

Escuela de Código para PILARES

Descripción de actividades

Parte 2: Módulo Elaboración de sistemas web (MW)



Escuela de Código para PILARES Descripción de actividades - Parte 2: Módulo Elaboración de sistemas web (MW) por Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México se distribuye bajo una [Licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-sa/4.0/).

Créditos¹

Redacción de actividades

Alejandra Sarahí Monroy Velázquez, Andrea García Ruiz, Carla Irena Blenda Palacios, Elisa Mariana Valdés Armada, Héctor Alfonso Islas García, Karen Alexa Alva Aguirre, Karina Flores García y Luz Elena Rueda Rojas

Coordinación de módulo

Zian Fanti Gutiérrez e Ivan Vladimir Meza Ruiz

Coordinación de la Transversalización de la Perspectiva de Género

Yuliana Ivette López Rodríguez

Revisiones

Jorge Daniel Guerrero, Citlalli Sánchez Mendoza, Carmen Daniela Garrido Juvencio y Karen Itzel Bruno Sainos

Supervisión PILARES

Jesús Felipe Alanis Manriquez, René Alejandro Rivas Robles y María del Rocío Estrada Monroy

Supervisión IIMAS

Adrián Durán Chavesti, Héctor Benítez Pérez, Helena Gómez Adorno, Ivan Vladimir Meza Ruiz, Nora Isabel Pérez Quesadas, María del Pilar Ángeles, Víctor Manuel Lomas Barrie y Zian Fanti Gutiérrez

Financiamiento:

Diseño de un programa de estudios para la capacitación en programación y habilidades en tecnologías de información y comunicación para la escuela de código dentro de PILARES de la Ciudad de México (SECTEI/284/2019)

¹ En orden alfabético.

Agradecimientos

Agradecemos el tiempo y la retroalimentación hecha a los materiales a:

- Ante Salcedo González, ITAM - Instituto Tecnológico Autónomo de México
- Blanca Esther Carvajal-Gómez, ESCOM - Escuela Superior de Cómputo - IPN
- Dagoberto Pulido Arias, IPN - Instituto Politécnico Nacional
- Eréndira Itzel García Islas, UNAM - Facultad de Ciencias
- Marco Antonio Moreno Ibarra, CIC - Centro de Investigación en Computación del IPN
- Ricardo Marcelín Jiménez, UAM-I Universidad Autónoma Metropolitana – Iztapalapa
- Salvador Elías Venegas Andraca, ITESM - Instituto Tecnológico y de Estudios Superiores de Monterrey

También agradecemos el apoyo y seguimiento al personal de SECTEI, en particular de:

- José Bernardo Rosas Fernandez
- Federico Antonio Hernández Loranca
- Rogelio Artemio Morales Martínez
- Adrián Eleazar Contreras Martínez
- Benigno Antonio González Núñez

Índice

Créditos	2
Agradecimientos	3
Índice	4
Módulo: Elaboración de sistemas web	6
Taller 1: Construyendo mi primer sitio web	7
Actividad 1: ¿Qué es una página web?	8
Actividad 2: Conociendo desde adentro una página web	14
Actividad 3: Agregando nuevos elementos a tu página web	18
Actividad 4: “El Konmarie” del HTML	21
Actividad 5: Ponle estilo a tu página (parte 1)	25
Actividad 6: Ponle tu estilo a tu página (parte 2)	30
Actividad 7: Desde cero	34
Actividad 8: Web de verdad	37
Actividad 9: De mí para el mundo	40
Taller 2: Creando un punto de venta en internet	44
Actividad 0: Git	45
Actividad 1: JavaScript con tenedor	51
Actividad 2: JavaScript con cuchara y tenedor.	60
Actividad 3: JavaScript con cuchara, tenedor y cuchillo.	68
Actividad 4: Ingresando datos específicos	71
Actividad 5: Almacenando datos	74
Actividad 6: Creando y operando datos específicos	80
Actividad 7: Funciones	83

Actividad 8: Modificando mi página	86
Actividad 9: Toda acción tiene una reacción	91
Actividad 10: Instalación y configuración de servicios para sistemas web	95
Actividad 11: ¿A dónde van los datos?	98
Actividad 12: Uppssss me equivoqué!!!! (Modificando y borrando información)	102
Actividad 13: Altas, bajas y cambios.	105
Actividad 14: Y ¿dónde quedó mi información?	107

Taller 3: Dos versiones de una tienda	113
Actividad 1: Plataforma Sails	114
Actividad 2: Anatomía de una aplicación en Sails	118
Actividad 3: Conectando con la base de datos.	121
Actividad 4: Agregando productos.	125
Actividad 5: Rutas hacia los productos	128
Actividad 6: Agregar Lógica	132
Actividad 7: Dándole poder al administrador	134
Actividad 8: Mejorando la webapp	137
Actividad 9: Plataforma PHP Laravel	140
Actividad 10: Anatomía de una aplicación en PHP	144
Actividad 11: Conectando con la base de datos.	148
Actividad 12: Creando un API	152
Actividad 13: Rutas hacia los productos	155
Actividad 14: Dándole seguridad	158
Actividad 15: Ética de los datos	161

Módulo: Elaboración de sistemas web

En honor a **Radia Joy Perlman** (1951, EUA) doctora en Ciencias de la Computación. Desarrolló el Protocolo *Spanning Tree* (STP por sus siglas en inglés) que es un algoritmo que permite la conexión automática de los enlaces de aparatos, por ello, se le conoce como la creadora del internet. Actualmente es una de las personas más influyentes en el ámbito de las Tecnologías de la Información y sigue contribuyendo en el ámbito.

Taller 1: Construyendo mi primer sitio web

En honor de la estadounidense **Jude Milhon** (1939-2003) escritora, programadora, *hacker* y creadora del término así como de la asociación *Cyberpunk* encargada de la defensa de la privacidad digital.

Pugnó por una red pública cuando al internet sólo tenían acceso los militares de Estados Unidos y algunos institutos de investigación. Esto tuvo como consecuencia la creación del Proyecto de la Comunidad de Memoria que fue el primer sistema público de red en 1973.

Su activismo lo extendió a la defensa de los derechos civiles como el voto para las mujeres y en 1994 escribió el texto *Hacking the Wetware: The Nerd Girls Pillow-book* (Hackeando a las personas: El libro de cabecera de las 'nerd') donde invita a otras mujeres a formar parte de la tecnología, la programación y la web.

En este documento da cuenta de la hostilidad al género femenino en estas disciplinas y ofrece claves para que las nuevas integrantes aprendan a lidiar con la hostilidad producto de la cultura machista. Estaba convencida de que la tecnología es una herramienta para la resolución de problemas, por ello, una de sus frases célebres fue “las mujeres necesitan *módems*”.

Competencia del taller: Programar y diseñar un sitio web estático que pueda ser visto desde esta y que siga los requerimientos levantados con un “cliente”.

Actitudes: Curiosidad, disposición, constancia, persistencia, apertura a la incorporación de nuevos aprendizajes, capacidad de análisis, apertura al diálogo, escucha, trabajo en equipo, intercambio de opiniones, participación activa, interés y apertura a incorporar en las actividades la Perspectiva de Género para el logro de un bien común.

Actividad 1: ¿Qué es una página web?

Aprendizaje esperado: Identificar los diferentes tipos de componentes básicos que conforman un página web tradicional.		Duración de la actividad: 3 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Programa WordPerfect Presentations. 2. <i>MWT1A1_Anexos.pdf</i> 3. <i>MWT1A1_PlantillaMockup</i>	Evidencia: 1. Una carpeta general de su página web, que contenga: un maquetado o mockup de la página (con el nombre “MWT1A1_MiMockup”). 2. Archivo de texto “MWT1A1_DescripcionMockup.txt” con una descripción detallada de su página web, sus elementos y su estructuración.	Retroalimentación: 1. El mockup deberá contar con todos los componentes de página web que en esta actividad se abordaron: título, subtítulos, menú, párrafos, listas, ligas, ligas a correo, imágenes, espaciadores, entre otros. 2. La descripción de la página web evidenciará que la participante comprendió la funcionalidad de cada componente, así como la lógica de su estructuración.
Desarrollo de la actividad:		
<p>Primera parte: QQQ (Qué veo)¹</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>Se visitarán páginas web con el fin de identificar partes comunes.</p> <ol style="list-style-type: none"> Se facilitará a la participante algunos ejemplos (cinco aproximadamente) de página web, de preferencia con todos los componentes -título, subtítulos, menú, párrafos, listas, ligas, ligas a correo, imágenes, espaciadores, y otros- que se puedan identificar sencillamente. Se sugiere que del listado anexo al final de este documento sea seleccionada por lo menos una página de cada categoría. Con cada página seleccionada, la participante deberá exponer detalladamente: qué elementos observa dentro de cada página web, en qué lugar o posición se encuentra cada elemento dentro de la página, qué diferencias identifica con respecto a las otras páginas web consultadas y qué semejanzas. Sin ayuda, deberá explicar con sus propios conocimientos y vocabulario los puntos anteriores, haciendo uso del siguiente formato: 		

Nombre de la página web:			
Qué elementos veo	En qué lugar se encuentran	Qué diferencias observo	Qué semejanzas observo

- Al finalizar el punto anterior, la participante compartirá lo que redactó en cada columna, y en conjunto se llevará a cabo un ejercicio de reflexión, donde respondan preguntas como:
 - ¿Qué elementos se repiten más en todas las páginas web?
 - ¿Qué elementos se repiten menos?
 - ¿Qué elementos aparecen constantemente en la parte superior, media e inferior de las páginas?
 - ¿Cómo se llama cada elemento que identificaste?
- Para concluir se concretarán los conocimientos aquí tratados, se compartirá el nombre formal de cada componente de las páginas web y su función o utilidad. Si se prefiere, se puede presentar la infografía adjunta *PartesDePáginaWeb* correspondiente a la *Plantilla 1* de los *MWT1A1_Anexos.pdf*. En caso de optar por un material diferente, asegurarse que la estructura y que cada componente sea identificable y con alta calidad visual.

Segunda parte: Mi primera página web.

Sugerencia de tiempo invertido: 2 horas.

- Las participantes elaborarán un prototipo, boceto, maqueta o *mockup* de página web en WordPerfect Presentations (se recomienda trabajar sobre la *Plantilla 2* de los *MWT1A1_Anexos.pdf*). Este mockup será el primer paso en el proceso de diseño y creación de la página web que desarrollarán las participantes en las actividades posteriores de este taller. La temática del proyecto será decisión de la participante, no obstante se presenta la siguiente lista de sugerencias:
 - Ley General de Acceso de las Mujeres a una Vida Libre de Violencia
 - Derechos sexuales y reproductivos
 - Seguridad digital para mujeres

Este mockup (con el nombre “MWT1A1_MiMockup”) deberá presentarse en una carpeta nueva creada a través de la terminal (línea de comandos), con la nomenclatura “MWT1A1_NombreDelProyecto” (esta última parte de la

nomenclatura corresponde al nombre que le asigne a su página web), misma que deberá guardarse en la carpeta de evidencias de este módulo en la nube.

2. Una vez finalizado el diseño de su mockup, la participante creará un archivo de texto con nombre "MWT1A1_DescripcionMockup" (en la carpeta general de su proyecto) donde desarrolle a profundidad:

- ¿Con qué elementos cuenta mi página web?
- ¿Cuál es la utilidad de cada elemento -título, subtítulos, menú, párrafos, listas, ligas, ligas a correo, imágenes, espaciadores y otros?
- ¿En qué lugar de la página se encuentra cada elemento?
- ¿De qué trata mi página?
- ¿El objetivo de mi página responde de manera afirmativa a una de las "Preguntas para que mi proyecto tenga Perspectiva de Género" de la "Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios"?
 - Las líneas generales son:
 - ¿Mi propuesta tiene la intención de mejorar las condiciones de vida de las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada en retomar alguna problemática que afecta de manera directa a las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta toma como eje la vida de mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada a favorecer alguna situación donde mujeres y niñas de algún colectivo se encuentren en desventaja?
 - ¿Mi aporte busca contribuir en la transformación de modelos sociales y culturales que estereotipan el hacer y quehacer de las mujeres o niñas?
 - ¿Mi propuesta busca prevenir o dar atención a una problemática de violencia contra mujeres o niñas?
 - ¿Mi propuesta propone visibilizar la participación de las mujeres en un campo donde son poco visibles?
 - ¿Mi propuesta propone informar sobre derechos humanos de las mujeres?
- ¿Cómo se llama mi página?
- ¿Por qué elegí esa temática?

Esta descripción será lo más detallada posible. Una vez finalizada, la compartirá con la intención de que pueda llevarse a cabo un ejercicio de retroalimentación. El documento deberá guardarse en la carpeta de evidencias de este módulo en la

nube.

Notas para apoyar la actividad:

- En el caso que la participante elija otro contenido, será indispensable supervisar que la temática responda de manera afirmativa por lo menos a una de las preguntas que se enuncian en el apartado “Preguntas para que mi proyecto tenga Perspectiva de Género” de la *Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios* que marca los límites generales de este proyecto. Estas son las siguientes:
 - ¿Mi propuesta tiene la intención de mejorar las condiciones de vida de las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada en retomar alguna problemática que afecta de manera directa a las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta toma como eje la vida de mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada a favorecer alguna situación donde mujeres y niñas de algún colectivo se encuentren en desventaja?
 - ¿Mi aporte busca contribuir en la transformación de modelos sociales y culturales que estereotipan el hacer y quehacer de las mujeres o niñas?
 - ¿Mi propuesta busca prevenir o dar atención a una problemática de violencia contra mujeres o niñas?
 - ¿Mi propuesta propone visibilizar la participación de las mujeres en un campo donde son poco visibles?
 - ¿Mi propuesta propone informar sobre derechos humanos de las mujeres?

Links de apoyo:

Páginas sugeridas para la primera parte de la actividad:

Mujeres, ciencia y tecnología:

- Muñoz, Lourdes. (2010). *Internet un espacio de empoderamiento de las mujeres. Mujeres en red*.
<http://www.mujeresenred.net/spip.php?article1837>. (Febrero 2020).
- Laboratoria.
<https://www.laboratoria.la/> (Febrero 2020).
- Instituto Mexicano de la Juventud. (2017). *“Código X”, iniciativa tecnológica por las niñas y mujeres jóvenes*.
<https://www.gob.mx/imjuve/articulos/codigo-x-iniciativa-tecnologica-por-las-ninas-y-mujeres-jovenes?idiom=es>
(Febrero 2020).
- Coursera.
<https://es.coursera.org/> (Febrero 2020).

Derechos y salud sexual:

- Mexfam.
<https://mexfam.org.mx> (Febrero 2020).
- DKT de México.
<https://dkt.com.mx/> (Febrero 2020).
- AHF México.
<https://pruebadevih.com.mx/> (Febrero 2020).
- Amnistía Internacional. (Sin fecha). *Derechos sexuales y reproductivos*.
<https://www.es.amnesty.org/en-que-estamos/temas/derechos-sexuales-reproductivos/> (Febrero 2020).
- Planned Parenthood. (Sin fecha). *Métodos Anticonceptivos*.
<https://www.plannedparenthood.org/es/temas-de-salud/anticonceptivos> (Febrero 2020).

Derechos humanos de la mujer:

- INMUJER INSTITUTO NACIONAL DE LAS MUJERES. (Sin fecha). *Puntogénero: formación para la igualdad*.
<http://puntogenero.inmujeres.gob.mx/> (Febrero 2020).
- Instituto Nacional de las Mujeres. (2016). *Los Derechos Humanos de las Mujeres*.
<https://www.gob.mx/inmujeres/articulos/los-derechos-humanos-de-las-mujeres?idiom=es> (Febrero 2020).

Feminismo:

- Archivos Históricos del Feminismo. (Sin fecha). *La Correa Feminista*.
https://archivos-feministas.cieg.unam.mx/publicaciones/la_correa_feminista.html (Febrero 2020).
- Oxfam México. (Sin fecha). *Órbita Feminista 2018*.
<https://www.oxfamMexico.org/orbita-feminista> (Febrero 2020).
- edX. (Sin fecha). *Cursos de feminismo y derechos de las mujeres*.
<https://www.edx.org/es/aprende/feminismo-derecho-mujeres> (Febrero 2020).

Violencia de género contra las mujeres y niñas:

- ONU MUJERES México. (2018). *Adhesiones a la Campaña HeForShe en México 2014-2018*.
<https://mexico.unwomen.org/es/noticias-y-eventos/articulos/2016/08/adhesiones-heforshe-mexico> (Febrero 2020).
- Gobierno de la Ciudad de México. (2019). *¿Cómo denunciar actos de violencia contra las mujeres?*.

<https://www.cdmx.gob.mx/portal/articulo/distintos-tipos-de-violencia/> (Febrero 2020).

Referencias:

- Pimienta, J. (2012). *Estrategias de enseñanza-aprendizaje. Docencia universitaria basada en competencias*. México. Pearson. 195p.

Anexos:

El archivo "MWT1A1_Anexos.pdf" contiene:

Plantilla 1: Infografía partes de una web.

Plantilla 2: Mockup.

Actividad 2: Conociendo desde adentro una página web

Aprendizaje esperado: Crear una página web estática de forma local, usando componentes básicos		Duración de la actividad: 3 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Editor de textos especializado en web: Visual Studio Code. (Identación automática, uso de colores para etiquetas, atributos y propiedades HTML, detección de palabras clave, detección de errores, etc. deseable depuración javascript). 2. Infografía <i>MWT1A2_InfografiaHTML.</i> 3. Hojas de papel tamaño carta o fichas bibliográficas. 4. Lápices, plumas o colores. 5. Tijeras. 	<p>Evidencia:</p> <ol style="list-style-type: none"> 1. Fichas de memorama sobre cada etiqueta (body, head, títulos, encabezados, párrafos, imágenes, !DOCTYPE, html y meta) con sus respectivos ejemplos. 2. Archivo HTML escrito en Visual Studio Code nombrado <i>index.html</i> estructurado con las etiquetas básicas trabajadas en esta actividad. 3. Carpetas <i>img</i> y <i>HTML</i> dentro de la carpeta <i>MWT1A1_NombreDelProyecto</i> creada previamente en la actividad 1. Dentro de “HTML” el archivo “<i>Index.html</i>”, y las imágenes dentro de la carpeta “<i>img</i>”. Las carpetas serán creadas desde la consola. 4. Uso de Visual Studio Code como herramienta para escribir código HTML. 	<p>Retroalimentación:</p> <ol style="list-style-type: none"> 1. El archivo “<i>index.html</i>” deberá desplegarse una página web estática de forma local en un navegador, con la estructura básica de etiquetas trabajadas en esta actividad (head, body, títulos, encabezados, párrafos e imágenes). 2. Las fichas de memorama deberán contener cada uno de los aspectos sugeridos en la primera parte, punto dos, y siguiendo el formato que se requiere: Ficha 1: Nombre, función, atributos; Ficha 2: <etiqueta de apertura> CONTENIDO </etiqueta de cierre> 3. Se sugiere una jerarquía de guardado de carpetas. (“<i>MWT1A1_NombreDelProyecto</i>”> HTML, img), con el contenido correspondiente

		en cada una (HTML> <i>index.html</i> ; img> imágenes del proyecto), sin embargo, no es restrictiva, se puede dejar a criterio propio de la tallerista.
Desarrollo de la actividad:		
<p>Primera parte: Descubro los secretos de las páginas web. Sugerencia de tiempo invertido: 1 hora 30 minutos. Se sugiere iniciar con la infografía <i>InfografíaHTML</i> que corresponde a la plantilla uno de <i>MWT1A2_anexos</i> para introducir aspectos básicos del HTML. En caso de que se decida emplear otro recurso, se deberá verificar que contenga la siguiente información:</p> <ul style="list-style-type: none"> - Qué es HTML, la estructura básica de un archivo HTML y de los elementos de HTML, así como un código HTML básico, con el objetivo de dar una breve introducción al lenguaje. <ol style="list-style-type: none"> 1. Cada participante debe realizar una búsqueda de las etiquetas básicas de HTML: body, head, títulos, encabezados (<i>h1, h2, h3, h4, h5 y h6</i>), párrafos, imágenes, !DOCTYPE, html y meta. Considerando los siguientes aspectos: <ul style="list-style-type: none"> ● Nombre de la etiqueta. ● Función. ● Atributos principales. 2. Dividir una hoja de papel en ocho partes y recortar los rectángulos (también se pueden emplear fichas bibliográficas), éstas serán utilizadas para escribir en un rectángulo la etiqueta con su ejemplo y en otro rectángulo la función de la etiqueta; con la finalidad de elaborar un memorama para trabajarlo en la actividad 4 y, a su vez, que sirvan como material de consulta. Las fichas se pueden decorar de manera libre al gusto de la participante siempre y cuando no afecte el objetivo del memorama. Ejemplo, Ficha 1: <i><title> Mi página web </title></i> Ficha 2: Esta etiqueta muestra el título de la página web en la pestaña del navegador. <p>Segunda parte: Arrancando mi primer proyecto. Sugerencia de tiempo invertido: 1 hora 30 minutos. Basándose en el maquetado del proyecto, las participantes comenzarán a escribir el código HTML en <i>Visual Studio Code</i> para construir su página web y sus contenidos, desde estructura, e imágenes y texto. Es importante recalcar que el código que generen se</p>		

seguirá trabajando en las siguientes actividades.

Se supervisará que la temática que la participante elija para hacer su página web responda en sus objetivos de contenido de manera afirmativa de por lo menos una de las preguntas que se enuncian en el apartado “Preguntas para que mi proyecto tenga Perspectiva de Género” del *Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios* que marca los límites generales de este proyecto.

1. Dentro de la carpeta *MWT1A1_NombreDelProyecto* crear dos carpetas, una debe ir nombrada “HTML” y otra “img”.
2. En Visual Studio Code, crear un archivo nombrado “index.html” y guardarlo dentro de la carpeta *HTML*; las imágenes que se utilicen para la página web deberán ser guardadas en la carpeta *img*.
3. La participante aplicará los conceptos vistos en esta actividad en el maquetado de la página que realizó en la Actividad 1, comenzando a escribir el código HTML, apoyándose del código de ejemplo que aparece en la infografía sugerida.
4. Codificar la página web siguiendo la estructura HTML revisada.

Notas para apoyar la actividad:

- Se sugiere que las participantes utilicen la jerarquía de carpetas para crear una página web (HTML, img, etcétera), sin embargo queda a criterio de la tallerista especificarla.
- Las participantes deberán escribir su código y guardarlo de manera constante para visualizarlo en el navegador y así observar sus modificaciones o avances.
- Apoyar a las participantes a despejar sus dudas respecto al uso del editor de textos, así como en la aplicación de los conocimientos adquiridos al desarrollo de su página web, en caso de que lo necesiten.

Notas para mejorar las habilidades de las participantes:

Recomendar:

- Instalar *Visual Studio Code* en la computadora personal o laptop de la participante.
- Consultar las tarjetas de memorama cada que se requiera para afianzar los conocimientos.
- El uso de *shortcuts* y autocompletado como herramientas para escribir de manera más rápida y práctica el código HTML en *Visual Studio Code*.

Para profundizar en el tema se sugiere visitar las siguientes páginas:

- Khan Academy. (Sin fecha). Unidad: *Introducción a HTML/CSS: hacer páginas web*.
<https://es.khanacademy.org/computing/computer-programming/html-css> (Febrero 2020)
- w3Schools. (Sin fecha). *HTML Tutorial*.

<https://www.w3schools.com/html/default.asp> (Febrero 2020)

Links de apoyo:

- Páginas que les ayudarán a realizar la búsqueda, que se solicita en la primera parte de la actividad:
 - Álvarez, M. (2001). Qué es HTML .
<https://desarrolloweb.com/articulos/que-es-html.html>
 - MDN Web Docs moz://a. (Sin fecha). *Lista de Elementos HTML5*.
https://developer.mozilla.org/es/docs/HTML/HTML5/HTML5_lista_elementos (Febrero 2020)
 - Bartolomé, M. (Sin fecha). *Lista de Etiquetas HTML*.
<https://www.mclibre.org/consultar/htmlcss/html/html-etiquetas.html> (Febrero 2020)
 - MDN Web Docs moz://a. (Sin fecha) . *Referencia de Elementos HTML*.
<https://developer.mozilla.org/es/docs/Web/HTML/Elemento> (Febrero 2020)

Referencias:

- Pino, J. (Sin fecha). ¿Qué es el HTML?.
<https://devcode.la/blog/que-es-html/> (febrero 2020).
- Álvarez, M. (2001). Qué es HTML.
<https://desarrolloweb.com/articulos/que-es-html.html> (Febrero 2020).

Anexos:

MWT1A2_anexos.

Plantilla 1: InfografíaHTML.

Actividad 3: Agregando nuevos elementos a tu página web

Aprendizajes esperados: Usar los componentes HTML dentro de una página web. Experimentar con las propiedades de los componentes HTML.		Duración de la actividad: 5 horas.
Recursos 1. Hojas de papel o fichas bibliográficas. 2. Lápiz, pluma o colores. 3. Tijeras.	Evidencia/Producto Evidencia: <ul style="list-style-type: none"> Fichas de memorama sobre cada etiqueta (citas, comentarios del HTML, negritas, itálicas, saltos de línea, espaciadores horizontales, listas, listas anidadas y ligas así como la estructura de elementos anidados) con sus respectivos ejemplos. Archivo modificado de la parte dos de la actividad MWT1A2 con los nuevos componentes de HTML: citas, comentarios del HTML, negritas, itálicas, saltos de línea, espaciadores horizontales, listas, listas anidadas y ligas así como la estructura de elementos anidados. 	Retroalimentación/Evaluación Retroalimentación: <ul style="list-style-type: none"> Las fichas de memorama deberán contener cada uno de los aspectos sugeridos en la primera parte, (punto dos) siguiendo el formato que se solicitó desde la actividad anterior. El archivo deberá visualizarse con los elementos añadidos dentro de un navegador. La página web deberá cotener todos los elementos HTML requeridos: (Textos en diferentes formatos, saltos de líneas, listas, ligas, atributos, imágenes y ligas a otras páginas o dentro de esta).
Desarrollo de la actividad: Primera parte: Agregando nuevas etiquetas. Sugerencia de tiempo invertido: 2 horas. En esta actividad las participantes elaborarán una nueva búsqueda de etiquetas de HTML (negritas, itálicas, saltos de línea, espaciadores horizontales, citas, listas, listas anidadas, ligas y comentarios del HTML) y la estructura de elementos anidados para elaborar nuevas fichas (siguiendo las instrucciones de la actividad anterior) que se puedan consultar posteriormente si lo requieren; asimismo deberán modificar el archivo del proyecto que se está desarrollando desde las actividades anteriores para integrar los nuevos componentes aprendidos.		

1. Se solicitará a la participante realizar una búsqueda en internet de las etiquetas de html correspondientes con las funciones: Formato de texto, comentarios del HTML, listas y ligas. Se deberán considerar los siguientes aspectos:
 - Nombre de la etiqueta.
 - Función.
 - Atributos principales.

Segunda parte: Modificando mi proyecto.

Sugerencia de tiempo invertido: 3 horas.

Basándose en el archivo elaborado en la actividad 2 (*index.html*), la participante incorporará las nuevas etiquetas aprendidas.

1. Se solicitará abrir el archivo *index.html* generado en la actividad anterior.
2. Se indicará a la participante que aplique las nuevas etiquetas actividad para modificar el texto generado (negritas **** ****, itálicas *<i>* **, saltos de línea **
** **<p>** **<pre>**, espaciadores horizontales **<hr>**)
3. Posteriormente se solicitará que la participante genere mediante las etiquetas (citas **<q>**, listas **** **** **<dl>** **<dt>** **<dd>**, listas anidadas **** y comentarios **<!-- -->**) una cita, un lista, una listas anidadas y un comentario incorporando el contenido que considere pertinente en relación con el tema elegido para su página web.
4. Que incorpore en su página un enlace local y un enlace global que redireccionen a otras páginas web mediante el uso de la etiqueta de (ligas **<a>** ****).
5. Al finalizar la incorporación de las etiquetas, deberá visualizar las modificaciones de su archivo en el navegador.

Notas para apoyar la actividad:

- Podrán hacer uso de los *shortcuts* para escribir de forma más fácil y rápida su código, esto es escribiendo la etiqueta sin los picoparentesis (< >) y dando una tabulación, para generar el autocompletado de la etiqueta.
- Deberán escribir su código y guardarlo de manera constante para visualizarlo en el navegador y así observar sus modificaciones o avances.
- Se recomienda apoyar a las participantes a despejar sus dudas respecto al uso del editor de textos, y a la aplicación de los conocimientos adquiridos al desarrollo de su página web, en caso de que lo necesiten.

Links de apoyo:

- Digital Guide IONOS by 1&1. (2019). *Tutorial básico de HTML: crea tu propio proyecto web*. <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/aprende-html-tutorial-para-principiantes/> (Febrero 2020)
- Lucas, Gabriel. (2010). *CREACIÓN DE PÁGINAS WEB. UNIDAD 7 - Maquetación web* .

<https://sites.google.com/site/glgcreaciondepaginasweb/home/unidad-7> (Febrero 2020)

- GooglePlay. (Sin fecha). *Aprende HTML [Apps on Google Play]*.
<https://play.google.com/store/apps/details?id=com.sololearn.htmltrial> (Febrero 2020)
- World Tech Makers. (2017). *Tips para maquetar una página web*.
<https://medium.com/@worldtechmakers/tips-para-maquetar-una-p%C3%A1gina-web-bc6b6788be1> (Febrero 2020)

Referencias:

- García , Patricia. (2002). Las carreras en Ingeniería en el marco de la globalización: una perspectiva de género. *Revista Latinoamericana de Estudios Educativos* XXXII, (3), 91_105
<https://www.redalyc.org/articulo.oa?id=27032305> (Febrero 2020)
- Gauchat, Juan. (2012). *El gran libro de HTML5, CSS3 y Javascript* (3ª ed.). Barcelona, España: Marcombo.

Actividad 4: “El Konmarie” del HTML

Aprendizaje esperado: Usar los componentes HTML que permiten agrupar secciones en un sitio web y su visualización.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Hojas de papel o fichas bibliográficas. 2. Lápiz, pluma o colores. 3. Tijeras. 	Evidencia: <ol style="list-style-type: none"> 1. Fichas de memorama sobre cada etiqueta(span, div, Id y class) con sus respectivos ejemplos. 2. Archivo modificado de la parte uno de la actividad MWT1A3 para integrar los nuevos componentes de HTML: id, clases, estilos, atributos, etc. 	Retroalimentación: <ol style="list-style-type: none"> 1. Identificar las dificultades de la participante durante el desarrollo del juego de memorama. 2. El archivo deberá visualizarse dentro de un navegador integrando los nuevos elementos abordados durante la actividad. 3. La página web deberá de tener todos los elementos HTML requeridos.
Desarrollo de la actividad:		
<p>Presentación: El método <i>Konmari</i> propuesto por la autora japonesa Marie Kondo consiste en un sistema de organización, con el cual se pretenden optimizar espacios como oficinas y hogares, a partir agrupaciones lógicas relacionadas con características de características establecidas. Las etiquetas y atributos que se revisarán en esta actividad, conforman la base estructural de las páginas web, por lo que la organización lógica atañe no únicamente a la creatividad, sino a la mejor implementación y desarrollo de páginas web.</p> <p>Primera parte: Explorar la composición en páginas web.</p>		

Sugerencia de tiempo invertido: 1 hora 30 minutos.

Se investigará sobre las etiquetas de bloques e identificarán su uso en páginas web.

1. Las participantes realizarán una búsqueda puntual sobre nuevas etiquetas (span y div) y atributos (id y class) de HTML5, con el objetivo de identificar las características de su aplicación en páginas web, para posteriormente elaborar nuevas fichas, siguiendo el formato establecido en la MWT1A2.
2. La participante identificará los nuevos elementos de esta actividad, desplegando la ventana Inspect (clic derecho, Inspect Element), en páginas web relacionadas con los temas:
 - Ley General de Acceso de las Mujeres a una Vida Libre de Violencia.
 - Derechos sexuales y reproductivos .
 - Seguridad digital para mujeres.

La temática será decisión de la participante.

Segunda parte: Memoria y etiquetas.

Sugerencia de tiempo invertido: 30 minutos.

Se consolidarán las fichas realizadas durante las actividades pasadas para construir un juego de memorama.

1. Con las fichas realizadas en la MWT1A2, MWT1A3 y MWT1A4, se conforma el material suficiente para realizar un memorama, con el cual se pretende reforzar la identificación de las etiquetas de HTML5 vistas durante éste taller, además de fungir como un glosario de consulta en posteriores actividades.
2. Se realizarán de una a tres rondas de memorama, individual, en parejas o equipos, según la disposición de las participantes.

Tercera parte: Re-diseñar, ajustar y mostrar.

Sugerencia de tiempo invertido: 3 horas.

Partiendo del archivo *index.html* que modificó en la MWT1A3 se integrarán las nuevas etiquetas y atributos aprendidos.

1. Abrir el archivo *index.html* generado en la actividad anterior.
2. La participante deberá aplicar las nuevas etiquetas (div y span) y atributos (id y class).
 - Para la integración de los nuevos elementos, la participante programará una Navbar y un Footer, cada cual teniendo de dos a tres elementos identificables, estos pueden ir desde “contacto”, “redes sociales”, “acerca de nosotros”, etc. o una modificación a su archivo que integre los conocimientos adquiridos durante esta actividad.
3. Al finalizar la incorporación de las etiquetas y atributos, la participante deberá visualizar su archivo en el navegador

(apoyándose de Inspect).

Notas para apoyar la actividad:

En “Explorar la composición en páginas web”, en el caso que la participante elija otro contenido, será indispensable supervisar que la temática responda de manera afirmativa por lo menos a una de las preguntas que se enuncian en el apartado “Preguntas para que mi proyecto tenga Perspectiva de Género” del *Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios* que marca los límites generales de este proyecto.

- ¿Mi propuesta tiene la intención de mejorar las condiciones de vida de las mujeres y niñas de algún colectivo?
- ¿Mi propuesta está encaminada en retomar alguna problemática que afecta de manera directa a las mujeres y niñas de algún colectivo?
- ¿Mi propuesta toma como eje la vida de mujeres y niñas de algún colectivo?
- ¿Mi propuesta está encaminada a favorecer alguna situación donde mujeres y niñas de algún colectivo se encuentren en desventaja?
- ¿Mi aporte busca contribuir en la transformación de modelos sociales y culturales que estereotipan el hacer y quehacer de las mujeres o niñas?
- ¿Mi propuesta busca prevenir o dar atención a una problemática de violencia contra mujeres o niñas?
- ¿Mi propuesta propone visibilizar la participación de las mujeres en un campo donde son poco visibles?
- ¿Mi propuesta propone informar sobre derechos humanos de las mujeres?

