INFORME POSCUENTA OCTUBRE 2024

Girón, 01 noviembre del 2024

Señor(a)
ADRIANA MARTINEZ RIVERA
SUPERVISOR CONTRATO No. **CO1.PCCNTR.5842778**
Centro Industrial de Mantenimiento Integral
Girón

Asunto: Informe poscuenta de ejecución contractual Mes noviembre del año 2024

JOHN FREDDY BECERRA CASTELLANOS, identificado con C.C. 109860945; en calidad de contratista del CIMI, contrato **5842778**/2024, relaciono las actividades desarrolladas **desde el 17 de octubre del 2024 hasta el 31 de octubre** del 2024:

No	Obligaciones	Acciones realizadas	Evidencias
1	Realizar la planeación de las actividades de aprendizaje del proceso formativo basado en el desarrollo curricular del programa de formación	No realizó actividad durante el mes en curso	
2	Participar cuando el centro de formación lo requiera, en jornadas de diseño y desarrollo curricular, proyectos de investigación o transferencias técnicas o pedagógicas, procesos de selección, diseño de talleres e instrumentos de evaluación, procesos de Registro calificado y autoevaluación, labores de montaje y puesta en marcha de equipos y maquinaria, actividades de inducción y reinducción, procesos de investigación, desarrollo tecnológico e innovación nacional e internacional, identificación, estructuración y definición de las especificaciones técnicas de los bienes y servicios, promoción del portafolio de servicios.	No realizó actividad durante el mes en curso	No aplica

3	Ejecutar la formación profesional integral en los diferentes niveles y modalidades (titulada, complementaria y virtual) de acuerdo con el diseño curricular, desarrollo curricular y proyecto formativo de los programas asociados a cada red de conocimiento y área temática del objeto contratado.	Ejecutar Proceso de Formación Titulada correspondiente a las Fichas: 2824003 - 2696521 - 2894667 - 3065369	
4	Evaluar los aprendizajes previos y emitir juicios valorativos sobre el nivel de cumplimiento de los resultados de aprendizaje de las competencias del programa, adquiridos por los aprendices en el desarrollo de su formación, aplicando los procedimientos, plazos y herramientas tecnológicas que la entidad defina.	No se realizó actividad durante el mes en curso	
5	Entregar los soportes del procedimiento de Ingreso de aprendices al programa de formación tales como: Ficha de matrícula; fotocopia del documento de identidad, y/o requisitos definidos en el diseño curricular, cuando se desarrolle formación complementaria.	No realizó actividad durante el mes en curso	No aplica
6	Registrar, verificar y hacer seguimiento oportuno en el sistema de información que la entidad defina para la Gestión de la Formación Profesional Integral mediante las siguientes actividades: a) Verificando en el sistema que la totalidad de los aprendices se encuentren matriculados en el respectivo programa de formación. b) Registrar oportunamente los eventos de formación, proyectos y rutas de aprendizaje requeridas para el cumplimiento de las metas institucionales. c). Registrando juicios evaluativos del	No se realizó actividad durante el mes en curso	No aplica

	reconocimiento de aprendizajes previos en caso de requerirse, los juicios evaluativos, rutas de aprendizaje, para los beneficiarios nuevos, reintegrados o trasladados. d). Comunicando al Coordinador Académico oportunamente anomalías, inconsistencias, novedades de aprendices y hallazgos en el registro de la información.		
7	Aplicar y hacer cumplir el reglamento del aprendiz, así como el manual de convivencia del Centro de Formación	No realizó actividad durante el mes en curso	No aplica
8	Acatar los lineamientos del Sistema Integrado de Gestión y el Sistema de Seguridad de Salud en el Trabajo y asistir a las convocatorias que el centro programa.		No aplica
9	Apoyar el seguimiento de la etapa productiva de los aprendices del centro de formación y sus sedes de acuerdo a su perfil y cumpliendo con el procedimiento de formación profesional integral. (cuando aplique).		No aplica
10	Responder por los bienes y elementos puestos a su disposición para el cumplimiento del objeto del contrato y una vez finalizado, quedar a paz y salvo con el Almacén.		No aplica
11	Asistir y participar activamente en las reuniones programadas por la(s) red(es) de conocimiento propias del Centro de Formación a las cuales sea invitado.		No aplica
12	Utilizar el LMS Institucional como apoyo a la formación en cualquiera de las estrategias institucionales.		No aplica
13	Cumplir con los lineamientos de orientación del desempeño del		No aplica

