DEFINICIÓN DE JSX

Utiliza React para definir la estructura de los componentes de la interfaz de usuario, es una extensión de la sintaxis de JavaScript que permite definir la estructura de los componentes de la interfaz de usuario en React.

RENDERIZACIÓN DE ELEMENTOS

En React, la renderización de elementos es un proceso importante para mostrar contenido de la interfaz de usuario.

La renderización es el proceso de crear una representación visual de un elemento en la interfaz de usuario.

Durante este proceso, React compara el nuevo estado con el estado anterior y determina qué elementos necesitan ser actualizados en la interfaz de usuario.

FORMULARIO DE JS

En React, los formularios son una forma común de interactuar con el usuario. En esta presentación, vamos a hablar sobre cómo crear y manejar formularios en React y cómo se pueden utilizar para crear aplicaciones web modernas.

En React, los eventos de formulario como "submit", "change", "reset", etc., se manejan utilizando funciones de JavaScript.

En React, la validación de formularios se puede realizar utilizando la biblioteca de validación de formularios o escribiendo funciones personalizadas de validación.

CREACIÓN DE FUNCIONES QUE INTERACTÚEN CON EL DOM

Existen muchas librerías y frameworks disponibles en línea que pueden ayudar a extender las funcionalidades de JavaScript. Investiga y selecciona el recurso adecuado para tu aplicación.

Utiliza la documentación del recurso externo para aprender cómo utilizarlo en tu aplicación y cómo crear la función que se necesita.

Una vez que hayas creado la función utilizando el recurso externo, es necesario probarla y depurarla para asegurarse de que funcione correctamente.

El DOM es una estructura en forma de árbol que representa la estructura de una página web en términos de objetos, lo cual es útil para manipular el contenido, estilo y comportamiento de la página web mediante código JavaScript.

Para manipular los elementos del DOM, primero es necesario seleccionarlos, los elementos pueden ser seleccionados mediante distintas funciones, como getElementByld, getElementByClassName y getElementsByTagName.

Estas funciones retornan uno o varios elementos del DOM.

Una vez seleccionado un elemento del DOM, es posible modificar su contenido, esto puede hacer mediante el uso de las propiedades innerHTML y textContent.

La propiedad innerHTML permite cambiar el contenido de un elemento mediante HTML, mientras que la propiedad textContent permite cambiar el contenido mediante texto plano.

REFLEXIONES:

Debemos estar consientes de que el diseño de una página web es la primera impresión que los usuarios tienen de un sitio, la usabilidad de la página es web es vital manteniendo un diseño atractivo y bien estructurado.

Podemos indicar que el aprendizaje de la materia no con lleva a pode implementar en nuestra labor diaria ya sea este en un negocio propio o nuestro lugar de trabajo.

CONCLUSIONES DE APRENDIZAJE:

Es importante el aprendizaje sobre diseño de paginas web la cual no conlleva a llevar un orden una estructura, lógica del contenido es fundamental para el diseño de una página web.

El uso efectivo de CSS en cual podemos elegir estilos colores, los cuales no con lleva a que sean coherentes con la identidad de marca y atractivos para el usuario.

PREGUNTAS DE EVALUACIÓN:

**¿Cuáles son algunas tecnologías comunes utilizadas en el desarrollo de aplicaciones web?**

* HTML (HyperText Markup Language) y CSS (Cascading Style Sheets) para la estructura y el diseño visual de las páginas web.
* JavaScript para la interactividad y la lógica del lado del cliente.
* Frameworks y bibliotecas populares como React, Angular y Vue.js para simplificar el desarrollo y mejorar la eficiencia.

**¿Cuáles son los aspectos clave a considerar al desarrollar una aplicación web segura?**

* Validación y filtrado de datos: Implementar medidas de seguridad para validar y filtrar los datos de entrada para prevenir ataques de inyección, como SQL o XSS.
* Autenticación y autorización: Implementar un sistema de autenticación seguro y gestionar los permisos de acceso para los usuarios.
* Protección contra ataques de denegación de servicio (DoS): Implementar medidas para detectar y mitigar ataques de DoS que podrían afectar la disponibilidad de la aplicación web.