En “Re-diseñar, ajustar y mostrar”, punto dos, se pretende que el proyecto, cuente con las características de Header, Body y Footer, de no ser el caso, se sugiere atender en primera instancia esas características. Con ello, el producto final, contará con una mayor cantidad de elementos aplicados.

Se sugiere se proporcionen las “referencias de práctica” a las participantes y en caso de que así se requiera, el monitoreo para el cumplimiento de Re-diseñar, ajustar y mostrar.

Links de apoyo:

Primera parte:

- INMUJER Instituto Nacional de las Mujeres.(Sin fecha). *Puntogénero: formación para la igualdad*
<http://puntogenero.inmujeres.gob.mx/> (Febrero 2020)
- GOBIERNO DE MÉXICO. (Sin fecha). *Instituto Nacional de las Mujeres. Los Derechos Humanos de las Mujeres.*,
<https://www.gob.mx/inmujeres/articulos/los-derechos-humanos-de-las-mujeres?idiom=es> (Febrero 2020)

Segunda parte:

- Materialize. (Sin fecha)). *Navbar*.
<https://materializecss.com/navbar.html> (Febrero 2020)
 - *Footer*.
<https://materializecss.com/footer.html> (Febrero 2020)

Notas para mejorar habilidades de participantes:

Se recomienda visitar los siguientes links (en inglés):

- w3schools.com. (Sin fecha). HTML Id Attribute.
https://www.w3schools.com/hTmL/html_id.asp (Febrero 2020)
 - HTML class Attribute.
https://www.w3schools.com/hTmL/html_classes.asp (Febrero 2020)
 - HTML <div> Tag.
https://www.w3schools.com/tags/tag_div.ASP (Febrero 2020)
 - HTML Tag.
https://www.w3schools.com/tags/tag_span.asp (Febrero 2020)

Referencias:

- Kondo, Marie. (2016). *La magia del orden*. España: Aguilar.

Actividad 5:Ponle estilo a tu página (parte 1)

Aprendizaje esperado: Aplicar estilos en una página web, así como poner en práctica las reglas, propiedades y valores de los componentes CSS.		Duración de la actividad: 3 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Anexos <i>MWT1A5_Anexos</i>. 2. Archivo WordPerfect presentations <i>MWT1A5_PáginaWeb.pptx</i> 	Evidencias: <ol style="list-style-type: none"> 1. Un recuadro con las traducciones de los selectores contenidos en la plantilla <i>Instrucciones</i> de MWT1A5_anexos. 2. Un archivo <i>MWT1A5_PáginaWeb</i> en WordPerfect Presentations con la simulación de la aplicación de las <i>Instrucciones</i> a su formato. 3. Una carpeta <i>MWT1A5_ActividadPintorasMexicanas</i> que contenga tres versiones del archivo index.html (formato en línea 1, formato dentro de html 2 y formato ligado desde un archivo css 3) y un archivo style.css. 	Retroalimentación: <ol style="list-style-type: none"> 1. El archivo CSS contará con todas las etiquetas que se hayan solicitado para darle estilo al archivo HTML en la tercera parte de la actividad. 2. Se comprobará mediante la visualización de las tres páginas web en el navegador, que la declaración de las etiquetas en CSS se realizó de acuerdo a lo solicitado, y que luzcan de igual forma.

Presentación:

Durante siglos pasados, en México a las mujeres se les asignó por género el lugar de lo privado y lo doméstico, por ello, tenían menos participación en las decisiones sociales; políticas y económicas; inclusive si participaban en un espacio público sus actividades artísticas eran menos reconocidas. Sin embargo, poco a poco comenzaron a ganar terreno estos espacios, principalmente a lo largo de todo el Siglo XX. Este contexto no excluyó a las pintoras mexicanas, entre ellas Frida Kahlo y María Izquierdo, a quienes se les reconoce como las primeras artistas que rompieron con las barreras sociales y culturales producto de la misoginia impedían que su obra fuerareconocida a nivel nacional.

Actualmente, debido a las resistencias de estas y otras mujeres las desigualdades han comenzado a disminuir y las mujeres ya pueden acceder de manera más sencilla a recursos como la educación y el reconocimiento en diversos ámbitos de su desenvolvimiento. Esta herencia de derechos, en la siguiente actividad, participantes, al igual que las artistas citadas, desarrollarán sus habilidades creativas y artísticas a través de dar estilo a las páginas web con CSS.

Primera parte: Coloreando en la web.

Sugerencia de tiempo invertido: 1 hora.

Las participantes colorearán y darán estilo a una plantilla de página web en WordPerfect Presentations, de tal forma que puedan inferir qué es CSS, cuál es la regla de sintaxis, qué son los selectores, las propiedades y los valores. El ejercicio consiste en:

1. Se le proporcionará a la participante:

- Una plantilla de página web en blanco y negro en un archivo de WordPerfect Presentations (se recomienda utilizar el archivo *MWT1A5_PáginaWeb* de los anexos).
- Una hoja con propiedades de CSS a aplicar para darle estilo a la plantilla de página web (se recomienda utilizar la plantilla denominada *Instrucciones* de los anexos *MWT1A5_anexos*).

2. Una vez que la participante cuente con los materiales anteriores:

1. Llenará un recuadro similar al siguiente usando las instrucciones. Los selectores, propiedades y valores de la plantilla instrucciones:

	A qué se le aplicará el cambio	Qué cambio se aplicará	
	Selector	Propiedad	Nuevo valor
Instrucción	h1	text-align:	center

Traducción	encabezado	alineación del texto	centro
Instrucción	p	margin-left:	20px
Traducción	párrafo	margen izquierdo	20 puntos

2. La participante transcribirá cada instrucción de la hoja que se le proporcionó, y deberá buscar en internet el significado de cada propiedad y valor, de tal forma que pueda llenar las filas de “traducción”.
3. Una vez llenado el recuadro, deberá proceder a ejecutar (simular) cada una de las instrucciones en el *MWT1A5_PáginaWeb*. De acuerdo con el ejemplo anterior, la participante tendría que cambiar la alineación del encabezado al centro, y aumentar a 20 puntos el margen izquierdo del párrafo.
3. Finalizado el juego, se motivará la reflexión utilizando preguntas guía que estimulen el diálogo, por ejemplo:
 - De acuerdo con el ejercicio anterior, ¿en qué crees que consiste CSS?
 - ¿Qué es un selector?
 - ¿Qué es una propiedad?
 - ¿Qué es un valor?
 - ¿En qué orden van el selector, la propiedad y el valor en la declaración de CSS?

Para profundizar en la definición de CSS y otras cuestiones teóricas, se recomienda consultar las fuentes que se anexan al final de este documento u otras.

Segunda parte: Mi primera pintura web.

Sugerencia de tiempo invertido: 2 horas.

Se presentará a las participantes los siguientes elementos teóricos:

- Cómo dar estilo a una página web desde el mismo archivo HTML: en la misma línea de cada etiqueta HTML, y en una hoja de estilo interna.
- Cómo dar estilo a través de una hoja de estilo externa (ligando el archivo HTML y el CSS).

Para ello, se explicará cada elemento, o se solicitará a la participante una búsqueda en internet para explicitar cada punto (se pueden utilizar las fuentes anteriormente sugeridas u otras distintas).

Una vez resueltos los puntos anteriores, se le proporcionará a la participante un breve código HTML predeterminado, al cual tendrá que dar estilo a través de una secuencia de instrucciones también determinadas. Dicha secuencia tendrá que ser aplicada en las tres distintas formas de insertar una hoja de estilo anteriormente se vistos (en la misma línea de cada etiqueta HTML, en el

mismo archivo HTML, y ligando un archivo HTML a una hoja de estilo externa).

1. Se presentará un breve código HTML con todos los elementos necesarios para aplicar las instrucciones que se proporcionarán (se recomienda usar el archivo *index.html* anexo en la carpeta *MWT1A5_ActividadPintorasMexicanas*).
2. Se brindará la secuencia de instrucciones a seguir. Se recomienda utilizar la siguiente:
 - a. Aumentar el tamaño de letra al título.
 - b. Aumentar el tamaño de letra del primer subtítulo.
 - c. Cambiar el color de fondo del segundo subtítulo.
 - d. Aumentar el tamaño de letra al primero y al tercer párrafo.
 - e. Con un selector de clase, aplicar un margen del lado izquierdo a la etiqueta div.
 - f. Establecer el mismo color de letra a todos los párrafos y al título.
 - g. Cambiar el color del fondo de toda la página.
3. Una vez que la participante cuente con los elementos anteriores, tendrá que abrir el archivo HTML y una pestaña del navegador en la que visualizará paralelamente cada uno de los cambios aplicados al código.
4. Primero ejecutará las instrucciones insertando los atributos de estilo en la misma línea de cada etiqueta HTML. Copiar la versión del archivo “index.html” resultante a “*index1.html*”.
5. En segundo lugar, ejecutará las instrucciones en el mismo archivo HTML, insertando una hoja de estilo interna. Copiar la versión del archivo “index.html” resultante a “*index3.html*”.
6. Por último, ejecutará las mismas instrucciones en un archivo externo al HTML, al cual llamará “style.css”, y posteriormente ligará ambos archivos. Copiará la versión del archivo “index.html” resultante a “*index3.html*”.

Finalizado todo el proceso, se incitará la reflexión mediante preguntas generadoras, por ejemplo:

- ¿Qué diferencias observaste entre las tres formas de insertar una hoja de estilo?
- ¿Qué similitudes?
- ¿Qué ventajas y desventajas tiene cada forma de aplicar CSS?
- ¿Cuál forma se te facilitó más y cuál se te dificultó?
- ¿Qué observaste en el navegador?
- ¿Cuál fue el resultado visual de cada forma de aplicar CSS?

Notas para para apoyar la actividad:

- Será necesario que la participante cree una nueva estructura de carpetas. En su carpeta de evidencias de este módulo, creará una carpeta de nombre “MWT1A5_ActividadPintorasMexicanas”; dentro de esta, tendrá tres carpetas más: una de nombre “html”, otra llamada “css”, y una más “img”. Dentro de la carpeta *html*, la participante deberá guardar el código

predeterminado que se proporcionó en la tercera parte de la actividad (con el nombre “index.html”), mientras que en la carpeta “css”, guardará el archivo de CSS con el nombre “style.css” que creó para darle estilo al HTML.

- Es esencial fomentar en las participantes la buena práctica de crear una nueva estructura de carpetas por cada página web que desarrollen.
- En la tercera parte de la actividad, las participantes deberán visualizar en el navegador el resultado de cada una de las tres formas de dar estilo al código HTML.

Links de apoyo:

Para aprender más:

- Eguiluz, J. (Sin fecha). *Introducción a CSS*.
<https://uniwebsidad.com/libros/css> (Febrero 2020)
- Khan Academy. (Sin fecha). *Unidad: Introducción a HTML/CSS: hacer páginas web*.
<https://es.khanacademy.org/computing/computer-programming/html-css#intro-to-html> (Febrero 2020)
- w3schools.com. (Sin fecha). *CSS Tutorial*.
<https://www.w3schools.com/css/default.asp> (Febrero 2020)
- códigofacilito. (2015). *Selectores de CSS*.
<https://codigofacilito.com/articulos/selectores-de-css> (Febrero 2020)

Para poner en práctica los conocimientos:

- CSS Diner. (Sin fecha). *Where we feast on CSS Selectors!*
<https://laboratoria.github.io/css-diner/> (Febrero 2020)

Anexos:

- *MWT1A5_Anexos.pdf*
Plantilla 1: Instrucciones
- *MWT1A5_PlantillaPáginaWeb.pptx*

Actividad 6: Ponle tu estilo a tu página (parte 2)

Aprendizaje esperado: Utilizar los componentes CSS para incorporar diseño gráfico a una página Web.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Hojas de papel tamaño carta o fichas bibliográficas. 2. Lápiz, pluma o colores. 3. Tijeras. 4. Carpeta <i>MWT1A6_ActividadPintorasMexicanas.zip</i>. 5. <i>MWT1A6_PlantillaEjemplo.pdf</i> 	Evidencia: <ol style="list-style-type: none"> 1. Fichas que contendrán la información acerca de propiedades básicas de CSS: funciones y valores. 2. Archivo <i>index.html</i> de la carpeta anexa <i>MWT1A6_ActividadPintorasMexicanas</i> con estilo añadido. 3. Archivo <i>Index.html</i> (ubicado en la carpeta <i>MWT1A1_NombreDelProyecto</i> del proyecto principal con estilo añadido. 	Retroalimentación: <ol style="list-style-type: none"> 1. Las fichas de propiedades básicas de CSS deberán contener cada uno de los aspectos sugeridos en la primera parte “Mis herramientas de estilo” y siguiendo el formato que se requiere: en una cara de la ficha deberá ir la función de la propiedad y al reverso, la propiedad junto con los valores que ésta puede tomar. 2. Se sugiere usar el anexo <i>MWT1A6_PlantillaEjemplo</i> como referente de producto final al añadir las propiedades CSS del archivo anexo <i>Index.html</i> de la carpeta <i>MWT1A6_ActividadPintorasMexicanas</i>. 3. El archivo <i>Index.html</i> de la carpeta del proyecto (<i>MWT1A1_NombreDelProyecto</i>) con estilo añadido siguiendo el maquetado que se ha venido trabajando desde la actividad 1. 4. Se deben visualizar cada una de las modificaciones de estilo en el navegador, si alguna modificación no

		se puede observar es porque se ha escrito mal una regla CSS.
Desarrollo de la actividad:		
<p>Primera parte: Mis herramientas de estilo.</p> <p>Sugerencia de tiempo invertido: 1 hora y 30 minutos.</p> <p>Las participantes harán una búsqueda en internet sobre las propiedades básicas de CSS que se sugieren a continuación: <i>font-family, font-size, color, text-align, width, height, border-style, border-width, border-color, margin, padding, visibility, text-shadow, box-shadow, float, list-style-type, list-style-position, y background-image</i>.</p> <p>Deberán dividir una hoja de papel en ocho partes y recortar los rectángulos formados (también se pueden emplear fichas bibliográficas); en ellas se escribirá al frente la función de la propiedad y al reverso la propiedad junto con los posibles valores que ésta puede tomar.</p> <p>Segunda parte: Mi segunda pintura web.</p> <p>Sugerencia de tiempo invertido: 1 hora y 30 minutos.</p> <p>Las participantes añadirán estilo al archivo sugerido en <i>index.html</i> , que se ubica en la carpeta anexa a esta actividad: <i>MWT1A6_ActividadPintorasMexicanas</i>. Podrán efectuar cambios adicionales a su consideración e incluir otras propiedades que no se mencionan en esta actividad, así como dar estilo mediante selectores, clases, id, etcétera. Podrán crear cuantos consideren necesarios, siempre y cuando cubran las especificaciones listadas en este ejercicio. Se sugiere consultar la plantilla “<i>Pintoras mexicanas en el siglo XX</i>” en el anexo <i>MWT1A6_PlantillaEjemplo.pdf</i>.</p> <ol style="list-style-type: none"> 1. Solicitar la creación de una nueva carpeta con nombre “css” dentro de <i>MWT1A6_ActividadPintorasMexicanas</i>, esta se replicará en el portafolio de evidencias correspondiente a este taller . Dentro de esta carpeta nueva, la participante creará un archivo con nombre “style.css” 2. La participante ligará este archivo CSS al HTML <i>index.html</i> que se le proporcionó. 3. Aplicará las propiedades de estilo siguiendo las características que se enlistan en el apartado siguiente: <ul style="list-style-type: none"> ● Cambiar el tipo y color de letra a elección de la participante. ● Cambiar el tamaño de letra a modo que el título principal sea el más grande, los subtítulos más pequeños y el resto del texto sea más pequeño que los subtítulos. El tamaño se especificará en <i>px</i>. ● El título deberá ir centrado. ● La información correspondiente a la biografía de cada una de las pintoras deberán ir dentro de un contenedor con 		

borde de color, tamaño y tipo a elección de la participante.

- Dentro de este contenedor se deberá de ajustar la posición, altura y ancho de los contenedores necesarios para que se pueda visualizar la información de las biografías de las pintoras de forma horizontal, es decir, una a lado de la otra.
- Cambiar la altura y el ancho de cada imagen, para que todas tengan las mismas dimensiones.
- Cambiar el punto por default de la lista de “Referencias”, por un rectángulo o círculo sin relleno.
- Aplicar sombra al título o subtítulos, y al recuadro generado anteriormente.
- El margin y padding de los elementos será a la elección de la participante, considerando que debe de utilizar estas propiedades en al menos 2 elementos, cada uno.
- Se podrá dar color de fondo o poner una imagen de fondo a elección de la participante.
- Probar la propiedad “visibility” en un elemento de la página y observar qué sucede cambiando el valor de la propiedad.

Tercera parte: Mi obra maestra web.

Sugerencia de tiempo invertido: 2 horas

Las participantes darán estilo a su proyecto “MWT1A1_NombreDelProyecto”, aplicando los conocimientos adquiridos de CSS (añadir carpeta CSS, ligar archivos CSS a HTML, aplicación de reglas CSS).

Notas para apoyar la actividad:

- Se debe asegurar y fomentar que las participantes adopten como práctica la creación una estructura de carpetas, es decir, dentro de una carpeta de un proyecto de página web general, incluir dentro las carpetas: “css” e “img”, donde irán guardando los archivos correspondientes en cada una.
- Las participantes deberán escribir su código, lo guardarán e irán actualizando el navegador donde se visualizarán las modificaciones del estilo de la página.
- Se despejarán las dudas o preguntas que las participantes llegasen a tener respecto de la implementación de las propiedades para dar estilo a la página web.
- Las carpetas y archivos deberán crearse desde la consola.

Links de apoyo:

- Sobre las propiedades CSS:
 - Aprende Web. (2018). *Manual de CSS*.

- <https://aprende-web.net/css/index.php> (Febrero 2020).
 - Chuburu, Laura. (Sin fecha). *Principales propiedades CSS*.
<https://www.laurachuburu.com.ar/tutoriales/propiedades-css.php> (Febrero 2020).
- Para profundizar en el tema se sugiere visitar las siguientes páginas:
 - Khan Academy. (Sin fecha). *Unidad: Introducción a HTML/CSS: hacer páginas web*.
<https://es.khanacademy.org/computing/computer-programming/html-css> (Febrero 2020).
 - w3Schools. (Sin fecha). CSS Tutorial
<https://www.w3schools.com/css/default.asp> (Febrero 2020).

Referencias:

- Gómez, Germaine. (2000). Historia de Mujeres Artistas en México del siglo XX.
https://museodemujeres.com/images/biblioteca/historia_de_mujeres.pdf (Febrero 2020).

Anexos:

- MWT1A6_ActividadPintorasMexicanas.zip
Carpeta que contiene dos archivos, una carpeta llamada *img* que contiene una imagen, y un archivo "*index.html*" recurso de apoyo para practicar aplicación de estilos.
- MWT1A6_PlantillaEjemplo.pdf
Plantilla 1: Pintoras mexicanas en el siglo XX.

Actividad 7: Desde cero

Aprendizaje esperado: Generar una página web desde cero, usando todos los conocimientos adquiridos en las actividades anteriores.		Duración de la actividad: 8 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
	Evidencia: Maquetado del diseño del proyecto en formato <i>.ppt</i> o <i>.pdf</i> .	Retroalimentación: El maquetado deberá contener las categorías que contendrá la página web, con los temas organizados jerárquicamente, desplegados en un organizador gráfico.
Desarrollo de la actividad:		
<p>Sugerencia de tiempo invertido: 8 horas.</p> <p>Esta actividad se enfocará en articular los conocimientos adquiridos en las actividades anteriores y concretarlos en el desarrollo de una página web bajo pedido. Se pretende que las participantes puedan diseñar una página web enfocada a un proyecto de interés propio o en algún proyecto de la comunidad de la participante.</p> <ol style="list-style-type: none"> 1. Se solicitará a la participante que seleccione una temática para la elaboración de su página web, esta temática puede corresponder a algún proyecto de interés propio, de importancia para la comunidad de la participante o pertenecen a algún proyecto de PILARES como los de Educación para la Autonomía Económica. En cualquiera de los tres casos, la participante deberá atender a la incorporación de la Perspectiva de Género, para ello se recomienda revisar el documento <i>Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios</i> 2. Una vez que la participante haya elegido su proyecto, se le indicará que realice una investigación de las necesidades de difusión del mismo, ya sea por medio de la búsqueda de información de manera autónoma y/o por medio de entrevistas con la comunidad y las personas interesadas (clientes) para que pueda plantear un borrador de diseño de los apartados de la página. 3. Se otorgará el tiempo necesario a la participante, para poder reunirse con las personas interesadas o clientes en el diseño de la página así como para recabar la información que considere pertinente (comentarios, infografías, gráficos, fotografías, entre otros). 		

4. Una vez obtenida la información necesaria, se le indicará a la participante comenzar el maquetado del diseño del proyecto. Si lo requiere podrá entablar reuniones con la tallerista y las personas interesadas en la realización de la página web.

Se recomienda seguir los siguientes pasos para diseñar el proyecto:

- Definir las categorías que contendrá la página web.
 - Realizar una organización de los temas que contendrá.
 - Elaborar una estructura jerárquica de la página web, esto se puede hacer mediante el uso de un organizador gráfico (mapa conceptual).
 - Realizar el diseño arquitectónico del maquetado de cada página web utilizando el programa WordPerfect Presentations o algún programa similar que contenga las funciones necesarias para llevarlo a cabo.
5. Para finalizar la actividad, la participante tendrá que presentar el maquetado terminado del diseño del proyecto en formato .ppt o .pdf con el nombre “T1A7_MaquetadoProyectoPaginaWeb”, el cual tendrá que ser aprobado por las personas interesadas en la página web y la tallerista.

Notas para apoyar la actividad:

- Si la participante elige que su interés es otro, se recomienda que para la construcción de su contenido la tallerista supervise que la temática responda de manera afirmativa por lo menos a una de las preguntas que se enuncian en el apartado “Preguntas para que mi proyecto tenga Perspectiva de Género” del *Escuela de Código para PILARES Guía de la Perspectiva de Género en el programa de estudios* que marca los límites generales de este proyecto. Estas son:
 - ¿Mi propuesta tiene la intención de mejorar las condiciones de vida de las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada en retomar alguna problemática que afecta de manera directa a las mujeres y niñas de algún colectivo?
 - ¿Mi propuesta toma como eje la vida de mujeres y niñas de algún colectivo?
 - ¿Mi propuesta está encaminada en favorecer alguna situación donde mujeres y niñas de algún colectivo se encuentren en desventaja?
 - ¿Mi aporte busca contribuir en la transformación de modelos sociales y culturales que estereotipan el hacer y quehacer de las mujeres o niñas?
 - ¿Mi propuesta busca prevenir o dar atención a una problemática de violencia contra mujeres o niñas?
 - ¿Mi propuesta propone visibilizar la participación de las mujeres en un campo donde son poco visibles?
 - ¿Mi propuesta propone informar sobre derechos humanos de las mujeres?
- Se sugiere que para esta actividad se realice un proyecto de PILARES Educación para la Autonomía Económica como los de

(Carpintería, Diseño de moda y textiles, Joyería, Huertos Urbanos, entre otros).

Actividad 8: Web de verdad

Aprendizaje esperado: Desplegar una página web a través de un servidor local.		Duración de la actividad: 5 horas 30 mins.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Computadora con acceso a internet. 2. Node.js, npm y expressjs. 3. El <i>MWT1A8_Anexo.pdf</i> 	Evidencia: <ol style="list-style-type: none"> 1. La página maquetada en la actividad MWT1A7 será programada en HTML y CSS. 2. La página deberá ser visible a través de una url local localhost:3000 3. Archivo con los comandos ejecutados. 4. Capturas de pantalla del proyecto visualizado en localhost:3000. 	Retroalimentación: La página deberá estar completa y se deberá de visualizar desde un servidor web local.
Desarrollo de la actividad:		
<p>Primera parte: ¿Qué hay detrás del internet?</p> <p>Sugerencia de tiempo invertido: 30 minutos.</p> <ol style="list-style-type: none"> 1. Presentará los siguientes casos u otros ejemplos, en los que la participante identificará características básicas del modelo cliente-servidor, reflejadas en experiencias cotidianas, posteriormente responderá de forma oral a preguntas problematizadoras que permitan formular una definición general y trazar un esquema sobre de dicho modelo. Véase “Casos 1 y 2” y “Preguntas problematizadoras” en el <i>MWT1A8_Anexo.pdf</i>. 2. La participante deberá realizar una búsqueda en la web, acerca del modelo cliente-servidor, se recomienda que ésta se lleve a cabo en el apartado de “imágenes” del buscador utilizado para comparar los esquemas con los realizados en el primer punto. <p>Haciendo uso del esquema más general del modelo, se deberá dar una interpretación de este (orientado a la parte del despliegue de sitios web), investigando más a detalle las partes que lo conforman, características y procesos involucrados. Verifique la información, con la finalidad de esclarecer dudas o sesgos.</p> <p>Segunda parte: Web a medida.</p> <p>Sugerencia de tiempo invertido:4 horas.</p>		

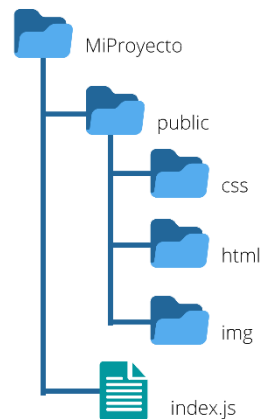
Recuperar el producto de la MWT1A7 para programar el sitio web (HTML y CSS) del nuevo proyecto.

Tercera parte: Convirtiendo mi computadora en servidor.

Sugerencia de tiempo invertido: 1 horas.

La participante levantará un servidor en su computadora para visualizar su proyecto desde la web. Para ello seguirá los siguientes pasos:

1. Generará la siguiente estructura de carpetas utilizando únicamente la línea de comandos. (Uso de comandos mkdir, cd, touch)



1. Colocará los archivos correspondientes al maquetado de la MWT1A7 a través de la línea de comandos (uso de mv o cp).
2. La participante deberá seguir las instrucciones de “Servidor a la Express” (las cuales pueden encontrarse como anexo a esta actividad) y en caso de tener alguna duda durante la realización del ejercicio, podrá solicitar apoyo a la tallerista.

Como evidencia deberá adjuntarse el archivo con los comandos ejecutados, así como capturas del proyecto visualizado en localhost:3000.

Notas para apoyar la actividad:

Para la realización de “¿Qué hay detrás del internet?”, se sugiere utilizar esquemas que comparten características con el

proporcionado en el apartado de *links de apoyo* o en su defecto, utilizar dicho esquema.

Links de apoyo:

Conocimientos técnicos

- SolariConsultants. (2011). *Qué es World Wide Web WWW?* [video]
<https://www.youtube.com/watch?v=b1hIBjsdo2A> (Febrero de 2020)
- unComo. (2012). *Cómo funciona una Web.* [video]
<https://www.youtube.com/watch?v=XHOo3EQGmfA> (Febrero de 2020)
- ZoneClass. (2017). ¿Qué es un servidor web? | zoneclass.com 🖱️ [video]
<https://www.youtube.com/watch?v=Zrybo9Pt7ys> (Febrero de 2020)

Ejemplo de esquema

- Negociosdelweb.com. (Sin fecha) ¿Qué es un servidor?
<https://www.negociosdelweb.com/wp-content/uploads/2018/09/imagen-2.jpg> (Marzo de 2020)

Actividad 9: De mí para el mundo

Aprendizaje esperado: Publicar una página web en alguna plataforma gratuita de <i>hosting</i> .		Duración de la actividad: 3 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Una cuenta en GitHub o GitLab. 2. Hojas de papel. 3. Lápices, plumas o plumones. 	<p>Evidencia:</p> <ol style="list-style-type: none"> 1. Página web visible desde el servicio de páginas de GitHub y GitLab. <p>Producto:</p> <ol style="list-style-type: none"> 1. Repositorio en la plataforma GitHub o GitLab. 	<p>Retroalimentación:</p> <ol style="list-style-type: none"> 1. La descripción de su perfil de GitHub debe resaltar características de los conceptos aprendidos y desarrollados por la participante. 2. El proyecto realizado durante este taller debe estar alojado en un repositorio de GitHub o GitLab que incluya su estructura de carpetas (HTML, CSS y JS). 3. El link generado para su página en GitHub pages debe ser accesible desde cualquier computadora o celular. 4. Todos los elementos desarrollados en sus archivos HTML y CSS deben ser visualizados en su página alojada en GitHub pages. <p>Evaluación:</p> <ol style="list-style-type: none"> 1. El código debe ser correcto y debe cumplir con la función definida desde la actividad MWT1A7. 2. La página debe ser visible desde una URL de la plataforma GitHub o GitLab.

Desarrollo de la actividad:		
<p>Primera parte: ¿Qué necesito para exponer mi obra?</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>Se le solicitará a la participante:</p> <ol style="list-style-type: none"> 1. Elaborar nuevas fichas de consulta (con el mismo formato que en las actividades anteriores de este taller), a partir de la búsqueda de los conceptos hosting (gratuito y de pago), dominio y url (se recomienda utilizar las referencias adjuntas al final de este documento). 2. Realizar un cuadro comparativo con las ventajas y desventajas que identifique al utilizar un hosting de pago y uno gratuito. 3. Compartir su cuadro comparativo de tal forma que se puedan discutir las ventajas y desventajas que ella identificó, agregar otras, o inclusive eliminar algunas (asegurarse de explicar siempre la razón de tal cambio). <p>Finalmente, se le presentarán a la participante algunos ejemplos de hosting de pago y gratuito.</p> <p>Segunda parte: Exponiendo al mundo mi obra de arte.</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>La participante subirá su proyecto desarrollado a lo largo de las actividades 7 y 8 de este taller a un hosting gratuito. Para ello, se le solicitará realizar las siguientes instrucciones:</p> <p>Nota: Se sugiere utilizar GitHub pages.</p> <ol style="list-style-type: none"> 1. Crear una cuenta en GitHub. https://github.com/ (o GitLab) 2. Editar su perfil, agregar una fotografía y una descripción de su personalidad, por ejemplo “Participante en proyectos culturales que benefician a mi entorno”, “Aprendiz en la Escuela de Código, PILARES”. 3. Crear un repositorio público en GitHub, llamarlo como su proyecto y agregar una descripción de este. 4. Subir su proyecto a GitHub de forma gráfica, no usar la consola de git para esta actividad. 5. Crear una rama “gh-pages” en el repositorio del proyecto. <p>Nota: La rama debe llevar este nombre, no puede ser cambiado.</p> <ol style="list-style-type: none"> 6. Ubicarse en la rama creada en el punto anterior. 7. Ir a settings/configuración y en la parte inferior, en la sección denominada “GitHub Pages” observar la <i>url</i> que se muestra. 8. Copiar la <i>url</i> y pegarla en algún navegador. 9. Visualizar que su página se haya subido correctamente al hosting gratuito. 10. Ingresar desde otra pestaña del navegador, desde un navegador distinto, en el celular o bien en otra computadora, para 		

visualizar que la página se muestra cuando se accede a la url.

Tercera parte: ¿Qué tal está GitHub?

Sugerencia de tiempo invertido: 1 hora.

Una vez que la participante subió su proyecto a la plataforma GitHub, se solicitará que escriba la respuesta de las siguientes preguntas, una ficha por cada pregunta:

1. ¿Cómo fue tu experiencia en GitHub?
2. ¿Qué crees que sea GitHub?
3. ¿Para qué crees que funciona?
4. ¿Volverías a utilizarlo?

Se le indicará guardar las fichas hasta que nuevamente se requiera utilizarlas en el siguiente taller.

Notas para apoyar la actividad:

- Asegurarse que en la descripción de su perfil de GitHub, las participantes agreguen un texto que fomente su autoestima o que resalte sus cualidades.
- En caso de que la participante no conozca el manejo de git, asegurarse de que todas las instrucciones de la segunda parte de la actividad se realicen de forma gráfica. En caso contrario, la participante podrá usar la terminal para subir su proyecto a GitHub pages.

Links de apoyo:

Páginas sugeridas para la primera parte de la actividad:

- HostGator Español. (2017). *¿Qué es un Web Hosting?* [video]
<https://www.youtube.com/watch?v=8gEdpb7blb4> (Marzo 2020)
- Bluuweb ! (2016). *Qué es el hosting, Dominio y Diseño Web | Primeros pasos para tu proyecto web.* [video]
<https://www.youtube.com/watch?v=b8TcajoGRvk> (Marzo 2020)
- Borja, C. (2018). *Que es HOSTING y DOMINIO de una PÁGINA WEB* ✓✓ Fácil y rápido. [video]
<https://www.youtube.com/watch?v=BEADXj5QvF0> (Marzo 2020)
- Vera, O. (2016). *Qué es dominio, qué es hosting | Conceptos básicos para iniciar tu negocio online.* [video]
<https://www.youtube.com/watch?v=60QJm2S-Vec>
- Profesional Hosting. (2018). *¿Qué es un hosting o Alojamiento Web? ¿Tipos de hosting?* [video]
<https://www.youtube.com/watch?v=5OjSrKZ964&t=164s>

- HTML Rules. (2019). *Partes de una URL*. [video]
<https://www.youtube.com/watch?v=zKUxt5Tlu9k>
- Fons, R. (2017). *¿Qué es un DOMINIO en Internet?* [video]
<https://www.youtube.com/watch?v=lZOMytFG1X4>
 - ¿Qué es una URL? [video]
<https://www.youtube.com/watch?v=C2Yd0W1CR5U>

Páginas sugeridas para la segunda parte de la actividad.