	instructor en ambientes virtuales de aprendizaje, de acuerdo a lo establecido en el aplicativo compromISO – Guía Orientación Formación ambientes virtuales de aprendizaje (Aplica para instructores que apoyan formación virtual).		
14	Apoyar los procesos de Contratación de Bienes y Servicios en todas sus etapas Preparatoria (requerimiento de la necesidad, cotizaciones, consolidación de información, verificación de precios en el mercado), Precontractual (evaluaciones, respuesta observaciones) y post contractual (apoyo técnico de la ejecución)		No aplica
15	En aplicación de la Ley 1474 de 2011, realizar cuando sea requerido por la entidad el apoyo a la Supervisión de los contratos que celebre el SENA, de acuerdo con el Manual de Supervisión del SENA vigente.		No aplica
16	En aplicación de la Ley 1474 de 2011, realizar cuando sea requerido por la entidad el apoyo a la Supervisión de los contratos que celebre el SENA, de acuerdo con el Manual de Supervisión del SENA vigente.		Se presenta formato GF y GC a la coordinación académica:
17	Las demás necesarias para el cabal cumplimiento del objeto contractual.		No aplica

Cordialmente,



John Freddy Becerra castellanos

Contratista
C.C. No. 1098604945

Recibí a satisfacción:

ADRIANA MARTINEZ RIVERA
SUPERVISOR CONTRATO No. **CO1. PCCNTR.5842778 / 2024**

Evidencia formación

Ficha 2696521

En este periodo se procede a solicitar todos los proyectos, se da un tiempo de 30 minutos por aprendiz para que ellos expongan su proyecto final y seguido a esto se da una apreciación y los puntos a mejorar de ser requerido.

Se logra evaluar todos los aprendices y se da acompañamiento para atender los puntos a mejorar en cada proyecto.

Igualmente se trabaja con algunos puntos solicitados por los aprendices donde ellos manifiestan que tuvieron problemas para unir los temas vistos en formación.

En este momento de la formación los aprendices cuentan con los conocimientos mínimos para desarrollar un proyecto web y versionarlo con le herramienta de control de eversiones **git**, ellos entregan un repositorio donde se encuentra contenido el código fuente separado por los siguientes temas:

- HTML
- CSS
- JavaScript
- GIT

Todo esto en un proyecto que cumpla con los estándares mínimos de calidad para así obtener un proyecto que este lo más cercanos a la realidad que los aprendices afrontaran en su etapa productiva.

Este reporte culmina indicando que todo el tiempo fue trabajo evaluativo con los aprendices, se evalúan individualmente a cada aprendiz.

Ficha 2824003

Se inicia con una evaluación de los conceptos básicos de la sintaxis con el lenguaje de programación **JavaScript**, en esta evaluación se buscaba evaluar la lógica de programación de los aprendieses indicando que es un lenguaje de programación nuevo para ellos y dejando como base que ellos ya cuentan con los conocimientos previos con otro lenguaje de programación como los es **.NET**

Esta evaluación se tomo como base la guía de aprendizaje que ellos resolvieron en la cual se logra sentar como base la sintaxis del lenguaje de programación JavaScript.

Continuemos con el tema asincronía, este es un tema fundamental en la programación web, en este tema se abarcarán los siguientes temas:

- Callback
- Promesas
- Async await
- Fetch

Ficha 2894667

Finalizamos la última practica guía por el instructor donde se realiza un clon a la aplicación de videos youtube, en esta practica se abarcaron todos los temas vistos en formación y se solicito a los aprendices como forma de evaluación de desempeño la normalización del proyecto, con esto se logro evaluar temas como (BEM, GIT, Especificad, CSS, HTML, selectores y buenas prácticas).