**¿Cómo se puede hacer una página web responsive o adaptable a diferentes dispositivos?**

* Diseño flexible: Utilizar unidades de medida relativas y rejillas flexibles para que los elementos se ajusten automáticamente según el tamaño de la pantalla.
* Media queries: Aplicar estilos diferentes basados en características específicas del dispositivo, como el ancho de la pantalla, la resolución y la orientación.
* Imágenes adaptables: Utilizar imágenes con dimensiones y resoluciones adecuadas para diferentes dispositivos, o utilizar técnicas como el "responsive images" para cargar imágenes optimizadas según el dispositivo.

**¿Qué es el desarrollo de aplicaciones web progresivas (PWA)?**

Las aplicaciones web progresivas son aplicaciones web que utilizan tecnologías modernas para brindar una experiencia similar a la de una aplicación nativa. Se pueden instalar en el dispositivo del usuario y funcionar sin conexión a Internet. Las PWA ofrecen características como notificaciones push, acceso a hardware del dispositivo y capacidad de respuesta rápida, lo que mejora la experiencia del usuario.

**¿Qué es la adaptabilidad móvil y por qué es importante en el desarrollo web?**La adaptabilidad móvil se refiere a la capacidad de un sitio web para adaptarse y funcionar correctamente en diferentes dispositivos móviles, independientemente de su tamaño de pantalla y orientación. Es importante en el desarrollo web debido al creciente uso de dispositivos móviles para acceder a Internet. Al tener un diseño web adaptable, se mejora la experiencia del usuario y se garantiza que el contenido se muestre de manera óptima en una variedad de dispositivos móviles.

PROYECTO DE APLICACIÓN DE A ASIGNATURA

Diseño y desarrollo de las páginas “Nosotros”, “Productos” y “Contactos” de un sitio web de tipo empresarial.

OBJETIVO DE PROYECTO DE APLICACIÓN

La adaptabilidad móvil se refiere a la capacidad de un sitio web para adaptarse y funcionar correctamente en diferentes dispositivos móviles, independientemente de su tamaño de pantalla y orientación. Es importante en el desarrollo web debido al creciente uso de dispositivos móviles para acceder a Internet. Al tener un diseño web adaptable, se mejora la experiencia del usuario y se garantiza que el contenido se muestre de manera óptima en una variedad de dispositivos móviles.

CARACTERISTICAS DEL PROYECTO

Realizar un sitio web que complemente la ventana “Home”, incluya las ventanas “Nosotros”, “Productos” y “Contactos”. El contenido de cada ventana debe estar acorde a la descripción de la página. La ventana “Nosotros” indicará a que se dedica la empresa. La ventana “Productos” contendrá los artículos que ofrece a la venta la institución. La venta “Contactos” contendrá la información de contacto como mapa del sitio, números de contacto, redes sociales, email, etc.

# **BIBLIOGRAFÍA**

Rodrigo González González y Jorge Jimeno Bernal. (2012). *Check list / Listas de chequeo: ¿Qué es un checklist y cómo usarlo?* Obtenido de Pdcahome: https://www.pdcahome.com/check-list/

Auditool. (09 de 11 de 2017). *Técnicas de Auditoría Asistidas por Computador - TAAC*. Obtenido de Comunidad Contable: http://www.comunidadcontable.com/BancoConocimiento/Otros/tecnicas-de-auditoria-asistidas-por-computador-taac.asp?#:~:text=Las%20técnicas%20de%20auditoría%20asistidas,para%20aplicar%20ciertas%20rutinas%20pre-

AVAST. (s.f.). *Qué es el phishing.* Obtenido de Avast: https://www.avast.com/es-es/c-phishing

Becerra, V. (2 de 04 de 2013). *Recursos Humanos.* Obtenido de EmprendePyme: https://www.emprendepyme.net/recursos-humanos

Benget. (s.f.). *Benchmarking.*

Betancourt, D. F. (21 de 05 de 2017). *¿Qué es una salida no conforme?* Obtenido de ingenioempresa: https://ingenioempresa.com/salidas-no-conformes/#:~:text=salida%20no%20conforme-,¿Qué%20es%20una%20salida%20no%20conforme%3F,para%20la%20ejecución%20de%20actividades.