- Fazt. (2018). *Github Pages, Hosting Gratuito para Páginas web*. [video]
<https://www.youtube.com/watch?v=QaxgF4v4hms&t=1s> (Marzo 2020)
- Santana, D. (Sin fecha). *Como crear una cuenta en Github y subir tus programas*. [video]
<https://www.youtube.com/watch?v=YYiEz8nBokE> (Marzo 2020)
- Falcón, M. (2017). *Cómo crear tu página web gratis en GitHub*-www.falconsolutions.cl. [video].
<https://www.youtube.com/watch?v=f3S5oCCYto8> (Marzo 2020)
- Laboratoria. (2017). *3.Cómo crear una cuenta en Github*. [video]
<https://www.youtube.com/watch?v=kU31qRwz27c> (Marzo 2020)

Notas para mejorar habilidades de participantes:

En caso de que la participante desee poner a prueba sus habilidades, puede subir su proyecto a un hosting gratuito adicional a GitHub pages. Se recomienda que se suba a Heroku y ocupe todas las herramientas digitales necesarias para lograr este objetivo. Esto deberá realizarlo con la menor ayuda posible.

Taller 2: Creando un punto de venta en internet

En honor a **Montserrat Boix Piqué (Barcelona, 1960)**. Licenciada en Ciencias de la Información por la Universidad de Barcelona. Debido a su formación en el uso de las TIC, comunicación y género, señala los retos que enfrenta la Sociedad de la Información para disminuir la brecha digital de género.

Actualmente colabora en diversos servicios informativos internacionales. Es investigadora, consultora y formadora en comunicación. En 1997 creó *Mujeres en Red* periódico Feminista para el intercambio de información en internet sobre Derechos Humanos y Empoderamiento de mujeres hablantes del idioma español en América Latina y Europa. Actualmente el sitio <http://www.mujeresenred.net> es un referente en contenidos sobre feminismo.

Una de las motivaciones de Montserrat Boix Piqué para la creación de este proyecto interdisciplinarios fue pensar las Tecnologías de la Información y la Comunicación (TIC) como una herramienta para el empoderamiento de las mujeres pues entiende la información como un elemento de transformación social.

Competencia del taller: Programar y diseñar un punto de venta en internet usando JavaScript que pueda ser visto en la web.

Actitudes: Curiosidad, disposición, constancia, persistencia, apertura a la incorporación de nuevos aprendizajes, capacidad de análisis, apertura al diálogo, escucha, trabajo en equipo, intercambio de opiniones, participación activa, interés y apertura por incorporar en las actividades la Perspectiva de Género para el logro de un bien común.

Actividad 0: Git

Aprendizaje esperado: Usar un sistema de versiones Git en línea para almacenar todo el código y evidencias producidas.		Duración de la actividad: 3 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none">1. Cuenta en GitHub.2. Cuenta en GitLab.3. Hojas de papel o fichas de trabajo.4. Lápices, plumas o plumones.	Evidencia: <ol style="list-style-type: none">1. Repositorio en GitHub y GitLab con el nombre “paginaTaller1” que contenga los archivos <i>index.html</i> de la actividad 4 del taller 1, y el archivo <i>Index.css</i> de la actividad 6 del taller 1, ambos de este módulo de Elaboración de Sistemas Web.2. 11 fichas de consulta (mínimo).	Retroalimentación: <ul style="list-style-type: none">• Las tarjetas de consulta deberán responder a cada una de las preguntas de la primera parte de esta actividad.• Deberán tener escrita la pregunta por el anverso, y en el reverso, la respuesta a tal pregunta.• El historial del repositorio “paginaTaller1” debe mostrar 2 commits.• Debe hacerse un commit y push por cada archivo con el que la participante cuente. Es decir, la cantidad de archivos debe ser equivalente a la cantidad de veces que se ejecute un commit y un push sobre el repositorio.• Corroborar que no haya contenido errores al subir los archivos a los repositorios en GitHub y GitLab.• El repositorio en GitHub y GitLab, debe tener la misma cantidad de archivos y commits.
Desarrollo de la actividad:		
Primera parte: Conociendo Git.		
Sugerencia de tiempo invertido: 2 horas.		

Las participantes tendrán un acercamiento mucho más completo al sistema de control de versiones Git.

1. Se solicitará a la participante retomar las tarjetas que redactó en la tercera parte de la actividad 9 del taller anterior (se dará la instrucción de no leerlas hasta que se indique lo contrario).
2. Posteriormente, se le pedirá que realice una búsqueda de información mediante algunos enlaces que se le proporcionen (previamente elegidos, con información concisa). Se recomienda utilizar los enlaces anexos al final de este documento.
3. Tal información será correspondiente a las interrogantes:
 - a. ¿Qué es un SVC (Sistema de Control de Versiones)?
 - b. ¿Qué es git?
 - c. ¿Qué sistemas Git gratis existen?
 - d. ¿Para qué sirve el comando *git*?
 - e. ¿Para qué sirve el comando *git init*?
 - f. ¿Para qué sirve el comando *git add*?
 - g. ¿Para qué sirve el comando *git commit*?
 - h. ¿Para qué sirve el comando *git push*?
 - i. ¿Cuáles son los pasos a seguir para subir un proyecto a un repositorio?
 - j. ¿Por qué git te permite trabajar colaborativamente en un proyecto?
 - k. ¿Qué errores o conflictos pueden surgir al trabajar colaborativamente en un proyecto con git?
4. La respuesta a las preguntas será redactada en una ficha por pregunta.
5. Una vez realizado el paso anterior, se dará la instrucción de leer en voz alta las fichas de la actividad 9 del taller 1, y posteriormente las fichas de esta actividad.
6. Mediante la comparación de sus fichas, la participante deberá realizar un ejercicio de reflexión; este deberá ser guiado a través del planteamiento de las siguientes interrogantes:
 - a. ¿Es similar lo que pensabas que era el sistema Git, y lo que ahora sabes que es?
 - b. ¿Coincide la experiencia que tuviste utilizando GitHub, con la información que encontraste sobre su funcionamiento?
 - c. Retomando tu experiencia, ¿hay algo que no comprendías cuando usaste GitHub por primera vez y que ahora después de tu búsqueda de información se aclaró?
 - d. ¿Consideras que es lo mismo Git y GitHub?
 - e. ¿Te queda alguna duda?

Nota: En caso de que la participante conteste afirmativamente a la pregunta 4, aclarar las diferencias entre git y GitHub, por las cuales no se consideran lo mismo.

Segunda parte: Usando Git.

Sugerencia de tiempo invertido: 1 hora.

1. Se solicitará a la participante crear una cuenta en GitLab. https://gitlab.com/users/sign_up
2. La participante editará su perfil, agregará una fotografía y una descripción de su personalidad, por ejemplo “Participante en proyectos culturales que benefician mi entorno”, “Aprendiz en la Escuela de Código, PILARES”.
3. Se le indicará crear un repositorio local con el nombre “paginaTaller1”, en el cual ubicará los archivos *index.html* e *Index.css* generados en las actividades 4 y 6 respectivamente, del taller 1 de este módulo.
4. Posteriormente, y a través de la secuencia de comandos investigados en la primera parte de esta actividad, deberá subir dichos archivos a un repositorio remoto en su nueva cuenta de GitLab, todo esto a través de la consola de git. Para reforzar los conocimientos adquiridos acerca de git, la participante deberá subir cada archivo de manera individual, uno por cada commit y push.
5. Una vez ejecutado el push, la participante visualizará en su cuenta de GitLab que el repositorio se subió correctamente.
6. Cuando los pasos anteriores se hayan realizado sin complicaciones, se solicitará replicarlos para subir el proyecto a la plataforma GitHub.

Notas para apoyar la actividad:

- Se recomienda solicitarle a la participante buscar más comandos básicos de Git, y que los ponga en práctica.

Links de apoyo (parte 1):

- JointDeveloper. (2017). *Sistemas de control de versiones, qué son y por qué amarlos*. <https://medium.com/@jointdeveloper/sistemas-de-control-de-versiones-qu%C3%A9-son-y-por-qu%C3%A9-amarlos-24b6957e716e> (Abril 2020)
- Atlassian Bitbucket. (Sin fecha). *Software de control de versiones para equipos profesionales*. <https://bitbucket.org/product/es/version-control-software> (Abril 2020)
- Platizi. (2019). *¿Qué es Git y GitHub?-Repositorios, ramas y mucho más*. [video]. <https://www.youtube.com/watch?v=DinilgacaWs> (Abril 2020)
- Códigofacilito. (2015). *Qué Es Git*. <https://codigofacilito.cm/articulos/que-es-git> (Abril 2020)
- EDteam. (2019). *¿Qué es git y cómo funciona?* [video] <https://www.youtube.com/watch?v=jGehuhFhtnE> (Abril 2020)

- Rubio, J. (2019). *Qué es GIT y para qué sirve.*
<https://openwebinars.net/blog/que-es-git-y-para-que-sirve/> (Abril 2020)
- Kjchints. (2020). *Git Explicado Fácilmente.*[video]
<https://www.youtube.com/watch?v=2mxh3tgx71c> (Abril 2020)
- Gustavo, B. (2019). *¿Qué es GitHub y para qué se utiliza?*
<https://www.hoErnestoGarcía.Tips.de/resolución.de/conflictosGit.https://medium.com/get-on-board-dev/tips-de-resoluci%C3%B3n-de-conflictos-git-b6a85b204f9bstinger.mx/tutoriales/que-es-github/> (Abril 2020)
- Escuela Web. (2014). *Introducción a Git y Control de Versiones | Escuela Web.* [video]
<https://www.youtube.com/watch?v=SMWltqi3Y-0> (Abril 2020)
- Xataka. (Sin fecha). *Qué es GitHub y qué es lo que le ofrece a los desarrolladores.*
<https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores> (Abril 2020)

Sistemas Git gratis

- GitHub. (2016). *What is GitHub?* [video]
<https://www.youtube.com/watch?v=w3jLJU7DT5E> (Abril 2020)
- LevelUpTuts. (2018). *What Is GitLab?* [video]
<https://www.youtube.com/watch?v=gbJUasioKil> (Abril 2020)
- Atlassian. (2016). *Bitbucket - the Git solution for professional teams.*[video]
<https://www.youtube.com/watch?v=BD8xfCILcBs> (Abril 2020)
- Cleveroad— Web and Mobile App Development. (2017). *Top 3 Source Code Repository Hosts.*[video]
<https://www.youtube.com/watch?v=AuLmDhY3Kjc> (Abril 2020)

Configuración de un sistema git en línea

- Fazt. (2018). *Git y GitHub | Curso práctico de Git y GitHub desde cero.* [video]
<https://www.youtube.com/watch?v=HiXLkL42tMU> (Abril 2020)
- Laboratoria. (2017). *4. Cómo configurar Git.* [video]
<https://www.youtube.com/watch?v=1RJHiHzUnk8> (Abril 2020)
- Git. (Sin fecha). *1.6. Inicio- Sobre el Control de Versiones - Configurando Git por primera vez.*
<https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Configurando-Git-por-primera-vez> (Abril 2020)

¿Cómo crear un repositorio en GitHub?

- Laboratorio. (2017). *2. Cómo crear un repositorio en GitHub*. [video]
https://www.youtube.com/watch?v=Qy_rNobHRY0 (Abril 2020)
- Conociendo GitHub. (Sin fecha). *Aprendiendo a usar GitHub*.
<https://conociendogithub.readthedocs.io/en/latest/data/dinamica-de-uso/#colaborar-en-un-proyecto-ajeno> (Abril 2020)

Comandos para subir un repositorio

- Lago, E. (2017). *CÓMO SUBIR EL CÓDIGO DE TU PROYECTO A GITHUB*.
<https://www.ecodeup.com/como-subir-el-codigo-de-tu-proyecto-a-github/> (Abril 2020)
 - (2018). *Cómo crear un repositorio local con Git y subirlo a GitHub*. [video].
<https://www.youtube.com/watch?v=l2MdKm5RN3o> (Abril 2020)
- MitoCode. (2018). *Curso de Git y Github - 3 Primer Repositorio y Commit*. [video].
<https://www.youtube.com/watch?v=sZySm2Xh0vU> (Abril 2020)
- Saldaña, N. (2018). *Cómo Crear Un Repositorio En GitHub Desde Terminal*. [video]
<https://www.youtube.com/watch?v=tkQp1BvnzsY> (Abril 2020)
- Dudler, R. (Sin fecha). *Guit - La guía sencilla*.
<https://rogerdudler.github.io/git-guide/index.es.html> (Abril 2020)

Trabajo colaborativo

- Git. (Sin fecha). *5.1 Git en entornos distribuidos -Flujos de trabajo distribuidos*.
<https://git-scm.com/book/es/v2/Git-en-entornos-distribuidos-Flujos-de-trabajo-distribuidos> (Abril 2020)
- Pérez, A. (2010). *Git y cómo trabajar con un repositorio de código distribuido*.
<https://www.adictosaltrabajo.com/2010/07/12/git/> (Abril 2020)
- Atlassian Bitbucket. (Sin fecha). *Comparar Workflows*.
<https://www.atlassian.com/es/git/tutorials/comparing-workflows> (Abril 2020)

Conflictos en git

- García, E. (2019). *Tips de resolución de conflictos Git*.
<https://medium.com/get-on-board-dev/tips-de-resoluci%C3%B3n-de-conflictos-git-b6a85b204f9b> (Abril 2020)

Links para aprender más

- Códigofacilito. (Sin fecha). Curso de Git Gratis.
<https://codigofacilito.com/cursos/git> (Abril 2020)

Actividad 1: JavaScript con tenedor

Aprendizaje esperado: Distinguir el uso de JavaScript dentro de los sistemas web, en particular en las páginas web.		Duración de la actividad: 4 hrs 30 minutos
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Hojas de papel o fichas de trabajo. 2. Lápices, plumas o plumones. 	<p>Evidencia:</p> <ol style="list-style-type: none"> 1. 11 fichas de consulta (mínimo). <p>Productos:</p> <ol style="list-style-type: none"> 2. Una página web con estilo e interacciones básicas de JS en un nuevo repositorio de Git. 3. Una nueva estructura de carpetas (HTML, CSS, JS) dentro de la carpeta MWT2A1_[NombreDeLaPágina], que a su vez estará contenida en su carpeta de evidencias de este taller. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • Las tarjetas de consulta deberán responder a cada una de las preguntas de la segunda parte de esta actividad. • Deberán tener escrita la pregunta por el anverso, y en el reverso, la respuesta correspondiente. <p>Evaluación:</p> <p>La página web deberá estar compuesta por:</p> <ol style="list-style-type: none"> 1. Un archivo de extensión <i>.html</i> con las etiquetas correspondientes a título, subtítulo, encabezado, pie de página, imágenes, ligas, párrafos y menú. 2. Un archivo de extensión <i>.css</i> con por lo menos una de cada una de las propiedades siguientes: <i>font-family</i>, <i>font-size</i>, <i>color</i>, <i>text-align</i>, <i>width</i>, <i>height</i>, <i>border-style</i>, <i>border-width</i>, <i>border-color</i>, <i>margin</i>, <i>padding</i> y <i>background-image</i>. 3. Un archivo de extensión <i>.js</i> con las funciones <i>alert</i>, <i>confirm</i> y <i>prompt</i>. <p>Al visualizarse de manera local, la página</p>

		<p>web deberá:</p> <ol style="list-style-type: none"> 1. Mostrar un mensaje de alerta con la leyenda <i>¡Bienvenidas mujeres emprendedoras!</i> 2. Posteriormente un mensaje de confirmación con la leyenda <i>¿Deseas saber acerca de autonomía económica de las mujeres?</i> 3. Finalmente, un cuadro de diálogo solicitando el nombre de la usuaria, para inmediatamente después mostrar el mensaje personalizado <i>Hola [nombre] te invitamos a ser emprendedora y a desarrollar tu autonomía económica.</i>
--	--	--

Desarrollo de la actividad:

Primera parte: ¿Para qué JavaScript?

Sugerencia de tiempo invertido: 1 hora.

1. Se proporcionará a la participante un ejemplo de página web estática y uno de página web dinámica desarrollada con JavaScript, ambas acerca de un mismo tópico (deberá procurarse que la diferencia visual entre ellas sea muy notoria). Es sumamente importante evitar señalar a la participante cuál de ellas es estática y cuál dinámica. Se sugiere utilizar los links anexos al final del documento para esta primera parte de la actividad.
2. La participante realizará una comparación profunda entre ambas páginas. Con la intención de facilitar este ejercicio, se le podrán sugerir algunas preguntas guías, como por ejemplo:
 - ¿Qué es lo primero que llama tu atención en la primera página?
 - ¿Qué es lo primero que llama tu atención en la segunda página?
 - ¿Qué elementos se repiten en ambas páginas?
 - ¿Qué elementos contiene únicamente la primera página?
 - ¿Qué elementos contiene únicamente la segunda página?
 - ¿Las características de los elementos (subtítulos, enlaces, imágenes y otros), cambian de una página a otra? ¿En

qué forma?

3. También se solicitará que comparta las conclusiones a las que llegó, para discutir las en conjunto, y guiar su reflexión de tal forma que pueda identificar por su propia cuenta, cuál página es la estática, cuál la dinámica y cuáles son las características particulares de ambas.
4. Una vez que la participante haya identificado sin dificultad en qué consisten las páginas estáticas y las dinámicas, se le hará saber que la diferencia de la estática, la dinámica fue desarrollada con JavaScript. Posteriormente, se le solicitará responder lo siguiente en un sólo lado de una ficha (siguiendo el mismo formato que ha trabajado a lo largo de este módulo):
 - Tomando como base las características de la página dinámica, ¿qué crees que es JavaScript, y para qué crees que funciona?

5. Por último, se recomienda facilitarle a la participante los links anexos al final de este documento en la sección *Links de apoyo (primera parte)*, para que pueda visualizar más páginas dinámicas creadas con JS y que entienda el alcance de este lenguaje en una página web. De mostrarle dichos links, será imprescindible proporcionarle la descripción adjunta a cada sitio, con la intención de que conozca algunos proyectos dedicados al emprendimiento femenino.

En caso de no elegir los links anexos, será necesario asegurarse de que aquellos que se decidan proporcionar, pertenezcan al mismo eje temático y que incluyan una descripción como la mostrada en los ejemplos.

Segunda parte: ¡Ah, JavaScript!

Sugerencia de tiempo invertido: 2 horas.

Se le proporcionará a la participante un listado de páginas web, cuyo contenido sea claro, preciso y de fácil comprensión (se sugiere utilizar el listado anexo), de donde pueda recabar información para elaborar fichas de consulta sobre las siguientes interrogantes:

1. ¿Qué es un lenguaje de programación?
2. ¿Qué es JavaScript y para qué sirve?
3. ¿Qué es la *interactividad* en las páginas web y por qué es importante?
4. ¿Con qué lenguajes se puede crear una página interactiva?
5. ¿Qué es el DOM?
6. ¿Qué son los *scripts*?
7. ¿Cómo introducir JavaScript en mi mismo archivo HTML?
8. ¿Cómo introducir JavaScript ligando un archivo externo a mi HTML?
9. ¿Cómo mostrar mensajes de *alerta* en mi página web?

10. ¿Cómo mostrar mensajes de *confirmación* en mi página web?

11. ¿Cómo solicitarle a las personas visitantes de mi página web que ingresen alguna información?

12. ¿Qué son las funciones en JavaScript?

Se supervisará que dichas fichas tengan de un lado cada una de las preguntas anteriores, y del otro, la respuesta correspondiente de acuerdo a la información encontrada. Una vez que la participante cuente con todas sus fichas, deberá compartir sus respuestas, de tal forma que mediante un ejercicio de discusión, se concuerde si la información está completa o si requiere modificaciones.

Tercera parte: ¡Vamos a emprender!

Sugerencia de tiempo invertido: 1 hora y 30 minutos.

1. Se solicitará a la participante crear una nueva página web:

Se sugiere que el tema y contenido siga la línea temática de las actividades de este taller, se recomienda el tópico *Autonomía económica de las mujeres*.

2. Deberá supervisarse que la participante:

- a. Realice una nueva estructura de carpetas para los archivos a desarrollar de su página (HTML, CSS Y JS).
- b. Cree un sólo código HTML con las etiquetas correspondientes a los componentes básicos de una página (título, subtítulo, encabezado, pie de página, imágenes, ligas, párrafos y menú).
- c. Dé estilo a su página a través de ligar su HTML con un archivo CSS, el cual contendrá por lo menos una propiedad de cada uno de los siguientes tipos: font-family, font-size, color, text-align, width, height, border-style, border-width, border-color, margin, padding, y background-image.

3. A continuación, deberá incorporar las interacciones que conoció en la realización de sus fichas de consulta. Para ello, se le presentará una serie de elementos que deberá contener su página web. Se recomienda utilizar los siguientes: un mensaje de alerta (utilizando la función *alert*) que aparezca antes de visualizar su página, con la leyenda *¡Bienvenidas mujeres emprendedoras!* junto con un botón de *aceptar*; un mensaje de confirmación (utilizando la función *confirm*) que aparezca inmediatamente después del mensaje de alerta, con la leyenda *¿Deseas saber acerca de autonomía económica de las mujeres?* y dos botones, uno con la opción *aceptar* y el otro con la opción *cancelar*; y por último, un cuadro de diálogo (utilizando la función *prompt*) posterior al mensaje de confirmación, donde se le solicite su nombre a las visitantes de su página web, para posteriormente mostrar el mensaje personalizado *Hola [nombre] te invitamos a ser emprendedora y a desarrollar tu autonomía económica*.

- a. Las interacciones anteriores deberán estar ubicadas en un nuevo archivo de extensión *.js* guardado en la carpeta correspondiente dentro de la nueva estructura de carpetas. La participante deberá ligar este

archivo al HTML.

- b. Al final de la actividad, su página tendrá que visualizarse sin ningún error en algún navegador de forma local.
- c. Una vez que se determine que su página ya no cuenta con ningún error, se le solicitará a la participante que la suba a un nuevo repositorio de *Git*.

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- Se sugiere que las conclusiones de la primera parte de la actividad (el ejercicio de comparación), las escriba la participante en un archivo de texto, y posteriormente las comparta para discutir las.
- Se recomienda que la participante redacte las características de las páginas estáticas y las páginas dinámicas en una ficha de consulta aparte, para futuras referencias.
- **Es imprescindible que todos los enlaces o sitios web que se decida compartir con las participantes, cuenten con una redacción clara y concisa, sean de fuentes confiables, y que el contenido sea libre de violencia de género (o violencia en general), del fomento de estereotipos y del consumo de sustancias tóxicas o dañinas para la salud.**
- Se supervisará que las fichas de consulta acerca de los mensajes de alerta, confirmaciones y solicitud de texto al usuario, correspondan a las funciones de JavaScript: *prompt*, *window.alert* o *alert*, *confirm*, o similares, siempre y cuando no requieran generar de bases de datos, o tengan un nivel técnico más avanzado.
- La participante deberá crear en la nube una nueva carpeta de evidencias para este taller 2.
- Dentro de esta, deberá crearse una carpeta “MWT2A1_[NombreDeLaPágina]” con el nombre que llevará la página web de la participante. Las carpetas de HTML, CSS y JS de la parte 3 de la actividad, deberán guardarse dentro de esta última carpeta.
- Deberá asegurarse que el archivo HTML de la participante, lleve por nombre *index.html*, así como el archivo CSS, el de *style.css*. Lo anterior con la intención de fomentar la común nomenclatura de estos archivos en el desarrollo web.

Notas para mejorar habilidades de participantes:

- Tomar algún curso en línea gratuito de programación javascript en paralelo a este curso.

Links de apoyo (primera parte):

- Ana Victoria García es la fundadora de Victoria147, plataforma creada en 2012 para fomentar el empoderamiento de las

mujeres por medio de capacitaciones que promuevan la participación de las mujeres en el emprendimiento y la equidad de género.

https://victoria147.org/?utm_source=google&utm_medium=cpc&utm_campaign=victoria147-branding&utm_content=20200213-adaptable&gclid=Cj0KCQjw3qzzBRDnARIsAECmryp6Xj2WDGe5ygyjD7BT-TkNaWtYTxtGxrsyfHxwZhOBMqEk-qgKlcaAjwQEALw_wcB (Marzo 2020).

- Leticia Jáuregui, fundadora de la asociación civil *Comunidades de Emprendedoras sociales* en 2008. Su proyecto pretende elevar la participación de las mujeres en el mundo económico, venciendo las barreras culturales, tecnológicas, económicas, políticas y sociales que aún en la actualidad las limitan.
<http://crea.org.mx/> (Marzo 2020).
- Linda Franco es una mexicana que forma parte de la lista de Innovadores de Latinoamérica menores de 35 años, publicada por el *MIT Technology Review*. Es cofundadora del startup *Machina*, empresa que se dedica a la creación de ropa inteligente. Uno de sus productos más famosos es la *MIDI Jacket* que permite crear música a través de los sensores que posee la chamarra. Otro de sus productos, contiene un botón de pánico que envía una alerta con su ubicación a 3 contactos predefinidos por la usuaria.
<https://www.machina.cc/> (Marzo 2020).
- “Alba [medina] es una emprendedora en serie e inversionista en start-ups globales. Es Co-fundadora de Dalia Empower (el primer ecosistema verticalmente integrado para el empoderamiento de las mujeres en América Latina). Es una defensora apasionada de mujeres y niñas, así como miembro activo en más de 15 consejos nacionales e internacionales. Como pasatiempo, disfruta escribir sobre el empoderamiento femenino, participar en conferencias sobre igualdad de género y emprendimiento[...].” (DaliaEmpower)
<https://daliaempower.com/> (Marzo 2020).
- Nathalie Ekleme es una mexicana originaria de Guadalajara, Jalisco. Fundadora de la empresa Pitahia, dedicada a la elaboración de esmaltes naturales y orgánicos. El nombre de su empresa es en honor a la fruta Pitaya originaria de México. Ha tenido gran éxito y su producto se envía a cualquier parte de la República Mexicana.
<https://www.pitahia.com/index.php> (Marzo 2020).

Links de apoyo (parte 2):

¿Qué es JavaScript y para qué sirve?:

- Grados, J. (2018). *¿Qué es JavaScript?*
<https://devcode.la/blog/que-es-javascript/> (Marzo 2020).

- Pelhon, L. (2019). *Qué es JavaScript: primeros pasos*.
<https://www.hostgator.mx/blog/que-es-javascript/> (Marzo 2020).
- pildorasinformaticas.(2014). *Curso JavaScript desde 0. Introducción. Vídeo 2*.
https://www.youtube.com/watch?v=QrVO_dvjU-M (Marzo 2020).
- Pérez, D. (2007). *¿Qué es Javascript?*.
<http://www.maestrosdelweb.com/que-es-javascript/> (Marzo 2020).
- MDN Web Docs moz://a. (Sin fecha). *¿Qué es JavaScript?*
https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript (Marzo 2020).

¿Dónde colocar el código en JavaScript?

- Pildorasinformaticas. (2014). *Curso JavaScript desde 0. Sintaxis Básica I. Ubicación del código. Video 3*.
<https://www.youtube.com/watch?v=zSYj95x5WrU> (Marzo 2020).
- B, G. (2020). *Cómo Insertar JavaScript en HTML*.
<https://www.hostinger.mx/tutoriales/insertar-javascript-en-html/#Insertar-JavaScript-en-HTML> (Marzo 2020).

Ejemplos de JavaScript:

- MINISTERIO DE EDUCACIÓN, CULTURA Y DEPORTE. (2007). *Ejemplos prácticos de JavaScript (I)*.
<http://recursostic.educacion.es/observatorio/web/ca/software/programacion/490-lorena-arranz> (Marzo 2020).
- Martín, D. (Sin fecha). *Referencia rápida de JavaScript básico con ejemplos*.
<https://www.diegocmartin.com/primeros-pasos-con-javascript-ejemplos-basicos-en-js/> (Marzo 2020).

Interactividad en las páginas web:

- Córdoba, Estefanía. (Sin fecha). *Interactividad del sitio Web*.
<https://imaginativas.es/seo/interactividad-del-sitio-web/> (Marzo 2020).
- Firtman, M. (2001). *Interactividad en la Web*.
<http://www.maestrosdelweb.com/intera/> (Marzo 2020).

DOM:

- MDN Web Docs moz://a. (Sin fecha). *¿Qué es el DOM?*.
https://developer.mozilla.org/es/docs/Referencia_DOM_de_Gecko/Introducci%C3%B3n#Qu%C3.A9_es_el_DOM.3

[E](#) (Marzo 2020).

¿Qué son los scripts?:

- Noguera, B. (Sin fecha). *¿Qué son los Scripts?*
<https://culturacion.com/que-son-los-scripts/> (Marzo 2020).

Mensajes de alerta:

- Barrena, A. (2017). *Cómo mostrar mensajes de alerta con JavaScript*.
<https://www.anerbarrena.com/mostrar-mensajes-alerta-javascript-5568/> (Marzo 2020).
 - *JavaScript alert: Mostrar mensajes de alerta en JS*.
<https://www.anerbarrena.com/javascript-alert-confirm-prompt-js-5497/> (Marzo 2020).
- Codeando. (2014). *Curso Javascript- Ventana de dialogo alert*. [video]
<https://www.youtube.com/watch?v=UBMbQZ3y35Q> (Marzo 2020).
- w3schools.com. (Sin fecha). *Window alert() Method*.
https://www.w3schools.com/jsref/met_win_alert.asp (Marzo 2020).
- MDN Web Docs moz://a. (Sin fecha). *Window.alert()*.
<https://developer.mozilla.org/es/docs/Web/API/Window/alert> (Marzo 2020).

Mensajes de confirmación:

- Barrena, A. (2017). *JavaScript confirm: Mostrar mensajes de confirmación en JS*.
<https://www.anerbarrena.com/javascript-confirm-js-5508/> (Marzo 2020).
- w3schools.com. (Sin fecha). *Window confirm() Method*.
https://www.w3schools.com/jsref/met_win_confirm.asp (Marzo 2020).

Cuadro de diálogo para solicitar el ingreso de información:

- Barrena, A. (2017) *JavaScript prompt: Mostrar mensajes y cuadros de diálogo en JS*.
<https://www.anerbarrena.com/javascript-prompt-js-5509/> (Marzo 2020).
- w3schools.com. (Sin fecha) *Window prompt() Method*.
https://www.w3schools.com/jsref/met_win_prompt.asp (Marzo 2020).
- Codeando. (2014). *Curso JavaScript -Función prompt*. [video]
<https://www.youtube.com/watch?v=8AneGTSMisc> (Marzo 2020).

Funciones en JavaScript

- Álvarez, M. (2001). *Funciones en Javascript*.
<https://desarrolloweb.com/articulos/583.php> (Marzo 2020).

Curso completo de nivel básico de JS

- Udemy. (Sin fecha). *Introducción a la programación con JavaScript*.
<https://www.udemy.com/course/javascript-introduccion-a-la-programacion/> (Marzo 2020).
 - *Cómo programar para Emprendedores-JavaScript*.
<https://www.udemy.com/course/curso-de-javascript/> (Marzo 2020).

Actividad 2: JavaScript con cuchara y tenedor.

Aprendizaje esperado: Integrar elementos básicos del lenguaje JavaScript, dentro de una página web.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none">1. Hojas de papel o papel reciclado.2. Lápices, plumas o plumones.	<p>Evidencia:</p> <ul style="list-style-type: none">• Cinco fichas de consulta (mínimo).• Tres cuadros sinópticos. <p>Productos:</p> <ul style="list-style-type: none">• Una página web con estilo, la fecha y hora del sistema en un nuevo repositorio de Git.• Una nueva estructura de carpetas (HTML, CSS, JS) dentro de la carpeta <i>TiendaLinea[NombreDeLaPágina]</i>, que a su vez estará contenida en su carpeta de evidencias de este taller.	<p>Retroalimentación:</p> <ul style="list-style-type: none">• Las tarjetas de consulta así como los cuadros sinópticos, deberán responder a cada una de las preguntas de la primera parte de esta actividad.• Las fichas deberán tener escrita la pregunta por el anverso, y en el reverso, la respuesta a tal pregunta.• El primer cuadro sinóptico deberá corresponder a las preguntas 1-6; el segundo de la 8 a la 10; y el tercero, a las preguntas 12 y 13.• El primero, deberá contener los métodos: <i>console.log</i>, <i>console.table</i>, <i>console.error</i>, <i>console.clear</i> y <i>console.time</i>.• El segundo, deberá contener los tipos de variables: <i>string</i>, <i>number</i>, <i>boolean</i>, <i>null</i>, <i>undefined</i> y <i>symbols</i>.• La obtención de la fecha y hora del sistema deberá hacerse a través uso de JavaScript. <p>Evaluación: La página web deberá estar compuesta por:</p>

		<ol style="list-style-type: none"> 1. Un archivo de extensión <i>.html</i> con las etiquetas correspondientes a título, subtítulo, encabezado, pie de página, imágenes, ligas, párrafos y menú. 2. Un archivo de extensión <i>.css</i> con por lo menos 1 de cada una de las propiedades siguientes: <i>font-family, font-size, color, text-align, width, height, border-style, border-width, border-color, margin, padding</i> y <i>background-image</i>. 3. Un archivo de extensión <i>.js</i> que haga uso del Objeto <i>Date</i> para la obtención de la fecha y hora del sistema; y con las operaciones necesarias para tener la hora en formato de 12 horas. <p>Al visualizarse de manera local, la página web deberá mostrar la fecha (dd/mm/año) y hora (hh:mm:ss) del sistema de la computadora en la que se visualice.</p>
--	--	---

Desarrollo de la actividad:

Primera parte: Conociendo lo básico.

Sugerencia de tiempo invertido: 2 horas y 30 minutos.

Se le proporcionará a la participante un listado de páginas web, cuyo contenido sea claro, preciso y de fácil comprensión (se sugiere utilizar el listado anexo), o bien algún libro o manual (se recomienda utilizar alguno de los anexos) de donde pueda recabar información sobre las siguientes interrogantes:

1. ¿Qué es la consola del navegador y para qué sirve?
2. ¿Cómo mostrar un mensaje de texto en la consola del navegador?
3. ¿Cómo mostrar un mensaje de error?

4. ¿Cómo mostrar tablas?
5. ¿Cómo limpiar la consola?
6. ¿Cómo mostrar un temporizador en la consola?
7. ¿Cómo realizar comentarios en código JavaScript?
8. ¿Qué es una variable en JavaScript y cuál es su utilidad?
9. ¿Qué tipos de variables existen?
10. ¿Cuál es el alcance o ámbito de las variables?
11. ¿En qué consiste la nomenclatura CamelCase y en dónde se utiliza?
12. ¿Qué es un tipo de dato y para que se utilizan?
13. ¿Qué tipos de datos existen y para qué es cada uno?
14. ¿Qué tipos de operadores lógicos existen y cuál es la utilidad de cada uno?
15. ¿Cómo se maneja una fecha en javascript?
16. ¿Qué función de javascript muestra la fecha actual del sistema?