Para evaluar el tema se forman grupos de dos personas y se entrega a cada grupo un tema, donde los aprendices pasan a código una propuesta de página web, la propuesta esta en formato **.psd** y ellos solo pueden extraer los recursos gráficos el resto de contenido se debe pasar a etiquetas **html** y se maqueta con el lenguaje de maquetado **CSS.** El trabajo grupal se basa en una problemática que se logra observar en el grupo “Un aprendiz se encuentra aislado de los demás compañeros” se busca que el aprendiz socialice mediante el trabajo.

Para el siguiente informe se estar emitiendo el juicio evaluativo de un resultado de aprendizaje (crea componentes web)

Ficha 2894326

Se realizar la explicación del modulo **css flexbox** con este modulo logramos ubicar en el plano cartesiano cualquier componente web, logramos posicionar los componentes tanto en la **main axis** como en el **cross axis**.
Se atienen las preguntas del grupo porque en este modulo igualmente se requiere tener conocimiento de **CSS**, luego de esto iniciamos con los ejercicios prácticos para poder transferir el conocimiento adquirido de los aprendices.

Ficha 3065369

INFORME POSCUENTA OCTUBRE 2024

Ficha nueva en el grupo de desarrollo, se realiza una charla en la cual se define el perfil de un desarrollador web, se habla de los diferentes perfiles y métodos de trabajo para un desarrollador web.

Se inicia a trabajar con la terminal **CLI** realizado ejercicios de creación, modificación y eliminación de archivos y directorios mediante la terminal, con estos ejercicios los aprendices logran comprender el funcionamiento de una terminal básica y se explica el porque es importante manejar la terminal.

Demos paso a trabajar con la herramienta de control de versiones **GIT** con la cual se solicita creen repositorio para que iniciemos a trabajar con ejemplos de archivos, ramas y confirmaciones.

Continuamos con el primer lenguaje de etiquetado **HTML** se explica la estructura básica de un archivo html y de igual forma la estructura de una etiqueta html, etiqueta de apertura y etiqueta de cierre, se habla de los diferentes atributos que tiene una etiqueta y como modifica el comportamiento de la misma.

Finalizamos el reporte con un ejercicio practico de formularios y los diferentes tipos de imput que tenemos en html para capturar la infomació.

Centro Industrial de Mantenimiento Integral

Versión: 07

FORMATO HORAS APLICADAS A LA FORMACIÓN U OTRAS ACTIVIDADES

NOMBRE DEL INSTRUCTOR: John Freddy Becerra Gallos No. DE IDENTIFICACIÓN: 1098604943

FECHA INICIO REPORTE: 16-octubre-2024 FECHA FINALIZACIÓN DEL REPORTE: 31-Octubre-2024

FECHA	HORA INCIO	HORA FINALIZACION	No HORAS	ACTIVIDAD DE APRENDIZAJE U OTRAS ACTIVIDADES DESARROLLADAS	No FICHA (Cuando aplique)	No CEDULA APRENDIZ Ó PERSONA QUE AUTORIZA LA ACTIVIDAD	NOMBRE DEL APRENDIZ Ó PERSONA QUE AUTORIZA LA ACTIVIDAD	FIRMA APRENDIZ O PERSONA QUE AUTORIZA LA ACTIVIDAD	OBSERVACIONES
16/10/24	6:30	11:30	5	Revision proyecto	2696521	1099739376	Jesus Galvis		
17/10/24	6:30	11:30	5	Revision proyecto	2696521	1096539681	Zunnion Muñoz		
18/10/24	6:30	9:30	3	Taller JavaScript	2824003	1095304196	Valerie Alcantara		
19/10/24	2:00	9:00	7	Evaluacion JavaScript	2824003	1095339128	Jose Luis Perez	Jose Luis P.	
21/10/24	6:30	11:30	5	Trabajo css Clon	3065358	1095305335	Breyner Acosta		2894667
22/10/24	18:30	5:30	5	terminal CLI	3065358	1096066019	Santiago Espitia	Santiago Espitia	
22/10/24	6:30	9:30	3.5	Asincronia JS	2824003	1038665378	Hernán David	Cordero B.	
23/10/24	6:30	11:30	5	Proyectos	2696521	1117620883	Juan Faranda		
24/10/24	6:30	11:30	5	Proyectos	2696521	1095909246	Juan Sebastian T	Juan T	
25/10/24	6:30	11:30	5	Taller grupal final css	2894667	1099739621	Yerinson Suarez F.	Yerinson S.	
25/10/24	6:30	9:30	3	Asincronia JS	2824003	1098831628	Guillermo Garrido		
26/10/24	2:00	9:00	7	Flexbox CSS	2944326	91480x	Jose Villabona		
28/10/24	6:30	11:30	5	Taller grupal final css	2894667	1097049091	Reinel Hernandez	Reinel Hernandez	
29/10/24	12:30	5:30	5	Entrada HTML	3065358	1095306461	Santiago carrillo	Santiago carrillo	
29/10/24	6:30	9:30	3.5	Asincronia JS	2824003	1098668540	Hernán David Cordero		
30/10/24	6:30	11:30	5	Proyecto	2696521	1095909521	Angelica Agudelo	Angelica	
31/10/24	6:30	11:30	5	Proyecto	2696521	1095909521	Angelica Agudelo	Angelica	
01/11/24	6:30	11:30	5	Taller grupal final css	2896507	1096206861	Santiago muñoz	Santiago.m	
		TOTAL HORAS							

Nota: El número de documento, nombres y firmas, se deben registrar por fila, no se admiten firmas o datos cruzados.

FIRMA COORDINADOR ACADEMICO

John Becerra.

FIRMA INSTRUCTOR

Powered by CamScanner

PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

1) IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- **Denominación del Programa de Formación**: Tecnología en Análisis y Desarrollo de Software
- **Código del Programa de Formación:** 2824003
- **Nombre del Proyecto:** Construcción de software integrador de tecnologías orientadas a servicios
- **Fase del Proyecto:** Ejecución
- **Actividad de Proyecto:** Analizar los conocimientos de la programación asíncrona y bajo el lenguaje de programación JavaScript.
- **Competencia:** DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y METODOLOGÍAS DE DESARROLLO
- **Resultados de Aprendizaje Alcanzar:** CREAR COMPONENTES FRONT-END DEL SOFTWARE DE ACUERDO CON LAS NECESIDADES DEL CLIENTE.
- **Duración de la Guía:** 50 Horas

2) PRESENTACIÓN

Aprendiz SENA:

Como cualquier otro lenguaje de programación, JavaScript tiene algunas características especiales: sintaxis, modelo de objetos, etc. Claramente, cualquier cosa que diferencia un lenguaje de otro. Además, descubrirás rápidamente que JavaScript es un lenguaje relativamente especial en su acercamiento a las cosas. Esta parte es esencial para cualquier principiante de programación e incluso para aquellos que ya conocen un lenguaje de programación debido a que las diferencias con otros lenguajes de programación son numerosas.

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Claramente JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

3) FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

A. Actividades de reflexión inicial.

- Desarrollo de la actividad tipo investigación en la cual el aprendiz debe consultar en internet los siguientes interrogantes y dar un ejemplo de cada uno:
- Programación asíncrona:
    - Que son los temporizadores setimeout y setinterval **(de un ejemplo)**
    - Que es la asincronía y el evento loop **(de un ejemplo)**
    - Que son los callbacks **(de un ejemplo)**
    - Que son las promesas **(de un ejemplo)**
    - Que son las funciones asíncronas (async / await) **(de un ejemplo)**

Duración de la actividad 1 hora.

B. Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.

Discusión en grupo.

- Se discutirá con el grupo el significado de los tipos de datos, estructuras de control objetos y funciones de lenguaje JavaScript.
- Se realizarán preguntas para definir el conocimiento previo y se despejarán las dudas que tengas respecto al diseño web.
- Se realizará una profundización de la investigación con diferentes ejemplos relacionados en la búsqueda y ejemplos propuestos en el momento de la socialización.

Duración de la actividad 4 hora.