Una vez respondidas las preguntas, se le solicitará a la participante elaborar:

- Tres cuadros sinópticos: el primero sobre el tema *la consola del navegador* (preguntas 1, 2, 3, 4, 5 y 6); el segundo acerca de *variables* (preguntas 8, 9 y 10); y el tercero sobre los *tipos de datos* (preguntas 12 y 13). Se supervisará que dichos cuadros sinópticos: mantengan una estructura que vaya de lo general a lo particular; y que estén conformados por conceptos primarios (por ejemplo *variable*), secundarios (por ejemplo *tipos de variables*), terciarios -de ser el caso- (por ejemplo *string*), definiciones, explicaciones y/o funciones de los conceptos.
- Fichas de consulta (con el formato hasta ahora trabajado) correspondientes a las preguntas 7, 11, 14, 15 y 16.

Finalmente, la participante explicará oralmente cada uno de sus cuadros sinópticos, para que posteriormente se realice un ejercicio de discusión donde se determine si la información está completa o si requieren modificaciones.

Segunda parte: Poniendo a prueba.

Sugerencia de tiempo invertido: 1 hora.

Para poner en práctica los conocimientos aprendidos en el ejercicio anterior, se solicitará la elaboración de algunos scripts en JavaScript que lleven a cabo ciertas tareas; estos deberán tener una complejidad de acuerdo al nivel técnico de la participante. Se recomienda utilizar los siguientes:

1. Script que imprima en pantalla un “Hola mundo”; la cadena debe ser declarada como una variable.
2. Script que realice la suma de dos números y muestre el resultado en pantalla.
3. Script que muestre un mensaje únicamente en la consola del navegador.

4. Script que determine si un número es par.

En caso de que la participante tenga dudas acerca de cómo elaborar los programas, se le motivará para que realice una búsqueda en internet sobre los elementos necesarios para encontrar o descubrir las soluciones. Se procurará no intervenir en la resolución de los problemas, así como en la búsqueda de información.

Cada uno de los scripts se almacenarán en una nueva carpeta de nombre “MWT2A2_ScriptsEjercicio”, dentro de su carpeta de evidencias de este taller. Los scripts se guardarán con las siguientes denominaciones: script 1= “HolaMundo.html”; script 2= “Suma.html”; script 3= “MensajeEnConsola.js”; script 4= “NumeroPar.html”

Tercera parte: A emprender.

Sugerencia de tiempo invertido: 1 hora 30 minutos.

Se solicitará a la participante la creación de un nuevo sitio web, proyecto que irá desarrollando a lo largo de las siguientes actividades de este taller.

1. La temática de su página web deberá tener como objetivo la venta de algún producto o servicio que ella o alguna conocida ofrezca; es decir, una tienda en línea.
2. En caso de tratarse de algún producto o servicio que no sea propio, la participante deberá realizar como primer paso la toma de requerimientos que deberá incluir el sitio web. De ser un producto propio, deberá pensar qué elementos querrá que contenga su sitio web.
3. En esta actividad, corresponderá solicitarle a la participante que elabore únicamente la página de inicio (index.html) de dicho proyecto. Para ello, la participante:
 1. Creará una nueva estructura de carpetas para los archivos a desarrollar de su sitio web (HTML, CSS Y JS). Estas carpetas a su vez, deberán estar contenidas dentro de una nueva carpeta general de su proyecto, que deberá ser nombrada con la siguiente estructura “TiendaLinea[NombreDelProyecto]”
 2. Creará un código HTML.
 3. Le dará estilo a su página a través de ligar su HTML con un archivo CSS.
 4. Creará un archivo JS que permita obtener la fecha y la hora del sistema y mostrarla en su archivo HTML. La fecha deberá tener el siguiente formato: “dd/mm/año”; mientras que la hora se presentará en formato de 12 horas con la siguiente estructura: “hh:mm:ss”. Ambos datos deberán cambiar conforme cambie la fecha y hora del sistema (de la computadora), y se les deberá aplicar estilo a través de su archivo CSS.
 5. Al final de la actividad, su página tendrá que visualizarse sin ningún error en algún navegador de forma local.
 6. Una vez que se determine que su página ya no cuenta con ningún error, se le solicitará a la participante que cree un nuevo repositorio con nombre “TiendaEnJavaScript” en GitHub y suba su proyecto.

Notas para apoyar la actividad:

- Si se desconocen las características de un cuadro sinóptico (tallerista y participante), deberán buscar ejemplos en internet o consultar *Pimienta*, Julio (2012).
- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- **Es imprescindible que todas los enlaces o sitios web que se decida compartir con las participantes, cuenten con una redacción clara y concisa, sean de fuentes confiables, y que el contenido sea libre de violencia de género (o violencia en general), del fomento de estereotipos y del consumo de sustancias tóxicas o dañinas para la salud.**
- Se supervisará que la información encontrada acerca de la consola del navegador, correspondan a los métodos: `console.log`, `console.table`, `console.error`, `console.clear` y `console.time`
- Se supervisará que la información encontrada acerca de la consola del navegador corresponda a los métodos: `console.log`, `console.table`, `console.error`, `console.clear` y `console.time`
- La carpeta general de su proyecto (*TiendaLinea[NombreDelProyecto]*), se ubicará dentro de su carpeta de evidencias de este taller.
- Se recomienda mostrarle a la participante cómo hacer uso de bibliotecas para JavaScript.

Notas para mejorar las habilidades de las participantes:

- Se recomienda que las participantes realicen la búsqueda de otras funciones básicas con MongoDB, distintas a las aquí trabajadas.
- Se recomienda solicitar a la participante que busque la diferencia de una base de datos SQL y una NoSQL.

Links de apoyo (parte 1):

Consola del navegador

- Transferwise.com. (Sin fecha). *¿Cómo abrir la consola de tu navegador?*
<https://transferwise.com/es/help/a/2954851/como-abrir-la-consola-de-tu-navegador> (Marzo 2020)
- CÓDIGO FUENTE. (2018). *Editores y Consola de desarrollo para JavaScript*.
https://www.codigofuente.org/editores-consola-desarrollo-javascript/?vlogger_serie_in=1134 (Marzo 2020)
- Norfipc.com. (Sin fecha). *Como usar la consola de JavaScript del navegador web, ejemplos*.
<https://norfipc.com/inf/como-usar-consola-javascript-navegador-web.php> (Marzo 2020)
- Araque, M. (Sin fecha). *usando console.time*.

<https://marioaraque.com/console-time-ejemplo-uso> (Marzo 2020)

- w3schools.com. (Sin fecha). *HTML DOM console.time() Method*.
https://www.w3schools.com/jsref/met_console_time.asp (Marzo 2020)

Comentarios en código JavaScript

- Padial, J. (2016). *Cómo poner comentarios en javascript*.
<https://cybmeta.com/como-poner-comentarios-en-javascript> (Marzo 2020)
- w3schools.com. (Sin fecha). *JavaScript Comments*.
https://www.w3schools.com/js/js_comments.asp (Marzo 2020)

Tipos de variables en JS

- Alvarez, M. (2020). *Variables en Javascript*.
<https://desarrolloweb.com/articulos/508.php> (Marzo 2020)
- JAVASCRIPT.INFO. (2020). *Variables*
<https://javascript.info/variables> (Marzo 2020)
- Openclassrooms.com. *Variables*.
<https://openclassrooms.com/en/courses/4309431-aprende-a-programar-con-javascript/4318441-variables> (Marzo 2020)
- FalconMasters. (2017) *Curso Básico de Javascript 2.- Variables*. [video]
<https://www.youtube.com/watch?v=sXzoPsl7FH0> (Marzo 2020)
- Uniwebstad. (Sin fecha) *Introducción a Java Script. 3.2. Tipos de variables*.
<https://uniwebstad.com/libros/javascript/capitulo-3/tipos-de-variables> (Marzo 2020)
- CÓDIGO FUENTE. (2018). *Variables y Constantes en JavaScript*.
https://www.codigofuente.org/variables-y-constantas-en-javascript/?vlogger_serie_in=1134 (Marzo 2020)
- Codexemplar.org. (Sin fecha). *Variables. < Sintaxis básica de JavaScript*.
http://www.codexemplar.org/curso/curso_4_2_a.php (Marzo 2020)
- Alvarez, M. (2020). *Ámbito de las variables en Javascript*.
<https://desarrolloweb.com/articulos/517.php> (Marzo 2020)
- Miranda, D. (2016) *let y const: variables en ECMAScript 2015 (ES6)*.
<https://desarrolloweb.com/articulos/conociendo-variables-ecmascript.html> (Marzo 2020)
- Francesc Ricart. (Sin fecha). *TIPOS DE VARIABLES EN JAVASCRIPT*.

<https://francescricart.com/tipos-variables-javascript/> (Marzo 2020)

- Nancy Developer Web. (Sin fecha). *Unidad 2. VARIABLES Y TIPOS DE DATOS EN JAVASCRIPT*.
<https://nancydeveloperweb.wordpress.com/unidad-2-variables-y-tipos-de-datos-en-javascript/> (Marzo 2020)

Nomenclatura CamelCase

- Neumoytoraxpanama.org. (Sin fecha). *CamelCase*.
<https://www.neumoytoraxpanama.org/cloud/resources/documentos/camelcase.pdf> (Marzo 2020)
- Alonso, A. (2017). *Convención de nombres: desde el CamelCase hasta el kebab-case*.
<https://medium.com/@alonsus91/convenci%C3%B3n-de-nombres-desde-el-camelcase-hasta-el-kebab-case-787e56d6d023> (Marzo 2020)
- w3schools.com. (Sin fecha). *JavaScript Style Guide*.
https://www.w3schools.com/js/js_conventions.asp (Marzo 2020)

Tipos de datos

- MDN Web Docs moz://a. (Sin fecha). *Tipos de datos y estructuras en JavaScript*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Data_structures (Marzo 2020)
- todoJS.com. (2016). *Tipos de datos en Javascript (ES6)*.
<https://www.todojs.com/tipos-datos-javascript-es6/> (Marzo 2020)

Operadores en JS

- Escuela Javascript. (Sin fecha). *¿Qué operadores puedo usar en JavaScript?*
<https://www.escuelajavascript.com/que-operadores-puedo-usar-en-javascript/> (Marzo 2020)
- Uniwersidad. (Sin fecha). *Introducción a JavaScript. 3.3. Operadores*.
<https://uniwersidad.com/libros/javascript/capitulo-3/operadores> (Marzo 2020)
- CÓDIGO FUENTE. (2018) *Operadores en Javascript*.
<https://www.codigofuente.org/operadores-en-javascript/> (Marzo 2020)

Manejo de fechas en JS

- FalconMasters. (2015). *Curso Básico de JavaScript 24.- Trabajando con Fechas y Tiempo (Objeto Date)*. [video]
<https://www.youtube.com/watch?v=JWCDkQnRwd8> (Marzo 2020)
- *Tutorial Como hacer un Reloj con HTML, CSS y Javascript*. [video].

- <https://www.youtube.com/watch?v=lowfM4bkyb8> (Marzo 2020)
- Ortiz, John. (2019). Javascript - Ejercicio 134: Obtener la Fecha y Hora de un Objeto Date. [video]
<https://www.youtube.com/watch?v=8G5V9qKawZ0> (Marzo 2020)
- w3schools.com. (Sin fecha). JavaScript Date Reference.
https://www.w3schools.com/jsref/jsref_obj_date.asp (Marzo 2020)

Links para aprender más:

- Khan Academy. (Sin fecha). *Usar la Hora de Código en tu salón de clases.*

<https://es.khanacademy.org/computing/hour-of-code/hour-of-code-for-teachers/a/using-hour-of-code-in-your-classroom>

(Marzo 2020)

1. Cree, una nueva estructura de carpetas para los archivos a desarrollar de su página (HTML, CSS Y JS).
2. Cree al menos dos códigos HTML con las etiquetas correspondientes a los componentes básicos de una página, vistos en el taller anterior. Los dos archivos deberán ser linkeados uno con el otro, para utilizarse como enlace entre ellas.
3. Dé estilo a su página a través de ligar su HTML con un archivo CSS, aplicando lo aprendido en el taller 1.

Referencias:

- Pimienta, J. (2012) *Estrategias de enseñanza-aprendizaje. Docencia universitaria basada en competencias.* México. Pearson.

Actividad 3: JavaScript con cuchara, tenedor y cuchillo.

Aprendizaje esperado: Implementar operaciones matemáticas básicas usando JavaScript, e implementar operaciones básicas con cadenas de caracteres.		Duración de la actividad: 8 horas
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Plantillas del anexo disponibles en <i>MWT2A3_Anexos.pdf</i>	Evidencia <ul style="list-style-type: none">● Resolver las plantillas del juego de relación de columnas. Producto <ul style="list-style-type: none">● Desarrollar un código JavaScript el cual solicite el ingreso del precio de un artículo y la cantidad de artículos que lleva un cliente y muestre la cantidad total a pagar.	Retroalimentación <ul style="list-style-type: none">● Se deberán de explorar la mayoría de las funciones de operaciones con números, cadenas y conversión de tipos e imprimir el resultado en la consola.● Se deberán reconocer las funciones de un código en un caso concreto. Evaluación <ul style="list-style-type: none">● Se deberá crear un código que pida el ingreso del precio de un artículo y la cantidad de artículos que lleva el cliente para mostrar el total a pagar.
Desarrollo de la actividad:		
<p>Primera parte: Investigando los tipos de datos. Sugerencia de tiempo invertido: 4 horas. Se le proporcionará a la participante el link para acceder a páginas web y material audiovisual cuyo contenido sea claro, preciso y de fácil comprensión (se sugiere utilizar el listado de referencias), con el cual pueda recopilar la información necesaria para comprender los siguientes conocimientos:</p> <ul style="list-style-type: none">● Números<ul style="list-style-type: none">● Operaciones con números (clase Math):<ul style="list-style-type: none">● Adición● Suma● Resta● Multiplicación		

- Asignación
- Comparación
- Cadenas:
 - Concatenación
 - Agregar al final
 - Caracteres especiales.
 - Clase String.
- Conversión de Tipos:
 - String
 - Number

Durante el proceso de búsqueda y recopilación de información, se podrán realizar las precisiones necesarias del contenido según lo requiera la participante.

Una vez que la participante concluya y se le hayan aclarado todas las dudas; se le proporcionarán dos plantillas A y B (disponibles en el anexo *MWT2A3_Anexos.pdf*) con las cuales se llevará a cabo un juego; en este se deben relacionar la columna A (screens de distintos códigos) con los de la comuna B (Indicaciones de que regresaría el código al compilar) para que la participante reafirme el conocimiento y pueda tener ejemplos concisos para construir su propio código.

Segunda parte: Calculando el total

Sugerencia de tiempo invertido: 4 horas.

Se le proporcionará al participante un pseudo código (disponible en el la plantilla C del anexo *MWT2A3_Anexos.pdf*). Con base en este se tendrá que desarrollar en la consola de la computadora un código de javascript que pida:

- El ingreso del precio de un artículo y la cantidad de artículos que lleva el cliente.
- Se debe ingresar por teclado:
 1. El número de artículos (deben de ser números enteros).
 2. El precio de cada artículo .
- En la consola se debe mostrar lo que debe de pagar el cliente.

Links de apoyo:

Para el manejo de cadenas:

- Make it Real. (2020). *JavaScript Básico - Cadenas de texto* [vídeo].
<https://www.youtube.com/watch?v=GBsCjdi22k> (Marzo 2020)
- Pildorasinformaticas. (2014).. *Curso Java. Manipulación de cadenas. Clase String I. Vídeo 11.*
<https://www.youtube.com/watch?v=YMyJLs22uY4> (Marzo 2020)

Números:

- Ventura, J. (2018). *Operaciones fundamentales con JavaScript* [vídeo].
https://www.youtube.com/watch?v=73do-mwIT_Y (Marzo 2020)
- Uniwebsidad. (Sin fecha). *Introducción a JavaScript. 4.1. Funciones.*
<https://uniwebsidad.com/libros/javascript/capitulo-4/funciones> (Marzo 2020)

Conversión de tipos:

- MDN Web Docs moz://a. (Sin fecha).. *Manejar texto-cadena en JavaScript; Handling text — strings in JavaScript.*
https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Strings (Marzo 2020)

Pseudo código

- Facultad de Ingeniería UNAM. (2018). *Guía práctica de estudio 05: Pseudocódigo.*
http://lcp02.fi-b.unam.mx/static/docs/PRACTICAS_FP/fp_p5.pdf (Marzo 2020)

Actividad 4: Ingresando datos específicos

Aprendizaje esperado: Crear formularios en páginas web y obtener los datos obtenidos a partir del llenado de los mismos.		Duración de la actividad: 6 horas 30 minutos.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Plantilla del anexo disponibles en <i>MWT2A4_Anexos.pdf</i>	Evidencia: <ul style="list-style-type: none"> ● <i>Printscreen</i> de las páginas web con los formularios identificados. ● Resolver la plantilla del <i>Challenge</i> en el que se identifican las características de los tipos de validación. Producto: <ul style="list-style-type: none"> ● Agregar a una página web un código JavaScript embebido que contenga un formulario de ingreso de datos para la creación de un catálogo de productos, con los siguientes campos: Nombre del producto, departamento al que pertenece y precio. Los datos deberán de validarse y si pasan la validación deberán de ser mostrados. 	Retroalimentación: <ul style="list-style-type: none"> ● Se deberá poder identificar un formulario en una página web. ● Comprenderá lo que son y su utilidad en una página web y para qué sirven las etiquetas <form>, <input>, <label>, <textarea>, <button>. ● Se deberán identificar las características de los tipos de validación HTML y API. Evaluación: <ul style="list-style-type: none"> ● Se deberán de entregar tres archivos, el .HTML, el .css y el .js cada uno con el código respectivo HTML, de estilo y el JavaScript. ● Las validaciones se deberán de realizar usando las funciones del API de JavaScript para tal propósito.
Desarrollo de la actividad:		
Primera parte: Introducción a los formularios. Sugerencia de tiempo invertido: 3 horas. <ol style="list-style-type: none"> 1. Se solicitará a la participante realizar una búsqueda que responda a dos preguntas ¿qué es un formulario? ¿para qué sirven los formularios en una página web?. 2. Posteriormente se le pedirá que ingrese a tres páginas de compra en línea. En estas tendrá que identificar dónde se ubican los formularios, tomar una impresión (<i>Printscreen</i>) de pantalla y señalar dónde los identifica. Para esto puede 		

hacer uso de la herramienta Kolourpaint.

3. Se le proporcionará a la participante material en el cual puedan buscar las etiquetas **<form>**, **<input>**, **<label>**, **<textarea>**, **<button>** (se sugiere utilizar el listado de referencias).
4. Puesto que las etiquetas **<form>**, **<input>** contienen una variedad inmensa de atributos, se les solicitará realizar una lista de los atributos que les gustaría implementar en su página web respecto de formularios y botones.
5. En este listado deberán anotar la sintaxis de cómo se construye cada uno de ellos, para su posterior consulta.

Segunda parte: Validación HTML y API

Sugerencia de tiempo invertido: 30 minutos.

1. Se le proporcionará a la participante material en el cual pueda buscar los tipos de validación.
2. HTML y API y sus características (se sugiere utilizar el listado de referencias).
3. Una vez concluida la búsqueda se le proporcionará una plantilla (disponibles en el anexo *MWT2A4_Anexos.pdf*), la cual contiene un *Challenge* en el cual tendrán que identificar por medio de una X las características de cada tipo de validación.

Tercera parte: Creando mi primer formulario en HTML.

Sugerencia de tiempo invertido: 3 horas.

Se le solicitará a la participante crear un formulario en HTML en el proyecto de página web que se ha desarrollado a lo largo del taller, el cual contenga un campo de texto en el cual el usuario pueda ingresar los datos de los productos de la tienda online y un botón el cual enviará los datos a un servidor web.

1. Se utilizarán los siguientes elementos HTML: **<form>**, **<label>**, **<input>**, **<text>**, **<button>**.
2. La participante deberá declarar la etiqueta **<form>** que incluirá toda la información que se requiere recaudar de este formulario.
3. Se deberá indicar la dirección donde se enviarán los datos cuando el usuario pulse el botón enviar, en este caso se trabajará a nivel local. (En caso de omitirse esta información, los datos se enviarán a la propia página actual, volviéndola a cargar.)
4. Se utilizará la etiqueta **<input>** para obtener información a modo de campo de entrada de datos en el formulario. Se debe solicitar al usuario el nombre del producto con la etiqueta **<input type="text">** y repetirlo para los campos de 'departamento' y 'precio'.
5. Posteriormente, se colocará un botón con la etiqueta **<button>** para enviar el formulario con la etiqueta **<input type="submit">**.
6. Una vez creado el formulario se obtendrá uno desordenado, por lo cual se añadirá CSS para que su visualización se encuentre organizada.

NOTA: Se debe de crear una carpeta específica para cada parte del entregable, una para HTML, otra para CSS y otra para JAVASCRIPT.

Links de apoyo:

- Lenguajehtml.com. (Sin fecha). *Creación de un formularios.*
<https://lenguajehtml.com/p/html/formularios/crear-un-formulario> (Marzo 2020)
- Mclibre.org. (Sin fecha). *Formularios.*
<https://www.mclibre.org/consultar/htmlcss/html/html-formularios.html#formulario-form> (Marzo 2020)
- MDN Web Docs moz://a. (Sin fecha). *input.*
<https://developer.mozilla.org/es/docs/Web/HTML/Elemento/input> (Marzo 2020)
 - *Validación de formularios de datos.*
https://developer.mozilla.org/es/docs/Learn/HTML/Forms/Validacion_formulario_datos (Marzo 2020)
 - *Validación de restricciones.*
https://developer.mozilla.org/es/docs/HTML/HTML5/Validacion_de_restricciones#Restricciones_complejas_usando_la_API_de_Restricciones_de_HTML5 (Marzo 2020)
 - *Introducción a las APIs web.*
https://developer.mozilla.org/es/docs/Learn/JavaScript/Client-side_web_APIs/Introducción (Marzo 2020)
- htmlquick.com. (Sin fecha). *Tutoriales HTML.*
<https://www.htmlquick.com/es/tutorials/forms/2.html> (Marzo 2020)

Actividad 5: Almacenando datos

Aprendizaje esperado: Almacenar datos de forma temporal y manipularlos en una página web usando JavaScript.		Duración de la actividad: 7 horas
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> Proyecto TiendaLinea[NombreDelProyecto] trabajado en las actividades anteriores. Cualquier medio a elegir para realizar las fichas de trabajo. 	<p>Evidencia:</p> <ul style="list-style-type: none"> Carpeta “MWT2A5_Arreglos” que contendrá dos archivos, “arreglos.js”, el cual contendrá los scripts correspondientes a los ejercicios de arreglos de la primer parte de la actividad, y “productos.html” el archivo HTML al cual estará ligado. Carpeta “MWT2A5_Iteradores” que contendrá dos archivos “ciclos.js” el cual contendrá los scripts correspondientes a los ejercicios realizados en la segunda parte de la actividad, e “iteradores.html” el HTML al cual estará ligado. Carpeta “MWT2A5_FichasTrabajo” que contendrá las fichas de trabajo digitalizadas de la búsqueda realizada sobre arreglos e iteradores. <p>Producto:</p> <ul style="list-style-type: none"> Crear una página web nombrada “Pedido.html” para el código html, y por su parte, otra “Pedido.js” correspondiente al código 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> Cada carpeta deberá contener los archivos .html y deberán encontrarse ligados respectivamente con los archivos js. Las fichas de trabajo deberán contener título del tema, subtítulo, información sobre el tema del cual se hizo la búsqueda (Arreglos e Iteradores) y fuente de consulta con por lo menos autor y/o dominio de la página de la cual se hizo la búsqueda, título del artículo o documento de consulta y fecha en la que se consultó. <p>Evaluación:</p> <ul style="list-style-type: none"> Para la página que simula la caja registradora, los productos se deberán de ir guardando en un arreglo, uno para productos y otro para precios. Deberá de preguntar por el siguiente producto y su precio repetidamente hasta que se terminen de agregar productos, cada producto se deberá agregar y se deberán actualizar las cantidades, cantidad y coste total. Los cálculos se deberán de realizar

	JavaScript embebido que simule una caja registradora.	usando las funciones propias de los arreglos. <ul style="list-style-type: none"> ● Se deberá asegurar que los iteradores se usen para realizar el cálculo en el total de productos y precio final.
--	---	---

Desarrollo de la actividad:

Primera parte: Elaborar y manipular mi lista de productos.

Sugerencia de tiempo invertido: 2 horas y 30 minutos.

La participante realizará una búsqueda respondiendo a las siguientes preguntas que servirán como introducción al tema de los Arreglos en *JavaScript*, mientras se realiza la búsqueda, se sugiere realizar fichas de trabajo organizando la información recabada de la misma, se podrán realizar diagramas, mapas, dibujos, esquemas, cuadros sinópticos, mapas conceptuales, etcétera, que de igual manera, fungirán como material de consulta para el futuro.

- ¿Qué son los *arrays* o arreglos en JavaScript?
- ¿Cuál es la sintaxis de un arreglo?
- ¿Cómo se manejan las posiciones de los elementos en un arreglo?
- ¿Cómo crear un arreglo?
- ¿Qué tipos de datos puede almacenar un arreglo?
- ¿Qué es un método de un arreglo?
- ¿Para qué me sirven los arreglos?
- ¿Para qué me sirven los métodos de estos?
- ¿Los datos almacenados en un arreglo son temporales o permanentes?

La participante realizará lo siguiente:

1. Retomar la lista de productos que conforman el punto de venta que se ha trabajado.
2. Crear una carpeta con nombre "*MWT2A5_Arreglos*" en la carpeta de evidencias, dentro de esta crear un archivo HTML nombrado "productos.html" y una carpeta nombrada "js", dentro de la misma, crear un archivo nombrado "*arreglos.js*"
3. Una vez ligados ambos documentos, en el archivo "*arreglos.js*" realizar un script que contenga lo siguiente:
 - Crear un arreglo y almacenar cada producto de su lista.
 - Calcular el tamaño del arreglo. (*length*)

- Meter un elemento al final. (*push*)
- Meter un elemento al inicio. (*unshift*)
- Sacar un elemento del final. (*pop*)
- Sacar un elemento del inicio. (*shift*)
- Ordenarlo. (*sort*)
- Ponerlo en reversa. (*reverse*)
- Copiar una porción del arreglo. (*slice*)
- Agregar, reemplazar y eliminar elementos. (*splice*)
- Crear un nuevo arreglo y concatenar ambos. (*concat*). El arreglo resultante se trabajará en la parte dos.

Se podrán agregar más scripts a la lista propuesta integrando más métodos de arreglos. En caso de que la participante tenga dudas acerca de cómo elaborar el script, se le motivará para que realice una búsqueda en internet sobre los elementos necesarios para encontrar o descubrir las soluciones. Se procurará no intervenir en la resolución de los problemas, así como en la búsqueda de información. Al finalizar el segmento deberán añadir la carpeta “MWT2A5_Arreglos” a la carpeta de evidencias en la nube, así como una carpeta de nombre “MWT2A5_FichasTrabajo” con la digitalización de las fichas de trabajo realizadas sobre arreglos.

Segunda parte: Manipular datos.

Sugerencia de tiempo invertido: 2 horas y 30 minutos.

La participante realizará una búsqueda respondiendo a las siguientes preguntas que servirán como introducción al tema de los Iteradores en *JavaScript*, mientras se realiza la búsqueda, se sugiere realizar fichas de trabajo organizando la información recabada de la misma, se podrán realizar diagramas, mapas, dibujos, esquemas, cuadros sinópticos, etcétera., que de igual manera, fungirán como material de consulta para el futuro.

- ¿Qué son los *iteradores* o ciclos en JavaScript?
- ¿Con qué otro nombre aparte de los mencionados se le conocen? y ¿Por qué crees que les llaman así?
- ¿Cuál es la sintaxis de un ciclo *for*, de un *forEach*, de un *while* y de un *do-while*?
- ¿Qué diferencias hay entre cada uno de ellos?
- ¿Para qué sirven?

La participante realizará lo siguiente:

1. Crear una carpeta con nombre “MWT2A5_Iteradores” en la carpeta de evidencias, dentro de esta, generar un archivo HTML nombrado “iteradores.html” y una carpeta nombrada “js”, dentro de la misma, crear un archivo nombrado

“ciclos.js”

2. Una vez ligados ambos documentos, escribir un script que mediante *forEach*, *for*, *while* y *do-while* muestre el arreglo elemento por elemento. En el mismo archivo como comentario responder las siguientes preguntas:
 - ¿Qué diferencias hay entre cada implementación de cada iterador?
 - ¿Observaste alguna diferencia entre cada impresión? Si la respuesta anterior fue afirmativa, ¿Por qué?
3. Utilizando *while*:
 - Mostrar los números del 1 al 10 de forma ascendente.
 - Mostrar los números del 1 al 10 de forma descendente.
4. Utilizando *for*:
 - Mostrar la tabla de multiplicar de un número.
 - Mostrar todas las tablas de multiplicar.
5. Utilizando *do-while*:
 - Pedir 10 datos de precios e irlos sumando, al final debe mostrar el total de la suma.
 - Pedir cinco datos de precios y mostrar el promedio de estos.
6. Utilizando el ciclo que la participante crea que se adapte mejor, escribir un script que recorra un arreglo de números y encuentre el número máximo y el número mínimo del arreglo. Utilizar condicionales.

Se podrán agregar más scripts a la lista propuesta de modo que la participante pueda practicar la implementación de los iteradores a la resolución de problemas. En caso de que la participante tenga dudas acerca de cómo elaborar el script, se le motivará para que realice una búsqueda en internet sobre los elementos necesarios para encontrar o descubrir las soluciones. Se procurará no intervenir en la resolución de los problemas, así como en la búsqueda de información. Al finalizar el segmento, deberán añadir la carpeta “MWT2A5_Iteradores” a la carpeta de evidencias en la nube, así como añadir a la carpeta “MWT2A5_FichasTrabajo” la digitalización de las fichas de trabajo realizadas sobre iteradores.

Tercera parte: ¿Y mis cuentas?

Sugerencia de tiempo invertido: 2 horas.

La participante implementará la simulación de una caja registradora en la página web que ha trabajado en las actividades anteriores, para ello deberá crear un nuevo archivo llamado “*Pedido.html*” que tendrá ligado otro nuevo archivo llamado “*Pedido.js*”. La simulación de la caja deberá comprender:

1. Un formulario, con espacio para diez productos , con su nombre y precio, y un botón para enviar la información de los productos. Se podrá dar estilo al formulario.
2. Se deberá hacer uso de arreglos para almacenar los productos y precios, así como usar los iteradores correspondientes

para ir solicitando los datos, y realizae los cálculos correspondientes dentro de este. Se deberán de actualizar las cantidades, cantidad y costo total.

3. Al final de la actividad, la simulación tendrá que visualizarse sin ningún error en la consola del navegador de forma local. Una vez que se determine que su página ya no cuenta con ningún error, se le solicitará a la participante que la suba al repositorio de Git correspondiente.

Notas para apoyar la actividad:

- En caso de que algún script implemente entrada de texto, en la parte uno o dos, las participantes podrán separar los scripts en diferentes archivos *JavaScript* con su respectivo HTML para que no entren en conflicto. La jerarquía de la creación de carpetas y archivos seguirá la misma forma propuesta en el número uno de la parte correspondiente.

Notas para mejorar las habilidades de las participantes:

Recomendar:

- El uso de *shortcuts* y autocompletado como herramientas para escribir de manera más rápida y práctica el código JavaScript en Visual Studio Code.
- La escritura de comentarios dentro de los códigos JavaScript para explicar el funcionamiento de los scripts.
- La práctica de mantener una estructura adecuada para la organización de los archivos (HTML, CSS, JS).

Links de apoyo:

- Páginas que ayudarán a realizar fichas de trabajo:
 - Portal Académico CCH. (Sin fecha). *Fichas de trabajo etapas de la escritura académica: fichas de trabajo*.
<https://portalacademico.cch.unam.mx/alumno/tlriid2/unidad3/fichasTrabajo> (Marzo 2020)
- Páginas que les ayudarán a realizar la búsqueda que se solicita en la primera parte de la actividad:
 - Grados, J. (Sin fecha). *Manejo de Arrays en JavaScript*.
<https://devcode.la/tutoriales/manejo-de-arrays-en-javascript/> (Marzo 2020)
 - MDN Web Docs moz://a. (Sin fecha). *Array*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Objetos_globales/Array (Marzo 2020)
- Páginas que les ayudarán a realizar la búsqueda, que se solicita en la segunda parte de la actividad:
 - Coding Potions. (2019). *Javascript - foreach y y todo sobre los bucles js* 🏆.
<https://codingpotions.com/javascript-foreach-bucles> (Marzo 2020)

- MDN Web Docs [moz://a](https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Bucles_e_iteraci%C3%B3n). (Sin fecha). *Bucles e Iteración*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Bucles_e_iteraci%C3%B3n (Marzo 2020)

Actividad 6: Creando y operando datos específicos

Aprendizaje esperado: Crear y manipular representaciones abstractas de elementos de la realidad en forma de datos.		Duración de la actividad: 9 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Plantilla del anexo disponibles en <i>MWT2A6_Anexos.pdf</i>	Producto: <ul style="list-style-type: none"> ● Agregar a una página web un código JavaScript embebido que simule un catálogo. Deberá contener: un formulario para agregar un nuevo producto; una manera de desplegar todos los productos del catálogo. 	Evaluación: <ul style="list-style-type: none"> ● Se deberán declarar dos objetos uno para el catálogo y otro para los productos del catálogo. ● El catálogo deberá de tener como un atributo una lista de productos. ● Cada producto deberá tener distintos atributos como: nombre del producto, departamento y precio. ● A partir de los datos extraídos del formulario se deberá crear un nuevo <i>Producto</i> y agregarlo al catálogo.
Desarrollo de la actividad:		
<p>Primera parte: Objetos triviales.</p> <p>Sugerencia de tiempo invertido: 3 horas.</p> <ol style="list-style-type: none"> 1. Se le solicitará a la participante realizar una búsqueda en internet del tema los objetos en javascript, deberá abordar los subtemas construcción del objeto, acceso al objeto, declaración y acceso de sus propiedades y el operador <i>this</i>. 2. Podrá realizar las anotaciones que considere necesarias para comprender el tema. 3. Una vez terminado jugará con la participante un juego de trivia, en el cual se le harán preguntas al azar sobre los temas investigados. Se recomienda que realicen tres rondas de cinco preguntas, aunque la tallerista puede incluir más si lo ve conveniente. <ul style="list-style-type: none"> ● Se encuentra disponible en el anexo <i>MWT2A6_Anexos.pdf</i> una plantilla con 15 sugerencias de preguntas. ● También se sugiere realizar la encuesta en la plataforma en línea Kahoot! si las condiciones lo permiten. 4. Dependiendo de la cantidad de respuestas correctas (menos de 60% de respuestas correctas) valore realizar un espacio para aclarar las dudas. 		