C. Actividades de apropiación del conocimiento.

En esta sección trabajaremos una actividad

En este espacio de trabajo se busca identificar e iniciar el aprendizaje en los elementos fundamentales que componen el pensamiento lógico de la programación y su estructura, permitiendo plantear conceptos y luego incorporarlos en ejercicios prácticos, realizando los siguientes algoritmos:

Antes de iniciar a codificar una solución a los enunciados:

a) Cree un repositorio local
b) Cree un archivo index.html
c) Cree una lista de navegación con los ejercicios a solucionar (esta lista debe tener un diseño agradable, aplicar estilos css)
d) Agregue los cambios al staging área
e) Confirme los cambios, el nombre de la confirmación puede ser “Inicio del taller”
f) Cree una rama con el nombre desarrollo
g) Ubíquese en la nueva rama
h) Cree el primer archivo ejercicio1.html y su archivo ejercicio1.js (todos los ejercicios deben estar separados por archivos y se debe poder navegar por los diferentes vínculos del html)
i) Agregue los cambios al staging área
j) Confirme los cambios, el nombre de la confirmación puede ser “primer ejercicio”
k) Trabaje sobre la rama desarrollo todos los ejercicios
l) Al finalizar los ejercicios tome una captura de la terminal con el log de confirmaciones
m) Ubíquese en la rama main
n) Realice el merge de la rama desarrollo
o) Tome una captura de la terminal donde se logre evidenciar que elimino la rama desarrollo
p) Cree un repositorio remoto con el nombre introducción a javascript
q) Agregue el origen a su repositorio local
r) Realice un push a la rama main
s) Cree una página con la solución de los ejercicios
t) Descargue el repositorio y súbalo a territorium o en su defecto lo envía por correo al instructor (territorium podría estar fuera de servicio)
u) Envie el link para su validación

1. Escriba una función imprimirNumeros(desde, hasta) que genere un número cada segundo, comenzando desde desde y terminando con hasta.

   Haz dos variantes de la solución describiendo el paso a paso de cada una de ellas:

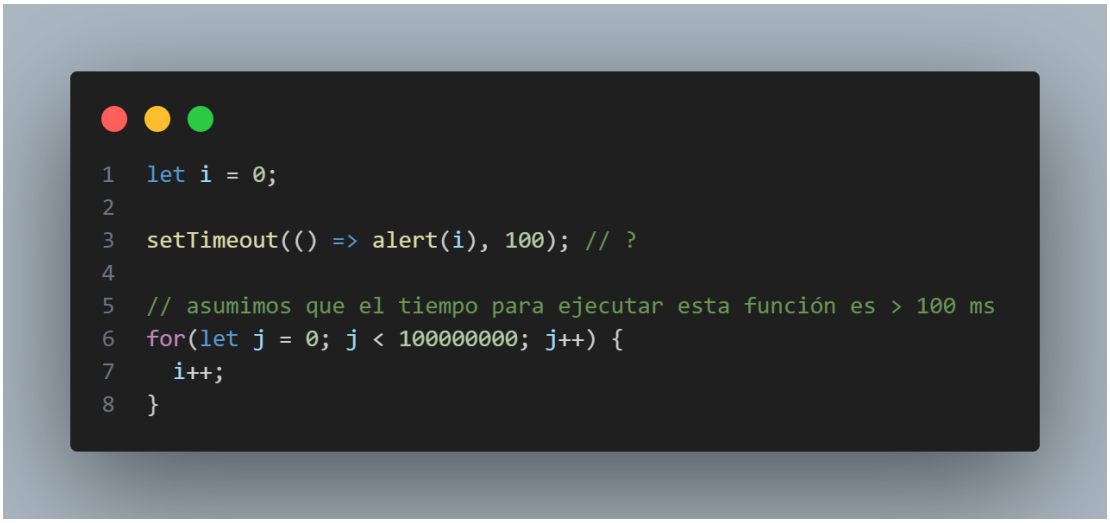
   - Usando setInterval.
   - Usando setTimeout anidado.

2. En el siguiente código hay una llamada programada setTimeout, luego se ejecuta un cálculo pesado que demora más de 100 ms en finalizar.

   ¿Cuándo se ejecutará la función programada y por qué se ejecutará?

   a) Después del bucle.
   b) Antes del bucle.
   c) Al comienzo del bucle.