Segunda parte: Preparando el sitio para agregar productos

Sugerencia de tiempo invertido: 4 horas.

1. Creando formularios:

La participante deberá crear un formulario para agregar nuevos productos, se sugiere tomar como base lo realizado en la T2A4P1 del taller ya que se requieren los mismos datos para registrar nuevos productos.

2. Objetos:

- Para poder organizar todos los productos será necesario crear un catálogo (para esto se recomienda tener un listado previo de los productos que se van a vender), el cual será el primer objeto que se va a declarar, (Es importante tener previamente definido qué propiedades y métodos son los que tendrá este objeto). Se deberá considerar que el catálogo deberá tener como atributo una lista de productos.
- Para declararlo se deberá crear un nuevo código fuente de javascript, la creación de un objeto se comienza con la definición e inicialización de una variable.
- Se solicitará ingresar lo siguiente debajo del código JavaScript: `'var = catalogo {};` ' se tendrá que guardar y actualizar.
- Una vez creado el objeto se deberá de actualizar con la información que este contendrá. Cada par nombre/valor deberá estar separado por una coma, el nombre y el valor en cada caso deberán estar separados por dos puntos. Revisar el manual sugerido en dudas de sintaxis.
- Se deberánb repetir los mismos pasos para declarar los productos del catálogo como objeto.

3. Lista de productos:

Se necesitará enlistar los productos que se agregarán al catálogo utilizando el constructor del objeto **Array** (Se recomienda utilizar el manual de código)

4. Una vez finalizado lo anterior se tendrá que declarar una clase *Producto* con las propiedades; nombre del producto y existencia,
5. Posteriormente se tendrá que definir como métodos su impresión y otra que retorna *true* si hay productor registrados anteriormente o *false* en caso contrario.
6. Por último, con la función **main** se deberá definir un arreglo con cuatro elementos de tipo *Producto* y se tendrá que calcular cuántos productos hay.

Tercera parte: ¿Dónde están los productos agregados?

Sugerencia de tiempo invertido: 2 horas.

1. La participante deberá agregar un botón a su catálogo para "refrescar" la página.

2. Una vez que este haya sido creado, deberá ejecutarlo y explicar qué es lo que pasó con los productos que ya habían sido añadidos.

Links de apoyo:

Páginas de auxilio (Objetos y dudas de sintaxis):

- MDN Web Docs moz://a. (Sin fecha), *Conceptos básicos de los objetos JavaScript*,
<https://developer.mozilla.org/es/docs/Learn/JavaScript/Objects/Basics> (Abril 2020)
 - *Trabajando con objetos*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Trabajando_con_objetos (Abril 2020)
- tutorialesprogramacionya.com. (Sin fecha), *35 - POO - arreglos con objetos*.
<https://www.tutorialesprogramacionya.com/kotlinya/detalleconcepto.php?punto=35&codigo=35&inicio=30> (Abril 2020)

Páginas de auxilio para la segunda parte 'Lista de productos' (array):

- MDN Web Docs moz://a. (Sin fecha). *Array*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Objetos_globales/Array (Abril 2020)

Páginas de auxilio para la primera parte (This):

- MDN Web Docs moz://a. (Sin fecha). *this*.
<https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Operadores/this> (Abril 2020)

Página de auxilio para la segunda parte (catalogos):

- github.com. (2014). *Catalogo-online/Productos.html*.
<https://github.com/bay007/Catalogo-online/blob/master/Productos.html> (Abril 2020)

Página para realizar trivia online:

- kahoot
<https://kahoot.com/> (Abril 2020)

Actividad 7: Funciones

Aprendizaje esperado: Declarar y usar funciones en JavaScript para su aplicación en la construcción de bibliotecas de funciones y una estructuración del código desarrollado.		Duración de la actividad: 12 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Plantilla del anexo disponibles en <i>MWT2A7_Anexos.pdf</i>	Evidencia: 1. Agregar a una página web un código JavaScript embebido que simule un catálogo. Se tomará como base lo alcanzado en primera parte de la actividad seis. Con la diferencia de que a los objetos Catálogo y Producto se les deberán agregar las siguientes funciones: número de productos en un catálogo; productos que empiecen con una letra determinada; el nombre completo de un producto como una sola cadena; peso del producto; etc.	Retroalimentación: <ul style="list-style-type: none"> Se deberán reconocer las funciones con base en un código concreto. Evaluación: <ul style="list-style-type: none"> Se deberá de asegurar el uso de funciones independientes, para obtener los datos solicitados. En la implementación de las funciones se debe hacer uso de otras funciones implementadas por la participante.
Desarrollo de la actividad:		
Primera parte: Lotería de funciones Sugerencia de tiempo invertido: 3 horas. <ol style="list-style-type: none"> Se le solicitará a la participante realizar una búsqueda en internet del tema Funciones (métodos) en javascript, deberá abordar los subtemas regreso de valores, alcance y llamadas. Podrá realizar las anotaciones que considere necesarias para comprender el tema. Una vez terminado se propondrá jugar lotería. Se seguirán las reglas del juego de lotería tradicional. <ul style="list-style-type: none"> Se proporcionará a la participante y a la tallerista (si hubiera más de una participante se puede jugar entre ellas) una plantilla con 9 códigos (disponibles en el anexo <i>MWT2A7_Anexos.pdf</i>) 		

- La tallerista contará con diversas tarjetas con el nombre de los distintos tipos de funciones, las cuales enunciará en voz alta.
 - Ambas tendrán que poner una ficha (puede ser una bolita de papel, frijol o similar) en el código que corresponda con la función mencionada.
 - Gana quien diga *¡lotería!* primero.
4. Una vez concluido el juego se brindará un espacio para aclarar dudas acerca del tema.
 5. Posteriormente la participante escribirá y ejecutará las funciones de su carta de lotería. Para analizar y confirmar el funcionamiento de las funciones deberán comparar con lo que se imprimen en pantalla y practicar llamados de funciones desde la consola.
 6. Para finalizar, la participante deberá crear sus propias funciones (se sugiere reescribir las de la Actividad 3) y los resultados deberán ser mostrados en consola.

Segunda parte: Practicando las funciones

Sugerencia de tiempo invertido: 3 horas.

Con base en los contenidos creados en la Actividad 6 (códigos) se deberán ejecutar las sentencias correspondientes (revisar manual si existe duda de sintaxis) para resolver los siguientes casos:

1. Se indicará a la participante utilizar 'El método **eval()** ' para mostrar el nombre completo de un producto como una sola cadena de caracteres.
2. Además se deberá utilizar "La función **isNaN**" para verificar que todos los precios de los productos sean números enteros.
3. También se utilizará la función "**show_props()** " para llamar a la lista de productos del catálogo y mostrar los productos que haya en existencia.
4. Por último, se tendrá que definir una función para listar los productos por orden alfabético.

Tercera parte: Optimizando el código.

Sugerencia de tiempo invertido: 3 horas.

1. Se solicitará implementar en el código una función *recursiva* que realice la suma de los productos que se han agregado al catálogo.
2. Se preguntará a la participante ¿Qué otras funciones de los códigos de la actividad 6 podrían ser útiles en el manejo del catálogo?
3. La participante deberá ejecutar las funciones que considere pertinentes y anotar las mejoras realizadas en el código así

como de qué manera estas le ayudan a manejar sus elementos.

Cuarta parte: Reescribiendo todo a funciones

Sugerencia de tiempo invertido: 3 horas.

1. Con base en lo realizado en la tercera parte, se solicitará reorganizar todo el código de la Actividad 5 Parte 3 funciones.

Links de apoyo:

Páginas de auxilio (Funciones):

- MDN Web Docs moz://a. (Sin fecha). *Funciones.*
<https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Funciones> (Abril 2020)
 - *Referencia sobre funciones de JavaScript.*
<https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Funciones> (Abril 2020)
- uniwebsidad. (Sin fecha). *Introducción a JavaScript 4.1. Funciones.*
<https://uniwebsidad.com/libros/javascript/capitulo-4/funciones> (Abril 2020)

Manuales sugeridos para la segunda parte:

- Vc.ehu.es. (Sin fecha). *CAPÍTULO 5 JavaScript.*
<http://www.vc.ehu.es/jiwotvim/ISOFT2009-2010/Teoria/BloqueIV/JavaScript.pdf> (Abril 2020)
- wiener.edu.pe. (Sin fecha) *JavaScript (Manual).*
<http://www.wiener.edu.pe/manuales/4to-ciclo/programacion-web-2/curso-practico-de-javascript.pdf> (Abril 2020)

Página de auxilio Parte 3 (funciones recursivas):

- RIP Tutorial. (Sin fecha). *JavaScript Recursive Fuction.*
<https://riptutorial.com/es/javascript/example/5806/funcion-recursiva> (Abril 2020)

Página de auxilio Parte 4:

- Fenollosa, A. (2018). *JAVASCRIPT EJEMPLO DE CARRITO DE COMPRA.*
<https://programadorwebvalencia.com/javascript-ejemplo-carrito-de-compra/> (Abril 2020)

Actividad 8: Modificando mi página

Aprendizaje esperado: Cambiar el comportamiento visual de una página Web usando JavaScript.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Página web <i>RegistroProducto.html</i>, del proyecto <i>TiendaLinea(NombreDelPoyecto)</i>, trabajada anteriormente. 2. Cualquier medio a elegir para realizar los organizadores gráficos y árbol de nodos. 	<p>Evidencia:</p> <ul style="list-style-type: none"> • Carpeta “MWT2A8_OrganizadorGráfico” con los organizadores gráficos elaborados y árbol de nodos. <p>Producto:</p> <ul style="list-style-type: none"> • Página “RegistroProducto.html” modificada, trabajada en la actividad 4 y 7. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • El organizador gráfico en forma de mesa de la idea principal deberá exponer el concepto de DOM, sus funciones, concepto de árbol DOM y nodo. • El organizador gráfico en forma de mapa conceptual deberá representar jerárquicamente relaciones entre conceptos y cuidar que el contenido sea sintético y preciso. • Los métodos a completar en la tabla de la primera parte, punto dos, deberán corresponder a la acción que se pide. • En el árbol de nodos cada nodo deberá corresponder con la estructura del archivo HTML del cual se realiza. <p>Evaluación:</p> <ul style="list-style-type: none"> • La versión modificada de “RegistroProducto.html” deberá mostrar su contenido en pantalla. Revisar que en consola ya no se muestre nada. • Es necesario hacer uso de Template Strings en el código JavaScript, así

		<p>como dar estilo a toda la página.</p> <ul style="list-style-type: none"> • El formulario de registro de productos no debe omitir ninguna información al momento de desplegar el resumen de productos. • Omitir el funcionamiento de un botón en JavaScript y solo usarlo para disparar el evento de cambio de estilos y contenidos.
Desarrollo de la actividad:		
<p>Primera parte: ¿Cómo accedo a mi página para modificarla?</p> <p>Sugerencia de tiempo invertido: 3 horas.</p> <p>Como introducción al tema, la participante realizará una búsqueda de información sobre el tema <i>Document Object Model</i> (DOM) en <i>JavaScript</i>. Se contemplarán las preguntas guía sugeridas contenidas abajo (también se podrán utilizar materiales de consulta realizados en actividades previas). La búsqueda se dividirá en tres partes, se sugiere realizar un organizador gráfico en cada una que integre y organice la información adquirida y asimismo pueda fungir como material de consulta en el futuro.</p> <p><i>1. Introducción al DOM.</i></p> <p>En esta sección, las participantes realizarán un organizador gráfico en forma de mesa acerca de la idea principal, es decir, “se escribe la idea principal en la “superficie” de la mesa y los detalles en las patas” (UNAM, 2020), sintetizarán la información recolectada de la búsqueda de las siguientes cuestiones.</p> <ol style="list-style-type: none"> ¿Qué es el DOM? ¿Para qué sirve? y ¿cómo funciona? ¿Qué es el árbol DOM? ¿Que es un nodo, en el DOM? y ¿Cómo acceder a los nodos del DOM? <p><i>2. Métodos en el DOM.</i></p> <p>Las participantes completarán una tabla a través de la búsqueda de métodos dentro del DOM; la tabla consta de dos columnas, en la columna izquierda deberá el <i>cómo</i> lograr ciertas tareas en el DOM, y en la columna derecha deberán escribir el método correspondiente con el que se llega a tal tarea, la pregunta 7 podrá responderse fuera del cuadro.</p> <p>Ejemplo:</p>		

Cómo	Métodos
Encontrar elementos por ID.	<code>document.getElementById(id)</code>
Encontrar elementos por nombre de etiqueta.	...

1. Cómo realizar acciones directas dentro del DOM:

- Encontrar elementos HTML por id, clase y por nombre de etiqueta. (`getElementById(id)`, `getElementsByTagName(nombre)`, `getElementsByClassName(nombre)`)
- Cambiar el valor de un atributo de un elemento. (`element.attribute = new value`)
- Cambiar el estilo de un elemento HTML. (`element.style.property = new style`)
- Encontrar elementos HTML mediante selectores CSS (`querySelectorAll()`)
- Cambiar el HTML interno de un elemento (`element.innerHTML = "nuevo contenido"`)
- Cambiar el flujo de salida HTML. (`document.write(texto)`)

2. Cómo realizar acciones con los nodos del DOM:

- Crear (`document.createElement(elemento)`)
- Agregar (`document.appendChild(elemento)`)
- Remover (`document.removeChild(elemento)`)
- Reemplazar (`document.replaceChild(nuevoElemento, antiguoElemento)`)

3. ¿Qué diferencias existen entre `document.write` e `innerHTML`? ¿En qué casos se debe utilizar cada uno?

3. *Template Strings e introducción a eventos.*

Las participantes realizarán una búsqueda respondiendo a las cuestiones mencionadas abajo, realizarán un organizador gráfico en forma de mapa conceptual.

- ¿Qué son los template strings?
- ¿Cómo utilizarlos? y ¿Para qué sirven?
- ¿Qué es un evento?
- ¿Cómo puedo utilizar un evento para cambiar un comportamiento en mi página?
- ¿Cómo agregar un evento a un botón?

4. *Mi árbol de nodos.*

Las participantes elaborarán un árbol de jerarquía de nodos representando el código HTML de la página principal del

proyecto *TiendaLinea[NombreDelProyecto]*, podrán utilizar cualquier herramienta para dibujarlo.

Una vez finalizado este segmento, se habrán de recopilar todas las evidencias digitalizadas (organizadores gráficos y árbol de nodos) e ingresarlas en una nueva carpeta de nombre “MWT2A8_OrganizadorGráfico”, la cual será subida a la carpeta de evidencias en la nube.

Segunda parte: Modifico mi página con el DOM.

Sugerencia de tiempo invertido: 2 horas.

La participante re-escribirá el código correspondiente a la página “RegistroProducto.html”, el código *JavaScript* deberá implementar los conocimientos adquiridos en esta actividad, de modo que al realizar un registro de los productos, debajo del formulario se deberá mostrar un resumen listando los productos registrados.

Contemplar lo siguiente:

1. Implementar el evento correspondiente para desplegar el resumen del registro. Por el momento se deberá omitir el funcionamiento de un botón en *JavaScript* y solo usarlo para mostrarlo.
2. Se deberá dar estilo a la página web o complementar dándole estilo al resumen del registro.
3. Hacer uso de *template strings* al momento de reescribir el código.
4. Al final de la actividad, la página web tendrá que visualizarse sin ningún error.
5. Una vez que se determine que su página ya no cuenta con ningún error, se le solicitará a la participante que suba las modificaciones al repositorio de Git correspondiente.

Como cierre de actividad las participantes discutirán en parejas, grupos o con la tallerista las respuestas a la siguientes preguntas:

- ¿Por qué al refrescar o recargar la página se pierden los datos “guardados”?
- ¿Qué podría hacer para que no se pierdan?

Notas para mejorar las habilidades de las participantes:

Recomendar:

- El uso de *shortcuts* y autocompletado como herramientas para escribir de manera más rápida y práctica el código JavaScript en Visual Studio Code.
- La escritura de comentarios dentro de los códigos JavaScript para explicar el funcionamiento de los scripts.

Referencias:

- fcaenlinea1.unam.mx. (Sin fecha). *Organizadores gráficos..*
http://fcaenlinea1.unam.mx/anexos/organizadores_graficos.pdf (Marzo 2020)

Links de apoyo:

- Recursos para realizar organizadores gráficos.
 - Pimienta, J. (2012). *Estrategias de enseñanza-aprendizaje. Docencia universitaria basada en competencias*. México. Pearson .
http://prepajocotepec.sems.udg.mx/sites/default/files/estrategias_pimiento_0.pdf. (Abril 2020)
 - fcaenlinea1.unam.mx. (Sin fecha). *Organizadores gráficos*.
http://fcaenlinea1.unam.mx/anexos/organizadores_graficos.pdf (Marzo 2020)
- Páginas que les ayudarán a realizar la búsqueda, que se solicita en la primera parte de la actividad:
 - Maestros del web by Platzi. (2007). *Introducción a la manipulación del DOM mediante Javascript*.
<http://www.maestrosdelweb.com/dom/>(Marzo 2020)
 - Millanao, Marcia. (2017). *Manejo de Dom. Navegar por nodos,-*.
<https://medium.com/laboratoria-how-to/manejo-de-dom-f15e8816e696> (Marzo 2020)
 - w3schools.com. (Sin fecha). *JavaScript HTML DOM Document*.
https://www.w3schools.com/js/js_htmlDOM_document.asp (Marzo 2020)
 - *JavaScript HTML DOM Elements*.
https://www.w3schools.com/js/js_htmlDOM_elements.asp (Marzo 2020)
 - *JavaScript HTML DOM - Changing HTML*.
https://www.w3schools.com/js/js_htmlDOM_html.asp (Marzo 2020)
 - *JavaScript HTML DOM - Changing CSS*.
https://www.w3schools.com/js/js_htmlDOM_css.asp (Marzo 2020)
 - *JavaScript HTML DOM Elements (Nodes)*.
https://www.w3schools.com/js/js_htmlDOM_nodes.asp (Marzo 2020)
 - desarrolloweb.com. (2017). *Template Strings en ES6*.
<https://desarrolloweb.com/articulos/template-strings-es6.html> (Marzo 2020)
 - MDN Web Docs moz://a. (Sin fecha). *Plantillas literales (Plantillas de cadena)*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/template_strings (Marzo 2020)

Actividad 9: Toda acción tiene una reacción

Aprendizaje esperado: Implementar la funcionalidad requerida para que los componentes de una página web reaccionen a cambios producidos por el usuario durante su interacción con ésta.		Duración de la actividad: 4 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Página web “Pedido.html” del proyecto TiendaLinea(NombreDelPoyecto], trabajada anteriormente. 2. Cualquier medio a elegir para realizar la tabla de búsqueda de eventos.	Evidencia: <ul style="list-style-type: none"> ● Carpeta “MWT2A9_Eventos” que contendrá la tabla de búsqueda de eventos (evento, descripción, elementos, ejemplo) digitalizada. Producto: <ul style="list-style-type: none"> ● Código HTML de “Pedido.html” modificado mediante el DOM. 	Retroalimentación: <ul style="list-style-type: none"> ● En la tabla de búsqueda de eventos, los apartados <i>descripción</i>, <i>elementos para los que está definido</i> y <i>sintaxis</i>, deberán corresponder al evento que se está describiendo. Evaluación: En la modificación del archivo “Pedido.html”: <ul style="list-style-type: none"> ● Revisar que el registro de productos se agregue de forma dinámica. ● La versión modificada de “Pedido.html” deberá mostrar su contenido en pantalla. Revisar que en consola ya no se muestre nada. ● Es necesario hacer uso de Template Strings en el código JavaScript, así como dar estilo a la página. ● El formulario de registro de productos no debe omitir ninguna información al momento de desplegar el resumen de productos, asimismo ahora sólo deberá contar con tres campos.
Desarrollo de la actividad:		

Primera parte: Enciclopedia de eventos.

Sugerencia de tiempo invertido: 1 hora 30 minutos.

La participante elaborará una búsqueda en internet de los eventos más comunes en JavaScript. Con la información recopilada deberán realizar una tabla de cuatro columnas, en la primera se deberá escribir el nombre del evento, en la segunda columna una breve descripción del funcionamiento de este, en la tercera columna los elementos para los que está definido, y en la cuarta la sintaxis de este. Cabe aclarar que en la actividad anterior se vio el tema de Eventos, sin embargo, en esta sección se profundizará sobre los distintos tipos de Eventos.

Ejemplo:

<i>Evento</i>	<i>Descripción</i>	<i>Elementos para los que está definido</i>	<i>Sintaxis</i>
onfocus	Un elemento obtiene el foco	<button>, <input>, <label>, <select>, <textarea>, <body>	<element onfocus="myScript">
...

Se sugiere que la búsqueda cubra al menos los siguientes eventos:

- onclick
- ondblclick
- onmousedown
- onmousemove
- onmouseover
- onmouseout
- onkeydown
- onkeypress
- onkeyup
- onfocus
- onblur
- onchange
- onreset
- onresize
- onselect

- onsubmit
- onunload

Debajo del cuadro la participante incluirá los siguientes métodos, junto con la sintaxis y explicación de cada uno:

- addEventListener()
- removeEventListener()

Una vez finalizado este segmento, se ingresará la tabla de búsqueda de eventos a una nueva carpeta de nombre “MWT2A9_Eventos”, la cual será subida a la carpeta de evidencias en la nube.

Segunda parte: Animando mi página web.

Sugerencia de tiempo invertido: 2 horas y 30 minutos.

La participante re-escribirá el código correspondiente a la página “Pedido.html”, el código JavaScript deberá implementar los conocimientos adquiridos en esta actividad, de modo que al efectuar un pedido de los productos, debajo del formulario se deberá mostrar el resumen listando los productos pedidos. Contemplar lo siguiente:

1. El registro de los pedidos se agregará de forma dinámica por lo que ahora el formulario constará de solo tres campos: nombre, precio y cantidad.
2. Los cálculos del total y subtotal se calcularán dinámicamente.
3. Implementar los eventos correspondientes para el funcionamiento del registro de pedidos y para mostrar el resumen de estos.
4. En el resumen se deberá visualizar el listado de los productos pedidos, el total y subtotal.
5. La página web deberá presentar estilo.
6. Al final de la actividad, la página web tendrá que visualizarse sin ningún error.

Una vez que se determine que su página ya no cuenta con ningún error, se le solicitará a la participante que suba las modificaciones al repositorio de Git correspondiente.

Notas para mejorar las habilidades de las participantes:

Recomendar:

- El uso de shortcuts y autocompletado como herramientas para escribir de manera más rápida y práctica el código JavaScript en Visual Studio Code.
- La escritura de comentarios dentro de los códigos JavaScript para explicar el funcionamiento de los scripts.

Links de apoyo:

- Páginas que les ayudarán a realizar la búsqueda, que se solicita en la primera parte de la actividad:
 - MDN Web Docs moz://a. (Sin fecha). *Introducción a eventos*.
https://developer.mozilla.org/es/docs/Learn/JavaScript/Building_blocks/Eventos (Marzo 2020)
 - NorfiPC. (Sin fecha). *Lista de atributos de eventos de javascript*.
<https://norfipc.com/inf/javascript-lista-eventos-paginas-web.html#> (Marzo 2020)
 - desarrolloweb.com. (2003). *Los eventos en Javascript*.
<https://desarrolloweb.com/articulos/1235.php> (Marzo 2020)
 - Zubieta, Maria. (2017). *Eventos JS*.
<https://medium.com/laboratoria-how-to/eventos-js-la-fiesta-del-c%C3%B3digo-448fcf595a36> (Marzo 2020)
 - w3schools. (Sin fecha). *JavaScript HTML DOM EventListener*.
https://www.w3schools.com/js/js_htmldom_eventlistener.asp (Marzo 2020)
 - Uniwebsidad. (Sin fecha). *Introducción a AJAX. 6.3.Handlers y listeners*.
<https://uniwebsidad.com/libros/ajax/capitulo-6/handlers-y-listeners> (Marzo 2020)

Actividad 10: Instalación y configuración de servicios para sistemas web

Aprendizaje esperado: Instalar en un entorno de servidor web, con los frameworks Node.js y Express.js.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none">Proyecto TiendaLinea[NombreDelProyecto] trabajado en las actividades anteriores.<i>MWT2A10_Anexo.pdf</i>	Evidencia: <ol style="list-style-type: none">Generar y desplegar una página web dinámica con enlaces internos a través de un servidor.	Retroalimentación: <ol style="list-style-type: none">Se despliega a través del servidor Express.js, de forma local.Asegurarse de que se entendió el concepto de cliente - servidor, teniendo énfasis en la comunicación entre ambos.
Desarrollo de la actividad:		
<p>Primera parte: Diferentes, pero todos importantes. Sugerencia de tiempo invertido: 2 horas. Se realizará una búsqueda sobre las principales características de JavaScript, node.js, Express.js y middleware, con la información obtenida se elaborará un organizador gráfico (OG), a elección de la participante, en el cual se diferencien las principales características de cada uno siendo clave el contraste entre las mismas. El OG deberá incluir las siguientes características:</p> <ul style="list-style-type: none">JavaScript<ul style="list-style-type: none">Lenguaje de programaciónInterpretadoTipado débilNode JS<ul style="list-style-type: none">Entorno de ejecución para javascript (ejecutar javascript fuera de un navegador web)Asíncrono y orientado a eventosContiene un gestor de paquetes npmExpress JS		

- o Paquete/ Librería de npm
- o Framework para realizar páginas web
- Middleware
 - o Ciclo request - response
 - o Se ejecuta una función una después de otra (en orden)
 - o ¿De qué se compone una función middleware en express?
 - o Tipos de middleware

El OG se mostrará y explicará a la tallerista u otras participantes, según sea la disposición específica del grupo. Puede hacerse uso de los links de referencia como material de apoyo para la realización de este ejercicio.

Segunda parte: De la receta a mi proyecto.

Sugerencia de tiempo invertido: 1 hora 30 minutos.

Se otorgará a la participante una copia de “Servidor a la Express.js /Servidor estilo Express.js” (que se puede encontrar en *MWT2A10_Anexo.pdf*) o un ejercicio similar, en el que se ponga en práctica lo aprendido hasta ahora.

La participante ya está familiarizada con este anexo, pues se utilizó en la MWT1A8 para desplegar el servidor. Ahora se vuelve a presentar la receta, pero con espacios en blanco a rellenar, con el propósito de que la participante reconozca y entienda de mejor manera el proceso.

Tercera parte: Construcción de proyectos complejos.

Sugerencia de tiempo invertido: 1 hora

La participante conocerá la herramienta *express generator* para la creación rápida y fácil de un proyecto en express. Se deberán investigar y realizar los siguientes puntos:

- 1) ¿Qué es *express application generator*? ¿Para qué sirve?
- 2) ¿Qué es un motor de plantillas?
- 3) ¿Qué motores de plantillas admite *express application generator*? Investiga al menos dos
- 4) ¿Con qué comando podemos generar un proyecto sin motor de plantillas definido?
- 5) Crea un proyecto con alguno de los motores de plantillas investigados en el punto 3 (proyecto 1)

- 6) Crea un proyecto sin motor de plantillas definido (proyecto 2)
- 7) Observa la diferencia entre los archivos creados en ambos proyectos y realiza una comparación
- 8) ¿Cómo se puede ejecutar el proyecto?
- 9) Ejecuta el proyecto 1. ¿Qué observas?
- 10) Ejecuta el proyecto 2. ¿Qué observas?
- 11) ¿Existe alguna diferencia entre sus resultados?
- 12) En caso de que existieran errores ¿qué los originó? ¿cómo se podrían resolver?
- 13) Adjunta una captura de la estructura de carpetas generada e investiga brevemente qué tipo de archivos almacena cada carpeta
- 14) ¿En qué se diferencia a la estructura generada con la receta “Servidor a la Express”? ¿Existen similitudes?
- 15) ¿Por qué consideras que se necesitan más carpetas, conforme aumenta la complejidad del proyecto?

Enlaces:

- Express. (Sin fecha). Generador de aplicaciones Express, Documentación oficial express generator.
<https://expressjs.com/es/starter/generator.html>

Actividad 11: ¿A dónde van los datos?

Aprendizaje esperado: Enviar, recibir y procesar información entre un cliente y un servidor web.		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Proyecto TiendaLinea[NombreDelProyecto] trabajado en las actividades anteriores.	Evidencia: Exposición oral sobre la comunicación entre el servidor, una computadora y la relación con Express. Producto: Generar una página web con dos formularios: <ol style="list-style-type: none"> 1. Para ingresar información. 2. Despliegue la información solicitada. 	Retroalimentación: Asegurarse que se comprendió la relación de Express y la comunicación entre el servidor y una computadora verificando el funcionamiento en la página. Evaluación: La información ingresada se deberá procesar en el servidor, para que sea modificada y cuando se envíe de regreso se despliegue la información modificada.
Desarrollo de la actividad:		
<p>Primera parte: ¡Así funciona la web!</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <ol style="list-style-type: none"> 1. La participante realizará una búsqueda a través de Internet centrándose en los siguientes temas: <ol style="list-style-type: none"> a. Protocolo HTTP y TCP b. API Rest c. Métodos HTTP (definición, uso de GET y POST) <p>Posterior a ello se deberá complementar la información recabada con algunos videos donde la explicación sea asequible o bien, utilizar el material recomendado para dicho ejercicio.</p> 2. Con la información recabada en el punto anterior, la participante realizará un bosquejo de la comunicación cliente-servidor en la web (es posible retomar el esquema realizado en el taller anterior: MWT1A9) agregando los 		

conceptos de los protocolos HTTP y TCP, destacando la estructura de un request y response en HTTP, el lugar en que se situaría el API Rest y la relación de lo anterior con Express.js.

Segunda parte: Hablándole al patito de hule (Metodología del Patito de hule)

Sugerencia de tiempo invertido: 1 hora y 30 minutos.

La metodología del patito de hule surge en los ambientes de programación, donde en ocasiones los proyectos se ven truncados por problemas técnicos, obligando a los programadores a hablarse a sí mismos, o a algún objeto de su escritorio, como un patito de hule, buscando respuestas, situación que puede ocurrir en nuestra vida cotidiana.

Retomando la primera parte de la actividad, la participante realizará una exposición oral del esquema generado y la información recabada. Como evidencia deberá grabar un audio no mayor a 10 minutos, el objetivo principal de este audio es explicar cómo funciona la web y en dónde encajan los conceptos aprendidos en esta actividad.

Tercera parte: De la teoría a la realidad.

Sugerencia de tiempo invertido: 1 hora

Con el fin de entender y reafirmar cómo funciona el protocolo HTTP y la estructura de las solicitudes, la participante analizará los archivos que se cargan al acceder a cualquier sitio, además revisará los headers de request y response para algunos de los archivos cargados. Los sitios se dejan a la elección de la tallerista pero se recomienda que sean sitios que aborden la perspectiva de género.

Nota: Se espera que esta actividad se asemeje a la que se observa en el vídeo “¿Qué es HTTP? - Introducción Práctica” que se puede encontrar en las referencias.

Cuarta parte: Una mirada a los hogares.

Sugerencia de tiempo invertido: 2 horas.

La participante realizará un formulario que implemente los métodos GET y POST en Express.js. El formulario tendrá como propósito saber quién tiene más participación en el hogar ante determinadas actividades, por lo tanto deberá tener campos donde se pueda capturar esta información, algunos ejemplos de actividades pueden ser: lavar los platos, cocinar, barrer, planchar, ir al super, etcétera.

Las especificaciones para esta actividad serán las siguientes:

- Se implementarán dos formularios los cuales serán idénticos y deberá desplegar al menos cinco actividades, cada una con su correspondiente radio button para hombre y mujer, así como un único botón para enviar.
- Un formulario a través del botón enviar implementará el método GET mientras que el segundo implementará el método POST.
- Los datos serán enviados al servidor el cual guardará los datos enviados, que serán las actividades realizadas por género, en un arreglo independiente para hombre y mujer y realizará la suma de actividades por género en el *backend* de la aplicación. El resultado se deberá de mostrar en una página adicional que se carga automáticamente al presionar el botón de “enviar” donde se mostrarán las actividades por género (*frontend*), la página debe destacar qué actividad es más realizada ya sea por hombres o mujeres.
- Se deberá probar el buen funcionamiento de esta con al menos cinco registros, es mejor si esta información proviene de sus compañeros para tener un panorama óptimo de la situación. Se deberá mostrar a la participante la diferencia entre los métodos GET y POST.

Nota: Los datos en este paso son volátiles, esto es, no se almacenan en alguna base de datos.

Links de apoyo:

- ¿Qué es HTTP? ¿Qué es TCP?
 - Appdelante. (2018). *¿Qué es HTTP?-Introducción Práctica*. [video]
<https://www.youtube.com/watch?v=WlIkVOWIT4&t=11s> (Abril 2020)
 - unComo. (2017). *Cómo funciona una Web*. [video]
<https://www.youtube.com/watch?v=XHOo3EQGmfA> (Abril 2020)
 - Trojan Cookie Protocol. (2017). *HTTP Explained | Request-and-Response | Web Server/Client | Connection Type | TCP | Port Number*. [video]
<https://www.youtube.com/watch?v=RUD8Q8E4mXs> (Abril 2020, Recurso en inglés)
- ¿Qué es una API Rest?

- Appdelante. (2018). *Rest y los Verbos de HTTP en 20 minutos*. [video]
<https://www.youtube.com/watch?v=OHBHeAPoZ8E> (Abril 2020)
- Métodos HTTP definición y explicación de GET y POST
 - Appdelante. (2018). *Rest y los Verbos de HTTP en 20 minutos*. [video]
<https://www.youtube.com/watch?v=OHBHeAPoZ8E> (Abril 2020)

Actividad 12: Uppssss me equivoqué!!!! (Modificando y borrando información)

Aprendizaje esperado: Actualizar y procesar la información enviada a un servidor web a partir de un cliente.		Duración de la actividad: 6 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. Proyecto TiendaLinea[NombreDelProyecto] trabajado en las actividades anteriores. 2. Tarjeta (hojas divididas en cuatro recortadas) o fichas bibliográficas. 3. Cartulina.	Evidencia: Tarjetas con métodos HTTP con su explicación y sus características. Producto: Dos formularios: 1. Un formulario editable que muestre un registro ya guardado. 2. Permite eliminar un registro previamente guardado.	Retroalimentación: Asegurarse de que ambas caras de las tarjetas correspondan . En caso de ser necesario reforzar o afianzar los conceptos de los métodos HTTP al finalizar la presentación de la participante. Evaluación: 1. Poder hacer cambios y guardarlos. 2. Poder eliminar registros guardados.
Desarrollo de la actividad:		
Primera parte: El recuento hasta ahora. Sugerencia de tiempo invertido: 2 horas. <ol style="list-style-type: none"> 1. La participante realizará una búsqueda sobre los métodos HTTP faltantes (PUT y DELETE), poniendo especial atención en la acción que realiza cada uno y sus principales características, se invita a indagar sobre otros métodos de crearlo conveniente. 2. Elaborará tarjetas con lo obtenido en la búsqueda, que por una cara muestren el nombre de los métodos y su escritura en Express.js y por la cara contraria contengan la explicación y las principales características. Se deberán incluir también los métodos GET y POST. 3. En una cartulina, la participante elaborará un bosquejo de mapa sobre los métodos HTTP, representando gráficamente al cliente y al servidor y dejando un espacio para sobreponer las tarjetas anteriormente elaboradas, colocándolas con la cara de los nombres y escritura Express.js expuesta. 4. La participante presentará su esquema a la tallerista o a sus compañeras, haciendo mención de las características y explicaciones de cada uno de los métodos HTTP, en caso de ser necesario, podrá revisar el reverso de las tarjetas (Las tarjetas se pueden engrapar, o mantenerse unidas mediante un clip para facilitar su consulta). 		

Segunda parte : Agregando una API a mi sitio.

Sugerencia de tiempo invertido: 3 horas

La participante creará una API REST que implemente la consulta, creación, actualización y eliminación en el *backend* del servidor del punto de venta. Se puede usar el ejemplo visto en la actividad anterior (Actividad 12), la cual ya cuenta con la página para la creación de un nuevo registro y para mostrar un registro. Se deberán agregar páginas para editar el registro mostrado (actualización); y otra para mostrar los registros agregados (consulta); en esta última cada registro se mostrará en un estilo de lista y cada elemento tendrá ligas para la edición de dicho renglón, para editar y para eliminar en el *frontend* del servidor. También es viable modificar la página de productos que se ha trabajado o generar una nueva con alguna otra temática.

Contestar las siguientes preguntas:

- ¿Para qué sirve una API Rest?
- ¿Cuál es el usuario final de una API Rest?
- ¿Por qué razón sería útil una API Rest para mi proyecto?
- Investigar el nombre de al menos 10 API's públicas ya existentes y por lo menos 2 cosas que se puedan hacer con ellas.

Tercera parte: Y ahora ¿cómo pruebo mi API?

Sugerencia de tiempo invertido: 1 hora

La participante observará cómo se comporta su página cuando se realiza una petición con los métodos aprendidos, a través de la extensión de Google Chrome Postman o Restful para Mozilla Firefox.

Notas para apoyar la actividad: Se recomienda hacer uso de las referencias para las actividades o buscar otras fuentes que sigan la misma línea temática.

Links de apoyo (primera parte):

1. Appdelante. (2018). *Rest y los verbos de HTTP en 20 minutos. [video]*
<https://www.youtube.com/watch?v=OHBHeAPoZ8E> (Abril 2020)
2. [Esquema] phpenthusiast.com.(Sin fecha). Sin título.
Tomado de Developing in spanish (2019) Fundamentos JavaScript: Introducción a las API REST

https://phpenthusiast.com/theme/assets/images/blog/what_is_rest_api.png (Abril 2020)

Links de apoyo (segunda parte):

1. Why GitHub? (Sin fecha). Public-apis.
<https://github.com/public-apis> (Enero 2021)

Links de apoyo (segunda parte):

1. Azaustre, C. (2016). *Cómo usar POSTMAN para testear tu API REST | Curso NodeJS y MongoDB #06.* [video]
<https://www.youtube.com/watch?v=YMJEAFVl63E> (Abril 2020).

Actividad 13: Altas, bajas y cambios.

Aprendizaje esperado: actividad: Explicar la utilidad de un sistema de altas, bajas y Cambios (CRUD), así como su arquitectura básica basada en el paradigma Modelo-Vista-Controlador (MVC).		Duración de la actividad: 5 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
3. Proyecto TiendaLinea[NombreDelProyecto] trabajado en las actividades anteriores. 4. Preguntas guía y MWT2A10_Anexo.pdf	Producto: <ul style="list-style-type: none"> ● Implementar un sistema basado en Node.js y Express.js que implemente el paradigma de Modelo Vista Controlador para un sistema de altas, bajas y cambios. 	Evaluación: La participante será capaz de explicar en qué consiste el MVC. El almacenamiento de la información se puede realizar de forma local y volátil. De tal manera que la información no persista permanentemente.
Desarrollo de la actividad:		
<p>Primera parte: Divide y vencerás.</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>Dividir para vencer, hace referencia a un recurso metodológico para el abordaje de problemas, proyectos o situaciones que surgen durante el desarrollo de alguna actividad como la programación. Dividir un tema, para ser abordado de una manera más sencilla, es una buena práctica en programación, pues constantemente equipos de programación se enfrentan a proyectos y problemáticas en el desarrollo de algún producto, por lo que éste recurso metodológico propicia el trabajo en equipo y la división del problema para así lograr sortear, en las mejores condiciones, las problemáticas que se presenten.</p> <ol style="list-style-type: none"> 1. La participante realizará una búsqueda en Internet sobre los conceptos y características de MVC y CRUD. 2. Con la información recabada, la participante diseñará un ejercicio (puede usarse de ejemplo “Servidor a la Express.js / Servidor estilo Express.js” del MWT2A10_Anexo.pdf). Dicho ejercicio responderá de manera directa o indirecta las siguientes cuestiones que involucran conceptos y su asociación con el proyecto de la tienda: <ul style="list-style-type: none"> ○ ¿Qué es el MVC? ○ ¿Por qué es importante el MVC? ○ ¿Es posible aplicar o ya está siendo aplicado el MVC a nuestro proyecto? ○ ¿Se puede aplicar del todo hasta ahora o hay alguna pieza faltante? ○ ¿Quién sería el modelo? 		

- ¿Quién sería la vista?
- ¿Quién sería el controlador?
- ¿El MVC es algo exclusivo del desarrollo web?
- ¿A qué hace referencia CRUD?
- ¿Qué partes del CRUD ya han sido aplicadas al proyecto?

Segunda parte : CRUD y MVC aplicados a mi proyecto.

Sugerencia de tiempo invertido: 4 horas.

- La participante aplicará los cambios correspondientes para implementar CRUD al proyecto de la tienda. En el proyecto deberán ser identificables los siguientes cambios :

CREATE

- Agregar el código correspondiente para que los productos registrados se guarden en el servidor, en forma de un arreglo en el *backend*. Adicional a ello se debe modificar la parte de pedidos para que el precio se tome de la información de productos en el servidor.

READ

- Generar una nueva página que despliegue el listado de productos registrados obtenidos del servidor en el *frontend*, con el nombre "ListadoProductos.html"

UPDATE

- Modificar la página de listado de productos para que sea posible desde ahí actualizar el precio de un producto.

DELETE

- Agregar el proceso de borrado en la página del listado de productos (*backend y frontend*).

Una vez que en el sitio se vean reflejados estos cambios, resaltar que ya no se pierde la información de productos si se recarga la página, sin embargo si se recarga el sitio sí se pueden perder. En este punto podrá ser inferida la pieza faltante del MVC.

Links de apoyo (primera parte).

- Vida MMR. (2018). *¿Qué es patrón Modelo Vista Controlador o MVC y cómo funciona?* [#BasicosDev](#). [video]

<https://www.youtube.com/watch?v=zhSDjntidws> (Abril 2020)

Actividad 14: Y ¿dónde quedó mi información?

Aprendizaje esperado: Persistir la información generada en un sistema de altas, bajas y cambios (CRUD).		Duración de la actividad: 8 hrs 30 minutos
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
1. MongoDB 2. NodeJS 3. Anexos <i>MWT2A16_Anexos</i>	Evidencia: <ol style="list-style-type: none"> 1. Fichas de consulta. 2. Una tabla. 3. Una página web con un formulario que solicite: nombre de contacto, número de casa, número de celular y relación. 4. Una base de datos con nombre <i>miAgendaConMongo</i>. 5. Una página web que muestre los contactos almacenados. Producto: <ol style="list-style-type: none"> 1. Página <i>Lista de pedidos</i>. 	Retroalimentación: <ol style="list-style-type: none"> 1. Las fichas de consulta deberá corresponder a las preguntas 1, 2, 7, 8, y 9 de la segunda parte de la actividad. 2. La tabla deberán corresponder a las preguntas 3- 6 de la segunda parte de la actividad. Asimismo, deberá poseer el formato que se ejemplifica en esta parte de la actividad. 3. La página web que muestre los contactos almacenados deberá mostrar los datos completos que se recolectaron inicialmente en la página web con formulario. Evaluación: <ol style="list-style-type: none"> 1. La página <i>Lista de pedidos</i> deberá contar con: <ol style="list-style-type: none"> 1. CRUD para el Schema <i>Pedido</i> en MongoDB que registre los datos ingresados en la vista <i>Pedido.html</i>. 2. Vista de la página con nombre <i>ListaDePedidos.html</i> que

		será mostrada respecto a la consulta sobre el Schema <i>Pedido</i> .
Desarrollo de la actividad:		
<p>Primera parte: Datos por aquí y por allá. Sugerencia de tiempo invertido: 30 minutos.</p> <ol style="list-style-type: none"> 1. Se le presentará a la participante una historia acerca de la situación de una mujer que haga uso de una base de datos en la vida cotidiana, como por ejemplo, una agenda telefónica, un diccionario, una cuenta de registro, listas de clientes, entre otros. En dicha historia (se recomienda utilizar la plantilla 1 de <i>MWT2A14_Anexos</i>), deberán insinuarse las características de las bases de datos. 2. Una vez que la situación haya sido presentada a la participante, esta deberá realizar una reflexión a través de responder a algunas preguntas con referencia a la definición y las características de las bases de datos (se recomienda utilizar la plantilla 2 de <i>MWT2A14_Anexos.pdf</i>). Posteriormente, deberá compartir de forma oral sus conclusiones. <p>Nota: Es imprescindible dirigir todo el ejercicio hacia la reflexión de que las bases de datos pueden ser encontrados en la vida diaria, y que no necesariamente están registradas de manera digital.</p> <p>Segunda parte: Datos en una base. Sugerencia de tiempo invertido: 2 horas.</p> <p>Después de haber llevado a cabo la reflexión en la primera parte, se introducirá a la participante el concepto de <i>bases de datos</i>. Antes de solicitar una búsqueda de información en internet, se preguntará:</p> <ul style="list-style-type: none"> • ¿Habías escuchado el término anteriormente? ¿En qué contexto? • ¿Crees que en la primera parte de la actividad se estaba haciendo referencia a una base de datos? • ¿Cómo definirías una base de datos? <p>Posteriormente, se le proporcionarán a la participante algunos enlaces previamente seleccionados, donde podrá consultar las siguientes cuestiones:</p> <ol style="list-style-type: none"> 1. ¿Que es una base de datos? 2. ¿Qué es MongoDB? 3. ¿Cuál es la función para crear un objeto con MongoDB y JavaScript? 4. ¿Cuál es la función para obtener un objeto con MongoDB y JavaScript? 5. ¿Cuál es la función para actualizar o modificar un objeto con MongoDB y JavaScript? 		

6. ¿Cuál es la función para eliminar un objeto con MongoDB y JavaScript?
7. ¿A qué pertenecen las cuatro acciones correspondientes a las cuatro preguntas anteriores?
8. ¿Cuáles son las instrucciones o pasos para configurar una base de datos en MongoDB con Node.js?
9. ¿Qué empresas utilizan MongoDB en sus sitios web?

La participante deberá realizar una tabla como la que se muestra a continuación, únicamente con la información correspondiente a las preguntas 3 - 6. Del resto de ellas, deberá realizar fichas de consulta como las trabajadas en las actividades anteriores.

	Funciones con MongoDB y JavaScript
Para crear un objeto	
Para obtener un objeto	
Para actualizar o modificar un objeto	
Para eliminar un objeto	

Tercera parte: Te registro, te llamo o te olvido.

Sugerencia de tiempo invertido: 2 horas.

Se solicitará la creación de una página web que contenga un formulario básico para poder realizar una agenda de contactos. Dicha agenda deberá ser llenada con algunas de las mujeres más cercanas a la participante, como pueden ser familiares, amigas u otras. Con la intención de que sea un ejercicio sencillo, se limitará el número de contactos a cuatro como mínimo y máximo ocho.

1. El formulario deberá solicitar información de contacto de los campos: nombre completo, número de casa, número de celular y relación (de dónde conocen a la persona).
2. Una vez creado el formulario, la participante deberá crear un Schema llamado "ContactoSchema" en una base de datos con MongoDB. La base de datos deberá llamarse "miAgendaConMongo". El Schema deberá tener los campos correspondientes para almacenar la información solicitada en el formulario.

Nota: Es necesario hacer énfasis en que la participante tiene que elegir cuidadosamente el tipo de dato que corresponde a cada campo del Schema, de tal forma que el almacenamiento no falle.

3. Una vez que la página anterior funcione, la participante deberá crear otra, con la intención de mostrar todos los contactos registrados. Para esto deberá hacer uso de la función para mostrar objetos con MongoDB.

Nota: No es necesario que las páginas de esta parte se muestren muy estilizadas; la participante deberá enfocarse en la funcionalidad, sin importar el diseño o la manera en que se visualicen los objetos contenidos. Se asegurará que lo registrado en la base de datos corresponda con lo que se puede visualizar cuando consulta la misma base de datos.

Cuarta parte: ¡Ya casi terminamos!

Sugerencia de tiempo invertido: 4 horas.

La participante escribirá el código correspondiente a la página “ListaDePedidos.html”. El código *JavaScript* deberá encargarse del mostrar todos los pedidos registrados a través de la página *Pedido.html* y que deben encontrarse registrados en la base de datos. La participante contemplará lo siguiente:

1. Para poder realizar la lista de pedidos se desarrollará el CRUD en la base de datos, correspondiente a la vista de captura de un pedido, para ello es necesario crear un Schema “Pedido” que guarde los datos que se registran en la vista *Pedido.html* creada en la actividad 9 de este taller. El schema deberá asegurar que la información de un pedido deje de ser volátil y ahora se encuentre sobre la base de datos creada en MongoDB.
2. Implementará la función que permita mostrar todos los elementos de la base de datos correspondientes a los pedidos realizados.
3. Los pedidos deberán mostrarse identificados uno de otro. Es importante que la consulta para mostrar la lista se realizarse sobre el Schema *Pedido*.
4. La búsqueda de pedidos dentro del Schema deberá efectuarse por id.
5. Deberá mostrar los detalles de cada pedido, es decir de los atributos que se tengan.
6. Cada pedido deberá mostrar el total a pagar.
7. La vista para mostrar la lista de pedidos deberá ser implementada como una vista aparte de todas de las demás, esta vista llevará el nombre “ListaDePedidos.html”.
8. Una vez terminada la implementación de los pasos anteriores, levantará el servidor, verificará que las vistas funcionen correctamente y que los datos persistan sobre la base de datos. Modificará lo necesario en caso de presentar algún error.
9. Si no existen errores, realizar commit del proyecto en el repositorio *TiendaenJavaScript* en GitHub.

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- **Es imprescindible que todos los enlaces o sitios web que se decidan compartir con la participante, cuenten con una redacción clara y concisa, provengan de fuentes confiables, y que el contenido sea libre de violencia de género o del fomento de estereotipos.**
- Se supervisará que la información encontrada acerca de las funciones básicas con MongoDB y JavaScript, correspondan a: `save()`, `find()`, `update()` y `remove()`.
Para el uso de estas funciones y de la base de datos en general, es necesario hacer uso de Mongoose.
- La tabla elaborada en la segunda parte de la actividad, puede ser realizada a mano en una hoja de papel, o en algún procesador de texto (siempre y cuando el archivo resultante sea guardado en su carpeta de evidencias de este taller).
- La participante deberá crear dentro de su carpeta de evidencias, una nueva carpeta de nombre "MWT2A14_AgendaTelefónica". Será necesario asegurarse que dentro de ésta, estén contenidas las debidas estructuras de carpetas para los archivos a desarrollar de las dos páginas web que se solicitan en esta tercera parte de la actividad.

Notas para mejorar las habilidades de las participantes:

- Se recomienda que las participantes realicen la búsqueda de otras funciones básicas con MongoDB, distintas a las aquí trabajadas.
- Se recomienda solicitar a la participante que busque la diferencia de una base de datos SQL y una NoSQL.

Links de apoyo:

Bases de datos:

- EDteam. (2020). *¿Qué son las BASES DE DATOS y cómo funcionan?* [video]
<https://www.youtube.com/watch?v=knVwokXITGI> (Abril 2020)
- Killersites.com. (Sin fecha). *Sitios Web vinculados a bases de datos ¿Qué son y cómo se construyen?*
<https://www.killersites.com/translations/spanish/sitiosWebVinculadosAbasesDeDatos.html> (Abril 2020)
- Bitech Studio. (2018). *¿Qué es MongoDB? Breve explicación animada.* [video]
<https://www.youtube.com/watch?v=DM4gD6Z5zFU> (Abril 2020)

Configuración de la base de datos desde NodeJS

- codigofacilito.(2016). 17.- *Curso NodeJS - Conexión a MongoDB*. [video]
<https://www.youtube.com/watch?v=AbELfRULn1U> (Abril 2020)
- Fazt Code. (2020). *Nodejs & Mongodb CRUD | 4, Bases De Datos Mongodb (Mongoose)*. [video]
<https://www.youtube.com/watch?v=AknTRNvX9rA> (Abril 2020)
- Solid GEAR S.L. (2017). *Node JS - Integración con MongoDB*. [video]
<https://www.youtube.com/watch?v=g0UYD4sXot4> (Abril 2020)
- Kathleen, Juell. (2020). *Cómo instalar MongoDB con su aplicación de Node*.
<https://www.digitalocean.com/community/tutorials/how-to-integrate-mongodb-with-your-node-application-es>

(Abril 2020)

- Mariños, J. (Sin Fecha). *Cómo conectar MongoDB en Node.js*.
<https://devcode.la/tutoriales/como-utilizar-mongodb-en-nodejs/> (Abril 2020)

CRUD con MongoDB

- Azuastre, C. (2016). *Cómo almacenar datos en MongoDB con HTTP POST en tu API | Curso NodeJS y MongoDB #08*. [video]
<https://www.youtube.com/watch?v=2246l6G-1GI> (Abril 2020)
 - *Cómo devolver datos con GET desde un API REST | Curso NodeJS y MongoDB #9*. [video]
<https://www.youtube.com/watch?v=IQR6sZhoZ24> (Abril 2020)
 - *Cómo implementar DELETE y UPDATE de registros en tu API REST | Curso de NodeJS y MongoDB #10*. [video]
<https://www.youtube.com/watch?v=CjOMThGjiMo> (Abril 2020)

Páginas complementarias:

- Munro, J. (2017). *Una introducción a Mongoose para MongoDB y Node.js*.
<https://code.tutsplus.com/es/articles/an-introduction-to-mongoose-for-mongodb-and-nodejs--cms-29527> (Abril 2020)

Taller 3: Dos versiones de una tienda

En honor a **Mercè Castells Batlló**. A partir de 1999 inició el proyecto Adhocnet, SL, empresa dedicada al desarrollo de proyectos web, marketing online y social media. Durante este proceso, decidió crear “empreendedoras.com” un lugar vigente de encuentro virtual para mujeres que desean conformar sus propios negocios. Como parte de su emprendedurismo y conformación de autonomía económica, actualmente se dedica a la cosmética natural, elaboración de jabones artesanos y aromaterapia. Imparte cursos virtuales sobre el tema en “aulanatural.com” y comparte información en el blog “aula-natural.com”.

La semblanza de Mercè Castells Batlló es un referente de conformación de autonomía, pues en su ejercicio profesional se observa que ha conjugado sus gustos trasladándolos a lo público, en ese sentido, se vuelven políticos, pues pocas veces las mujeres por condición de género se atreven a mostrarse en lo público. Uno de los obstáculos para lograrlo es el miedo.

Competencia del taller: Programar y diseñar dos puntos de venta en internet usando los frameworks Sails (Javascript) y Laravel (PHP) que pueda ser visto en la web.

Actitudes: Curiosidad, disposición, constancia, persistencia, trabajo individual y en equipo, capacidad de análisis, apertura al diálogo, escucha, trabajo en equipo, intercambio de opiniones, participación activa, apertura a incorporar en las actividades la Perspectiva de Género para el logro de un bien común.

Actividad 1: Plataforma Sails

Aprendizaje esperado: Instalar la plataforma de desarrollo Sails y describir sus partes.		Duración de la actividad: 4 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
1. Node.js 2. Express.js 3. Sails.js 4. Cualquier herramienta/material para elaborar mapas cognitivos de cajas.	Evidencia: <ul style="list-style-type: none"> ● Mapa de cajas sobre búsqueda de <i>Framework</i> y <i>Sails</i>. ● Conclusiones a las que se llegaron a partir de la discusión sobre marcos de trabajo Sails vs. Express. ● Posibles modificaciones del proyecto de tienda en línea trabajada en el taller dos de un marco de trabajo Express a uno Sails. Producto: <ul style="list-style-type: none"> ● Crear un proyecto default generado por el Framework Sails. 	Retroalimentación: <ul style="list-style-type: none"> ● Se deberá observar la síntesis de la información recabada en la búsqueda, organización de la misma por preguntas y las participantes lo deberán escribir con sus propias palabras de manera completa y concreta. ● Las conclusiones hechas por las participantes sólo serán una muestra de introducción al tema, sin embargo, se podrán realizar observaciones y comentarios generales acerca de la importancia y diferencias de trabajar en un <i>framework</i> distinto y específicamente en <i>Sails</i>. ● Revisar que las modificaciones basadas en el proyecto anterior en un nuevo framework sean concordantes a Sails. Evaluación: <ul style="list-style-type: none"> ● Se debe verificar que el proyecto se ejecute dentro de un entorno cliente-servidor.

Desarrollo de la actividad:

Primera parte: ¿Sails?

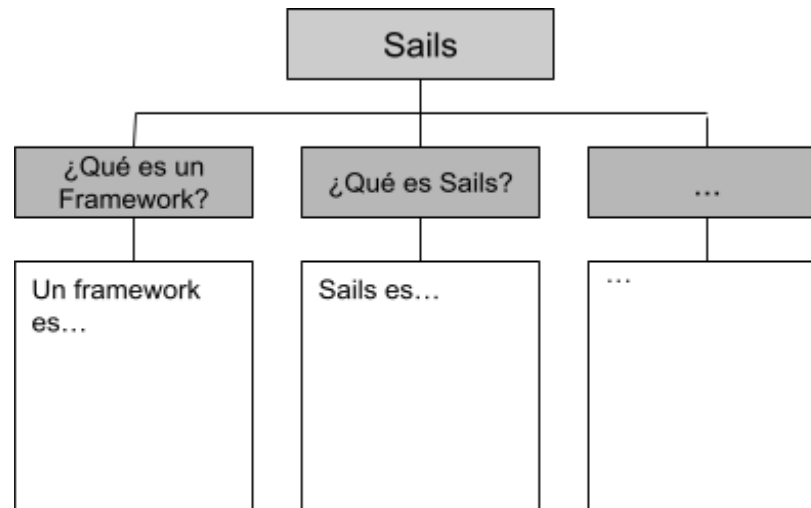
Sugerencia de tiempo invertido: 2 horas.

La participante realizará una búsqueda inicial acerca del concepto *framework* y el marco de trabajo *Sails*, sobre el cual trabajará en las próximas actividades, se sugiere consultar la documentación del mismo para dar respuesta a las siguientes preguntas:

- ¿Qué es un *framework*?
- ¿Qué es *Sails*?
- ¿Qué tareas hace fáciles *Sails*?
- ¿Qué dependencias tiene *Sails*?
- ¿Qué se necesita para poder ejecutar *Sails*?
- ¿Qué es una *Web App*?
- Los comandos necesarios para:
 - Instalar *Sails*.
 - Crear un nuevo proyecto.
 - Levantar el servidor de *Sails*.

A partir de la información recabada, se realizará un mapa de cajas, como se muestra en el ejemplo a continuación.

Ejemplo:



Segunda parte: Mi proyecto en Sails.

Sugerencia de tiempo invertido: 1 hora.

A partir de lo revisado anteriormente, las participantes completarán las siguientes tareas:

1. Abrir la consola de elección y crear un nuevo proyecto con nombre “*TiendaEnLineaSails*” en la ubicación de su preferencia, el proyecto será de tipo *Web App*.
2. Posicionarse en la carpeta donde se ubica el proyecto.
3. Levantar el servidor, y abrir en el navegador la dirección donde se visualizará la página de inicio del proyecto.
4. Navegar a través de las distintas partes del proyecto generado por default. (En este punto, sólo observar el directorio de carpetas y sus archivos).
5. Crear un nuevo repositorio en *GitHub* y subir el proyecto.

Tercera parte: Express vs Sails.

Sugerencia de tiempo invertido: 1 hora.

La participante discutirá en pareja (ya sea de participante a participante, si es que se prestan las condiciones, o bien, de tallerista a participante), la respuesta a las siguientes preguntas:

- ¿Diferencias que identifican entre Sails y Express?
 - ¿Por qué se trabaja sobre el framework Sails?
 - ¿Qué les proporciona Sails como herramienta de trabajo, que Express no tiene?
 - ¿Cómo modificarían su tienda en línea realizada en el Taller 2 para trabajar con este framework?
1. Al finalizar la discusión, la o las participantes podrán retomar el mapa de cajas realizado en la primera parte de esta actividad para escribir las conclusiones a las que llegaron.
 2. Por otro lado, revisarán el proyecto de tienda en línea realizado en el taller dos y comentarán las posibles modificaciones que harían, las conclusiones a las que lleguen podrán anotarse en el mismo mapa de cajas de la primera parte de esta actividad, es importante aclarar que fungirá como material para definir el *frontend*.

Una vez finalizado cada segmento, guardar el mapa de cajas digitalizado con sus respectivas partes en una carpeta nombrada “MWT3A1_Framework”, la cual será subida a la carpeta de evidencias en la nube.

Notas para apoyar la actividad:

- Se recomienda utilizar la consola de *Visual Studio Code* para mayor comodidad, sin embargo, la participante podrá usar la consola con la que más cómoda se sienta.

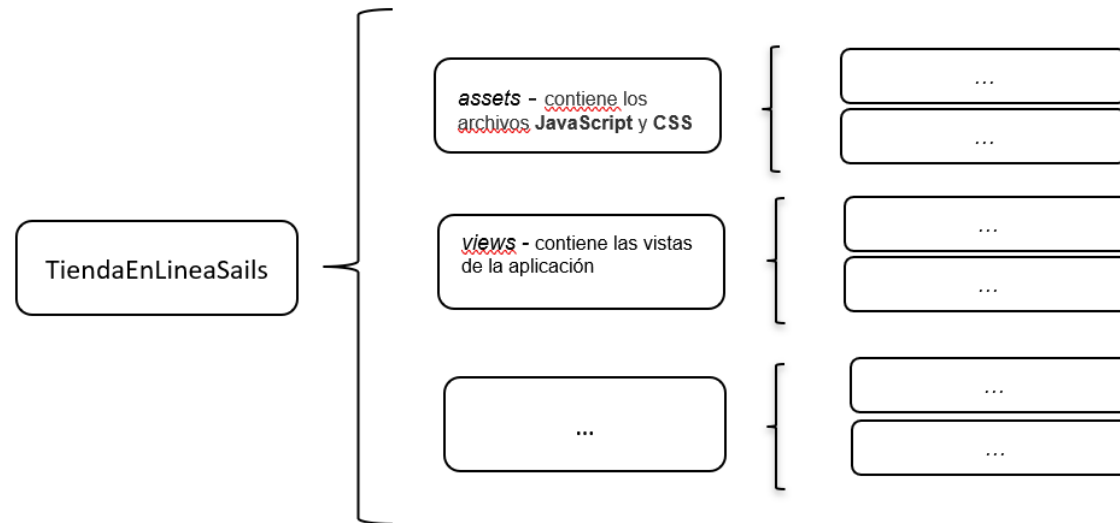
Links de apoyo:

- Recursos para elaborar organizadores gráficos
Pimienta, J. (2012). *Estrategias de enseñanza-aprendizaje. Docencia universitaria basada en competencias*. México. Pearson.
http://prepajocotepec.sems.udg.mx/sites/default/files/estrategias_pimiento_0.pdf. (Abril 2020)
- Documentación de Sails [Pagina principal]
sails v1x. (Sin fecha). *The MVC Framework for Node.js*.
<https://sailsjs.com> (Marzo 2020)

Actividad 2: Anatomía de una aplicación en Sails

Aprendizaje esperado: Resumir y enumerar los elementos de una aplicación Sails.		Duración de la actividad: 4 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
1. <i>TiendaEnLineaSails</i> , proyecto default generado en el framework <i>Sails</i>	Evidencia: <ul style="list-style-type: none"> ● Cuadro sinóptico con la jerarquía de carpetas y archivos del proyecto. ● Mapa cognitivo de cajas que describa el <i>frontend</i> de su proyecto. ● <i>Mockup</i> de la página web. Producto: <ul style="list-style-type: none"> ● Basados en el proyecto <i>TiendaEnLineaSails</i>, generar la misma aplicación pero traducida al idioma español. ● Generar el diseño del sistema web para una tienda en línea. 	Retroalimentación: <ul style="list-style-type: none"> ● Corroborar que el cuadro sinóptico abarque las carpetas y archivos esenciales del proyecto y que las descripciones correspondan a la carpeta o archivo que se describe. ● El mapa cognitivo de cajas deberá tocar cada uno de los temas y exponerlos de forma breve. ● Revisar que el mockup abarque los elementos del front de la página web. Evaluación: Se debe verificar que el proyecto se ejecute dentro de un entorno cliente-servidor. <ul style="list-style-type: none"> ● El diseño generado deberá de implementar la arquitectura modelo vista controlador (MVC).
Desarrollo de la actividad:		
Primera parte: Navego con Sails. Sugerencia de tiempo invertido: 2 horas. 1. La participante hará una revisión detallada sobre el directorio de carpetas creadas con Sails y realizará un esquema tipo cuadro sinóptico, con cualquier medio que sea elegido por la participante, donde plasmará la jerarquía de carpetas y archivos del proyecto " <i>TiendaEnLineaSails</i> " en cada recuadro escribirá el nombre del archivo o carpeta incluyendo una breve descripción de contenido o función, según sea el caso.		

Ejemplo:



2. En un mapa cognitivo de cajas la participante describirá los elementos del *frontend* del proyecto “TiendaEnLineaSails”. Se deberán contemplar los siguientes puntos:

- Página principal
- Menú
- Login
- Signup
- Recuperación de password
- FAQ
- Banners

3. La participante elaborará un *mockup* como primer borrador para el diseño de las páginas descritas.

Una vez finalizado este segmento, en una carpeta nombrada “MWT3A2_NavegoSails” añadir las versiones digitales del cuadro sinóptico, el mapa cognitivo de cajas y el mockup. Dicha carpeta será añadida a la carpeta de evidencias en la nube.

Segunda parte: Sails políglota.

Sugerencia de tiempo invertido: 2 horas.

La participante codificará la configuración de los idiomas que va a soportar el proyecto “TiendaSails” por medio de la internacionalización *i18n*. Se deberá tomar en cuenta lo siguiente:

1. Modificar los archivos necesarios para traducir la app a Español, tomar en cuenta los archivos de las siguientes rutas:
 - a. `config/i18n.js`
 - b. `config/locales/es.json`
2. Visualizar la página con las modificaciones hechas en la dirección del servidor por default en el navegador.
3. Subir las modificaciones realizadas al repositorio de GitHub donde se encuentra el proyecto.

Notas para apoyar la actividad:

- La participante podrá auxiliarse de la documentación disponible en la página de Sails, o de cualquier otro recurso para realizar las modificaciones.

Links de apoyo:

- Recursos para elaborar organizadores gráficos
- Pimienta, J. (2012). *Estrategias de enseñanza-aprendizaje. Docencia universitaria basada en competencias*. México. Pearson. http://prepajocotepec.sems.udg.mx/sites/default/files/estrategias_pimiento_0.pdf. (Abril 2020)
- Páginas de la documentación de Sails que les ayudarán a configurar la internacionalización.
- sails v1x. (Sin fecha).. *sails.config.i18n*
<https://sailsjs.com/documentation/reference/configuration/sails-config-i-18-n> (Marzo 2020)
 - *Internationalization*.
<https://sailsjs.com/documentation/concepts/internationalization> (Marzo 2020)
 - *Locales*.
<https://sailsjs.com/documentation/concepts/internationalization/locales> (Marzo 2020)

Actividad 3: Conectando con la base de datos.

Aprendizaje esperado: Conectar Sails con base de datos no estructurada en la nube.		Duración de la actividad: 6 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
1. Tener una cuenta en un sistema de bases de datos no estructuradas (MongoDB) en la nube. 2. Mongo DB instalado en la computadora. 3. Plantillas MWT3A3_Anexos.pdf 4. Cualquier herramienta/material para elaborar un mapa cognitivo de cajas.	Evidencia: <ul style="list-style-type: none"> • Mapa cognitivo de cajas sobre bases de datos. • Plantillas MWT3A3_Anexos respondidas. Producto: <ul style="list-style-type: none"> • Basado en la actividad 2 conectar la aplicación con el servicio de BD en línea. • Generar el diseño de los modelos requeridos para la tienda en línea (producto, compra, carrito). 	Retroalimentación: <ul style="list-style-type: none"> • El mapa deberá sintetizar información relacionada con la búsqueda a partir de las preguntas guía en la primera parte de esta actividad en forma concreta y completa. • El llenado de las plantillas deberá cumplir con lo requerido en la tercera parte de esta actividad. Evaluación: <ul style="list-style-type: none"> • La conexión de BD deberá funcionar sin ningún error. • El diseño generado de los modelos debe estar basado en la arquitectura MVC.
Desarrollo de la actividad:		
Primera parte: Aprendo averiguando. Sugerencia de tiempo invertido: 2 horas. 1. La participante realizará una búsqueda guiándose de las siguientes preguntas: <ul style="list-style-type: none"> • ¿Qué es un diagrama entidad-relación? • ¿Qué elementos componen un diagrama entidad-relación? • ¿Cómo identificar las relaciones para este diagrama? • ¿Cómo es una base de datos no estructurada? • ¿Qué ventajas hay al usar una base de datos en la nube? • ¿Qué es Mongo Atlas? y ¿Cuáles son sus características? 		