   ¿Qué va a mostrar alert()?



```
let i = 0;

setTimeout(() => alert(i), 100); // ?

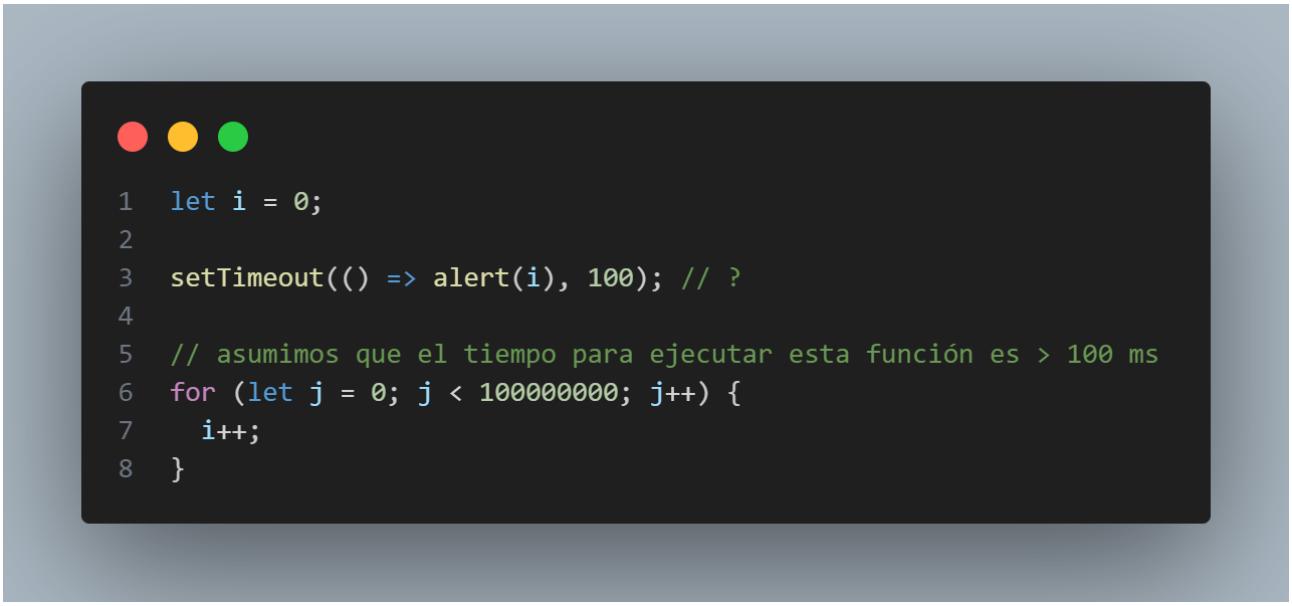
// asumimos que el tiempo para ejecutar esta función es > 100 ms
for(let j = 0; j < 100000000; j++) {
  i++;
}
```

3. Crear una función **map** que acepte un array y un **callback** y que:

   - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
   - obtenga el resultado
   - lo pushee a un nuevo array
   - devuelva el array final con el resultado de cada una de las llamadas al **callback**.

4. Crear una función **filter** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - sí dicho **callback** devuelve true, pushea el elemento a un nuevo array
    - devuelva el array final con los elementos que pasaron el "filtro".
5. Crear una función **every** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva true si todas las llamadas al **callback** devolvieron true
6. Crear una función **some** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva true si al menos una de las llamadas al **callback** devolvió true
7. Crear una función **find** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva el elemento pasado como argumento del primer **callback** que devuelva true
    - sí ningún **callback** devuelve true, devuelva undefined
8. Crear una función **findIndex** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva el índice del elemento pasado como argumento del primer **callback** que devuelva true
    - sí ningún **callback** devuelve true, devuelva undefined
9. Crear una función **dropWhile** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva un array con los elementos a partir del primer **callback** que devolvió false
    - (Es decir, crea un nuevo array y va recorriendo elemento por elemento, mientras el **callback** de true, no los agrega, cuando el **callback** da false por primera vez agrega todos los elementos restantes a partir de dicho elemento inclusive)

10. Crear una función **takeWhile** que acepte un array y un **callback** y que:
    - por cada elemento del array ejecute el **callback** pasándole dicho elemento como argumento
    - devuelva un array con los elementos hasta el primer **callback** que devolvió false
    - (Inverso del **dropWhile**)
11. ¿Cuál es el resultado del código a continuación?