- ¿Qué es ORM? y ¿Cuáles son sus características?
- ¿Qué ORM utiliza Sails?
- ¿Qué es un Adaptador de Base de Datos?
- ¿Cómo se construye un adaptador?

2. A partir de la búsqueda realizada se sugiere elaborar un mapa cognitivo de cajas para organizar la información utilizando cualquier medio que desee, este fungirá como material de consulta para el futuro. Una vez finalizado este segmento, creará una carpeta con nombre “*MWT3A3_BaseDeDatos*” que contendrá el mapa de cajas digitalizado y esta se subirá a la carpeta de evidencias de la nube.

Segunda parte: Mi base de datos.

Sugerencia de tiempo invertido: 2 horas.

La participante realizará las siguientes tareas, para levantar el *cluster* en MongoDB, crear su primer usuario y conectar la base de datos a su proyecto de Tienda en línea con Sails.

1. Crear una cuenta en MongoDB Atlas.
2. Una vez iniciada la sesión, hacer click en *Build a Cluster*.
3. Realizar las configuraciones de las secciones *Cloud Provider & Region*, *Cluster Tier*, y *Additional Settings*.
4. Poner un nombre al *cluster* y levantarlo.
5. Configurar la seguridad de conexión, en “*Network Access*” y añadir en “*IP WhiteList*” las IP’s autorizadas para conectarse al *cluster*.
6. Añadir el primer usuario Administrador en “*DataBase Access*”, seleccionar “*Atlas admin*” y ponerle un nombre y password que deseen, para este último se podrá hacer uso del autogenerado del password, para tener una contraseña más segura.
7. Dar click en “*Connect*” y luego en “*Connect Your Application*”, seleccionar el controlador *Node.js* y la versión que están usando.
8. Copiar la cadena que se genera, la cual servirá para conectar la base de datos.
9. Modificar el archivo *datastore.js* que se encuentra en la carpeta *config* de su proyecto, ahí escribirá el código del adaptador correspondiente a la Base de Datos de MongoDB, en el adaptador se pegará la cadena como *url* de conexión.
10. Escribir el código necesario para encriptar los *passwords* de los usuarios, de modo que al hacer consultas a la base de datos no se muestre el *password* original.

Una vez hecha la conexión de la base de datos, realizar las pruebas necesarias para verificar que esté funcionando el proyecto. Luego, deberá subir las modificaciones realizadas al repositorio de GitHub donde se encuentra el proyecto.

Tercera parte: Comienzo a diseñar.

Sugerencia de tiempo invertido: 2 horas.

La participante creará un diagrama entidad-relación para entender las relaciones entre los objetos o entidades de su tienda en línea (se recomienda utilizar el anexo *MWT3A3_Anexos*). Para ello la participante:

1. Enlistará los objetos o entidades que contiene su tienda en línea.
2. Responderá las siguientes interrogantes con las cantidades mínimas y máximas solicitadas. Estas deberán estar representadas entre paréntesis, separadas por medio de dos puntos (mínimo:máximo). Ejemplo: (1:10). De tratarse de una cantidad máxima irrepresentable, es posible utilizar el símbolo asterisco. Ejemplo: (1:*).
 - a. Un **producto**, ¿En cuántas **listas de productos** puede aparecer? Respuesta= (min:max)
 - b. Una **lista de productos**, ¿Cuántos **productos** puede contener? Respuesta= (min:max)
 - c. Un **producto**, ¿Cuántas veces puede solicitarse en un **pedido**? Respuesta= (min:max)
 - d. En un **pedido**, ¿Cuántos **productos** se pueden solicitar? Respuesta= (min:max)
 - e. Un **pedido** ¿Cuántas veces se ingresa en una **lista de pedidos**? Respuesta= (min:max)
 - f. En una **lista de pedidos** ¿cuántos **pedidos** se solicitan? Respuesta= (min:max)
3. Las preguntas anteriores se agrupan en tres pares, ya que cada par corresponde a una misma relación de entidades. Se indicará a la participante que identifique los tres pares de preguntas, así como las dos entidades que conforman cada relación, y el verbo que las conecta (se recomienda utilizar la plantilla *MWT3A3_Anexos*).
4. Finalmente, con la plantilla 4 de los anexos *MWT3A3_Anexos* la participante deberá:
 - a. Unir con una línea los objetos que tengan una relación entre sí.
 - b. Anotar el mínimo y el máximo que respondieron en las preguntas en los paréntesis que correspondan de su lógica.
 - c. Anotar en el rombo los verbos que relacionan a ambos objetos.

Una vez finalizado el segmento, guardar la plantilla digitalizada en la carpeta de evidencias “*MWT3A3_BaseDeDatos*” en la nube.

Notas para apoyar la actividad:

- Se deberá contemplar que la secuencia de pasos para levantar un *cluster* en MongoDB podría cambiar en el futuro, se recomienda corroborar la secuencia de tareas antes de asignarla a las participantes, en dado caso de que varíe, realizar los cambios correspondientes.
- En caso de que la participante se le dificulte realizar las tareas, deberá leer la documentación de MongoDB Atlas, y de Sails JS.

Links de apoyo:

- Mongo DB. (Sin fecha). *Mongo DB Atlas*.
<https://www.mongodb.com/cloud/atlas> (Marzo 2020)
- Documentación de *MongoDB Atlas*
<https://docs.atlas.mongodb.com/> (Marzo 2020)
- Documentación de Sails. [Pagina principal.
sails v1x. (Sin fecha). *The MVC Framework for Node.js*.
<https://sailsjs.com> (Marzo 2020)

Actividad 4: Agregando productos.

Aprendizaje esperado: Manipular un modelo para la figura de producto.		Duración de la actividad: 6 horas
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Diagrama entidad relación realizado en la actividad MWT3A3. 2. Cualquier material o herramienta para elaborar un organizador gráfico. 	<p>Evidencia:</p> <ul style="list-style-type: none"> ● Organizador gráfico a elección digitalizado y guardado en la carpeta MWT3A4_Modelos. <p>Producto:</p> <ul style="list-style-type: none"> ● El proyecto <i>TiendaEnLineaSails</i> deberá contener el modelo, vistas y controladores necesarios para hacer un CRUD de los elementos básicos de una tienda en línea, que son: producto, compra y carrito. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> ● El organizador gráfico será de libre elección, la información que contenga tendrá que ser concreta y completa. <p>Evaluación:</p> <ul style="list-style-type: none"> ● Se debe verificar que se puedan realizar operaciones CRUD con los modelos generados y que la información quede almacenada en la base de datos.
Desarrollo de la actividad:		
<p>Primera parte: ¿Cómo le doy forma a mi tienda?</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <ol style="list-style-type: none"> 1. La participante realizará una búsqueda de información apoyándose de las siguientes preguntas: <ul style="list-style-type: none"> ● ¿Qué es un modelo? y ¿para qué sirve? ● ¿Con qué comando se crea un modelo y un controlador en Sails? ● ¿Cómo se define un modelo en Sails? ● ¿Qué es un atributo? ● ¿Cómo se definen los atributos de un modelo en Sails? ● ¿Qué es una asociación? ● ¿Cómo crear asociaciones entre modelos en Sails? (<i>one to one, one to many, many to many</i>) ● ¿Qué es un controlador? y ¿para qué sirve? ● ¿Cómo se define un controlador en Sails? 2. La participante organizará la información recabada en el organizador gráfico que desee, esté fungirá como material de consulta 		

para el futuro, y se guardará una versión digitalizada en una carpeta nombrada “MWT3A4_Modelos”, dicha carpeta será subida a la carpeta de evidencias en la nube.

Segunda parte: Dando forma a mi tienda.

Sugerencia de tiempo invertido: 4 horas.

La participante creará los modelos y controladores de Producto, Carrito y Compra. Tomará como base el diagrama entidad-relación realizado en la actividad MWT3A3. La participante realizará lo siguiente:

1. Con los comandos revisados, crear los modelos y controladores desde consola.
2. Revisar que en la carpeta *api/models* se encuentren los archivos *js* de cada modelo.
3. Escribir el código necesario para definir cada modelo en su archivo correspondiente, considerar sus atributos y asociaciones.
4. Revisar que en la carpeta *api/controllers* se encuentren los archivos *js* de cada modelo.
5. Escribir el código necesario para definir las acciones que atenderán las peticiones HTTP: *find, create, update, destroy, populate, add* y *remove* para cada modelo.

Una vez hechas las modificaciones en el proyecto Tienda en línea con Sails, utilizar la herramienta **RESTED**, extensión para el navegador Mozilla Firefox, o **POSTMAN** para Google Chrome, para agregar productos a la base de datos con peticiones HTTP.

Al terminar la actividad y que se haya verificado la funcionalidad del proyecto, subir las modificaciones realizadas al repositorio de GitHub donde se encuentra el proyecto.

Notas para apoyar la actividad:

- La tallerista podrá responder a dudas o bien, se podrán realizar búsquedas en internet.
- En caso de que la participante se le dificulte realizar las tareas, deberá leer la documentación de Sails JS, o cualquier otro recurso.

Links de apoyo:

- Extensión RESTED.
Firefox Browser ADD-ONS. (2019). </> RESTED por Espen H
<https://addons.mozilla.org/es/firefox/addon/rested/?src=search> (Marzo 2020)
- Extensión POSTMAN.
chrome web store. (Sin fecha). *Postman*.

<https://chrome.google.com/webstore/detail/postman/fhbjgbiflinjbdggehcddcbncdddomop?hl=es-419> (Marzo 2020)

- Documentación de Sails. [Pagina principal]
sails v1x. (Sin fecha). *The MVC Framework for Node.js*.
<https://sailsjs.com> (Marzo 2020)

Actividad 5: Rutas hacia los productos

Aprendizaje esperado: Agregar rutas y vistas asociadas a sistema web.		Duración de la actividad: 9 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
	Evidencia: <ul style="list-style-type: none"> • Juego interactivo online. Producto: <ul style="list-style-type: none"> • A partir de la primera parte de la actividad 4, el sistema deberá poder redireccionar a cada una de las vistas necesarias para Listar, Actualizar, Agregar y Eliminar los elementos de cada uno de los modelos generados. 	Retroalimentación: <ul style="list-style-type: none"> • El juego deberá presentar información sobre los contenidos. Evaluación: <ul style="list-style-type: none"> • Verificar que se incluyan las vistas para listar, actualizar, agregar, eliminar los elementos del modelo.
Desarrollo de la actividad:		
<p>Primera parte: Vistas, rutas y visualización.</p> <p>Sugerencia de tiempo invertido: 6 horas.</p> <ol style="list-style-type: none"> Se le solicitará a la participante realizar una búsqueda en internet que responda a las siguientes preguntas. <ul style="list-style-type: none"> Tema: Agregar vistas para los productos <ol style="list-style-type: none"> ¿Qué son las vistas en una página web? ¿Qué son las Vistas en Sails? Mencionar los cuatro tipos de vistas y para qué sirve o se utiliza cada una. Tema: Agregar rutas para cada uno de los productos. <ol style="list-style-type: none"> ¿Qué es una ruta? ¿Qué tipos existen? ¿Cuál es su sintaxis? Tema: Visualización de productos <ol style="list-style-type: none"> ¿Qué es Bootstrap? ¿Cómo funciona en el diseño web? ¿Qué tareas facilita? 		

- d. ¿Cómo se instala?
- e. ¿Cuál es la compatibilidad entre html, css y Bootstrap?

Tema: Aplicar Bootstrap 4 a las vistas generadas y al sistema Web.

- a. ¿Cómo se implementa Embedded javascript para las vistas (EJS)?
- b. ¿Qué es embedded javascript?
- c. ¿Cuál es su sintaxis?
- d. ¿Cuál es el uso que se le da?

2. Una vez concluida la investigación, se solicitará que cree un juego (con base en los temas investigados) en la plataforma Cerebriti. Existen varios tipos de juegos, la participante podrá elegir el que más le parezca, ello con el objetivo de que jerarquice el conocimiento adquirido y pueda presentarlo de manera lógica en un producto lúdico.

3. Al terminar, tendrá que proporcionar el enlace del juego a la tallerista para que esta lo juegue y valore si los contenidos se incorporaron de manera precisa en el mismo.

Segunda parte: Creando rutas

Sugerencia de tiempo invertido: 3 horas.

Creación de rutas en un sistema web usando Sails:

1. Para esto se creará el archivo ***config/routes.js***, en el cual se definirán las rutas de aplicación, este archivo se usará para agregar nuevos controladores a la aplicación.
2. Para la realización de esta parte, deberá cerciorarse que se establezca la conexión a la base de datos previamente creada en la Actividad 3.
3. En el proyecto actual, se agregarán vistas y rutas para listar, actualizar, agregar y eliminar cada uno de los modelos que integran la base de datos:
 - Creará un archivo de controlador vacío.
 - En el nuevo archivo de controlador ***api/controllers/nombreDelMetodo.js***.
 - Posteriormente se creará un nuevo archivo de vista en la carpeta *views* llamada ***NombreDelMetodo.ejs*** con el html para la vista.

- Por último, se definirá una ruta en **config/routes.js** que llame al método *NombreDelMetodo* en el controlador *Controller*.

Creación de visualizaciones:

1. Se deberá crear la carpeta *Views*, en la cual se almacenarán las vistas de la aplicación.

Nota: En cuanto al contenido, las opciones por defecto generan el código en el formato (JavaScript incrustado) **EJS**. Además, la carpeta contiene vistas para el manejo de errores así como el archivo de diseño (**layout.ejs**) y los puntos de vista para el controlador. Cada vez que se crea un nuevo controlador, por ejemplo *UserController*, se creará una carpeta con el mismo nombre del modelo; y dentro las diferentes vistas como *index.ejs*, *show.ejs*, *edit.ejs*, *destroy.ejs*, *associations.ejs*, etcétera. Cada una de las vistas, deberá introducir el html y código para cargar información de la base de datos.

2. En la carpeta *Views* se creará otra carpeta con nombre el **static**.

3. Dentro de esta se creará el fichero **index.ejs**, en él se escribirá el siguiente código html:

```
< h1>'tiendita virtual'< /h1>
```

4. Se tendrá que acceder a la carpeta **config**, seleccionar al fichero **routes.ejs** y cambiar la vista que se mostrará al arrancar el servidor.

```
('/: { 'static/index' }
```

4. Se deberá abrir el fichero **layout.ejs**, que está dentro de la carpeta *views*, donde se encontrará una línea que pone: **<%- body -%>**. Esto lo que hará es cargar en **layout.ejs** en lo que hay en **'static/index'** y en los diferentes **ficheros .ejs**.

5. Se añadirán los archivos **css** dentro de la carpeta **assets/styles** y los ficheros **javascript** dentro de la carpeta **assets/js**; el servidor los añadirá automáticamente al fichero, sin necesidad de hacerlo manualmente.

Bootstrap:

1. Se descargarán los ficheros bootstrap necesarios para añadirlos en sus correspondientes carpetas, con el fin de tener un **index** más atractivo. (se recomienda usar los sugeridos en la bibliografía)
2. Se añadirán los ficheros en **assets/js** y en **assets/styles** según correspondan.
3. Todo el código deberá estar dentro del fichero **index.ejs** de la carpeta **static**.

Links de apoyo:

- Para la elaboración del juego

Cerebriti.(Sin fecha). *El mayor catálogo de juegos educativos y de conocimiento en español.*

<https://www.cerebriti.com/> (Enero 2021)

- Delgado, J. (2015) Tutorial de Sails.js: Instalación y primer proyecto.

<https://openwebinars.net/blog/tutorial-sailsjs-instalacion/> (Febrero 2021)

- Sails. (Sin fecha). *Routes.*

<https://sailsjs.com/documentation/concepts/routes> (Febrero 2021)

- *Views.*

<https://sailsjs.com/documentation/concepts/views> (Febrero 2021)

- *Actions and controllers.*

<https://sailsjs.com/documentation/concepts/actions-and-controllers> (Febrero 2021)

Actividad 6: Agregar Lógica

Aprendizaje esperado: Incorporar lógica de negocios para manipular carrito y venta de productos.		Duración de la actividad: 8 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Material requerido para realizar el diagrama de caso (hojas de papel y plumones, app móvil o programa en la computadora). 2. Proyecto de tienda en línea. 	<p>Evidencia: Diagrama de caso del uso UML.</p> <p>Productos: A partir de la segunda parte, el sistema deberá agregar la lógica de negocio para poder realizar una compra. A partir de la segunda parte, se deberán crear archivos para las partes comunes dentro del sitio web.</p>	<p>Retroalimentación:</p> <ul style="list-style-type: none"> ● Se podrán pedir diagramas de caso de uso UML para sustentar la lógica de negocio del sistema. <p>Evaluación Cumplir con los cambios observables en el proyecto de tienda en línea.</p>
Desarrollo de la actividad:		
<p>Primera parte: Lo que veo de mi página a través de las otras.</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <ol style="list-style-type: none"> 1. La participante navegará en distintas páginas de venta de su elección. Identificará características, diseño y uso de los diferentes menús, poniendo principal atención en cómo se despliegan los productos, en qué consiste el proceso de compra, así como la función que desempeña el carrito o bolsa de compras. 2. La participante buscará en páginas web, vídeos o infografías, las características del diagrama de caso de uso de UML. Con la información recabada en los puntos 1 y 2, elaborará un diagrama de caso de uso de UML para el proceso de compra de un producto para su proyecto, en donde será clara la identificación de los participantes y las acciones que realiza cada uno de ellos. El diagrama fungirá como un boceto previo a la implementación de la lógica de negocio en la tienda en línea. El diagrama será revisado por la tallerista en conjunto con la participante, para identificar fallos o confusiones antes de su implementación. <p>Segunda parte: Del boceto a mi página.</p> <p>Sugerencia de tiempo invertido: 6 horas.</p>		

La participante realizará los cambios correspondiente a su proyecto de tienda en línea para la inclusión del carrito de compras, para ello utilizará como guía el diagrama anteriormente elaborado y revisado en conjunto con la tallerista.

Los siguientes cambios deberán ser observables en el sitio:

- Se deberá contar con un icono que conduzca al contenido del carrito de compras.
- Se deberá añadir un botón “Agregar producto” en la vista de Productos.
- Se deberá generar una nueva vista, añadir su ruta y agregar el código correspondiente al controlador para desplegar el contenido del carrito. La vista deberá mostrar la información del producto, cálculo de totales y subtotales, un botón “Eliminar producto” para cada uno de los productos y un único botón “Comprar”.
- Cada que se presione el botón “Agregar producto”, el id del producto deberá guardarse en el modelo Carrito.
- Cada que se presione el botón “Comprar”, el id del carrito deberá guardarse en el modelo Compra.
- Con el botón “Eliminar producto” se eliminará el producto del modelo Carrito con el id indicado.
- Se deberán implementar las cuatro acciones del CRUD para Producto. Para Carrito bastará únicamente CREATE y DELETE.

Será importante que la participante identifique el lugar adecuado tanto en la vista como en el código para colocar el icono que conduzca a los elementos agregados en el carrito, ya que este debe ser accesible desde cualquier dirección de su sitio, justo como se debió identificar en la Primera parte de la actividad. De manera que la adición se verá reflejada en el archivo layout.ejs y se prestará también para la reflexión de la reutilización de código con Sails.

Notas para apoyar la actividad:

- En la primera parte: “Lo que veo de mi página a través de las otras”, la participante puede utilizar inspect/inspeccionar para visualizar los componentes de las páginas y complementar su exploración.
- En la segunda parte: “Del boceto a mi página”, la participante puede apoyarse en tutoriales, manuales y otros materiales de guía para la modificación de su tienda en línea, se recomienda sugerir que la participante busque estos materiales por su cuenta y en el caso de tener otras complicaciones o dudas que no pueda resolver, apoyarse con la tallerista.

Actividad 7: Dándole poder al administrador

Aprendizaje esperado: Incorporar un sistema de administración del sitio web.		Duración de la actividad: 10 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ul style="list-style-type: none"> Cuenta en alguna red social (Facebook, Twitter, etc.) 	<p>Evidencia:</p> <ul style="list-style-type: none"> Tres publicaciones en una red social que abarquen los temas investigados. <p>Producto:</p> <ol style="list-style-type: none"> A partir de la parte 2, se deberá crear el modelo Usuario, así como el controlador y sus correspondientes vistas para poder realizar acciones CRUD. A partir de la parte 1, se deberán integrar las directivas de restricción de vistas y acciones basadas en perfiles de usuario. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> Las publicaciones deberán tratar los temas investigados. <p>Evaluación:</p> <ul style="list-style-type: none"> Se deberá comprobar que las actualizaciones al modelo Usuario se registren en la BD. Se deberá revisar que no sólo las vistas se restrinjan, las acciones CRUD también deberán estar basadas en permisos y roles.
Desarrollo de la actividad:		
<p>Primera parte: Administración, permisos y acciones.</p> <p>Sugerencia de tiempo invertido: 4 horas.</p> <ol style="list-style-type: none"> Se le solicitará a la participante realizar una búsqueda en internet que responda a las siguientes preguntas. <p>Tema: Vistas de administración:</p> <ol style="list-style-type: none"> ¿Qué son? ¿Cómo se otorgan? ¿Por qué son importantes? <p>Tema: Permisos asociados al administrador.</p> <ol style="list-style-type: none"> ¿Qué son los permisos? ¿Por qué son necesarios? ¿Qué políticas de acceso sigue Sails? ¿Qué son los roles? 		

- e. ¿Qué hace cada uno?
- f. ¿Por qué es importante establecerlos y delimitarlos?
- g. ¿Qué es la política de roles?

Tema: Acciones CRUD

- a. ¿Qué son las acciones CRUD?
- b. ¿Cómo se implementan?

2. Una vez concluida la investigación, se solicitará que en una red social (Facebook, Twitter, etc) de su preferencia realice tres publicaciones (una para cada tema) que expliquen de manera breve lo que entendió del tema, deberá utilizar el hashtag #EscueladeCódigoSistemasWeb. Esto con el fin de abrir el diálogo de los temas con personas interesadas en el cual la participante pueda exponer los aprendizajes adquiridos y de igual manera retroalimentarse con el conocimiento de la comunidad en internet.

3. La participante podrá recurrir a la tallerista para aclarar alguna duda antes de realizar la publicación.

Nota: Puesto que en las redes sociales se puede interactuar con todo tipo de personas, se sugiere que se supervise la interacción de la participante, para evitar malos entendidos o situaciones desagradables derivadas por otro usuario.

Segunda parte: Creando roles.

Sugerencia de tiempo invertido: 4 horas.

1. Se deberán crear los roles correspondientes para asignar a los usuarios.
2. Para determinar qué pueden hacer estos grupos de usuarios (roles), se deberán asignar los *permissions*.
3. Se deberá entregar un esquema en el que se especifiquen los usuarios, roles y permisos. (Se recomienda consultar la bibliografía sugerida para la creación de este esquema).
4. Por defecto, el módulo *sails-permissions* crea dos roles *admin* y *registered*. La participante deberá hacer una lista indicando de qué es capaz cada rol y a quién se asigna cada uno. Para esto deberá:
 - Instalar ***sails-permissions*** y ***sails-auth*** (si no está previamente instalado).
 - Ejecutar el generador: ***sails generate permissions-api***
 - Configurar Usuario administrador: se deberán establecer las variables de entorno: ***ADMIN_USERNAME***, ***ADMIN_EMAIL*** y ***ADMIN_PASSWORD***.

Nota: Para fines de desarrollo, se puede configurar el usuario administrador por *default* (consultar bibliografía de apoyo).

- Ejecutar y verificar que se hayan creado los roles correspondientes.

Tercera parte: Asignando permisos.

Sugerencia de tiempo invertido: 2 horas.

1. Se deberán investigar los siguientes atributos: *Model, action, relation, role, user, criteria* y posteriormente ocupar cada uno de estos para el rol de administrador y otorgarle permisos.

- A partir de la segunda parte se deberá crear el modelo Usuario, así como el controlador y sus correspondientes vistas para poder realizar acciones CRUD.
- La participante debe restringir las vistas para que las páginas de usuarios sólo las vean usuarios que estén logueados, y las páginas de administrador que estén disponibles sólo para administradores.

Actividad 8: Mejorando la webapp

Aprendizaje esperado: Incorporar normas de diseño e interacción <i>User Experience UX a la Web App.</i>		Duración de la actividad: 7 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
	Evidencia: <ul style="list-style-type: none">• A partir de la parte dos de la actividad se deberán integrar directivas de UX a las vistas para los modelos Producto, Carrito, Compra.	Retroalimentación: <ul style="list-style-type: none">• Considerar que las participantes podrían usar frameworks de diseño especializado en UX.• Se sugerirá hacer uso de librerías de JavaScript para realizar esta tarea.
Desarrollo de la actividad:		
<p>Primera parte: La UX.</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <p>1. Se le solicitará a la participante realizar una búsqueda en internet que responda a las siguientes preguntas.</p> <ol style="list-style-type: none">¿Qué es UX (User Experience)?¿Cuáles son las diferencias entre, diseño de interfaces, diseño de interacción y experiencia de usuario?¿Cuáles son las bibliotecas de JavaScript para diseño web?¿Qué es React y por qué debería usarse en este proyecto?¿Qué es el diseño de material Bootstrap? <p>2. Se recomienda realizar un organizador gráfico para depositar los contenidos que se obtengan de la búsqueda. Estos pueden ser cuadros comparativos, mapas conceptuales, cuadros sinópticos, entre otros. La participante puede elegir el que le sea más práctico para ella.</p> <p>Segunda parte: Poniéndose las botas.</p> <p>Sugerencia de tiempo invertido: 2 horas.</p>		

1. Se visitarán tres páginas de su elección para destacar los elementos que les llamen la atención (colores, botones, figuras, vistas de la página en general) con el objetivo de que puedan ir planeando el diseño propio de su tienda.
2. Deberá de comparar las librerías javascript y el material de diseño de bootstrap.
3. Al finalizar la participante deberá seleccionar las bibliotecas de JavaScript que va a usar para su proyecto o elegir un kit de diseño de material bootstrap que se adecue al diseño que quieren realizar.

Tercera parte: Agregando diseño a nuestra página.

Sugerencia de tiempo invertido: 3 horas.

1. La participante deberá reutilizar el código de la Actividad 6 puesto que se dará diseño a los botones y lo previamente diseñado.
2. Se deberá buscar la documentación y código de ejemplo de los siguientes elementos para poder añadir su diseño a la página web:
 - Página principal de la tienda: en esta se modificarán las vistas para producto, carrito y compra sin modificar su ruta.
-Para ello se deberá crear un icono que represente cada objeto (catálogo, productos, carrito).
 - Botones: se modificarán los botones agregar producto, eliminar producto y comprar
 - Carrito: se modificará la vista del contenido del carrito para que despliegue los productos que ha agregado el cliente en forma de lista.
3. Se seleccionará una plantilla, que será la base de su diseño.
4. Se creará un directorio en la raíz de documentos del web server.
5. Se deberá extraer el archivo zip con la plantilla elegida de Bootstrap o javascript y copiar el contenido en el directorio recién creado.
6. Una vez copiado el contenido, se realizará una prueba para confirmar que todo funciona.
7. Se incorporarán los materiales de diseño a la plantilla seleccionada.
8. Del material buscado en la segunda parte de la actividad, deberá descargar uno y extraer el contenido correspondiente a *material design*.
9. Se copiarán los archivos *"material-design.css"* y *"ripples.css"* de la carpeta *"dist/css"* y se colocarán en el directorio css del proyecto.
10. Posteriormente se copiarán los archivos *material.js* y *ripples.js* de la carpeta *dist/js* y se colocarán en el directorio js del proyecto.
11. Realizar las pruebas necesarias para asegurarse que se visualiza el diseño.

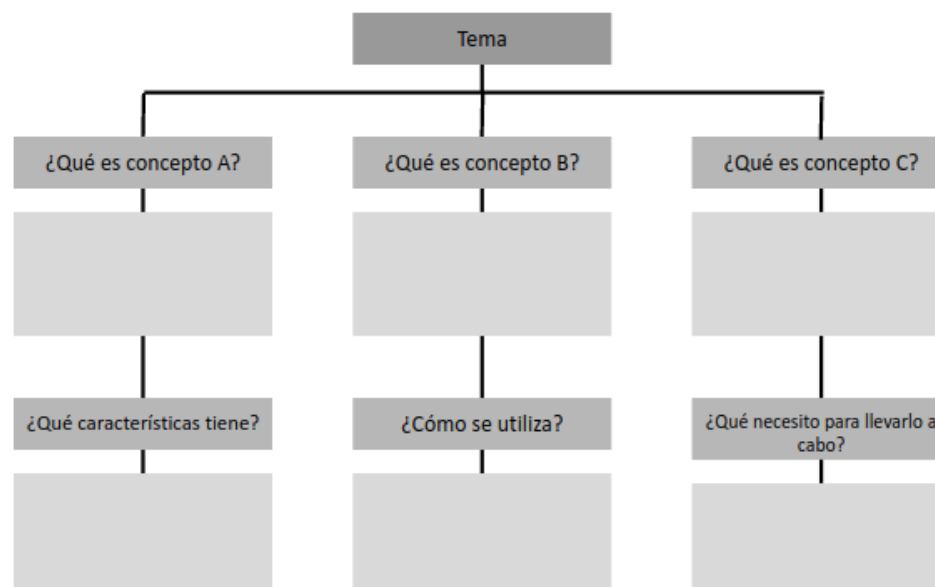
Nota para apoyar la actividad:

- Para que *Material design* pueda utilizarse en el proyecto se deben incluir las líneas de código necesarias en el archivo **index.html** (colocar los links hacia los archivos CSS en el <head> del archivo index.html).
- Si se utilizará bootstrap, se recomienda instalar *BootstrapStudio*.

Actividad 9: Plataforma PHP Laravel

Aprendizaje esperado: Crear una página básica PHP.		Duración de la actividad: 5 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Laravel. 2. Composer. 3. NPM. 4. PHP >= 7.2.5 5. BCMath PHP Extension 6. Ctype PHP Extension 7. Fileinfo PHP extension 8. JSON PHP Extension 9. Mbstring PHP Extension 10. OpenSSL PHP Extension 11. PDO PHP Extension 12. Tokenizer PHP Extension 13. XML PHP Extension 14. Hojas de papel o papel reciclado. 15. Lápices, plumas o plumones. 	<p>Evidencia:</p> <ul style="list-style-type: none"> • Un mapa cognitivo de cajas. <p>Producto:</p> <ul style="list-style-type: none"> • Proyecto <i>MiTiendaEnPhp</i> 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • El mapa cognitivo de cajas deberá conjuntar la información correspondiente a las preguntas de la primera parte de esta actividad. • Será necesario que esté elaborado de acuerdo al formato que se especifica en la parte 2 de la actividad. • Deberá estar almacenado en la carpeta de evidencias de este taller. En caso de haberse realizado de forma manual, será necesario digitalizar el documento para guardarlo en la carpeta de evidencias. <p>Evaluación:</p> <ul style="list-style-type: none"> • El nuevo proyecto en Laravel y PHP deberá: contar un login, incorporar blade y bootstrap para el desarrollo de las páginas de su tienda, y ejecutar commit en el repositorio <i>TiendaEnLaravel</i> del proyecto en Github.
Desarrollo de la actividad:		
<p>Primera parte: Conociendo lo básico.</p> <p>Sugerencia de tiempo invertido: 1 hora y 30 minutos.</p> <p>Las participantes tendrán un primer acercamiento a PHP y a su framework Laravel.</p> <ol style="list-style-type: none"> 1. Se solicitará la búsqueda de información necesaria para responder las siguientes interrogantes: 		

- a. ¿Qué es PHP y qué características tiene?
 - b. ¿Qué es Laravel y cuál es el comando para crear un proyecto?
 - c. ¿Qué es LAMP y por qué lo necesito para trabajar con Laravel?
 - d. ¿Qué es Composer, composer install, composer update?
 - e. ¿Qué es Artisan y cómo se levanta un servidor local?
 - f. ¿Qué es un blade en Laravel y cómo se utiliza?
 - g. ¿Con qué comandos se crea un login en un proyecto?
 - h. ¿Cómo integrar Bootstrap a un proyecto en Laravel?
2. Con la información encontrada, la participante deberá realizar un mapa cognitivo de cajas con el formato que se presenta a continuación. Los recuadros superiores corresponderán a las primeras interrogantes enunciadas en cada pregunta, por ejemplo: *¿Qué es PHP* y qué características tiene? En cambio, los recuadros inferiores deberán ser completados con las segundas interrogantes (de haberlas), es decir: *¿Qué es PHP y qué características tiene?*
 3. Posteriormente, se dará paso a una reflexión a través de las siguientes preguntas guías:
 - a. ¿Piensas que será más fácil realizar una aplicación web con PHP y Laravel? ¿Por qué?
 - b. De acuerdo con tu experiencia utilizando Sails y con la información que encontraste acerca de PHP y



Laravel, ¿qué diferencias y semejanzas esperas encontrar?

Segunda parte: Vamos a presentarnos.

Sugerencia de tiempo invertido: 30 minutos.

Las participantes crearán su primer proyecto con Laravel. Para ello, se darán las siguientes instrucciones:

1. La participante creará un nuevo proyecto de Laravel con el nombre “MiPrimerProyecto”
2. Deberá actualizar las dependencias necesarias con Composer.
3. Visualizará el proyecto con algún editor de textos.
4. Creará un login para el proyecto utilizando los comandos investigados en la primera parte de la actividad.

Es importante que en este primer proyecto la participante se familiarice con el funcionamiento del framework.

Tercera parte: Poniendo en práctica.

Sugerencia de tiempo invertido: 3 horas.