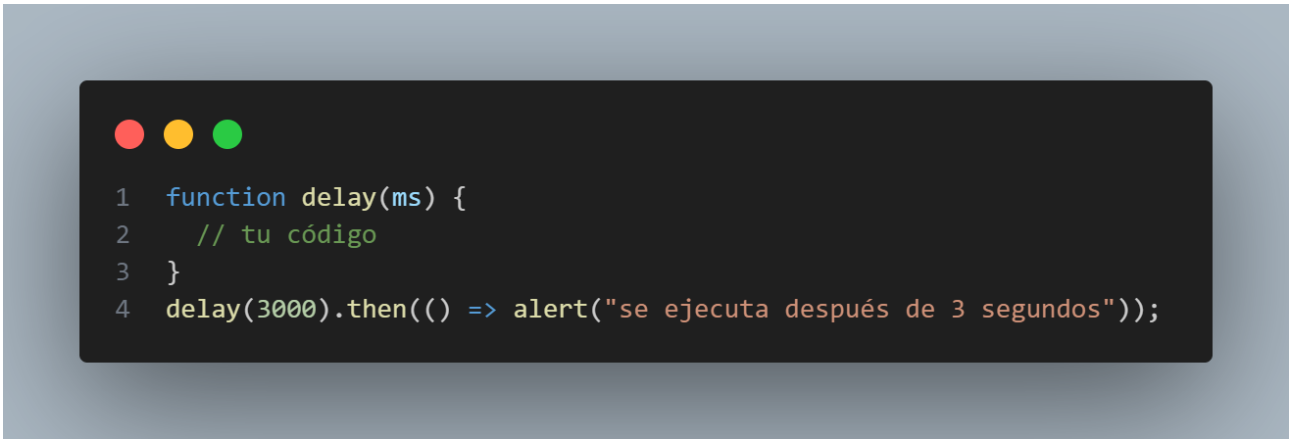
```
let i = 0;

setTimeout(() => alert(i), 100); // ?

// asumimos que el tiempo para ejecutar esta función es > 100 ms
for (let j = 0; j < 100000000; j++) {
  i++;
}
```

12. La función incorporada setTimeout utiliza callbacks. Crea una alternativa basada en promesas.

    La función delay(ms) debería devolver una promesa. Esa promesa debería resolverse después de ms milisegundos, para que podamos agregarle. then, así:



```
function delay(ms) {
  // tu código
}
delay(3000).then(() => alert("se ejecuta después de 3 segundos"));
```

13. Crea una **Promise** que se resuelva después de 3 segundos y luego imprima "Promise resuelta" cuando se cumpla.
14. Define una función asincrónica que espere 1 segundo y luego devuelva una cadena que diga "Operación completada". Utiliza async/await.
15. Crea una Promise que se rechace después de 2 segundos y captura el error para imprimir "Error: Promise rechazada".
16. Crea tres Promises consecutivas, donde cada una se resuelva después de 1 segundo y devuelva un número diferente. Luego, encadena las tres Promises para sumar los resultados y mostrar el resultado final.
17. Realiza una llamada a un archivo local Muestra en formato json, luego muestre los datos obtenidos en la consola.
18. Después de realizar una llamada a un archivo local en formato json, utiliza el método then() para filtrar los datos y mostrar solo los elementos que cumplan ciertos criterios (por ejemplo, mostrar solo los nombres que comiencen con "A").
19. Crea una función asincrónica que realice una llamada a un archivo local en formato json y luego manipule los datos recibidos para mostrar información específica.
20. Crea tres Promises que se resuelvan después de diferentes intervalos de tiempo y luego utilice Promise.all() para mostrar un mensaje cuando todas se hayan resuelto.
21. Realiza tres Promises, algunas de las cuales se resuelvan y otras se rechacen. Utiliza Promise.allSettled() para obtener información sobre el estado de todas las Promises.
22. Crea un bucle que realice llamadas asincrónicas utilizando async/await para procesar una lista de elementos uno por uno.

23. Realiza una llamada a un archivo local en formato json y maneja posibles errores utilizando try/catch para mostrar un mensaje de error en caso de fallo.
24. Crear un objeto proxy usando la clase Proxy
25. Usar un proxy para la validación de los valores de propiedades
    - Cuando declaremos que el atributo es numérico solo perimimos que ingresen números
    - Cuando declaremos que el atributo es alfanumérico solo permitimos que ingresen letras
    - Cuando declaremos que el atributo es un email, solo permitimos que ingresen correos
    - Cuando declaremos que el atributo es de tipo date, solo permitimos que ingresen fechas
    - Para todos los atributos no se puede ingresar espacios en blanco tanto al inicio como al final de estos y se debe mostrar el error personalizado por consola

Toda la actividad se realizará con el acompañamiento del instructor, podrá realizar las preguntas requeridas al instructor para solucionar la actividad.