Las participantes crearán una tienda en línea con Laravel. Para ello, se darán las siguientes instrucciones:

1. La participante creará un nuevo proyecto de Laravel con el nombre “MiTiendaEnPhp”
2. Deberá actualizar las dependencias necesarias con Composer.
3. Visualizará el proyecto con algún editor de textos.
4. Explorará las carpetas generadas al momento de crear un proyecto.
5. Creará un login para el proyecto utilizando los comandos investigados en la primera parte de la actividad.
6. Reescribirá el proyecto creado en el taller anterior.
7. Creará un *blade* para el encabezado y pie de página utilizando bootstrap.
8. El blade deberá ser implementado en cada una de las páginas creadas para la tienda virtual.
9. Levantará el servidor con el comando de artisan investigado.
10. Visualizará que la página no genere errores.
11. Comprobará que el sistema de login funcione.
12. Se le solicitará subir el proyecto a un nuevo repositorio en GitHub llamado “TiendaEnLaravel”

Una vez finalizadas las indicaciones anteriores, se realizará una recuperación oral (a través del diálogo) de la experiencia de la participante utilizando por primera vez PHP y creando su proyecto en Laravel. Para este ejercicio, se guiará el diálogo por medio de preguntas como:

1. ¿Habías trabajado anteriormente con PHP?
2. ¿Te pareció difícil?

3. ¿Qué fue lo que más se te dificultó?
4. ¿Qué fue lo que más se te facilitó?
5. ¿Habías trabajado anteriormente con Laravel?
6. Recuperando la reflexión comparativa entre Sails y Laravel en la primera parte, ¿conducen las diferencias y las semejanzas que esperabas encontrar?

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- Para responder las preguntas de la primera parte, se motivará el uso de la documentación de PHP y Laravel.

Enlaces para aprender más

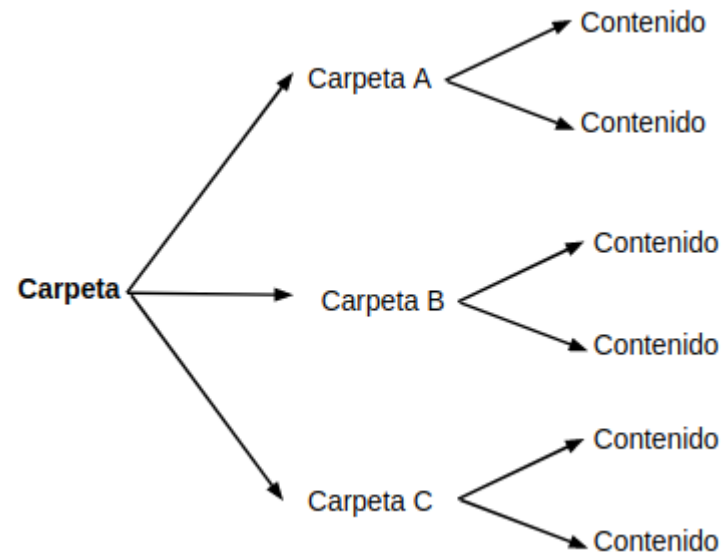
- Bitech Studio.(2016). *¿Qué es PHP? bien explicado*. [video]
<https://www.youtube.com/watch?v=ykGRYEX0n60>
- HTML Rules. (2019). *Qué es PHP*. [video]
<https://www.youtube.com/watch?v=w6Br0008Woc&t=2s>
- Kiko Palomares. (2019). *¿QUÉ es LARAVEL y para qué sirve?- Frameworks de PHP*. [video]
<https://www.youtube.com/watch?v=l7X3R57vdlk&t=209s>
- Laravel.(Sin fecha). *The PHP Framework for Web Artisans. Documentation*
<https://laravel.com/>
- Palacios, D.(Sin fecha). Curso de Laravel desde cero.
<https://styde.net/laravel-5/>

Actividad 10: Anatomía de una aplicación en PHP

Aprendizaje esperado: Enumerar los elementos de PHP generados y su estructura con Lavarel.		Duración de la actividad: 4 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Paquete de traducciones en español para Laravel. 2. Hojas de papel o papel reciclado. 3. Lápices, plumas o plumones. 	Evidencia: <ul style="list-style-type: none"> • Diagrama de árbol. • Proyecto <i>MiTiendaEnPhp</i> traducida al español. 	Retroalimentación: <ul style="list-style-type: none"> • El diagrama de árbol deberá conjuntar la información correspondiente a las preguntas de la primera parte de esta actividad. • Será necesario que esté elaborado de acuerdo al formato que se especifica. • Deberá estar almacenado en la carpeta de evidencias de este taller. En caso de haberse realizado de forma manual, será necesario digitalizar el documento para guardarlo en la carpeta de evidencias. • El proyecto <i>MiTiendaEnPhp</i> deberá estar en español, y ejecutar commit en el repositorio <i>TiendaEnLaravel</i> del proyecto en GitHub.
Desarrollo de la actividad:		
<p>Primera parte: Conociendo más de Laravel.</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>Las participantes tendrán un acercamiento al framework Laravel.</p> <ol style="list-style-type: none"> 1. Mediante la exploración del sistema de archivos de su nuevo proyecto creado en Laravel, o en todo caso la búsqueda de información, se solicitará a la participante que responda las siguientes interrogantes: <ol style="list-style-type: none"> a. ¿Qué archivos se generan por default cuando se crea un proyecto nuevo en Laravel? b. ¿Qué contiene la carpeta database? 		

- c. ¿Qué contiene la carpeta bootstrap?
- d. ¿Qué contiene la carpeta app?
- e. ¿Qué contiene la carpeta resources?
- f. ¿Qué contiene la carpeta routes?
- g. ¿Qué contiene la carpeta config?
- h. ¿Qué contiene la carpeta app/Http/Controllers?
- i. ¿Qué es la internacionalización i18n?
- j. ¿Qué hay en la carpeta resources/views?
- k. ¿Cómo configurar un proyecto en Laravel para que sea multilinguaje?
- l. ¿En qué carpeta se deben guardar las vistas correspondientes al front-end de la aplicación?

2. Con la respuestas a las preguntas, la participante deberá realizar un diagrama de árbol con el siguiente formato:



Segunda parte: En español por favor.

Sugerencia de tiempo invertido: 3 horas.

Las participantes traducirán su proyecto al idioma español. Para ello, se darán las siguientes instrucciones:

1. La participante deberá trabajar con el mismo proyecto creado en la actividad pasada *MiTiendaPhp*.
2. Se le solicitará la búsqueda de un paquete de la comunidad de Laravel en español que proporcione los archivos de idioma que se encuentran en la ruta `resources/lang` en español.
3. El paquete descargado deberá incluirse en la ruta **resources/lang**
4. Se solicitará cambiar lo necesario en el archivo **app.php**, que se encuentra en la ruta **config/app.php**, para que el sistema esté en español.
5. Se indicará utilizar la función **@lang** para realizar las traducciones de los mensajes generados en inglés, en las vistas del login generado en la actividad anterior.
6. La participante modificará los archivos que se encuentran en la ruta **config/app.php/en** para generar la traducción de las llaves que se modificaron en el punto anterior con la etiqueta **@lang**.
7. La participante colocará todas las traducciones para las llaves modificadas.
8. Guardará los cambios en los archivos.
9. Levantará el servidor.
10. Visualizará que la página funcione y que se realizó la traducción.
11. Si no hubiesen errores, la participante hará commit de su proyecto en GitHub al repositorio *TiendaEnLaravel*.

Nota: En caso de ser necesario, se motivará que la participante consulte la documentación oficial para obtener información de las funciones a utilizar.

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- Para responder las preguntas de la primera parte con respecto, se motivará el uso de la documentación de PHP y Laravel.

Enlaces para aprender más:

- Bitech Studio.(2016). *¿Qué es PHP? bien explicado*. [video]
<https://www.youtube.com/watch?v=ykGRYEX0n60>
- HTML Rules. (2018). *Qué es PHP*. [video]
<https://www.youtube.com/watch?v=w6Br00O8Woc&t=2s>
- Palomares, k. (2019) *¿Qué es Laravel y para qué sirve?- Frameworks de PHP*. [video]
<https://www.youtube.com/watch?v=l7X3R57vdlk&t=209s>
- Laravel. (Sin fecha). *The PHP Framework for Web Artisans. Documentación*

<https://laravel.com/>

- Palacios, D.(2015). Traducciones al módulo de login y registro en Laravel 5.1
<https://styde.net/traduccion-al-modulo-de-login-y-registro-en-laravel-5-1/>
- Why GitHub? (Sin fecha). Laraveles/ spanish. *Repositorio de idioma de Laravel 5*
<https://github.com/Laraveles/spanish>
- Palacios, D.(Sin fecha). Documentación de Laravel 6 en español.
<https://documentacion-laravel.com/>
- Curso de Laravel desde cero.
<https://styde.net/laravel-5/>

Actividad 11: Conectando con la base de datos.

Aprendizaje esperado: Conectar Laravel con base de datos SQLite.		Duración de la actividad: 3 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Diagrama entidad-relación de la actividad MWT3A3_Anexos. 2. Hojas de papel o papel reciclado. 3. Lápices, plumas o plumones. 	<p>Evidencia:</p> <ul style="list-style-type: none"> • Organizador gráfico. <p>Productos:</p> <ul style="list-style-type: none"> • Base de datos en SQLite llamada <i>tiendaLaravel</i>. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • El organizador gráfico deberá contener la información solicitada en las seis preguntas de la segunda parte de la actividad. • El organizador gráfico deberá estar almacenado en la carpeta de evidencias de este taller. En caso de haberse realizado de forma manual, será necesario digitalizar el documento para guardarlo en la carpeta. <p>Evaluación:</p> <ul style="list-style-type: none"> • La base de datos deberá crearse utilizando los comandos: composer dump-autoload php artisan view:clear php artisan cache:clear php artisan migrate
Desarrollo de la actividad:		
<p>Primera parte: Desarrollo de diagrama E-R</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>La participante deberá crear un diagrama entidad-relación para entender las relaciones entre los objetos o entidades de su tienda en línea (se recomienda utilizar el anexo <i>MWT3A3_Anexos</i>). Para ello la participante:</p> <ol style="list-style-type: none"> 1. Listará los objetos o entidades que contiene su tienda en línea. 2. Responderá las siguientes interrogantes con las cantidades mínimas y máximas solicitadas. Estas deberán estar representadas entre paréntesis, separadas por medio de dos puntos (mínimo:máximo). Ejemplo: (1:10). De tratarse de una 		

cantidad máxima irrepresentable, es posible utilizar el símbolo asterisco. Ejemplo: (1:*)).

- a. Un **producto**, ¿en cuántas **listas de productos** puede aparecer? Respuesta= (min:max)
 - b. Una **lista de productos**, ¿cuántos **productos** puede contener? Respuesta= (min:max)
 - c. Un **producto**, ¿cuántas veces puede solicitarse en un **pedido**? Respuesta= (min:max)
 - d. En un **pedido**, ¿cuántos **productos** se pueden solicitar? Respuesta= (min:max)
 - e. Un **pedido** ¿cuántas veces se ingresa en una **lista de pedidos**? Respuesta= (min:max)
 - f. En una **lista de pedidos** ¿cuántos **pedidos** se solicitan? Respuesta= (min:max)
3. Las preguntas anteriores se agrupan en tres pares, ya que cada par corresponde a una misma relación de entidades. Se indicará a la participante que identifique los tres pares de preguntas, así como las dos entidades que conforman cada relación, y el verbo que las conecta (se recomienda utilizar la plantilla *MWT3A3_Anexos*).
4. Finalmente, en la parte la plantilla 4 de los anexos *MWT3A3_Anexos* la participante deberá:
- a. Unir con una línea los objetos que tengan una relación entre sí.
 - b. Anotar el mínimo y el máximo que respondieron en las preguntas en los paréntesis que correspondan de acuerdo a su lógica.
 - c. Anotar en el rombo los verbos que relacionan a ambos objetos.

Segunda parte: Conexión a la base de datos.

Sugerencia de tiempo invertido: 1 horas.

La participante realizará la conexión a una base de datos creada con SQLite para el proyecto creado desde la actividad 9 *MiTiendaEnPhp*. Para ello, se solicitará a la participante retomar el diagrama entidad-relación elaborado en la Actividad 3 de este taller. En caso de haber alterado la estructura de la tienda en línea (haber aumentado o disminuido la cantidad de objetos o entidades) a lo largo de las actividades hasta la presente, será necesario que complemente dicho diagrama con los nuevos elementos, o de ser necesario, elaborarlo nuevamente. Habiendo retomado el diagrama, y teniendo bien definidas las relaciones de entidades en su tienda, se indicará a la participante seguir la secuencia:

Nota: Es importante no ayudar a la participante a resolver los siguientes pasos. En su lugar, se motivará a buscar las soluciones en internet.

1. Creará una base de datos en SQLite llamada “tiendaLaravel”.
2. Colocará una contraseña a la base de datos.
3. Modificará el archivo para realizar la configuración de la BD en el proyecto *MiTiendaEnPhp*, es necesario realizar la configuración con el nombre de la base creada y la contraseña de los dos puntos anteriores.
4. Ejecutará los comandos necesarios para realizar la conexión de la base con el proyecto, se recomienda limpiar la caché del

proyecto.

5. Levantará el servidor.
6. Visualizará el proyecto en el navegador.
7. Registrará usuarias en el sistema de login creado en la actividad anterior.
8. Verificará que los usuarios se hayan guardado en la base de datos.
9. Si no hubiesen errores, la participante hará commit de su proyecto en github al repositorio *TiendaEnLaravel*.

Tercera parte: ¿Cómo le hice?

Sugerencia de tiempo invertido: 1 hora.

La participante entenderá el manejo de una base datos a través de Laravel con ayuda de un ORM.

1. Con la información investigada para poder realizar la actividad anterior y con base en su experiencia, responderá las siguientes preguntas en cualquier tipo de organizador gráfico que prefiera (deberá ser subido a la carpeta de evidencias de este taller):
 - a. ¿Qué es una base de datos relacional?
 - b. ¿Qué archivos deben modificarse para configurar una base de datos diferente?
 - c. ¿Cuáles son los comandos para conectar una base de datos con un proyecto en Laravel?
 - d. ¿Qué es un ORM y cuales son sus ventajas?
 - e. ¿Cuál es el ORM que utiliza Laravel?
 - f. ¿Cómo se crean las relaciones de una base de datos en Laravel?
2. Una vez reunida la información, se llevará a cabo una recuperación oral de la experiencia de la participante al crear la base de datos y conectarla a su tienda en línea sin ayuda. Se dialogará acerca de los problemas a los que se enfrentó, las cosas que más se le dificultaron y cómo las solucionó. Es imprescindible que durante o al finalizar el diálogo, se resalte la importancia de la autonomía en el proceso de aprendizaje, y se motive a la participante para tomar decisiones por cuenta propia.

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.
- Para responder las preguntas de la primera parte con respecto, se motivará el uso de la documentación de PHP y Laravel.

Enlaces para aprender más:

- Laravel.(Sin fecha).*The PHP Framework for Web Artisans.Documentación.*
<https://laravel.com/>
 - Database: Getting Started.
<https://laravel.com/docs/7.x/database>
 - Eloquente: Getting Started. <https://laravel.com/docs/7.x/eloquent>
- Palacios, D.(Sin fecha). Primeros pasos con Laravel 6.
<https://styde.net/laravel-6/>
 - Curso de Laravel desde cero. <https://styde.net/laravel-5/>

Actividad 12: Creando un API

Aprendizaje esperado: Crear API para la manipulación de Modelos.		Duración de la actividad: 5 horas y 30 minutos.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Material investigado de fuentes fidedignas. 2. Proyecto de tienda en línea. 	<p>Evidencia: Preguntas del apartado De modelos y migraciones y del apartado Controlando peticiones y agregando las rutas.</p> <p>Producto:</p> <ul style="list-style-type: none"> • A partir de la segunda parte de la actividad 12 agregar los recursos necesarios para poder agregar Productos con peticiones HTTP 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • Verificar que las respuestas dadas en la parte 1 de la actividad correspondan al tema de APIs. <p>Evaluación:</p> <ul style="list-style-type: none"> • Llevar a cabo la “Parte tres: Result OK” de la actividad MWT3A12.
Desarrollo de la actividad:		
<p>Primera parte: ¿Y cómo se hace una API en Laravel?</p> <p>Sugerencia de tiempo invertido: 1 hora y 30 minutos.</p> <p>La participante realizará una búsqueda en la web para encontrar material de apoyo que le ayude a comprender cómo funciona la implementación de modelos en Laravel. Se recomienda contestar las siguientes preguntas que le ayudarán a conducir su investigación:</p> <p>De modelos y migraciones</p> <ul style="list-style-type: none"> • ¿Qué es y para qué sirve un modelo? • ¿Qué es una migración? • ¿Cuál es la relación entre un modelo y una migración? • ¿Cuál es la convención para el nombre de modelos y tablas en Laravel? • ¿Por qué es importante tener en mente las convenciones? • ¿Es posible no seguir las convenciones y qué podría implicar? • ¿Cuáles son las columnas predeterminadas que Laravel agrega a todas tus tablas? • ¿En qué parte del código puedo editar la estructura de mis tablas? • ¿Cuál es el comando para crear un modelo? 		

- ¿Cuál es el comando para crear una migración?
- ¿Existe una forma de realizar las dos acciones anteriores con un solo comando?
- ¿Cuál es el comando para realizar una migración?
- ¿A través de qué métodos es posible indicar relaciones entre modelos y cuál es el proceso para indicarlo en el código?

Controlando peticiones y agregando las rutas

- ¿Cuál es el rol del controlador?
- ¿Cuál es el comando para crear un controlador?
- Completa la siguiente la tabla:

Verbo HTTP	Método en el Controller
GET	
POST	
UPDATE	
DELETE	

- Existe una forma de crear un controlador con todos los métodos ya definidos. ¿Cómo?
- ¿Con qué comando es posible ver todas las rutas de mi proyecto?
- ¿Qué archivo se debe modificar para agregar una nueva ruta?

Segunda parte: ¡Manos a la obra!

Sugerencia de tiempo invertido: 3 horas.

Con la información recabada en la primera parte de la actividad, la participante realizará las modificaciones necesarias a su proyecto de la tienda en línea para:

- Crear los modelos Producto, Compra y Carrito así como las tablas asociadas a estos. Se deberá cumplir con las siguientes características:
 - Producto contendrá los atributos: id, nombre, precio, descripción.

- Carrito contendrá el atributo id y tendrá una relación 1:n con Producto.
 - Compra contendrá el atributo id y tendrá una relación 1:1 con Carrito.
- Crear los controladores para los modelos anteriormente mencionados, se deberá agregar funcionalidad para los métodos HTTP: GET, POST, UPDATE Y DELETE. A través de estos métodos se deberá realizar la modificación pertinente en la base de datos.
- Añadir las rutas para poder hacerle peticiones a los controladores.

Tercera parte: Result OK.

Sugerencia de tiempo invertido: 1 hora.

Una vez que la participante ha realizado los cambios pertinentes a su proyecto será posible verificar su buen funcionamiento de la siguiente forma:

1. Primero revisará que las rutas se han configurado correctamente para cada uno de los métodos HTTP en cada uno de sus modelos. Para esto listará todas las rutas del proyecto con el comando que investigó en la primera parte.
2. Una vez que identificó como están conformadas sus rutas, hará uso de una herramienta de terceros para realizar peticiones HTTP y observar si su código funciona de forma correcta. Las herramientas de terceros varían de navegador en navegador pero son fáciles de usar, para Firefox existe el complemento RESTED y para el navegador Chrome Postman.

Actividad 13: Rutas hacia los productos

Aprendizaje esperado: Agregar vistas asociadas al punto de venta.		Duración de la actividad: 10 horas.
Recursos	Evidencia/Producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Material investigado de fuentes fidedignas. 2. Proyecto de tienda en línea. 	<p>Evidencia:</p> <ul style="list-style-type: none"> • Manual de directivas. <p>Producto:</p> <ul style="list-style-type: none"> • Modificación de la Tienda en línea partiendo del producto de la MWT3A12. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> • Asegurarse de la veracidad del material hallado, verificando que la información provenga de sitios oficiales de frameworks como el de Laravel. • En caso de ser necesario, reforzar los contenidos del Manual de directivas generado por la participante. <p>Evaluación:</p> <ul style="list-style-type: none"> • Llevar a cabo las acciones mencionadas en la “Cuarta parte: Poniendo todo en marcha”.
Desarrollo de la actividad:		
<p>Primera parte: MVC ¿y la V?</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <ol style="list-style-type: none"> 1. La participante buscará material de apoyo que le ayudará a conocer la utilización de vistas en Laravel y el motor de plantillas Blade. Centrándose en tratar de dar respuesta a los siguientes puntos: <ol style="list-style-type: none"> a. ¿Qué es Blade y cómo ayuda en nuestro proyecto? b. ¿Cómo Blade contribuye a la reutilización de código? c. Directivas de Blade. 2. Con ayuda del material de apoyo, pondrá en práctica las directivas básicas de Blade para elaborar un manual en el que se incluirán capturas que muestren el código, el resultado de las directivas en funcionamiento y una explicación en sus propias palabras de lo que hace la directiva. Se espera que se incluyan las directivas para definir y heredar un Layout, 		

desplegar variables así como las diferentes estructuras de control.

Nota: La participante podrá apoyarse de inspeccionar/inspect en el navegador para identificar la estructura y entender mejor cómo se integran las vistas.

Segunda parte: Construyendo las paradas.

Sugerencia de tiempo invertido: 1 hora.

La participante realizará los cambios pertinentes a su proyecto para añadir vistas que ayuden a desplegar los datos obtenidos de los modelos, haciendo énfasis en la reutilización de código que provee el gestor de plantillas Blade. Las vistas básicas necesarias para el proyecto se enumeran a continuación:

- Listar productos: Permite desplegar todos los productos disponibles de la tienda.
Se deberá hacer uso de Cards provistas por Bootstrap para mejorar la estética de la aplicación. En cada Card aparte de desplegarse la información del Producto será necesario añadir un botón que permita al usuario agregar un producto a su carrito de compras.
Deberá ser clara la inclusión de @foreach para iterar sobre los datos y mostrarlos en la vista.
- Agregar producto: Consistirá en un formulario, en el que se pedirán los campos nombre, precio y descripción.
- Actualizar / eliminar producto: En esta vista se mostrarán los nombres de los productos en forma de lista junto a dos botones, uno para actualizar y otro para eliminar. El botón actualizar deberá conducir a otra vista para la actualización.
- Actualizar producto: En esta vista deberá ser posible visualizar los datos anteriores del producto y modificarlos a través de un formulario.
- Mostrar carrito: Listará los productos que se hayan agregado al carrito en forma de lista cada uno con su respectivo botón para eliminarlo del carrito.

Adicionalmente contendrá en la parte inferior el cálculo del total y un botón con la leyenda “Comprar”.

Nota: La reutilización de código deberá ser clara para los elementos que comparten características, algunas recomendaciones son: header y footer así como para las 2 vistas que involucran listar los productos.

Recordar también incluir el icono del carrito de compras, que debe ser visible desde cualquier ubicación del sitio.

Tercera parte: Construyendo los caminos.

Sugerencia de tiempo invertido: 2 horas.

Si se tienen las vistas, ahora ¿cómo le decimos a Laravel cuándo mostrarlas? Esto sería similar a tener las paradas de un tren pero sin las vías que nos lleven hacia ellas, se invita a la tallerista a reflexionar lo anterior con la participante. A partir del planteando

las siguientes preguntas:

- ¿Hasta ahora con qué elemento de Laravel hemos construido los “camino”?
- En la MWT3A12 se crearon algunos “camino” para la API de los diversos modelos, ¿estos camino son los que necesita mi tienda?
- Siguiendo con la analogía del tren, realizar un mapa de cómo se conectan las diversas vistas de la aplicación.

Con el mapa que se acaba de construir, eliminar las rutas a los modelos anteriormente agregadas y generar las nuevas en el archivo routes.php.

Cuarta parte: Poniendo todo en marcha.

Sugerencia de tiempo invertido: 5 horas.

La participante deberá agregar la lógica de negocio a su proyecto, para lo cual agregará los métodos necesarios en el controlador correspondiente a cada Modelo además de corregir las rutas para agregar o modificar la información necesaria. Los cambios esperados serán los siguientes:

- Agregar, actualizar y eliminar un producto a través de las nuevas vistas para que el resultado se vea reflejado en la base de datos
- Para el listado de productos y del carrito, realizar la consulta en la base de datos que retorne la información requerida y pasar esta información a la vista para su despliegue.
- Al presionar el botón “Comprar” guardar la información del carrito en la tabla compra.
- Agregar la función que calcule el total de los productos del carrito para su despliegue en la vista.

Actividad 14: Dándole seguridad

Aprendizaje esperado: Incorporar vistas de administrador y políticas de seguridad al sistema.		Duración de la actividad: 4 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
<ol style="list-style-type: none"> 1. Tablas de identificador y Rol de Usuario. 2. Material necesario para elaborar los diagramas de secuencia (hojas de papel y marcadores, app móvil, programa en la computadora, etc. 	<p>Evidencia:</p> <ul style="list-style-type: none"> ● Tablas de Identificador y Rol de Usuario. ● Diagramas de secuencia de Roles de Usuario y Proceso de Login y Registro. <p>Producto:</p> <ul style="list-style-type: none"> ● A partir de la segunda parte de la actividad seis, se deberá crear el modelo Usuario, así como el controlador y sus correspondientes vistas para poder realizar acciones CRUD. ● A partir de la segunda parte de la actividad siete, se deberán integrar las directivas de restricción de vistas y acciones basadas en perfiles de usuario. 	<p>Retroalimentación:</p> <ul style="list-style-type: none"> ● Asegurarse que se comprendieron los conceptos básicos dentro del Rol de Usuario y el Proceso de Login y Registro, partiendo de la reorganización de la información proporcionada en los diferentes materiales de la actividad. <p>Evaluación:</p> <ul style="list-style-type: none"> ● Se deberá comprobar a través de las vistas que las actualizaciones al modelo Usuario se registren en la BD. ● Se deberá revisar que no sólo las vistas se restrinjan, las acciones CRUD también deberán estar basadas en permisos y roles.
Desarrollo de la actividad:		
<p>Primera parte: Regístrate para comenzar.</p> <p>Sugerencia de tiempo invertido: 1 hora.</p> <p>La participante reflexionará sobre la importancia de la creación de usuarios, creación de roles, así como la interacción de estos con su sistema. Para lo cual se recomienda visitar por lo menos tres de los sitios sugeridos con el fin de realizar lo siguiente:</p> <ul style="list-style-type: none"> ● ¿Cuál es el primer paso que se debe realizar para poder trabajar o hacer uso de las plataformas? ● ¿Por qué crees que es importante este paso? ● ¿En qué otro lugar has visto el uso de mecanismos de autenticación? 		

- ¿Es importante para una tienda en línea saber quien realizó una compra?
- ¿Consideras que a tu proyecto de la tienda en línea le hace falta un sistema de autenticación? (De no estar segura de esta respuesta, es posible visitar tiendas en línea ya puestas en marcha)
- Identificar las acciones que pueden realizar los usuarios dependiendo de su rol para completar una tabla como la que se muestra a continuación con Sí/No:

Identificador	Acción
1	Visualizar los productos disponibles en la tienda
2	Agregar y eliminar productos al carrito de compras
3	Comprar los productos contenidos en el carrito de compras
4	Agregar, actualizar y eliminar información de los productos

Rol de usuario	Acción			
	1	2	3	4
Usuario no registrado				
Usuario registrado				
Administrador				

Nota: Con el fin de completar esta tabla la participante podrá visitar tiendas en línea que ya se encuentren en la web para indagar en su funcionamiento.

De considerarse necesario se podrán agregar más acciones.

Una vez que se han identificado los roles de usuarios y las acciones que pueden realizar, deberá generar un diagrama de secuencia (UML) para cada una de las acciones indicadas en la sección pasada. Si los usuarios tienen actividades en común solo se deberá realizar un diagrama.

Generará también un diagrama de secuencia para el proceso de login y registro

Nota: Los diagramas deberán incluir el método del controlador, las vistas y modelos afectados.

La participante realizará las modificaciones necesarias al diagrama E-R generado en MWT3A11 para incluir la relación entre las nuevas entidades Usuario y Rol, así como para relacionar la Compra de un Carrito con un Usuario.

Segunda parte: ¡Alto! Identifíquese.

Inversión de Sugerencia de tiempo invertido: 3 horas.

La participante generará las modificaciones correspondientes para incluir el módulo de autenticación de Laravel. También realizará los cambios necesarios para incluir el nombre de usuario en el modelo Usuario.

Nota: La participante podrá realizar cambios a las vistas a fin de mejorar la estética y adaptarlas mejor a su proyecto.

La participante se basará en los diagramas de secuencia y en el nuevo modelo entidad relación generados en la primera parte de la actividad para incorporar los cambios correspondientes a su proyecto a fin de:

1. Implementar los nuevos modelos y sus relaciones.
2. Incorporar las acciones planteadas en los diagramas de secuencia.
3. Restringir las vistas correspondientes dependiendo el rol del usuario.

Actividad 15: Ética de los datos

Aprendizaje esperado: Reflexionar sobre aspectos legales, de conformado de información, seguridad de la información y privacidad.		Duración de la actividad: 5 horas.
Recursos	Evidencia/producto	Retroalimentación/Evaluación
1. Plantilla del anexo disponibles en <i>MWT3A15_Anexos.pdf</i>	Evidencia: <ul style="list-style-type: none"> 11 fichas de consulta (mínimo). 	Retroalimentación: <ul style="list-style-type: none"> Las tarjetas de consulta así como los cuadros sinópticos, deberán responder a cada una de las preguntas de la primera parte de esta actividad. Las fichas deberán tener escrita la pregunta por el anverso, y en el reverso, la respuesta a tal pregunta correspondiente.
Desarrollo de la actividad:		
<p>Primera parte: ¡Los datos!</p> <p>Sugerencia de tiempo invertido: 2 horas.</p> <p>A partir de la metodología de <i>aprendizaje basado en problemas</i>, las participantes realizarán una profunda reflexión acerca de las implicaciones del manejo de información personal, la ética, privacidad y seguridad alrededor de este importante tema.</p> <ol style="list-style-type: none"> Se le presentará a la participante una breve historia que narre la situación donde los datos de su tarjeta de crédito de alguna cliente de su tienda en línea se vean comprometidos (se recomienda utilizar los anexos <i>MWT3A15_Anexos.pdf</i>). De esta forma se expondrá la importancia del manejo responsable de tales datos como entidad particular. Posteriormente de presentar la situación, la participante deberá responder algunas preguntas guías para reflexionar acerca de la importancia de asegurar la privacidad de la información personal de las y los clientes: <ol style="list-style-type: none"> ¿Qué medidas de seguridad tiene hasta el momento tu tienda en línea para garantizar que los datos de la tarjeta de crédito de la cliente no fueron expuestos o vulnerados realmente por tu página? ¿Crees que sea necesario tenerlas? ¿Por qué? ¿Por qué crees que es importante garantizar la privacidad de los datos de los clientes? ¿Qué ventajas representaría para tu tienda en línea garantizar la seguridad y privacidad de los datos de las y los 		

clientes?

Segunda parte: Extracción de datos ¿Mito o realidad?

Sugerencia de tiempo invertido: 1 hora.

Se solicitará a la participante leer al menos tres notas periodísticas o artículos de revistas, que traten sobre algunos casos reales en donde los datos de las o los usuarios han sido comprometidos, debido a que la base de datos de la empresa que los poseía fue vulnerada. Se recomienda utilizar los links proporcionados en la sección de “enlaces para aprender más” y en caso de no hacer uso de estos, la participante llevará a cabo la búsqueda de información, considerando que las lecturas deben visualizar lo común y peligroso que resulta ser el robo de información.

A continuación, deberá llevar a cabo una reflexión basada en la información que leyó y responderá a las siguientes interrogantes:

1. ¿Conocías alguno de estos casos de extracción de datos?
2. ¿Antes de las lecturas considerabas que tus datos se encontraban seguros cuando los proporcionabas? ¿Sigues pensando lo mismo?
3. ¿Cuáles son los datos que generalmente son más vulnerados?
4. ¿Para qué crees que se utilizan esos datos?
5. ¿Qué tan importante consideras que es la seguridad de los datos en un sistema web? ¿Por qué?

Tercera parte: Decálogo.

Sugerencia de tiempo invertido: 2 horas.

1. Se solicitará a la participante buscar documentos de organismos internacionales donde se presenten recomendaciones o directrices para el manejo y la privacidad de datos e información personal en internet (se recomienda utilizar los links para aprender más).
2. Posteriormente, con dicha documentación y retomando los ejercicios de la primera y la segunda parte de la actividad, la participante deberá realizar un decálogo con los diez pasos, reglas o principios que seguirá su tienda en línea para garantizar la privacidad de los datos e información de los clientes.
3. Una vez realizado su decálogo, deberá explicarlo oralmente, de tal forma que exprese el camino que recorrió su reflexión desde la primera parte de esta actividad hasta llegar a dicho decálogo.

Notas para apoyar la actividad:

- Se asesorará en todo momento la búsqueda de información de la participante, y se realizarán intervenciones teóricas para aclarar dudas, confusiones o concretar la información.

Links para aprender más:

- Contreras, M. (2014) . *eBay hackeado, piden que todos los usuarios cambien contraseñas.*
 . <https://www.fayerwayer.com/2014/05/ebay-hackeado/> (Enero 2021)
- Mendiola, J. (2016). *Dropbox reconoce el 'hackeo' de 60 millones de cuentas: cómo saber si la tuya está afectada.*
https://elpais.com/tecnologia/2016/08/31/actualidad/1472642567_500051.html
- Rodríguez, P. (Sin fecha). *Hay empresas que tienen extensos informes con tus datos personales recopilados en Internet y los venden por más de cien euros.*
<https://www.xataka.com/privacidad/hay-empresas-que-tiene-extensos-informes-tus-datos-personales-recopilados-internet-venden-cien-euros>
- Kerr, Dara. (2014). *'Hackeo' a Home Depot puso en riesgo 56 millones de tarjetas.*
<https://www.cnet.com/es/noticias/robo-de-informacion-en-home-depot-afecta-a-56-millones-de-tarjetas/>
- El país. (2019). *Marriot recibe una multa de 110 millones por el robo de datos de clientes.*
https://elpais.com/economia/2019/07/10/actualidad/1562756692_805212.html



Escuela de Código para PILARES Descripción de actividades - Parte 2: Módulo Elaboración de sistemas web (MW) por Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México se distribuye bajo una [Licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-sa/4.0/).