Duración de la actividad 8 horas.

D. Actividades de transferencia del conocimiento.

Esta actividad se validará como una actividad de evaluación del tema, para presentar esta actividad el aprendiz debe presentar la investigación junto con la solución del taller práctico después se entregarán 5 ejercicios para su solución de forma individual.

Esta actividad es individual donde se busca que el aprendiz sin apoyo del instructor mida su avance y apropiación del tema.

Duración de la actividad 13 horas.

4) ACTIVIDADES DE EVALUACIÓN

Tome como referencia la técnica e instrumentos de evaluación citados en la guía de Desarrollo Curricular

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento: **Evidencias de Desempeño:** • Evaluación(es). • Taller(es). **Evidencias de Producto:** • Código fuente • Repositorio remoto y local. • Link de publicación	Identifica y describe, en un sistema de información dado, los datos de entrada, procesamiento de los datos e información generada, según necesidades del cliente.	**Técnica:** Observación directa

5) GLOSARIO DE TÉRMINOS

- **JavaScript:** es un lenguaje de programación de alto nivel, interpretado y multiplataforma. Es uno de los tres pilares fundamentales de la web, junto con HTML (Hypertext Markup Language) y CSS (Cascading Style Sheets).
- **GIT:** sistema de control de versiones distribuido ampliamente utilizado para el seguimiento de cambios en el código fuente durante el desarrollo de software.
- **Compilación:** proceso de traducir el código fuente de un programa informático a un formato ejecutable que la computadora puede entender y ejecutar.
- **Repositorio:** Un repositorio en el contexto de la informática, especialmente en el desarrollo de software, se refiere a un lugar centralizado donde se almacenan y gestionan archivos y recursos relacionados con un proyecto específico.
- **Staging área:** El área de staging, también conocida como "área de preparación" o "índice", es una etapa intermedia en el proceso de trabajar con Git. Cuando realizas cambios en tus archivos de código fuente y estás listo para confirmar esos cambios en Git, primero los

"añades" al área de staging antes de "confirmar" definitivamente esos cambios en el historial de Git.

- **Commit:** En Git, un "commit" es una operación que registra los cambios realizados en los archivos de un repositorio en un punto específico de tiempo. Cada commit tiene un mensaje descriptivo que proporciona información sobre los cambios realizados en ese commit en particular. Los commits son la unidad básica de trabajo en Git y forman parte del historial de versiones del repositorio.
- **Branch:** En Git, una "branch" (rama) es una línea independiente de desarrollo que permite a los desarrolladores trabajar en características específicas del proyecto sin afectar el flujo de trabajo principal o "master". Cada rama en un repositorio de Git representa una versión separada del código, y los cambios realizados en una rama no afectan a otras ramas hasta que se fusionan.
- **Merge:** En Git, el comando merge se utiliza para combinar los cambios de una rama (branch) en otra. Esta operación se realiza típicamente cuando se quiere incorporar el trabajo realizado en una rama secundaria de vuelta a la rama principal (por lo general, master o main).
- **Deploy:** El término "deploy" (despliegue) en el contexto del desarrollo de software se refiere al proceso de implementar y poner en funcionamiento una aplicación o una actualización de software en un entorno de producción, donde los usuarios finales pueden interactuar con ella. Este proceso implica tomar el código fuente de la aplicación, compilarlo (si es necesario), configurar cualquier infraestructura requerida, y finalmente lanzar la aplicación para su uso público.

6) REFERENTES BILBIOGRÁFICOS

https://developer.mozilla.org/es/docs/Web/JavaScript

7) CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha

Autor (es)	**John Freddy Becerra Castellanos**	**Instructor**	**CIMI**	**24 – octubre - 2024**

8) CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	**Cargo**	**Dependencia**	**Fecha**	**Razón del Cambio**
Autor (es)					